UNIVERSIDADE DE CAXIAS DO SUL

ÁREA DO CONHECIMENTO DE CIÊNCIAS DA VIDA

INSTITUTO DE BIOTECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM BIOTECNOLOGIA

Workflow Científico de Anotação Genômica Funcional e Curadoria Manual de Genomas

Eduardo Balbinot

CAXIAS DO SUL

2020

Eduardo Balbinot

Workflow Científico de Anotação Genômica Funcional e Curadoria Manual de Genomas

Dissertação apresentada ao Programa de Pós-graduação em Biotecnologia da Universidade de Caxias do Sul, visando a obtenção de grau de Mestre em Biotecnologia.

Orientador: Prof. Dr. Aldo José Pinheiro Dillon
Coorientador: Profa. Dra. Scheila de Avila e Silva

CAXIAS DO SUL
2020

Dados Internacionais de Catalogação na Publicação (CIP)
Universidade de Caxias do Sul
Sistema de Bibliotecas UCS - Processamento Técnico

B172w Balbinot, Eduardo

Workflow científico de anotação genômica funcional e curadoria manual de genomas [recurso eletrônico] / Eduardo Balbinot. – 2020.

Dados eletrônicos.

Dissertação (Mestrado) - Universidade de Caxias do Sul, Programa de Pós-Graduação em Biotecnologia, 2020.

Orientação: Aldo José Pinheiro Dillon.

Coorientação: Scheila de Avila e Silva.

Modo de acesso: World Wide Web

Disponível em: https://repositorio.ucs.br

1. Bioinformática. 2. Biotecnologia. 3. Genoma. 4. Fluxo de trabalho. I. Dillon, Aldo José Pinheiro, orient. II. Silva, Scheila de Avila e, coorient. III. Título.

CDU 2. ed.: 57:004

Catalogação na fonte elaborada pela(o) bibliotecária(o)
Ana Guimarães Pereira - CRB 10/1460

Eduardo Balbinot

Workflow Científico de Anotação Genômica Funcional e Curadoria Manual de Genomas

Dissertação apresentada ao Programa de Pós-graduação em Biotecnologia da Universidade de Caxias do Sul, visando a obtenção do título de Mestre em Biotecnologia.

Orientador: Prof. Dr. Aldo José Pinheiro Dillon

Co-orientadora: Profa. Dra. Scheila de Avila e Silva

DISSERTAÇÃO APROVADA EM 17 DE DEZEMBRO DE 2020.

_____

Orientador: Prof. Dr. Aldo José Pinheiro Dillon

_____

Co-orientadora: Profa. Dra. Scheila de Avila e Silva

_____

Prof. Dr. Flávio Horita

_____

Prof. Dr. Frederico Kremer

_____

Profa. Dra. Marli Camassola

AGRADECIMENTOS

À Deus, por me conceder sabedoria e saúde para seguir sempre em frente e de cabeça erguida. Obrigado por ser minha força e meu guia em todas as etapas da minha vida.

Aos meus pais, Vanius Balbinot e Dolores Kuhn Balbinot, que sempre me deram apoio, mesmo nos momentos mais difíceis. Obrigado por acreditarem nos meus sonhos e me mostrarem que seria possível realizá-los. Devo tudo o que sou hoje a vocês. Amo vocês com amor eterno!

À minha namorada, Renata Parisotto, que acompanhou de perto as dificuldades encontradas nesta jornada e esteve, pacientemente, ao meu lado com um cafezinho, uma palavra de apoio e um sorriso. Obrigado, também, pelo seu olho clínico ao me auxiliar na revisão da escrita. Sem você, nada disso seria possível. Amo você!

Aos meus orientadores, Dra. Scheila de Avila e Silva e Dr. Aldo José Pinheiro Dillon, pelos ensinamentos e pela disponibilidade em acompanhar e orientar o desenvolvimento deste trabalho. O entendimento da Biologia foi um desafio para mim desde o início e vocês, pacientemente, esclareceram conceitos e compartilharam seus conhecimentos, me ajudando a trilhar caminhos que eu não acreditava que seriam possíveis. Foi uma honra ter sito orientado por vocês. Muito obrigado!

A todos os meus colegas do Laboratório de Bioinformática e Biologia Computacional da Universidade de Caxias do Sul, que também compartilharam seus conhecimentos e, direta ou indiretamente, contribuíram para que eu chegasse até aqui. Um agradecimento especial ao colega Alexandre Rafael Lenz, que acompanhou diretamente o desenvolvimento da solução, sugerindo melhorias, validando os resultados e, principalmente, esclarecendo dúvidas relacionadas à Biologia. Você foi essencial, meu amigo, muito obrigado!

Agradecimentos também são direcionados à Universidade de Caxias do Sul (UCS), pela oportunidade de realização desta pesquisa e à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pelo financiamento desta pesquisa em forma de bolsa, modalidade Programa de Suporte à Pós-Graduação de Instituições Comunitárias de Educação Superior (PROSUC).

LISTA DE QUADROS

LISTA DE ILUSTRAÇÕES

LISTA DE ABREVIATURAS E SIGLAS

AAs ..... enzimas para atividades auxiliares
AJAX .... *Asynchronous Javascript and XML*
API ..... *Application Programming Interface*
BLAST ... *Basic Local Alignment Search Tool*
CAZymes enzimas ativas sobre carboidratos
CBMs .... módulos de ligação ao carboidrato
CDD ..... *Conserved Domains Database*
CEs ..... carboidrato esterases
COG ..... *Clusters of orthologous groups*
CSRF .... *Cross Site Request Forgery*
CSS3 ..... *Cascading Style Sheets*
DBG ..... grafos de Bruijn
DDBJ .... *DNA DataBank of Japan*
ddNTP's .. didesoxiribonucleotídeos trifosfato
DNA ..... ácido desoxirribonucleico
EMBL ... *European Molecular Biology Laboratory*
GAG ..... *Genome Annotation Generator*
GenSAS .. *Genome Sequence Annotation Server*
GHs ..... glicosil hidrolases
GO ..... *Gene Onthology*
GO FEAT . *Gene Ontology Functional Enrichment Annotation Tool*
GPI ..... Glicosilfosfatidilinositol
GTs ..... glicosiltransferases
HTML5 .. *HyperText Markup Language*
INSDC ... *International Nucleotide Sequence Database Collaboration*
JSON .... *JavaScript Object Notation*
KEGG .... *Kyoto Encyclopedia of Genes and Genomes*
LINEs .... elementos dispersos longos
mRNAs .. RNAs mensageiros
MTV ..... *Model-Template-View*
NCBI .... *National Center for Biotechnology Information*
NGS ..... *next-generation sequencing*
OLC ..... *overlap layout consensus*
PacBio ... *Pacific Biosciences*

PBKDF2 . *Password-Based Key Derivation Function 2*
PCR ..... reação em cadeia da polimerase
PIR ..... *Protein Information Resource*
PLs ..... polissacarídeo liases
PPR ..... *peptide pattern recognition*
RAST .... *Rapid Annotations using Subsystems Technology*
RASTtk .. *RAST tool kit*
rRNAs ... RNAs ribossômicos
SBS ..... sequenciamento por síntese
SFLD .... *Structure–Function Linkage Database*
SINEs .... elementos dispersos curtos
SMRT .... *Single Molecule Real-Time*
SOLid .... *Sequencing by Oligonucleotide Ligation and Detection*
SQL ..... *Structured query language*
SRA ..... *Sequence Read Archive*
TBL ..... *feature table format*
tRNAs .... RNAs transportadores
UniProtKB *UniProt Knowledge Base*
WGS ..... *whole genome sequencing*
XML ..... *Extensible Markup Language*

RESUMO

Com o advento das tecnologias de sequenciamento de nova geração, o sequenciamento de genomas não se apresentou mais como uma barreira tecnológica. Analisar a estrutura e atribuir um significado biológico para sequências genômicas e proteômicas *in silico* (anotação genômica), no entanto, se tornou a nova tarefa desafiadora. A anotação funcional é uma etapa crucial neste processo e tem por intuito compreender os processos biológicos dos organismos e guiar novas pesquisas. Entretanto, trata-se de uma tarefa complexa, já que envolve a utilização de um grande número de bancos de dados, *web servers* e ferramentas para realizar comparações das sequências de interesse com outras sequências disponíveis em repositórios de domínios de proteínas. Cada ferramenta possui arquivos de saída independentes e em formatos específicos, dificultando a organização dos resultados de anotação em um único contexto e o trabalho colaborativo de múltiplos pesquisadores em um mesmo projeto. O objetivo desta pesquisa compreendeu o desenvolvimento de um *workflow* científico de anotação genômica funcional e curadoria manual de genomas, através da implementação de uma ferramenta *web* colaborativa que permite a execução automática de ferramentas de anotação funcional e integração dos resultados em uma plataforma unificada. A solução ainda permite a realização de curadoria manual de informações estruturais e funcionais para cada gene, além de suportar a exportação dos dados de anotação para os formatos exigidos no processo de submissão do banco de dados GenBank. O *worflow* foi testado e validado no processo de anotação genômica das linhagens selvagem (2HH) e mutante (S1M29) do fungo *Penicillium echinulatum*, conduzido pelo Laboratório de Bioinformática e Biologia Computacional da Universidade de Caxias do Sul e submetido ao GenBank através dos *BioProjects* PRJNA520890 e PRJNA521489, respectivamente. Além disso, uma versão experimental da ferramenta foi disponibilizada gratuitamente através do endereço https://seq2annot.org. A solução mostrou-se eficiente em função de abstrair a complexidade de execução de ferramentas externas e integrar os resultados de anotação em um ambiente colaborativo unificado. O mecanismo central de tarefas assíncronas permitiu a possibilidade de escalar horizontalmente os recursos de servidor à medida que ocorrer o aumento da demanda de trabalho. A decisão arquitetural de dividir trabalhos em pequenas unidades de trabalho independentes também garantiu que novas ferramentas de anotação funcional possam ser facilmente desenvolvidas e acopladas à aplicação, tornando-a uma solução promissora para ser constantemente aprimorada e utilizada em projetos de anotação genômica em larga escala no futuro.

Palavras-chave: Anotação genômica, anotação funcional, *workflow* de anotação, curadoria manual.

ABSTRACT

With the advent of new generation sequencing technologies, genome sequencing was no longer a scientific barrier. Analysing the structure and giving biological meaning to genomic and proteomic sequences *in silico* (genome annotation) became, nevertheless, the new challenging task. Functional annotation is a crucial step in this process and aims to understand organisms' biological processes and guide new research. However, it is a complex task, as it involves using a large set of databases, web servers, and tools to compare all the sequences of interest with other sequences available in protein domain repositories. Each of these tools has independent and format-specific output files, making it difficult to keep annotation data organized in a single-context environment and preventing researchers from working collaboratively within a project. This research aimed to develop a scientific workflow for functional annotation and manual curation of genomes by implementing a collaborative web application that supports the automatic execution of functional annotation tools and integrates results into a single-context platform. The solution also supports manual curation of structural and functional data for each gene and allows users to export annotation results to file formats required by GenBank database's submission process. The workflow was tested and validated in the annotation process of wild-type (2HH) and mutant (S1M29) strains of *Penicillium echinulatum*, conducted by Computational Biology and Bioinformatics Laboratory at University of Caxias do Sul and submitted to GenBank under *BioProjects* PRJNA520890 and PRJNA521489, respectively. Besides, an experimental version of the tool was made freely available at https://seq2annot.org. The solution has proven effective as it abstracts the complexity of the external tools' execution and integrates annotation results in a unified collaborative environment. The core mechanism of asynchronous task execution provided the ability to horizontally scale server resources as demand grows over time. The architectural decision of splitting jobs into single independent tasks also ensured that new functional annotation tools can be easily developed and plugged into the application, making it a promising tool to be continuously improved and used in large-scale genome annotation projects in the future.

Keywords: Genome annotation, functional annotation, annotation workflow, manual curation.

SUMÁRIO

1 INTRODUÇÃO

Desde a descoberta da estrutura da dupla-hélice por Watson e Crick (WATSON; CRICK *et al.*, 1953), inúmeros trabalhos vem descrevendo o importante papel biológico do ácido desoxirribonucleico (DNA). Isso possibilitou o desenvolvimento de tecnologias de sequenciamento gradativamente mais eficientes, apresentando variações em custo, velocidade de sequenciamento, tamanho de *reads* e acuracidade de resultados (GIANI *et al.*, 2019).

As tecnologias de sequenciamento de DNA existem desde o início da década de 1970. O tradicional método de *Sanger* (SANGER; NICKLEN; COULSON, 1977), utilizado no sequenciamento do primeiro genoma completo de um organismo vivo, *Haemophilus influenza* (FLEISCHMANN *et al.*, 1995), foi o principal entre os métodos pioneiros e serviu de base para a primeira geração de sequenciadores automatizados. Entretanto, o sequenciamento completo de genomas (*whole genome sequencing* (WGS)) através desta tecnologia ainda era inviável na maior parte dos casos, em função de seu alto custo e de ser um trabalho complexo e demorado (BESSER *et al.*, 2018). A necessidade de tecnologias com menores custos e maior velocidade de sequenciamento intensificou-se com o Projeto Genoma Humano (VENTER *et al.*, 2001), uma colaboração internacional de 3,8 bilhões de dólares que teve por objetivo sequenciar e interpretar os 3,2 bilhões de pares de nucleotídeos do genoma humano.

O avanço e a redução de custos do poder computacional, somados a novos métodos de processamento paralelo, proporcionaram uma grande mudança no início dos anos 2000, com o advento da nova geração de tecnologias de sequenciamento (*next-generation sequencing* (NGS)) (GIANI *et al.*, 2019), que abriu novas possibilidades de lidar com genomas maiores. O poder dos resultados da redução de custos e de tempo de sequenciamento foi provado pelo aumento exponencial de sequenciamentos e re-sequenciamentos de genomas completos. A eficiência deste método levou-o rapidamente a ser utilizado em práticas de laboratório e saúde pública, fazendo com que, atualmente, a genômica represente um papel crucial no campo das ciências da vida (BESSER *et al.*, 2018).

A grande quantidade de dados biológicos gerada pelo crescente número de sequenciamentos elevou o papel e a importância da bioinformática, em função da necessidade de armazenar, organizar a analisar em detalhe as sequências obtidas (SHAIK *et al.*, 2019a). Para o armazenamento, por exemplo, bancos de dados em níveis primário e secundário foram criados. Os primários, como o GenBank (BENSON *et al.*, 2018) e o *European Molecular Biology Laboratory* (EMBL) (KANZ *et al.*, 2005), são responsáveis por armazenar os dados experimentais e não tratados, como sequências de nucleotídeos ou de aminoácidos. Já os secundários, como o *Protein Information Resource* (PIR) (WU *et al.*, 2003), o InterPro (MITCHELL *et al.*, 2018) e o UniProt (CONSORTIUM, 2018), armazenam resultados de análises realizadas a partir das informações de bancos de dados primários, através do uso de *softwares* especializados ou de

anotações manuais (SHAIK *et al.*, 2019b).

O significado de uma sequência genômica ou proteica é determinado pelo processo denominado *anotação genômica*, que ainda se mostra uma tarefa desafiadora para biólogos e cientistas com pouco *expertise* em informática, já que envolve a utilização de uma grande quantidade de algoritmos e *softwares* específicos (WANG *et al.*, 2017). O processo de anotação genômica consiste em três etapas: 1) identificar regiões repetitivas das sequências de DNA; 2) realizar a predição e identificação de genes que codificam proteínas, RNAs transportadores (tRNAs) e RNAs ribossômicos (rRNAs), em um processo denominado anotação estrutural e 3) atribuir uma função e inferir um significado biológico para as proteínas que os genes codificam, em um processo denominado anotação funcional (HUMANN *et al.*, 2019).

A anotação funcional é um dos processos mais importantes na análise *in silico* do material genético. A quantidade de sequências depositadas nos bancos de dados demandam análise funcional, com o intuito de dar nomes aos produtos gênicos e, assim, compreender os processos biológicos dos organismos estudados e guiar novas pesquisas (MCDONNELL; STRASSER; TSANG, 2018). As ferramentas computacionais utilizadas neste processo fazem uso de algoritmos de busca de similaridade, tais como o *Basic Local Alignment Search Tool* (BLAST) (ALTSCHUL *et al.*, 1990) e o HMMER (HMMER, 2020). Estes e outros algoritmos similares realizam comparações das sequências que estão sendo estudadas com outras sequências depositadas em bancos de dados de proteínas ou de domínios de proteínas, com o objetivo de encontrar similaridades e, assim, atribuir funções e um significado biológico (CRUZ *et al.*, 2019). As abordagens utilizadas para tanto incluem a caracterização de domínios, famílias, buscas por grupos de ortólogos e detecção de similaridades com sequências já caracterizadas e revisadas (HARIDAS; SALAMOV; GRIGORIEV, 2018; ANGEL *et al.*, 2018). Alguns exemplos de ferramentas e bancos de dados utilizadas neste processo incluem *Gene Onthology* (GO) (ASHBURNER *et al.*, 2000; ACENCIO *et al.*, 2019), InterPro (MITCHELL *et al.*, 2018), UniProt (CONSORTIUM, 2019), Pfam (EL-GEBALI *et al.*, 2019), Prosite (SIGRIST *et al.*, 2012), SMART (LETUNIC; BORK, 2018), PRINTS (ATTWOOD *et al.*, 2012), SignalP (ARMENTEROS *et al.*, 2019b), TMHMM (KROGH *et al.*, 2001), OrthoDB (KRIVENTSEVA *et al.*, 2019), *Clusters of orthologous groups* (COG) (GALPERIN *et al.*, 2019), KEGG Orthology (KANEHISA *et al.*, 2016) e EggNOG (HUERTA-CEPAS *et al.*, 2016).

O processo de anotação funcional é complexo, uma vez que se observa a gama de bancos de dados e ferramentas a serem utilizados para enriquecer a anotação e garantir resultados precisos. É notável a consequente dificuldade de manter organizadas e agrupadas, em um mesmo contexto, as informações funcionais geradas por cada uma das ferramentas de anotação, já que cada um dos *softwares* a serem utilizados proporciona a possibilidade de exportação de resultados em formatos específicos, gerando arquivos de difícil interpretação e de visualização independente (HUMANN *et al.*, 2019; ARAUJO *et al.*, 2018). Para a análise de múltiplas sequências, inclusive, algumas das ferramentas existentes exigem instalação local de *softwares*,

dependências e bibliotecas, incluindo a ausência total de interface visual com o usuário e a necessidade de execução por linha de comando, o que não permite um ambiente colaborativo entre pesquisadores e pode exigir conhecimentos elevados de informática por parte do usuário.

A necessidade de um ambiente integrado e unificado se intensifica no processo de curadoria manual, que consiste em uma revisão cuidadosa das informações anotadas, exigindo esforço e tempo por parte do pesquisador, na tentativa de buscar por possíveis inconsistências na anotação funcional (MCDONNELL; STRASSER; TSANG, 2018; ANGEL *et al.*, 2018). Esta etapa é necessária pois, apesar dos algoritmos de predição de genes serem poderosos e estarem em constante evolução, erros de anotação estrutural relacionados a genes não preditos, predições falsas e genes unificados ou cortados podem ocorrer, levando a consequentes erros de anotação funcional (SALZBERG, 2019; CRUZ *et al.*, 2019). A correção manual dos erros estruturais encontrados e re-anotação funcional para estes casos é indispensável para a publicação de resultados precisos e confiáveis e para evitar a propagação de erros em pesquisas futuras.

Diante de todas estas atividades que envolvem a anotação funcional, é evidente a necessidade de um ambiente colaborativo e unificado, que proporcione aos pesquisadores envolvidos a execução automática das ferramentas necessárias para realizar o processo de anotação funcional e possibilite a visualização, interpretação, reorganização e edição dos dados estruturais e funcionais em um único contexto, sem a necessidade de instalação de *softwares* ou bibliotecas adicionais. As desvantagens encontradas nas soluções existentes para este propósito incluem: 1) limitações ou inexistência de funcionalidades gratuitas; 2) exigência de execução por linha de comando ou instalação local, impossibilitando o acesso *web* e *online*; 3) inexistência de divisão de projetos e funcionalidades colaborativas para trabalho em conjunto com outros pesquisadores; 4) carência de consulta a bancos de dados específicos; 5) impossibilidade de editar informações estruturais e funcionais, para curadoria manual e 6) limitações na exportação dos resultados. No que se refere a este último item, julga-se importante que a solução possibilite não somente a exportação dos dados para formatos de arquivos comumente utilizados em ferramentas de bioinformática, mas também que seja possível exportá-los para formatos utilizados no momento da publicação no banco de dados GenBank, por ser um dos mais populares para o armazenamento de sequências de nucleotídeos e proteínas.

Frente à esta lacuna encontrada, esta pesquisa visa criar um *workflow* científico de anotação genômica funcional e curadoria manual de genomas, através do desenvolvimento de uma ferramenta *web* colaborativa que permita a execução automática de ferramentas de anotação funcional e integração dos resultados em um contexto unificado, facilitando a visualização, a interpretação, a edição e a exportação dos dados de anotação. A motivação para esta pesquisa deu-se devido à necessidade de anotação genômica das linhagens selvagem (2HH) e mutante (S1M29) do fungo *Penicillium echinulatum* (LENZ *et al.*, 2020), um projeto do Laboratório de Bioinformática e Biologia Computacional da Universidade de Caxias do Sul[1] (Caxias do

[1] https://www.ucs.br

Sul, Brasil). O estudo detalhado deste fungo é promissor pelo fato do seu excelente potencial para a produção de celulases, utilizadas na produção de xaropes de glicose que, se fermentados, podem ser utilizados para a produção de etanol de segunda geração ou de outros produtos biotecnológicos.

1.1 OBJETIVOS

Esta pesquisa visa alcançar os seguintes objetivos:

1.1.1 Geral

Desenvolver um *workflow* científico de anotação genômica funcional e curadoria manual de genomas, através da implementação de uma ferramenta *web* colaborativa que permita a execução automática de ferramentas de anotação funcional e integração dos resultados em um contexto unificado, facilitando a visualização, a interpretação, a edição e a exportação dos dados de anotação.

1.1.2 Específicos

- Selecionar os bancos de dados e ferramentas de anotação funcional que serão incorporados à solução.
- Identificar a forma de acesso e os formatos de saída que são gerados ao consultar cada um dos bancos de dados e ferramentas envolvidos.
- Desenvolver uma solução *web* que permita o trabalho colaborativo de múltiplos pesquisadores no processo de anotação funcional e curadoria manual de um determinado organismo, tendo como etapa inicial a importação dos arquivos de genes preditos, provenientes da fase de anotação estrutural.
- Incorporar à solução funcionalidades através das quais o usuário possa agendar consultas automáticas às ferramentas e bancos de dados disponíveis.
- Proporcionar uma visualização compreensível, intuitiva e integrada dos resultados, bem como a possibilidade de filtrar, reorganizar e editar as informações.
- Incorporar à solução funcionalidades que permitam a exportação dos genes anotados para os formatos exigidos no processo de publicação do genoma no banco de dados GenBank.
- Validar a solução proposta no processo de anotação genômica das linhagens selvagem (2HH) e mutante (S1M29) do fungo *Penicillium echinulatum*.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 SEQUENCIAMENTO DE DNA

O ácido desoxirribonucleico, ou DNA, é o material genético encontrado nas células da maior parte dos organismos vivos, consistindo em um polímero de repetidas unidades de nucleotídeos, distribuídos ao longo de duas cadeias no modelo de dupla hélice, isto é, unidos e enrolados formando um espiral. Cada nucleotídeo é constituído de um açúcar, um grupo fosfato e uma base nitrogenada, sendo que esta última pode ser: adenina (A), guanina (G), citosina (C) e timina (T). Os nucleotídeos de cada uma das fitas estão conectados através de ligações covalentes fosfodiésteres, entre o açúcar de um nucleotídeo e o grupo fosfato de outro, enquanto a união entre as duas fitas é mantida por ligações de hidrogênio entre as bases nitrogenadas de cada uma das cadeias (SHETTY; AMIRTHARAJ; SHAIK, 2019).

O termo *sequenciamento* refere-se ao processo de determinar a ordem exata de bases nitrogenadas (A, G, C e T) presentes no DNA, sendo que o conjunto completo de sequências codificadoras e não codificadoras define o *genoma*. O genoma carrega a informação necessária para a vida de um organismo e, a possibilidade de sequenciá-lo de forma completa, possibilitou o estudo de todos os processos moleculares envolvidos em sistemas celulares. Além disso, graças à atual disponibilidade de técnicas de sequenciamento e à constante evolução das ferramentas de bioinformática, o sequenciamento de genomas tem colaborado com o surgimento de novas ciências "ômicas", como a proteômica e a transcriptômica (CHAITANYA, 2019). O Quadro 1 apresenta uma relação dos métodos de sequenciamento e da performance de algumas plataformas relacionadas aos mesmos.

Quadro 1 – Métodos e plataformas de sequenciamento.

Geração	**Método**	**Plataforma**	**Tamanho das reads**	**Throughput Pb / Tempo**	**Tempo do cliclo**	**Vantagens**	**Limitações**
1ª Geração	Sanger	Thermo Fisher 3730	400pb - 900pb	0.00069Gb - 0.0021Gb	35 min - 2 horas	- Alta acuracidade de resultados; - Tamanho das *reads*.	- Alto custo; - Baixa velocidade de sequenciamento.
2ª Geração	Pirosequenciamento	454 Life Sciences GS FLX	700pb	0,7Gb / dia	1 dia	- Tamanho das *reads*.	- Erros de *frameshift*; - Baixa velocidade de sequenciamento.
2ª Geração	Sequenciamento por síntese	Illumina HiSeq X	150pb	1,8Tb / <3 dias	∼3 dias	- Alta velocidade de sequenciamento.	Geração de *short-reads*.
2ª Geração	Sequenciamento por ligação	Thermo Fisher SOLid 5500xl	75pb	30-45Gb / dia	1 dia	- Alta acuracidade de resultados, em função do *two-base encoding*.	- Erros ao identificar regiões palindrômicas; - Geração de *short-reads*.
2ª Geração	Sequenciamento por semi-condutor	Thermo Fisher Ion GeneStudio S5 series	200-600pb	Até 50Gb / dia	2-12 horas	- Curta duração dos ciclos. - Tamanho das *reads*.	- Erros ao identificar regiões homopoliméricas.
3ª Geração	Single Molecule Real-Time	Pacific Bioscience Sequel	∼10kb - 30kb	-	-	- Tamanho das *reads*; - Alta velocidade de sequenciamento; - Sequenciar regiões repetitivas com maior eficiência.	- Alta taxa de erros de sequenciamento.
3ª Geração	Nanopore	Oxford Nanopore MinION	Ilimitado	-	-	- Tamanho das *reads*; - Sequenciar regiões repetitivas com maior eficiência; - Portabilidade.	- Alta taxa de erros se sequenciamento.

Fonte: Elaborada pelo autor.

Dois métodos se destacam por formar a base das técnicas de sequenciamento e abrir

caminho para tecnologias subsequentes, sendo eles o *Método de Maxam-Gilbert* (também conhecido como método de degradação química) (MAXAM; GILBERT, 1977) e o *Método de Sanger* (também conhecido como método terminador de cadeia ou *chain terminator method*) (SANGER; NICKLEN; COULSON, 1977), ambos envolvendo fragmentação de DNA e a marcação radioativa em pontos específicos, possibilitando a determinação da ordem dos nucleotídeos por eletroforese em géis de poliacrilamida (SHENDURE *et al.*, 2017). O método de Maxam-Gilbert, que utilizava compostos químicos para clivar a molécula de DNA em bases nitrogenadas específicas, apesar de ter sido extremamente popular por alguns anos, não teve sucesso posterior, em função da sua complexidade técnica e da necessidade de uso de produtos químicos perigosos (GIANI *et al.*, 2019).

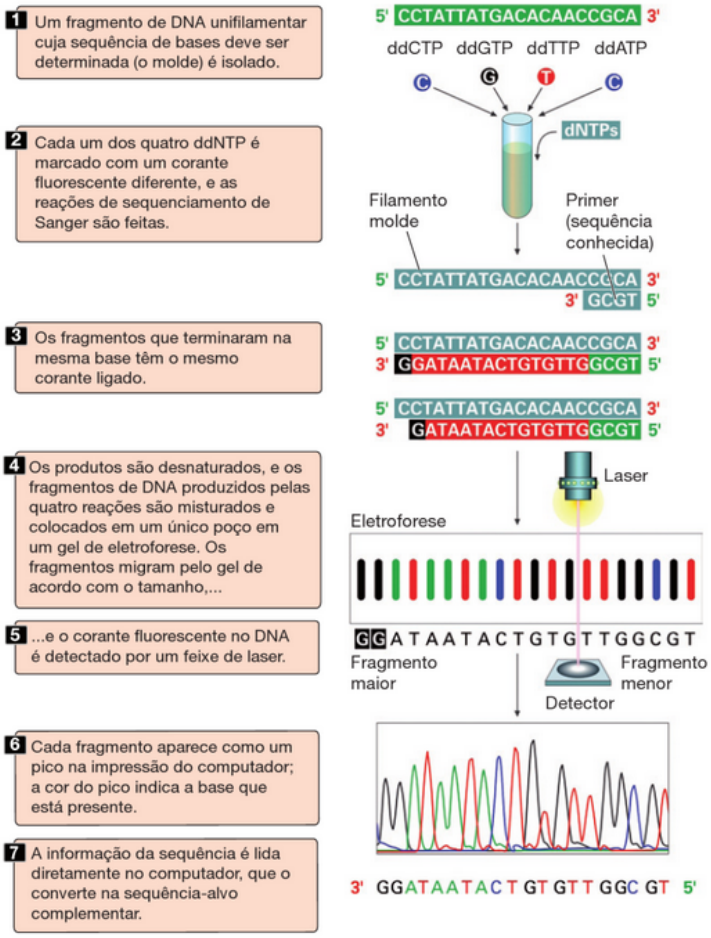
O método de Sanger (Figura 1), por sua vez, foi melhor aceito e se tornou um padrão de sequenciamento. O procedimento é baseado na interrupção da replicação do DNA em pontos diferentes, através da utilização de nucleotídeos modificados denominados didesoxiribonucleotídeos trifosfato (ddNTP's), sendo um para cada base nitrogenada: dATP, dTTP, dCTP, dGT. Estes nucleotídeos terminadores de cadeia não possuem o grupo 3-OH, impedindo a formação de uma nova ligação fosfodiéster pela enzima DNA polimerase. Consequentemente, ocorre uma interrupção da replicação e a geração de fragmentos de DNA de diferentes comprimentos. Uma vez que os didesoxiribonucleotídeos são previamente marcados com corantes de cores diferentes e que cada um dos fragmentos gerados possui uma unidade dos mesmos, é possível identificá-los através de eletroforese em géis de poliacrilamida ou em máquinas automatizadas de sequenciamento (SLATKO; GARDNER; AUSUBEL, 2018).

Apesar do alto custo e da baixa velocidade de sequenciamento, o método de Sanger dominou as três décadas seguintes e continua sendo utilizado para diversas finalidades, em função da alta acuracidade dos resultados e dos tamanhos das *reads* produzidas (de 400 até 900 pares de bases). A continuidade da utilização do método se deu graças a aprimoramentos que ocorreram nos anos subsequentes, tornando a técnica mais simples, rápida e segura. Os avanços se devem, primeiramente, à substituição dos marcadores radioativos por marcadores fluorescentes, sendo estes detectados por comprimentos de onda específicos, permitindo que a reação ocorra em apenas um tubo, ao invés de quatro. Os géis foram substituídos, ainda, por capilares que, além de muito menores, operam com maiores voltagens para reduzir o tempo de sequenciamento (SHAIK *et al.*, 2019a). As séries *3500* e *3730* da *Thermo Fisher*[1] são exemplos de sequenciadores automáticos que utilizam o método de Sanger.

A busca por tecnologias de sequenciamento mais rápidas, econômicas e alternativas ao uso de eletroforese iniciou-se ainda entre as décadas de 1980 e 1990. Os esforços para alcançá-las intensificaram-se após a conclusão do projeto Genoma Humano (VENTER *et al.*, 2001), um projeto de alto custo, consumo de uma grande quantidade de recursos e duração de 13 anos, comprovando que os métodos de sequenciamento existentes até então seriam pouco

[1] https://www.thermofisher.com

Figura 1 – Método de Sanger.



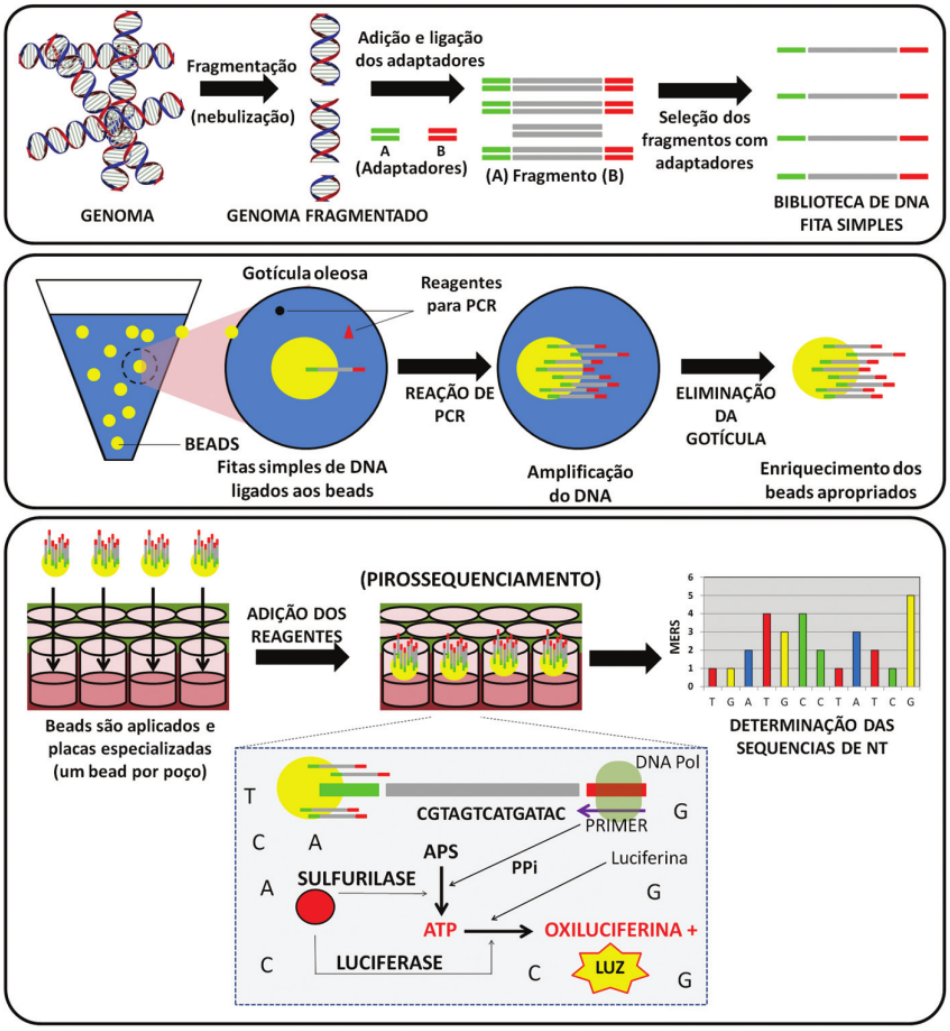
Fonte: Adaptada de Borges-Osório e Robinson (2013, p. 576).

viáveis para o sequenciamento de genomas completos (*whole-genome sequencing* ou WGS). Foi então que houve o início de uma revolução das plataformas de sequenciamento e o surgimento das tecnologias de sequenciamento de nova geração (*next-generation sequencing ou NGS*), também conhecidas como a segunda geração de plataformas de sequenciamento. Entre as inovações mais notáveis destacam-se: 1) a possibilidade de produzir milhões de reações de sequenciamento paralelamente, de forma simultânea, permitindo o sequenciamento em larga escada (*high-troughput*); 2) a substituição do clone bacteriano dos fragmentos de DNA por amplificação *in vitro*; 3) os resultados são diretamente processados e passíveis de leitura por programas de computador, sem a necessidade de eletroforese e 4) a redução significativa do custo por nucleotídeo sequenciado (SINGH *et al.*, 2018).

A primeira tecnologia da nova geração foi o *pirosequenciamento*, presente nos sequenciadores da *454 Life Sciences* (MARGULIES *et al.*, 2005) (Figura 2). Na fase de preparo da amostra, o DNA de interesse é fragmentado, sendo que cada fragmento é ligado à uma nanosfera, ou *bead*. Os fragmentos ligados em cada *bead* são, então, amplificados através de reação em cadeia da polimerase (PCR) em emulsão, resultando em *beads* contendo milhões de cópias de um

mesmo fragmento. Cada *bead* é colocada em uma lâmina de sequenciamento, juntamente com os reagentes necessários para a sua amplificação e, finalmente, ocorre o pirosequenciamento, que consiste na dedução da sequência do fragmento de DNA através da detecção de luz emitida pela liberação de pirofosfato para cada nucleotídeo incorporado (SLATKO; GARDNER; AUSUBEL, 2018).

Figura 2 – Pirosequenciamento na plataforma 454.



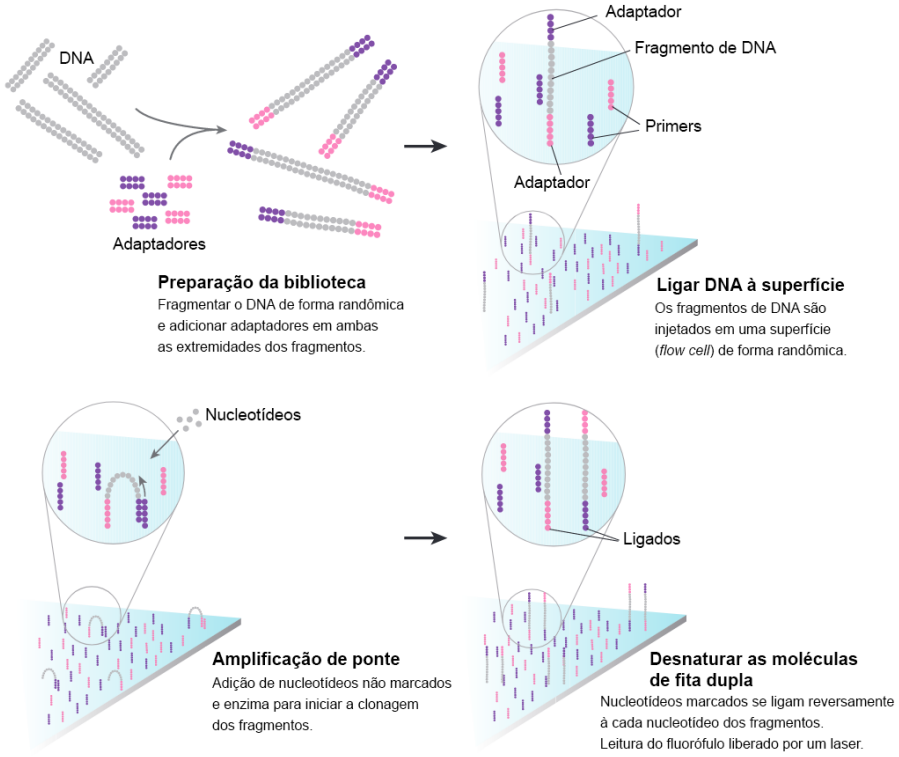
Fonte: Adaptada de Moreira (2015, p. 44).

.

Entre alguns modelos da *454 Life Sciences*, o GS20 (2005) produzia *reads* de aproximadamente 100 pares de bases, tinha 99% de acuracidade de resultados e podia sequenciar até 25 milhões de pares de bases em um tempo de 4 horas, com apenas um sexto do custo dos métodos convencionais. O modelo GS FLX, de 2011, já produzia *reads* de tamanhos maiores (aproximadamente 700 pares de bases), podendo sequenciar até 0,7 giga pares de bases em 24 horas. Apesar de este último e de novos modelos que surgiram até 2014 produzirem *reads* de tamanhos relativamente longos, observa-se a ocorrência de erros de *frameshift* (LIU *et al.*, 2012; KCHOUK; GIBRAT; ELLOUMI, 2017).

Após o sucesso do modelo GS FLX, houve o surgimento de um grande número de plataformas de sequenciamento massivamente paralelas. A plataforma *Illumina*[2], por exemplo, utiliza a tecnologia de sequenciamento por síntese (SBS), uma abordagem muito similar à eletroforese capilar, sendo que a principal diferença se dá na substituição dos didesoxiribonucleotídeos por nucleotídeos modificados e marcados com fluorescência. O processo, ilustrado na Figura 3, consiste em: 1) fragmentar o DNA de forma randômica e adicionar adaptadores nas extremidades 5' e 3' (fase de preparação da biblioteca); 2) injetar os fragmentos em uma lâmina (*flow cell*) e cloná-los através de amplificação de ponte (fase de geração de cluster) e 3) adicionar sequencialmente os nucleotídeos marcados para que se liguem reversamente à cada nucleotídeo de um fragmento, tornando possível a leitura do fluoróforo liberado através de um laser. Uma vantagem considerável é a grande quantidade de sequenciamentos por ciclo, entretanto, a geração de *reads* de tamanhos menores (*short-reads*) é considerada uma desvantagem, principalmente para projetos de sequenciamento genomas *de novo*[3]. Um exemplo de sequenciador é o modelo Illumina HiSeq X, capaz de realizar em torno de 6 bilhões de leituras por ciclo, gerando 1,8 Terabytes de dados e *reads* de 150 pares de bases (SHETTY; AMIRTHARAJ; SHAIK, 2019).

Figura 3 – Sequenciamento por síntese na plataforma *Illumina*.



Fonte: Adaptada de Mardis (2008).

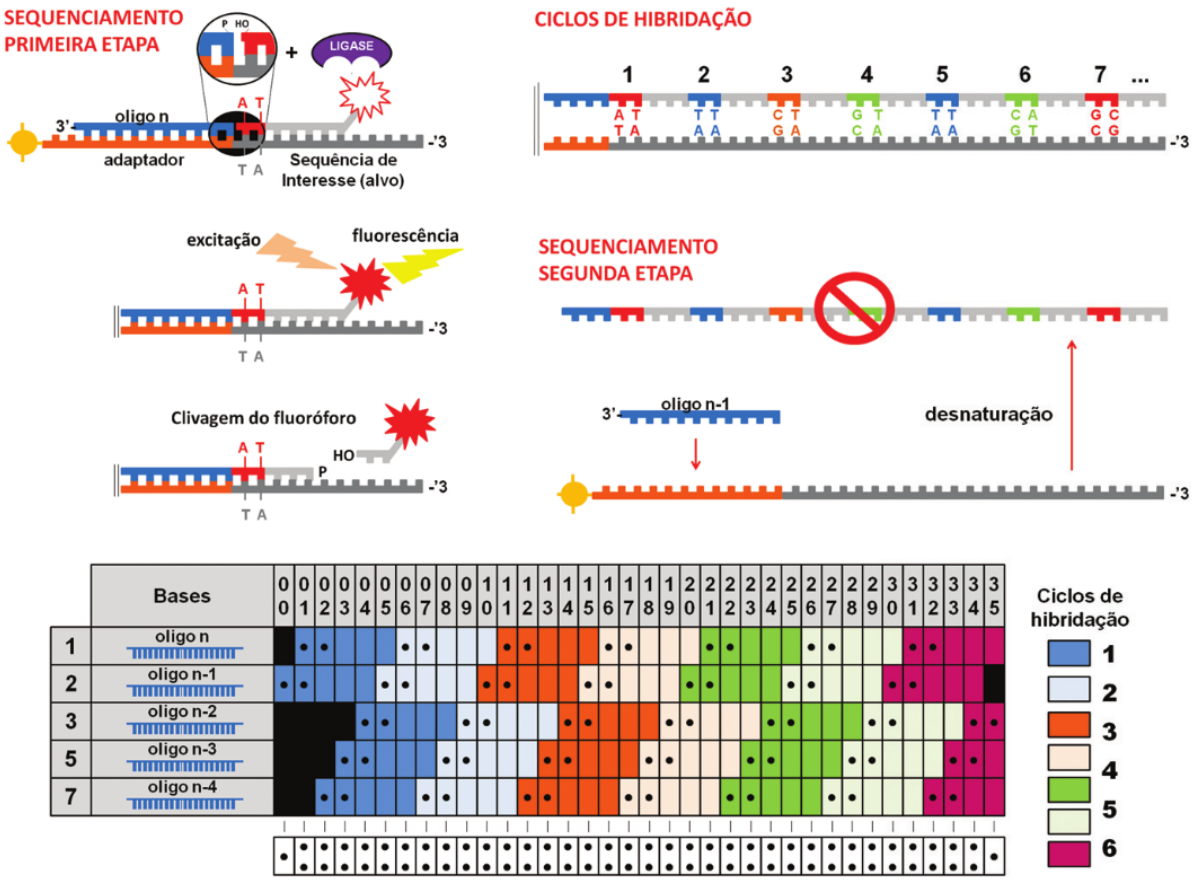
A plataforma *Sequencing by Oligonucleotide Ligation and Detection* (SOLid), da *Thermo*

[2] https://www.illumina.com

[3] Em projetos de genomas *de novo*, não há um genoma de referência para auxiliar nos alinhamentos das *short-reads* e montagem do *assembly*. Historicamente, principalmente por restrições de tempo e custo, somente um indivíduo por espécie era sequenciado e representava o genoma de "referência"para tal espécie (GIANI *et al.*, 2019).

Fisher, aplica um processo de sequenciamento por ligação (Figura 4), que também conta com fragmentação de DNA, ligação de adaptadores, amplificação (através de PCR em emulsão) e determinação da sequência através da liberação de fluoróforo. A diferença principal, neste caso, se dá através da utilização da enzima DNA ligase no processo de sequenciamento, ao invés da enzima DNA polimerase. (CHAITANYA, 2019). O ponto positivo desta plataforma é a alta acuracidade dos resultados, já que cada base é lida duas vezes (*two-base enconding*). A desvantagem está nas *reads* de tamanhos muito pequenos e em apresentar problemas ao identificar sequências palindrômicas. O tamanho máximo das *reads* (75 pares de bases) é dado pelo modelo *5500xl*, que sequencia até de 30 a 45 Gigabytes de dados em um ciclo de 24 horas (HUANG *et al.*, 2012).

Figura 4 – Sequenciamento por ligação na plataforma *Solid*.

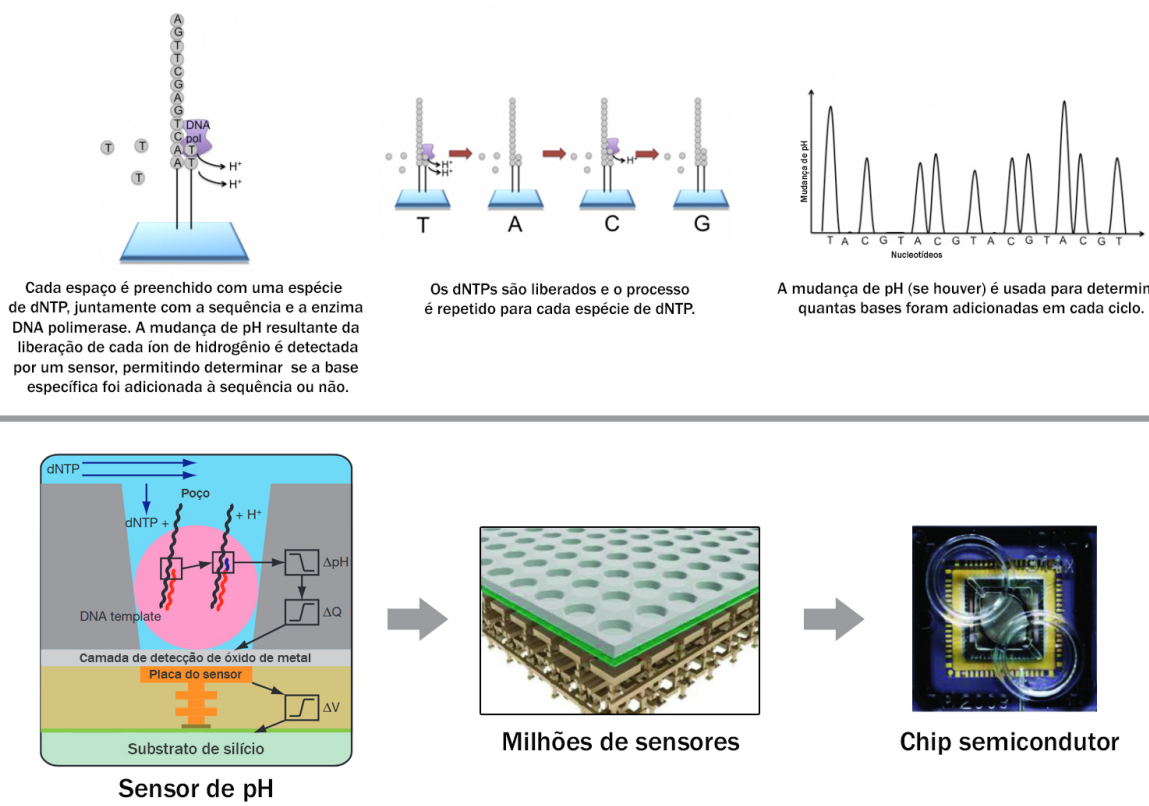


Fonte: Adaptada de Moreira (2015, p. 52).

Outra plataforma importante é a *Ion Torrent*, comercializada também pela *Thermo Fisher*. Esta tecnologia utiliza a metodologia de sequenciamento por semi-condutor (Figura 5) e diferencia-se das demais tecnologias de segunda geração por não utilizar marcadores fluorescentes na identificação dos nucleotídeos. Neste caso, micro-sensores detectam a mudança de pH resultante da liberação de um íon de hidrogênio, durante o processo de sequenciamento de cada nucleotídeo incorporado. Os sequenciadores que compõem esta plataforma produzem

até 50 *Gigabytes* de dados por dia em ciclos de 2 à 12 horas, gerando *reads* de 200 à 600 pares de bases. A tecnologia é atrativa em função das *reads* de tamanhos maiores e da curta duração dos ciclos. A desvantagem se dá em função de apresentar erros de sequenciamento em regiões homopoliméricas (KCHOUK; GIBRAT; ELLOUMI, 2017).

Figura 5 – Sequenciamento por semi-condutor na plataforma *Ion Torrent*.



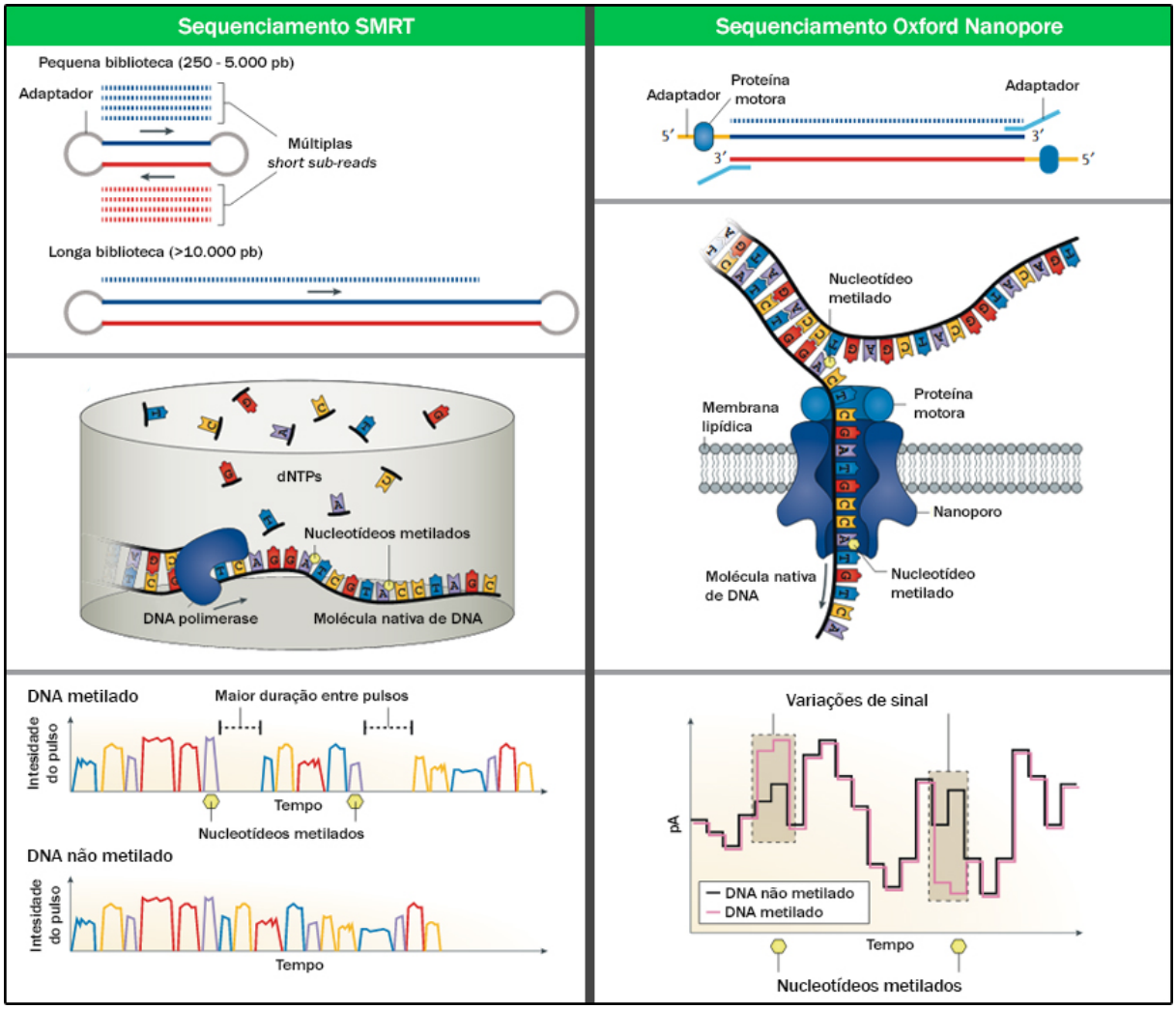
Fonte: Adaptada de Rothberg *et al.* (2011), EBI (2019).

Apesar das tecnologias de segunda geração ainda dominarem o mercado, novas abordagens de sequenciamento surgiram posteriormente e continuam evoluindo, caracterizando a terceira geração de sequenciadores. Os principais objetivos dos equipamentos desta geração são: 1) eliminar a necessidade da fase de amplificação por PCR, uma vez que este é um processo que eleva o tempo e o custo do sequenciamento e 2) produzir *reads* de tamanhos maiores, já que o sequenciamento de pequenos fragmentos torna a fase de montagem de genoma muito complexa e, muitas vezes, é incapaz de lidar com regiões repetitivas existentes em genomas complexos (GIANI *et al.*, 2019). As duas principais abordagens que surgiram para endereçar estes problemas são a *Single Molecule Real-Time* (SMRT), comercializada pela *Pacific Biosciences* (PacBio)[4] e a *Nanopore*, comercializada pela Oxford Nanopore Technologies[5] (Figura 6). A primeira utiliza nano-poços, cada qual contendo uma enzima DNA polimerase e permite detectar em tempo real

[4] https://www.pacb.com
[5] https://nanoporetech.com

a ação da incorporação da enzima em cada base (cinética enzimática), produzindo *reads* com tamanho médio de 10-30Kpb. A segunda realiza a leitura em tempo real de um sinal elétrico gerado pela passagem de cada base nitrogenada por um poro proteico, sendo que não há limite de tamanho de leitura, já que o dispositivo lê o tamanho do DNA que for inserido (a maior molécula sequenciada até o momento é de 2,3Mpb) (PAYNE *et al.*, 2019). Apesar das *reads* de tamanhos muito maiores e da consequente possibilidade de sequenciar regiões repetitivas com maior acuracidade, ainda é um desafio diminuir as taxas de erro de sequenciamento, que se apresentam maiores se comparadas às taxas de erro dos equipamentos de segunda geração (KCHOUK; GIBRAT; ELLOUMI, 2017).

Figura 6 – Sequenciamento SMRT e *Oxford Nanopore*.



Fonte: Adaptada de Beaulaurier, Schadt e Fang (2019).

2.2 MONTAGEM DO GENOMA

A saída dos equipamentos de sequenciamento de nova geração consiste em milhares de arquivos de *short-reads*, cada qual contendo algumas centenas de nucleotídeos. Como etapa

subsequente ao processo de sequenciamento, inicia-se a fase de montagem do genoma, que consiste em recombinar e realinhar tais sequências, com o intuito de recriar a sequência original de DNA que havia sido submetida ao equipamento e permitir pesquisas científicas posteriores. Este processo tornou-se uma tarefa desafiadora, especialmente para genomas *de novo* já que, nestes casos, os *softwares* de montagem (*assemblers*) devem se basear apenas nas *reads* sequenciadas para montar o genoma completo, não existindo nenhum genoma de referência para auxiliá-los no alinhamento (LI; WANG; LIU, 2017). A montagem de um genoma é constituída de etapas que devem ser executadas de modo sequencial (Quadro 2) e que são essenciais para a obtenção da sequência genômica.

Quadro 2 – Etapas da montagem de genomas.

Etapa	**Objetivo**	**Ferramentas**
Pré-processamento	Verificar e remover adaptadores e bases de baixa qualidade.	FastQC, Trim Galore!, Trimmomatic.
Montagem	Recombinar e realinhar as *reads* em fragmentos maiores (*contigs*), com o intuito de recriar a sequência original de DNA.	*Assemblers* de abordagem grafos de Bruijn (DBG) ou *overlap layout consensus* (OLC); Exemplos: ABySS e Velvet (DBG) ou Edena (OLC).
Verificação do *Assembly*	Analisar a contiguidade da montagem e identificar se o genoma está pronto para a fase de anotação.	Quast (avaliação da métrica N50); BUSCO (avaliação de completude do genoma).

Fonte: Adaptada de Li, Wang e Liu (2017), Angel *et al.* (2018).

O primeiro passo a ser realizado no processo de montagem de genoma é uma etapa de pré-processamento que consiste na verificação e remoção de adaptadores e bases de baixa qualidade, ambos característicos dos equipamentos de sequenciamento de nova geração. Adaptadores são sequências adicionadas na fase de preparação da biblioteca e bases de baixa qualidade existem pois, uma vez que o sequenciamento é realizado em ciclos, os reagentes podem gradativamente perder sua vigorosidade e ocasionar leituras errôneas da intensidade da fluorescência liberada pelos mesmos (ABNIZOVA; BOEKHORST; ORLOV, 2017). Os arquivos de saída produzidos pelos sequenciadores, além de conterem sequências de nucleotídeos, incluem também os respectivos escores de qualidade para cada uma das bases, identificado a probabilidade de cada uma ter sido identificada corretamente pelo sequenciador. Um dos *softwares* mais utilizados para realizar verificações de qualidade é o FastQC (ANDREWS *et al.*, 2010) que, através de gráficos e tabelas, proporciona ao usuário uma visão geral das regiões que podem conter problemas de preparação de biblioteca ou de sequenciamento. Uma vez identificadas as regiões, é necessário removê-las através da utilização de *softwares* específicos para este fim, tais como o Trim Galore! (KRUEGER, 2015), o PRINSEQ (SCHMIEDER; EDWARDS, 2011) ou o Trimmomatic (BOLGER; LOHSE; USADEL, 2014).

Após a fase de verificação de qualidade, inicia-se a fase de montagem. Nos últimos anos, muitos algoritmos de sequenciamento de genomas *de novo* foram desenvolvidos para alinhar grandes quantidades de *short-reads* e transformá-las em fragmentos maiores, denominados *contigs*. Escolher o *assembler* apropriado, por outro lado, tornou-se uma tarefa desafiadora. Os

algoritmos de montagem mais utilizados fazem uso das abordagens grafos de Bruijn (DBG) e *overlap layout consensus* (OLC), ambas baseadas na teoria dos grafos. A primeira tem vantagem sobre a segunda por consumir menos tempo computacional e memória, além de apresentar melhor eficiência para alinhamentos de *short-reads*. O método é baseado na divisão das *reads* em partes menores de tamanho k (k-mers), que formam os nodos do grafo, sendo estes conectados quando dividem a sobreposição exata de um k-1 mer. A vantagem do método OLC sobre o DBG se dá no alinhamento de *long-reads*, sendo que realiza alinhamentos das sequências, calcula o melhor grafo de sobreposição e, por fim, gera uma sequência consenso dos *contigs* encontrados no grafo (KHAN *et al.*, 2018). Como exemplos de *softwares* de montagem, pode-se citar o ABySS (SIMPSON *et al.*, 2009) e o Velvet (ZERBINO; BIRNEY, 2008), que adotam a abordagem DBG e o Edena (HERNANDEZ *et al.*, 2008), que adota a abordagem OLC.

Na fase de montagem, é comum que sejam realizadas tentativas paralelas utilizando diversos *assemblers* ou, até mesmo, alterando parâmetros de configuração de um mesmo *assembler*, com o intuito de obter resultados com o mínimo de fragmentos possível. Determinar o momento de parada e definição de que o genoma está pronto para as próximas fases de anotação, entretanto, é um ponto chave neste processo. Existem algumas estatísticas que podem ser utilizadas para auxiliar na escolha entre os diferentes *assemblies* obtidos como, por exemplo, a métrica N50. O valor de N50 está relacionado com o tamanho do *contig* que esteja posicionado na região que representa 50% do tamanho total do genoma, depois de os *contigs* serem ordenados do menor para ao maior. Ou seja, esta métrica permite compreender que 50% do genoma contém *contigs* de tamanho igual ou maior do que este, tornando esta uma métrica muito importante para avaliar a contiguidade da montagem, uma vez que valores maiores indicam menores níveis de fragmentação. É importante ressaltar que o propósito do N50 é avaliar o *assembly* no contexto da contiguidade, não necessariamente indicando que o mesmo está correto, já que há possibilidade de que alguns *softwares* de montagem produzam *contigs* longos por conectar regiões de forma errônea em ordem ou orientação (ANGEL *et al.*, 2018). O *software* Quast (GUREVICH *et al.*, 2013) pode ser utilizado para comparar esta e outras métricas entre diferentes *assemblies* a auxiliar o pesquisador a melhorar e selecionar o melhor *assembly*.

Se o objetivo final da pesquisa é identificar genes que produzem proteínas, a utilização da ferramenta BUSCO (SEPPEY; MANNI; ZDOBNOV, 2019) é uma opção muito útil para avaliar a completude do genoma e servir como complemento à métrica N50. A solução inclui em seus *datasets* grupos de ortólogos provenientes do banco de dados OrthoDB, sendo que são selecionados apenas os genes que estão presentes em mais de 90% das espécies de um determinado grupo e que sigam o princípio da "cópia única"(*single-copy orthologs*), isto é, que estejam presentes em mais de 90% destas mesmas espécies sem duplicações ou perdas ao longo do processo evolutivo. Estas regras permitem estabelecer uma expectativa evolucionária de que os genes selecionados em cada grupo devem ser encontrados em quaisquer novos genomas sequenciados de organismos deste mesmo grupo. Assim, é possível comparar diversos *assemblies* e selecionar o que possua um melhor grau de completude e maior quantidade de genes completos

dentro da linhagem que se enquadra.

2.3 BANCOS DE DADOS BIOLÓGICOS

As tecnologias de sequenciamento de nova geração e a popularização dos projetos genoma vêm gerando enormes quantidades de dados, que crescem em uma taxa exponencial. Os dados armazenados são relacionados à diversas áreas, tais como: genômica, expressão e função gênica, proteômica, metabolômica, filogenia, entre outras. A necessidade de gerenciar, armazenar e consultar estas informações de forma eficiente levou à criação de diversos tipos de bancos de dados, que logo se tornaram indispensáveis ferramentas de representação de dados biológicos na comunidade científica. A maior parte dos bancos de dados biológicos estão disponíveis através de *websites*, através dos quais o usuário pode pesquisar e navegar de forma *online* (Quadro 3). Cada qual possui também suas ferramentas específicas e *web services*[6] de acesso para a realização de consultas avançadas ou em lote (BHATT; PATEL; JOSHI, 2018).

De acordo com o tipo ou natureza das informações armazenadas, os bancos de dados biológicos podem ser classificados em três categorias: primários, secundários e especializados. Os bancos de dados primários, também chamados de arquivos ou repositórios, contém dados experimentais de sequências de nucleotídeos ou proteínas, publicados diretamente pelo pesquisador individual, sendo que este próprio possui privilégios de alterar o que foi publicado. Apesar da rápida publicação, é possível que os dados publicados contenham eventuais erros, já que não há a garantia de análise cuidadosa das informações por parte do pesquisador. Outro problema encontrado é a tendência de redundância de dados, já que dois ou mais pesquisadores podem realizar a submissão de uma mesma sequência de forma independente (CHRISTENSEN; VRIES, 2018).

Entre os bancos de dados primários, destaca-se o GenBank (BENSON *et al.*, 2018), como o principal banco de dados público de sequência de nucleotídeos que, em sua versão mais recente (240.0, de outubro de 2020), continha mais de 219 milhões de sequências depositadas. Criado e mantido pelo *National Center for Biotechnology Information* (NCBI), o Genbank é parte do consórcio *International Nucleotide Sequence Database Collaboration* (INSDC), juntamente com outros dois bancos de dados primários, sendo eles: *DNA DataBank of Japan* (DDBJ) (OGASAWARA *et al.*, 2020) e *European Molecular Biology Laboratory* (EMBL) (KANZ *et al.*, 2005). A maior parte das revistas internacionais exige que os autores submetam as sequências de seus projetos em uma destas três plataformas, como parte dos requisitos para o aceite da publicação. As informações destes bancos de dados são compartilhadas e sincronizadas diariamente, sendo necessário apenas realizar pesquisas em um dos três. Apesar de disponibilizarem os mesmos dados, cada um possui *design* e formatos de exibição diferentes. (SHAIK *et al.*, 2019b). Ainda é possível citar como banco de dados primário o *Sequence Read*

[6] Sistemas de *software* projetados para permitir a interoperabilidade de aplicações heterogêneas através da *internet*. A troca de informações é realizada através de formatos padronizados para comunicação de dados, tais como o *Extensible Markup Language* (XML) e o *JavaScript Object Notation* (JSON) (PRIYADHARSHINI *et al.*, 2015).

Quadro 3 – Classificação dos bancos de dados biológicos.

Categoria	**Nome**	**Tipo de informação**	**URL de acesso**
Primário	GenBank	Sequências de DNA e proteínas	https://www.ncbi.nlm.nih.gov/genbank
Primário	EMBL	Sequências de DNA e proteínas	https://www.ebi.ac.uk/ena
Primário	DDBJ	Sequências de DNA e proteínas	https://www.ddbj.nig.ac.jp
Primário	SRA	Sequências de DNA e proteínas	https://www.ncbi.nlm.nih.gov/sra
Secundário	RefSeq	Sequências de DNA e proteínas	https://www.ncbi.nlm.nih.gov/refseq
Secundário	Pfam	Famílias de proteínas	https://pfam.xfam.org
Secundário	Prosite	Famílias de proteínas	https://prosite.expasy.org
Secundário	InterPro	Famílias de proteínas	https://www.ebi.ac.uk/interpro
Secundário	UniProt/TrEMBL	Sequências de proteínas e anotações (não revisadas)	https://www.uniprot.org
Secundário	UniProt/Swissprot	Sequências de proteínas e anotações (revisadas)	https://www.uniprot.org
Secundário	UniRef	Agrupamentos de sequências de proteínas por similaridade	https://www.uniprot.org/uniref
Secundário	UniParc	Sequências de proteínas não redundantes	https://www.uniprot.org/uniparc
Secundário	GO	Função dos genes (estrutura de grafo)	http://geneontology.org
Especializado	KEGG	Interrelação de genes, moléculas e vias metabólicas	https://www.genome.jp/kegg
Especializado	CAZy	Classificação de enzimas CAZy	http://www.cazy.org
Especializado	dbCAN	Classificação de enzimas CAZy	http://bcb.unl.edu/dbCAN2
Especializado	OrthoDB	Grupos de ortólogos	https://www.orthodb.org
Especializado	EggNOG	Grupos de ortólogos	http://eggnog5.embl.de
Especializado	COG	Grupos de ortólogos	https://www.ncbi.nlm.nih.gov/COG
Especializado	KEGG Orthology	Grupos de ortólogos	https://www.genome.jp/kegg/ko.html

Fonte: Adaptada de Shaik *et al.* (2019b), Bhatt, Patel e Joshi (2018), Christensen e Vries (2018), Selzer, Marhöfer e Koch (2018).

Archive (SRA) (LEINONEN *et al.*, 2010), um repositório público cujo propósito específico é o armazenamento de dados experimentais de sequenciamentos de plataformas de nova geração (em sua maioria *short-reads*), oferecendo suporte para as mais recentes plataformas.

Os bancos de dados secundários, por outro lado, contém resultados de análises realizadas com base nas informações dos bancos de dados primários, tais como a inferência de classes comuns entre sequências de proteínas e até anotação de sequências não conhecidas. As informações contidas nestas plataformas permitem analisar e tirar novas conclusões sobre os dados não-tratados, proporcionando um conhecimento mais sofisticado sobre os mesmos. Alguns, inclusive, contam com informações que ficam somente acessíveis publicamente após curadoria manual e controle de qualidade, eliminando a chance de erros e de redundância dos dados publicados (SELZER; MARHÖFER; KOCH, 2018).

O banco de dados RefSeq (O'LEARY *et al.*, 2016), por exemplo, é um repositório criado pelo NCBI em 1999 com o intuito de proporcionar uma coleção de dados compreensível, integrada e não-redundante de sequências de DNA, transcritos e proteínas. Este banco de

dados contém cópias de genomas já depositados no banco de dados GenBank, porém, estes são selecionados apenas ao atenderem os requisitos de qualidade, completude e unicidade. Os genomas selecionados passam por uma *pipeline*[7] de anotação genômica própria (sendo que há duas *pipelines*, uma para eucariotos e uma para procariotos), seguida de um processo de curadoria manual, proporcionando uma referência estável para identificação e caracterização de genes, análises de mutação e polimorfismo, estudos de expressão gênica e análises comparativas de genomas.

O Pfam (EL-GEBALI *et al.*, 2019), por outro lado, é um banco de dados de famílias de proteínas. Cada família é representada por múltiplos alinhamentos de sequências de proteínas de alta qualidade, que servem de ponto de partida para a construção de modelos ocultos de Markov[8], sendo utilizados para a identificação de padrões e probabilidades de aparecimento de um determinado aminoácido em cada posição das sequências. O resultado destes alinhamentos possibilita a identificação de regiões conservadas e a consequente inferência das famílias de proteínas. O Pfam ainda cria agrupamentos de nível maior (*clans*), compostos de famílias que possuem similaridade estrutural ou são identificadas como evolucionalmente relacionadas.

Um banco de dados similar ao Pfam é o Prosite (SIGRIST *et al.*, 2012), também utilizado para a caracterização e classificação de proteínas. A classificação também é produto de múltiplos alinhamentos, identificando pequenas regiões conservadas de 10-20 aminoácidos, que geralmente representam um ponto importante na função da proteína. A diferença se dá na forma como as regiões conservadas são armazenadas no banco de dados, sendo neste através de expressões regulares. Na representação da expressão regular, aminoácidos individuais são indicados por uma única letra e separados por hifens. Quando há a possibilidade de vários aminoácidos ocuparem uma mesma posição, estes são indicados dentro de colchetes e, para o caso de repetição de aminoácidos, o número de repetições é indicado entre parênteses. Posições que podem ser ocupadas por qualquer aminoácido são indicadas através da letra minúscula *x* e uma lista de aminoácidos dentro de chaves significa negação, isto é, qualquer aminoácido exceto os que estiverem na lista. O valor *[GSTNE]-[GSTQCR]-[FYW]-{ANW}-x(2)-P* é um exemplo de uma típica expressão regular no banco de dados Prosite.

O UniProt (CONSORTIUM, 2019) é uma coleção de sequências de proteínas e suas informações funcionais. A base de dados, denominada *UniProt Knowledge Base* (UniProtKB), é composta por uma combinação de dois outros bancos de dados: TrEMBL e Swissprot. O primeiro engloba os dados de sequências de proteínas do EMBL, sendo estas automaticamente anotadas

[7] Terminologia utilizada no campo da arquitetura da computação. Consiste na divisão de um processo computacional em múltiplos subprocessos, que executam de forma ordenada e compartilham um fluxo de dados de entrada, sendo que a saída do processamento de um estágio é utilizada como entrada para o processamento do próximo (TSAI; CHEN; TSENG, 2014).

[8] Um modelo estatístico para análise de padrões, muito utilizado para análises de sequências biológicas e para reconhecimento de fala, escrita e gestos. Uma cadeia de Markov é uma máquina de estados cujas transições são reguladas por probabilidades. Enquanto em um modelo regular de Markov estes estados são visíveis ao observador, no modelo oculto de Markov existem estados que não são diretamente observáveis (ANNACHHATRE; AUSTIN; STAMP, 2015).

por sistemas automatizados e, portanto, não apresentando garantia de qualidade da anotação. O banco de dados Swissprot, por outro lado, contém anotações funcionais revisadas, extraídas da literatura e adicionadas cuidadosamente por um grupo de especialistas em curadoria manual, tornando este banco de dados uma referência para anotação de proteínas. Em seu último *release* estatístico (outubro de 2020), havia mais de 195 milhões de sequências no UniProt/TrEMBL e quase 564 mil sequências manualmente revisadas no UniProt/Swissprot. O UniProt ainda conta com as bases de dados adicionais UniRef, contendo agrupamentos de sequências de acordo com seus níveis de similaridade e UniParc, cujo foco principal é apresentar as sequências de proteínas disponíveis publicamente de forma não-redundante, fusionando proteínas idênticas (mesmo as com referência a organismos diferentes).

O InterPro (MITCHELL *et al.*, 2018) é um repositório que realiza a integração dos resultados de diversos outros bancos de dados secundários em um único contexto, com o intuito de classificar sequências de proteínas em famílias e predizer a presença de funcionalidades e regiões conservadas importantes. Além dos já citados bancos de dados Pfam e Prosite, a plataforma integra outros bancos de dados de domínios de proteínas, sendo eles: CATH-Gene3D (LEWIS *et al.*, 2018), *Conserved Domains Database* (CDD) (MARCHLER-BAUER *et al.*, 2017), HAMAP (PEDRUZZI *et al.*, 2015), PANTHER (MI *et al.*, 2017), PIRSF (NIKOLSKAYA *et al.*, 2006), PRINTS (ATTWOOD *et al.*, 2012), ProDom (BRU *et al.*, 2005), SMART (LETUNIC; BORK, 2018), *Structure–Function Linkage Database* (SFLD) (AKIVA *et al.*, 2014), SUPERFAMILY (PANDURANGAN *et al.*, 2019) E TIGRFAMs (HAFT *et al.*, 2012). As entradas neste banco de dados não são adicionadas de forma automática, sendo que passam por inspeção manual para evitar falsos positivos e eliminar redundância de dados, já que muitas classificações derivadas de bancos de dados distintos podem representar a mesma função biológica e, consequentemente, devem estar agrupadas em uma única entrada do banco de dados UniProt. A ferramenta especializada de consulta aos dados deste banco de dados é denominada InterProScan (JONES *et al.*, 2014), que pode ser utilizada de três formas: 1) *online*; 2) no computador local do usuário, através do *download* da ferramenta e 3) através de *web services*, com bibliotecas disponíveis para as linguagens *Python* e *Ruby*.

O banco de dados GO (ASHBURNER *et al.*, 2000; ACENCIO *et al.*, 2019) é considerado o maior repositório de informações relacionadas à função dos genes. Diferentemente dos bancos de dados de famílias de proteínas, este possui uma estrutura que pode ser definida em termos de um grafo, na qual cada nó do grafo representa uma classe de função (também chamada de termo) e as arestas representam as relações entre estes termos. Apesar da estrutura de grafo, existe uma estrutura hierárquica secundária, na qual os termos "filhos"são mais especializados do que os termos "pai", sendo que ainda é possível que um termo filho herde características de mais do que um termo pai. Os níveis base desta hierarquia são representados pelas três classes principais de termos GO: função molecular, componente celular e processo biológico. Embora seja aparentemente complexo, o vocabulário é controlado e compreensível tanto por seres humanos quanto por computadores, sendo este último fator essencial para suportar a

pesquisa biológica moderna. Através da realização de alinhamentos das sequências de interesse, os pesquisadores podem identificar os termos relacionados com cada sequência e compreender como cada gene individual contribui para o todo em níveis celulares, moleculares e de organismo.

Os bancos de dados especializados englobam informações que podem ser utilizadas para contextos mais específicos. O *Kyoto Encyclopedia of Genes and Genomes* (KEGG) (KANEHISA *et al.*, 2016), por exemplo, é uma coleção de bancos de dados que, juntos, têm o objetivo de estabelecer *links* entre grupos de genes em um genoma e as funções de alto nível da célula e do organismo. Através de representações gráficas na forma de fluxogramas, é possível compreender a inter-relação de genes, moléculas e vias metabólicas e relacioná-los aos processos celulares dos organismos. Os dezoito bancos de dados que representam a coleção são divididos nas categorias: Informações de Sistemas, Informações Genômicas, Informações Químicas e Informações de Saúde.

Alguns bancos de dados são especializados na classificação de enzimas ativas sobre carboidratos (CAZymes). O estudo destas enzimas tem grande significância na indústria da agricultura e da bioenergia, já que os carboidratos são a principal fonte de alimento de animais e micróbios, além de servirem de matéria-prima para a produção de biocombustíveis e biomateriais (ZHANG *et al.*, 2018). O banco de dados CAZy (LOMBARD *et al.*, 2014), desde 1990, identificou e classificou mais de 360 famílias, divididas em seis classes principais: glicosiltransferases (GTs), glicosil hidrolases (GHs), polissacarídeo liases (PLs), carboidrato esterases (CEs), módulos de ligação ao carboidrato (CBMs) e enzimas para atividades auxiliares (AAs). É possível categorizar sequências de proteínas de acordo com as famílias CAZy através do servidor dbCAN (ZHANG *et al.*, 2018), que combina resultados do seu próprio banco de dados (*dbCAN HMM database*), do banco de dados CAZy e da biblioteca *peptide pattern recognition* (PPR) (BUSK; LANGE, 2013).

O OrthoDB (KRIVENTSEVA *et al.*, 2019), o EggNOG, (HUERTA-CEPAS *et al.*, 2016) o COG (GALPERIN *et al.*, 2019) e o KEGG Orthology (KANEHISA *et al.*, 2016) (uma das coleções do banco de dados KEGG) são exemplos de repositórios de grupos de ortólogos e suas respectivas anotações funcionais. A ortologia e a paralogia são conceitos centrais para a biologia evolutiva, sendo que o primeiro caso é resultado de eventos de especiação (um gene duplicado de um ancestral comum, se diferenciando gradualmente, mas mantendo a mesma função) enquanto o segundo é resultado da duplicação de genes em uma mesma espécie. Essa diferenciação é muito importante, já que assume-se que genes ortólogos têm maior tendência a conservar suas funções, tornando a busca por ortólogos um dos pilares da maioria dos métodos de anotação genômica (GABALDÓN; KOONIN, 2013).

2.4 ANOTAÇÃO GENÔMICA

A queda nos custos de sequenciamento ao longo do tempo tornou possível que um número gigantesco de sequências de genomas gradativamente maiores fosse gerado. Enquanto

realizar o sequenciamento de genomas não se apresentou mais como uma barreira científica, por outro lado, analisar e identificar o significado biológico das sequências geradas se tornou uma tarefa desafiadora. Isto porque cadeias de nucleotídeos, por si próprias, não têm propósito algum, e só começam a fazer sentido quando busca-se identificar a localização de elementos estruturais (genes) e atribuir funções aos mesmos, em um processo denominado *anotação genômica* (WANG *et al.*, 2017; YANDELL; ENCE, 2012). A anotação genômica é uma etapa crucial no processo de compreensão dos processos biológicos dos organismos estudados e na subsequente realização de novas pesquisas (MCDONNELL; STRASSER; TSANG, 2018).

Antes de iniciar o processo de anotação, é de extrema importância verificar se o *assembly* está pronto para ser anotado, através da utilização de estatísticas de avaliação, tais como a métrica N50 e verificação de completude do genoma com a ferramenta BUSCO, citadas anteriormente. Uma vez concluída esta etapa, inicia-se um fluxo típico de anotação genômica, como pode ser observado na Figura 7.

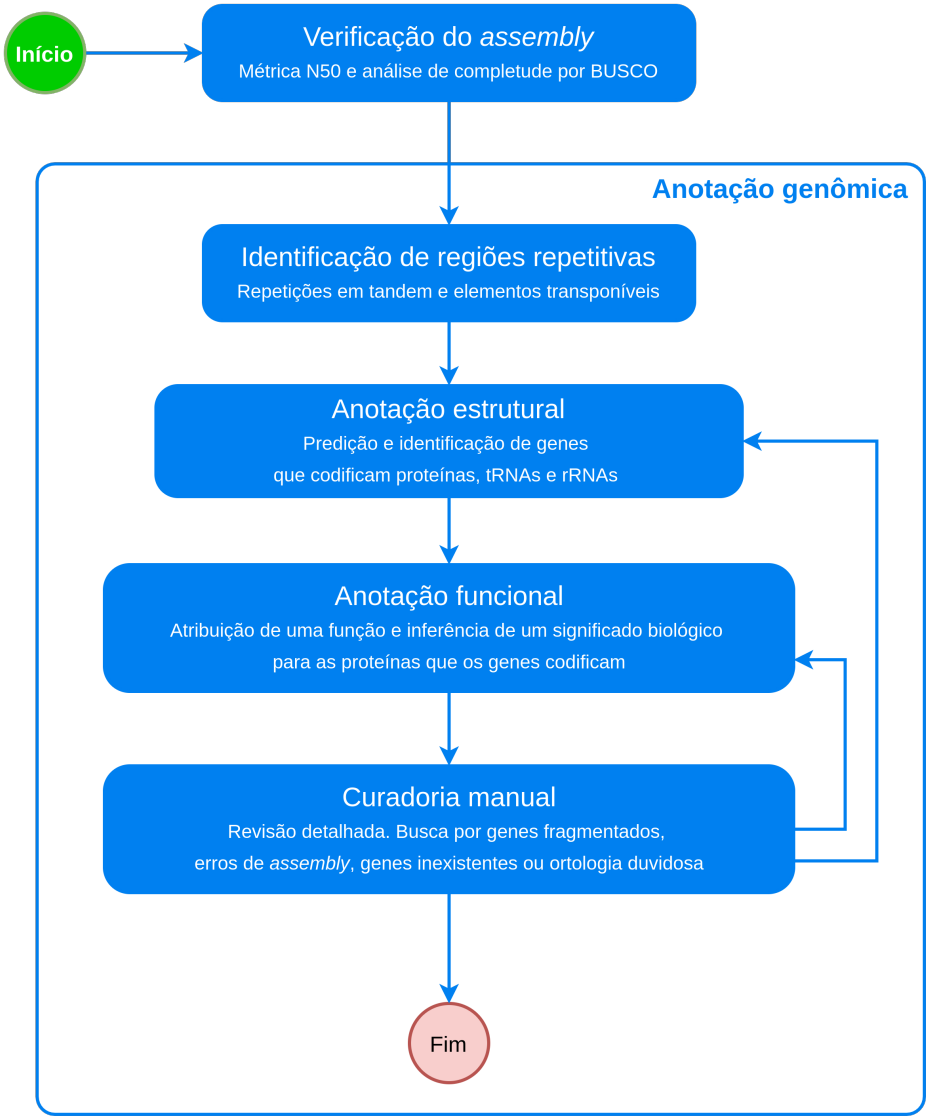
2.4.1 Identificação de regiões repetitivas

A identificação de regiões repetitivas é, em geral, a primeira etapa do processo de anotação. O termo "repetitivo", neste contexto, é utilizado para descrever dois tipos de sequências: as repetições em tandem e os elementos transponíveis. As repetições em tandem consistem nos DNAs satélite, minissatélite e microssatélite, enquanto os elementos transponíveis (ou elementos dispersos) são regiões capazes de se movimentar ao longo do genoma. Os elementos transponíveis desempenham um papel chave na estrutura do genoma, sendo que estão relacionados a varições genéticas e a evolução, uma vez que a sua movimentação ao longo do genoma pode afetar a estrutura, a função e a expressão dos genes. Os tipos mais comuns de elementos transponíveis são os elementos dispersos longos (LINEs) e os elementos dispersos curtos (SINEs) (YANDELL; ENCE, 2012; GOEL; KARIR; GARG, 2017).

As regiões repetitivas são estruturas complexas e podem ser muito abundantes no genoma de alguns organismos. De 5 a 10 por cento dos genomas bacterianos são constituídos por regiões repetitivas, sendo que este número se intensifica em organismos eucariotos, representando mais de dois terços do genoma humano e até 90 por cento de alguns genomas, como o do trigo (CHOULET *et al.*, 2010). Computacionalmente, estas regiões atrapalham a anotação e futuros alinhamentos, sendo a sua identificação um passo indispensável antes da realização das anotações estrutural e funcional. Ao serem identificadas pelos *softwares* de busca de regiões repetitivas, estas sequências são "mascaradas", isto é, cada nucleotídeo identificado como repetitivo é marcado com a letra "N"ou, em alguns casos, com uma letra minúscula a,c,t,g - sendo este último um processo denominado *soft masking* (YANDELL; ENCE, 2012).

Muitas ferramentas foram desenvolvidas para a identificação de regiões repetitivas. Um dos *softwares* mais utilizados é o RepeatMasker (SMIT; HUBLEY; GREEN, 2013-2015), que utiliza um método baseado na comparação de sequências com bibliotecas de repetições já

Figura 7 – Típico fluxo de anotação genômica.



Fonte: Adaptada de Angel *et al.* (2018).

conhecidas e depositadas em bancos de dados como o RepBase (BAO; KOJIMA; KOHANY, 2015), uma biblioteca de repetições manualmente anotadas para eucariotos. Outros *softwares* englobam métodos baseados em aprendizado (ANDRIEU *et al.*, 2004), métodos baseados em assinatura (MCCARTHY; MCDONALD, 2003), métodos para genomas *de novo* (BAO; EDDY, 2002; EDGAR; MYERS, 2005) e até métodos que combinam resultados de diferentes ferramentas (FLUTRE *et al.*, 2011; FLYNN *et al.*, 2019; GIRGIS, 2015).

2.4.2 Anotação estrutural

Anotação estrutural é o processo de determinar corretamente a localização e estrutura dos genes que codificam proteínas, tRNAs e rRNAs em um genoma, etapa esta também conhecida como predição de genes. Este é um conceito bem conhecido e compreendido, sendo que muitas

ferramentas e algoritmos de sucesso já foram desenvolvidos para solucionar o problema, cada qual com suas vantagens e limitações. De forma geral, existem três abordagens principais para a predição de genes: a predição intrínseca (também conhecida como predição *ab initio*), a predição extrínseca (também conhecida como método empírico ou baseado em evidência) e a predição combinada, que une características das duas primeiras (ANGEL *et al.*, 2018).

A predição intrínseca é um método computacional de predição de genes baseado na utilização de modelos matemáticos capazes de detectar certas características específicas presentes no DNA para identificar os genes e suas estruturas de íntrons e éxons. Estas características podem ser desde "sinais", isto é, sequências específicas que indicam a presença de um gene em determinadas posições, até propriedades estatísticas que sejam características de genes codificadores de proteínas. Uma premissa básica para esta abordagem é possuir um bom *set* de treinamento, já que os preditores de genes deste tipo possuem modelos e parâmetros pré-compilados para organismos específicos, englobando dados como frequências de códons e de distribuição de íntrons. Logo, a não ser que o genoma do organismo a ser estudado esteja muito próximo dos genomas disponíveis, o preditor deve ser especificamente reconfigurado e retreinado a partir do genoma de estudo. De qualquer forma, este fator apresenta-se também como uma vantagem, já que é possível construir modelos específicos capazes de realizar predições eficientes para espécies específicas. Porém, a utilização exclusiva de parâmetros estatísticos, sem a busca por evidências externas, pode reduzir a acuracidade da predição em alguns casos, principalmente no que se refere às estruturas de íntrons e éxons (YANDELL; ENCE, 2012; ANGEL *et al.*, 2018). Os *softwares* Augustus (STANKE; MORGENSTERN, 2005) e FGENESH (SOLOVYEV *et al.*, 2006) são exemplos de ferramentas que utilizam o método de predição intrínseca.

A predição extrínseca, por outro lado, tem como princípio central a comparação do genoma estudado com RNAs mensageiros (mRNAs) e sequências de proteínas de genomas similares depositados em bancos de dados biológicos. Informações de alta qualidade sobre proteínas e transcritos podem dar grandes indícios da presença e da localização dos genes, além de auxiliar na predição das suas informações estruturais. Para tanto, os *softwares* de predição extrínseca utilizam algoritmos de busca e alinhamento, como o BLAST, para buscar regiões de similaridade entre as sequências alvo e as sequências candidatas, podendo resultar em alinhamentos completos ou parciais. A evidência da presença de um potencial gene, portanto, é dada pelo alto grau de similaridade da sequência analisada com outra sequência extrínseca de mRNA ou proteína. Uma vez que esta abordagem baseia-se completamente em alinhamentos de sequências, é notável que a qualidade dos resultados esteja diretamente relacionada com a acuracidade das informações depositadas nos bancos de dados alvo. Além disso, uma vez que organismos eucarióticos multi-celulares têm apenas um subgrupo de genes expressos em um determinado momento (em tecidos e em ambientes específicos), podem existir situações nas quais não há informações extrínsecas suficientes para comparação (WANG *et al.*, 2017). O *software* FGENESH+ (SOLOVYEV; BIOINFORMATICS, 2004) é um exemplo de ferramenta que utiliza o método de predição extrínseca.

A predição combinada, entretanto, têm sido a forma mais eficaz de predição de genes, integrando as vantagens de ambas as abordagens intrínseca e extrínseca. Os métodos de ambas foram combinados na forma de *pipelines* ou *workflows* de anotação genômica que, em geral, iniciam com uma fase *ab initio* e têm seus resultados complementados por alinhamentos com referências externas. Exemplos de *softwares* e *pipelines* que incluem abordagens intrínsecas e extrínsecas incluem desde a adição de análises de evidências externas ao *software* AUGUSTUS (STANKE *et al.*, 2008; STANKE; TZVETKOVA; MORGENSTERN, 2006) até ferramentas como FGENESH++ (SOLOVYEV *et al.*, 2006), MAKER (CANTAREL *et al.*, 2008) e BRAKER (HOFF *et al.*, 2019).

O fator mais importante a ser considerado ao selecionar o preditor mais eficiente para um determinado organismo é a acuracidade, que pode ser medida através das métricas de sensibilidade (Sn) e especificidade (Sp). *Sensibilidade* é a proporção de verdadeiros positivos, isto é, nucleotídeos que foram preditos como codificantes e realmente são codificantes. A *especificidade*, por outro lado, é a proporção de verdadeiros negativos, ou seja, nucleotídeos que foram preditos como não codificantes e realmente são não codificantes. Além do nível de nucleotídeos, as métricas de sensibilidade e especifidade também podem ser utilizadas em níveis de éxons, transcritos e genes. A nível de éxons, a predição é considerada correta quando os dois pontos de corte estão corretos. A nível de transcritos, a predição é considerada correta quanto todos os éxons são preditos corretamente, sem a inclusão de éxons adicionais ou faltantes. Por fim, a nível de gene, a predição é considerada correta somente quando qualquer um de seus transcritos está correto, ou seja, quando pelo menos uma isoforma do gene é exatamente igual à anotação de referência (WANG *et al.*, 2017; ABRIL; CASTELLANO, 2019).

2.4.3 Anotação funcional

Após a anotação estrutural, inicia-se o processo de anotação funcional, que consiste em realizar a inferência *in silico* de um significado biológico para as proteínas que os genes codificam. O objetivo principal desta fase é a atribuição de nomes aos produtos gênicos, geralmente baseados na caracterização funcional dos genes preditos. A grande quantidade de sequências depositadas nos bancos de dados demandam análise funcional, com o intuito de compreender os processos biológicos dos organismos estudados e guiar novas pesquisas (MCDONNELL; STRASSER; TSANG, 2018). Muitas *pipelines* e *workflows* foram desenvolvidos para automatizar a anotação funcional, sendo especialmente relevantes no contexto da NGS, dada a capacidade atual de sequenciar, montar e anotar genomas completos em curtos períodos de tempo como, por exemplo, em menos de um mês (ANGEL *et al.*, 2018; CRUZ *et al.*, 2019).

A anotação funcional é realizada, basicamente, através de comparações. As ferramentas computacionais utilizadas neste processo fazem uso de algoritmos de busca de similaridade, tais como BLAST e HMMER, comparando as sequências de interesse com outras sequências presentes em repositórios públicos de proteínas revisadas, tais como o Swissprot, por exemplo. Antes

da atribuição de nomes aos produtos gênicos, é importante caracterizar elementos funcionais adicionais, sendo que as três abordagens gerais utilizadas neste processo incluem: 1) caracterizar elementos proteicos, tais como como termos GO, domínios, famílias, presença de peptídeo sinal (que sugere a secreção da proteína) e topologia transmembrana; 2) buscar informações sobre grupos de ortólogos e 3) detectar similaridades com proteínas já caracterizadas e revisadas (HARIDAS; SALAMOV; GRIGORIEV, 2018; ANGEL *et al.*, 2018).

Anotações funcionais de alta qualidade são caracterizadas por descrições detalhadas, precisas e úteis sobre o produto dos genes que codificam proteínas. Apesar da utilização de termos simples, como "Desidrogenase", ser aceitável, é recomendável definir nomes mais precisos e refinados como, para este mesmo caso, "Piruvato desidrogenase (NADP+)"(CRUZ *et al.*, 2019). Deve-se tomar cuidado ao copiar nomes de proteínas já caracterizadas meramente pela similaridade da sequência, já que sequências com baixo grau de identidade já podem ser consideradas homólogas, porém, não necessariamente possuem a mesma função, mesmo que dividam alguns domínios comuns. McDonnell, Strasser e Tsang (2018) apontam, por exemplo, que somente proteínas que possuem identidade maior ou igual a 98% sobre os seus tamanhos completos podem ser consideradas funcionalmente equivalentes e podem ter seus nomes copiados. Isto intensifica a importância de caracterizar e anotar informações adicionais, como famílias e grupos de proteínas relacionados ao produto do gene a ser anotado, além da busca por grupos de ortólogos.

Dentre as mais importantes caracterizações a serem utilizadas para enriquecer a anotação funcional está a busca por termos GO, que proporciona conhecimentos significativos sobre a função dos genes. O produto de um gene específico é anotado com os termos de maior granularidade possível na ontologia relacionada e, pelo princípio da transitividade, as anotações dos termos de nível superior ficam implícitas, até chegar nos níveis base, representados pelas três categorias principais: função molecular, componente celular e processo biológico. Cada produto de um gene pode ser anotado com zero ou mais termos de cada ontologia. É importante ressaltar que o fato de não serem encontrados termos GO para um determinado produto não tem relação com ausência de função, isto é, apenas significa que ainda não há evidências sobre o seu papel (ASHBURNER *et al.*, 2000; ACENCIO *et al.*, 2019). Amigo (CARBON *et al.*, 2009) é uma ferramenta que pode ser utilizada para realizar BLAST de sequências com o intuito de identificar termos GO.

Para a busca e identificação de domínios e famílias de proteínas, pode-se realizar buscas em bancos de dados como o Pfam, Prosite, PRINTS e SMART, por exemplo. Da mesma forma que os termos GO, vários números de acesso podem ser obtidos a partir de cada um dos banco de dados, cada qual representando um grupo ou classificação de proteínas, de acordo com domínios conservados identificados em partes da sequência. Na maioria dos casos, os parâmetros de *cut-off* (ou pontos de corte) que determinam os graus de similaridade que devem ser considerados para que uma determinado domínio seja identificado como verdadeiro são passíveis de configuração,

tornando os resultados sensíveis à alterações de parâmetros (CRUZ *et al.*, 2019). Para realizar consultas à estes bancos de dados, pode-se utilizar as próprias ferramentas fornecidas por cada um, sendo que parte do processo pode ser automatizado através do uso do banco de dados InterPro que, através da ferramenta InterProScan, permite a realização de consultas à diversas bases de dados de famílias e domínios de proteínas em um único contexto.

A busca pela presença de peptídeo sinal é muito relevante, dada a importância de compreender o direcionamento das proteínas para os compartimentos subcelulares e também para o meio extracelular (KOCH; MOSER; MÜLLER, 2003). Diversas soluções de bioinformática foram desenvolvidas para detectá-los, destacando-se a ferramenta SignalP (ARMENTEROS *et al.*, 2019b) que, em sua versão mais recente (SignalP-5.0), utiliza uma abordagem baseada em redes neurais profundas (*deep neural networks*). Esta solução, além de predizer a presença de peptídeo sinal para os reinos *Bacteria*, *Archaea* e *Eukarya*, é capaz de identificar a localização dos seus respectivos pontos de clivagem. É possível, ainda, predizer a topologia de regiões transmembrana através de ferramentas como o TMHMM (KROGH *et al.*, 2001), que utiliza modelos ocultos de Markov para predizer o total de hélices e as suas orientações de entrada e saída em relação à membrana.

A utilização de proteínas ortólogas revisadas pode proporcionar ao pesquisador um sólido embasamento para a anotação funcional de um genoma. Tendo como princípio a "conjectura da função", na qual "ortólogos realizam funções idênticas ou mais precisamente biologicamente equivalentes, em diferentes organismos"(GABALDÓN; KOONIN, 2013), esta abordagem é muito indicada para identificar similaridade de função. O método é apoiado por diversos bancos de dados e ferramentas que dão suporte à identificação de ortólogos, tais como o OrthoDB, o EggNOG o COG e o KEGG Orthology.

Por fim, a anotação funcional pode ser complementada com a identificação de características específicas do reino ou filo do organismo a ser estudado. Especificamente para fungos, por exemplo, pode-se utilizar o dbCAN2 para a predição de enzimas Cazy, o Kinannote (GOLDBERG *et al.*, 2013) para a predição de quinases, PredGPI (PIERLEONI; MARTELLI; CASADIO, 2008) para a predição de proteínas ancoradas por Glicosilfosfatidilinositol (GPI), BioV Suite (REDDY; JR, 2012) para a predição de transportadores, antiSMASH (BLIN *et al.*, 2017) para a predição de *clusters* de metabólitos secundários, BLAST no banco de dados MEROPS para a predição de peptidases e SUPERFAMILY para a predição de fatores de transcrição.

As *pipelines* e *workflows* automatizados de anotação funcional têm por objetivo executar diversas das ferramentas citadas e agrupar os seus resultados, com o intuito de facilitar o processo como um todo. Dentre elas, destaca-se a ferramenta Blast2GO (CONESA *et al.*, 2005), que tornou-se um *software* comercial e agora é parte do módulo de análise funcional do produto *OmicsBox*, da empresa BioBam[9]. A ferramenta InterProScan, inclusive, por agrupar resultados de diversos bancos de dados, também pode ser considerada uma *pipeline* de anotação. Pode-se

[9] https://www.biobam.com

destacar, ainda, a solução web GO FEAT (ARAUJO *et al.*, 2018), o *software* Trinotate (BRYANT *et al.*, 2017) para a anotação de transcriptomas e a *pipeline* Funannotate (PALMER; STAJICH, 2020), especializada inicialmente apenas em anotação de fungos, mas posteriormente evoluída para permitir genomas maiores. Além de possuir *workflows* de anotação funcional, esta última *pipeline* também engloba processos de predição e comparação de genomas.

2.4.4 Curadoria manual

Os rápidos avanços na tecnologia de sequenciamento de genomas, paradoxalmente, tornaram a anotação genômica menos (e não mais) precisa, mesmo tratando-se do processo mais importante na direção do entendimento da biologia dos genomas. Os principais desafios podem ser divididos em duas categorias: 1) a anotação automatizada de genomas *draft* ainda é muito difícil, em função da grande quantidade de ferramentas e bancos de dados necessários no processo e 2) erros e contaminações nos *assemblies draft* levam à consequentes erros de anotação e tendem a se propagar através das espécies. Além disso, quanto mais genomas *draft* são produzidos, mais erros podem ser criados e propagados (SALZBERG, 2019).

Encontrar genes em bactérias é considerado uma tarefa relativamente fácil, pelo principal motivo de que cerca de 90% dos genomas bacterianos são compostos por regiões codificantes de proteínas. A tarefa de encontrar genes, neste caso, é basicamente decidir qual dos seis possíveis *reading frames* contém uma proteína, sendo que os *softwares* atuais aproveitam este conceito e produzem resultados altamente precisos. Em eucariotos, por outro lado, o problema torna-se muito mais difícil, uma vez que os genes são poucos e muito distantes, além de serem interrompidos por íntrons. Incorporar o sequenciamento de RNA na busca pelos genes é uma solução para este problema, porém, possui algumas ressalvas, uma vez que alguns genes são expressos em baixos níveis e em apenas tecidos específicos. Além disso, genomas *draft* podem ter milhares de *contigs* desconectados, ou seja, muitos genes podem estar fragmentados entre os *contigs*, com ordens e orientações desconhecidas. Todas estas dificuldades, ainda, podem ser somadas à erros no próprio *assembly* (relacionados à contaminações por transferências horizontais de genes, por exemplo), demonstrando a complexidade do processo de anotação como um todo (SALZBERG, 2019).

Tanto a anotação estrutural quanto a anotação funcional são processos propensos à erros. É evidente, portanto, a necessidade de uma revisão aprofundada dos resultados e, muitas vezes, de re-anotação, através de um processo denominado curadoria manual. Trata-se de uma tarefa que exige esforço e tempo, através da qual o pesquisador identifica possíveis genes problemáticos ou suspeitos, em função da presença de domínios específicos, ortologia duvidosa ou até a ausência de quaisquer elementos funcionais em determinados genes, sugerindo genes fragmentados, erros de *assembly* ou até genes inexistentes, que não são funcionais ou foram anotados por engano. Para melhorar os resultados, inclusive, é possível realizar várias pesquisas e combinar os resultados obtidos de diversas fontes para gerar uma anotação consensual, dado o crescente

número de sequências em repositórios públicos. É importante, ainda, adotar critérios rigorosos de nomenclatura de produtos gênicos, a fim de evitar a propagação de erros. Apesar de trabalhoso, este é um processo altamente necessário na tentativa de elevar o grau de confiabilidade da anotação (ANGEL *et al.*, 2018).

A curadoria manual pode ser um processo altamente trabalhoso no que se refere à edição das informações anotadas. As ferramentas de anotação funcional geram resultados em formatos específicos que, mesmo seguindo padrões utilizados em dados biológicos (GFF3, FASTA, GenBank), são de difícil edição e visualização em um único contexto. Algumas ferramentas foram desenvolvidas para agrupar as informações de anotação e editá-las em um ambiente único. Dentre elas, destaca-se a ferramenta Apollo (DUNN *et al.*, 2019), uma solução que permite que os pesquisadores realizem *upload* dos dados de anotação provenientes de diversas fontes e realizem inspeções e edições na estrutura e função dos elementos genômicos. A primeira versão deste sistema era uma aplicação *desktop* independente e, posteriormente, evoluiu para uma aplicação *web* colaborativa, na qual diversos pesquisadores podem trabalhar em conjunto, com suporte à edições em tempo real.

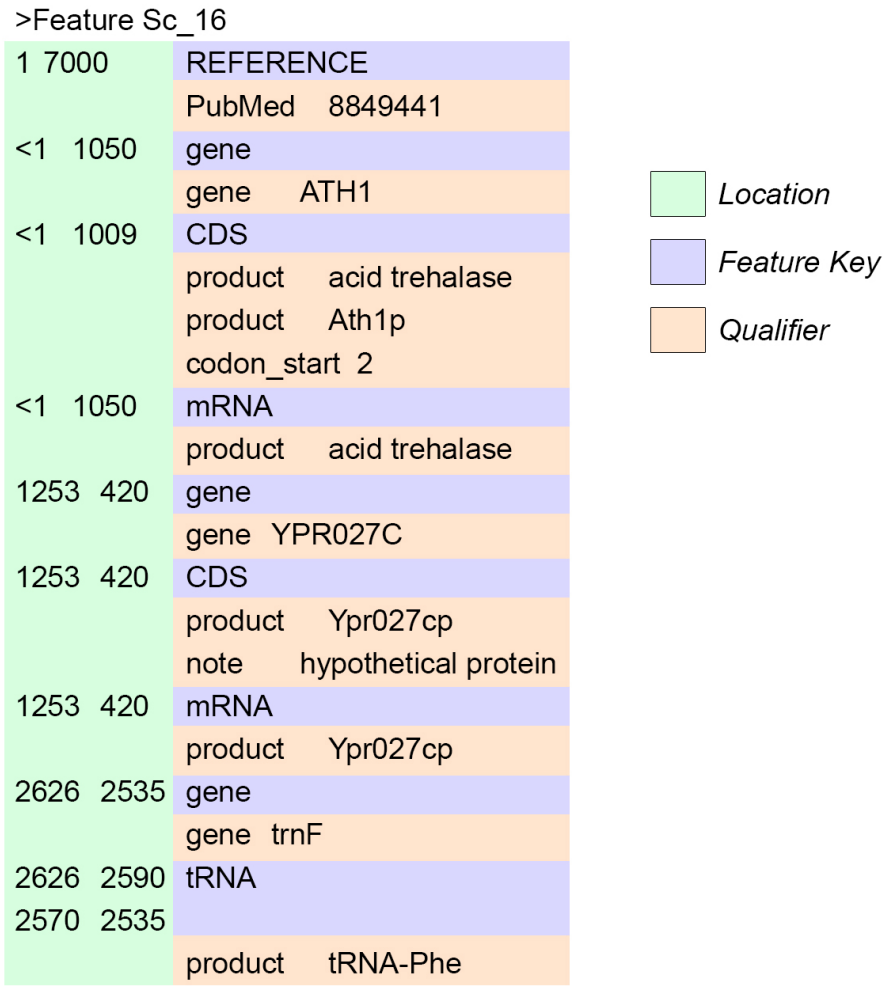
2.5 PUBLICAÇÃO

Os resultados dos projetos de anotação genômica podem ser submetidos a bancos de dados públicos (como o GenBank), com o intuito de divulgar a pesquisa e disponibilizar os dados para a comunidade científica. Para realizar submissões a estas plataformas, é necessário converter os dados de anotação para formatos específicos exigidos por cada uma, além de executar ferramentas de validação que buscam por possíveis erros e visam garantir que as informações submetidas estejam corretas. (GEIB *et al.*, 2018).

Para realizar submissões no banco de dados GenBank, por exemplo, os dados de anotação estrutural e funcional devem ser primeiramente convertidos para um formato denominado *feature table format* (TBL), ilustrado na Figura 8. O *design* deste formato é baseado em uma abordagem tabular, na qual todos os itens do genoma (mRNAs, tRNAs, rRNAs, etc) são considerados *features* e compostos por três componentes básicos: 1) *Feature Key* é a especificação do grupo funcional (gene, CDS, mRNA, tRNA, etc); 2) *Location* é a instrução de como encontrar a *feature* no genoma e 3) *Qualifiers* são as informações auxiliares sobre a *feature*, sendo que dentre as informações contidas neste item estarão os dados de anotação funcional. O arquivo exige que sejam seguidas regras, padrões e convenções de vocabulário, para garantir que a sincronização diária com os outros bancos de dados DDBJ e EMBL ocorra de forma correta (BENSON *et al.*, 2018).

Após a preparação do arquivo TBL, é necessário executar a ferramenta *tbl2asn*, um software de linha de comando fornecido pelo próprio NCBI que, além de converter o arquivo para o formato final a ser utilizado no momento da submissão (SQN), realiza inúmeras buscas por discrepâncias ou anotações suspeitas de importante análise posterior. Exemplos de erros

Figura 8 – Exemplo de parte de um arquivo em formato TBL.



Fonte: Adaptada de NCBI (2020).

que podem ser identificados e podem impedir a submissão incluem nomes de produtos gênicos suspeitos ou fora de padrão, íntrons muito pequenos, estruturas de íntrons e éxons que estejam fora dos limites do *contig*, éxons sobrepostos, entre outros.

Com o *assembly* de um genoma composto por centenas de sequências, cada qual demarcada com suas anotações estruturais e funcionais, é notável a necessidade de automatização na tarefa de preparação dos arquivos para submissão. Deixando de lado as soluções comerciais ou as desenvolvidas internamente por alguns laboratórios, a abordagem tradicional é converter os arquivos através do uso de *scripts* e bibliotecas, por intermédio de linguagens de programação como Python e Perl (COCK *et al.*, 2009; STAJICH *et al.*, 2002), por exemplo. Entretanto, algumas soluções gratuitas e mais robustas foram desenvolvidas, propondo a geração automática de todos ou parte dos arquivos necessários no processo de submissão, tendo como base arquivos comumente de posse do pesquisador após o processo de anotação genômica. Pode-se citar como exemplo a ferramenta *Genome Annotation Generator* (GAG) (GEIB *et al.*, 2018) que, apesar de se tratar de um *software* de linha de comando, permite a geração do arquivo TBL, a execução

automática da ferramenta *tbl2asn* e, inclusive, a correção de alguns erros de validação identificados pelo *tbl2asn*. O *software* recebe como entrada o arquivo FASTA do genoma (produzido por grande parte dos *assemblers*), o arquivo GFF3 (contendo anotações estruturais e até funcionais) e, de forma opcional, um arquivo tabular de anotações funcionais.

2.6 FERRAMENTAS RELACIONADAS

É notável a quantidade de ferramentas necessárias para realizar o processo de anotação como um todo. Tratando-se especificamente da anotação funcional, para cada gene do genoma, é preciso consultar diversos bancos de dados na busca por elementos proteicos, ortologia e homologia, na tentativa de dar nome aos produtos gênicos. Além disso, cada ferramenta a ser utilizada possui suas próprias características, regras de utilização e arquivos de saída específicos. A execução de todos os *softwares* para cada gene do genoma de forma individual torna-se inviável, especialmente para projetos de genomas completos. Tratando-se de grandes quantidades de genes, alguns dos bancos de dados exigem o *download* e execução local da ferramenta para executar trabalhos em lote, sendo necessária a configuração de parâmetros e execução via linha de comando, o que pode estar além das habilidades do pesquisador e dificultar o trabalho.

Todas estas evidências confirmam a necessidade de criação de soluções que executem automaticamente as consultas a bancos de dados externos para todos os genes do genoma e agrupem os resultados em um único contexto, sem a necessidade de execução de *softwares* de forma paralela. Estas soluções devem permitir, ainda, a realização de curadoria manual através da edição de informações estruturais e funcionais dos genes e, quando a anotação estiver concluída, disponibilizar os arquivos prontos para publicação. As ferramentas aqui relacionadas, portanto, estão dentro deste contexto. Alguns critérios de análise foram definidos, com base no contexto da presente pesquisa, sendo eles: 1) Acesso gratuito; 2) Acesso *web* e *online*; 3) Possibilidade de executar ferramentas de anotação funcional; 4) Possibilidade de dividir o trabalho em projetos; 5) Possibilidade de compartilhar projetos e trabalhar simultaneamente; 6) Curadoria manual: Possibilidade de editar dados de anotação funcional; 7) Curadoria manual: Possibilidade de editar dados de anotação estrutural; 8) Possibilidade de exportar resultados e 9) Possibilidade de exportar resultados para publicação no banco de dados GenBank. O Quadro 4 apresenta uma comparação destes critérios para todas as ferramentas analisadas.

A ferramenta *Gene Ontology Functional Enrichment Annotation Tool* (GO FEAT) (ARAUJO *et al.*, 2018) é uma plataforma *web* para anotação funcional, com código *open-source* sob os termos da licença MIT. Uma instância pública do *software* está disponível *online*, porém, é possível realizar o *download* do código fonte e executar a solução em um servidor local. A ferramenta permite o gerenciamento e compartilhamento de projetos, sendo que cada um recebe como entrada arquivos FASTA que contenham sequências de proteínas ou de nucleotídeos. Uma vez importadas as sequências, o *software* inicia as etapas de busca por homologia e busca por domínios, exibindo resultados de alinhamentos com UniProt, NCBI Protein, KEGG,

Quadro 4 – Critérios de comparação de ferramentas relacionadas.

Critério	**GOFEAT**	**OmicsBox**	**GenSAS**	**RAST**
1- Acesso gratuito	Sim	Não	Sim	Sim
2- Acesso web e online	Sim	Não	Sim	Sim
3- Possibilidade de executar ferramentas de anotação funcional	Sim	Sim	Sim	Sim
4- Possibilidade de dividir o trabalho em projetos	Sim	Sim	Sim	Sim
5- Possibilidade de compartilhar projetos e trabalhar simultaneamente	Sim	Não	Sim	Sim
6- Curadoria manual: Possibilidade de editar dados de anotação funcional	Não	Não	Não	Não
7- Curadoria manual: Possibilidade de editar dados de anotação estrutural	Não	Não	Sim	Não
8- Possibilidade de exportar resultados	Sim	Sim	Sim	Sim
9- Possibilidade de exportar resultados para publicação no banco de dados GenBank	Não	Sim	Não	Não

Fonte: Elaborada pelo autor.

Interpro, Pfam, Gene Onthology e SEED (OVERBEEK *et al.*, 2005). Observa-se, entretanto, a impossibilidade de edição das informações após a análise, não sendo permitido alterar quaisquer informações sem a criação de um novo projeto. Este quesito, somado à ausência total de importação e gerenciamento de dados estruturais, impossibilita o agrupamento de informações de anotação em um único contexto e, principalmente, a realização de curadoria manual dentro da plataforma. O *software* possui uma interface amigável, exibindo gráficos analíticos e possibilitando a exportação dos resultados para diversos formatos. Por outro lado, a ferramenta não permite a exportação das informações diretamente para publicação no banco de dados GenBank, sendo que a inexistência desta funcionalidade pode ser justificada pela ausência do gerenciamento de informações estruturais dos genes.

A ferramenta OmicsBox é um produto comercial da empresa BioBam. Trata-se de uma plataforma completa de bioinformática, englobando desde a montagem de genomas até as etapas de anotação estrutural e anotação funcional. A solução é divida em quatro módulos principais: O módulo *Genome Analysis* inclui a análise de qualidade e montagem de genomas, identificação de regiões repetitivas e busca de genes; O módulo *Transcriptomics* engloba o estudo de transcriptomas, incluindo análise de qualidade, montagem, alinhamentos de RNAs e análise de expressão diferencial; O módulo *Functional Analysis* é composto pela ferramenta Blast2GO, que é também parte do produto comercial e possui uma versão gratuita. O módulo engloba a criação de *workflows* personalizados de anotação funcional, incluindo a execução de BLAST, InterProScan, Gene Onthology e EggNOG, sendo possível visualizar informações estatísticas e árvores dos termos GO obtidos. Por fim, o módulo *Metagenomics* possui todas as ferramentas necessárias para uma análise completa de microbiomas, permitindo classificação taxonômica, montagem e predição de genes em metagenomas. Diversas ferramentas adicionais também estão

disponíveis na solução, destacando-se aqui o *Genome Browser*, através do qual é possível ter uma visualização completa do genoma e identificar a presença de todas as estruturas e características anotadas. A solução ainda permite a exportação dos resultados para diversos formatos, incluindo os necessários para publicação no banco de dados GenBank. Como pontos negativos, destaca-se o acesso não gratuito, a ausência de versão *web* e *online*, a impossibilidade de compartilhar projetos para trabalho em equipe simultaneamente e a ausência de funcionalidades de edição manual das informações estruturais e funcionais.

O *software Genome Sequence Annotation Server* (GenSAS) (HUMANN *et al.*, 2019) é uma solução *web* gratuita de anotação genômica estrutural e funcional que engloba uma grande quantidade de ferramentas. Através de um guia passo-a-passo intuitivo e explicativo, o usuário inicia o processo realizando o *upload* das sequências do *assembly* do genoma em formato FASTA, com a opção de enviar arquivos adicionais contendo outras evidências que já estejam eventualmente disponíveis como, por exemplo, genes já preditos por outra ferramenta em formato GFF3. Caso o usuário somente tenha realizado o *upload* do arquivo do genoma, o próximo passo é a anotação estrutural, englobando a identificação de regiões repetitivas e a predição de genes, podendo o usuário optar pela execução das ferramentas AUGUSTUS, BRAKER, GeneMark-ES (TER-HOVHANNISYAN *et al.*, 2008), Genscan (BURGE; KARLIN, 1998), GlimmerM (PERTEA; SALZBERG, 2003) e SNAP (KORF, 2004). Posteriormente, na etapa de anotação funcional, o usuário pode optar por executar BLAST ou DIAMOND (BUCHFINK; XIE; HUSON, 2015) nos bancos de dados SwissProt, TrEMBL e NCBI RefSeq, além de realizar a busca por elementos proteicos através de Pfam, InterProScan, SignalP e TargetP (ARMENTEROS *et al.*, 2019a). Em função da integração nativa com a ferramenta Apollo é possível, ainda, realizar a visualização do genoma e a curadoria manual das informações estruturais. No que se refere à publicação do genoma, a ferramenta permite a exportação dos dados de anotação para os formatos GFF3, FASTA e outros formatos de texto. Como desvantagens, observa-se a ausência da opção de exportar os resultados diretamente para os formatos de publicação do banco de dados GenBank e a impossibilidade de realizar curadoria manual nas informações funcionais (somente através da importação manual de arquivos, não entrando nos critérios de avaliação estabelecidos).

A ferramenta *Rapid Annotations using Subsystems Technology* (RAST) (AZIZ *et al.*, 2008) é uma solução *web* gratuita de anotação estrutural e funcional de genomas procarióticos, criada em 2008. O principal diferencial deste sistema em relação aos demais se dá no fato de que este utiliza como base principal de comparação o banco de dados SEED, cujos dados são organizados em *subsistemas* ou, em outras palavras, conjuntos de funções logicamente relacionadas. A acurácia e consistência dada por esta metodologia tornaram esta ferramenta uma das mais populares para anotação de genomas microbianos. Apesar das vantagens de possuir sua própria *pipeline* de anotação padrão, foi criado, em 2015, o módulo adicional *RAST tool kit* (RASTtk) (BRETTIN *et al.*, 2015), permitindo que os usuários realizem customizações na *pipeline*, selecionado quais ferramentas desejam executar e até importando suas próprias

anotações de fontes externas. Apesar de ser possível editar manualmente a função de um gene e importar dados de anotação de fontes externas, não considerou-se que é possível editar dados de anotação estrutural e funcional diretamente através da plataforma. Observou-se ainda, a carência de consulta ao banco de dados Pfam e aos bancos de dados EggNOG e OrthoDB. A solução permite que o pesquisador divida e trabalho em projetos e compartilhe com outros usuários, para trabalho simultâneo. É possível exportar os dados de anotação para diversos formatos, inclusive os formatos utilizados pelo banco de dados GenBank, porém, não é possível exportar os resultados para os formatos exigidos no momento da publicação.

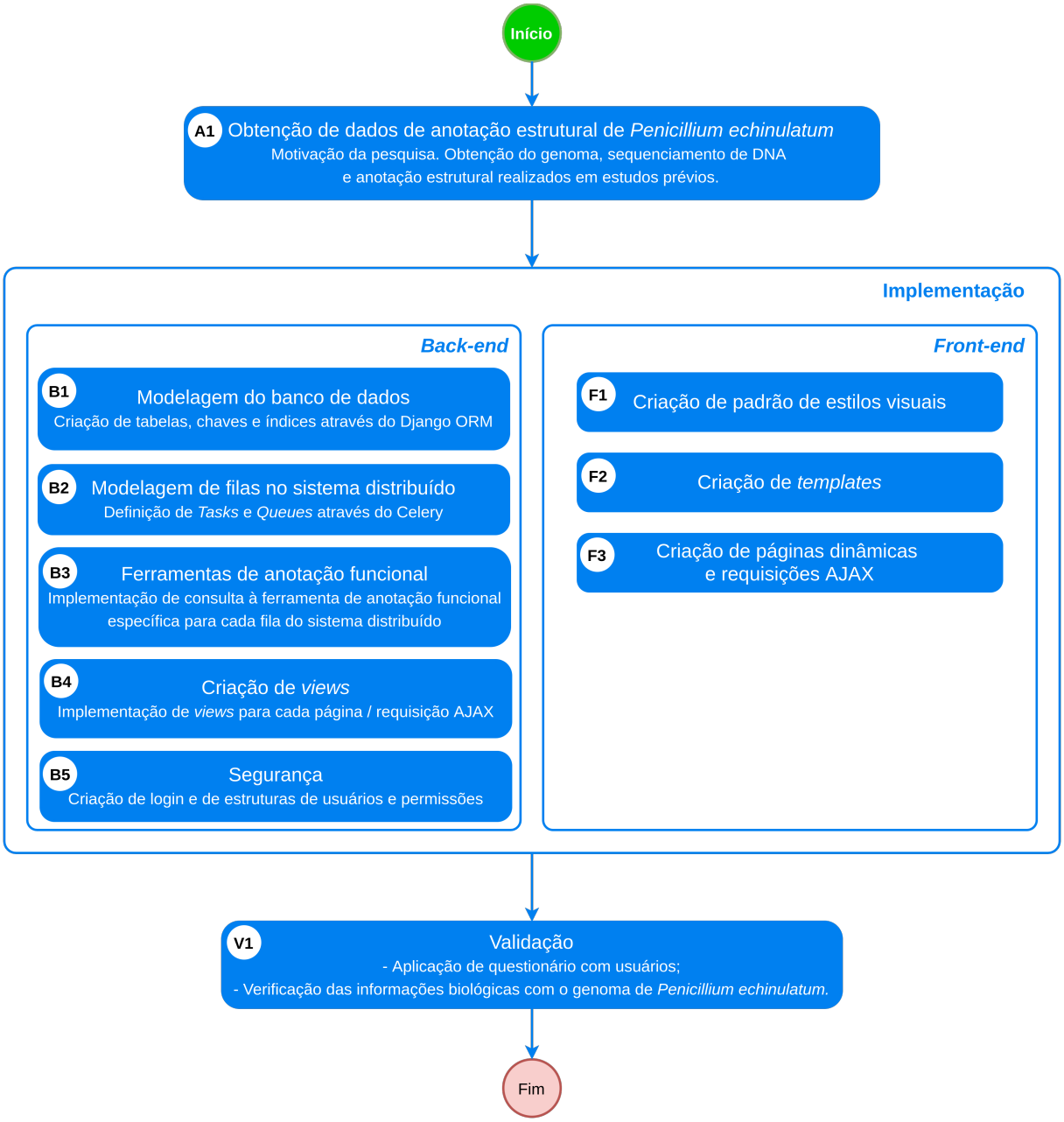
Outras soluções passíveis de comparação não foram consideradas por se enquadrarem em pelo menos um dos seguintes casos: 1) Ausência de qualquer interface com o usuário, exigindo a execução por linha de comando ou 2) Necessidade de extensa configuração de parâmetros ou necessidade de instalação de grande quantidade de dependências, como bibliotecas, módulos e ferramentas. Entende-se que, ao se enquadrarem nestes casos, tais ferramentas afastam-se do contexto desta pesquisa por não proporcionarem ao usuário um contexto único de gerenciamento das informações de anotação ou exigirem do pesquisador conhecimentos avançados de informática e esforços adicionais para instalação de bibliotecas. Apesar disso, muitas apresentam resultados satisfatórios, destacando-se as ferramentas Trinotate, Funannotate, DFAST (TANIZAWA; FUJISAWA; NAKAMURA, 2018), Megannotator (LUGLI *et al.*, 2016) e KAAS (MORIYA *et al.*, 2007).

A análise dos processos envolvidos em todas de todas as fases que envolvem um Projeto Genoma, desde o sequenciamento até a publicação, reforçam a necessidade de apoiar os pesquisadores através da criação de soluções que abstraiam a complexidade existente nas mais variadas etapas. Especialmente na anotação funcional, a grande quantidade de bancos de dados envolvidos e os arquivos de saída independentes, provenientes das ferramentas de anotação, justificam o desenvolvimento de uma solução para organizar o processo em um ambiente único. Pelo fato de todas as soluções analisadas não possuírem ao menos duas das funcionalidades definidas nos critérios de análise, o desenvolvimento de uma nova solução foi justificado.

3 METODOLOGIA

A Figura 9 apresenta o fluxo de etapas adotado na metodologia. O detalhamento de cada um dos processos ilustrados está descrito nas subseções deste capítulo.

Figura 9 – Fluxo de etapas adotado na metodologia.



Fonte: Elaborada pelo autor.

3.1 OBTENÇÃO DOS DADOS DE ANOTAÇÃO ESTRUTURAL DE *PENICILLIUM ECHINULATUM*

As fases de obtenção do genoma, sequenciamento de DNA e anotação estrutural do fungo *Penicillium echinulatum* aqui descritas (Figura 9, A1) foram realizadas em estudos prévios e não fazem parte do escopo da presente pesquisa. O *workflow* apresentado neste trabalho teve como motivação realizar as etapas de anotação funcional e curadoria manual que se mostraram necessárias após a realização das etapas apresentadas a seguir.

A linhagem selvagem 2HH de *Penicillium* foi isolada em 1979 do trato digestório da larva do coleóptero *Anobium punctatum* no sul do Brasil (CARRAU *et al.*, 1981). Este fungo foi previamente classificado por morfologia como *Penicillium echinulatum* nos anos 80. O mutante S1M29 foi obtido através de outro mutante 9A02S1 por mutagênese, através do emprego de peróxido de hidrogênio e seleção dos mutantes em um meio suplementado com 2-deoxiglicose (DILLON *et al.*, 2011). S1M29 é o melhor mutante até o momento, resultado de aprimoramentos de linhagem que acumularam diversas mutações (SCHNEIDER *et al.*, 2016; SCHNEIDER *et al.*, 2018). Este mutante proporciona uma melhor hidrólise de biomassa, em função de um significante aumento nos títulos enzimáticos.

Para a fase de sequenciamento, DNA de alto peso molecular foi extraído de ambas as linhagens selvagem 2HH e mutante S1M29, através da utilização de um protocolo para isolamento de DNA (SAMBROOK *et al.*, 1989). Este foi utilizado para gerar bibliotecas para a plataforma *Illumina* através de um protocolo não modificado *Illumina TruSeq* (ILLUMINA, 2012). Ambos os genomas foram sequenciados utilizando a plataforma *Illumina HiSeq 2000*, conduzidos pela empresa Ambry Genetics[1] em 2013. *Reads* de 100 pb sequenciadas em ambas as extremidades (*paired-end*) foram obtidas para cada linhagem, totalizando 29.316.764.800 pb com *raw coverage* de 961x para a linhagem S1M29 e 25.514.587.400 pb com *raw coverage* de 836x para a linhagem 2HH. Em 2014, o sequenciamento de RNA da linhagem S1M29 foi realizado para auxiliar na posterior predição de genes, sendo que as bibliotecas foram construídas utilizando o protocolo *Illumina TruSeq* (ILLUMINA, 2014) com seleção de poly-A.

O *software* FastQC foi utilizado para avaliação da qualidade das *reads*, seguido da ferramenta Trim Galore!, que removeu pequenos fragmentos (30 pb), adaptadores e bases de baixa qualidade. As *reads* de alta qualidade resultantes foram utilizadas na montagem do genoma, que deu-se através do *assembler* multi-k SPAdes (BANKEVICH *et al.*, 2012), sendo que o melhor valor de k foi predito pelo *software* KmerGenie (CHIKHI; MEDVEDEV, 2014). O *software* Quast foi utilizado para avaliar estatisticamente o genoma, resultando em 697 *scaffolds*, tamanho 30.43 Mb e N50 igual a 151.4 Kb para a linhagem 2HH e 673 *scaffolds*, tamanho 30.41 Mb e N50 igual a 185.175 Kb para a linhagem S1M29. Além disso, o percentual de GC para ambos os *assemblies* resultou em 50.3%. A avaliação de completude da ferramenta BUSCO, utilizando os

[1] https://www.ambrygen.com

datasets Eukaryota (303 genes), *Fungi* (290 genes), *Dikarya* (1.312 genes), *Ascomycota* (1.315 genes) e *Eurotiomycetes* (4.046 genes) resultou em um percentual de completude médio de mais de 98% para ambas as linhagens.

Na fase de anotação estrutural, foram utilizadas as ferramentas RepeatMasker e RepeatModeler (SMIT; HUBLEY; GREEN, 2013-2015) para identificação de regiões repetitivas. A predição de genes foi realizada independentemente por um grupo de ferramentas de busca de genes, sendo que os modelos de genes consenso para cada uma das linhagens foram gerados pelo *software EVidence Modeler* (HAAS *et al.*, 2008). O grupo de preditores de genes utilizados inclui os preditores *ab-initio* Augustus e GeneMark-ES (TER-HOVHANNISYAN *et al.*, 2008), os preditores baseados em evidência BLAT (KENT, 2002), GMAP-GSNAP (WU *et al.*, 2016) e Exonerate (TER-HOVHANNISYAN *et al.*, 2008), além da ferramenta tRNAscan-SE (LOWE; CHAN, 2016) para a predição de tRNAs. A predição consenso resultou em 8.366 genes (sendo 188 tRNAs) para a linhagem 2HH e 8.375 genes (sendo 197 tRNAs) para a linhagem S1M29.

3.2 DESENVOLVIMENTO DA APLICAÇÃO

3.2.1 Implementação

Uma relação de todas as linguagens, tecnologias e bibliotecas utilizadas em cada contexto do desenvolvimento da aplicação pode ser observada no Quadro 5.

Quadro 5 – Linguagens, tecnologias e bibliotecas utilizadas no desenvolvimento da aplicação.

Nome da tecnologia	**Tipo da tecnologia**	**Contexto**	**URL de acesso**
Python (3.6)	Linguagem de programação	*Back-end*	https://www.pyhon.org
Django (2.1.7)	*Framework*	*Back-end*	https://www.djangoproject.com
Celery (4.3)	Sistema distribuído	*Back-end*	http://www.celeryproject.org
RabbitMQ (3.7)	*Message Broker*	*Back-end*	https://www.rabbitmq.com
BioPython (1.74)	Biblioteca	*Back-end*	https://biopython.org
PostgreSQL (9.4)	Banco de dados	*Back-end*	https://www.postgresql.org
HTML5	Linguagem de marcação	*Front-end*	https://www.w3.org/html
CSS3	Linguagem de estilos visuais	*Front-end*	https://www.w3.org/css
JavaScript	Linguagem de programação	*Front-end*	https://www.ecma-international.org /publications/standards/Ecma-262.htm
jQuery (3.3.1)	Biblioteca	*Front-end*	https://jquery.com
Bootstrap (4.3.1)	Biblioteca	*Front-end*	https://getbootstrap.com

Fonte: Elaborada pelo autor.

O *back-end*[2] do sistema foi desenvolvido na linguagem Python (3.6) (PYTHON, 2020). Esta é uma linguagem gratuita, interpretada, multiplataforma e extremamente poderosa, sendo largamente utilizada nas áreas de desenvolvimento de aplicações *web* e *desktop*, educação e

[2] Camada de um *software* que não é visível ao usuário, composta por serviços que realizam o processamento da lógica de negócio, isto é, atendem às requisições feitas pelos usuários e garantem o armazenamento de dados. Em um ambiente *web* cliente-servidor, estes serviços são executados em um ou mais servidores externos ao dispositivo do usuário, conectando-se à este através de uma rede local ou através da *internet* (YUNRUI, 2018; JADEJA; MODI, 2012).

computação científica. A linguagem Python, juntamente com as linguagens Perl (PERL, 2020) e R (R, 2020), está entre as linguagens mais utilizadas no contexto da bioinformática, sendo que existem bibliotecas e *scripts* disponíveis gratuitamente para auxiliar o pesquisador na manipulação de dados biológicos e na solução de problemas específicos da área.

O *framework*[3] Django (2.1.7) (DJANGO, 2020) foi utilizado com o intuito de proporcionar maior produtividade, solidez e segurança no processo de desenvolvimento (MACIEL, 2020). Sua estrutura principal é baseada em uma arquitetura *Model-Template-View* (MTV), que permite a divisão e organização do desenvolvimento em diferentes contextos. O *model* possui ferramentas para lidar com a criação e manipulação de dados no banco de dados. O *template* permite a criação de páginas modelo com variáveis específicas a serem substituídas posteriormente pela *view*. A *view*, por sua vez, é o contexto principal, no qual ocorre a implementação da lógica de negócio (Figura 9, B4), conectando o *model* ao *template*. Cada requisição do usuário possui uma *view* específica para atendê-la, buscando os dados solicitados no banco de dados e renderizando e enviando ao usuário um *template* com base nestes dados, permitindo a visualização do que foi solicitado.

O *framework* Django ainda permite a criação de aplicações *web* escaláveis de forma eficiente, uma vez que a maior parte dos detalhes relacionados à estrutura de aplicações *web* já estão previamente implementados, tais como controle de sessão, gerenciamento de usuários e permissões, autenticação, autorização e um módulo de persistência objeto-relacional (Django ORM). Este último módulo permite realizar interações de alto nível com o banco de dados, relacionando classes e objetos à tabelas e linhas do banco de dados, além de controlar de forma eficaz e objetiva as migrações do modelo do banco de dados entre diferentes versões do sistema. Estas implementações que já fazem parte do *core* do *framework* permitem que o desenvolvedor tenha maior foco na solução que deseja implementar, tendo menor esforço no desenvolvimento de funcionalidades que, apesar de necessárias, são comuns à quaisquer aplicações *web* (MACIEL, 2020).

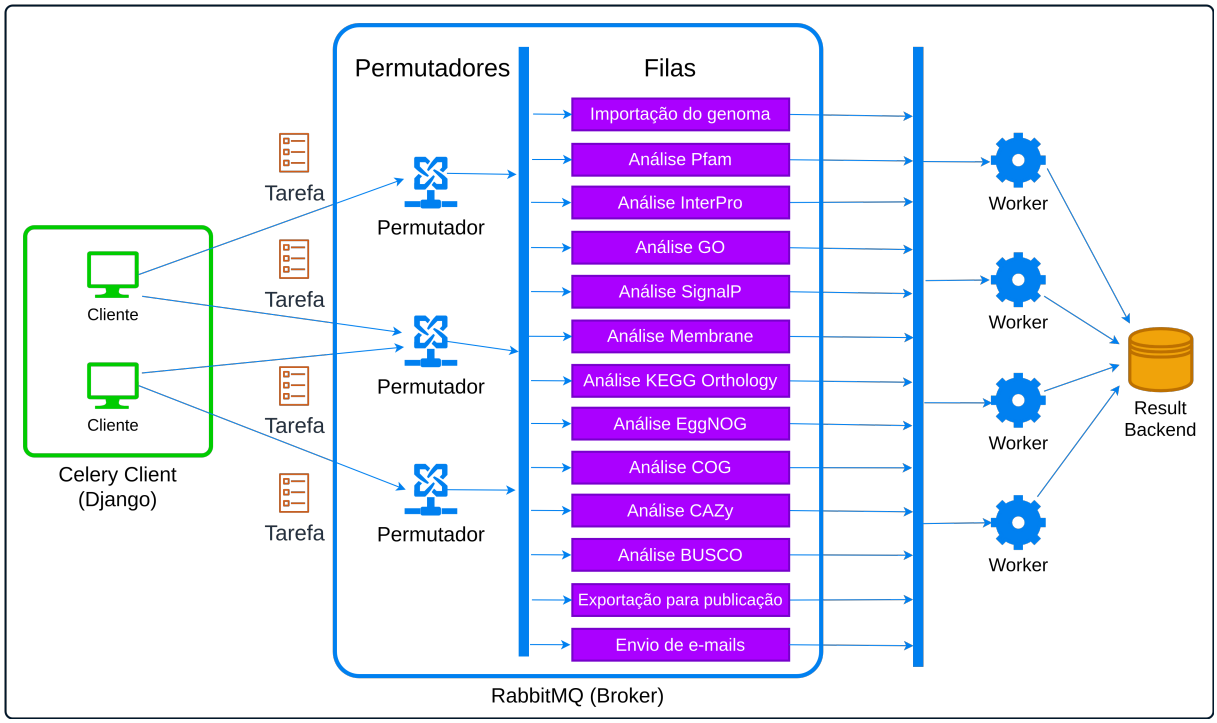
Considerando que um dos objetivos principais do sistema é a execução automática de ferramentas de anotação funcional para cada gene de um determinado projeto, é notável que o processamento de tais tarefas consuma recursos do servidor, principalmente se solicitadas por diversos usuários de forma simultânea. Além disso, algumas das ferramentas de anotação podem ser executadas remotamente através de comunicação via *web services* e podem levar diversos minutos ou até horas para executar (ARAUJO *et al.*, 2018), tanto por envolver alinhamentos complexos ou por sobrecarga do servidor remoto. Assim, há a necessidade de um mecanismo de agendamento de requisições de forma assíncrona (Figura 9, B2), com o intuito de controlar e limitar a execução simultânea de tarefas e evitar que o usuário necessite aguardar por requisições longas e possivelmente receba erros de *timeout*. Para atender a este requisito, foi utilizado o

[3] Conjunto de funções e componentes contendo arquitetura e estrutura internas básicas para o desenvolvimento de aplicações, permitindo o reaproveitamento de código e evitando a reprogramação desnecessária de tarefas comuns em diferentes sistemas (JOBSTRAIBIZER, 2009).

sistema distribuído[4] Celery (4.3) (CELERY, 2020).

Cada tarefa a ser executada pelo Celery é tratada pela solução através de unidades de trabalho denominadas *Tasks*, sendo que estas são adicionadas à filas específicas e consumidas por serviços de execução denominados *Workers*, que monitoram constantemente as filas verificando a existência de novas tarefas para serem executadas. Foram criadas filas diferentes para cada tipo de tarefa a ser executada de maneira assíncrona no sistema, conforme pode ser observado na Figura 10.

Figura 10 – Fluxo de execução de tarefas assíncronas.



Fonte: Elaborada pelo autor.

A comunicação entre a aplicação cliente, responsável pela criação das *Tasks*, e os *Workers* é realizada através de um mecanismo de troca de mensagens denominado *Message Broker* (SINGH; VINIOTIS, 2017), sendo que o *Broker* escolhido foi o RabbitMQ (3.7) (RABBITMQ, 2020). Dentro do *Broker*, os permutadores (ou *Exchanges*) realizam o roteamento das tarefas que chegam, direcionando-as para cada uma das filas, dependendo do tipo de permutação escolhido. A possibilidade de alocar diversos *Brokers* e *Workers* em um mesmo servidor ou em múltiplos servidores é um ponto chave no sistema, garantindo a escalabilidade horizontal da aplicação em caso da necessidade de aumento do poder computacional em função de um número maior de usuários ou projetos ao longo do tempo. A extensão *django-celery-results* (1.0.4)

[4] Conjunto de componentes computacionais que executam suas tarefas de maneira autônoma e se comunicam através de troca de mensagens. Estes componentes, entretanto, atuam de forma à serem apresentados ao usuário como um único sistema coerente. (STEEN; TANENBAUM, 2016).

(DJANGOCELERYRESULTS, 2020) foi utilizada para armazenar metadados do andamento das tarefas (*result backend*) em tabelas do banco de dados integradas ao *framework* Django.

Um mecanismo extra para impedir que um único usuário ou grupo de usuários de um mesmo projeto agendem um número excessivo de análises ao mesmo tempo também foi incluso no projeto. Uma vez que o sistema tem o conhecimento da quantidade de tarefas que estão nas filas e qual usuário requisitou cada uma, foi possível limitar o número de análises funcionais que podem estar simultaneamente agendadas e aguardando execução em um projeto. Isto tornou possível que pesquisadores de diferentes projetos tenham a mesma prioridade ao agendar análises funcionais, além de permitir um controle mais eficiente dos recursos de servidor que estão sendo utilizados. Esta e outras limitações que refletem na performance do servidor (o número máximo de projetos por usuário; o número máximo de colaboradores adicionais por projeto; o número máximo de genes por projeto e o número máximo de produtos gênicos por gene) são parâmetros dinâmicos que podem ser definidos pelos administradores do sistema na *interface* de administração do sistema, através de uma classe denominada Plano de Usuário (*UserPlan*).

A biblioteca BioPython (1.74) (COCK *et al.*, 2009) foi utilizada para lidar com os arquivos de formatos comumente utilizados na bioinformática, como o FASTA, o GFF3 e o GenBank. Além de interpretar corretamente estes formatos de arquivo e abstraí-los através de objetos da linguagem *Python*, a solução conta com uma coleção completa de ferramentas úteis para trabalhos *in silico* de biologia molecular, desde transcrição e tradução até alinhamentos de sequências e cálculos de massa molecular. Um exemplo prático de utilização da biblioteca no sistema desenvolvido é a geração da sequência de aminoácidos de uma proteína tendo como base o arquivo FASTA do genoma e as informações de início e término de cada éxon de um determinado gene.

O banco de dados utilizado foi o PostgreSQL (9.4) (POSTGRESQL, 2020) (Figura 9, B1). Este é um banco de dados objeto-relacional gratuito e *open-source* que proporciona armazenamento seguro de dados e possibilidade de escalonamento para lidar com complexas cargas de trabalho. O PostgreSQL pode ser executado na grande maioria dos sistemas operacionais e ganhou uma forte reputação ao longo do tempo pela sua confiabilidade e integridade de dados, proporcionando um ambiente seguro para administradores garantirem a proteção dos dados armazenados e criarem ambientes tolerantes a falhas.

No *front-end*[5] do sistema (Figura 9, F1, F2 e F3), além das linguagens *HyperText Markup Language* (HTML5), *Cascading Style Sheets* (CSS3) e JavaScript, foi utilizada a biblioteca jQuery (3.3.1) (JQUERY, 2020) e a biblioteca Bootstrap (4.3.1), (BOOTSTRAP, 2020). A primeira é uma coleção de funções em JavaScript para manipulação de documento, proporcionando

[5] Camada de um *software* responsável pela *interface* que é apresentada ao usuário, permitindo que este visualize as informações e realize interações. Em um ambiente *web* cliente-servidor, o código do *front-end* é executado no próprio navegador de *internet* utilizado pelo usuário (YUNRUI, 2018; JADEJA; MODI, 2012).

maior agilidade na manipulação de elementos da página e na realização de requisições em formato *Asynchronous Javascript and XML* (AJAX), através de uma *Application Programming Interface* (API) com suporte a múltiplos navegadores. A segunda é uma biblioteca que facilita a criação de páginas *web* responsivas por possuir classes CSS predefinidas e incorporar *plugins* em JavaScript para a exibição de componentes normalmente necessários em aplicações *web*, como alertas, janelas modais, abas, menus em *dropdown*, entre outros.

3.2.2 Ferramentas de anotação funcional selecionadas

O Quadro 6 apresenta as ferramentas de anotação funcional selecionadas e sua função no *workflow* de anotação funcional (Figura 9, B3). Para analisar domínios e famílias de proteínas, foram selecionados os bancos de dados Pfam, InterPro e GO. A consulta ao banco de dados Pfam foi realizada através do *download* e execução do pacote *stand-alone* da ferramenta HMMER (3.3.1) (FINN; CLEMENTS; EDDY, 2011). A ferramenta específica para busca por domínios de proteínas utilizada foi o *hmmscan*, com retorno de dados em formato *JavaScript Object Notation* (JSON) e parâmetros *–cut_ga*, *–acc*, *–tbl_out* <*caminho_arquivo_saída*>, <*caminho_arquivo_banco*> e <*caminho_arquivo_proteína*> para, respectivamente: 1) definir limites de *cut-off* selecionados automaticamente para cada família, não permitindo falsos positivos; 2) exibir preferivelmente *acession_numbers* no arquivo de saída; 3) especificar o caminho do arquivo de saída; 4) definir o caminho do arquivo do banco de dados Pfam e 5) definir o caminho do arquivo em formato FASTA contendo a sequência de aminoácidos da proteína a ser analisada.

Quadro 6 – Ferramentas de anotação funcional selecionadas.

Ferramenta	**Banco de Dados**	**Função**	**URL de acesso**
HMMER (3.3.1)	- Pfam	Predição de domínios e famílias de proteínas	https://www.ebi.ac.uk/Tools/hmmer/search/hmmscan
InterProScan (5.46)	- InterPro - GO	Predição de domínios e famílias de proteínas	https://www.ebi.ac.uk/interpro/search/sequence
SignalP (5.0)	- SignalP	Predição de peptídeo sinal	http://www.cbs.dtu.dk/services/SignalP
TMHMM (2.0)	- TMHMM	Predição de topologia transmembrana	http://www.cbs.dtu.dk/services/TMHMM
eggNOG-mapper (v2)	- KEGG Orthology - EggNOG - COG	Predição de grupos de ortólogos	http://eggnog-mapper.embl.de
eggNOG-mapper (v2)	- CAZy	Predição de enzimas CAZy	http://eggnog-mapper.embl.de
BUSCO (v3)	- OrthoDB	Análise de completude do proteoma	https://busco.ezlab.org

Fonte: Elaborada pelo autor.

Os bancos de dados InterPro e GO foram consultados através da ferramenta InterProScan (5.46), através do *download* do pacote *stand-alone* para execução no servidor local. Os parâmetros *-i* <*caminho_arquivo_proteína*>, *-b* <*caminho_arquivo_saída*>, e *-f json* foram utilizados para, respectivamente: 1) definir o caminho do arquivo em formato FASTA contendo a sequência

de aminoácidos da proteína a ser analisada; 2) especificar o caminho do arquivo de saída e 3) especificar o formato JSON para a geração do arquivo de saída.

As ferramentas SignalP (5.0) e TMHMM (2.0) foram selecionadas para realizar predições de peptídeo sinal e topologia transmembrana, respectivamente. Ambas foram obtidas através do *download* do pacote *stand-alone* para execução no servidor local, sendo que há restrições de uso exclusivamente acadêmico para utilização neste modo. Na ferramenta SignalP, os parâmetros *-fasta <caminho_arquivo_proteína>* e *-stdout* foram utilizados para, respectivamente: 1) definir o caminho do arquivo em formato FASTA contendo a sequência de aminoácidos da proteína a ser analisada e 2) escrever a saída no *prompt* do sistema, ao invés de realizar a geração de um arquivo. No arquivo de saída, o identificador *CS pos* foi utilizado para obter o ponto de clivagem do peptídeo sinal. O software TMHMM foi utilizado com um único parâmetro não nomeado: o caminho do arquivo em formato FASTA contendo a sequência de aminoácidos da proteína a ser analisada. No arquivo de saída, os identificadores *PredHel* e *Topology* indicam a presença e topologia das hélices transmembrana.

Para a análise de grupos de ortólogos, foram selecionados os bancos de dados KEGG Orthology, EggNOG e COG. A busca aos três bancos de dados foi realizada através da ferramenta eggNOG-mapper (v2) (HUERTA-CEPAS *et al.*, 2017), obtida através do *download* do pacote *stand-alone* para execução no servidor local. Os parâmetros *-i <caminho_arquivo_proteína>*, *–output <caminho_arquivos_saída>*, *-m diamond* e *–no_file_comments* foram utilizados para, respectivamente: 1) definir o arquivo em formato FASTA contendo a sequência de aminoácidos da proteína a ser analisada; 2) definir o caminho da pasta para geração dos arquivos de saída; 3) utilizar o *software* DIAMOND para realizar as tarefas de alinhamentos, no lugar do *software* BLAST e 4) remover trechos de comentários nos arquivos de saída. Dentre os arquivos de saída obtidos, utilizou-se o arquivo separado por tabulações **.emapper.annotations* para identificar os termos EggNOG, KO e COG.

Para complementar a anotação funcional, foram incorporadas as funcionalidades de predição de enzimas CAZy (específica para fungos), e análise de completude do proteoma através da ferramenta BUSCO (v4). Para a predição de enzimas CAZy, foi utilizada a ferramenta eggNOG-mapper (v2), obtida da mesma forma e executada com os mesmos parâmetros do contexto da análise de ortólogos, sendo que o resultado referente à esta predição também é encontrado no arquivo de saída **.emapper.annotations*. A ferramenta BUSCO foi obtida através do *download* do pacote *stand-alone* para execução no servidor local e os *datasets* selecionados foram: *Eukaryota*, *Fungi*, *Dikarya*, *Ascomycota* e *Eurotiomycetes*. Os parâmetros *-i <caminho_arquivo_proteína>*, *–config <caminho_arquivo_configuração>*, *-o <caminho_arquivos_saída>*, *-l <caminho_pasta_dataset>*, *-m proteins* e *–offline* foram utilizados para, respectivamente: 1) definir o arquivo em formato FASTA contendo a sequência de aminoácidos da proteína a ser analisada; 2) especificar o caminho do arquivo de configuração da ferramenta; 3) definir o caminho da pasta para geração dos arquivos de saída; 4) definir o caminho da pasta do *dataset*

específico a ser consultado; 5) definir o modo de consulta para sequência de aminoácidos, ao invés de sequência de nucleotídeos e 6) impedir o *download* automático dos arquivos do *dataset*, caso este não seja encontrado no sistema. No arquivo de resultado *full_table.tsv* é possível identificar os termos do banco de dados BUSCO que foram encontrados para a sequência informada e, baseado em todos os termos presentes no *dataset*, informar ao usuário o percentual de completude do proteoma de todo o projeto em relação à cada *dataset* específico.

3.2.3 Segurança

A primeira etapa realizada com relação à segurança (Figura 9, B5) foi a implementação de um sistema de *login*. O *framework* Django inclui nativamente um sistema de autenticação e autorização, com estruturas de sessões, usuários, grupos e permissões já implementadas (MELÉ, 2020). Ao implementar a página de login e utilizar o decorador *login_required()* em cada *view* do sistema, foi possível garantir que nenhum acesso não autenticado seja permitido no sistema. Usuários não autenticados, inativos ou com tempo de sessão expirado são automaticamente redirecionados para a página de *login*, não sendo possível navegar para nenhuma página do sistema sem realizar a autenticação novamente. As senhas dos usuários são criptografadas com o algoritmo *Password-Based Key Derivation Function 2* (PBKDF2) e função *hash* SHA256.

No que se refere à autorização, esta foi implementada manualmente, sem a utilização da estrutura de permissões nativamente oferecida pelo *framework* Django. Isto porque, no âmbito da solução da presente pesquisa, ao invés de um usuário ter uma lista de permissões em todo o contexto do sistema, aqui um mesmo usuário pode ter diferentes permissões em diferentes "Projetos"(abstração para os diferentes organismos estudados por um mesmo usuário dentro da plataforma). Em cada projeto, um usuário pode atuar como administrador (usuário que criou o projeto) ou como colaborador (usuário que foi convidado para participar de um projeto). Para cada colaborador, o administrador do projeto pode definir as permissões 1) *Manage files*: Usuário pode realizar o gerenciamento de arquivos externos; 2) *Schedule imports*: usuário pode agendar a importação de arquivos de genoma e de anotação estrutural; 3) *Schedule feature analysis*: usuário pode agendar a execução automática das ferramentas de anotação funcional; 4) *Manually edit genes and features*: usuário pode realizar curadoria manual e 5) *Schedule export*: usuário pode agendar a exportação do projeto para publicação no banco de dados GenBank. A tentativa de requisição de qualquer *view* do sistema em projetos no qual o usuário é membro, porém, não possui a permissão devida para executar a ação específica, tem como retorno o código de erro 403 (*Forbidden*).

Ataques muito comuns em aplicações *web* estão relacionados à *Cross Site Request Forgery* (CSRF) (CALZAVARA *et al.*, 2020). Este tipo de ataque ocorre quando o usuário, ao acessar uma página contendo um *script* malicioso, tem dados não desejados submetidos pelo seu navegador à outra aplicação *web* vulnerável, na qual este mesmo usuário está autenticado. Este ataque ocorre de forma silenciosa e imperceptível ao usuário, podendo ser indistinguível

pela aplicação *web* vulnerável em meio a outras requisições que tenham sido realizadas de forma intencional. Como forma de prevenção à estes ataques, foram aplicadas as funcionalidades de proteção de CSRF oferecidas pelo *framework* Django que, para os métodos considerados não seguros (POST, PUT, DELETE), verifica os cabeçalhos *Referer* e *Origin* para validação de mesma origem e exige que os dados da requisição contenham um *token* de validação (*CSRF Token*) que deve ser igual ao *token* presente nos *cookies* da requisição.

Proteções contra ataques de injeção de *Structured query language* (SQL) (HALFOND *et al.*, 2006) também são tratados automaticamente por um sistema específico de proteção do *framework* Django. Este tipo de ataque permite que usuários maliciosos executem comandos SQL arbitrários em um banco de dados, podendo resultar em perda de dados. Conforme recomendado pela documentação, objetos do tipo *QuerySet* são utilizados para realizar consultas, permitindo que os parâmetros sejam totalmente separados da *query* e que caracteres de *escape* sejam adicionados pelo *driver* do banco de dados para evitar o ataque. Além disso, nenhum comando SQL é executado de forma nativa no sistema como um todo.

3.2.4 Validação

A avaliação de qualidade do sistema foi realizada na forma de um questionário (Quadro 7), sendo observados os aspectos de qualidade de *software* definidos pela ISO/IEC 25010 (ISO25010, 2011). Foram definidos 21 itens para compor o questionário, com opções de resposta "Concordo"(C) e "Discordo"(D), sendo obrigatório o preenchimento de uma observação para os casos nos quais foi selecionada a opção "Discordo". Os itens que compõem o aspecto *Usabilidade* foram utilizados para validar o objetivo específico de proporcionar uma visualização compreensível, intuitiva e integrada dos resultados de anotação, enquanto os itens que compõem os aspectos *Funcionalidade*, *Confiabilidade*, *Eficiência*,*Compatibilidade* e *Segurança* foram utilizados para validar as funcionalidades e operações do sistema, correspondentes aos demais objetivos específicos. O questionário foi aplicado em 4 usuários em potencial, todos com formações relacionadas às áreas biológicas ou de ciências da computação.

A validação dos dados biológicos apresentados no sistema (Figura 9, V1) foi realizada por usuário especialista, no processo de anotação funcional das linhagens selvagem (2HH) e mutante (S1M29) do fungo *Penicillium echinulatum*. A anotação de cada uma das linhagens foi realizada em duas etapas: 1) através da utilização do sistema, agendando automaticamente as análises funcionais para todas as ferramentas englobadas e 2) através da e execução manual (via *download* ou utilização de *web services*) de cada uma das ferramentas de anotação funcional.

Quadro 7 – Questionário de avaliação de qualidade do sistema.

Questionário de avaliação de qualidade da ferramenta Seq2Annot					
Legenda: C (Concordo) | D (Discordo). Obrigatório informar observação para a opção "Discordo".					
	Profissão				
	Área de atuação				
Aspecto	**Sub-aspecto**	**Item**	**C**	**D**	**Observação**
Funcionalidade	Adequação	O sistema dispõe de todas as funcionalidades necessárias para sua execução.			
Funcionalidade	Acurácia	O sistema faz o que foi proposto de forma correta.			
Funcionalidade	Acurácia	O sistema é preciso na execução de suas funções e nos resultados apresentados.			
Confiabilidade	Tolerância à falhas	O sistema não apresenta falhas com frequência.			
Confiabilidade	Tolerância à falhas	O sistema tem respostas adequadas quando ocorrem falhas.			
Confiabilidade	Tolerância à falhas	O sistema informa ao usuário quando entradas de dados inválidas são inseridas.			
Confiabilidade	Recuperabilidade	O sistema é capaz de recuperar dados em caso de falhas.			
Confiabilidade	Conformidade	O sistema está de acordo com as normas, leis, etc.			
Usabilidade	Inteligibilidade	É possível entender facilmente o conceito e a aplicação do sistema.			
Usabilidade	Inteligibilidade	É fácil de entender as diferentes funcionalidades oferecidas pelo sistema.			
Usabilidade	Inteligibilidade	O sistema apresenta os dados dispostos de forma coerente e de fácil entendimento.			
Usabilidade	Atratividade	O sistema possui uma interface atrativa.			
Usabilidade	Facilidade de uso	O sistema é fácil de operar e controlar.			
Usabilidade	Facilidade de uso	O sistema facilita a entrada de dados pelo usuário.			
Usabilidade	Facilidade de uso	O sistema facilita a saída de dados pelo usuário.			
Usabilidade	Operacionalidade	O sistema fornece ajuda de forma clara.			
Eficiência	Tempo	O tempo de resposta do sistema é adequado.			
Compatibilidade	Interoperabilidade	Os diferentes módulos/seções do sistema interagem de forma correta.			
Compatibilidade	Interoperabilidade	O sistema tem capacidade de processamento multiusuário.			
Segurança	Confidencialidade	O sistema dispõe de segurança através de login, senhas e permissões de usuário.			
Segurança	Integridade	O sistema não apresenta inconsistência ou perda de dados.			

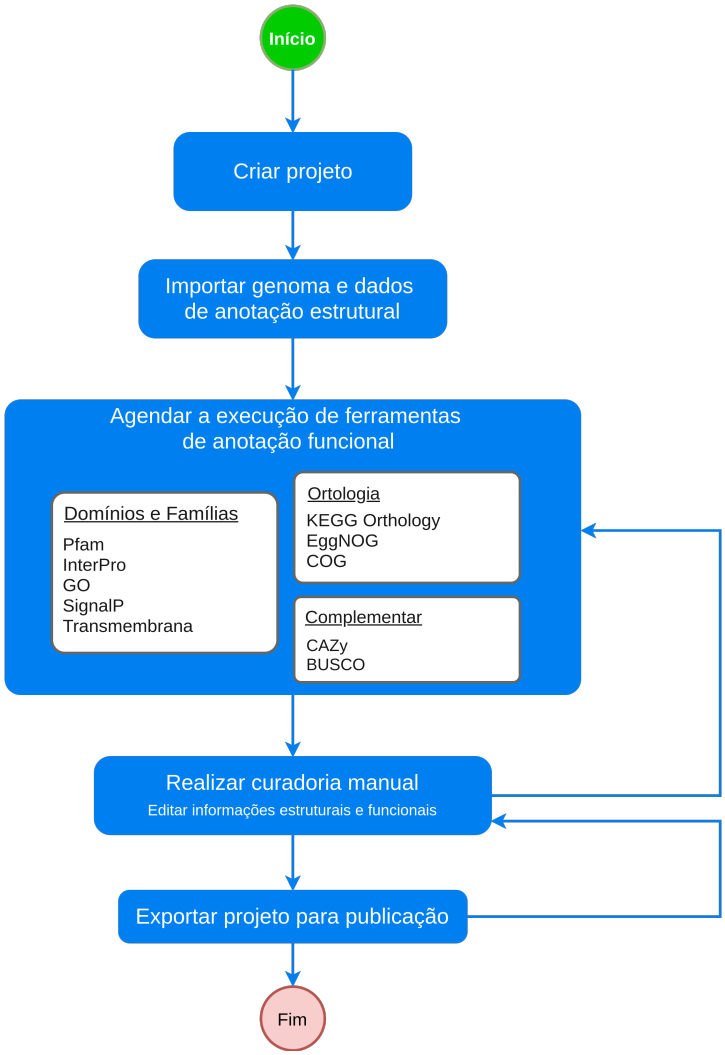
Fonte: Adaptada de Jensen *et al.* (2012).

4 RESULTADOS

4.1 *WORKFLOW* CIENTÍFICO DE ANOTAÇÃO GENÔMICA FUNCIONAL E CURADORIA MANUAL DE GENOMAS

Os resultados do *workflow* proposto estão apresentados na forma de um artigo científico, intitulado *Seq2Annot: a scientific workflow for functional annotation and manual curation of genomes*. O artigo foi submetido para publicação na revista *Bioinformatics*[1], da *Oxford Academic*. As etapas do *workflow* estão apresentadas na Figura 11.

Figura 11 – *Workflow* científico de anotação genômica funcional e curadoria manual de genomas.



Fonte: Elaborada pelo autor.

[1] https://academic.oup.com/bioinformatics

Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Original Paper

OXFORD

Genome analysis

Seq2Annot: a scientific workflow for functional annotation and manual curation of genomes

Eduardo Balbinot [1,*], Alexandre Rafael Lenz[1,2], Fernanda Pessi de Abreu [1], Pedro Lenz Casa [1] , Scheila de Avila e Silva [1] and Aldo José Pinheiro Dillon [3]

[1] Computational Biology and Bioinformatics Laboratory, Universidade de Caxias do Sul, Caxias do Sul, Brazil.
[2] Departamento de Ciências Exatas e da Terra, Universidade do Estado da Bahia, Salvador, Brazil.
[3] Biotechnology Institute, Universidade de Caxias do Sul, Caxias do Sul, Brazil.

*To whom correspondence should be addressed.

Associate Editor: XXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Functional annotation is a crucial step in genome annotation, aiming to understand organisms' biological processes and guide new research. However, it is a complex task, as it involves the use of a considerable number of databases, web servers and computational tools in order to infer possible biological functions. Each of these tools has independent and format-specific output files, making it difficult to keep annotation data organized and hampering the procedure of manual curation. Some of them also require local installation of libraries and execution in command-line interfaces, which compromises working in a team and eventually requires deeper computing skills from researchers. Due to this scenario, the integration of functional annotation results and the ability to perform manual curation in a unified platform are implementations of interest in the field of bioinformatics.
Results: Here we present Seq2Annot, a scientific workflow for functional annotation and manual curation of genomes. It consists of a collaborative web application that automatically executes common tools used in functional annotation pipelines and integrates annotation data into a single-context platform. The solution also supports manual curation of structural and functional data for each gene, and allows users to export annotation files into formats required by GenBank's submission process.
Availability: An experimental version of Seq2Annot is freely available at https://seq2annot.org.
Contact: ebalbinot@ucs.br, arlenz@ucs.br, fpabreu1@ucs.br, plcasa@ucs.br, sasilva6@ucs.br and ajpdillo@ucs.br.

1 Introduction

The efficiency of Next-Generation Sequencing technologies has revolutionized the field of genomics. As massive parallel processing approaches made sequencing more affordable and less time-consuming, researchers began addressing larger genomes (Giani *et al.*, 2019; Besser *et al.*, 2018). In this manner, the exponential increase in the amount of biological data saw a rising demand for storing, organizing and analysing both genomic and transcriptomic data, elevating the role and the importance of bioinformatics (Shaik *et al.*, 2019).

At this point, sequencing no longer poses a scientific barrier. In fact, analysing the structure and giving biological meaning to genomic and proteomic sequences *in silico* (genome annotation) became the new challenging task (Wang *et al.*, 2017). The process consists of three major steps: (i) identifying and marking repetitive regions, like tandem repeats and transposable elements; (ii) predicting the existence and location of structural elements, such as protein-coding genes, tRNAs, rRNAs and siRNAs genes (structural annotation); and (iii) inferring and determining the function for each gene product (functional annotation) (Humann *et al.*, 2019).

Functional annotation is an essential procedure of genome annotation. The large amount of raw biological data in public repositories demand functional characterization to understand organisms' biological processes

and guide new research (McDonnell *et al.*, 2018). However, it is a complex task, as it requires comparing all the sequences of interest with other sequences available in protein domain repositories, with the purpose of finding similarities and conserved motifs that will help researchers infer the function of each gene product (Cruz *et al.*, 2019). The large set of databases, web servers, and computational tools that must be individually accessed and executed for all genes within a project makes functional annotation a laborious task. A typical pipeline encompasses the gathering of as much information as possible before properly naming genes, such as: (i) searching for protein elements, such as Gene Ontology (GO) terms, domains, families, presence of signal peptide and transmembrane protein topology; (ii) finding orthologous groups; and (iii) finding similarities with already characterized and reviewed proteins (Del Angel *et al.*, 2018).

Manually running the tools involved in the process can also lead to difficulties in organizing and grouping annotation data without a single-context environment. Besides that, the output files are generated independently and in different formats, once again influencing result analysis, given that resources are not integrated (Araujo *et al.*, 2018; Humann *et al.*, 2019). Most web servers also have limitations for batch jobs, requiring the local installation of multiple libraries and execution in command-line interfaces. These conditions impose restrictions on the possibility of teamwork and encompass deeper computational knowledge from researchers. A single environment to support functional annotation is even more necessary for the purpose of manual curation, a careful analysis in which researchers look for suspicious outputs that might suggest incorrectly predicted genes (McDonnell *et al.*, 2018). In this case, the researcher might have to manually edit structural and functional data, an additional meticulous and time-consuming job that requires editing individual annotation files. Finally, getting the project ready for submission in public databases, such as GenBank (Benson *et al.*, 2018), requires an extra set of command-line tools to convert annotation data into file formats required by each platform (Geib *et al.*, 2018).

In this paper, we present Seq2Annot, a workflow designed to abstract the complexity of functional annotation and manual curation of genomes. It consists of a collaborative web application that automatically executes common tools and resources used in functional annotation pipelines and integrates annotation data into a single-context platform. The tool also supports manual curation by providing features to manually edit both structural and functional data for each gene. Besides this, the solution automatically converts annotation data into file formats required by GenBank submission. The workflow was tested and validated in the annotation process of wild-type (2HH) and mutant (S1M29) strains of *Penicillium echinulatum* (Lenz *et al.*, 2020), conducted by XXX Laboratory at University XXX (XXX). Annotated genomes of both strains were deposited in GenBank under *BioProject* accession numbers XXX and XXX, respectively.

2 Materials and methods

2.1 Implementation

The system *back-end* was developed in Python (3.6) (Python, 2020). The web framework Django (2.1.7) (Django, 2020) was used to provide more productivity, security, and scalability to the project. We also used the BioPython (1.74) (Cock *et al.*, 2009) library to handle file formats commonly used in bioinformatics (like FASTA, GFF3 and GenBank) and

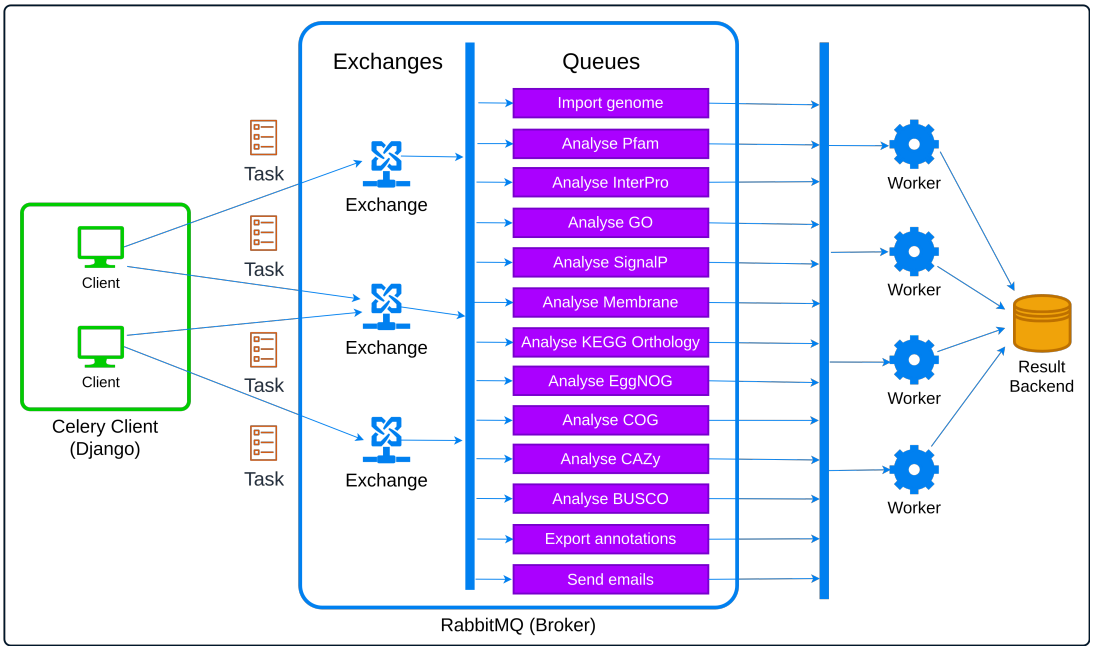


Fig. 1: Seq2Annot's distributed system architecture to handle jobs asynchronously, based on Celery framework. As users schedule the execution of functional annotation tools and other long-running jobs, Django requests are mapped to *tasks* that are then filtered and routed to specific *queues* through *exchanges*. *Workers* handle task execution by constantly monitoring queues. The *message broker* RabbitMQ handles communication between clients and *workers*.

to translate DNA sequences into protein sequences, based on start and stop exon positions. The relational database PostgreSQL (9.4) (PostgreSQL, 2020) was adopted, and both database interactions and migration control are performed by Django's object-relational mapping (ORM) system.

The core mechanism of asynchronous task execution (Figure 1) represents a key role in this solution. Handling multiple executions of functional annotation tools generates heavy server workloads, as many of them require complex alignments when searching for sequence similarities. Furthermore, several users may simultaneously request functional analyses for large sets of genes within different projects. This scenario would result in a multitude of jobs that cannot be executed at the same time in order to avoid server overload. The option of remotely executing tools when possible would also lead to the same problem, as we might experience remote server downtimes or overwhelmingly long-running tasks due to intensive workloads. The distributed system Celery (4.3) (Celery, 2020) was used to address this issue by organizing, controlling, and limiting the concurrent execution of multiple jobs. In this case, user experience is also enhanced, as users do not have to wait on the page while time-consuming tasks are either running or waiting for execution.

Every request a user performs for a long-running job, like the execution of functional annotation tools or data importing and exporting, is translated in the *back-end* into a single unit of work (task), which is then routed to a queue, depending on its type. Specific queues were created for all types of tasks that are executed asynchronously, so that server resources can be allocated differently to each of them. The execution of each task is handled by services called *workers*, which constantly monitor queues, searching for pending tasks yet to be processed. The *message broker* RabbitMQ (3.7) (RabbitMQ, 2020) was used to handle the communication between clients (user requests) and *workers*, also being responsible for filtering incoming tasks as well as distributing them to their specific queues through message routing agents called *exchanges*. The library *django-celery-results* (1.0.4) (DjangoCeleryResults, 2020) was used as the *result backend* to store task results and metadata in database tables integrated with Django's ORM system. The ability to distribute work across different *threads* or servers through Celery and RabbitMQ provides great horizontal scaling capabilities to the solution. As the list of pending tasks in specific queues starts to grow, system administrators can effortlessly allocate more *workers* to handle a more significant number of concurrent jobs.

An extra mechanism to avoid a single user or group of users in a team to schedule too many jobs at the same time was also implemented. Considering that the system keeps track of how many tasks are in the queues and which user requested each of them, we were able to limit the number of functional analyses that can be concurrently scheduled for execution within a project. This approach allowed us to give users from different teams the same priority when scheduling functional analyses. Besides this, other limitations that might reflect on server performance include: (i) the maximum number of projects per user, (ii) the maximum number of additional collaborators per project, (iii) the maximum number of genes per project and (iv) the maximum number of features per gene. These are dynamic parameters that can be set on each user by system administrators on Django's admin interface, through a model called *UserPlan*.

The *front-end* was developed in the languages *HyperText Markup Language* (HTML5), *Cascading Style Sheets* (CSS3), and JavaScript, supported by Django's template system specially designed for dynamic page generation. We used the jQuery (3.3.1) (jQuery, 2020) library for more agility when manipulating page elements and performing *Asynchronous Javascript and XML* (AJAX) requests. The library Bootstrap (4.3.1) (Bootstrap, 2020) was also used to add responsive capabilities and powerful pre-built components, like grids, modal windows, dropdown menus, input elements and many others.

2.2 Functional annotation tools

The functional annotation tools currently supported by Seq2Annot mainly cover protein domain and orthology prediction. The role of each tool in the annotation workflow and databases queried by them are also detailed in Table 1.

Table 1. Functional annotation tools supported by Seq2Annot.

Tool	**Databases**	**Role in workflow**
HMMER (3.3.1)	- Pfam	Protein domain prediction
InterProScan (5.46)	- InterPro - GO	Protein domain prediction
SignalP (5.0)	- SignalP	Signal peptide prediction
TMHMM (2.0)	- TMHMM	Transmembrane protein topology prediction
eggNOG-mapper (v2)	- KEGG Orthology - EggNOG - COG	Orthologous groups prediction
eggNOG-mapper (v2)	- CAZy	CAZy family prediction
BUSCO (v4)	- OrthoDB	Proteome completeness analysis

For analysing protein domains and families, databases Pfam (El-Gebali *et al.*, 2019), InterPro (Mitchell *et al.*, 2018), and GO (Acencio *et al.*, 2019) were selected. Pfam analysis is performed by running the stand-alone version of HMMER (3.3.1) (Finn *et al.*, 2011). We used *hmmscan* tool with parameters *–cut_ga*, *–acc*, *–tbl_out* <*output_file*>, <*db_file*>, and <*protein_file*> for (i) automatically defining *cut-off* limits for each family, avoiding false positives; (ii) preferring accessions over names in output; (iii) specifying the output file; (iv) defining Pfam database file; and (v) specifying the protein file in FASTA format. InterPro and GO analysis is performed by running the stand-alone version of InterProScan (5.46) (Jones *et al.*, 2014). The parameters *-i* <*protein_file*>, *-b* <*output_file*>, and *-f json* were used for (i) specifying the protein file in FASTA format; (ii) specifying the output file; and (iii) generating the output file in *JavaScript Object Notation* (JSON) format.

The stand-alone versions of SignalP (5.0) (Armenteros *et al.*, 2019) and TMHMM (2.0) (Krogh *et al.*, 2001) were used for performing signal peptide and transmembrane protein topology analysis, respectively. SignalP was used with parameters *-fasta* <*protein_file*> and *-stdout* for (i) specifying the protein file in FASTA format and (ii) writing the output to the system's prompt instead of writing to a file. Moreover, the identifier *CS pos* in the output file is used to extract the signal peptide's cleavage site position. TMHMM was used with a single unnamed parameter, which specifies a FASTA file containing the protein code to be analysed. Identifiers *PredHel* and *Topology* in the output file indicate the presence of transmembrane helices.

For predicting orthologous groups, we chose KEGG Orthology (Kanehisa *et al.*, 2016), EggNOG (Huerta-Cepas *et al.*, 2019), and Clusters of Orthologous Groups (COG) (Galperin *et al.*, 2019) databases by using the stand-alone version of eggNOG-mapper (v2) (Huerta-Cepas *et al.*, 2017) to perform queries. The parameters *-i* <*protein_file*>, *–output* <*output_folder*>, *-m diamond*, and *–no_file_comments* were used for (i) specifying the protein file in FASTA format; (ii) defining the folder where output files are generated; (iii) performing alignments using DIAMOND (Buchfink *et al.*, 2015) instead of BLAST (Altschul *et al.*, 1990); and (iv) removing comments from output files. KO term, EggNOG term, and COG classifications are extracted from the output file **.emapper.annotations*.

Complementary annotation tools to predict Carbohydrate-active enzymes (CAZy) families and analyse proteome completeness were

also added to the solution. CAZy prediction is performed by eggNOG-mapper (v2) with the same parameters and result file used in orthology analysis. Proteome completeness is analysed with the stand-alone version of BUSCO (v4) (Seppey *et al.*, 2019), whose datasets contain near-universal single-copy orthologs selected from OrthoDB (Kriventseva *et al.*, 2019). The parameters *-i <protein_file>*, *–config <config_file>*, *-o <output_folder>*, *-l <dataset_folder>*, *-m proteins*, and *–offline* were used for, respectively: (i) specifying the protein file in FASTA format; (ii) defining the config file; (iii) specifying the output folder; (iv) specifying the dataset folder; (v) running BUSCO in protein mode; and (vi) preventing automatic downloads if dataset files do not exist. BUSCO entry hits for the dataset analysed are extracted from the output file *full_table.tsv*.

2.3 Security

A login engine was implemented using Django's default authentication mechanism (Melé, 2020). The decorator *login_required()* used in all views of the system ensures unauthenticated or invalid requests (originated from users that have been deactivated or even from expired sessions) are denied and properly redirected to the login page, forcing users to log in.

Authorization was handled manually, not by using Django's default permissions structure, as we needed a deeper granularity of permissions at a project level. In addition to having a set of permissions for the entire application, users might have different permissions on each project. Firstly, a single user can either be the owner of a project or act as a collaborator. For each collaborator in a specific team project, the owner can set the following permissions: (i) *Manage files*: the user can manage external file uploads; (ii) *Schedule imports*: the user can schedule imports of previously uploaded genome and structural annotation files; (iii) *Schedule feature analysis*: the user can schedule automatic execution of functional annotation tools for a set of gene products; (iv) *Manually edit genes and features*: the user can perform manual curation, with the possibility of editing structural (imported data) and functional attributes; (v) *Schedule export*: the user can export project annotations for submission in GenBank. Any attempt to perform an action out of the users' permission scope in a specific project is denied with a 403 (Forbidden) error code.

Cross-site request forgery (CSRF) (Calzavara *et al.*, 2020) protection is also applied by using Django's CSRF middleware. CSRF attacks are a typical type of attack in which malicious users can perform actions using another user's credentials without that user's consent. To prevent this type of attack, headers *Origin* and *Referer* of requests made via unsafe methods (POST, PUT, and DELETE) must pass a verification process, and the request's data should contain a valid token (CSRF Token), which must match the same request's token cookie.

Structured Query Language (SQL) injection attacks (Halfond *et al.*, 2006) are also automatically handled by Django's SQL injection protection. This type of attack allows malicious users to execute arbitrary SQL code on a database, leading to data leakage. As recommended in the documentation, we use Django's *QuerySet* to perform queries, which separates the query's SQL code from its parameters. This approach allows parameters to be escaped by the database driver and consequently avoids SQL injection. No raw queries or custom SQLs have been implemented on this project, so full protection against this type of attack is guaranteed.

3 Results

3.1 Annotation workflow

In order to start using the application, first time users need to fill a sign-up form. The email address is an essential part of this process, as it is used in later steps to send notifications and invite new collaborators to join a team project. After signing up, new users receive an account confirmation email, and must click on the confirmation link to unlock their accounts and have access granted into the system.

Every new user is automatically assigned to a *User plan*, previously created by system administrators in Django's admin interface. User plans contain parameters that limit the actions a user can perform, like the maximum number of projects created or the maximum number of concurrent functional analyses scheduled (list of parameters discussed in Section 2.1). The same plan with a set of configurations can be applied to many users, and its parameters will be extended to the projects they own. A system preference holds the identifier of the default plan to be automatically set for newly created users.

When a user is logged into the system, the annotation workflow (Figure 2) starts, and the first step is to create a project. In addition to the organism's name and strain information, users must inform prefix patterns to be used in *contigs* and gene identifiers, as well as a suffix pattern for gene products. These patterns will be continuously validated when importing genes from external files or performing manual curation, avoiding typing errors and preventing invalid files from being imported.

After creating a project, the user is redirected to the *Project Dashboard*, a page divided into five tabs. The *General Info* tab allows managing the project's general information and team members. The *Files and Imports* tab is used to upload genome and structural annotation files and schedule an import process. The *Genes* tab allows users to manage the full list of genes within the project and schedule the execution of functional annotation tools or perform manual curation. The *Statistics* tab groups annotation data into simple reports so that users may analyse proteome completeness, GO term distribution, as well as totalizers of mRNAs, tRNAs, rRNAs, proteins containing signal peptide, transmembrane proteins, and functional terms returned from the execution of annotation tools. The *Export* tab is used to convert the project's annotation data into file formats required by GenBank submission.

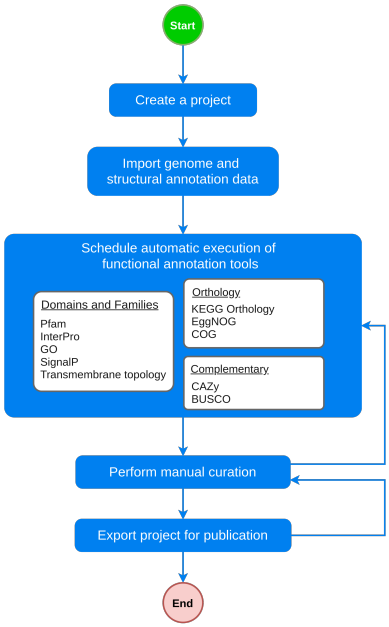


Fig. 2: Seq2Annot's workflow for functional annotation and manual curation of genomes.

The next step of the workflow is to import genome and structural annotation files. As our solution is targeted towards users that have already completed the process of gene prediction, two essential files are required: (i) genome file, in FASTA format, resulted from genome assembly process; and (ii) structural annotation file, in GFF3 format, resulted from gene prediction process. When requesting an import process from the *Files and Imports* tab, users must select both files from the uploaded files repository and wait for the execution to complete. The job status can be monitored in the *Imports* panel, and the user will receive an email notification when the process is finished. A list of import warnings showing eventual validation warnings or errors that occurred during the import process is also available.

When importing is complete, users can navigate through the full list of genes at the *Genes* tab. The list shows detailed structural information about genes and gene products (called gene *features* in this context), supporting multiple gene products per gene in case of alternative splicing. In this tab, users perform the most important step of the annotation workflow: scheduling automatic execution of functional annotation tools for a set of gene features. Using filters, researchers may select groups of features and add them to the *feature cart*, which consists of a temporary list of features waiting to be analysed. When all the desired features have been added to the cart, users then select the list of functional annotation tools to execute and schedule the analysis process. The status of each feature is changed to *in progress* until all asynchronous jobs are completed. As the execution of each tool is finished, results are displayed in the *Attributes* column. Information on the status of each task and the complete list of structural and functional attributes can be found on the *feature detail* page. In order to provide an integrative analysis and aid in manual curation of the attributes found, every functional term displayed carries a link to the related database web page containing detailed information.

Analysing particular groups of genes is an important part of this step. A vast number of structural and functional attributes can be filtered together so users can analyse specific parts of the genome, especially when performing manual curation. Researchers may want, for example, to schedule specific analysis only for CAZy proteins of a particular family, or perhaps only for genes inside a specific *contig*. For this reason, a powerful filtering engine was developed based on the *interpreter* design pattern (Nesteruk, 2019). In Seq2Annot, filtering is performed by entering a single-line expression in a syntax specific format, allowing users to create complex queries that support aggregations (through parentheses) and operators of types AND, OR, and NOT (expressed as *&*, *|* and *!*, respectively). In this case, the filter expression "*(cazy=GH12&contig=PECH_scaffold_1) | (!(pfam=PF01670.17)&start>=1000&stop<=20000&contig=PECH_scaffold_2)*" would return all gene products which match at least of one of the following conditions: (i) gene product is located in *contig* named *PECH_scaffold_1*, and it contains CAZy family *GH12*; or (ii) gene product is located between positions 1000 and 2000 of *contig* named *PECH_scaffold_2*, and it does not contain Pfam domain *PF01670.17*.

The next step in the annotation workflow is manual curation. The solution supports it by allowing users to manually edit all structural and functional data of genes and gene products on the *feature detail* page. For instance, if a gene was predicted erroneously, researchers can edit exons' start and stop positions and reschedule a new set of functional analysis for it. It is also possible to remove inexistent or problematic genes and even create new genes from scratch. In addition, importing functional attributes eventually obtained from external tools is also supported by uploading a tab-separated values (TSV) file at the *Files and Imports* tab. The latter feature is particularly useful for importing *Enzyme Commission numbers* (EC numbers), whose prediction is not yet supported by our tool.

The final workflow step takes place at the *Export* tab, which is used to export and convert annotation data into file formats required for submission to GenBank. In the *back-end*, the process mainly consists of generating a *feature table* (TBL) file, running *National Center for Biotechnology* (NCBI)'s *tbl2asn* tool (Benson *et al.*, 2018), and grouping all final annotation files into a single compressed file. Before scheduling an export process, users must upload a *submission template* (SBT) file, containing information about the publication and the list of researchers involved in the project. When exporting is complete, a compressed file is available containing all genome and annotation files: (i) *discrepancy.txt*, *errorsummary.val*, and *organism.val*: warning files resulted from *tbl2asn* tool; (ii) *organism.ecn*: EC number errors file identified by *tbl2asn*; (iii) *organism.sqn*: the main file requested in GenBank's submission process; (iv) *organism.gbf*: file containing complete genome structural and functional data, in GenBank format; (v) *organism.fsa*: a copy of the imported genome file in FASTA format; (vi) *organism.tbl*: structural and functional annotation file in TBL format; and (vii) *organism.sbt*: a copy of the imported *template* file. If any errors are found, researchers can return to manual curation and functional annotation steps and repeat the process as many times as necessary until the annotation is complete. The *Export* tab also supports exporting annotation results to a TSV file.

3.2 Test and validation: *Penicillium echinulatum* 2HH and S1M29 annotation

The workflow was tested and validated in the functional annotation process of *Penicillium echinulatum* wild-type (2HH) and mutant (S1M29) strains (Lenz *et al.*, 2020), conducted by the XXX Laboratory at the University of XXX (XXX). A *beta* version of Seq2Annot was used, which incorporated the last stable version of each functional annotation tool available on August 1st, 2018. Both 2HH and S1M29 strains were deposited in GenBank under *BioProjects* XXX and XXX, respectively. The annotation process resulted in 8366 genes for 2HH and 8375 genes for S1M29. An example of a transcription factor annotated for the 2HH strain is *PECH_001380*. Table 2 details all functional terms automatically annotated for this gene.

Table 2. Functional terms automatically annotated by Seq2Annot for gene PECH_001380 in *Penicillium echinulatum* 2HH.

Database	**Term**	**Description**
Pfam	PF04082.18	Fungal specific transcription factor domain
Pfam	PF00172.18	Fungal Zn(2)-Cys(6) binuclear cluster domain
InterPro	IPR001138	Zn(2)-C6 fungal-type DNA-binding domain
InterPro	IPR007219	Transcription factor domain, fungi
InterPro	IPR036864	Zn(2)-C6 fungal-type DNA-binding domain superfamily
GO	GO:0000981	DNA-binding transcription factor activity, RNA polymerase II-specific
GO	GO:0003677	DNA binding
GO	GO:0005634	nucleus
GO	GO:0006351	transcription, DNA-templated
GO	GO:0006355	regulation of transcription, DNA-templated
GO	GO:0008270	zinc ion binding
EggNOG	S8ASG9	– Fungal specific transcription factor domain —- ENOG502AP5Z (root level) —- ENOG502RZG2 (Eukaryota level) —- ENOG503A2VJ (Opisthokonta level) – C6 transcription factor (AmyR) — ENOG503P3CM (Fungi level) — ENOG503QTG3 (Ascomycota level) — ENOG5020EMN (Eurotiomycetes level) — ENOG503S3A6 (Eurotiales level)
COG	K	Transcription

The analysis of these functional terms and a subsequent manual curation process helped researchers identify it as an ortholog of AmyR, an important regulator of amylase encoding genes, in *P. oxalicum* 114-2 (*PDE_03964*) (Li *et al.*, 2015). Its gene product was annotated as *Amylolytic gene expression activator*.

3.3 Availability

An experimental version of Seq2Annot is freely available at https://seq2annot.org. Due to financial reasons, strict limitations regarding the number of concurrent jobs have been set to fit server infrastructure. Nevertheless, we encourage researchers to sign up for a trial by creating a project and running a small set of functional analyses to get familiar with the software. We will be pleased to receive feedback to constantly improve Seq2Annot and expand it into a bigger project.

Currently, new users are automatically assigned to a default plan with the following maximum values: (i) 1 project per user; (ii) 2 additional collaborators per project; (iii) 100 genes per project; (iv) 2 gene products per gene; and (v) 20 concurrent scheduled functional analysis tools. Internally, the number of queues and worker nodes was reconfigured to ensure only a small set of asynchronous jobs are executed concurrently.

4 Discussion

Functional annotation is a crucial step in genome annotation. The process involves using many tools and databases to understand the function of gene products and precisely name each gene. Manually running these tools is a complex and laborious process, making it difficult to organize and analyse annotation results and to work collaboratively in a team. Seq2Annot was developed to abstract this complexity by creating an organized workflow supported by a collaborative, single-context web platform. Besides automatically executing functional annotation tools, the solution also supports manual curation and converts annotation data into file formats required by GenBank submission.

In comparison to our solution, we selected and analysed four existing tools (Table 3) developed for similar purposes: GO FEAT (Gene Ontology Functional Enrichment Annotation Tool) (Araujo *et al.*, 2018), OmicsBox (a commercial tool developed by BioBam) (OmixBox, 2020), GenSAS (Genome Sequence Annotation Server) (Humann *et al.*, 2019) and RAST (Rapid Annotations using Subsystems Technology) (Aziz *et al.*, 2008).

Table 3. Evaluation of existing tools developed for similar purposes.

Criteria	GOFEAT	OmicsBox	GenSAS	RAST
Freely available	Yes	No	Yes	Yes
Online access	Yes	No	Yes	Yes
Automatic execution of functional annotation tools	Yes	Yes	Yes	Yes
Multiple projects	Yes	Yes	Yes	Yes
Teamwork	Yes	No	Yes	Yes
Manual curation: structural data	No	No	No	No
Manual curation: functional data	No	No	Yes	No
Exporting annotation data	Yes	Yes	Yes	Yes
Exporting annotation data to GenBank	No	Yes	No	No

To evaluate them, we defined nine functionalities considered by our team to be essential in a functional annotation workflow, as follows: (i) the tool is freely available; (ii) the tool can be accessed online; (iii) the tool supports the automatic execution of functional annotation tools; (iv) the tool supports working on multiple projects; (v) the tool supports adding additional collaborators to projects and working simultaneously; (vi) the tool supports manual curation by editing structural data; (vii) the tool supports manual curation by editing functional data; (viii) the tool allows exporting annotation data; and (ix) the tool allows exporting annotation data in file formats required for submission to GenBank. Considering that none of the tools analysed attended at least two of the described functionalities, we hold Seq2Annot as a long-needed and meaningful development in bioinformatics.

The core mechanism of asynchronous task execution provides great scalability support to the solution, making it possible to horizontally increase server resources as demand grows over time. The architectural decision of splitting jobs into single independent tasks also ensures that new functional annotation tools can be easily developed and plugged into Seq2Annot, making it a promising tool to be continuously improved and used in large-scale genome annotation projects in the future. In addition, future improvements could include automatic execution of genome assemblers and gene predictors to support previous genome annotation steps, making Seq2Annot a complete online and collaborative genome annotation pipeline.

Acknowledgements

The authors would like to thank XXX, XXX and XXX.

Funding

This research was supported by the XXX, XXX and XXX.
Conflict of Interest: none declared.

References

Acencio, M. L., Lægreid, A., Kuiper, M., *et al.* (2019). The gene ontology resource: 20 years and still going strong.

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic local alignment search tool. *Journal of molecular biology*, **215**(3), 403–410.

Araujo, F. A., Barh, D., Silva, A., Guimarães, L., and Ramos, R. T. J. (2018). GO FEAT: A rapid web-based functional annotation tool for genomic and transcriptomic data. *Scientific Reports*, **8**(1).

Armenteros, J. J. A., Tsirigos, K. D., Sønderby, C. K., Petersen, T. N., Winther, O., Brunak, S., von Heijne, G., and Nielsen, H. (2019). Signalp 5.0 improves signal peptide predictions using deep neural networks. *Nature biotechnology*, **37**(4), 420–423.

Aziz, R. K., Bartels, D., Best, A. A., DeJongh, M., Disz, T., Edwards, R. A., Formsma, K., Gerdes, S., Glass, E. M., Kubal, M., *et al.* (2008). The rast server: rapid annotations using subsystems technology. *BMC genomics*, **9**(1), 75.

Benson, D. A., Cavanaugh, M., Clark, K., Karsch-Mizrachi, I., Ostell, J., Pruitt, K. D., and Sayers, E. W. (2018). Genbank. *Nucleic acids research*, **46**(D1), D41–D47.

Besser, J., Carleton, H. A., Gerner-Smidt, P., Lindsey, R. L., and Trees, E. (2018). Next-generation sequencing technologies and their application to the study and control of bacterial infections. *Clinical microbiology and infection*, **24**(4), 335–341.

Bootstrap, B. (2020). Bootstrap: The most popular html, css, and js library in the world.

Buchfink, B., Xie, C., and Huson, D. H. (2015). Fast and sensitive protein alignment using diamond. *Nature methods*, **12**(1), 59.

Calzavara, S., Conti, M., Focardi, R., Rabitti, A., and Tolomei, G. (2020). Machine learning for web vulnerability detection: The case of cross-site request forgery. *IEEE Security & Privacy*.

Celery, C. (2020). Celery: Distributed task queue.

Cock, P. J., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B., *et al.* (2009). Biopython: freely available python tools for computational molecular biology and bioinformatics. *Bioinformatics*, **25**(11), 1422–1423.

Cruz, F., Lagoa, D., Mendes, J., Rocha, I., Ferreira, E. C., Rocha, M., and Dias, O. (2019). SamPler – a novel method for selecting parameters for gene functional annotation routines. *BMC Bioinformatics*, **20**(1), 454.

Del Angel, V. D., Hjerde, E., Sterck, L., Capella-Gutierrez, S., Notredame, C., Pettersson, O. V., Amselem, J., Bouri, L., Bocs, S., Klopp, C., *et al.* (2018). Ten steps to get started in genome assembly and annotation. *F1000Research*, **7**.

Django, D. (2020). The web framework for perfectionists with deadlines.

DjangoCeleryResults, D. (2020). Celery result back end with django.

El-Gebali, S., Mistry, J., Bateman, A., Eddy, S. R., Luciani, A., Potter, S. C., Qureshi, M., Richardson, L. J., Salazar, G. A., Smart, A., *et al.* (2019). The pfam protein families database in 2019. *Nucleic acids research*, **47**(D1), D427–D432.

Finn, R. D., Clements, J., and Eddy, S. R. (2011). Hmmer web server: interactive sequence similarity searching. *Nucleic acids research*, **39**(suppl_2), W29–W37.

Galperin, M. Y., Kristensen, D. M., Makarova, K. S., Wolf, Y. I., and Koonin, E. V. (2019). Microbial genome analysis: the cog approach. *Briefings in bioinformatics*, **20**(4), 1063–1070.

Geib, S. M., Hall, B., Derego, T., Bremer, F. T., Cannoles, K., and Sim, S. B. (2018). Genome annotation generator: a simple tool for generating and correcting wgs annotation tables for ncbi submission. *GigaScience*, **7**(4), giy018.

Giani, A. M., Gallo, G. R., Gianfranceschi, L., and Formenti, G. (2019). Long walk to genomics: History and current approaches to genome sequencing and assembly. *Computational and Structural Biotechnology Journal*.

Halfond, W. G., Viegas, J., Orso, A., *et al.* (2006). A classification of sql-injection attacks and countermeasures. In *Proceedings of the IEEE international symposium on secure software engineering*, volume 1, pages 13–15. IEEE.

Huerta-Cepas, J., Forslund, K., Coelho, L. P., Szklarczyk, D., Jensen, L. J., Von Mering, C., and Bork, P. (2017). Fast genome-wide functional annotation through orthology assignment by eggnog-mapper. *Molecular biology and evolution*, **34**(8), 2115–2122.

Huerta-Cepas, J., Szklarczyk, D., Heller, D., Hernández-Plaza, A., Forslund, S. K., Cook, H., Mende, D. R., Letunic, I., Rattei, T., Jensen, L. J., *et al.* (2019). eggnog 5.0: a hierarchical, functionally and phylogenetically annotated orthology resource based on 5090 organisms and 2502 viruses. *Nucleic acids research*, **47**(D1), D309–D314.

Humann, J. L., Lee, T., Ficklin, S., and Main, D. (2019). Structural and functional annotation of eukaryotic genomes with gensas. In *Gene Prediction*, pages 29–51. Springer.

Jones, P., Binns, D., Chang, H.-Y., Fraser, M., Li, W., McAnulla, C., McWilliam, H., Maslen, J., Mitchell, A., Nuka, G., *et al.* (2014). Interproscan 5: genome-scale protein function classification. *Bioinformatics*, **30**(9), 1236–1240.

jQuery, j. (2020). jquery: Write less, do more.

Kanehisa, M., Sato, Y., Kawashima, M., Furumichi, M., and Tanabe, M. (2016). Kegg as a reference resource for gene and protein annotation. *Nucleic acids research*, **44**(D1), D457–D462.

Kriventseva, E. V., Kuznetsov, D., Tegenfeldt, F., Manni, M., Dias, R., Simão, F. A., and Zdobnov, E. M. (2019). Orthodb v10: sampling the diversity of animal, plant, fungal, protist, bacterial and viral genomes for evolutionary and functional annotations of orthologs. *Nucleic acids research*, **47**(D1), D807–D811.

Krogh, A., Larsson, B., Von Heijne, G., and Sonnhammer, E. L. (2001). Predicting transmembrane protein topology with a hidden markov model: application to complete genomes. *Journal of molecular biology*, **305**(3), 567–580.

Lenz, A. R., Galan-Vasquez, E., Balbinot, E., Pessi de Abreu, F., Souza de Oliveira, N., Osorio da Rosa, L., de Avila e Silva, S., Camassola, M., Jose Pinheiro Dillon, A., and Perez-Rueda, E. (2020). Gene regulatory networks of penicillium echinulatum 2hh and penicillium oxalicum 114-2 inferred by a computational biology approach. *Frontiers in Microbiology*, **11**, 2566.

Li, Z., Yao, G., Wu, R., Gao, L., Kan, Q., Liu, M., Yang, P., Liu, G., Qin, Y., Song, X., *et al.* (2015). Synergistic and dose-controlled regulation of cellulase gene expression in penicillium oxalicum. *PLoS Genet*, **11**(9), e1005509.

McDonnell, E., Strasser, K., and Tsang, A. (2018). Manual gene curation and functional annotation. In *Methods in Molecular Biology*, volume 1775, pages 185–208. Humana Press Inc.

Melé, A. (2020). *Django 3 By Example: Build powerful and reliable Python web applications from scratch*. Packt Publishing Ltd.

Mitchell, A. L., Attwood, T. K., Babbitt, P. C., Blum, M., Bork, P., Bridge, A., Brown, S. D., Chang, H.-Y., El-Gebali, S., Fraser, M. I., *et al.* (2018). Interpro in 2019: improving coverage, classification and access to protein sequence annotations. *Nucleic acids research*, **47**(D1), D351–D360.

Nesteruk, D. (2019). Interpreter. In *Design Patterns in. NET*, pages 221–236. Springer.

OmixBox, O. (2020). Omicsbox | biobam | bioinformatics made easy.

PostgreSQL, P. (2020). Postgresql: The world's most advanced open source database.

Python, P. (2020). Welcome to python.org.

RabbitMQ, R. (2020). Messaging that just works — rabbitmq.

Seppey, M., Manni, M., and Zdobnov, E. M. (2019). Busco: assessing genome assembly and annotation completeness. In *Gene Prediction*, pages 227–245. Springer.

Shaik, N. A., Elango, R., Khan, M., and Banaganapalli, B. (2019). Introduction to bioinformatics. In *Essentials of Bioinformatics, Volume I*, pages 1–18. Springer.

Wang, R., Bao, L., Liu, S., and Liu, Z. (2017). Genome annotation: Determination of the coding potential of the genome. *Bioinformatics in Aquaculture: Principles and Methods*, pages 74–85.

4.2 VALIDAÇÃO

O *workflow* foi testado e validado no processo de anotação genômica das linhagens selvagem (2HH) e mutante (S1M29) do fungo *Penicillium echinulatum* (LENZ *et al.*, 2020), conduzido pelo Laboratório de Bioinformática e Biologia Computacional da Universidade de Caxias do Sul (Caxias do Sul, Brasil). Uma versão *beta* do sistema foi utilizada, incorporando a última versão estável de cada ferramenta de anotação funcional disponível em Agosto de 2018. Ambas as linhagens 2HH e S1M29 foram submetidas ao banco de dados GenBank através dos *BioProjects* PRJNA520890 e PRJNA521489, respectivamente. O processo de anotação resultou em 8366 genes para 2HH e 8375 genes para S1M29.

Os Quadros 8, 9, 10 e 11 apresentam os resultados dos questionários de avaliação de qualidade aplicados, utilizados para coleta de *feedback* e posterior realização de aprimoramentos na solução, conforme as sugestões dos usuários.

Quadro 8 – Resposta do questionário de avaliação de qualidade do sistema: Usuário 1.

Questionário de avaliação de qualidade da ferramenta Seq2Annot					
Legenda: C (Concordo) | D (Discordo). Obrigatório informar observação para a opção "Discordo".					
Profissão	Mestrando em Biotecnologia				
Área de atuação	Bioinformática				
Aspecto	**Sub-aspecto**	**Item**	**C**	**D**	**Observação**
Funcionalidade	Adequação	O sistema dispõe de todas as funcionalidades necessárias para sua execução.	X		
Funcionalidade	Acurácia	O sistema faz o que foi proposto de forma correta.	X		
Funcionalidade	Acurácia	O sistema é preciso na execução de suas funções e nos resultados apresentados.	X		
Confiabilidade	Tolerância à falhas	O sistema não apresenta falhas com frequência.	X		
Confiabilidade	Tolerância à falhas	O sistema tem respostas adequadas quando ocorrem falhas.			Não houve falhas para avaliar.
Confiabilidade	Tolerância à falhas	O sistema informa ao usuário quando entradas de dados inválidas são inseridas.	X		
Confiabilidade	Recuperabilidade	O sistema é capaz de recuperar dados em caso de falhas.			Não houve falhas para avaliar.
Confiabilidade	Conformidade	O sistema está de acordo com as normas, leis, etc.	X		
Usabilidade	Inteligibilidade	É possível entender facilmente o conceito e a aplicação do sistema.	X		
Usabilidade	Inteligibilidade	É fácil de entender as diferentes funcionalidades oferecidas pelo sistema.	X		
Usabilidade	Inteligibilidade	O sistema apresenta os dados dispostos de forma coerente e de fácil entendimento.	X		
Usabilidade	Atratividade	O sistema possui uma interface atrativa.	X		
Usabilidade	Facilidade de uso	O sistema é fácil de operar e controlar.	X		
Usabilidade	Facilidade de uso	O sistema facilita a entrada de dados pelo usuário.	X		
Usabilidade	Facilidade de uso	O sistema facilita a saída de dados pelo usuário.	X		
Usabilidade	Operacionalidade	O sistema fornece ajuda de forma clara.		X	Senti falta de algumas opções de ajuda.
Eficiência	Tempo	O tempo de resposta do sistema é adequado.	X		
Compatibilidade	Interoperabilidade	Os diferentes módulos/seções do sistema interagem de forma correta.	X		
Compatibilidade	Interoperabilidade	O sistema tem capacidade de processamento multiusuário.	X		
Segurança	Confidencialidade	O sistema dispõe de segurança através de login, senhas e permissões de usuário.	X		
Segurança	Integridade	O sistema não apresenta inconsistência ou perda de dados.	X		

Fonte: Elaborada pelo autor.

Quadro 9 – Resposta do questionário de avaliação de qualidade do sistema: Usuário 2.

Questionário de avaliação de qualidade da ferramenta Seq2Annot					
Legenda: C (Concordo) | D (Discordo). Obrigatório informar observação para a opção "Discordo".					
Profissão	Estudante de Ciências Biológicas				
Área de atuação	Bioinformática e Botânica				
Aspecto	**Sub-aspecto**	**Item**	**C**	**D**	**Observação**
Funcionalidade	Adequação	O sistema dispõe de todas as funcionalidades necessárias para sua execução.	X		
Funcionalidade	Acurácia	O sistema faz o que foi proposto de forma correta.	X		
Funcionalidade	Acurácia	O sistema é preciso na execução de suas funções e nos resultados apresentados.	X		
Confiabilidade	Tolerância à falhas	O sistema não apresenta falhas com frequência.	X		
Confiabilidade	Tolerância à falhas	O sistema tem respostas adequadas quando ocorrem falhas.			Não houve falhas para avaliar.
Confiabilidade	Tolerância à falhas	O sistema informa ao usuário quando entradas de dados inválidas são inseridas.	X		
Confiabilidade	Recuperabilidade	O sistema é capaz de recuperar dados em caso de falhas.			Não houve falhas para avaliar.
Confiabilidade	Conformidade	O sistema está de acordo com as normas, leis, etc.	X		
Usabilidade	Inteligibilidade	É possível entender facilmente o conceito e a aplicação do sistema.	X		
Usabilidade	Inteligibilidade	É fácil de entender as diferentes funcionalidades oferecidas pelo sistema.	X		
Usabilidade	Inteligibilidade	O sistema apresenta os dados dispostos de forma coerente e de fácil entendimento.	X		
Usabilidade	Atratividade	O sistema possui uma interface atrativa.	X		A interface é satisfatória, tanto para cientistas da área da vida, quanto para pesquisadores da área computacional.
Usabilidade	Facilidade de uso	O sistema é fácil de operar e controlar.	X		O sistema é adequado para os usuários, não sendo necessário treinamento ou instuções prévias pra seu uso.
Usabilidade	Facilidade de uso	O sistema facilita a entrada de dados pelo usuário.	X		
Usabilidade	Facilidade de uso	O sistema facilita a saída de dados pelo usuário.	X		
Usabilidade	Operacionalidade	O sistema fornece ajuda de forma clara.		X	Não é possível identificar com facilidade informações sobre o sistema. Um aspecto que poderia ser aperfeiçoado é a geração de um relatório de erros para o usuário, conforme encontrado em outros bancos de dados biológicos.
Eficiência	Tempo	O tempo de resposta do sistema é adequado.	X		
Compatibilidade	Interoperabilidade	Os diferentes módulos/seções do sistema interagem de forma correta.	X		
Compatibilidade	Interoperabilidade	O sistema tem capacidade de processamento multiusuário.	X		
Segurança	Confidencialidade	O sistema dispõe de segurança através de login, senhas e permissões de usuário.	X		
Segurança	Integridade	O sistema não apresenta inconsistência ou perda de dados.	X		

Fonte: Elaborada pelo autor.

Quadro 10 – Resposta do questionário de avaliação de qualidade do sistema: Usuário 3.

Questionário de avaliação de qualidade da ferramenta Seq2Annot					
Legenda: C (Concordo) ǀ D (Discordo). Obrigatório informar observação para a opção "Discordo".					
Profissão		Professor Universitário			
Área de atuação		Bioinformática			
Aspecto	**Sub-aspecto**	**Item**	**C**	**D**	**Observação**
Funcionalidade	Adequação	O sistema dispõe de todas as funcionalidades necessárias para sua execução.	X		
Funcionalidade	Acurácia	O sistema faz o que foi proposto de forma correta.	X		
Funcionalidade	Acurácia	O sistema é preciso na execução de suas funções e nos resultados apresentados.	X		
Confiabilidade	Tolerância à falhas	O sistema não apresenta falhas com frequência.	X		
Confiabilidade	Tolerância à falhas	O sistema tem respostas adequadas quando ocorrem falhas.	X		
Confiabilidade	Tolerância à falhas	O sistema informa ao usuário quando entradas de dados inválidas são inseridas.	X		
Confiabilidade	Recuperabilidade	O sistema é capaz de recuperar dados em caso de falhas.	X		
Confiabilidade	Conformidade	O sistema está de acordo com as normas, leis, etc.	X		
Usabilidade	Inteligibilidade	É possível entender facilmente o conceito e a aplicação do sistema.	X		
Usabilidade	Inteligibilidade	É fácil de entender as diferentes funcionalidades oferecidas pelo sistema.	X		
Usabilidade	Inteligibilidade	O sistema apresenta os dados dispostos de forma coerente e de fácil entendimento.		X	A busca automática após digitar cada caractere dificulta as pesquisas.
Usabilidade	Atratividade	O sistema possui uma interface atrativa.	X		
Usabilidade	Facilidade de uso	O sistema é fácil de operar e controlar.	X		
Usabilidade	Facilidade de uso	O sistema facilita a entrada de dados pelo usuário.	X		
Usabilidade	Facilidade de uso	O sistema facilita a saída de dados pelo usuário.		X	Exportar informações em diversos formatos e não somente para depósito no GenBank.
Usabilidade	Operacionalidade	O sistema fornece ajuda de forma clara.	X		
Eficiência	Tempo	O tempo de resposta do sistema é adequado.		X	O sistema depende de acesso a web services e a disponibilidade e carga dos web services torna algumas análises muito demoradas.
Compatibilidade	Interoperabilidade	Os diferentes módulos/seções do sistema interagem de forma correta.	X		
Compatibilidade	Interoperabilidade	O sistema tem capacidade de processamento multiusuário.	X		
Segurança	Confidencialidade	O sistema dispõe de segurança através de login, senhas e permissões de usuário.	X		
Segurança	Integridade	O sistema não apresenta inconsistência ou perda de dados.	X		

Fonte: Elaborada pelo autor.

Quadro 11 – Resposta do questionário de avaliação de qualidade do sistema: Usuário 4.

Questionário de avaliação de qualidade da ferramenta Seq2Annot					
Legenda: C (Concordo) ǀ D (Discordo). Obrigatório informar observação para a opção "Discordo".					
Profissão		Professor Universitário			
Área de atuação		Bioinformática			
Aspecto	**Sub-aspecto**	**Item**	**C**	**D**	**Observação**
Funcionalidade	Adequação	O sistema dispõe de todas as funcionalidades necessárias para sua execução.	X		
Funcionalidade	Acurácia	O sistema faz o que foi proposto de forma correta.	X		
Funcionalidade	Acurácia	O sistema é preciso na execução de suas funções e nos resultados apresentados.	X		
Confiabilidade	Tolerância à falhas	O sistema não apresenta falhas com frequência.	X		
Confiabilidade	Tolerância à falhas	O sistema tem respostas adequadas quando ocorrem falhas.	X		
Confiabilidade	Tolerância à falhas	O sistema informa ao usuário quando entradas de dados inválidas são inseridas.	X		
Confiabilidade	Recuperabilidade	O sistema é capaz de recuperar dados em caso de falhas.	X		
Confiabilidade	Conformidade	O sistema está de acordo com as normas, leis, etc.	X		
Usabilidade	Inteligibilidade	É possível entender facilmente o conceito e a aplicação do sistema.	X		
Usabilidade	Inteligibilidade	É fácil de entender as diferentes funcionalidades oferecidas pelo sistema.	X		
Usabilidade	Inteligibilidade	O sistema apresenta os dados dispostos de forma coerente e de fácil entendimento.	X		
Usabilidade	Atratividade	O sistema possui uma interface atrativa.	X		
Usabilidade	Facilidade de uso	O sistema é fácil de operar e controlar.	X		
Usabilidade	Facilidade de uso	O sistema facilita a entrada de dados pelo usuário.	X		
Usabilidade	Facilidade de uso	O sistema facilita a saída de dados pelo usuário.	X		
Usabilidade	Operacionalidade	O sistema fornece ajuda de forma clara.	X		
Eficiência	Tempo	O tempo de resposta do sistema é adequado.	X		
Compatibilidade	Interoperabilidade	Os diferentes módulos/seções do sistema interagem de forma correta.	X		
Compatibilidade	Interoperabilidade	O sistema tem capacidade de processamento multiusuário.	X		
Segurança	Confidencialidade	O sistema dispõe de segurança através de login, senhas e permissões de usuário.	X		
Segurança	Integridade	O sistema não apresenta inconsistência ou perda de dados.	X		

Fonte: Elaborada pelo autor.

5 CONCLUSÃO

O *workflow* de anotação genômica funcional e curadoria manual de genomas, denominado Seq2Annot, foi desenvolvido para abstrair a complexidade das etapas de anotação funcional, curadoria manual e publicação de um Projeto Genoma, através da criação de um fluxo de anotação centralizado e organizado em uma plataforma *web* colaborativa. Baseando-se em *pipelines* encontradas na literatura, seis ferramentas de anotação funcional foram selecionadas e incorporadas à solução, proporcionando aos usuários o agendamento de consultas automáticas a 10 bancos de dados relacionados a domínios de proteínas e grupos de ortólogos. Os resultados provenientes da execução de cada ferramenta são organizados em um contexto objetivo e unificado, permitindo a realização de filtros complexos para executar análises em partes específicas do genoma.

A solução ainda permite a edição de quaisquer informações estruturais ou funcionais para cada gene, tanto para a simples atribuição de nomes aos produtos gênicos quanto para a realização do processo de curadoria manual. Uma vez concluída a anotação funcional, é possível exportar os resultados de anotação, convertendo-os automaticamente para os formatos exigidos para publicação no banco de dados GenBank. O suporte a todas estas etapas tornou a ferramenta uma solução que apoia os pesquisadores em todas as fases de anotação genômica subsequentes à fase de anotação estrutural, suportando o trabalho colaborativo de múltiplos usuários em um mesmo projeto e eliminando a necessidade de utilização de quaisquer ferramentas, pacotes ou dependências externas.

A solução foi testada e validada no processo de anotação genômica das linhagens selvagem (2HH) e mutante (S1M29) do fungo *Penicillium echinulatum*, conduzido pelo Laboratório de Bioinformática e Biologia Computacional da Universidade de Caxias do Sul e submetido ao banco e dados GenBank através dos *BioProjects* PRJNA520890 e PRJNA521489, respectivamente. Questionários de avaliação de qualidade também foram aplicados para coletar *feedback* dos usuários e realizar melhorias.

O mecanismo central de execução de tarefas assíncronas tornou a solução largamente escalável, permitindo quem novos recursos de servidor sejam alocados de forma horizontal à medida que houver o aumento de demanda. A decisão arquitetural de dividir trabalhos em pequenas tarefas assíncronas independentes também garantiu que novas ferramentas de anotação funcional possam ser facilmente desenvolvidas e acopladas à ferramenta, tornando-a uma solução promissora para ser constantemente aprimorada e utilizada em projetos de anotação genômica em larga escala no futuro. Além disso, trabalhos futuros podem incorporar a execução automática de *assemblers* e preditores de genes para apoiar também as fases anteriores de montagem e anotação estrutural, transformando assim a ferramenta Seq2Annot em uma completa *pipeline online* colaborativa de anotação genômica.

6 REFERÊNCIAS

ABNIZOVA, I.; BOEKHORST, R. te; ORLOV, Y. Computational errors and biases of short read next generation sequencing. **J Proteomics Bioinform**, v. 10, n. 1, p. 1–17, 2017. Citado na página 24.

ABRIL, J. F.; CASTELLANO, S. Genome Annotation. In: **Encyclopedia of Bioinformatics and Computational Biology**. Elsevier, 2019. p. 195–209. Disponível em: <https://linkinghub.elsevier.com/retrieve/pii/B9780128096338202264>. Citado na página 34.

ACENCIO, M. L.; LÆGREID, A.; KUIPER, M. *et al.* The gene ontology resource: 20 years and still going strong. Oxford Academic, 2019. Citado 3 vezes nas páginas 13, 29 e 35.

AKIVA, E.; BROWN, S.; ALMONACID, D. E.; 2ND, A. E. B.; CUSTER, A. F.; HICKS, M. A.; HUANG, C. C.; LAUCK, F.; MASHIYAMA, S. T.; MENG, E. C. *et al.* The structure–function linkage database. **Nucleic acids research**, Oxford University Press, v. 42, n. D1, p. D521–D530, 2014. Citado na página 29.

ALTSCHUL, S. F.; GISH, W.; MILLER, W.; MYERS, E. W.; LIPMAN, D. J. Basic local alignment search tool. **Journal of molecular biology**, Elsevier, v. 215, n. 3, p. 403–410, 1990. Citado na página 13.

ANDREWS, S. *et al.* **FastQC: a quality control tool for high throughput sequence data**. [S.l.]: Babraham Bioinformatics, Babraham Institute, Cambridge, United Kingdom, 2010. Citado na página 24.

ANDRIEU, O.; FISTON, A.-S.; ANXOLABÉHÈRE, D.; QUESNEVILLE, H. Detection of transposable elements by their compositional bias. **BMC bioinformatics**, BioMed Central, v. 5, n. 1, p. 94, 2004. Citado na página 32.

ANGEL, V. D. D.; HJERDE, E.; STERCK, L.; CAPELLA-GUTIERREZ, S.; NOTREDAME, C.; PETTERSSON, O. V.; AMSELEM, J.; BOURI, L.; BOCS, S.; KLOPP, C. *et al.* Ten steps to get started in genome assembly and annotation. **F1000Research**, Faculty of 1000 Ltd, v. 7, 2018. Citado 9 vezes nas páginas 13, 14, 24, 25, 32, 33, 34, 35 e 38.

ANNACHHATRE, C.; AUSTIN, T. H.; STAMP, M. Hidden markov models for malware classification. **Journal of Computer Virology and Hacking Techniques**, Springer, v. 11, n. 2, p. 59–73, 2015. Citado na página 28.

ARAUJO, F. A.; BARH, D.; SILVA, A.; GUIMARÃES, L.; RAMOS, R. T. J. GO FEAT: A rapid web-based functional annotation tool for genomic and transcriptomic data. **Scientific Reports**, Nature Publishing Group, v. 8, n. 1, dec 2018. ISSN 20452322. Citado 4 vezes nas páginas 13, 37, 40 e 47.

ARMENTEROS, J. J. A.; SALVATORE, M.; EMANUELSSON, O.; WINTHER, O.; HEIJNE, G. V.; ELOFSSON, A.; NIELSEN, H. Detecting sequence signals in targeting peptides using deep learning. **Life science alliance**, Life Science Alliance, v. 2, n. 5, 2019. Citado na página 42.

ARMENTEROS, J. J. A.; TSIRIGOS, K. D.; SØNDERBY, C. K.; PETERSEN, T. N.; WINTHER, O.; BRUNAK, S.; HEIJNE, G. von; NIELSEN, H. Signalp 5.0 improves signal peptide predictions using deep neural networks. **Nature biotechnology**, Nature Publishing Group, v. 37, n. 4, p. 420–423, 2019. Citado 2 vezes nas páginas 13 e 36.

ASHBURNER, M.; BALL, C. A.; BLAKE, J. A.; BOTSTEIN, D.; BUTLER, H.; CHERRY, J. M.; DAVIS, A. P.; DOLINSKI, K.; DWIGHT, S. S.; EPPIG, J. T. *et al.* Gene ontology: tool for the unification of biology. **Nature genetics**, Nature Publishing Group, v. 25, n. 1, p. 25, 2000. Citado 3 vezes nas páginas 13, 29 e 35.

ATTWOOD, T. K.; COLETTA, A.; MUIRHEAD, G.; PAVLOPOULOU, A.; PHILIPPOU, P. B.; POPOV, I.; ROMA-MATEO, C.; THEODOSIOU, A.; MITCHELL, A. L. The prints database: a fine-grained protein sequence annotation and analysis resource—its status in 2012. **Database**, Narnia, v. 2012, 2012. Citado 2 vezes nas páginas 13 e 29.

AZIZ, R. K.; BARTELS, D.; BEST, A. A.; DEJONGH, M.; DISZ, T.; EDWARDS, R. A.; FORMSMA, K.; GERDES, S.; GLASS, E. M.; KUBAL, M. *et al.* The rast server: rapid annotations using subsystems technology. **BMC genomics**, Springer, v. 9, n. 1, p. 75, 2008. Citado na página 42.

BANKEVICH, A.; NURK, S.; ANTIPOV, D.; GUREVICH, A. A.; DVORKIN, M.; KULIKOV, A. S.; LESIN, V. M.; NIKOLENKO, S. I.; PHAM, S.; PRJIBELSKI, A. D. *et al.* Spades: a new genome assembly algorithm and its applications to single-cell sequencing. **Journal of computational biology**, Mary Ann Liebert, Inc. 140 Huguenot Street, 3rd Floor New Rochelle, NY 10801 USA, v. 19, n. 5, p. 455–477, 2012. Citado na página 45.

BAO, W.; KOJIMA, K. K.; KOHANY, O. Repbase update, a database of repetitive elements in eukaryotic genomes. **Mobile Dna**, Springer, v. 6, n. 1, p. 11, 2015. Citado na página 32.

BAO, Z.; EDDY, S. R. Automated de novo identification of repeat sequence families in sequenced genomes. **Genome research**, Cold Spring Harbor Lab, v. 12, n. 8, p. 1269–1276, 2002. Citado na página 32.

BEAULAURIER, J.; SCHADT, E. E.; FANG, G. Deciphering bacterial epigenomes using modern sequencing technologies. **Nature Reviews Genetics**, Nature Publishing Group, v. 20, n. 3, p. 157–172, 2019. Citado na página 23.

BENSON, D. A.; CAVANAUGH, M.; CLARK, K.; KARSCH-MIZRACHI, I.; OSTELL, J.; PRUITT, K. D.; SAYERS, E. W. Genbank. **Nucleic acids research**, Oxford University Press, v. 46, n. D1, p. D41–D47, 2018. Citado 3 vezes nas páginas 12, 26 e 38.

BESSER, J.; CARLETON, H. A.; GERNER-SMIDT, P.; LINDSEY, R. L.; TREES, E. Next-generation sequencing technologies and their application to the study and control of bacterial infections. **Clinical microbiology and infection**, Elsevier, v. 24, n. 4, p. 335–341, 2018. Citado na página 12.

BHATT, V. D.; PATEL, M.; JOSHI, C. G. An insight of biological databases used in bioinformatics. In: **Current trends in Bioinformatics: An Insight**. [S.l.]: Springer, 2018. p. 3–25. Citado 2 vezes nas páginas 26 e 27.

BLIN, K.; WOLF, T.; CHEVRETTE, M. G.; LU, X.; SCHWALEN, C. J.; KAUTSAR, S. A.; DURAN, H. G. S.; SANTOS, E. L. de L.; KIM, H. U.; NAVE, M. *et al.* antismash 4.0—improvements in chemistry prediction and gene cluster boundary identification. **Nucleic acids research**, Oxford University Press, v. 45, n. W1, p. W36–W41, 2017. Citado na página 36.

BOLGER, A. M.; LOHSE, M.; USADEL, B. Trimmomatic: a flexible trimmer for illumina sequence data. **Bioinformatics**, Oxford University Press, v. 30, n. 15, p. 2114–2120, 2014. Citado na página 24.

BOOTSTRAP, B. **Bootstrap: The most popular HTML, CSS, and JS library in the world**. 2020. Disponível em: <https://getbootstrap.com>. Citado na página 49.

BORGES-OSÓRIO, M. R.; ROBINSON, W. M. **Genética Humana 3ed**. [S.l.]: Artmed Editora, 2013. Citado na página 18.

BRETTIN, T.; DAVIS, J. J.; DISZ, T.; EDWARDS, R. A.; GERDES, S.; OLSEN, G. J.; OLSON, R.; OVERBEEK, R.; PARRELLO, B.; PUSCH, G. D. *et al.* Rasttk: a modular and extensible implementation of the rast algorithm for building custom annotation pipelines and annotating batches of genomes. **Scientific reports**, Nature Publishing Group, v. 5, p. 8365, 2015. Citado na página 42.

BRU, C.; COURCELLE, E.; CARRÈRE, S.; BEAUSSE, Y.; DALMAR, S.; KAHN, D. The prodom database of protein domain families: more emphasis on 3d. **Nucleic acids research**, Oxford University Press, v. 33, n. suppl_1, p. D212–D215, 2005. Citado na página 29.

BRYANT, D. M.; JOHNSON, K.; DITOMMASO, T.; TICKLE, T.; COUGER, M. B.; PAYZIN-DOGRU, D.; LEE, T. J.; LEIGH, N. D.; KUO, T.-H.; DAVIS, F. G. *et al.* A tissue-mapped axolotl de novo transcriptome enables identification of limb regeneration factors. **Cell reports**, Elsevier, v. 18, n. 3, p. 762–776, 2017. Citado na página 37.

BUCHFINK, B.; XIE, C.; HUSON, D. H. Fast and sensitive protein alignment using diamond. **Nature methods**, Nature Publishing Group, v. 12, n. 1, p. 59, 2015. Citado na página 42.

BURGE, C. B.; KARLIN, S. Finding the genes in genomic dna. **Current opinion in structural biology**, Elsevier, v. 8, n. 3, p. 346–354, 1998. Citado na página 42.

BUSK, P. K.; LANGE, L. Function-based classification of carbohydrate-active enzymes by recognition of short, conserved peptide motifs. **Appl. Environ. Microbiol.**, Am Soc Microbiol, v. 79, n. 11, p. 3380–3391, 2013. Citado na página 30.

CALZAVARA, S.; CONTI, M.; FOCARDI, R.; RABITTI, A.; TOLOMEI, G. Machine learning for web vulnerability detection: The case of cross-site request forgery. **IEEE Security & Privacy**, IEEE, 2020. Citado na página 52.

CANTAREL, B. L.; KORF, I.; ROBB, S. M.; PARRA, G.; ROSS, E.; MOORE, B.; HOLT, C.; ALVARADO, A. S.; YANDELL, M. Maker: an easy-to-use annotation pipeline designed for emerging model organism genomes. **Genome research**, Cold Spring Harbor Lab, v. 18, n. 1, p. 188–196, 2008. Citado na página 34.

CARBON, S.; IRELAND, A.; MUNGALL, C. J.; SHU, S.; MARSHALL, B.; LEWIS, S.; HUB, A.; GROUP, W. P. W. Amigo: online access to ontology and annotation data. **Bioinformatics**, Oxford University Press, v. 25, n. 2, p. 288–289, 2009. Citado na página 35.

CARRAU, J.; DILLON, A.; RIBEIRO, R.; LEYGUE-ALBA, N.; AZEVEDO, J. Produção de enzimas celulolíticas por microrganismos. **Simpósio Internacioanl de Engenharia Genética. Anais**, v. 39, p. 205–210, 1981. Citado na página 45.

CELERY, C. **Celery: Distributed Task Queue**. 2020. Disponível em: <http://www.celeryproject.org>. Citado na página 48.

CHAITANYA, K. Genome sequencing, assembly, and annotation. In: **Genome and Genomics**. [S.l.]: Springer, 2019. p. 181–210. Citado 2 vezes nas páginas 16 e 21.

CHIKHI, R.; MEDVEDEV, P. Informed and automated k-mer size selection for genome assembly. **Bioinformatics**, Oxford University Press, v. 30, n. 1, p. 31–37, 2014. Citado na página 45.

CHOULET, F.; WICKER, T.; RUSTENHOLZ, C.; PAUX, E.; SALSE, J.; LEROY, P.; SCHLUB, S.; PASLIER, M.-C. L.; MAGDELENAT, G.; GONTHIER, C. *et al.* Megabase level sequencing reveals contrasted organization and evolution patterns of the wheat gene and transposable element spaces. **The Plant Cell**, Am Soc Plant Biol, v. 22, n. 6, p. 1686–1701, 2010. Citado na página 31.

CHRISTENSEN, H.; VRIES, L. E. de. Databases and protein structures. In: **Introduction to Bioinformatics in Microbiology**. [S.l.]: Springer, 2018. p. 25–50. Citado 2 vezes nas páginas 26 e 27.

COCK, P. J.; ANTAO, T.; CHANG, J. T.; CHAPMAN, B. A.; COX, C. J.; DALKE, A.; FRIEDBERG, I.; HAMELRYCK, T.; KAUFF, F.; WILCZYNSKI, B. *et al.* Biopython: freely available python tools for computational molecular biology and bioinformatics. **Bioinformatics**, Oxford University Press, v. 25, n. 11, p. 1422–1423, 2009. Citado 2 vezes nas páginas 39 e 49.

CONESA, A.; GÖTZ, S.; GARCÍA-GÓMEZ, J. M.; TEROL, J.; TALÓN, M.; ROBLES, M. Blast2go: a universal tool for annotation, visualization and analysis in functional genomics research. **Bioinformatics**, Oxford University Press, v. 21, n. 18, p. 3674–3676, 2005. Citado na página 36.

CONSORTIUM, U. Uniprot: a worldwide hub of protein knowledge. **Nucleic acids research**, Oxford University Press, v. 47, n. D1, p. D506–D515, 2018. Citado na página 12.

____. ____. **Nucleic acids research**, Oxford University Press, v. 47, n. D1, p. D506–D515, 2019. Citado 2 vezes nas páginas 13 e 28.

CRUZ, F.; LAGOA, D.; MENDES, J.; ROCHA, I.; FERREIRA, E. C.; ROCHA, M.; DIAS, O. SamPler – a novel method for selecting parameters for gene functional annotation routines. **BMC Bioinformatics**, v. 20, n. 1, p. 454, dec 2019. ISSN 1471-2105. Disponível em: <https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-019-3038-4>. Citado 5 vezes nas páginas 13, 14, 34, 35 e 36.

DILLON, A.; BETTIO, M.; POZZAN, F.; ANDRIGHETTI, T.; CAMASSOLA, M. A new penicillium echinulatum strain with faster cellulase secretion obtained using hydrogen peroxide mutagenesis and screening with 2-deoxyglucose. **Journal of applied microbiology**, Wiley Online Library, v. 111, n. 1, p. 48–53, 2011. Citado na página 45.

DJANGO, D. **The Web framework for perfectionists with deadlines**. 2020. Disponível em: <https://www.djangoproject.com>. Citado na página 47.

DJANGOCELERYRESULTS, D. **Celery result back end with django**. 2020. Disponível em: <https://github.com/celery/django-celery-results>. Citado na página 49.

DUNN, N. A.; UNNI, D. R.; DIESH, C.; MUNOZ-TORRES, M.; HARRIS, N. L.; YAO, E.; RASCHE, H.; HOLMES, I. H.; ELSIK, C. G.; LEWIS, S. E. Apollo: Democratizing genome annotation. **PLoS computational biology**, Public Library of Science, v. 15, n. 2, p. e1006790, 2019. Citado na página 38.

EBI. **Ion Torrent: Proton / PGM sequencing**. 2019. Disponível em: <https://www.ebi.ac.uk/training/online/course/ebi-next-generation-sequencing-practical-course/what-next-generation-dna-sequencing/ion-torre>. Citado na página 22.

EDGAR, R. C.; MYERS, E. W. Piler: identification and classification of genomic repeats. **Bioinformatics**, Oxford University Press, v. 21, n. suppl_1, p. i152–i158, 2005. Citado na página 32.

EL-GEBALI, S.; MISTRY, J.; BATEMAN, A.; EDDY, S. R.; LUCIANI, A.; POTTER, S. C.; QURESHI, M.; RICHARDSON, L. J.; SALAZAR, G. A.; SMART, A. *et al.* The pfam protein families database in 2019. **Nucleic acids research**, Oxford University Press, v. 47, n. D1, p. D427–D432, 2019. Citado 2 vezes nas páginas 13 e 28.

FINN, R. D.; CLEMENTS, J.; EDDY, S. R. Hmmer web server: interactive sequence similarity searching. **Nucleic acids research**, Oxford University Press, v. 39, n. suppl_2, p. W29–W37, 2011. Citado na página 50.

FLEISCHMANN, R. D.; ADAMS, M. D.; WHITE, O.; CLAYTON, R. A.; KIRKNESS, E. F.; KERLAVAGE, A. R.; BULT, C. J.; TOMB, J.-F.; DOUGHERTY, B. A.; MERRICK, J. M. *et al.* Whole-genome random sequencing and assembly of haemophilus influenzae rd. **Science**, American Association for the Advancement of Science, v. 269, n. 5223, p. 496–512, 1995. Citado na página 12.

FLUTRE, T.; DUPRAT, E.; FEUILLET, C.; QUESNEVILLE, H. Considering transposable element diversification in de novo annotation approaches. **PloS one**, Public Library of Science, v. 6, n. 1, 2011. Citado na página 32.

FLYNN, J. M.; HUBLEY, R.; GOUBERT, C.; ROSEN, J.; CLARK, A. G.; FESCHOTTE, C.; SMIT, A. F. Repeatmodeler2: automated genomic discovery of transposable element families. **BioRxiv**, Cold Spring Harbor Laboratory, p. 856591, 2019. Citado na página 32.

GABALDÓN, T.; KOONIN, E. V. Functional and evolutionary implications of gene orthology. **Nature Reviews Genetics**, Nature Publishing Group, v. 14, n. 5, p. 360–366, 2013. Citado 2 vezes nas páginas 30 e 36.

GALPERIN, M. Y.; KRISTENSEN, D. M.; MAKAROVA, K. S.; WOLF, Y. I.; KOONIN, E. V. Microbial genome analysis: the cog approach. **Briefings in bioinformatics**, Oxford University Press, v. 20, n. 4, p. 1063–1070, 2019. Citado 2 vezes nas páginas 13 e 30.

GEIB, S. M.; HALL, B.; DEREGO, T.; BREMER, F. T.; CANNOLES, K.; SIM, S. B. Genome annotation generator: a simple tool for generating and correcting wgs annotation tables for ncbi submission. **GigaScience**, Oxford University Press, v. 7, n. 4, p. giy018, 2018. Citado 2 vezes nas páginas 38 e 39.

GIANI, A. M.; GALLO, G. R.; GIANFRANCESCHI, L.; FORMENTI, G. Long walk to genomics: History and current approaches to genome sequencing and assembly. **Computational and Structural Biotechnology Journal**, Elsevier, 2019. Citado 4 vezes nas páginas 12, 17, 20 e 22.

GIRGIS, H. Z. Red: an intelligent, rapid, accurate tool for detecting repeats de-novo on the genomic scale. **BMC bioinformatics**, BioMed Central, v. 16, n. 1, p. 227, 2015. Citado na página 32.

GOEL, N.; KARIR, P.; GARG, V. K. Role of dna methylation in human age prediction. **Mechanisms of ageing and development**, Elsevier, v. 166, p. 33–41, 2017. Citado na página 31.

GOLDBERG, J. M.; GRIGGS, A. D.; SMITH, J. L.; HAAS, B. J.; WORTMAN, J. R.; ZENG, Q. Kinannote, a computer program to identify and classify members of the eukaryotic protein kinase superfamily. **Bioinformatics**, Oxford University Press, v. 29, n. 19, p. 2387–2394, 2013. Citado na página 36.

GUREVICH, A.; SAVELIEV, V.; VYAHHI, N.; TESLER, G. Quast: quality assessment tool for genome assemblies. **Bioinformatics**, Oxford University Press, v. 29, n. 8, p. 1072–1075, 2013. Citado na página 25.

HAAS, B. J.; SALZBERG, S. L.; ZHU, W.; PERTEA, M.; ALLEN, J. E.; ORVIS, J.; WHITE, O.; BUELL, C. R.; WORTMAN, J. R. Automated eukaryotic gene structure annotation using evidencemodeler and the program to assemble spliced alignments. **Genome biology**, Springer, v. 9, n. 1, p. R7, 2008. Citado na página 46.

HAFT, D. H.; SELENGUT, J. D.; RICHTER, R. A.; HARKINS, D.; BASU, M. K.; BECK, E. Tigrfams and genome properties in 2013. **Nucleic acids research**, Oxford University Press, v. 41, n. D1, p. D387–D395, 2012. Citado na página 29.

HALFOND, W. G.; VIEGAS, J.; ORSO, A. *et al.* A classification of sql-injection attacks and countermeasures. In: IEEE. **Proceedings of the IEEE international symposium on secure software engineering**. [S.l.], 2006. v. 1, p. 13–15. Citado na página 53.

HARIDAS, S.; SALAMOV, A.; GRIGORIEV, I. V. Fungal genome annotation. In: **Fungal Genomics**. [S.l.]: Springer, 2018. p. 171–184. Citado 2 vezes nas páginas 13 e 35.

HERNANDEZ, D.; FRANÇOIS, P.; FARINELLI, L.; ØSTERÅS, M.; SCHRENZEL, J. De novo bacterial genome sequencing: millions of very short reads assembled on a desktop computer. **Genome research**, Cold Spring Harbor Lab, v. 18, n. 5, p. 802–809, 2008. Citado na página 25.

HMMER. 2020. Disponível em: <http://hmmer.org/>. Citado na página 13.

HOFF, K. J.; LOMSADZE, A.; BORODOVSKY, M.; STANKE, M. Whole-genome annotation with braker. In: **Gene Prediction**. [S.l.]: Springer, 2019. p. 65–95. Citado na página 34.

HUANG, Y.-F.; CHEN, S.-C.; CHIANG, Y.-S.; CHEN, T.-H.; CHIU, K.-P. Palindromic sequence impedes sequencing-by-ligation mechanism. In: SPRINGER. **BMC systems biology**. [S.l.], 2012. v. 6, n. S2, p. S10. Citado na página 21.

HUERTA-CEPAS, J.; FORSLUND, K.; COELHO, L. P.; SZKLARCZYK, D.; JENSEN, L. J.; MERING, C. V.; BORK, P. Fast genome-wide functional annotation through orthology assignment by eggnog-mapper. **Molecular biology and evolution**, Oxford University Press, v. 34, n. 8, p. 2115–2122, 2017. Citado na página 51.

HUERTA-CEPAS, J.; SZKLARCZYK, D.; FORSLUND, K.; COOK, H.; HELLER, D.; WALTER, M. C.; RATTEI, T.; MENDE, D. R.; SUNAGAWA, S.; KUHN, M. *et al.* eggnog 4.5: a hierarchical orthology framework with improved functional annotations for eukaryotic, prokaryotic and viral sequences. **Nucleic acids research**, Oxford University Press, v. 44, n. D1, p. D286–D293, 2016. Citado 2 vezes nas páginas 13 e 30.

HUMANN, J. L.; LEE, T.; FICKLIN, S.; MAIN, D. Structural and functional annotation of eukaryotic genomes with gensas. In: **Gene Prediction**. [S.l.]: Springer, 2019. p. 29–51. Citado 2 vezes nas páginas 13 e 42.

ILLUMINA, I. Truseq dna sample preparation guide, part 15026486 rev. c. **URL http://support.illumina.com**, 2012. Citado na página 45.

____. Truseq rna sample preparation v2 guide, part 15026495, rev. f. **URL http://support.illumina.com**, 2014. Citado na página 45.

ISO25010. **ISO/IEC 25010: International Organization for Standardization**. [S.l.], 2011. Citado na página 53.

JADEJA, Y.; MODI, K. Cloud computing-concepts, architecture and challenges. In: IEEE. **2012 International Conference on Computing, Electronics and Electrical Technologies (ICCEET)**. [S.l.], 2012. p. 877–880. Citado 2 vezes nas páginas 46 e 49.

JENSEN, R.; LOPES, M. H. B. d. M.; SILVEIRA, P. S. P.; ORTEGA, N. R. S. Desenvolvimento e avaliação de um software que verifica a acurácia diagnóstica. **Revista da Escola de Enfermagem da USP**, SciELO Brasil, v. 46, n. 1, p. 184–191, 2012. Citado na página 54.

JOBSTRAIBIZER, F. **Guia Profissional PHP**. [S.l.]: Digerati Books, 2009. Citado na página 47.

JONES, P.; BINNS, D.; CHANG, H.-Y.; FRASER, M.; LI, W.; MCANULLA, C.; MCWILLIAM, H.; MASLEN, J.; MITCHELL, A.; NUKA, G. *et al.* Interproscan 5: genome-scale protein function classification. **Bioinformatics**, Oxford University Press, v. 30, n. 9, p. 1236–1240, 2014. Citado na página 29.

JQUERY, j. **jQuery: Write less, do more**. 2020. Disponível em: <https://jquery.com>. Citado na página 49.

KANEHISA, M.; SATO, Y.; KAWASHIMA, M.; FURUMICHI, M.; TANABE, M. Kegg as a reference resource for gene and protein annotation. **Nucleic acids research**, Oxford University Press, v. 44, n. D1, p. D457–D462, 2016. Citado 2 vezes nas páginas 13 e 30.

KANZ, C.; ALDEBERT, P.; ALTHORPE, N.; BAKER, W.; BALDWIN, A.; BATES, K.; BROWNE, P.; BROEK, A. van den; CASTRO, M.; COCHRANE, G. *et al.* The embl nucleotide sequence database. **Nucleic acids research**, Oxford University Press, v. 33, n. suppl_1, p. D29–D33, 2005. Citado 2 vezes nas páginas 12 e 26.

KCHOUK, M.; GIBRAT, J.-F.; ELLOUMI, M. Generations of sequencing technologies: From first to next generation. **Biology and Medicine**, HATASO Enterprises LLC, v. 9, n. 3, 2017. Citado 3 vezes nas páginas 19, 22 e 23.

KENT, W. J. Blat—the blast-like alignment tool. **Genome research**, Cold Spring Harbor Lab, v. 12, n. 4, p. 656–664, 2002. Citado na página 46.

KHAN, A. R.; PERVEZ, M. T.; BABAR, M. E.; NAVEED, N.; SHOAIB, M. A comprehensive study of de novo genome assemblers: current challenges and future prospective. **Evolutionary Bioinformatics**, SAGE Publications Sage UK: London, England, v. 14, p. 1176934318758650, 2018. Citado na página 25.

KOCH, H.-G.; MOSER, M.; MÜLLER, M. Signal recognition particle-depencent protein targeting, universal to all kingdoms of life. In: **Reviews of physiology, biochemistry and pharmacology**. [S.l.]: Springer, 2003. p. 55–94. Citado na página 36.

KORF, I. Gene finding in novel genomes. **BMC bioinformatics**, BioMed Central, v. 5, n. 1, p. 59, 2004. Citado na página 42.

KRIVENTSEVA, E. V.; KUZNETSOV, D.; TEGENFELDT, F.; MANNI, M.; DIAS, R.; SIMÃO, F. A.; ZDOBNOV, E. M. Orthodb v10: sampling the diversity of animal, plant, fungal, protist, bacterial and viral genomes for evolutionary and functional annotations of orthologs. **Nucleic acids research**, Oxford University Press, v. 47, n. D1, p. D807–D811, 2019. Citado 2 vezes nas páginas 13 e 30.

KROGH, A.; LARSSON, B.; HEIJNE, G. V.; SONNHAMMER, E. L. Predicting transmembrane protein topology with a hidden markov model: application to complete genomes. **Journal of molecular biology**, Elsevier, v. 305, n. 3, p. 567–580, 2001. Citado 2 vezes nas páginas 13 e 36.

KRUEGER, F. Trim galore. **A wrapper tool around Cutadapt and FastQC to consistently apply quality and adapter trimming to FastQ files**, v. 516, p. 517, 2015. Citado na página 24.

LEINONEN, R.; SUGAWARA, H.; SHUMWAY, M.; COLLABORATION, I. N. S. D. The sequence read archive. **Nucleic acids research**, Oxford University Press, v. 39, n. suppl_1, p. D19–D21, 2010. Citado na página 27.

LENZ, A. R.; GALAN-VASQUEZ, E.; BALBINOT, E.; ABREU, F. Pessi de; OLIVEIRA, N. Souza de; ROSA, L. Osorio da; SILVA, S. de Avila e; CAMASSOLA, M.; DILLON, A. J. P.; PEREZ-RUEDA, E. Gene regulatory networks of penicillium echinulatum 2hh and penicillium oxalicum 114-2 inferred by a computational biology approach. **Frontiers in Microbiology**, Frontiers, v. 11, p. 2566, 2020. Citado 2 vezes nas páginas 14 e 63.

LETUNIC, I.; BORK, P. 20 years of the smart protein domain annotation resource. **Nucleic acids research**, Oxford University Press, v. 46, n. D1, p. D493–D496, 2018. Citado 2 vezes nas páginas 13 e 29.

LEWIS, T. E.; SILLITOE, I.; DAWSON, N.; LAM, S. D.; CLARKE, T.; LEE, D.; ORENGO, C.; LEES, J. Gene3d: extensive prediction of globular domains in proteins. **Nucleic acids research**, Oxford University Press, v. 46, n. D1, p. D435–D439, 2018. Citado na página 29.

LI, N.; WANG, X.; LIU, Z. Next-generation sequencing technologies and the assembly of short reads into reference genome sequences: Principles and methods. **Bioinformatics in Aquaculture: Principles and Methods**, p. 43–73, 2017. Citado na página 24.

LIU, L.; LI, Y.; LI, S.; HU, N.; HE, Y.; PONG, R.; LIN, D.; LU, L.; LAW, M. Comparison of next-generation sequencing systems. **BioMed Research International**, Hindawi Publishing Corporation, v. 2012, 2012. Citado na página 19.

LOMBARD, V.; RAMULU, H. G.; DRULA, E.; COUTINHO, P. M.; HENRISSAT, B. The carbohydrate-active enzymes database (cazy) in 2013. **Nucleic acids research**, Oxford University Press, v. 42, n. D1, p. D490–D495, 2014. Citado na página 30.

LOWE, T. M.; CHAN, P. P. trnascan-se on-line: integrating search and context for analysis of transfer rna genes. **Nucleic acids research**, Oxford University Press, v. 44, n. W1, p. W54–W57, 2016. Citado na página 46.

LUGLI, G. A.; MILANI, C.; MANCABELLI, L.; SINDEREN, D. van; VENTURA, M. Meganno-tator: a user-friendly pipeline for microbial genomes assembly and annotation. **FEMS microbiology letters**, Oxford University Press, v. 363, n. 7, 2016. Citado na página 43.

MACIEL, F. M. de B. **Python e Django: Desenvolvimento web Moderno e ágil**. [S.l.]: Alta Books, 2020. Citado na página 47.

MARCHLER-BAUER, A.; BO, Y.; HAN, L.; HE, J.; LANCZYCKI, C. J.; LU, S.; CHITSAZ, F.; DERBYSHIRE, M. K.; GEER, R. C.; GONZALES, N. R. *et al.* Cdd/sparcle: functional classification of proteins via subfamily domain architectures. **Nucleic acids research**, Oxford University Press, v. 45, n. D1, p. D200–D203, 2017. Citado na página 29.

MARDIS, E. R. Next-generation dna sequencing methods. **Annu. Rev. Genomics Hum. Genet.**, Annual Reviews, v. 9, p. 387–402, 2008. Citado na página 20.

MARGULIES, M.; EGHOLM, M.; ALTMAN, W. E.; ATTIYA, S.; BADER, J. S.; BEMBEN, L. A.; BERKA, J.; BRAVERMAN, M. S.; CHEN, Y.-J.; CHEN, Z. *et al.* Genome sequencing in microfabricated high-density picolitre reactors. **Nature**, Nature Publishing Group, v. 437, n. 7057, p. 376, 2005. Citado na página 18.

MAXAM, A. M.; GILBERT, W. A new method for sequencing dna. **Proceedings of the National Academy of Sciences**, National Acad Sciences, v. 74, n. 2, p. 560–564, 1977. Citado na página 17.

MCCARTHY, E. M.; MCDONALD, J. F. Ltr_struc: a novel search and identification program for ltr retrotransposons. **Bioinformatics**, Oxford University Press, v. 19, n. 3, p. 362–367, 2003. Citado na página 32.

MCDONNELL, E.; STRASSER, K.; TSANG, A. Manual gene curation and functional annotation. In: **Methods in Molecular Biology**. [S.l.]: Humana Press Inc., 2018. v. 1775, p. 185–208. Citado 5 vezes nas páginas 13, 14, 31, 34 e 35.

MELÉ, A. **Django 3 By Example: Build powerful and reliable Python web applications from scratch**. [S.l.]: Packt Publishing Ltd, 2020. Citado na página 52.

MI, H.; HUANG, X.; MURUGANUJAN, A.; TANG, H.; MILLS, C.; KANG, D.; THOMAS, P. D. Panther version 11: expanded annotation data from gene ontology and reactome pathways, and data analysis tool enhancements. **Nucleic acids research**, Oxford University Press, v. 45, n. D1, p. D183–D189, 2017. Citado na página 29.

MITCHELL, A. L.; ATTWOOD, T. K.; BABBITT, P. C.; BLUM, M.; BORK, P.; BRIDGE, A.; BROWN, S. D.; CHANG, H.-Y.; EL-GEBALI, S.; FRASER, M. I. *et al.* Interpro in 2019: improving coverage, classification and access to protein sequence annotations. **Nucleic acids research**, Oxford University Press, v. 47, n. D1, p. D351–D360, 2018. Citado 3 vezes nas páginas 12, 13 e 29.

MOREIRA, L. Ciências genômicas: fundamentos e aplicações. **Moreira, LM & Varani, AM Plasticidade e fluxo genômico. Ribeirão Preto: Sociedade Brasileira de Genética**, v. 1, p. 101–116, 2015. Citado 2 vezes nas páginas 19 e 21.

MORIYA, Y.; ITOH, M.; OKUDA, S.; YOSHIZAWA, A. C.; KANEHISA, M. Kaas: an automatic genome annotation and pathway reconstruction server. **Nucleic acids research**, Oxford University Press, v. 35, n. suppl_2, p. W182–W185, 2007. Citado na página 43.

NCBI. 2020. Disponível em: <https://www.ncbi.nlm.nih.gov/projects/Sequin/table.html>. Citado na página 39.

NIKOLSKAYA, A. N.; ARIGHI, C. N.; HUANG, H.; BARKER, W. C.; WU, C. H. Pirsf family classification system for protein functional and evolutionary analysis. **Evolutionary Bioinformatics**, SAGE Publications Sage UK: London, England, v. 2, p. 117693430600200033, 2006. Citado na página 29.

OGASAWARA, O.; KODAMA, Y.; MASHIMA, J.; KOSUGE, T.; FUJISAWA, T. Ddbj database updates and computational infrastructure enhancement. **Nucleic acids research**, Oxford University Press, v. 48, n. D1, p. D45–D50, 2020. Citado na página 26.

O'LEARY, N. A.; WRIGHT, M. W.; BRISTER, J. R.; CIUFO, S.; HADDAD, D.; MCVEIGH, R.; RAJPUT, B.; ROBBERTSE, B.; SMITH-WHITE, B.; AKO-ADJEI, D. *et al.* Reference sequence (refseq) database at ncbi: current status, taxonomic expansion, and functional annotation. **Nucleic acids research**, Oxford University Press, v. 44, n. D1, p. D733–D745, 2016. Citado na página 27.

OVERBEEK, R.; BEGLEY, T.; BUTLER, R. M.; CHOUDHURI, J. V.; CHUANG, H.-Y.; COHOON, M.; CRÉCY-LAGARD, V. de; DIAZ, N.; DISZ, T.; EDWARDS, R. *et al.* The subsystems approach to genome annotation and its use in the project to annotate 1000 genomes. **Nucleic acids research**, Oxford University Press, v. 33, n. 17, p. 5691–5702, 2005. Citado na página 41.

PALMER, J.; STAJICH, J. **Funannotate: eukaryotic genome annotation pipeline**. 2020. Disponível em: <https://github.com/nextgenusfs/funannotate>. Citado na página 37.

PANDURANGAN, A. P.; STAHLHACKE, J.; OATES, M. E.; SMITHERS, B.; GOUGH, J. The superfamily 2.0 database: a significant proteome update and a new webserver. **Nucleic acids research**, Oxford University Press, v. 47, n. D1, p. D490–D494, 2019. Citado na página 29.

PAYNE, A.; HOLMES, N.; RAKYAN, V.; LOOSE, M. Bulkvis: a graphical viewer for oxford nanopore bulk fast5 files. **Bioinformatics**, Oxford University Press, v. 35, n. 13, p. 2193–2198, 2019. Citado na página 23.

PEDRUZZI, I.; RIVOIRE, C.; AUCHINCLOSS, A. H.; COUDERT, E.; KELLER, G.; CASTRO, E. D.; BARATIN, D.; CUCHE, B. A.; BOUGUELERET, L.; POUX, S. *et al.* Hamap in 2015: updates to the protein family classification and annotation system. **Nucleic acids research**, Oxford University Press, v. 43, n. D1, p. D1064–D1070, 2015. Citado na página 29.

PERL, P. **The Perl Programming Language**. 2020. Disponível em: <https://www.perl.org>. Citado na página 47.

PERTEA, M.; SALZBERG, S. L. Using glimmerm to find genes in eukaryotic genomes. **Current protocols in bioinformatics**, Wiley Online Library, n. 1, p. 4–4, 2003. Citado na página 42.

PIERLEONI, A.; MARTELLI, P. L.; CASADIO, R. Predgpi: a gpi-anchor predictor. **BMC bioinformatics**, BioMed Central, v. 9, n. 1, p. 392, 2008. Citado na página 36.

POSTGRESQL, P. **PostgreSQL: The world's most advanced open source database**. 2020. Disponível em: <https://www.postgresql.org>. Citado na página 49.

PRIYADHARSHINI, V.; DIVYA, P.; PREETHI, D.; PAZHANIRAJA, N.; PAUL, P. V. A novel web service publishing model based on social spider optimization technique. In: IEEE. **2015 International Conference on Computation of Power, Energy, Information and Communication (ICCPEIC)**. [S.l.], 2015. p. 0373–0387. Citado na página 26.

PYTHON, P. **Welcome to Python.org**. 2020. Disponível em: <https://www.python.org>. Citado na página 46.

R, R. **R: A Language and Environment for Statistical Computing**. 2020. Disponível em: <https://www.R-project.org>. Citado na página 47.

RABBITMQ, R. **Messaging that just works — RabbitMQ**. 2020. Disponível em: <https://www.rabbitmq.com>. Citado na página 48.

REDDY, V. S.; JR, M. H. S. Biov suite–a collection of programs for the study of transport protein evolution. **The FEBS journal**, Wiley Online Library, v. 279, n. 11, p. 2036–2046, 2012. Citado na página 36.

ROTHBERG, J. M.; HINZ, W.; REARICK, T. M.; SCHULTZ, J.; MILESKI, W.; DAVEY, M.; LEAMON, J. H.; JOHNSON, K.; MILGREW, M. J.; EDWARDS, M. *et al.* An integrated semiconductor device enabling non-optical genome sequencing. **Nature**, Nature Publishing Group, v. 475, n. 7356, p. 348–352, 2011. Citado na página 22.

SALZBERG, S. L. Next-generation genome annotation: we still struggle to get it right. **Genome Biology**, v. 20, n. 1, p. 92, dec 2019. ISSN 1474-760X. Disponível em: <https://genomebiology.biomedcentral.com/articles/10.1186/s13059-019-1715-2>. Citado 2 vezes nas páginas 14 e 37.

SAMBROOK, J.; FRITSCH, E. F.; MANIATIS, T. *et al.* **Molecular cloning: a laboratory manual.** [S.l.]: Cold spring harbor laboratory press, 1989. Citado na página 45.

SANGER, F.; NICKLEN, S.; COULSON, A. R. Dna sequencing with chain-terminating inhibitors. **Proceedings of the national academy of sciences**, National Acad Sciences, v. 74, n. 12, p. 5463–5467, 1977. Citado 2 vezes nas páginas 12 e 17.

SCHMIEDER, R.; EDWARDS, R. Quality control and preprocessing of metagenomic datasets. **Bioinformatics**, Oxford University Press, v. 27, n. 6, p. 863–864, 2011. Citado na página 24.

SCHNEIDER, W. D. H.; GONÇALVES, T. A.; UCHIMA, C. A.; COUGER, M. B.; PRADE, R.; SQUINA, F. M.; DILLON, A. J. P.; CAMASSOLA, M. Penicillium echinulatum secretome analysis reveals the fungi potential for degradation of lignocellulosic biomass. **Biotechnology for biofuels**, BioMed Central, v. 9, n. 1, p. 66, 2016. Citado na página 45.

SCHNEIDER, W. D. H.; GONÇALVES, T. A.; UCHIMA, C. A.; REIS, L. dos; FONTANA, R. C.; SQUINA, F. M.; DILLON, A. J. P.; CAMASSOLA, M. Comparison of the production of enzymes to cell wall hydrolysis using different carbon sources by penicillium echinulatum strains and its hydrolysis potential for lignocelullosic biomass. **Process Biochemistry**, Elsevier, v. 66, p. 162–170, 2018. Citado na página 45.

SELZER, P. M.; MARHÖFER, R. J.; KOCH, O. Biological databases. In: **Applied Bioinformatics**. [S.l.]: Springer, 2018. p. 13–34. Citado na página 27.

SEPPEY, M.; MANNI, M.; ZDOBNOV, E. M. Busco: assessing genome assembly and annotation completeness. In: **Gene Prediction**. [S.l.]: Springer, 2019. p. 227–245. Citado na página 25.

SHAIK, N. A.; ELANGO, R.; KHAN, M.; BANAGANAPALLI, B. Introduction to bioinformatics. In: **Essentials of Bioinformatics, Volume I**. [S.l.]: Springer, 2019. p. 1–18. Citado 2 vezes nas páginas 12 e 17.

____. Introduction to biological databases. In: **Essentials of Bioinformatics, Volume I**. [S.l.]: Springer, 2019. p. 19–27. Citado 3 vezes nas páginas 13, 26 e 27.

SHENDURE, J.; BALASUBRAMANIAN, S.; CHURCH, G. M.; GILBERT, W.; ROGERS, J.; SCHLOSS, J. A.; WATERSTON, R. H. Dna sequencing at 40: past, present and future. **Nature**, Nature Publishing Group, v. 550, n. 7676, p. 345, 2017. Citado na página 17.

SHETTY, P. J.; AMIRTHARAJ, F.; SHAIK, N. A. Introduction to nucleic acid sequencing. In: **Essentials of Bioinformatics, Volume I**. [S.l.]: Springer, 2019. p. 97–126. Citado 2 vezes nas páginas 16 e 20.

SIGRIST, C. J.; CASTRO, E. D.; CERUTTI, L.; CUCHE, B. A.; HULO, N.; BRIDGE, A.; BOUGUELERET, L.; XENARIOS, I. New and continuing developments at prosite. **Nucleic acids research**, Oxford University Press, v. 41, n. D1, p. D344–D347, 2012. Citado 2 vezes nas páginas 13 e 28.

SIMPSON, J. T.; WONG, K.; JACKMAN, S. D.; SCHEIN, J. E.; JONES, S. J.; BIROL, I. Abyss: a parallel assembler for short read sequence data. **Genome research**, Cold Spring Harbor Lab, v. 19, n. 6, p. 1117–1123, 2009. Citado na página 25.

SINGH, A.; VINIOTIS, Y. Resource allocation for iot applications in cloud environments. In: IEEE. **2017 International Conference on Computing, Networking and Communications (ICNC)**. [S.l.], 2017. p. 719–723. Citado na página 48.

SINGH, S.; RAO, A.; MISHRA, P.; YADAV, A. K.; MAURYA, R.; KAUR, S.; TANDON, G. Bioinformatics in next-generation genome sequencing. In: **Current trends in Bioinformatics: An Insight**. [S.l.]: Springer, 2018. p. 27–38. Citado na página 18.

SLATKO, B. E.; GARDNER, A. F.; AUSUBEL, F. M. Overview of next-generation sequencing technologies. **Current protocols in molecular biology**, Wiley Online Library, v. 122, n. 1, p. e59, 2018. Citado 2 vezes nas páginas 17 e 19.

SMIT, A.; HUBLEY, R.; GREEN, P. **RepeatMasker Open-4.0**. 2013–2015. <http://www.repeatmasker.org>. Citado 2 vezes nas páginas 31 e 46.

SOLOVYEV, V.; BIOINFORMATICS, D. of. Statistical approaches in eukaryotic gene prediction. **Handbook of statistical genetics**, Wiley Online Library, 2004. Citado na página 33.

SOLOVYEV, V.; KOSAREV, P.; SELEDSOV, I.; VOROBYEV, D. Automatic annotation of eukaryotic genes, pseudogenes and promoters. **Genome biology**, BioMed Central, v. 7, n. 1, p. S10, 2006. Citado 2 vezes nas páginas 33 e 34.

STAJICH, J. E.; BLOCK, D.; BOULEZ, K.; BRENNER, S. E.; CHERVITZ, S. A.; DAGDIGIAN, C.; FUELLEN, G.; GILBERT, J. G.; KORF, I.; LAPP, H. *et al.* The bioperl toolkit: Perl modules for the life sciences. **Genome research**, Cold Spring Harbor Lab, v. 12, n. 10, p. 1611–1618, 2002. Citado na página 39.

STANKE, M.; DIEKHANS, M.; BAERTSCH, R.; HAUSSLER, D. Using native and syntenically mapped cdna alignments to improve de novo gene finding. **Bioinformatics**, Oxford University Press, v. 24, n. 5, p. 637–644, 2008. Citado na página 34.

STANKE, M.; MORGENSTERN, B. Augustus: a web server for gene prediction in eukaryotes that allows user-defined constraints. **Nucleic acids research**, Oxford University Press, v. 33, n. suppl_2, p. W465–W467, 2005. Citado na página 33.

STANKE, M.; TZVETKOVA, A.; MORGENSTERN, B. Augustus at egasp: using est, protein and genomic alignments for improved gene prediction in the human genome. **Genome biology**, BioMed Central, v. 7, n. 1, p. S11, 2006. Citado na página 34.

STEEN, M. van; TANENBAUM, A. S. A brief introduction to distributed systems. **Computing**, Springer, v. 98, n. 10, p. 967–1009, 2016. Citado na página 48.

TANIZAWA, Y.; FUJISAWA, T.; NAKAMURA, Y. Dfast: a flexible prokaryotic genome annotation pipeline for faster genome publication. **Bioinformatics**, Oxford University Press, v. 34, n. 6, p. 1037–1039, 2018. Citado na página 43.

TER-HOVHANNISYAN, V.; LOMSADZE, A.; CHERNOFF, Y. O.; BORODOVSKY, M. Gene prediction in novel fungal genomes using an ab initio algorithm with unsupervised training. **Genome research**, Cold Spring Harbor Lab, v. 18, n. 12, p. 1979–1990, 2008. Citado 2 vezes nas páginas 42 e 46.

TSAI, C.-J.; CHEN, Y.-T.; TSENG, C.-C. An efficient application processor architecture for multicore software video decoding. **IEEE Transactions on Circuits and Systems for Video Technology**, IEEE, v. 25, n. 2, p. 325–338, 2014. Citado na página 28.

VENTER, J. C.; ADAMS, M. D.; MYERS, E. W.; LI, P. W.; MURAL, R. J.; SUTTON, G. G.; SMITH, H. O.; YANDELL, M.; EVANS, C. A.; HOLT, R. A. *et al.* The sequence of the human genome. **science**, American Association for the Advancement of Science, v. 291, n. 5507, p. 1304–1351, 2001. Citado 2 vezes nas páginas 12 e 17.

WANG, R.; BAO, L.; LIU, S.; LIU, Z. Genome annotation: Determination of the coding potential of the genome. **Bioinformatics in Aquaculture: Principles and Methods**, John Wiley & Sons, Ltd Chichester, UK, p. 74–85, 2017. Citado 4 vezes nas páginas 13, 31, 33 e 34.

WATSON, J. D.; CRICK, F. H. *et al.* Molecular structure of nucleic acids. **Nature**, v. 171, n. 4356, p. 737–738, 1953. Citado na página 12.

WU, C. H.; YEH, L.-S. L.; HUANG, H.; ARMINSKI, L.; CASTRO-ALVEAR, J.; CHEN, Y.; HU, Z.; KOURTESIS, P.; LEDLEY, R. S.; SUZEK, B. E. *et al.* The protein information resource. **Nucleic acids research**, Oxford University Press, v. 31, n. 1, p. 345–347, 2003. Citado na página 12.

WU, T. D.; REEDER, J.; LAWRENCE, M.; BECKER, G.; BRAUER, M. J. Gmap and gsnap for genomic sequence alignment: enhancements to speed, accuracy, and functionality. In: **Statistical genomics**. [S.l.]: Springer, 2016. p. 283–334. Citado na página 46.

YANDELL, M.; ENCE, D. A beginner's guide to eukaryotic genome annotation. **Nature Reviews Genetics**, Nature Publishing Group, v. 13, n. 5, p. 329–342, 2012. Citado 2 vezes nas páginas 31 e 33.

YUNRUI, Q. Front-end and back-end separation for warehouse management system. In: IEEE. **2018 11th International Conference on Intelligent Computation Technology and Automation (ICICTA)**. [S.l.], 2018. p. 204–208. Citado 2 vezes nas páginas 46 e 49.

ZERBINO, D. R.; BIRNEY, E. Velvet: algorithms for de novo short read assembly using de bruijn graphs. **Genome research**, Cold Spring Harbor Lab, v. 18, n. 5, p. 821–829, 2008. Citado na página 25.

ZHANG, H.; YOHE, T.; HUANG, L.; ENTWISTLE, S.; WU, P.; YANG, Z.; BUSK, P. K.; XU, Y.; YIN, Y. dbcan2: a meta server for automated carbohydrate-active enzyme annotation. **Nucleic acids research**, Oxford University Press, v. 46, n. W1, p. W95–W101, 2018. Citado na página 30.