

UNIVERSIDADE DE CAXIAS DO SUL

MAURÍCIO TELES FAORO

**UMA PROPOSTA PARA INTEGRAÇÃO DE SERVIÇOS DO PORTAL UCSVIRTUAL
EM DISPOSITIVOS MÓVEIS**

CAXIAS DO SUL

2012

**UNIVERSIDADE DE CAXIAS DO SUL
CENTRO DE COMPUTAÇÃO E TECNOLOGIA DA INFORMAÇÃO
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

MAURÍCIO TELES FAORO

**UMA PROPOSTA PARA INTEGRAÇÃO DE SERVIÇOS DO PORTAL UCSVIRTUAL
EM DISPOSITIVOS MÓVEIS**

Trabalho de Conclusão de
Curso para obtenção do Grau
de Bacharel em Sistemas de
Informação da Universidade de
Caxias do Sul.
Área de concentração: Interface
de Software.
Orientadora Prof. Dra. Elisa Boff

**CAXIAS DO SUL
2012**

AGRADECIMENTOS

Agradeço primeiramente a minha orientadora Prof. Elisa Boff, que desde o início deste trabalho sempre esteve disposta a me auxiliar, sempre corrigindo meus erros e me dando o correto direcionamento para continuidade deste trabalho.

Também quero agradecer a minha família e aos meus amigos que sempre me apoiaram a dar continuidade neste trabalho, entendendo quando eu precisava ficar sozinho para realizar o mesmo e me apoiando e auxiliando com ideias sempre que possível.

Agradeço também a empresa que trabalho, Seventeen Tecnologia da Informação, e a minha coordenadora, Cintia Marchioro, que sempre me liberou quando precisei me ausentar do trabalho para realização de atividades relacionadas a este trabalho, e me apoiou na continuidade.

Enfim deixo um muito obrigado a todos que me apoiaram.

“O único lugar onde o sucesso vem antes do trabalho é no dicionário.”

Albert Einstein

RESUMO

O desenvolvimento de versões móveis de portais surge como uma necessidade, devido à grande utilização de plataformas móveis. Este desenvolvimento deve ser pensado utilizando-se de conceitos de usabilidade, garantindo a melhor experiência de uso possível ao usuário. Para este tipo de desenvolvimento, porém devem ser olhadas características específicas de usabilidade para dispositivos móveis que são diferentes das encontradas em aplicações WEB Convencionais.

Porém apenas o uso de boas práticas de usabilidade não garante que o aplicativo desenvolvido irá agradar aos usuários, tanto na interface, como na facilidade de uso, por isso torna-se necessário o estudo e utilização de técnicas e frameworks de avaliação de interface.

Antes do desenvolvimento de um aplicativo móvel, devem ser avaliadas as tecnologias existentes, pesando os prós e os contras de cada paradigma a ser utilizado, a fim de escolher a que mais se adéqua a necessidade.

Na Universidade de Caxias do Sul existe a necessidade de desenvolvimento de versão móvel do portal UCSVirtual. Neste cenário, este trabalho visa identificar as boas práticas de usabilidade que deverão ser adotadas, as tecnologias mais adequada para este desenvolvimento, bem como as técnicas de avaliação a serem utilizadas.

Palavras-Chave: UCSVirtual, portais, usabilidade, dispositivos móveis, android, iOS, HTML5, CSS3, Interface.

ABSTRACT

The development of mobile versions of websites is a necessity, given the wide use of mobile platforms. This development should be designed using the concepts of usability, ensuring the best user experience possible for the user. For this type of development, but should be looked at specific characteristics of usability for mobile devices that are different from those found in web applications Convention.

But only the use of good practices of usability does not guarantee that the application developed will please the users, both at the interface, such as ease of use, so it becomes necessary to study and use of techniques and frameworks for evaluating interface.

Before the development of a mobile application should be evaluated existing technologies, weighing the pros and cons of each paradigm to be used in order to choose the one that best fits the need.

At the University of Caxias do Sul there is a need to develop mobile version of UCSVirtual. In this scenario, this paper aims to identify best practices in usability that should be adopted, the technologies most suitable for this development, and assessment techniques to be used.

Keywords: UCSVirtual, portals, usability, mobile, android, iOS, HTML5, CSS3, Interface.

LISTA DE ILUSTRAÇÕES

Figura 1 - Relação entre paradigmas e técnicas de avaliação Preece et al. (2005).....	26
Figura 2 - Plataformas para Smartphones	33
Figura 3 - Arquitetura	34
Figura 4 - Arquitetura iOS	36
Figura 5 - Diagrama de Casos de Uso	51
Figura 6 - Interface Tela Login	52
Figura 7 – Erro Login	53
Figura 8 - Interface principal do sistema.....	55
Figura 9 - Acesso ao Menu	56
Figura 10 - Mural (Seleção de Disciplina)	57
Figura 11 - Mural	58
Figura 12 - Visualização de Recado no Mural	59
Figura 13 - Cronograma (Seleção de Disciplina)	61
Figura 14 - Acesso ao Cronograma.....	62
Figura 15 - Exibição do Cronograma	63
Figura 16 - Orientações da Aula	64
Figura 17 – Diagrama de Classes.....	65
Figura 18 – Modelo ER.....	66
Figura 19 – Arquitetura do Protótipo	68
Figura 20 – <i>Model</i> Aplicativo UCSVirtual Móvel.....	69
Figura 21 – Trecho código fonte <i>View</i>	69
Figura 22 – Código fonte Python no <i>Template</i> (HTML5)	70

LISTA DE TABELAS

Tabela 1 – Comparativo entre aplicativos WEB e Nativos	31
Tabela 2 - Compatibilidade <i>tags</i> HTML5	41
Tabela 3 - Funcionalidades mais utilizadas no UCSVirtual	48
Tabela 4 - Funcionalidades sugeridas para versão móvel do UCSVirtual	49
Tabela 5 –UC01 – Realizar Login	51
Tabela 6 - UC02 – Acessar Interface Principal	53
Tabela 7 – UC03 – Acessar Mural	56
Tabela 8 – UC04 – Acessar Cronograma	59

LISTA DE ABREVIATURAS

AVA	Ambiente Virtual de Aprendizagem
APP	Application

LISTA DE SIGLAS

API	Application Programming Interface
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
W3C	World Wide Web Consortium
WEB	World Wide Web
SDK	Software Development Kit
UCS	Universidade de Caxias do Sul

SUMÁRIO

1 INTRODUÇÃO	12
2 - PROJETO DE INTERFACES PARA DISPOSITIVOS MÓVEIS	14
2.1 CONCEITOS DE DESIGN DE INTERAÇÃO	14
2.2 PRINCÍPIOS DE USABILIDADE PARA DESENVOLVIMENTO MÓVEL.....	17
2.3 AVALIAÇÃO DE INTERFACE.....	22
2.3.1 Paradigmas de Avaliação de Interface.....	22
2.3.2 Técnicas para Avaliação de Interface	23
2.3.3 Framework para Avaliação de Interface	26
3 – TECNOLOGIAS	29
3.1 APLICATIVOS WEB X APLICATIVOS NATIVOS.....	29
3.2 PRINCIPAIS TECNOLOGIAS PARA DESENVOLVIMENTO DE APLICATIVOS NATIVOS	32
3.2.1 Desenvolvimento de aplicativos nativos para Android	33
3.2.2 Desenvolvimento de Aplicativos nativos para IOS.....	35
3.3 PRINCIPAIS TECNOLOGIAS PARA DESENVOLVIMENTO DE APLICATIVOS WEB	37
3.3.1 HTML5.....	37
3.3.2 CSS3	42
3.4 CONCLUSÃO DO CAPÍTULO.....	42
4 - PORTAL UCSVIRTUAL.....	45
5 - PROPOSTA DE INTERFACE PARA VERSÃO MÓVEL PARA O PORTAL UCSVIRTUAL.....	48
5.1 ANÁLISE E DESENVOLVIMENTO DO PROTÓTIPO	50
5.2 ARQUITETURA DO PROTÓTIPO	67
6 – AVALIAÇÃO DA INTERFACE DESENVOLVIDA	71
7 - CONSIDERAÇÕES FINAIS.....	76
REFERENCIAS BIBLIOGRÁFICA	78

1 INTRODUÇÃO

Atualmente vivemos em um mundo onde, cada dia mais, as pessoas estão conectadas através *tablets* e *smartphones*. Mobilidade da informação passou de luxo para necessidade na vida das pessoas e isto está fazendo com que o mundo se volte cada vez mais para o desenvolvimento de sites e aplicações que possibilitem acesso em dispositivos móveis.

Esta nova realidade está fazendo com o mundo se volte para tecnologias de acesso online, que permitam que as pessoas utilizem serviços web onde estiverem. É muito mais prático o acesso a informações através de *tablets* e *smartphones* do que se deslocar até o computador pessoal.

Ao mesmo passo que esta nova realidade está revolucionando a vida das pessoas, no sentido de mobilidade e acesso as informações, também está causando uma grande “dor de cabeça” para empresas que desejam disponibilizar serviços para dispositivos móveis. O problema do desenvolvimento móvel, em grande parte, se deve a grande variedade de dispositivos disponíveis no mercado, com diferentes resoluções de telas e sistemas operacionais, onde o desenvolvimento de uma solução genérica com bom nível de usabilidade se torna complexo.

A crescente demanda por acesso móvel, por parte dos usuários, também está presente na Universidade de Caxias do Sul, onde a mesma disponibiliza um portal web para que seus alunos, docentes e funcionários tenham acesso a informações e serviços acadêmicos. Este portal é denominado “UCSVirtual” e possui funcionalidades que vão do Ambiente Virtual de Aprendizagem (AVA) até o gerenciamento de matrículas.

Hoje em dia, o UCSVirtual é um portal estabilizado possuindo uma grande quantidade de acessos, sendo uma ferramenta indispensável na vida de quem estuda ou trabalha na Universidade de Caxias do Sul. O acesso a este portal é contemplado através de computadores com acesso a Internet, sendo o mesmo disponível a praticamente todos os *browsers*¹ presentes no mercado. Em alguns *tablets* e *smartphones* é possível acessar o portal também, porém o mesmo não possui interface adequada à resolução destes dispositivos e muitas de suas funcionalidades são parcialmente contempladas, desta maneira, causando transtornos para quem deseja acessar através de um dispositivo móvel.

¹ Programa para navegação na Internet

Para desenvolvimento de versão móvel do UCSVirtual a Universidade de Caxias do Sul, necessita identificar qual a tecnologia que melhor se adéqua a necessidade do portal. Mais do que definir a(s) linguagem(ns) de programação que será(ão) utilizada(s) é necessário definir o paradigma de desenvolvimento móvel. Segundo Fred Cavazza (Forbes, 2011) existem quatro possibilidades para o desenvolvimento de aplicativos móveis, que são: aplicações nativas, aplicações híbridas, aplicações WEB dedicadas e aplicações WEB.

Também é muito importante a análise da interface que será utilizada, segundo o guia de desenvolvimento Android (Android Developer, 2012), o design da interface é parte fundamental para tornar a aplicação atraente ao usuário e aumentar seu nível de aceitação. O Guia de desenvolvimento Android ressalta o fato de existência de vários *smartphones* e *tablets* com diferentes tamanhos de tela, o que, faz que o desenvolvimento de aplicações móveis deva tentar se adequar aos mais diferentes *hardwares* que acessaram a aplicação.

No caso da Universidade de Caxias do Sul, é necessário além da definição de interface e tecnologia, quais funcionalidades deverão estar disponibilizadas na versão móvel do portal UCSVirtual, pois o mesmo possui muitas funcionalidades, e deve se analisar quais realmente serão úteis aos seus usuários na versão móvel, além de analisar a viabilidade das mesmas.

O objetivo deste trabalho é definir a tecnologia e o modelo de interface adequados a uma versão móvel do UCSVirtual, além de mapear quais funcionalidades devem ser priorizadas neste desenvolvimento e desenvolver um protótipo da versão móvel do portal. Neste contexto, este trabalho vai buscar responder a seguinte questão de pesquisa:

“É possível implementar uma interface para acesso a alguns serviços do UCSVirtual que possa ser acessada em dispositivos móveis com diferentes plataformas?”

Para responder a esta questão, este trabalho irá abordar fundamentos de projeto de interfaces para dispositivos móveis (Capítulo 2), tecnologias para desenvolvimento de aplicações móveis (Capítulo 3), características e funcionalidades do UCSVirtual e também identificar os serviços mais utilizados pelos alunos a fim de propor uma interface para dispositivos móveis que faça acesso aos mesmos (Capítulo 4), desenvolvimento de um protótipo para a versão móvel do portal UCSVirtual (Capítulo 5), avaliação da interface do protótipo desenvolvido (Capítulo 6).

2 - PROJETO DE INTERFACES PARA DISPOSITIVOS MÓVEIS

A interface para dispositivos móveis é um problema que não é recente, o mesmo existia antes dos *smartphones* e *tablets*, como pode ser notado na obra de BREWSTER (1999), que reporta a popularização dos dispositivos móveis e a complexidade de desenvolvimento de aplicações para os mesmos devido ao reduzido tamanho da interface.

Com o passar do tempo surgiram e se popularizaram os *smartphones* e *tablets*, onde o problema inicial de tela reduzida foi em partes minimizado, pois houve um considerável crescimento no tamanho de telas destes dispositivos. Porém esta evolução trouxe outro grande problema da não padronização de tamanho de telas, hoje é possível encontrar *smartphones/tablets* que variam de duas a dez polegadas. Além da grande variação de tela a evolução destes dispositivos, os tornou cada vez mais potentes, podendo em algumas situações substituírem computadores e notebooks, devido a sua praticidade. Nielsen (2011) reporta que estamos no “ano” dos dispositivos móveis onde as empresas devem re-escrever seus sites e portais para versões móveis, porque a versão existente atualmente, provavelmente está muito abaixo das expectativas dos usuários.

2.1 CONCEITOS DE DESIGN DE INTERAÇÃO

Segundo Preece et al. (2005) a preocupação do design de interação é projetar produtos interativos que sejam de fácil aprendizagem, eficazes no uso e capazes de proporcionar ao usuário uma experiência gratificante. Para projetar produtos interativos e usáveis é necessário levar em consideração diversos fatores como qual usuário irá utilizá-lo, em qual local e quais momentos será utilizado. Segundo as autoras Preece et al. (2005) pontos que devem ser observados pelo designer são:

- a. Considerar no que seus usuários são bons ou não;
- b. Considerar quais alternativas poderão ser colocadas a fim de prestar auxílio aos usuários diante de atividades que eles ainda não conhecem;
- c. Pensar no que pode proporcionar experiências de qualidade aos usuários;
- d. Ouvir os usuários e envolvê-los no design;

- e. Utilizar técnicas baseadas no usuário que forem testadas e aprovadas durante o desenvolvimento do produto.

Também é importante envolver conhecimentos multidisciplinares no momento de desenvolvimento de um produto, pois quanto, mais perto da realidade do usuário, mais fácil de o produto atender as necessidades e satisfazer o mesmo.

Conforme Preece et al. (2005) o processo de design de interação possui quatro atividades básicas que devem ser realizadas e repetidas continuamente:

- a. Identificação das necessidades do produto avaliado e apresentação de propostas de requisitos para correção do problema.
- b. Desenvolvimento de designs alternativos ao pré-existente, capazes de preencher os requisitos propostos no item anterior.
- c. Construção de versões interativas dos designs propostos inicialmente, de maneira a serem analisados posteriormente.
- d. Avaliação do que está sendo construído, no decorrer do processo de desenvolvimento.

O processo de design de interação é realizado de forma a tentar atingir as seguintes metas:

1 - **Metas de Usabilidade** - As metas de usabilidade podem ser divididas conforme abaixo:

- a. Eficácia: meta que busca saber o quanto um sistema é bom ao fazer o que se espera dele.
- b. Eficiência: Refere-se à maneira que o sistema auxilia o usuário a realizar suas atividades
- c. Segurança: implica proteger o usuário de condições perigosas e situações indesejáveis, referindo-se às condições externas.
- d. Utilidade: O sistema deve fornecer ao usuário, o conjunto de funções necessário para realização da tarefa esperada pelo mesmo.
- e. Capacidade de Aprendizagem: refere-se à facilidade de aprendizagem de utilização do sistema por parte do usuário.
- f. Capacidade de memorização: refere-se à facilidade de se lembrar como se utiliza o sistema, mesmo que o mesmo não seja utilizado com frequência.

2 - Metas decorrentes da experiência do usuário - São metas que se preocupam com a criação de sistemas que sejam satisfatórios, agradáveis, divertidos, interessantes, úteis, motivadores, esteticamente apreciáveis, incentivadores de criatividade, compensadores e emocionalmente adequados. O objetivo desse cuidado está na experiência que os produtos passarão aos usuários e em como ele se sentirá no decorrer da interação.

Conforme Preece et al. (2005) os princípios de Design quando usados na prática são chamados de Heurística, e este conceito também se aplica aos princípios de usabilidade. Estes segundo Nielsen (2001) podem ser listados em dez itens conforme abaixo:

1. Visibilidade do Status do Sistema: O sistema mantém os usuários sempre informados do que está acontecendo, fornecendo um *feedback* adequado dentro de um tempo razoável.
2. Compatibilidade do sistema com o mundo real: A linguagem utilizada no sistema é de fácil compreensão para o usuário. Não devem ser utilizados termos orientados ao sistema.
3. Controle do usuário e liberdade: Fornecem alternativas que permitem ao usuário sair de funções executadas inesperadamente, “saídas de emergência” e são claramente identificadas.
4. Consistência e padrões: Evita fazer com que os usuários tenham que pensar se palavras, situações ou ações diferentes significam a mesma coisa.
5. Ajudar os usuários a reconhecer, diagnosticar e recuperar-se de erros: Utiliza linguagem simples para descrever a natureza do problema e sugere uma maneira de resolvê-lo.
6. Prevenção de erros: Sempre quando possível, impede a ocorrência de erros.
7. Reconhecimento em vez de memorização: Tornar objetos, ações e opções visíveis.
8. Flexibilidade e eficiência de uso: Fornece aceleradores invisíveis aos usuários inexperientes, os quais, no entanto, permitem aos mais experientes realizar tarefas com mais rapidez.
9. Estética e design minimalista: Evita o uso de informações irrelevantes ou raramente necessárias.
10. Ajuda e documentação: Fornece informações que podem ser facilmente encontradas e ajuda mediante uma série de passos concretos que podem ser facilmente seguidos.

Pode-se concluir através dos conceitos básicos de design de interação que a aplicação deve ser projetada sempre visando a usabilidade e funcionalidade para o usuário final da mesma, desta forma garantindo uma maior aceitação e nível de satisfação no produto desenvolvido. São

ênfatisados itens básicos que devem ser observados para atingir o objetivo principal, que é atender as necessidades do usuário da forma mais simples e usual possível.

2.2 PRINCIPIOS DE USABILIDADE PARA DESENVOLVIMENTO MÓVEL

Segundo o guia de desenvolvimento Android (Android Developer, 2012) a interface de uma aplicação móvel deve ser atraente e amigável ao usuário, onde que, além de se adequar aos vários tamanhos de tela disponíveis deve seguir alguns princípios, que deverão ter por objetivo: encantar o usuário, simplificar a vida do usuário e tornar a experiência de uso surpreendente. Entre estes princípios podemos citar:

- a. A interface deve ser bonita, animações ou efeitos sonoros sutis são bem vindos.
- b. Objetos reais são mais cognitivos do que menus e botões clássicos. Tornar mais cognitivo acesso as funcionalidades facilita o uso da aplicação.
- c. Deixar o usuário personalizar a aplicação faz com que o mesmo se sinta mais confortável a utilizar a mesma e tenha uma melhor experiência de uso.
- d. Fazer com que aplicação tenha certo nível de inteligência, fazendo com que funcionalidades usadas muito frequentemente pelo usuário tenham um acesso facilitado, reduzindo a redundância de informações digitadas pelo usuário.
- e. Não se deve utilizar textos e frase longas. Pessoas preferem ler sentenças curtas.
- f. Sempre quando possível deve se utilizar figuras ao invés de sentenças.
- g. A aplicação deve ser “inteligente” para decidir pelo usuário, porém também deve deixar o usuário decidir por si mesmo. Sempre deve ser possível o usuário voltar atrás em suas decisões.
- h. A aplicação somente deve mostrar o que usuário deseja ver. Deve-se tomar cuidado com informações desnecessárias.
- i. O usuário sempre deve saber se localizar na aplicação. Deve ser intuitivo a transição entre funcionalidades e quando houver processo deve haver um feedback do mesmo para o usuário não se sentir perdido.
- j. A aplicação deve sempre salvar as configurações e preferências do usuário independente do dispositivo móvel utilizado.

- k. Evite funcionalidades com ícones/botões/objetos similares com funções distintas, pode confundir o usuário.
- l. Somente interrompa a funcionalidade que o usuário está executando se for muito importante. Pessoas não gostam de ser interrompidas.
- m. Utilize funcionalidades de acesso similares a de outras aplicações. A experiência de uso do usuário com outras aplicações com padrão parecido favorecerá a utilização de sua aplicação.
- n. Evite que ocorram exceções técnicas em sua aplicação.
- o. Quebre tarefas complexas em tarefas menores, sempre dando *feedback* aos usuários.
- p. Simplifique ao máximo possível o trabalho do usuário.
- q. Torne as funcionalidades mais importantes, de acesso e execução mais rápidas.

No desenvolvimento de versões móveis de sites e portais o guia de desenvolvimento Android (Android Developer, 2012), também cita algumas boas práticas que devem ser observadas, a fim de garantir a portabilidade e usabilidade da aplicação WEB:

- a. Redirecionar dispositivos móveis para a versão móvel do site: Sempre quando acessado o site através de um dispositivo móvel, o servidor deve redirecionar a versão móvel do mesmo, porém deve ser possível ao usuário escolher a versão desktop do mesmo caso deseje.
- b. Utilize linguagem de marcação adequada para dispositivos móveis: a linguagem de marcação escolhida para o desenvolvimento deve ser compatível com os navegadores disponíveis para dispositivos móveis. Frames HTML e tabelas aninhadas devem ser evitados.
- c. Use apropriadamente os metadados de redimensionamento do site: O tamanho do site deve ser igual ao tamanho da tela do dispositivo móvel, faça com que sua aplicação se redimensione conforme o dispositivo utilizado.
- d. Evitar solicitações de dados excessivos: Como os dispositivos móveis normalmente têm uma velocidade de conexão muito mais lento do que um computador desktop, você deve fazer seu *website* carregar as páginas mais rápido possível. Uma maneira de acelerá-lo é evitar o carregamento de arquivos

adicionais, tais como folhas de estilo (CSS - *Cascading Style Sheets*) e arquivos de script no cabeçalho. Em vez disso, fornecer o seu CSS e JavaScript diretamente no cabeçalho (ou no final do corpo, para os scripts que você não precisa até que a página é carregada). Alternativamente, você deve otimizar o tamanho e a velocidade de seus arquivos comprimindo-os.

- e. Use um layout vertical linear: Evitar a necessidade de o usuário vá para a esquerda e direita, enquanto navegar pela sua página web. Rolagem para cima e para baixo é mais fácil para o usuário e torna a sua página web simples.

Segundo Nielsen Norman Group (2011) no desenvolvimento de *websites* para dispositivos móveis devem ser observados os itens abaixo listados, a fim de garantir um maior nível de satisfação do usuário:

- a) Comportamento de usuários de dispositivos móveis: O comportamento que o usuário possui acessando uma página web através de um dispositivo móvel é diferente do que quando utilizado via um “desktop”. Deve ser levado em consideração este item para o desenvolvimento.
- b) Considerações na estratégia de *design*: No desenvolvimento do *website* deve-se avaliar muito bem a estratégia de *design* a ser utilizada - criar um *website* dedicado a dispositivos móveis ou manter o acesso ao *website* regular para estes dispositivos, desenvolver um *website* ou uma aplicação nativa.
- c) Princípios básicos de usabilidade para dispositivos móveis: Devem ser desenvolvidas as funcionalidades básicas do sistema focando no desenvolvimento de dispositivos móveis. Funcionalidades como *logar-se* no sistema, estrutura de buscas e navegação devem possuir tratamento diferente do existente para desktop.

No que se refere à *tablets* o Nielsen Norman Group (2011) destaca que apesar do tamanho de tela destes dispositivos geralmente serem grandes o que faz com que os *websites* desenvolvidos para *desktop* possuam uma boa visualização, os usuários possuem dificuldades no momento de interação com este, por causa do tamanho dos botões e dos links, geralmente pequenos, causando assim dificuldade de acesso à informação correta através do toque.

Apesar dos *websites* convencionais possuírem a dificuldade acima citada, não é possível afirmar que o desenvolvimento de uma versão móvel irá solucionar todos os problemas. Em alguns casos usuários relatam preferir a versão *desktop* acessada no *tablet* do que a versão móvel. Entre os motivos que podemos citar tal preferência estão: os aplicativos às vezes contêm menos informações do que no site; o design do aplicativo é confuso ou faz o usuário trabalhar mais.

Ainda segundo Nielsen Norman Group (2011) para um aplicativo desenvolvido para *tablets* ter sucesso, o mesmo deve considerar os itens abaixo:

- a) Ser desenhado visando usuários constantes: O aplicativo deve ser desenvolvido focando os usuários que mais utilizam a versão normal do *website*. Dessa forma procurando suprir as necessidades encontradas pelos mesmos no acesso via *tablet*.
- b) O aplicativo deve possuir diferenciais em relação ao website convencional: O aplicativo deve possuir diferenciais, podem ser funcionalidades extras ou simplificar acesso a principais funcionalidades, dessa forma aumentando o nível de usabilidade.
- c) Os usuários não podem ter mais trabalho para executar tarefas no aplicativo do que no website convencional: O aplicativo não pode ser mais trabalhoso, isto causa baixa adesão por parte dos usuários.
- d) Os aplicativos para *tablets* não podem ser desenvolvidos da mesma forma que para *smartphones*: Além da diferença do tamanho de tela entre um *tablet* e um *smartphone*, existe também a diferença de uso, onde que o *tablet* acaba sendo um híbrido entre dispositivo móvel e dispositivo convencional.

Quando não existe a possibilidade de desenvolvimento de um *website* ou um aplicativo totalmente dedicado para *tablets*, algumas boas práticas podem ser adotadas:

- a) Evitar a utilização de conteúdo em *Flash*: A maioria dos *tablets* e *smartphones* disponíveis no mercado não conseguem executar aplicativos em Flash ou apresentam problemas para tanto.
- b) Criação de ícones (alvos) maiores para facilitar o acesso através do toque: É importante que o ícone possua um tamanho que o torne de fácil toque, ou seja, que o usuário não apresente dificuldade para acessar a função desejada.
- c) Espaçar os links sempre que possível: Deve existir espaçamento entre os links, a fim de garantir que o usuário não acesse o link errado.

d) Detectar a localização dos links: Os links devem ser detectáveis pelo usuário, este deve saber que ao tocar no mesmo algo ocorrerá, deverá ser implícito.

e) Minimizar a necessidade de digitação: Usuários de *tablets* e *smartphones* não gostam de digitar muitas informações. O site deve solicitar apenas às informações que realmente são necessárias e para tanto podemos seguir algumas dicas:

- Compute as informações para os usuários.
- Seja tolerante e ofereça correções. Não exija que o usuário digite a informação completa.
- Salve o histórico e permite que o usuário acesse as informações do histórico, a fim de facilitar a digitação.
- Use padrões para fazer sentido para o usuário.
- Se o aplicativo não armazena nenhuma informação confidencial (ex: cartão de crédito), não há necessidade de o usuário ser *deslogado*.

f) Agrupar controles e informações relacionadas: As informações que se referem ao mesmo assunto devem estar sempre agrupadas, facilitando a localização do usuário.

No desenvolvimento de aplicações ou *websites* para *tablets* também deve ser pensado no comportamento do mesmo quando a orientação é trocada (retrato para paisagem). Segundo Nielsen Norman Group (2011) no que se refere à orientação de tela, algumas recomendações são cabíveis:

- a) Mantenha o mesmo conteúdo disponível em ambas as orientações, em ambos os níveis do artigo e da página. Para tornar o conteúdo consistente ao nível da página procure por pontos naturais de quebra e os utilize em ambas as orientações.
- b) Mantenha o usuário na mesma localização quando este mudar a orientação.
- c) Se alguma funcionalidade é disponível em apenas uma orientação, avise o usuário desta particularidade.

Quando o aplicativo ou *website* possuir a possibilidade de realização de download de conteúdo, deve ser pensado na existência de barra de progresso a fim de manter o usuário informado do que está ocorrendo.

A existência de dicas ou opção de ajuda é sempre bem-vinda. Porém quando as mesmas são utilizadas devem ser tomados alguns cuidados:

- a) As instruções devem ser claras e simples.

b) Foque em uma funcionalidade por vez. Apresente somente as instruções que são necessárias para o usuário iniciar o uso.

Portanto podemos observar a existência de princípios básicos quando pensando em desenvolvimento móvel. Existindo princípios diferentes do desenvolvimento web ou desktop (embora boa parte dos conceitos de usabilidade tradicionais possam ser aplicados a dispositivos móveis também), onde questões como acesso via toque, mudança de orientação da tela devem ser observados.

2.3 AVALIAÇÃO DE INTERFACE

Segundo Preece et al. (2005) os usuários tem preferência por sistemas fáceis de apreender e utilizar assim como eficazes, eficientes, seguros e satisfatórios. Para garantir que seu sistema atinja o nível de usabilidade e design desejado pelo usuário é necessário além de seguir os princípios de design e usabilidade, realizar a avaliação da interface a fim de garantir que seu produto realmente atende o usuário.

Para Tognazzini (apud Preece et al. (2005)) existem cinco razões para investir em testes com o usuário, são elas:

1. Os problemas são consertados antes de o produto ser lançado, não depois.
2. A equipe pode se concentrar em problemas reais, não imaginários.
3. Os engenheiros codificam, em vez de debater.
4. O tempo para que o produto entre no mercado é menor.
5. Finalmente, após a entrega da primeira versão, apresentará um design sólido.

Pode-se perceber que a avaliação de interface tem por objetivo principal prevenir e solucionar os problemas antes de a aplicação chegar ao usuário final e desta forma evitar problemas e garantir o maior nível de satisfação possível com a aplicação, além de otimizar o tempo de desenvolvimento da mesma, gastando menos tempo com manutenções e ajustes.

2.3.1 Paradigmas de Avaliação de Interface

Conforme Preece et al. (2005) para realizarmos uma avaliação de interface é necessário estudarmos os paradigmas de avaliação que podemos definir da seguinte maneira:

a) Avaliação “rápida e rasteira” : É uma abordagem rápida e informal realizada pelo designer ou programadores junto ao usuário final para validar uma idéia ou verificar se a interface de um ícone ou site está de acordo com o que o usuário espera. Este paradigma geralmente é utilizado em projetos que possuem pouco prazo e necessitam de uma avaliação muito rápida.

b) Testes de Usabilidade: Surgiu na década de 80 onde foi muito utilizado, consiste em realização de testes com usuários, onde estes são submetidos a testes específicos que retratam as tarefas que serão realizadas pelos usuários finais do sistema. Neste paradigma os usuários que realizam o teste são “confinados”, ficando isolados de comunicação com outras pessoas, podendo ser filmados e terão seu desempenho avaliado para realização das tarefas. Também são aplicados questionários a estes para medir o grau de satisfação.

c) Estudos de campo: Os estudos de campo tem como uma de suas principais características a realização em ambientes reais com o objetivo de aumentar o entendimento sobre o que os usuários fazem naturalmente e como a tecnologia causa impacto nestas atividades. Nestes paradigmas de avaliação pode-se aproveitar uma grande variedade de elementos, tudo que o usuário realiza pode ser útil. No estudo de campo o avaliador pode ser uma pessoa que fica observando tudo que está ocorrendo no ambiente ou ainda alguém que está participando do teste.

d) Avaliação Preditiva: Os especialistas aplicam seu conhecimento a respeito de usuários típicos, geralmente guiados por heurísticas, visando prever problemas de usabilidade. Em uma outra abordagem podem envolver modelos teóricos. Uma das principais características deste paradigma é a não necessidade de usuários para realização dos testes. Para auxiliar nestes testes existe uma série de heurísticas, porém para WEB e dispositivos móveis a necessidade de criação de heurísticas específicas.

Neste trabalho será focada a utilização da avaliação preditiva, por questões econômicas e tempo de desenvolvimento.

2.3.2 Técnicas para Avaliação de Interface

Segundo Preece et al. (2005), os paradigmas de avaliação podem ser utilizados na prática através de cinco técnicas de avaliação, a saber:

- a) Observar usuários: Nesta técnica podem ser utilizadas anotações, áudio, vídeo e logs de interação objetivando a identificar necessidades. Nesta técnica os grandes desafios do avaliador são como realizar a observação sem atrapalhar os usuários e como fazer a análise dos dados.
- b) Solicitar as opiniões dos usuários: Esta técnica pode ser realizada com questionários ou entrevistas e visa entender a opinião do usuário sobre o produto, ter um *feedback*. Pode ser aplicada a centenas de usuários ou a apenas alguns.
- c) Solicitar as opiniões dos especialistas: Baseado em heurísticas especialistas realizam atividades típicas de usuários para identificarem problemas. Os programadores costumam apreciar esta técnica por ela ser relativamente barata e rápida.
- d) Testes com usuários: A base fundamental de testes de usabilidade tem sido medir o desempenho dos usuários em comparar dois ou mais designs. De forma geral são realizados em ambientes controlados, verificando a navegação realizada pelos usuários e quantia de erros encontrados durante o processo. São realizadas estatísticas com estes dados.
- e) Modelagem do desempenho das tarefas realizadas por usuários: Ocorreram muitas tentativas de se modelar a interação humano-computador de maneira que fosse possível prever a eficiência e os problemas associados aos diferentes tipos de design, em um estágio preliminar. Esta técnica funciona bem para sistemas limitados.

As técnicas de avaliação de interface podem ser associados aos paradigmas de avaliação de interface como podemos verificar na Figura 1.

Paradigmas de Avaliação				
Técnicas	"Rápida e Suja"	Testes de Usabilidade	Estudos de Campo	Preditiva

Observar os Usuários	Importante para observar como os usuários se comportam em seus ambientes naturais	Logs de vídeo e interação que podem ser analisados para identificar erros, identificar rotas no uso do software ou calcular o tempo de desempenho	A observação constitui uma parte central de qualquer estudo de campo. Em estudos etnográficos, os avaliadores ficam imersos no ambiente. Em outros tipos de estudo, o avaliador observa objetivamente.	N/A
Perguntar aos Usuários	Discussões individuais com usuários e usuários em grupos ou grupo de foco.	Os questionários de satisfação dos usuários são administrados para coletar as opiniões deles.	O avaliador pode entrevistar ou discutir o que vê com os participantes. AS entrevistas etnográficas são utilizadas em estudos etnográficos	N/A
Perguntar aos Especialistas	Fornecer críticas (denominadas "relatórios críticos") da usabilidade de um protótipo.	N/A	N/A	Os especialistas utilizam heurísticas logo no início do design, para prever a eficácia de uma interface.
Testes com o usuário.	N/A	Testar usuários na realização de tarefas típicas em um ambiente de laboratório controlado. Constitui o pilar dos testes de usabilidade.	N/A	N/A

Modelar o desempenho do usuário na realização de tarefas	N/A	N/A	N/A	Os modelos são utilizados para prever a eficácia de uma interface ou comparar tempos de desempenho em versões diferentes.
--	-----	-----	-----	---

Figura 1 - Relação entre paradigmas e técnicas de avaliação Preece et al. (2005)

2.3.3 Framework para Avaliação de Interface

Segundo Preece et al. (2005) avaliações bem planejadas são dirigidas por metas claras e perguntas adequadas. Para guiar uma avaliação de interface pode ser utilizado framework DECIDE que pode ser definido conforme abaixo:

- a) Determinar as metas que a avaliação irá abordar: Devem ser definidas quais são as metas de alto nível da avaliação, pois elas devem guiar a avaliação. Neste item devem ser observados os critérios abaixo:
 - Certificar-se de que os avaliadores entenderam as necessidades dos usuários.
 - Identifique a metáfora em que o *design* será baseado.
 - Assegure-se que a interface final é consistente.
 - Investigue até que ponto a tecnologia influencia as práticas de trabalho.
 - Identifique como a interface de um produto existente poderia ter sua usabilidade melhorada.
- b) Explorar as questões específicas a serem respondidas: Para que as metas sejam operacionais, devem-se identificar as questões cujas respostas satisfaçam a elas. As questões podem ser divididas em sub-questões muito específicas, de modo a tornar a avaliação ainda mais exclusiva. As sub-questões podem, por sua vez, ser posteriormente decompostas em questões ainda mais aprimoradas e assim por diante.

c) Escolher o paradigma de avaliação e a técnica de resposta para as perguntas: Após a identificação das metas e questões principais, deve-se definir o paradigma e as técnicas que serão utilizadas. Devem ser observadas as questões práticas e éticas para avaliar a viabilidade do paradigma e técnicas a serem utilizados.

d) Identificar questões práticas que devem ser abordadas, como a seleção dos participantes: Deve-se definir usuários apropriados para a avaliação e de qual maneira estes estarão envolvidos com o processo, os equipamentos que serão utilizados na avaliação, as restrições orçamentárias e de cronograma e o conhecimento especializado disponível na equipe de avaliação.

e) Decidir como lidar com as questões éticas: Muitas organizações profissionais definem códigos éticos e esperam que seus membros os sigam. Portanto é interessante seguir as seguintes recomendações para assegurar que as avaliações são realizadas eticamente:

- Informe os participantes do objetivos do estudo e exatamente o que eles devem esperar se participarem.
- Certifique-se de ter explicado que são confidenciais as informações demográficas, financeiras, de saúde e outras questões delicadas que os usuários fornecem ou que fiquem evidentes a partir dos testes.
- Certifique-se que os usuários saibam que estão livres para deixarem a avaliação quando não o mais quiserem participar.
- Quando possível, pague os usuários que participarem.
- Evite incluir citações ou descrições que revelem inadvertidamente a identidade das pessoas.
- Peça permissão aos usuários com antecedência para citá-los.

f) Avaliar, interpretar e apresentar os dados: É necessário tomar decisões de quais dados coletar, como analisá-los e como apresentar estes dados a equipe de desenvolvimento. Para tanto deve ser analisada e exposta à confiabilidade, validade, desvios, escopo e validade ecológica dos dados.

No desenvolvimento da versão móvel do portal UCSVirtual, utilizaremos neste trabalho o paradigma de *Avaliação Preditiva*, visto que, a mesma não necessita de realização de testes com o usuário final da aplicação. Nosso projeto visa desenvolver um protótipo da aplicação e em seu término não teremos a integração completa com o portal UCSVirtual, o que poderia, se realizado

testes com usuários, produzir um resultado de avaliação diferente do esperado, que é a avaliação das funcionalidades que serão desenvolvidas e não do aplicativo como um todo. Desta forma, será realizada a avaliação junto a especialistas.

Entendemos que a versão móvel do Portal UCSVirtual deverá respeitar as boas práticas de usabilidade para aplicações móveis e também as convencionais aplicadas aos demais tipos de aplicações. O resultado deste desenvolvimento deverá ser uma aplicação intuitiva, fácil de utilizar, com interface amigável e limpa, contendo um tratamento de erros adequados, possuindo mensagens e conceitos claros para os usuários, conseguindo ser facilmente utilizada através do toque (*touch*), facilitando as operações de maior uso, possuindo opções fáceis e simples para retorno de funcionalidades indesejadas e lidando de forma adequada com os diferentes tipos de resolução possíveis de serem encontrados.

3 – TECNOLOGIAS

Segundo Cavazza (Forbes, 2011) no desenvolvimento de aplicações para *smartphones* atualmente possuímos quatro possibilidades, que devem ser avaliadas com cautela, pois cada opção possui vantagens e desvantagens em relação às demais:

- a. Desenvolvimento de Aplicativos Nativos: Estas aplicações são poderosas, realistas e muito rápidas, porém possuem o advento de não serem portáteis, ou seja, é necessário o desenvolvimento de aplicativos diferentes para cada sistema operacional de *smartphone* disponível.
- b. Aplicações híbridas: Estas aplicações tem a vantagem de poder acessar recursos do hardware do telefone, ao mesmo tempo em que possuem a vantagem de possuírem parte web, que pode ser apresentada na versão do dispositivo para outras plataformas. Porém neste modelo de desenvolvimento tanto quanto o nativo é necessário o desenvolvimento de parte da aplicação em código nativo.
- c. Aplicações Web dedicadas: Este modelo de aplicação web é desenvolvida totalmente WEB, porém foca em uma plataforma específica, de modo a atender de forma específica as necessidades e padrões da mesma.
- d. Aplicações WEB: Aplicações puramente WEB possuem a vantagem de serem multiplataforma, ou seja, não a necessidade de adaptação do fonte para rodar em diferentes plataformas. Entre seus problemas podemos citar a não integração com recursos do hardware.

3.1 APLICATIVOS WEB X APLICATIVOS NATIVOS

No caso da versão móvel do UCS virtual iremos aprofundar nosso estudo entre aplicativos nativos e aplicativos web, pois o principal ponto levantado juntamente à equipe de desenvolvimento do UCS Virtual é definir se o custo benefício para desenvolver em multiplataforma é favorável.

Desta maneira comparando apenas estes dois modelos, primeiramente serão citadas algumas das vantagens do desenvolvimento nativo:

- a. Aplicativos nativos podem acessar a área de notificação do *smartphone*, uma importante característica disponível no *Android* e no *iOS*, onde é possível verificar atualizações no aplicativo sem estar acessando o mesmo.
- b. A experiência de uso do aplicativo será muito superior, pois o aplicativo é moldado conforme as particularidades do Sistema Operacional utilizado.
- c. São criados ícones de acesso direto ao aplicativo no *smartphone*, de maneira a tornar o acesso por parte do usuário muito fácil e simples.
- d. É possível acessar grande parte dos recursos do *smartphones* como câmera, agenda, entre outros.
- e. Não existe como usuário acessar erroneamente outro aplicativo, pois o acesso é realizado através de ícones, não há problema de digitação de url incorreto, como pode ocorrer com aplicativo puramente web.

Já em relação ao desenvolvimento de um aplicativo WEB podemos citar as vantagens abaixo conforme descrito por Narayanasamy (2010):

1. Capacidade da Interface: Com o desenvolvimento de aplicativos móveis (WEB) existe uma capacidade de exibição de imagens e vídeos superior a aplicativos nativos.
2. Multiplataforma: Não há necessidade de desenvolvimento de um aplicativo para cada plataforma existente.
3. Atualizações Automáticas: Com a aplicação WEB não há necessidade de realizar atualização de aplicativo no *smartphone* do usuário. As atualizações e correções de erro ocorrem de forma automática, quando o usuário acessa novamente o site, já estará acessando a última versão disponível do mesmo.
4. Aprovação Automática: Uma aplicação WEB, não precisa passar por aprovação de lojas virtuais, no caso de aplicações nativas há necessidade de aprovação do “*Android Market*” ou “*Apple Store*” por exemplo.

O portal Elliott Lemenager (2012) faz um comparativo entre os aplicativos web e os aplicativos nativos. Através do quadro apresentado na Tabela 1, podemos resumir este comparativo.

Tabela 1 – Comparativo entre aplicativos WEB e Nativos

Funcionalidade	Aplicações WEB	Aplicações Nativas	Venc.
Necessidade de Acesso a Internet	Não funcionam sem acesso a internet.	Não há necessidade de acesso a internet para funcionarem.	Nativo
Instalação e Atualização	Não necessita.	É necessário instalar o aplicativo e realizar as atualizações	WEB
Interface com o Usuário	Com as novas tecnologias WEB está se aprimorando e se aproximando do nível de excelência das versões nativas.	Possui um bom nível de adaptação ao usuário.	Nativo
Compatibilidade com Dispositivos	O conteúdo pode ser reformatado com CSS e se adequar a qualquer dispositivo.	Dependente da plataforma e do Hardware.	WEB
Animações e Gráficos	Com as novas tecnologias WEB está se aprimorando e se aproximando do nível de excelência das versões nativas.	Possui um bom nível de adaptação ao usuário.	Nativo
Streaming de Mídia	Ainda apresenta alguns problemas de compatibilidade.	Apresenta pequenos problemas no áudio/vídeo. Levemente superior a aplicativos WEB.	Nativo
Fontes	Quase perfeita, devido aos avanços ocorridos nas tecnologias	Apresenta nível próximo da perfeição.	Nativo

	de desenvolvimento WEB.		
Compartilhável	Sim. Os links são compartilháveis em qualquer rede social.	Somente se foi construído com integração com as redes sociais.	WEB
Acesso aos sensores do Hardware	Acesso através do browser é complicado. Geralmente não passa da localização geográfica.	Acesso a todos os recursos de sensores do dispositivo.	Nativo
Desenvolvimento	É necessário desenvolver apenas uma vez e estará disponível para todas as plataformas.	É necessário desenvolver uma versão para cada plataforma.	WEB
Venda/Propagandas no Aplicativo	Propaganda são de fácil inserção, porém a venda da funcionalidade é mais complexa.	Venda do aplicativo de forma simplificada, através das lojas virtuais de cada plataforma.	Nativo
Distribuição	A partir da publicação na WEB estará disponível para todos.	A necessidade da aprovação pelas lojas virtuais.	WEB
Acesso externo ao conteúdo da aplicação	Sim. Necessário apenas acessar o link.	Não, necessário de download do aplicativo.	WEB

Através do quadro acima podemos concluir que não existe o paradigma de desenvolvimento definitivo, ambos possuem prós e contras e se adaptam melhor a diferentes tipos de aplicação, cabendo ao desenvolvedor identificar qual atenderá melhor sua necessidade.

3.2 PRINCIPAIS TECNOLOGIAS PARA DESENVOLVIMENTO DE APLICATIVOS NATIVOS

No desenvolvimento de aplicativos móveis nativos, existem várias plataformas presentes no mercado como: *iOS*, *Android*, *BlackBerry*, *Windows Phone*, *Symbian*, *Bada*, entre outros. Neste trabalho estudaremos mais profundamente o desenvolvimento nativo para *iOS* (Sistema Operacional móvel da Apple) e para *Android* (Sistema Operacional móvel da Google), que segundo levantamento realizado pela *ComScore* em abril de 2012 juntos somam mais de 80% do mercado norte americano, conforme pode ser verificado na Figura 2.

Top Smartphone Platforms 3 Month Avg. Ending Apr. 2012 vs. 3 Month Avg. Ending Jan. 2012 Total U.S. Smartphone Subscribers Ages 13+ Source: comScore MobiLens			
	Share (%) of Smartphone Subscribers		
	Jan-12	Apr-12	Point Change
Total Smartphone Subscribers	100.0%	100.0%	N/A
Google	48.6%	50.8%	2.2
Apple	29.5%	31.4%	1.9
RIM	15.2%	11.6%	-3.6
Microsoft	4.4%	4.0%	-0.4
Symbian	1.5%	1.3%	-0.2

Figura 2 - Plataformas para Smartphones

3.2.1 Desenvolvimento de aplicativos nativos para Android

O sistema operacional *Android* da Google é um sistema operacional livre de código aberto, que utiliza uma versão modificada do sistema operacional Linux. Atualmente o sistema operacional *Android* é mantido pela OHA (*Open Handset Alliance*), grupo de 40 empresas criado para fortalecer e agilizar o desenvolvimento para esta plataforma.

Referente à arquitetura, podemos descrevê-lo com uma pilha de softwares para dispositivos móveis que inclui sistema operacional, *middleware* e aplicações-chave. Esta pilha possui quatro níveis (Google, 2011), conforme Figura 3.

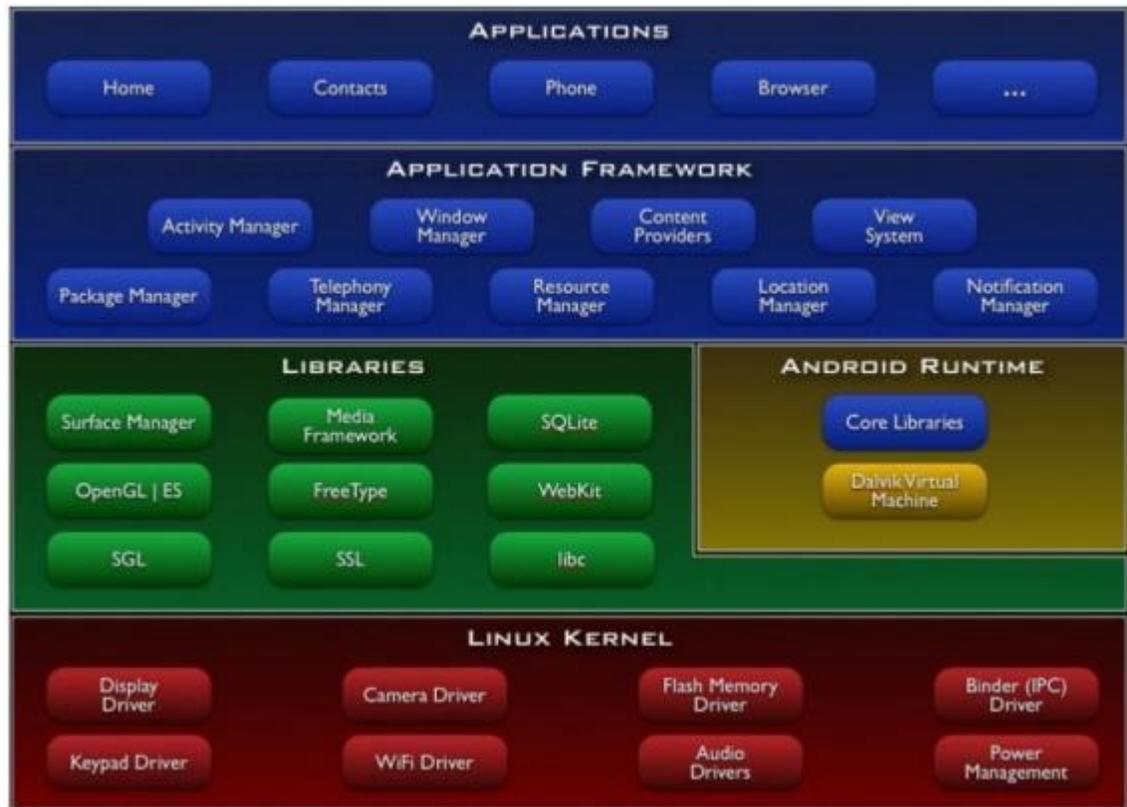


Figura 3 - Arquitetura

1 - Linux Kernel: O Google usou a versão 2.6 do Linux para construir o *kernel* do *Android*, que incluem serviços essenciais como: gerenciamento de memória, gerenciamento de processos, gerenciamento de energia, configurações de segurança, configurações de rede e *drivers*.

2 - Runtime Android e Libraries: acima do *kernel* estão as bibliotecas do *Android* e o *android runtime*. *Android runtime* consiste de um conjunto de bibliotecas que fornece a maioria das funcionalidades disponíveis nas principais bibliotecas da linguagem de programação Java e de uma Máquina Virtual Dalvik (DVM). Uma aplicação *Android* roda em seu próprio processo, com a sua própria instância da máquina virtual Dalvik. Além disso, o *Android* possui um conjunto de biblioteca em C\C++, onde as principais são: *System C library*, *Media Libraries*, *Surface Manager*, *LibWebCore*, *SGL*, *3D libraries*, *FreeType*, *SQLite*.

3 - Application Framework: consiste nos programas que gerenciam as funções básicas do telefone, tais como, alocação de recursos, aplicações de telefone, mudança entre processos ou programas e informações sobre a localização física do aparelho.

4 - Applications: funções básicas do dispositivo, como fazer chamadas telefônicas, acessar o navegador web ou acessar sua lista de contatos. Esta é a camada do usuário comum, que utiliza a interface de usuário.

No desenvolvimento de um aplicativo se trabalha basicamente na camada de Aplicação do *Android*, podendo acessar recursos das demais camadas. Uma aplicação *Android* pode ser dividida em componentes:

- a) Atividades: Funcionam como mediadores que definem como as informações serão apresentadas ao usuário, além de controlar o fluxo da aplicação. Elas podem interagir com o usuário e trocar informações com outras *activities* ou *services* (MEIER, 2009).
- b) Serviços: Programas que são executados em segundo plano. Não tem interação direta com o usuário e podem ser executados por tempo indeterminado.
- c) *Broadcast* e *Intent Receivers*: São componentes que ficam aguardando a ocorrência de um determinado evento, pode-se entender como evento a inicialização do sistema operacional, uma chamada de voz, a chegada de um SMS, um evento disparado por uma aplicação (MEIER, 2009).
- d) *Content Providers*: São os compartilhadores de conteúdo entre as aplicações, uma aplicação pode requisitar informações de outra, por exemplo, uma aplicação pode receber dados da lista de contatos que é nativa do *Android*, e com base nesses dados, realizar algum processamento (LECHETA, 2010).

O desenvolvimento de aplicações em *Android* é realizado através de SDK liberado pelo Google. O SDK é um conjunto de ferramentas utilizadas para desenvolver aplicações para a plataforma *Android*. Possui um emulador para simular o dispositivo móvel e uma API completa para a linguagem Java, com todas as classes necessárias para desenvolver as aplicações (BURNETTE, 2008).

3.2.2 Desenvolvimento de Aplicativos nativos para IOS.

O *iOS* é o sistema operacional móvel da Apple, é utilizado exclusivamente em seus dispositivos móveis (*iPod*, *iPhone*, *iPad*, ...). Desde que foi lançado possui índices de vendas excelentes e se tornou referência no mercado.

A arquitetura do *iOS* foi desenvolvida aproveitando muitas das tecnologias do Mac OS X (sistema operacional da Apple para desktops e notebooks). Seu *kernel*, por exemplo, é uma variante do existente no Mac OS X. A Figura 4, exibe a arquitetura das camadas do *iOS* em alto nível, onde podemos verificar a existências de quatro camadas: (Apple Developer, 2012)

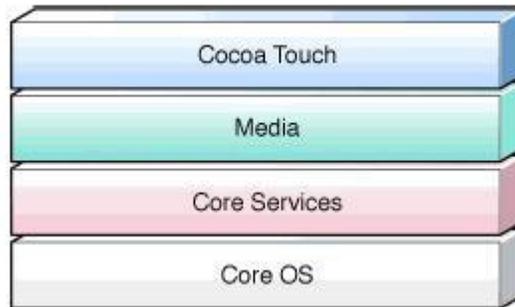


Figura 4 - Arquitetura iOS

1 - Core OS: É a camada mais baixa do iOS, consistindo em uma versão reduzida do núcleo do Mac OS X, sendo um sistema UNIX. Nesta camada também existem recursos de gerenciamento de energia em dispositivos móveis, segurança e rede.

2 - Core Services: Esta camada fornece serviços fundamentais que praticamente todos os aplicativos fazem uso, entre eles podemos citar: *Core Foundation* (interface baseada em C para coleções compostas de listas e dicionários), *Core Data* (framework para gerenciar os modelos de dados de uma aplicação), *Core Location* (gerenciamento de dados de localização), triangulação, Wi-Fi, interface de baixo nível para SQL Lite e *parser* para XML.

3 - Media: É a camada responsável pelos recursos multimídia disponíveis, tais como: gráficos, áudio, vídeos, animações.

4 - Cocoa Touch: contém os principais frameworks para construção de aplicativos para iOS, podendo ser divididas em duas sub-camadas, onde no nível mais baixo não é de interface com o usuário, possuindo recursos de rede, acesso a arquivos e APIs. No nível mais alto desta camada está presente toda a infra-estrutura de aplicativos e os componentes gráficos, além de acesso a alguns dispositivos de hardware e sensores como câmera e acelerômetro.

O desenvolvimento de aplicações para *iOs* é realizada utilizando a linguagem de programação *Objective-C*, a qual é a principal linguagem *Cocoa API* da Apple, e conseqüentemente é utilizada tanto para o desenvolvimento de aplicações para *iOS* como para *MAC OS X*. O ambiente de desenvolvimento para *iOS* é o *XCODE* (ambiente proprietário desenvolvido pela Apple que é compatível apenas com o sistema operacional *MAC OS X*).

3.3 PRINCIPAIS TECNOLOGIAS PARA DESENVOLVIMENTO DE APLICATIVOS WEB

Quando pensado no desenvolvimento de aplicativos WEB a grande tendência do mercado é a utilização do *HTML5* e *CSS3*. Grandes empresas estão migrando seus sites e portais para estas tecnologias, tanto para desenvolvimento de versão desktop como móvel. Um exemplo de migração para o *HTML5* e *CSS3* é a própria Universidade de Caxias do Sul, que está refatorando a versão desktop do portal *UCSVirtual* para estas tecnologias.

Segundo estudo realizado pela empresa *Strategy Analytics* (2011), a tendência de utilização do *HTML5* é aumentar cada vez mais em dispositivos móveis, visto que a maioria dos *smartphones* vendidos atualmente possui suporte a esta tecnologia e o *Adobe Flash* está em descontinuação para estes dispositivos.

3.3.1 HTML5

Segundo o livro *HTML5 – a linguagem de marcação que revolucionou a WEB* (Silva, 2011), o *HTML5* é uma evolução da linguagem de marcação *HTML* (*HTML*, *HTML+*, *HTML2.0*, *HTML3.0*, *HTML3.2*, *HTML4.0* e *HTML4.01*), que foi originalmente projetada para descrição de documentos científicos, e hoje largamente utilizada na Internet para o desenvolvimento dos mais variados sites e portais em suas diversas especificações.

A linguagem *HTML5* hoje não é suportada por todos os browsers disponíveis no mercado, na verdade, a norma de definição da mesma ainda não está homologada oficialmente pelo *W3C* (estando em fase de testes). Embora enfrentando alguns problemas de compatibilidade e sua não homologação oficial, a maioria dos *browsers* suporta grande parte das funcionalidades existentes na mesma, o que viabiliza sua utilização. Além disso, existem técnicas para detecção

se a funcionalidade é suportada pelo navegador utilizado, permitindo assim que o programador desenvolva soluções alternativas para funcionalidades não suportadas.

A grande vantagem de utilização do HTML5 é a transformação do HTML de um simples exibidor de conteúdo WEB em uma linguagem poderosa que permite a utilização de recursos em aplicações WEB que antes somente eram possíveis em aplicações *desktop*. Com uma aplicação HTML5 é possível criação de gráficos bidimensionais, exibição de vídeos e animações de forma nativa, realizar interação com a página mesmo estando *off-line*, edição de imagens, utilização de recursos de geolocalização, entre outros.

A experiência de uso de uma aplicação em HTML5 é muito superior ao que estamos acostumados com as versões anteriores HTML e elimina a necessidade de utilização de muitos *plug-ins* que eram utilizados para dar “vida” às aplicações WEB como o *Flash*, *JavaFX* e *Silverlight*. Com esta linguagem o poder de uma aplicação WEB está cada vez mais próximo do que é possível realizar em aplicações *desktop*, e grandes empresas estão apostando alto nesta tecnologia, como o Google, que desenvolveu um sistema operacional totalmente voltado para WEB, o *ChromeOS*.

Para exemplificar o que é possível fazer com o HTML5 podemos citar aplicações como o Google Docs², no qual o usuário possui uma experiência de uso muito semelhante ao que é possível quando utilizado um editor de textos em um Desktop. Pensando em dispositivos móveis, podemos exemplificar o potencial desta linguagem em web aplicativos como o Facebook³ e Gmail⁴.

Segundo a W3C (2012), as principais *tags* incluídas na especificação do HTML5, são:

- `<article>`: *Tag* incluída para facilitar a inserção de artigos dentro de uma página WEB, seu potencial uso é principalmente para postagem em fóruns, blogs, comentários e artigos.
- `<aside>`: *Tag* para definição de conteúdo dentro do conteúdo.
- `<audio>`: Esta *tag* define som como música e outros streamings de áudio. Possui vários atributos, entre os quais, podemos citar atributo que define que a música será

² <http://docs.google.com>

³ <http://www.facebook.com>

⁴ <http://www.gmail.com>

iniciada automaticamente ou atributo que definirá que a música ficará em loop na página.

- <BDI>: Está *tag* permite isolar uma parte do texto para que seja formatada de maneira diferente do restante.
- <canvas>: É utilizada para desenhar gráficos. É somente um container para gráficos, sendo que para realizar seu desenho é necessário uma linguagem de script como o JavaScript.
- <command>: Define um comando (radiobutton, checkbox ou commandbutton) que o usuário pode invocar. Possui vários atributos que podem, por exemplo: habilitar / desabilitar o componente, flegar⁵, colocar uma descrição ou ícone.
- <datalist>: Define uma lista de opções pré-definidas para o a *tag* de <input>, permitindo desta maneira a função de auto-completar.
- <details>: Especifica detalhes adicionais que o usuário pode ver ou esconder sobre demanda.
- <embed> : Define um container para uma aplicação externa ou plug-in.
- <figcaption>: Define uma legenda para um elemento de <figure>;
- <figure>: Utilizada para inserção de conteúdos como imagens, ilustrações, diagramas, fotos, lista de códigos, etc.
- <footer>: Define o rodapé de um documento ou seção. Normalmente um rodapé contém o autor do documento, informações de copyright, links para os termos de uso, informações de contato, etc. Não há limite referente à quantidade deste elemento existente no documento.
- <header>: Define o cabeçalho de um documento ou seção. Deve ser utilizado como um container de conteúdo introdutório ou para setar links de navegação. Não há limite referente à quantidade deste elemento existente no documento.
- <hgroup>: É utilizada para agrupar elementos de cabeçalho. É utilizado para agrupar elementos <h1> a <h6>, quando o cabeçalho possui múltiplos níveis.
- <keygen>: Especifica um campo gerador de chaves utilizado para formulários.

⁵ Flegar: marcar um componente do tipo check-box.

- `<Mark>`: Utilizada para realizar marcações no texto. Deve ser utilizada se há necessidade de destacar parte do texto.
- `<meter>`: Define uma medição escalar dentro de uma gama conhecida, ou um valor fracionado. Este também é conhecido como um indicador. Não deve ser utilizado como barra de progresso. Exemplos: uso do disco, a relevância do resultado de uma consulta, etc.
- `<nav>`: Define uma seção de elementos de navegação. Não deve ser utilizado em todos os links de navegação de um documento, somente nos bloco principal de itens de navegação.
- `<output>`: Representa o resultado de um cálculo (como um retornado de um script).
- `<progress>`: Representa o resultado de um progresso, como uma barra de progresso.
- `<rp>`: Define o que será mostrado se o browser não possuir suporte a anotações em ruby.
- `<rt>`: Define uma explicação ou pronúncia de caracteres em uma anotação em ruby.
- `<ruby>`: Define uma anotação em ruby.
- `<section>`: Define seções em um documento. Tal como capítulos, cabeçalhos, rodapés, ou quaisquer outras seções do documento.
- `<source>`: É usada para especificar vários recursos de mídia para elementos de mídia, tais como `<video>` e `<audio>`. A tag `<source>` permite que você especifique vídeo alternativo / arquivos de áudio que o navegador pode escolher, com base em seu tipo de mídia ou suporte codec.
- `<summary>`: A tag `<summary>` define uma posição visível para o elemento `<dt>`. O título pode ser clicado para exibir / ocultar os detalhes.
- `<time>`: Define tanto hora como data no calendário gregoriano.
- `<track>`: Especifica o texto das faixas de elementos de mídia (`<audio>` e `<video>`).
- `<vídeo>`: Define um vídeo como um filme ou outro streaming de vídeo.
- `<wbr>`: Especifica onde deve ocorrer uma quebra de linha em uma página.

A compatibilidade destas *tags* com os principais navegadores WEB existentes no mercado ocorre conforme a Tabela 2.

Tabela 2 - Compatibilidade *tags* HTML5

Tag	Internet Explorer	Firefox	Opera	Chrome	Safari
<article>	versão 9 em diante	Sim	Sim	Sim	Sim
<aside>	versão 9 em diante	Sim	Sim	Sim	Sim
<audio>	versão 9 em diante	Sim	Sim	Sim	Sim
<bdi>	Não	Sim	Não	Sim	Não
<canvas>	versão 9 em diante	Sim	Sim	Sim	Sim
<command>	versão 9 em diante	Não	Não	Não	Não
<datalist>	Não	Sim	Sim	Não	Não
<details>	Não	Não	Não	Sim	Não
<embed>	Sim	Sim	Sim	Sim	Sim
<figcaption>	versão 9 em diante	Sim	Sim	Sim	Sim
<figure>	versão 9 em diante	Sim	Sim	Sim	Sim
<footer>	versão 9 em diante	Sim	Sim	Sim	Sim
<header>	versão 9 em diante	Sim	Sim	Sim	Sim
<hgroup>	versão 9 em diante	Sim	Sim	Sim	Sim
<keygen>	Não	Sim	Sim	Sim	Somente no MAC
<mark>	versão 9 em diante	Sim	Sim	Sim	Sim
<meter>	Não	Não	Sim	Sim	Não
<nav>	versão 9 em diante	Sim	Sim	Sim	Sim
<output>	Não	Sim	Sim	Sim	Sim
<progress>	Não	Sim	Sim	Sim	Não
<rp>	versão 9 em diante	Sim	Sim	Sim	Sim
<rt>	versão 9 em diante	Sim	Sim	Sim	Sim
<ruby>	versão 9 em diante	Sim	Sim	Sim	Sim
<section>	versão 9 em diante	Sim	Sim	Sim	Sim
<source>	versão 9 em diante	Sim	Sim	Sim	Sim
<summary>	Não	Não	Não	Sim	Não
<time>	Não	Não	Não	Não	Não
<track>	Não	Não	Não	Não	Não
<video>	versão 9 em diante	Sim	Sim	Sim	Sim
<wbr>	Não	Sim	Sim	Sim	Sim

Como se pode perceber as maiorias das novas *tags* disponibilizadas no HTML5 são compatíveis com a maioria dos navegadores existentes e os navegadores que atualmente possuem maior suporte a elas são o *Chrome* e o *Opera*, enquanto o *Internet Explorer* apresenta maior nível de incompatibilidade.

3.3.2 CSS3

Segundo o livro CSS3: Visual QuickStart Guide, o CSS é uma linguagem utilizada para definição de aparência em pagina para WEB escritas utilizando a linguagem HTML. Foi criada com a ideia de separar a parte de interface do conteúdo de uma página HTML, dando mais autonomia para o designer e mantendo o controle melhor sobre a parte visual da aplicação WEB (Teague, 2010).

O objetivo de separar a parte do conteúdo da página da sua formatação, foi cumprido desde as primeiras versões desta linguagem, porém o controle total do designer na parte de interface não conseguiu ser atingido em sua totalidade, pois os programadores ainda necessitavam realizar algumas alterações diretamente no HTML para obterem alguns efeitos de interface que o CSS não conseguia suprir.

Nas versões posteriores do CSS (CSS1 e CSS2), já existiram alguns avanços no sentido de aumento do controle total do designer sobre a interface da aplicação, porém evolução significativa neste item está ocorrendo com o CSS3.

Deste modo, a novidade mais importante do CSS 3, para os desenvolvedores de webs, consiste na incorporação de novos mecanismos para manter um maior controle sobre o estilo com o qual se mostram os elementos das páginas, sem ter que recorrer a truques ou hacks, que a muitas vezes complicavam o código das webs.

O CSS3 vem sendo largamente utilizado combinado com o HTML5, permitindo a criação de aplicações poderosas com design atraente e mantendo o código das aplicações WEB elegantes. Muitas empresas estão apostando nesta combinação para a evolução de suas aplicações.

3.4 CONCLUSÃO DO CAPÍTULO

No decorrer deste capítulo podemos verificar que existem vantagens e desvantagens para o desenvolvimento de uma aplicação para dispositivos móveis utilizando tecnologia nativa ou aplicações web. Também diagnosticamos que quando pensado em tecnologias nativas o maior

foco do mercado está para as plataformas iOS e Android enquanto no desenvolvimento de Web Apps a aposta está na combinação entre HTML5 e CSS3.

Para auxiliar na decisão da tecnologia a ser utilizada no desenvolvimento da versão móvel do portal UCSVirtual, serão utilizados os itens descritos no quadro em que foi realizado o comparativo entre Aplicações Nativas e WEB Apps direcionando as necessidades do desenvolvimento: (Elliott Lemenager, 2012)

- **Necessidade de Acesso à Internet:** Independente de qual tecnologia utilizada, será necessário que a aplicação tenha acesso a Internet constantemente para busca de informações do portal.
- **Instalação e Atualização:** Neste item, podemos atribuir vantagem ao desenvolvimento de uma WEB App, pois o portal UCSVirtual, tende a sofrer atualizações constantemente, o que torna desejável que seus usuários acessem sempre a última versão disponível.
- **Interface com o Usuário:** Podemos atribuir leve vantagem ao desenvolvimento nativo, pois estará mais integrado aos padrões e conceitos da plataforma para a qual está sendo desenvolvida.
- **Compatibilidade com dispositivos:** Neste item certamente a vantagem no desenvolvimento de um WEB APP, pois não ficará preso a uma ou mais plataformas, podendo ser acessado por praticamente todas as plataformas disponíveis no mercado.
- **Animações e Gráficos:** Este item não possui grande relevância na escolha da tecnologia a ser utilizada. Não existem animações dentro do portal e gráficos são raramente utilizados.
- **Streaming de Mídia:** Atualmente não existem recursos de Streaming de Mídia no UCSVirtual, não possuindo relevância neste momento, porém podendo ter importância no futuro.
- **Fontes:** Neste item podemos atribuir leve vantagem ao desenvolvimento nativo, embora, esta questão esteja se aprimorando cada vez mais no desenvolvimento de WEB APPs.
- **Possibilidade de busca de conteúdo na WEB:** não possui relevância em nosso desenvolvimento.
- **Compartilhável:** Neste item podemos atribuir leve vantagem para o desenvolvimento de WEB APPs, visto que a aplicação estará conectada a internet durante todo o tempo de uso.

- **Discussão e Colaboração:** Neste item podemos atribuir leve vantagem para WEB App, onde todo o desenvolvimento estará na nuvem.
- **Acesso aos sensores do Hardware:** O portal UCSVirtual em sua versão móvel, não necessitará acessar recursos do hardware, pois sua maior utilização é para obtenção e envio de informações e arquivos, não sendo este um diferencial para escolha da tecnologia.
- **Desenvolvimento:** Neste item podemos atribuir grande vantagem para o desenvolvimento de um WEB App, pois não haverá necessidade de replicação do desenvolvimento e das manutenções necessárias.
- **Venda/Propagandas no Aplicativo:** não possui relevância em nosso desenvolvimento
- **Distribuição:** podemos atribuir vantagem ao desenvolvimento de um WEB App, pela simplificação da liberação do aplicativo e de suas atualizações.
- **Acesso externo ao conteúdo da aplicação:** Vantagem para o desenvolvimento de WEB App, pois é possível o acesso ao aplicativo através de dispositivos móveis de terceiros, não necessitando de *download* e instalação.

No caso do portal UCSVirtual existe vantagem para o desenvolvimento de um Web App, pois a maioria das vantagens existentes no desenvolvimento nativo não são aplicáveis ao portal e a questão de unificação do desenvolvimento, simplicidade de distribuição e atualização fazem a diferença no momento desta escolha, portanto nosso projeto focará no desenvolvimento de um Web APP utilizando HTML5 e CSS3 que são tecnologias que estão emergindo e atingindo alto nível de aceitação no mercado.

4 - PORTAL UCSVIRTUAL

O UCSVirtual é um portal disponibilizado pela Universidade de Caxias do Sul, para que seus alunos e professores tenham acesso aos principais serviços da universidade on-line. Sobre a visão do aluno de graduação existem as seguintes funcionalidades hoje disponibilizadas no mesmo:

1. Ambiente Virtual de Aprendizagem (AVA Graduação): Esta é a principal funcionalidade do UCSVirtual. Este possui vários serviços que podemos destacar abaixo:
 - a. Mensagens Instantâneas: Este serviço realiza o gerenciamento de mensagens recebidas e enviadas pelo aluno. Neste item geralmente encontram-se informações postadas por professores, coordenadores de cursos e até mesmo diálogo entre alunos. Este serviço agrupa as mensagens recebidas e enviadas.
Os serviços abaixo, a partir do item “b”, são habilitados uma instância para cada disciplina que o aluno esta cursando. Também é possível acessar o mesmo para as disciplinas cursadas em semestres anteriores.
 - b. Mural de Recados: Neste contém informações importantes postadas pelo professor da disciplina.
 - c. Cronograma: Contém as informações do cronograma, postadas pelo professor no início do semestre. Através deste é possível o aluno se programar referente aos dias que cada conteúdo, trabalhos e provas estão planejados.
 - d. Lista de Participantes: Este serviço contém o nome de todos os participantes da disciplina (alunos e professor) e link para acesso ao perfil de cada um destes.
 - e. Informações da Turma: Possui as informações referente à avaliação, plano de ensino e metodologia da disciplina.
 - f. Fórum: Serviço disponibilizado para que alunos e professores discutam sobre assuntos referentes à disciplina.
 - g. Acervo da Turma: Contém material postado pelo professor e disponível para download dos alunos referente ao conteúdo da disciplina.
 - h. Webfólio: Serviço onde o professor pode realizar a criação de pastas para postagem de trabalhos por parte dos alunos.
3. Komunikas: Rede social para alunos da UCS.

4. Biblioteca: Neste item é habilitado links para acesso aos serviços e bases de dados de pesquisa da Universidade de Caxias do Sul.
5. Atividade Complementares: Este serviço permite ao aluno informar cursos, seminários e demais atividades que podem ser utilizados para comporem as horas complementares necessárias nos cursos de graduação.
6. Avaliação On-line: serviço que permite ao aluno avaliar as disciplinas, a coordenação do curso, o curso e a estrutura da Universidade como um todo.
7. Demonstrativo de Pagamento: este serviço permite consulta de pagamentos realizados, bem como extrato para declaração do imposto de renda e emissão de boletos de mensalidades em aberto.
8. Documentos e Formulários para download: neste é possível realizar o download dos principais documentos e formulários normativos da Universidade de Caxias do Sul.
9. Estudo de Currículo: serviço que disponibiliza ao aluno o acompanhamento de seu currículo, facilitando o entendimento de quais disciplinas estão pendentes além de poder verificar o desempenho obtido nas disciplinas já cursadas.
10. Histórico de Impressões: Este serviço auxilia o aluno no controle de impressões que o mesmo ainda possui disponível, destacando quantas folhas já foram impressas e quantas ainda estão disponíveis para impressão.
11. Histórico Escolar: Este item contém todas as disciplinas cursadas com o aproveitamento obtido, além dos cursos de extensão e horas complementares.
12. Matrícula: Este serviço permite a realização de todo o gerenciamento do processo de matrícula do aluno. Desde a solicitação até a efetiva contratação das disciplinas e impressão do requerimento de matrícula (documento que contém as disciplinas contratadas, dia da semana, local, entre outras informações).
13. Portais dos Cursos: Acesso para cada curso de graduação da Universidade de Caxias do Sul.
14. Protocolo Acadêmico: Contém o histórico de todos os serviços solicitados ao protocolo acadêmico e a situação dos mesmos.
15. Registro de Aproveitamento: Este serviço disponibiliza as notas parciais e finais das disciplinas que estão sendo cursadas no semestre, além da quantidade de faltas registradas.

16. Seguro Educacional: Contém as informações do seguro educacional contratado no momento da matrícula.
17. Pesquisa: Item que disponibiliza documentos referentes à pesquisa além de informações referentes ao encontro de jovens pesquisadores.
18. Portais das Unidades: Disponibiliza link para acesso ao portal de cada um dos centros da Universidade de Caxias do Sul.

Neste capítulo, pode-se perceber que o portal UCSVirtual é uma ferramenta completa, para o gerenciamento das atividades acadêmicas dos alunos. Este portal foi responsável pela diminuição das fronteiras existentes entre o aluno e a universidade, simplificando o acesso a informações e facilitando a comunicação entre ambos. O desenvolvimento de uma versão móvel do UCSVirtual fará com que a UCS fique ainda mais próxima dos alunos, facilitando e ampliando o acesso aos seus serviços.

5 - PROPOSTA DE INTERFACE PARA VERSÃO MÓVEL PARA O PORTAL UCSVIRTUAL

Para identificar as funcionalidades que deveriam ser implementadas prioritariamente em uma versão móvel do UCSVirtual, foi realizada uma pesquisa junto a usuários do mesmo. Esta pesquisa ocorreu de forma virtual, através de formulários do Google Forms, e enviada para lista de email de alunos da UCS, entre os dias 11/06/2012 e 18/06/2012, sendo respondida por 78 alunos (usuários do UCSVirtual) da Universidade de Caxias do Sul.

Com base nesta pesquisa foi identificado que:

- 95% (74 alunos) dos alunos entrevistados utilizam o portal UCSVirtual.
- Em relação à frequência de utilização do UCSVirtual, foi identificado que 22% (17) dos alunos utilizam pelo menos uma vez por semana, 33% (26) dos alunos utilizam o portal todos os dias, 42% (33) utilizam mais de uma vez por semana.
- 69% (54 alunos) dos entrevistados possuem *smartphone* ou *tablet* e utilizam a Internet através destes dispositivos.
- 32% (25 alunos) utilizam a versão existente do UCSVirtual através de *tablets* ou *smartphones*.
- 91% (71 alunos) utilizariam uma versão móvel do UCSVirtual, se disponível.

Referente às funcionalidades do UCSVirtual de maior utilização, foi identificada a utilização conforme a Tabela 3.

Tabela 3 - Funcionalidades mais utilizadas no UCSVirtual

Funcionalidade	Qt. Utilização	Percentual
AVA Graduação – Webfólio	64	84,00%
AVA Graduação – Mural	58	76,00%
AVA Graduação – Acervo	58	76,00%
Registro de Aproveitamento	56	74,00%
Matrícula Graduação	55	72,00%
AVA Graduação - Cronograma das Aulas	55	72,00%
AVA Graduação - Lista de Participantes	49	64,00%
Gerenciador Financeiro	48	63,00%
AVA Graduação - Informações da	43	57,00%

Turma		
Estudo Currículo Graduação	40	53,00%
Boletim Matrícula Graduação	37	49,00%
Histórico – Graduação	37	49,00%
Avaliação On-line	36	47,00%
Mensagens Instantâneas	32	42,00%
AVA Graduação – Fórum	23	30,00%
Comprovante de Pagamentos	20	26,00%
Documentos – Graduação	20	26,00%
Atividades Complementares	14	18,00%
Histórico de Impressões	9	12,00%
Komunikas	2	3,00%

Referente ao desenvolvimento de uma versão móvel do UCSVirtual solicitamos aos usuários, quais funções ou sub-funções do portal os mesmos gostariam que fossem implementadas, e foram obtidas as respostas detalhadas na Tabela 2.

Tabela 4 - Funcionalidades sugeridas para versão móvel do UCSVirtual

Funcionalidade	Qt. Utilização	Percentual
AVA - Cronograma	17	21,79%
AVA – Mural	16	20,51%
Matrícula On-line	15	19,23%
AVA – Acervo	13	16,67%
Localização de Sala e Horário	11	14,10%
Registro de Aproveitamento	8	10,26%
AVA – Mensagens	8	10,26%
Serviços Biblioteca	6	7,69%
AVA – Webfólio	5	6,41%
AVA – Participantes	5	6,41%
Gerenciados Financeiro	5	6,41%
AVA – Fórum	3	3,85%
Histórico Graduação	3	3,85%
Estudo Currículo	3	3,85%
AVA - Informações da Turma	2	2,56%
Avaliação On-line	2	2,56%
Histórico de Impressões	2	2,56%
Documentos e Formulários	2	2,56%
Liberção Vaga em Espera	1	1,28%
UCS Mail	1	1,28%

Komunikas	1	1,28%
Cursos Extensão	1	1,28%
Atividades Complementares	1	1,28%

Com base na pesquisa realizada junto aos usuários do UCSVirtual, foi constatada a existência de público para uma versão móvel do portal, bem como as funcionalidades prioritárias para este desenvolvimento. Os recursos que aparecem como preferidos pelos usuários estão vinculados ao AVA do Portal, a saber Cronograma e Mural.

5.1 ANÁLISE E DESENVOLVIMENTO DO PROTÓTIPO

Este trabalho visa projetar a interface e o funcionamento da versão móvel do portal UCSVirtual. O desenvolvimento do protótipo foi focado nas seguintes funcionalidades: *login*, interface inicial e nos serviços de mural e cronograma do AVA. Embora não implementando todas as funcionalidades do UCSVirtual, este projeto criou uma interface que poderá ser aplicada a toda a versão móvel do portal.

Este desenvolvimento utilizou os casos de uso conforme Figura 5.

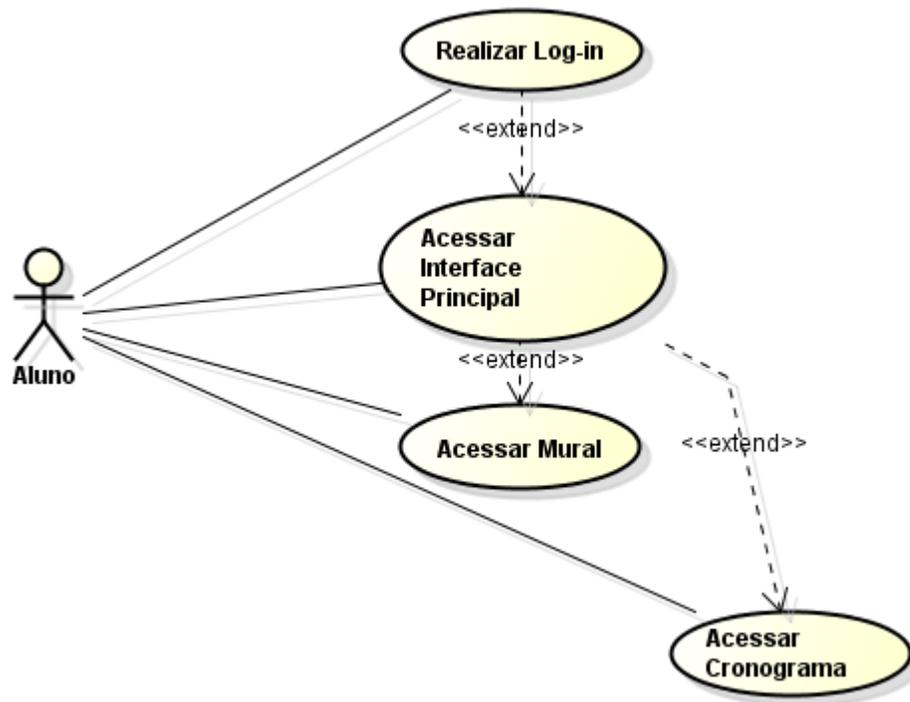


Figura 5 - Diagrama de Casos de Uso

No quadro da Tabela 5 pode ser verificado o caso de uso narrativo referente à realização de *login* no sistema.

Tabela 5 –UC01 – Realizar Login

Identificador:	UC01 – Realizar Login
Descrição:	Realização de Login no sistema.
Ator(res) Primário(s):	Alunos
Pré - Condições:	Não há
Fluxo Principal:	1. Aluno informa o usuário. (Se já informou alguma vez, o sistema oferece opção de auto-complete). 2. Aluno informa a senha. 3. Aluno clica em "Acessar". 4. Login no sistema é concluído com sucesso.
Fluxo Alternativo:	3.a Aluno clica em "Limpar". Sistema limpa os campos "usuário" e "senha". 3.b Aluno clica em "Acessar Versão Tradicional". O Aluno é redirecionado para a versão tradicional do portal para desktops. 4. O usuário ou senha informado estão incorretos. Sistema retorna mensagem avisando do ocorrido e campos de login são limpos.

O caso de uso de realização de *login* possui a interface conforme a imagem exibida na Figura 6.

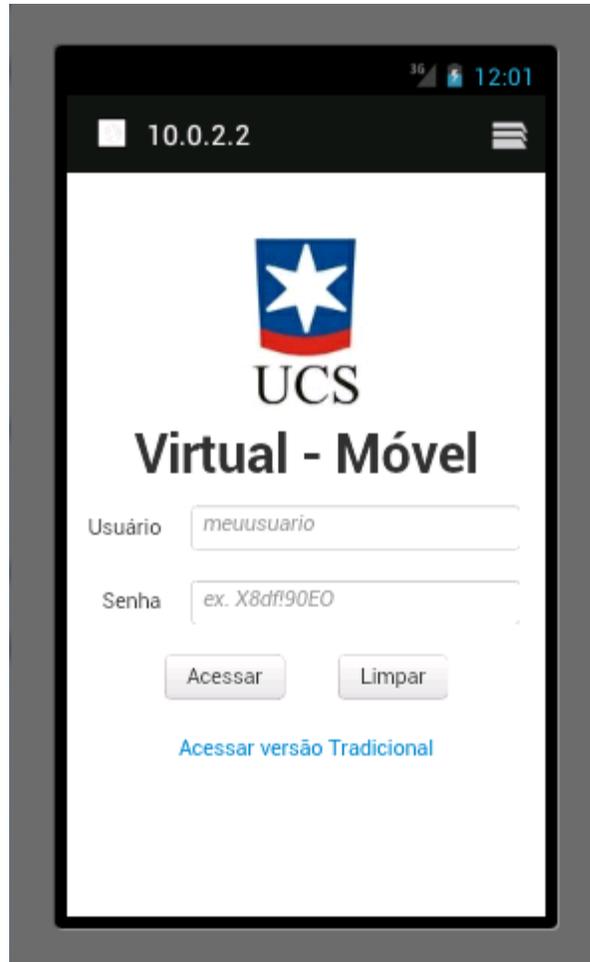


Figura 6 - Interface Tela Login

Quando ocorrer erro no login será exibida mensagem de erro alertando o usuário do ocorrido, conforme Figura 7.



Figura 7 – Erro Login

Na Tabela 6 pode ser verificado o caso de uso narrativo referente à realização de acesso à interface principal do sistema.

Tabela 6 - UC02 – Acessar Interface Principal

Identificador:	UC02 – Acessar Interface Principal
Descrição:	Aluno acessa a interface principal para escolha da funcionalidade que deseja acessar no sistema.
Ator(res) Primário(s):	Aluno.
Pré - Condições:	UC01 – Login
Fluxo Principal:	1. Sistema exibe tela do UCSVirtual Móvel, contendo: Mensagem de boas vindas ao usuário, ícones para: acesso ao menu, tela inicial do aplicativo, sair, e links para exibição das funcionalidades favoritas e atualizações recentes. Por default o sistema exibe as funcionalidades favoritas.

	2. O Aluno clica no ícone de menu. O sistema exibe lista de funcionalidades ao centro com opção de busca.
	3. O Aluno clica na funcionalidade escolhida. Ao centro da tela é executada a funcionalidade solicitada. Existindo ao topo link "Adicionar aos favoritos" que adiciona a funcionalidade na lista de favoritas.
	4. O Aluno clica no ícone "Inicio". O Sistema retorna para a tela inicial do aplicativo.
	5. O Aluno clica em "Recentes". São exibidas as atualizações recentes.
	6. O Aluno clica em "Favoritas". O sistema exibe as funcionalidades Favoritas.
	7. O Aluno clica em "Sair". O Sistema efetua o log-off.
Fluxo Alternativo:	1. Não existem funcionalidades favoritas. O Sistema exibe as atualizações recentes.
	5. Não existem atualizações recentes. O sistema exibe a mensagem ao centro: "Não existem funcionalidades Recentes".
	6. Não existem funcionalidades favoritas. O sistema exibe a mensagem ao centro: "Não existem funcionalidades Favoritas".

A Figura 8 representa a interface referente à tela principal do sistema, acessada após a finalização do caso de uso de realização do *login*.

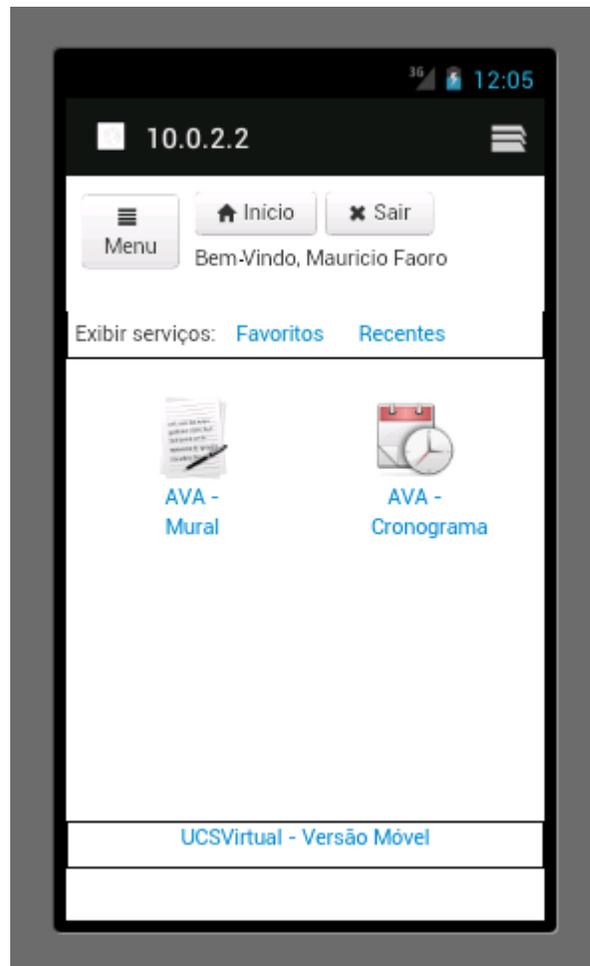


Figura 8 - Interface principal do sistema

A Figura 9 representa o acesso ao menu através da interface principal conforme detalhado no caso de uso de acesso a interface principal do sistema.



Figura 9 - Acesso ao Menu

Na Tabela 7 pode ser verificado o caso de uso narrativo referente ao acesso ao mural.

Tabela 7 – UC03 – Acessar Mural

Identificador:	UC03 – Acessar Mural
Descrição:	Acesso ao Mural
Ator(res) Primário(s):	Aluno
Pré - Condições:	UC02 - Interface Principal
Fluxo Principal:	<ol style="list-style-type: none"> 1. O Sistema exibe tela para seleção da disciplina que o aluno deseja consultar no mural. (Existindo a possibilidade todas. OBS: São consideradas somente as disciplinas do semestre corrente) 2. O Aluno seleciona a disciplina. 3. O Sistema exibe os recados deixado no mural da(s) disciplina(a) (Nome do Professor, Data/Hora e se a mensagem foi lida). Também é exibida funcionalidade

	para permitir a troca de disciplina.
	4. O Usuário clica no recado desejado. O Sistema exibe nova tela com a descrição da mensagem (com rolagem, caso a mensagem for muito grande) e possuindo opção de "voltar".
	5. O usuário clica em voltar e o sistema retorna a tela anterior.
Fluxo Alternativo	Não há.

A Figura 10 representa a seleção de disciplina, através do caso de uso de acesso ao mural.



Figura 10 - Mural (Seleção de Disciplina)

A Figura 11 representa o acesso ao mural propriamente dito, o qual é realizado após a seleção da disciplina, conforme detalhado no caso de uso narrativo.



Figura 11 - Mural

Após o acesso o mural, o aluno seleciona o recado que deseja ler, sendo exibida a mensagem do mesmo. A interface de exibição desta mensagem ocorre conforme a interface exemplificada na Figura 12.

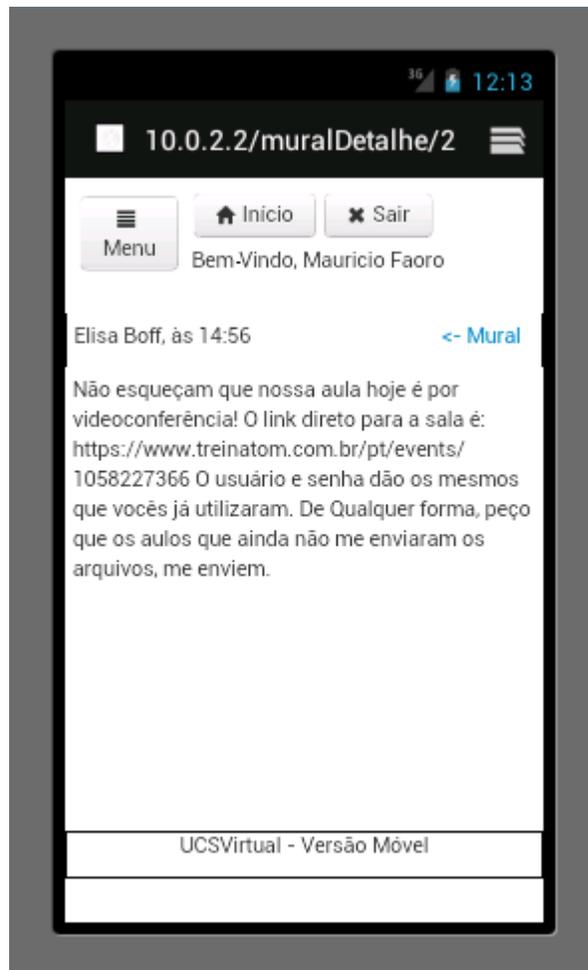


Figura 12 - Visualização de Recado no Mural

Na Tabela 8 pode ser verificado o caso de uso narrativo referente ao acesso ao cronograma das disciplinas.

Tabela 8 – UC04 – Acessar Cronograma

Identificador:	UC04 - Acessar Cronograma
Descrição:	Acesso ao cronograma.
Ator(res) Primário(s):	Aluno
Pré - Condições:	UC02 - Interface Principal
Fluxo Principal:	<ol style="list-style-type: none"> 1. O Sistema exibe tela para seleção da disciplina que o aluno deseja consultar o cronograma. (OBS: São consideradas somente as disciplinas do semestre corrente) 2. O Aluno seleciona a disciplina.

	3. O Sistema exibe a relação de data e número de aula (trazendo pré-selecionada, a aula que corresponderá à próxima)
	4. O Aluno clica na aula que deseja visualizar o cronograma.
	5. O Sistema exibe o cronograma da aula (com rolagem, para caso o cronograma for muito extenso) e opção de voltar e também opção para acessar as orientações da aula.
	6. O aluno clica em "Orientações da Aula", são abertas em nova tela as orientações da aula com a opção de voltar.
Fluxo Alternativo:	2. O Aluno não está matriculado em nenhuma disciplina. O sistema informa mensagem do ocorrido.

A Figura 13 representa a seleção de disciplina, através do caso de uso de acesso ao cronograma.



Figura 13 - Cronograma (Seleção de Disciplina)

A Figura 14 representa o acesso ao cronograma propriamente dito, o qual é realizado após a seleção da disciplina, conforme detalhado no caso de uso narrativo.

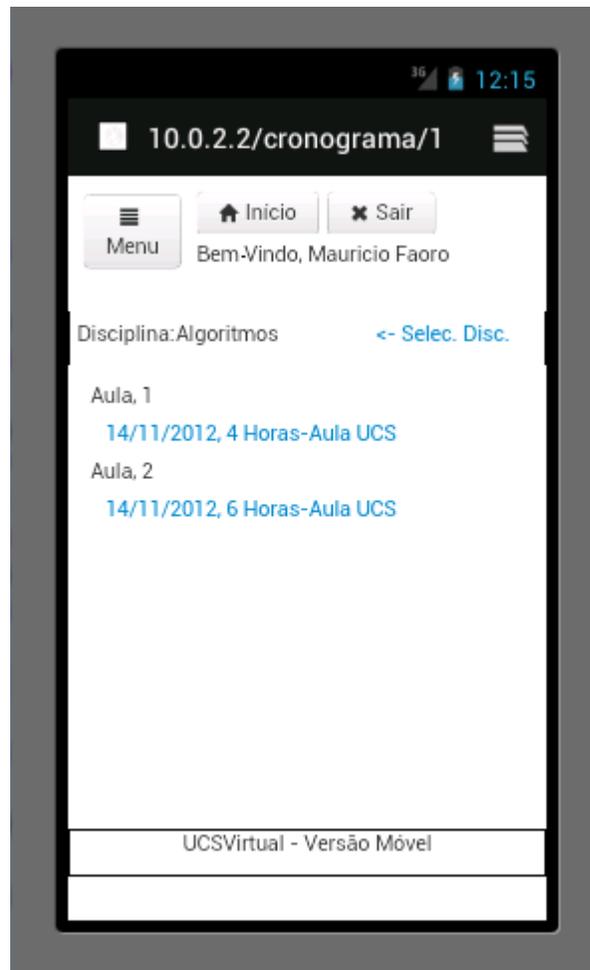


Figura 14 - Acesso ao Cronograma

Após o acesso ao cronograma, o aluno seleciona a aula que deseja verificar, sendo exibido o cronograma da mesma. A interface de exibição do cronograma ocorre conforme exemplificado na Figura 15.



Figura 15 - Exibição do Cronograma

Conforme detalhado no caso de uso narrativo de acesso ao cronograma, após a verificação do cronograma da aula, o aluno pode verificar as “Orientações da Aula”. As orientações da aula serão exibidas conforme o layout exemplificado na Figura 16.



Figura 16 - Orientações da Aula

Para demonstrar a evolução do processo de desenvolvimento deste aplicativo, pode ser verificado no Anexo II, os protótipos de interface desenvolvidos na primeira parte do trabalho que originarem as interfaces acima exibidas.

Na Figura 17 podemos verificar o diagrama de classe utilizado para desenvolvimento deste protótipo.

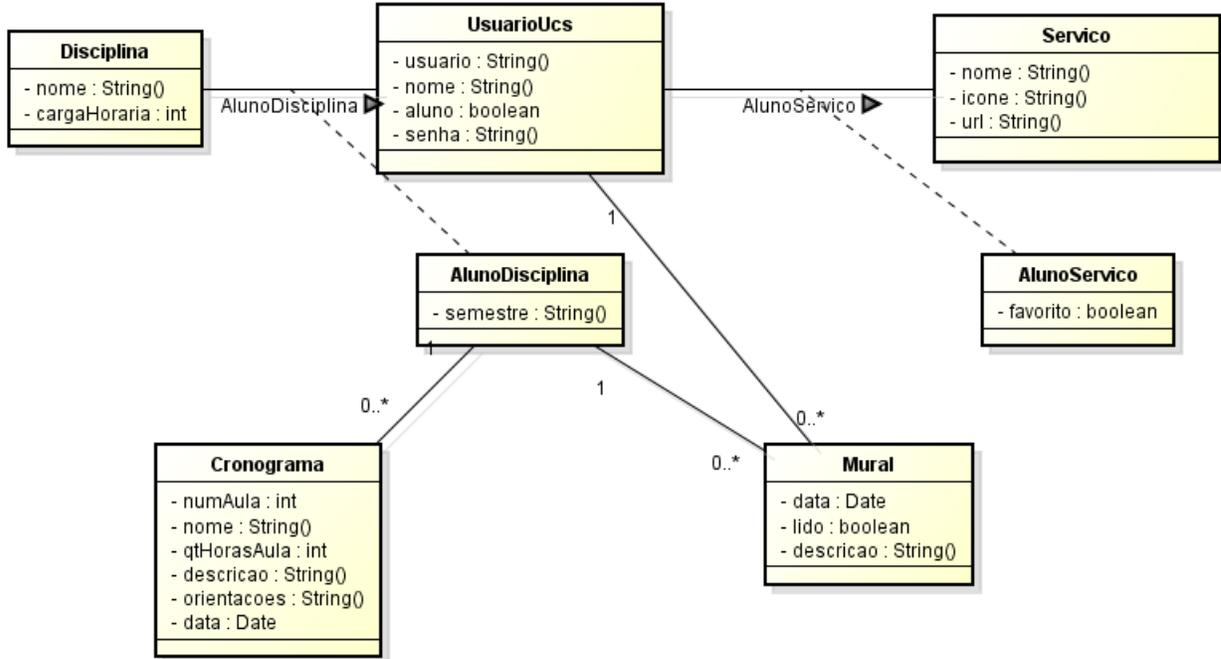


Figura 17 – Diagrama de Classes

Já na Figura 18 podemos verificar o modelo entidade relacional desenvolvido para tratar este desenvolvimento.

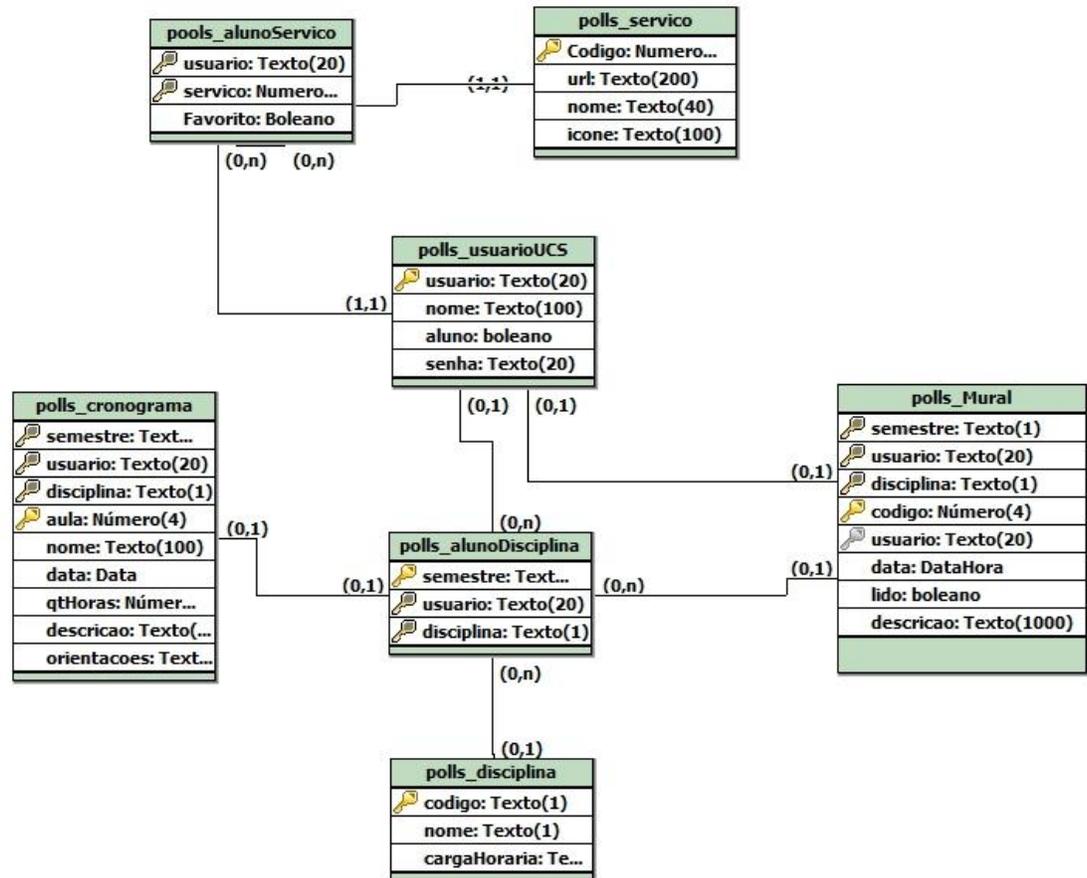


Figura 18 – Modelo ER

A tabela *polls_usuarioUCS*, é responsável por armazenar o nome do usuário, login, senha de acesso e se o mesmo é aluno.

A tabela *polls_servico*, armazena as informações referentes aos serviços disponibilizados no aplicativo, onde possuímos o nome do serviço, o caminho onde está disponível o ícone do serviço e a URL para acessar o mesmo.

A tabela *polls_alunoServico*, é responsável por armazenar os serviços que o aluno tem permissão de acesso no aplicativo, onde o campo favorito define se o mesmo deve ser exibido com seu ícone na interface principal do aplicativo.

A tabela *polls_disciplinas* armazena as disciplinas oferecidas e seu semestre, enquanto a tabela *polls_alunoDisciplina* faz o vínculo do aluno com a disciplina.

Enquanto a tabela *polls_mural* armazena as informações do mural vinculando a um aluno/disciplina e um remetente, a tabela cronograma armazena as informações do *polls_cronograma* vinculando a um aluno/disciplina.

5.2 ARQUITETURA DO PROTÓTIPO

Este protótipo foi desenvolvido utilizando as tecnologias HTML5 e CSS3 para a camada de aplicação e Python com framework Django na camada de persistência. O banco de dados utilizado foi o SQLite3.

A escolha de linguagem para a camada de persistência ocorreu pelo fato de a Universidade de Caxias do Sul utilizar Python com Django na aplicação atual do UCSVirtual e pretender continuar com estas linguagens para desenvolvimentos futuros. Já o banco de dados SQLite3 foi escolhido devido ao fato de o mesmo possuir fácil integração com o Python e por se tratar de um protótipo.

A arquitetura de desenvolvimento do protótipo é exemplificada pela Figura 19:

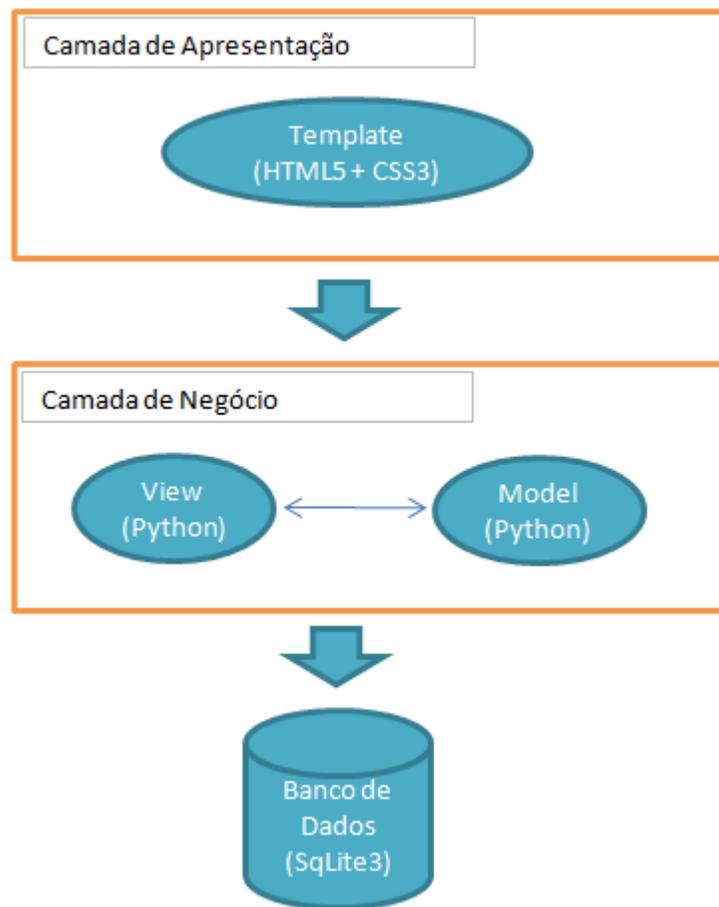


Figura 19 – Arquitetura do Protótipo

O padrão de desenvolvimento utilizado neste projeto foi MVT (*Model View Template*). Padrão de desenvolvimento para projetos que utilizam linguagem Python.

Este padrão consiste em definir as estruturas das classes de persistência no *Model*, na *View* realizar a lógica de negócio e no *Template* definir a estrutura HTML para apresentação ao usuário final.

A interface de acesso ao menu principal da aplicação foi desenvolvida como segue.

No *Model*, são definidas as estruturas de dados utilizadas no aplicativo. No caso do acesso a interface principal do sistema, precisamos das classes que armazenam o usuário, os serviços e os serviços relacionados ao usuário. Estas classes estão definidas na Figura 20.

```

class UsuarioUCS(models.Model):
    usuarioUCS = models.CharField(max_length=20)
    nome = models.CharField(max_length=100)
    senha = models.CharField(max_length=20)
    aluno = models.BooleanField()

    def __unicode__(self):
        return self.usuarioUCS

class Servico(models.Model):
    nome = models.CharField(max_length=100)
    icone = models.CharField(max_length=200)
    url = models.CharField(max_length=300)
    def __unicode__(self):
        return self.nome

class AlunoServico(models.Model):
    usuarioUCS = models.ForeignKey(UsuarioUCS)
    servico = models.ForeignKey(Servico)
    favorito = models.BooleanField()
    def servicos_do_aluno(username):
        return self.usuarioUCS.usuarioUCS == username

```

Figura 20 – Model Aplicativo UCSVirtual Móvel

Na *View* realizamos acesso aos dados do *Model* e chamamos as interfaces HTML5 do *Template* para serem exibidas, para tanto possuímos o código fonte abaixo exemplificado na Figura 21.

```

def index(request):
    if request.POST:
        username = request.POST['username']
        password = request.POST['password']
        user = authenticate(username=username, password=password)
        if user is not None:
            login(request, user)
        else:
            messages.error(request, 'Usuario ou Senha incorretos!')
    if not request.user.is_authenticated():
        return render_to_response('pools/index.html', context_instance=RequestContext(request))

    usuario = UsuarioUCS.objects.get(usuarioUCS=request.user.username)
    servicos_disponiveis = AlunoServico.objects.filter(usuarioUCS=usuario, favorito='true')
    return render_to_response('pools/appMain.html', {'nome': request.user.get_full_name(),
    'servicos' : servicos_disponiveis}, context_instance=RequestContext(request))

```

Figura 21 – Trecho código fonte *View*

No *Template* está contida toda a estrutura HTML5 do aplicativo. O arquivo de *template* que representa a tela principal do protótipo do UCSVirtual móvel é o arquivo “appMain.html”. Este arquivo contém todo o código HTML e também a tratativa para exibição dos dados recebido do Python, no caso nome do usuário e serviços favoritos do mesmo. A tratativa para exibição dos dados recebidos do Python ocorre conforme a Figura 22.

```

<div style=" height: 50%;"> Bem Vindo, {{ nome }} </div>
</div>
</header>

<section style="border: 1px black solid; padding: 5px" align="none">
  Exibir serviços: <a href="#" style="padding: 10px" > Favoritos </a> <a href="#" style="padding: 10px"> Recentes </a>
</section>

<section id="container_demo" style="height: 100%;">

  <div style=" min-height: 300px; padding: 5px 5px 5px 5px; " >

    {% if servicos %}
    <ul style=" list-style-type: none; float: left; width: 100%; " >
      {% for servico in servicos %}
      <li style="float: left; margin: 5px; width: 45%; height: 100px; text-align: center;">

        <figure>
          <a href=" {{ servico.servico.url }}" />
          
          <figcaption>{{ servico.servico.nome }}</figcaption>
        </figure>
      </li>
      {% endfor %}
    </ul>
    {% endif %}
  </div>
</section>

```

Figura 22 – Código fonte Python no *Template* (HTML5)

O código fonte completo da aplicação consta no Anexo III.

6 – AVALIAÇÃO DA INTERFACE DESENVOLVIDA

Conforme verificamos ao longo deste trabalho, no desenvolvimento de aplicações sejam elas voltadas para desktop, Internet ou dispositivos móveis, a realização de avaliação da interface desenvolvida é fundamental para minimizar os erros e garantir uma melhor experiência de uso na aplicação.

No protótipo da versão móvel do UCSVirtual realizamos a avaliação da interface desenvolvida. O método de avaliação utilizado foi a “Avaliação Heurística”, já descrita na seção 2.3.2, onde podemos observar os itens de usabilidade tradicionais e para dispositivos móveis que estão contemplados nesta aplicação.

Abaixo realizamos a análise de cada uma das interfaces desenvolvidas, destacando os princípios de usabilidade.

Começamos pela interface de login, representada pela Figura 6 do Capítulo 5, onde podemos observar:

- A funcionalidade de login ocupa toda a interface da aplicação, os ícones são grandes, e existe um bom espaçamento, o que garante facilidade de acesso a mesma através do toque, conforme indicam as Leis de Fitt - Preece et al. (2005).
- Existe *hiperlink* que permite ao usuário acessar a versão tradicional do UCSVirtual caso necessite acessar funcionalidades que ainda não estão contempladas na versão móvel.
- Para o campo de usuário existe a função de auto-completar, o que reduz o esforço de digitação do login do mesmo.
- A aplicação mantém o usuário conectado, até que o mesmo realize o *logout*, desta maneira simplificando o acesso a aplicação, não sendo necessário realizar o *login*, toda a vez que se deseja acessar a aplicação.
- Quando o usuário errar o “usuário” ou a “senha” digitados, o mesmo pode clicar na opção de “limpar” não sendo necessário apagar manualmente as informações.
- Caso o usuário digite “usuário” ou “senha” incorretamente e tente realizar login na aplicação é retornada mensagem destacada avisando do ocorrido, como pode ser verificado na Figura 7 do Capítulo 5.

Após interface principal do sistema é representada pela Figura 8 do Capítulo 5, na qual podemos observar os seguintes itens:

- A interface é exibida com atalhos para os serviços definidos como favoritos pelo usuário (exemplo: AVA – Mural e AVA – Cronograma), desta forma permitindo personalização da tela inicial.
- Os ícones de acesso às funcionalidades, possuem tamanho relativamente grande e bom espaçamento, o que facilita o acesso pelo toque.
- Existe opção para exibir as atualizações recentes, facilitando o acesso às informações disponíveis no aplicativo.
- Caso o usuário deseje acessar serviços, que não estão definidos como favoritos, existe botão de acesso ao menu no topo da tela.
- Os ícones utilizados para acesso ao menu, sair e tela inicial, bem como sua localização, são semelhantes aos existentes em aplicações populares para dispositivos móveis, o que torna o acesso a este aplicativo muito intuitivo.

Na interface de menu, representada pela Figura 9 do Capítulo 5 podemos observar:

- Os ícones para acesso ao menu, tela inicial e sair, também são disponibilizados nesta tela, fazendo com que o usuário possa voltar a tela inicial (favoritos e recentes) ou sair do sistema a qualquer momento e a partir da funcionalidade que está acessando.
- A seleção de serviços do portal possui campo para busca, desta maneira tornando mais flexível a utilização da interface por usuários avançados.
- O espaçamento entre os campos da interface garante facilidade no acesso via toque.

O serviço de seleção de disciplinas do cronograma é representado pela Figura 13 do Capítulo 5, nela podemos observar:

- Os ícones para acesso ao menu, tela inicial e sair, também são disponibilizados nesta tela, fazendo com que o usuário possa voltar a tela inicial (favoritos e recentes), acessar novamente o menu ou sair do sistema a qualquer momento e a partir da funcionalidade que está acessando.

- A seleção da disciplina do cronograma possui campo para busca, desta maneira tornando mais flexível a busca da disciplina que o usuário deseja visualizar o cronograma.
- O espaçamento entre os campos da tela garante facilidade no acesso via toque.

A Figura 14 do Capítulo 5 representa a seleção da aula do cronograma, nesta podemos observar:

- Os ícones para acesso ao menu, tela inicial e sair, também são disponibilizados nesta tela, fazendo com que o usuário possa voltar a tela inicial (favoritos e recentes), acessar novamente o menu ou sair do sistema a qualquer momento e a partir da funcionalidade que está acessando.
- É exibido abaixo dos botões de acesso ao menu, tela inicial e sair, o nome da disciplina selecionada e o botão troca de disciplina, desta forma, permitindo ao usuário voltar para a tela de seleção de disciplina, caso tenha selecionado a disciplina incorreta.
- O espaçamento entre os campos da tela garante facilidade no acesso via toque.

A Figura 15 do Capítulo 5 representa o conteúdo do cronograma da aula selecionada na tela anterior, podemos observar nesta:

- Os ícones para acesso ao menu, tela inicial e sair, também são disponibilizados nesta tela, fazendo com que o usuário possa voltar a tela inicial (favoritos e recentes), acessar novamente o menu ou sair do sistema a qualquer momento e a partir da funcionalidade que está acessando.
- É exibido abaixo dos botões de acesso ao menu, tela inicial e sair, o nome da aula selecionada, data da aula e botão voltar, desta forma, permitindo ao usuário voltar para a tela de seleção de aula do cronograma, caso tenha selecionado a aula incorreta.
- O espaçamento entre os campos da tela garante facilidade no acesso via toque.
- No canto inferior esquerdo da tela existe link para acesso as orientações da aula. O link está posicionado ao lado esquerdo, para que o usuário não clique no mesmo acidentalmente quando está rolando a informação do cronograma da aula.

Na Figura 16 do Capítulo 5 são exibidas as orientações da aula selecionada, nesta podemos observar:

- Os ícones para acesso ao menu, tela inicial e sair, também são disponibilizados nesta tela, fazendo com que o usuário possa voltar a tela inicial (favoritos e recentes), acessar novamente o menu ou sair do sistema a qualquer momento e a partir da funcionalidade que está acessando.
- É exibido abaixo dos botões de acesso ao menu, tela inicial e sair, o nome da aula selecionada, data da aula e botão voltar, desta forma, permitindo ao usuário voltar para a tela anterior, caso deseje.

A Figura 10 do Capítulo 5, representa a seleção de disciplinas do mural, onde pode-se observar:

- Os ícones para acesso ao menu, tela inicial e sair, também são disponibilizados nesta tela, fazendo com que o usuário possa voltar a tela inicial (favoritos e recentes), acessar novamente o menu ou sair do sistema a qualquer momento e a partir da funcionalidade que está acessando.
- A seleção da disciplina do mural possui campo para busca, desta maneira tornando mais fácil a busca da disciplina que o usuário deseja visualizar o cronograma.
- O espaçamento entre os campos da tela garante facilidade no acesso via toque.

A Figura 11 do Capítulo 5 representa a interface de seleção de recado no mural, onde podemos observar:

- Os ícones para acesso ao menu, tela inicial e sair, também são disponibilizados nesta tela, fazendo com que o usuário possa voltar a tela inicial (favoritos e recentes), acessar novamente o menu ou sair do sistema a qualquer momento e a partir da funcionalidade que está acessando.
- É exibido abaixo dos botões de acesso ao menu, tela inicial e sair, o nome da disciplina selecionada e o botão troca de disciplina, desta forma, permitindo ao usuário voltar para a tela de seleção de disciplina, caso tenha selecionado a disciplina incorreta, fornecendo o controle da navegação ao usuário.

- Nos itens do mural, entre parênteses, é exibida se o recado já foi lido pelo usuário, desta maneira facilitando o acesso ao recados não lidos.
- O espaçamento entre os campos da tela garante facilidade no acesso via toque.

A Figura 12 do Capítulo 5 representa o recado do item de mural selecionado, nesta interface podemos observar:

- Os ícones para acesso ao menu, tela inicial e sair, também são disponibilizados nesta tela, fazendo com que o usuário possa voltar a tela inicial (favoritos e recentes), acessar novamente o menu ou sair do sistema a qualquer momento e a partir da funcionalidade que está acessando.
- É exibido abaixo dos botões de acesso ao menu, tela inicial e sair, o nome do remetente do recado no mural, hora e botão voltar, desta forma, orientando e facilitando a navegação com possibilidade de voltar para a tela de seleção de recado do mural, caso tenha selecionado a aula incorreta.
- O espaçamento entre os campos da tela garante facilidade no acesso via toque.

Este capítulo teve o intuito de apresentar como os princípios de usabilidade foram considerados no projeto desta aplicação e como eles podem melhorar a experiência do usuário na sua utilização.

7 - CONSIDERAÇÕES FINAIS

O desenvolvimento de versões móveis para sites e portais vem se tornando uma necessidade para as empresas, visto a grande adesão de *tablets* e *smartphones*. Esta também é uma necessidade da UCS com o Portal de Serviços UCSVirtual.

Porém quando pensado em uma aplicação móvel deve-se ter foco na usabilidade, construindo a interface de modo que estimule o usuário a utilizá-la. Para desenvolvimento de uma aplicação é interessante seguir as boas práticas de usabilidade no desenvolvimento, além dos guias de referências para dispositivos móveis, onde questões como tamanho de tela reduzida e utilização do *touchscreen* devem ser pensadas desde o início do desenvolvimento.

Outro ponto a ser analisado é a escolha da tecnologia a ser utilizada. Quando o aplicativo necessita de recursos de hardware ou sistemas mais avançados é interessante a utilização de aplicativos nativos, enquanto quando não há necessidade de utilização destes recursos é mais adequado o desenvolvimento de um WEB APP.

No caso do UCSVirtual é mais adequado o desenvolvimento de um WEB APP. Neste trabalho foi desenvolvido um WEB APP utilizando as tecnologias HTML5 e CSS3 na camada de apresentação e Python para realização de persistências. A escolha destas linguagens se deu principalmente ao requisito de o aplicativo ser multiplataforma, ou seja, funcionar em Android, iOS, Windows Phone e outros, e também pela performance, pois o aplicativo desenvolvido realiza apenas leituras e persistências no banco de dados através da WEB, sendo mais rápido este tipo de acesso através de um APP WEB do que de um APP nativo.

A escolha dos serviços priorizados nesta implementação da versão móvel do portal UCSVirtual foi apoiada pela pesquisa realizada junto aos usuários. Esta pesquisa fez parte da etapa de Análise do processo de desenvolvimento, especificamente para o levantamento de requisitos, ou seja, as funcionalidades de Cronograma e Mural do AVA do UCSVirtual.

Após o término do desenvolvimento do protótipo da versão móvel do UCSVirtual realizamos a avaliação heurística da interface, onde podemos observar que vários itens das boas práticas de usabilidade tradicional e para dispositivos móveis foram respeitados durante este desenvolvimento.

As principais contribuições trazidas por este trabalho para a comunidade acadêmica relacionam-se com a escolha de tecnologia para desenvolvimento móvel, que permite portabilidade e mobilidade, cada caso deve ser analisada a melhor alternativa, a que trará melhor relação custo benefício em sua escolha. Como verificamos no decorrer do Capítulo 3, existem aplicativos, cujo o paradigma de desenvolvimento nativo é o mais adequado enquanto em outros é o paradigma de desenvolvimento WEB. Este trabalho também trouxe contribuições na questão de usabilidade, trazendo boas práticas que devem ser utilizadas quando realizado um desenvolvimento móvel, pensando principalmente na adequação dos layouts a um tamanho de tela reduzido e uma interação e navegação através do toque.

REFERENCIAS BIBLIOGRÁFICA

ANDROID DEVELOPER – Design - Google. Disponível em : <http://developer.android.com/design/index.html>. Acessado em 20 de junho, 2012.

Android: um novo paradigma de desenvolvimento móvel - Ramon Ribeiro Rabello. Disponível em: http://www.cesar.org.br/site/files/file/WM18_Android.pdf. Acessado em 4 de junho, 2012.

APPLE DEVELOPER – Apple. Disponível em: <https://developer.apple.com/>. Acessado em 15 de junho, 2012

Apps vs the Web - Matt Gemmell. Disponível em: <http://mattgemmell.com/2011/07/22/apps-vs-the-web/>. Acessado em 02 de junho, 2012.

BLACKBERRY DEVELOPER (HTML5) – RIM. Disponível em: <https://developer.blackberry.com/html5/>. Acessado em 02 de junho, 2012.

BREWSTER, Stephen A., CRYER, Peter G. **Maximising Screen-Space on Mobile Computing Devices**. 1999.

ComScore Reports April 2012 U.S. Mobile Subscriber Market Share – Comscore. Disponível em: http://www.comscore.com/Press_Events/Press_Releases/2012/6/comScore_Reports_April_2012_U.S._Mobile_Subscriber_Market_Share. Acessado em 7 de junho, 2012.

Desenvolvimento de Aplicações para Plataforma Google Android - Fábio de Jesus Lima Gomes, Manoel Taenan Ferreira de Souza, Rafael Madureira Lins de Araújo. Disponível em: <http://www.die.ufpi.br/ercemapi2011/minicursos/MC3.pdf>. Acessado em 02 de junho, 2012.

Ipad usability (2nd-edition) – NMGroup. Disponível em: http://www.nngroup.com/reports/mobile/ipad/ipad-usability_2nd-edition.pdf. Acessado em 25 de abril, 2012.

Jason Cranford Teague. **CSS3: Visual QuickStart Guide**. 5ª Edição. Editora - Pearson Education, 2010.

Mobile Usability Update - Jakob Nielsen. Disponível em: <http://www.useit.com/alertbox/mobile-usability.html>. Acessado em 20 de abril, 2012.

Mobile Web App vs. Native App? It's Complicated – Forbes. Disponível em: <http://www.forbes.com/sites/fredcavazza/2011/09/27/mobile-web-app-vs-native-app-its-complicated/>. Acesso em 16 de abril, 2012.

Mobile Web Apps vs Native Apps - Elliott Lemenager. Disponível em: <http://www.elliottlemenager.com/2012/02/09/mobile-web-apps-vs-native-apps/>. Acessado em 25 de maio, 2012.

Mobile Web vs. Native Apps. Revisited – Mobiletech. Disponível em <http://www.mobiletech.mobi/mobile-web-vs-native-apps-revisited/>. Acessado em 10 de abril, 2012.

One Billion HTML5 Phones to be Sold Worldwide in 2013 - Strategy Analytics. Disponível em: <http://www.strategyanalytics.com/default.aspx?mod=pressreleaseviewer&a0=5145>. Acessado em 06 de junho, 2012.

Phone Wars: Mobile Apps vs. Native Apps - Selena Narayanasamy. Disponível em: <http://www.blueglass.com/blog/phone-wars-mobile-web-apps-vs-native-apps/>. Acessado em 15 de maio, 2012.

Preece, J., Rogers, Y., Sharp, . **Design de interação : além da interação homem-computador**. 1ª Edição. Editora: ARTMED, 2005.

Silva, Maurício Samy. **HTML5 - a Linguagem da Marcação Que Revolucionou a Web**. Editora Novatec, 2011.

Técnicas de Prototipação para Smartphones no Apoio à Avaliação de Interfaces com o Usuário - Talita C. P. Britto, Livia C. G. Martins, Bruno E. Pentead, Junia C. Anacleto. Disponível em: <http://pt.scribd.com/doc/85653539/Prototipacao-de-Interfaces-para-Dispositivos-Moveis>. Acessado em 20 de maio, 2012.

W3C - Web Recommendation - Mobile Web Best Practices 1.0 – W3C. Disponível em: <http://www.w3.org/TR/mobile-bp/>. Acessado em 8 de maio, 2012.

W3C SCHOOLS (HTML5) – W3C. Disponível em: <http://www.w3schools.com/html5/default.asp>. Acessado em 15 de Junho, 2012.

Web Apps vs. Native Apps - Michael Schneider (Mobile Roadie CEO) Disponível em: <http://blog.mobileroadie.com/2010/01/web-apps-vs-native-apps/>. Acessado em 12 de maio, 2012.

APÊNDICE A – Pesquisa Realizada junto aos usuários do UCSVirtual para identificação das funcionalidades prioritárias para desenvolvimento de uma versão móvel do portal.

Questionário:

1. Você utiliza o UCSVirtual?

- Sim
- Não

2. Com qual frequência você utiliza o UCSVirtual?

- Pelo menos uma vez por semana
- Mais de uma vez por semana
- Todos os dias
- Somente no início e fim do semestre

3. Quais os serviços disponíveis no UCS Virtual que você mais utiliza?

- Atividades Complementares
- Avaliação On-line
- Boletim Matrícula Graduação
- Comprovante de Pagamentos
- Documentos - Graduação
- Estudo Currículo Graduação
- Gerenciador Financeiro
- Histórico - Graduação
- Histórico de Impressões
- Komunikas
- Matrícula Graduação
- Registro de Aproveitamento
- AVA Graduação - Mural
- AVA Graduação - Cronograma das Aulas
- AVA Graduação - Webfólio
- AVA Graduação - Acervo

- AVA Graduação - Lista de Participantes
- AVA Graduação - Informações da Turma
- AVA Graduação - Fórum
- Mensagens Instantâneas

4. Você possui smartphone ou tablet?

- Sim
- Não

5. Você costuma acessar a Internet através de seu tablet ou smartphone?

- Sim
- Não

6. Você acessa a versão existente do UCSVirtual no seu dispositivo móvel?

- Sim
- Não

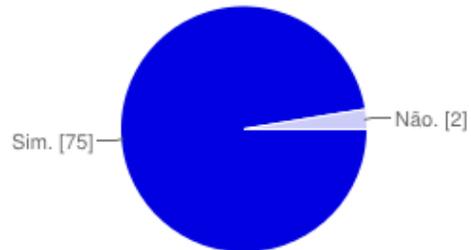
7. Você acessa a versão existente do UCSVirtual no seu dispositivo móvel?

- Sim
- Não

8. Quais serviços você gostaria que fossem disponibilizados em uma versão móvel do UCSVirtual?

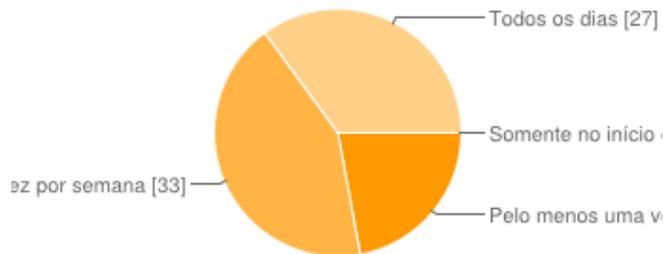
ANEXO I – Resultados da pesquisa realizada junto aos usuários do UCSVirtual.

Você utiliza o UCSVirtual?



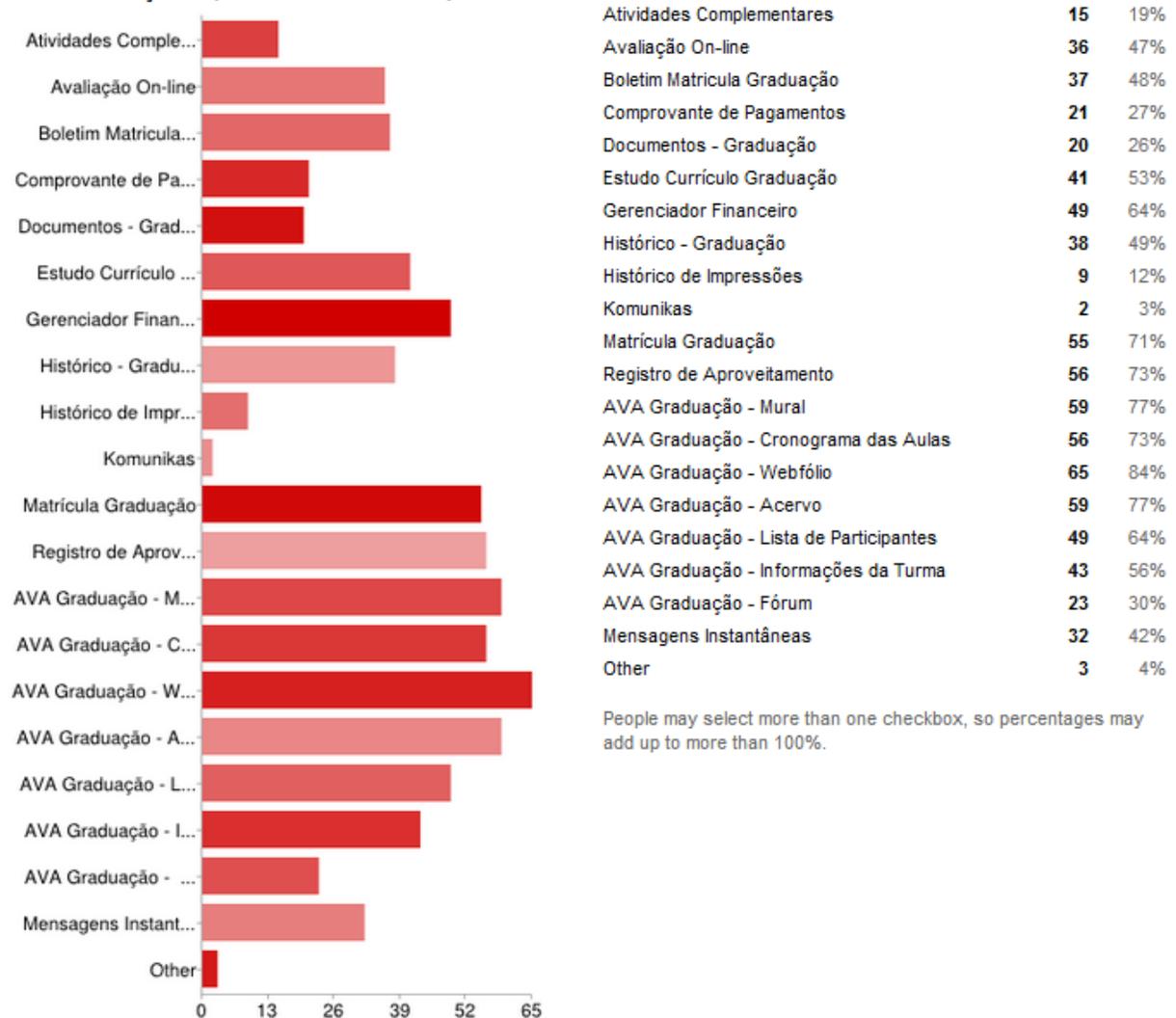
Sim.	75	95%
Não.	2	3%

Com qual frequência você utiliza o UCSVirtual?

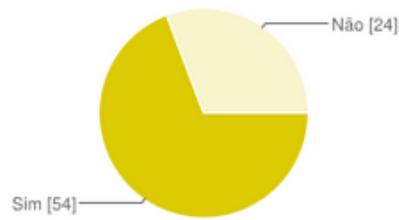


Pelo menos uma vez por semana	17	22%
Mais de uma vez por semana	33	42%
Todos os dias	27	34%
Somente no início e fim do semestre	0	0%

Quais os serviços disponíveis no UCS Virtual que você mais utiliza?

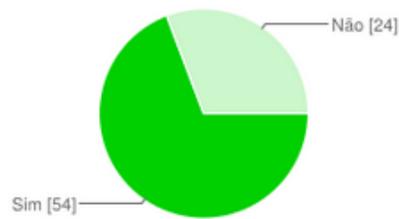


Você possui smartphone ou tablet?



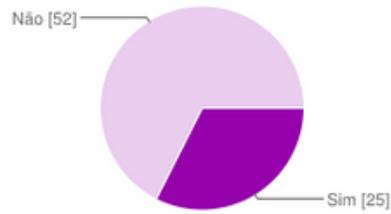
Sim	54	68%
Não	24	30%

Você costuma acessar a Internet através de seu tablet ou smartphone?



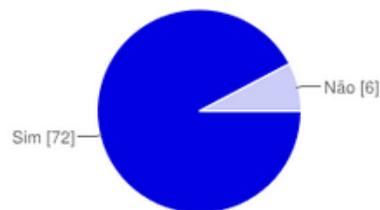
Sim	54	68%
Não	24	30%

Você acessa a versão existente do UCSVirtual no seu dispositivo móvel?



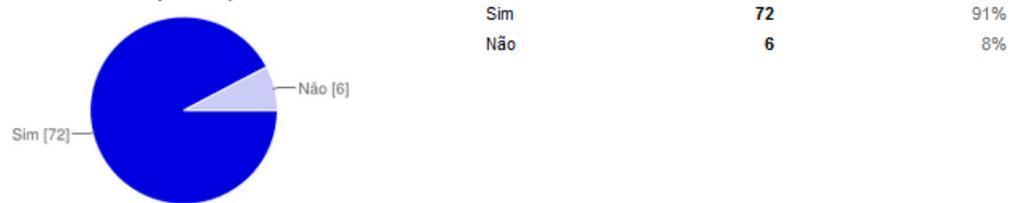
Sim	25	32%
Não	52	66%

Se existisse uma versão para dispositivos móveis do UCSVirtual você utilizaria?



Sim	72	91%
Não	6	8%

Se existisse uma versão para dispositivos móveis do UCSVirtual você utilizaria?



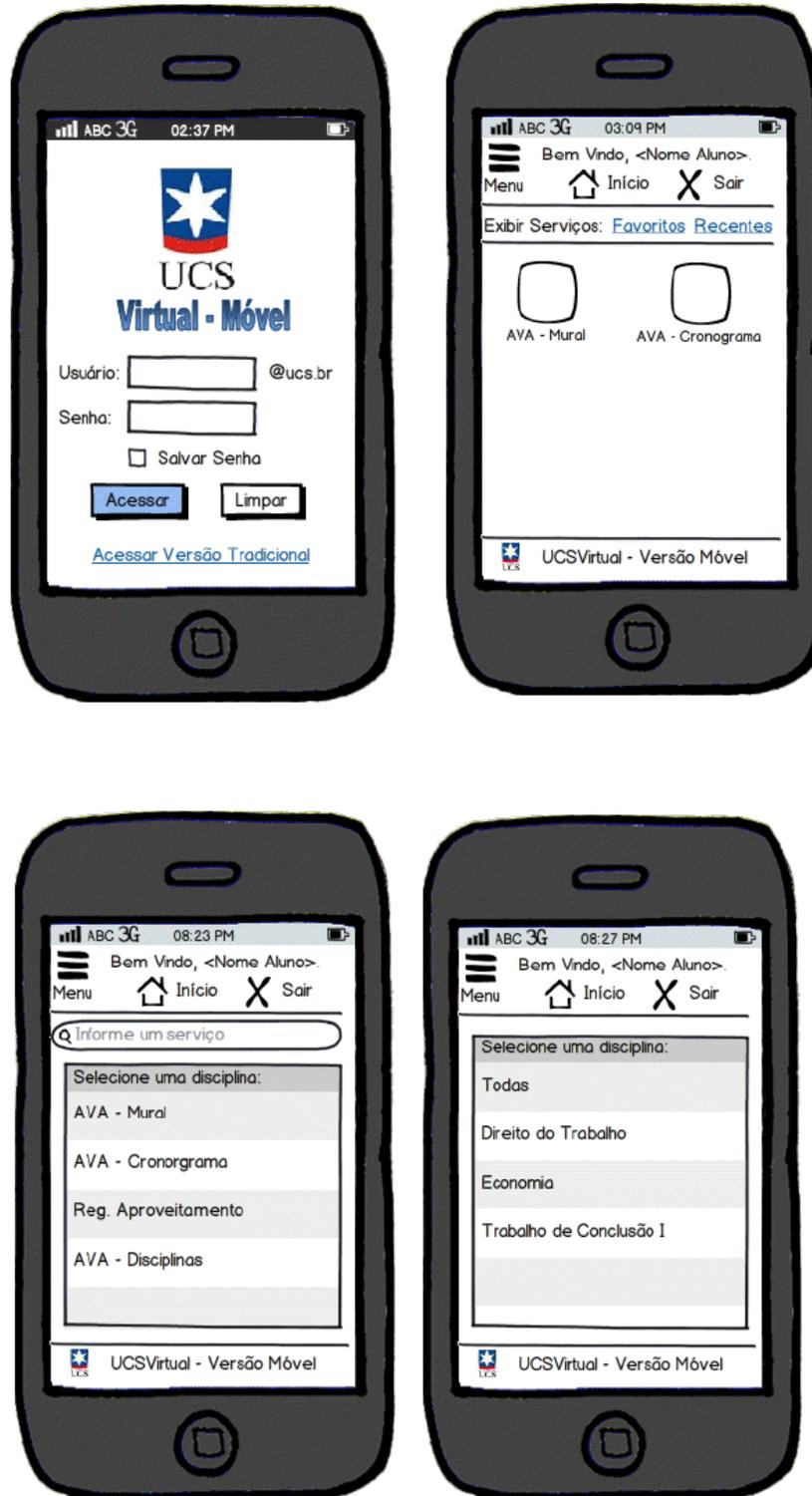
Quais serviços você gostaria que fossem disponibilizados em uma versão móvel do UCSVirtual?

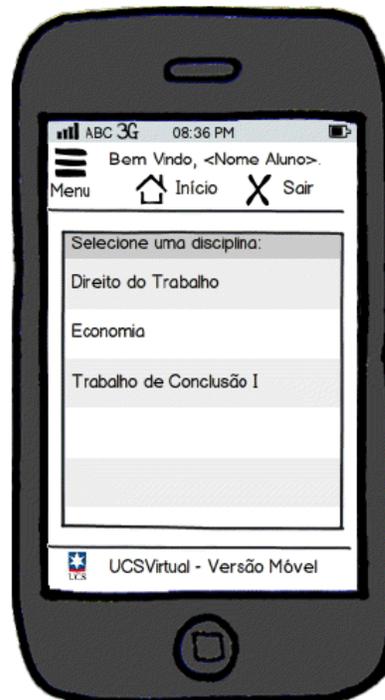
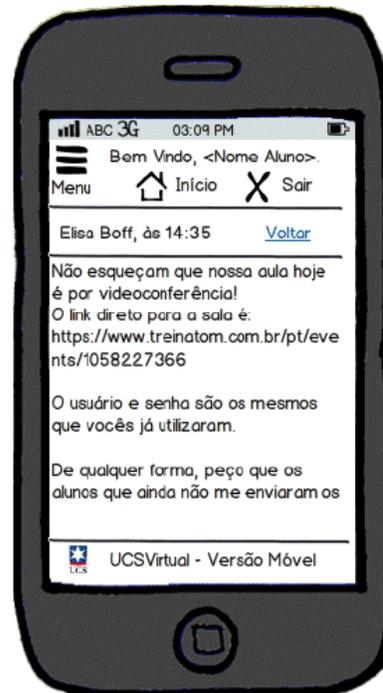
Registro de Aproveitamento, Localização de Salas. As funcionalidades que mais seriam úteis para mim seriam o Registro de Aproveitamento, Mural e Cronograma. Localização de salas, Liberação de Vaga em Espera, Forum, Mural, Cronograma, Webfólio, Acervo. AVA mural, cronograma, acervo, lista de participantes, informações da turma. Os serviços que oferecem informações como Mural e Cronograma são mais importantes. Matrícula e webfolio eu evitaria de usar de um aparelho móvel. AVA Graduação, Sistema de Matrícula e Mensagens instantâneas. Mural, cronograma, acervo, localização de salas, mensagens e matricul ...

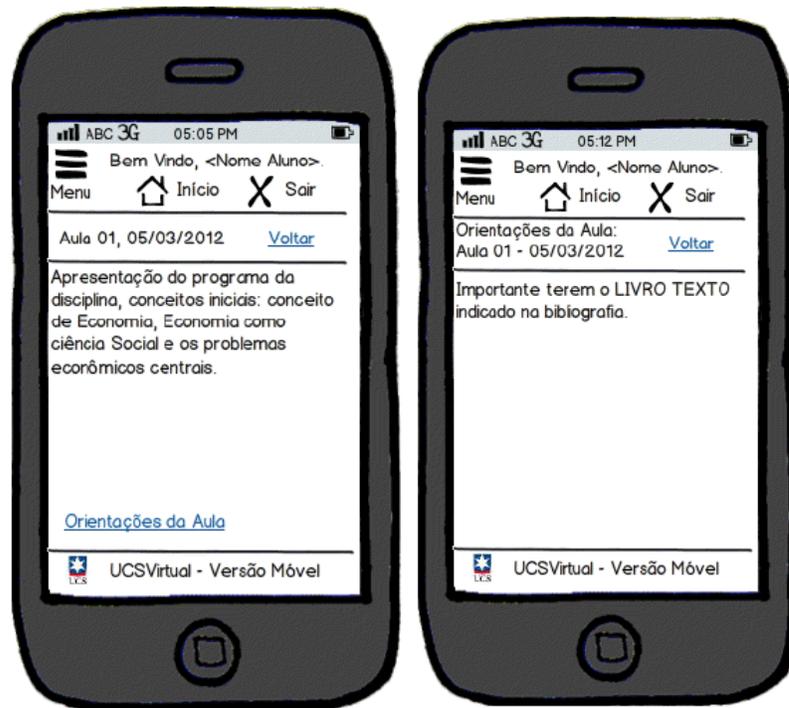
Number of daily responses



ANEXO II – Protótipos de Interface TCC I







ANEXO III – Código fonte completo da aplicação

View

```

from django.shortcuts import render_to_response
from django.contrib.auth import authenticate, login, logout
from django.template import RequestContext
from django.contrib import messages
from django.http import HttpResponseRedirect
from ucsvitualmobile.polls.models import *

def index(request):
    if request.POST:
        username = request.POST['username']
        password = request.POST['password']
        user = authenticate(username=username, password=password)
        if user is not None:
            login(request, user)
        else:
            messages.error(request, 'Usuario ou Senha incorretos!')
    if not request.user.is_authenticated():
        return render_to_response('pools/index.html',
context_instance=RequestContext(request))

    usuario = UsuarioUCS.objects.get(usuarioUCS=request.user.username)
    servicos_disponiveis = AlunoServico.objects.filter(usuarioUCS=usuario, favorito='true')
    return render_to_response('pools/appMain.html', {'nome': request.user.get_full_name(),
'servicos' : servicos_disponiveis}, context_instance=RequestContext(request))

def sair(request):
    logout(request)
    messages.success(request, 'Logout realizado com sucesso!')
    return HttpResponseRedirect('/')

def appMain(request):
    usuario = UsuarioUCS.objects.get(usuarioUCS=request.user.username)
    servicos_disponiveis = AlunoServico.objects.filter(usuarioUCS=usuario)
    return render_to_response('pools/appMain.html', {'nome': request.user.get_full_name(),
'servicos' : servicos_disponiveis}, context_instance=RequestContext(request))

def selecaoDisciplinaMural(request):
    usuario = UsuarioUCS.objects.get(usuarioUCS=request.user.username)
    disciplinas = AlunoDisciplina.objects.filter(usuarioUCS=usuario)
    return render_to_response('pools/selecaoDisciplinaMural.html', {'nome':
request.user.get_full_name(), 'disciplinas' : disciplinas},
context_instance=RequestContext(request))

def todosServicos(request):
    usuario = UsuarioUCS.objects.get(usuarioUCS=request.user.username)
    servicos_disponiveis = AlunoServico.objects.filter(usuarioUCS=usuario)
    return render_to_response('pools/todosServicos.html', {'nome':
request.user.get_full_name(), 'servicos' : servicos_disponiveis},
context_instance=RequestContext(request))

def mural(request, poll_name):
    usuario = UsuarioUCS.objects.get(usuarioUCS=request.user.username)
    disciplina_pool = Disciplina.objects.get(nome=poll_name)
    disciplina = AlunoDisciplina.objects.get(usuarioUCS=usuario, disciplina=disciplina_pool)
    mural = Mural.objects.filter(alunoDisciplina=disciplina)
    return render_to_response('pools/mural.html', {'nome': request.user.get_full_name(),
'mural': mural, 'nomDisci':poll_name}, context_instance=RequestContext(request))

def muralDetalhe(request, codigo):
    mural = Mural.objects.get(pk=codigo)
    mural.lido = 'true'
    mural.save()
    return render_to_response('pools/muralDetalhe.html', {'nome':
request.user.get_full_name(), 'mural': mural}, context_instance=RequestContext(request))

```

```

def selecaoDisciplinaCronograma(request):
    usuario = UsuarioUCS.objects.get(usuarioUCS=request.user.username)
    disciplinas = AlunoDisciplina.objects.filter(usuarioUCS=usuario)
    return render_to_response('pools/selecaoDisciplinaCronograma.html', {'nome':
request.user.get_full_name(), 'disciplinas' : disciplinas},
context_instance=RequestContext(request))

def cronograma(request, disciplina_id):
    usuario = UsuarioUCS.objects.get(usuarioUCS=request.user.username)
    disciplina_pool = Disciplina.objects.get(pk=disciplina_id)
    disciplina = AlunoDisciplina.objects.get(usuarioUCS=usuario, disciplina=disciplina_pool)
    cronograma = Cronograma.objects.filter(alunoDisciplina=disciplina)
    nomeDisciplina = disciplina_pool.nome
    return render_to_response('pools/cronograma.html', {'nome': request.user.get_full_name(),
'cronograma': cronograma, 'nomDisci':nomeDisciplina}, context_instance=RequestContext(request))

def cronogramaDetalhe(request, codigo):
    cronograma = Cronograma.objects.get(pk=codigo)
    return render_to_response('pools/cronogramaDetalhe.html', {'nome':
request.user.get_full_name(), 'cronograma': cronograma},
context_instance=RequestContext(request))

def cronogramaOrientacoes(request, codigo):
    cronograma = Cronograma.objects.get(pk=codigo)
    return render_to_response('pools/cronogramaOrientacoes.html', {'nome':
request.user.get_full_name(), 'cronograma': cronograma},
context_instance=RequestContext(request))

```

Model

```

from django.db import models

# Create your models here.
class UsuarioUCS(models.Model):
    usuarioUCS = models.CharField(max_length=20)
    nome = models.CharField(max_length=100)
    senha = models.CharField(max_length=20)
    aluno = models.BooleanField()

    def __unicode__(self):
        return self.usuarioUCS

class Servico(models.Model):
    nome = models.CharField(max_length=100)
    icone = models.CharField(max_length=200)
    url = models.CharField(max_length=300)
    def __unicode__(self):
        return self.nome

class AlunoServico(models.Model):
    usuarioUCS = models.ForeignKey(UsuarioUCS)
    servico = models.ForeignKey(Servico)
    favorito = models.BooleanField()
    def servicos_do_aluno(username):
        return self.usuarioUCS.usuarioUCS == username

class ServicosRecentes(models.Model):
    usuarioUCS = models.ForeignKey(UsuarioUCS)
    servico = models.ForeignKey(Servico)

class Disciplina(models.Model):
    nome = models.CharField(max_length=100)
    cargaHoraria = models.IntegerField()
    def __unicode__(self):
        return self.nome

class AlunoDisciplina(models.Model):
    usuarioUCS = models.ForeignKey(UsuarioUCS)

```

```

disciplina = models.ForeignKey(Disciplina)
semestre = models.CharField(max_length=6)

class Mural(models.Model):
    alunoDisciplina = models.ForeignKey(AlunoDisciplina)
    usuarioUCSRemetente = models.ForeignKey(UsuarioUCS)
    data = models.DateTimeField(auto_now=True)
    lido = models.BooleanField()
    descricao = models.CharField(max_length=2000)

class Cronograma(models.Model):
    alunoDisciplina = models.ForeignKey(AlunoDisciplina)
    numAula = models.IntegerField()
    nome = models.CharField(max_length=100)
    # Data
    qtHorasAula = models.IntegerField()
    descricao = models.CharField(max_length=5000)
    orientacoes = models.CharField(max_length=5000)
    data = models.DateTimeField(auto_now=True)

```

Templates

index.html

```

<!DOCTYPE html>
<html lang="en" class="no-js" <!--<![endif]-->
  <head>
    <meta name="viewport" content="width=device-width, height=device-height, user-
scalable=no">
    <meta charset="UTF-8" />
    <title>UCSVirtual</title>

    <script src="http://code.jquery.com/jquery-1.8.2.min.js"></script>

    <link rel="stylesheet" type="text/css" href="/static/css/style.css" />
    <link rel="stylesheet" type="text/css" href="/static/css/bootstrap.min.css" />

  </head>
  <body >

    <header>
      <div class="logo" align="center">
        
        <h1>Virtual - Móvel</h1>
      </div>
    </header>

    <section id="container_demo" >

      <a class="hiddenanchor" id="toregister"></a>
      <a class="hiddenanchor" id="tologin"></a>

      <div style="height: auto; padding: 3px" >
        <form action="/" autocomplete="on" class="form-horizontal"
method="post" >
          {% csrf_token %}

          <div class="control-group" >
            <label for="username" class="control-label"
labelnomargin " >Usuário </label>

            <div class="controls controlsnomargin" >
              <input id="username" name="username"
required="required" type="text" placeholder="meuusuario"/>
            </div>

          </div>

```

```

        <div class="control-group">
            <label for="password" class="youpasswd control-label
labelnomargin" > Senha </label>
            <div class="controls controlsnomargin" >
                <input id="password" name="password"
required="required" type="password" placeholder="ex. X8df!90EO" />
            </div>
        </div>

        <p class="login button">
            <input id="test" type="submit" value="Acessar"
class="btn" />
            <input type="reset" value="Limpar" class="btn" />
        </p>

    </form>
</div>

</section>

    <footer style="margin-bottom: 1px; padding: 0px" align="center">
        {% if messages %}
        <ul class="messages">
            {% for message in messages %}
                <p align="center"> <li{% if message.tags %} class="{{ message.tags
}}"{% endif %} ><mark><b>{{ message }}</b></mark></li> </p>
            {% endfor %}
        </ul>
        {% endif %}
        <p align="center"> <a href="https://ucsvirtual.ucs.br/" align="center">
Acessar versão Tradicional </a> </p>
    </footer>

</body>
</html>

```

appMain.html

```

<!DOCTYPE html>
<html lang="en" class="no-js"> <!--<![endif]-->
    <head>
        <meta charset="UTF-8" />
        <meta name="viewport" content="width=device-width, height=device-height,user-
scalable=no">
        <title></title>
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <link rel="stylesheet" type="text/css" href="/static/css/style.css" />
        <link rel="stylesheet" type="text/css" href="/static/css/bootstrap.css" />
    </head>

    <body class="containers" height="100%">

        <header style=" height: auto;">

            <form action="/">
                <div style="float: left; padding: 12px 10px 12px 10px; " >

                    <button class="btn" style="height: auto;" formtarget="_self"
formaction="/todosServicos" >
                        <i class="icon-align-justify"></i>
                    <br>Menu
                </button>

```

```

</div>
<div style=" float: none; padding: 5px; margin-left: 25%; " >
    <div style="float: left; padding: 5px 5px 0 0; height:50%;">
        <button formtarget="_self" formaction="/" class="btn
">
            <i class="icon-home"></i>
                Início
            </button>
        </div>
        <div style="float: none; padding: 5px 5px 0 0; height:50%; ">
class="btn">
            <button formtarget="_self" formaction="/sair"
                <i class="icon-remove"></i>
                    Sair
                </button>
            </div>
            <div style="float: none; padding: 5px 0 0 0; ">
                Bem Vindo, {{ nome }}
            </div>
        </div>
    </div>
</form>
</header>
<section style="border: 1px black solid; padding: 5px" align="none">
    Exibir serviços: <a href="#" style="padding: 10px" > Favoritos </a> <a
href="#" style="padding: 10px"> Recentes </a>
</section>
<section id="container_demo" style="height: 100%;">
    <div style=" min-height: 300px; padding: 5px 5px 5px 5px; " >
        {% if servicos %}
            <ul style=" list-style-type: none; float: left; width: 100%; "
>
                {% for servico in servicos %}
                    <li style="float: left; margin: 5px; width: 45%; height:
100px; text-align: center;">
                        <figure>
                            <a href="{{ servico.servico.url }}" />
                            
                            <figcaption>{{ servico.servico.nome
}}</figcaption>
                        </figure>
                    </li>
                {% endfor %}
            </ul>
        {% endif %}
    </div>
</section>
<footer style="border: 1px black solid; margin-botton: 1px; padding: 0px;
align="center">
    <p align="center"> UCSVirtual - Versão Móvel </p>
</footer>
</body>
</html>

```

todosServicos.html

```

<!DOCTYPE html>
<!--[if lt IE 7 ]> <html lang="en" class="no-js ie6 lt8"> <![endif]-->
<!--[if IE 7 ]> <html lang="en" class="no-js ie7 lt8"> <![endif]-->
<!--[if IE 8 ]> <html lang="en" class="no-js ie8 lt8"> <![endif]-->
<!--[if IE 9 ]> <html lang="en" class="no-js ie9"> <![endif]-->
<!--[if (gt IE 9)!(IE)]><!--> <html lang="en" class="no-js"> <!--<![endif]-->
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, height=device-height,user-
scalable=no">
    <!-- <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1"> -->
    <title></title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!--link rel="stylesheet" type="text/css" href="css/demo.css" /-->

    <link rel="stylesheet" type="text/css" href="/static/css/style.css" />
    <link rel="stylesheet" type="text/css" href="/static/css/bootstrap.css" />

    <!--link rel="stylesheet" type="text/css" href="css/animate-custom.css" /-->
  </head>
  <body class="containers" height="100%">

    <header style=" height: auto;">

      <form action="/">
        <div style="float: left; padding: 12px 10px 12px 10px; " >

          <button class="btn" style="height: auto;" formtarget="_self"
formaction="/todosServicos" >
            <i class="icon-align-justify"></i>

            <br>Menu
          </button>

        </div>

        <div style=" float: none; padding: 5px; margin-left: 25%; " >

          <div style="float: left; padding: 5px 5px 0 0; height:50%;">

            <button formtarget="_self" formaction="/" class="btn"
">
              <i class="icon-home"></i>

              Início
            </button>

          </div>

          <div style="float: none; padding: 5px 5px 0 0; height:50%; ">
            <button formtarget="_self" formaction="/sair"
class="btn">
              <i class="icon-remove"></i>

              Sair
            </button>

          </div>

          <div style="float: none; padding: 5px 0 0 0; ">
            Bem Vindo, {{ nome }}
          </div>

        </div>
      </form>

    </header>

```

```

<!-- Criar extensão -->
<section style="border: 1px black solid; padding: 5px" align="none">
    <div class="input-prepend">
        <span class="add-on"><i class="icon-search"></i></span>
        <input class="searchInput" id="prependingInput" size="16"
type="text" placeholder="Informe um serviço">
    </div>
</section>
<!-- EOB -->

<section id="container_demo" style="height: 100%;" formaction="/">

    <!-- Criar extensão -->
    <div style=" min-height: 300px; padding: 5px 5px 5px 5px; " >
        <table class="table table-striped table-bordered" >
            <!--caption>Cabeçalho geral</caption-->

            {% if servicos %}
            <tbody>
                {% for servico in servicos %}
                <tr>
                    <td><a href="/{{ servico.servico.url }}"> {{
servico.servico.nome }}</a></td>
                </tr>
                {% endfor %}
            </tbody>
            {% endif %}

        </table>

    </div>
<!-- EOB -->

</section>

    <footer style="border: 1px black solid; margin-botton: 1px; padding: 0px"
align="center">
        <p align="center"> UCSVirtual - Versão Móvel </p>
    </footer>

</body>
</html>

```

selecaoDisciplinaMural.html

```

<!DOCTYPE html>
<!--[if lt IE 7 ]> <html lang="en" class="no-js ie6 lt8"> <![endif]-->
<!--[if IE 7 ]> <html lang="en" class="no-js ie7 lt8"> <![endif]-->
<!--[if IE 8 ]> <html lang="en" class="no-js ie8 lt8"> <![endif]-->
<!--[if IE 9 ]> <html lang="en" class="no-js ie9"> <![endif]-->
<!--[if (gt IE 9)|!(IE)]><!--> <html lang="en" class="no-js"> <!--<![endif]-->
<head>
    <meta name="viewport" content="width=device-width, height=device-height,user-
scalable=no">
    <meta charset="UTF-8" />
    <!-- <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1"> -->
    <title></title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!--link rel="stylesheet" type="text/css" href="css/demo.css" /-->

    <link rel="stylesheet" type="text/css" href="/static/css/style.css" />
    <link rel="stylesheet" type="text/css" href="/static/css/bootstrap.css" />

    <!--link rel="stylesheet" type="text/css" href="css/animate-custom.css" /-->
</head>

```

```

<body class="containers" height="100%">
  <header style=" height: auto;">
    <form action="/">
      <div style="float: left; padding: 12px 10px 12px 10px; " >
        <button class="btn" style="height: auto;" formtarget="_self"
formaction="/todosServicos" >
          <i class="icon-align-justify"></i>
          <br>Menu
        </button>
      </div>
      <div style=" float: none; padding: 5px; margin-left: 25%; " >
        <div style="float: left; padding: 5px 5px 0 0; height:50%;">
          <button formtarget="_self" formaction="/" class="btn
" >
            <i class="icon-home"></i>
            Inicio
          </button>
        </div>
        <div style="float: none; padding: 5px 5px 0 0; height:50%; ">
          <button formtarget="_self" formaction="/sair"
class="btn">
            <i class="icon-remove"></i>
            Sair
          </button>
        </div>
        <div style="float: none; padding: 5px 0 0 0; ">
          Bem Vindo, {{ nome }}
        </div>
      </div>
    </form>
  </header>

  <section style="border: 1px black solid; padding: 5px" align="none">
    <div class="input-prepend">
      <span class="add-on"><i class="icon-search"></i></span>
      <input class="searchInput" id="prependingInput" size="16"
type="text" placeholder="Informe uma disciplina">
    </div>
  </section>
  <!-- EOB -->

  <section id="container_demo" style="height: 100%;">
    <div style=" min-height: 300px; padding: 5px 5px 5px 5px; " >
      <table class="table table-striped table-bordered" >

        {% if disciplinas %}
        <tbody>

```

```

        <tr>
            <td><a href="/mural/{{ disciplina.disciplina.nome
}}">Todas</a></td>
        </tr>
        {% for disciplina in disciplinas %}
        <tr>
            <td><a href="/mural/{{ disciplina.disciplina.nome }}"> {{
disciplina.disciplina.nome }}</a></td>
        </tr>
        {% endfor %}
    </tbody>
</table>

</div>

<!-- EOB -->

</section>

<footer style="border: 1px black solid; margin-botton: 1px; padding: 0px"
align="center">
    <p align="center"> UCSVirtual - Versao Movel </p>
</footer>

</body>
</html>

```

mural.html

```

<!DOCTYPE html>
<!--[if lt IE 7 ]> <html lang="en" class="no-js ie6 lt8"> <![endif]-->
<!--[if IE 7 ]> <html lang="en" class="no-js ie7 lt8"> <![endif]-->
<!--[if IE 8 ]> <html lang="en" class="no-js ie8 lt8"> <![endif]-->
<!--[if IE 9 ]> <html lang="en" class="no-js ie9"> <![endif]-->
<!--[if (gt IE 9)|!(IE)]><!--> <html lang="en" class="no-js"> <!--<![endif]-->
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, height=device-height,user-
scalable=no">
    <!-- <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1"> -->
    <title></title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!--link rel="stylesheet" type="text/css" href="css/demo.css" /-->

    <link rel="stylesheet" type="text/css" href="/static/css/style.css" />
    <link rel="stylesheet" type="text/css" href="/static/css/bootstrap.css" />

    <!--link rel="stylesheet" type="text/css" href="css/animate-custom.css" /-->
</head>
<body class="containers" height="100%">

    <header style=" height: auto;">

        <form action="/">
        <div style="float: left; padding: 12px 10px 12px 10px; " >

            <button class="btn" style="height: auto;" formtarget="_self"
formaction="/todosServicos" >

                <i class="icon-align-justify"></i>

                <br>Menu
            </button>

        </div>

        <div style=" float: none; padding: 5px; margin-left: 25%; " >

            <div style="float: left; padding: 5px 5px 0 0; height:50%;">

```

```

" >
        <button formtarget="_self" formaction="/" class="btn
                <i class="icon-home"></i>
                Início
        </button>
    </div>
    <div style="float: none; padding: 5px 5px 0 0; height:50%; ">
class="btn">        <button formtarget="_self" formaction="/sair"
                <i class="icon-remove"></i>
                Sair
        </button>
    </div>
    <div style="float: none; padding: 5px 0 0 0; ">
        Bem Vindo, {{ nome }}
    </div>
</div>
</form>
</header>
<!-- Criar extensão -->
<section style="border: 1px black solid; padding: 5px" align="none">
    <label style="float: left;"> Disciplina: </label>
    <label id="selectedItem" style="float: left;"> {{ nomDisci }} </label>
    <label id="backTo" style="float: none; margin-left: 65%;"><a
href="/selecaoDisciplinaMural" ><- Selec. Disc.</a></label>
</section>
<!-- EOB -->
<section id="container_demo" style="height: 100%;">
    <!-- Criar extensão -->
    <div style=" min-height: 300px; padding: 5px 5px 5px 5px; " >
        {% if mural %}
        <ul style=" list-style-type: none; float: none; width: 100%;
padding: 1px 0 0 0;" >
            {% for m in mural %}
                <li style=" float: none; margin: 5px; text-align: left;">
                    Data: {{ m.data|date:"d/m/Y"}}
                    <ul>
                        <li style="margin: 5px; ">
                            <a href="/muralDetalhe/{{ m.id }}">
{{ m.usuarioUCSRemetente.nome }} , às {{ m.data|date:"H:i"}} ({{ m.lido|yesno:"Lido,Não Lido" }})
                        </a>
                            </li>
                        </ul>
                    </li>
                </li>
            {% endfor %}
        </ul>
    </div>
<!-- EOB -->
</section>

```

```

                <footer style="border: 1px black solid; margin-bottom: 1px; padding: 0px"
align="center">
                <p align="center"> UCSVirtual - Versão Móvel </p>
            </footer>

        </body>
</html>

```

muralDetalhe.html

```

<!DOCTYPE html>
<!--[if lt IE 7 ]> <html lang="en" class="no-js ie6 lt8"> <![endif]-->
<!--[if IE 7 ]> <html lang="en" class="no-js ie7 lt8"> <![endif]-->
<!--[if IE 8 ]> <html lang="en" class="no-js ie8 lt8"> <![endif]-->
<!--[if IE 9 ]> <html lang="en" class="no-js ie9"> <![endif]-->
<!--[if (gt IE 9) || (IE)]><!--> <html lang="en" class="no-js"> <!--<![endif]-->
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, height=device-height,user-
scalable=no">
    <!-- <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1"> -->
    <title></title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!--link rel="stylesheet" type="text/css" href="css/demo.css" /-->

    <link rel="stylesheet" type="text/css" href="/static/css/style.css" />
    <link rel="stylesheet" type="text/css" href="/static/css/bootstrap.css" />

    <!--link rel="stylesheet" type="text/css" href="css/animate-custom.css" /-->
</head>
<body class="containers" height="100%">

    <header style=" height: auto;">

        <form action="/">
        <div style="float: left; padding: 12px 10px 12px 10px; " >

            <button class="btn" style="height: auto;" formtarget="_self"
formaction="/todosServicos" >
                <i class="icon-align-justify"></i>

                <br>Menu
            </button>

        </div>

        <div style=" float: none; padding: 5px; margin-left: 25%; " >

            <div style="float: left; padding: 5px 5px 0 0; height:50%;">

                <button formtarget="_self" formaction="/" class="btn"
">
                    <i class="icon-home"></i>

                    Início
                </button>

            </div>

            <div style="float: none; padding: 5px 5px 0 0; height:50%; " >
class="btn">
                <button formtarget="_self" formaction="/sair"

                    <i class="icon-remove"></i>

                    Sair
                </button>

            </div>

            <div style="float: none; padding: 5px 0 0 0; ">
                Bem Vindo, {{ nome }}

```

```

        </div>

        </div>
    </form>

</header>

<!-- Criar extensão -->
<section style="border: 1px black solid; padding: 5px" align="none">

    <label id="itemDescription" style="float: left;"> {{
mural.usuarioUCSRemetente.nome }}, às {{ mural.data|date:"H:i"}} </label>
    <label id="backTo" style="float: none; margin-left: 80%;"><a
href="/mural/{{ mural.alunoDisciplina.disciplina.nome }}" ><- Mural</a></label>

</section>
<!-- EOB -->

<section id="container_demo" style="height: 100%;">

    <!-- Criar extensão -->
    <div style="min-height: 300px; padding: 5px 5px 5px 5px; " >

        <text>
            {{ mural.descricao }}
        </text>

    </div>

    <!-- EOB -->

</section>

<footer style="border: 1px black solid; margin-bottom: 1px; padding: 0px"
align="center">
    <p align="center"> UCSVirtual - Versão Móvel </p>
</footer>

</body>
</html>

```

selecaoDisciplinaCronograma.html

```

<!DOCTYPE html>
<!--[if lt IE 7 ]> <html lang="en" class="no-js ie6 lt8"> <![endif]-->
<!--[if IE 7 ]> <html lang="en" class="no-js ie7 lt8"> <![endif]-->
<!--[if IE 8 ]> <html lang="en" class="no-js ie8 lt8"> <![endif]-->
<!--[if IE 9 ]> <html lang="en" class="no-js ie9"> <![endif]-->
<!--[if (gt IE 9)|!(IE)]><!--> <html lang="en" class="no-js"> <!--<![endif]-->
<head>
    <meta name="viewport" content="width=device-width, height=device-height,user-
scalable=no">
    <meta charset="UTF-8" />
    <!-- <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1"> -->
    <title></title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!--link rel="stylesheet" type="text/css" href="css/demo.css" /-->

    <link rel="stylesheet" type="text/css" href="/static/css/style.css" />
    <link rel="stylesheet" type="text/css" href="/static/css/bootstrap.css" />

    <!--link rel="stylesheet" type="text/css" href="css/animate-custom.css" /-->
</head>
<body class="containers" height="100%">

    <header style=" height: auto;">

        <form action="/">
        <div style="float: left; padding: 12px 10px 12px 10px; " >

```

```

        <button class="btn" style="height: auto;" formtarget="_self"
formaction="/todosServicos" >
            <i class="icon-align-justify"></i>
            <br>Menu
        </button>

    </div>

    <div style=" float: none; padding: 5px; margin-left: 25%; " >
        <div style="float: left; padding: 5px 5px 0 0; height:50%;">
            <button formtarget="_self" formaction="/" class="btn"
">
                <i class="icon-home"></i>
                Inicio
            </button>

        </div>

        <div style="float: none; padding: 5px 5px 0 0; height:50%; ">
class="btn">
            <button formtarget="_self" formaction="/sair"

                <i class="icon-remove"></i>
                Sair
            </button>

        </div>

        <div style="float: none; padding: 5px 0 0 0; ">
            Bem Vindo, {{ nome }}
        </div>

    </div>
</form>

</header>

<section style="border: 1px black solid; padding: 5px" align="none">
    <div class="input-prepend">
        <span class="add-on"><i class="icon-search"></i></span>
        <input class="searchInput" id="prependedInput" size="16"
type="text" placeholder="Informe uma disciplina">
    </div>

</section>
<!-- EOB -->

<section id="container_demo" style="height: 100%;">

    <div style=" min-height: 300px; padding: 5px 5px 5px 5px; " >

        <table class="table table-striped table-bordered" >

            {% if disciplinas %}
            <tbody>
                {% for disciplina in disciplinas %}
                <tr>
                    <td><a href="/cronograma/{{ disciplina.disciplina.id }}">
{{ disciplina.disciplina.nome }}</a></td>
                </tr>
                {% endfor %}
            </tbody>
            {% endif %}

```

```

        </table>

    </div>

    <!-- EOB -->

</section>

    <footer style="border: 1px black solid; margin-bottom: 1px; padding: 0px"
align="center">
        <p align="center"> UCSVirtual - Versao Movel </p>
    </footer>

</body>
</html>

```

cronograma.html

```

<!DOCTYPE html>
<!--[if lt IE 7 ]> <html lang="en" class="no-js ie6 lt8"> <![endif]-->
<!--[if IE 7 ]> <html lang="en" class="no-js ie7 lt8"> <![endif]-->
<!--[if IE 8 ]> <html lang="en" class="no-js ie8 lt8"> <![endif]-->
<!--[if IE 9 ]> <html lang="en" class="no-js ie9"> <![endif]-->
<!--[if (gt IE 9)|!(IE)]><!--> <html lang="en" class="no-js"> <!--<![endif]-->
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, height=device-height,user-
scalable=no">
    <!-- <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1"> -->
    <title></title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!--link rel="stylesheet" type="text/css" href="css/demo.css" /-->

    <link rel="stylesheet" type="text/css" href="/static/css/style.css" />
    <link rel="stylesheet" type="text/css" href="/static/css/bootstrap.css" />

    <!--link rel="stylesheet" type="text/css" href="css/animate-custom.css" /-->
</head>
<body class="containers" height="100%">

    <header style=" height: auto;">

        <form action="/">
        <div style="float: left; padding: 12px 10px 12px 10px; " >

            <button class="btn" style="height: auto;" formtarget="_self"
formaction="/todosServicos" >

                <i class="icon-align-justify"></i>

                <br>Menu
            </button>

        </div>

        <div style=" float: none; padding: 5px; margin-left: 25%; " >

            <div style="float: left; padding: 5px 5px 0 0; height:50%;">

                <button formtarget="_self" formaction="/" class="btn

                    <i class="icon-home"></i>

                    Inicio
                </button>

            </div>

            <div style="float: none; padding: 5px 5px 0 0; height:50%; ">

```

```

class="btn">
        <button formtarget="_self" formaction="/sair"
                <i class="icon-remove"></i>
                Sair
        </button>
    </div>
    <div style="float: none; padding: 5px 0 0 0; ">
        Bem Vindo, {{ nome }}
    </div>
</div>
</form>
</header>
<!-- Criar extensão -->
<section style="border: 1px black solid; padding: 5px" align="none">
    <label style="float: left;"> Disciplina: </label>
    <label id="selectedItem" style="float: left;"> {{ nomDisci }} </label>
    <label id="backTo" style="float: none; margin-left: 65%;"><a
href="/selecaoDisciplinaCronograma" ><- Selec. Disc.</a></label>
</section>
<!-- EOB -->
<section id="container_demo" style="height: 100%;">
    <!-- Criar extensão -->
    <div style=" min-height: 300px; padding: 5px 5px 5px 5px; " >
        {% if cronograma %}
        <ul style=" list-style-type: none; float: none; width: 100%;
padding: 1px 0 0 0;" >
            {% for c in cronograma %}
            <li style=" float: none; margin: 5px; text-align: left;">
                Aula, {{ c.numAula }}
                <ul>
                    <li style="margin: 5px; ">
                        <a href="/cronogramaDetalhe/{{ c.id
}}"> {{ c.data|date:"d/m/Y"}}, {{ c.qtHorasAula }} Horas-Aula UCS </a>
                    </li>
                </ul>
            </li>
            {% endfor %}
        </ul>
        {% endif %}
    </div>
<!-- EOB -->
</section>
<footer style="border: 1px black solid; margin-botton: 1px; padding: 0px"
align="center">
    <p align="center"> UCSVirtual - Versão Móvel </p>
</footer>
</body>
</html>

```

cronogramaDetalhe.html

```
<!DOCTYPE html>
```

```

<!--[if lt IE 7 ]> <html lang="en" class="no-js ie6 lt8"> <![endif]-->
<!--[if IE 7 ]> <html lang="en" class="no-js ie7 lt8"> <![endif]-->
<!--[if IE 8 ]> <html lang="en" class="no-js ie8 lt8"> <![endif]-->
<!--[if IE 9 ]> <html lang="en" class="no-js ie9"> <![endif]-->
<!--[if (gt IE 9)|!(IE)]><!--> <html lang="en" class="no-js"> <!--<![endif]-->
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, height=device-height,user-
scalable=no">
  <!-- <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1"> -->
  <title></title>
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!--link rel="stylesheet" type="text/css" href="css/demo.css" /-->

  <link rel="stylesheet" type="text/css" href="/static/css/style.css" />
  <link rel="stylesheet" type="text/css" href="/static/css/bootstrap.css" />

  <!--link rel="stylesheet" type="text/css" href="css/animate-custom.css" /-->
</head>
<body class="containers" height="100%">

  <header style=" height: auto;">

    <form action="/">
    <div style="float: left; padding: 12px 10px 12px 10px; " >

      <button class="btn" style="height: auto;" formtarget="_self"
formaction="/todosServicos" >
        <i class="icon-align-justify"></i>

        <br>Menu
      </button>

    </div>

    <div style=" float: none; padding: 5px; margin-left: 25%; " >

      <div style="float: left; padding: 5px 5px 0 0; height:50%;">

        <button formtarget="_self" formaction="/" class="btn"
">
          <i class="icon-home"></i>

          Início
        </button>

      </div>

      <div style="float: none; padding: 5px 5px 0 0; height:50%; "
class="btn">
        <button formtarget="_self" formaction="/sair"

          <i class="icon-remove"></i>

          Sair
        </button>

      </div>

      <div style="float: none; padding: 5px 0 0 0; ">
        Bem Vindo, {{ nome }}
      </div>

    </div>
  </form>

</header>

<!-- Criar extensão -->
<section style="border: 1px black solid; padding: 5px" align="none">

```

```

        <label id="itemDescription" style="float: left;"> Aula {{
cronograma.numAula }}, {{ cronograma.data|date:"d/m/Y"}} </label>
        <label id="backTo" style="float: none; margin-left: 65%;"><a
href="/cronograma/{{ cronograma.alunoDisciplina.disciplina.id }}" ><- Cronograma</a></label>

</section>
<!-- EOB -->

<section id="container_demo" style="height: 100%;">

    <!-- Criar extensão -->
    <div style="min-height: 300px; padding: 5px 5px 5px 5px; " >

        <text>
            {{ cronograma.descricao }}
        </text>

        <label id="detalhe" style="min-height: 25px; padding: 15px 0px 0px
0px; " ><a href="/cronogramaOrientacoes/{{ cronograma.id }}" >Orientações da Aula</a></label>
        </div>

        <div >

        </div>
    <!-- EOB -->

</section>

<footer style="border: 1px black solid; margin-botton: 1px; padding: 0px"
align="center">
    <p align="center"> UCSVirtual - Versão Móvel </p>
</footer>

</body>
</html>

```

cronogramaOrientacoes.html

```

<!DOCTYPE html>
<!--[if lt IE 7 ]> <html lang="en" class="no-js ie6 lt8"> <![endif]-->
<!--[if IE 7 ]> <html lang="en" class="no-js ie7 lt8"> <![endif]-->
<!--[if IE 8 ]> <html lang="en" class="no-js ie8 lt8"> <![endif]-->
<!--[if IE 9 ]> <html lang="en" class="no-js ie9"> <![endif]-->
<!--[if (gt IE 9)|!(IE)]><!--> <html lang="en" class="no-js"> <!--<![endif]-->
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, height=device-height,user-
scalable=no">
    <!-- <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1"> -->
    <title></title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!--link rel="stylesheet" type="text/css" href="css/demo.css" /-->

    <link rel="stylesheet" type="text/css" href="/static/css/style.css" />
    <link rel="stylesheet" type="text/css" href="/static/css/bootstrap.css" />

    <!--link rel="stylesheet" type="text/css" href="css/animate-custom.css" /-->
</head>
<body class="containers" height="100%">

    <header style=" height: auto;">

        <form action="/">
        <div style="float: left; padding: 12px 10px 12px 10px; " >

            <button class="btn" style="height: auto;" formtarget="_self"
formaction="/todosServicos" >

                <i class="icon-align-justify"></i>

```

```

                </br>Menu
            </button>

        </div>

        <div style=" float: none; padding: 5px; margin-left: 25%; " >
            <div style="float: left; padding: 5px 5px 0 0; height:50%;">
                <button formtarget="_self" formaction="/" class="btn
                " >
                    <i class="icon-home"></i>
                    Início
                </button>

            </div>

            <div style="float: none; padding: 5px 5px 0 0; height:50%; " >
                <button formtarget="_self" formaction="/sair"
                class="btn">
                    <i class="icon-remove"></i>
                    Sair
                </button>

            </div>

            <div style="float: none; padding: 5px 0 0 0; " >
                Bem Vindo, {{ nome }}
            </div>

        </div>
    </form>

</header>

<!-- Criar extensão -->
<section style="border: 1px black solid; padding: 5px" align="none">
    <label id="itemDescription" style="float: left;"> Orientações: </label>

    <label id="itemDescription" style="float: left;"> Aula {{
cronograma.numAula }}, {{ cronograma.data|date:"d/m/Y"}} </label>
    <label id="backTo" style="float: none; margin-left: 80%;"><a
href="/cronogramaDetalhe/{{ cronograma.id }}" ><- Voltar</a></label>

</section>
<!-- EOB -->

<section id="container_demo" style="height: 100%;">
    <!-- Criar extensão -->
    <div style="min-height: 300px; padding: 5px 5px 5px 5px; " >
        <text>
            {{ cronograma.orientacoes }}
        </text>

    </div>

    <!-- EOB -->

</section>

    <footer style="border: 1px black solid; margin-botton: 1px; padding: 0px"
align="center">
    <p align="center"> UCSVirtual - Versão Móvel </p>

```

```
</footer>  
</body>  
</html>
```