

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS DE TELECOMUNICAÇÕES

RODRIGO LANGOWSKI

**MOBALUNO: INTERFACE *MOBILE* INTEGRADA A SISTEMA DE
GESTÃO EDUCACIONAL**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA
2014

RODRIGO LANGOWSKI

**MOBALUNO: INTERFACE *MOBILE* INTEGRADA A SISTEMA DE
GESTÃO EDUCACIONAL**

Trabalho de Conclusão de Curso
apresentado ao Departamento de Eletrônica,
como requisito parcial para obtenção de
grau de Tecnólogo em Sistemas de
Telecomunicações, da Universidade
Tecnológica Federal do Paraná.

Orientador: Prof. Luís Alberto Lucas

CURITIBA
2014

TERMO DE APROVAÇÃO

RODRIGO LANGOWSKI

MOBALUNO: INTERFACE *MOBILE* INTEGRADA A SISTEMA DE GESTÃO EDUCACIONAL

Este trabalho de conclusão de curso foi apresentado no dia 14 de março de 2014, como requisito parcial para obtenção do título de Tecnólogo em Sistemas de Telecomunicações, outorgado pela Universidade Tecnológica Federal do Paraná. O aluno Rodrigo Langowski foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Prof. Dr. Milton Luiz Polli
Coordenador de Curso
Departamento Acadêmico de Mecânica

Prof. Esp. Sérgio Moribe
Responsável pela Atividade de Trabalho de Conclusão de Curso
Departamento Acadêmico de Eletrônica

BANCA EXAMINADORA

Prof. César M. Vargas Benítez.
UTFPR

Prof. Maurício Alves Mendes.
UTFPR

Prof. Luís Alberto Lucas
Orientador - UTFPR

“A Folha de Aprovação assinada encontra-se na Coordenação do Curso”

Muitas pessoas devem a grandeza de suas vidas aos problemas que tiveram de vencer. (BADEN-POWELL, Robert, 1922).

RESUMO

LANGOWSKI, Rodrigo. **Interface móbile integrada a sistema de gestão educacional**. 2014. 46 f. Monografia (Graduação) – Curso Superior de Tecnologia em Sistemas de Telecomunicações, UTFPR, Curitiba, 2014.

O objetivo deste trabalho é apresentar uma interface que seja de fácil integração a um sistema de gestão educacional existente. O acesso ao sistema de gestão será via smartphone através da internet. O aplicativo permite informar aos usuários que estejam dentro de alguma estrutura hierárquica (universidade, empresa) realizando transmissões para usuários selecionados ou em broadcast. O estudo foi focado no ambiente educacional. O aplicativo permite, por exemplo, o envio de notas para os alunos de maneira individualizada ou um informativo da instituição a respeito de recesso por causa de um feriado.

Palavras-chave: Gestão Educacional. Aplicativo.

ABSTRACT

LANGOWSKI, Rodrigo. **Interface móbile integrada a sistema de gestão educacional**. 2014. 46 f. Monografia (Graduação) – Curso Superior de Tecnologia em Sistemas de Telecomunicações, UTFPR, Curitiba, 2014.

The objective of this work is to present an interface that is easy to integrate to an existing educational management system, so that a device like a smartphone can access the system through the internet. The developed application can be used on any platform on witch a smartphone device operates. The study was conducted in corporate and educational environments because the application contains the usability of informing users that are within a certain hierarchical structure performing transmissions to selected users or in broadcasting. For instance an educational setting, where it is possible to send grades to the students individually or information about the institution like withdraw due to holiday.

Keywords: Educational Management. Application.

LISTA DE FIGURAS

Figura 1: Tabelas relacionadas a mensagens.	177
Figura 2: Tabelas relacionadas à estrutura da instituição.	199
Figura 3: Tabelas relacionadas ao usuário.	20
Figura 4: Planejamento da página de aluno.	222
Figura 5: Planejamento da página de professor ou administrador.	23
Figura 6: Página de login.	244
Figura 7: Página do Aluno com mensagens novas.	244
Figura 8: Página do Aluno sem mensagens novas.	255
Figura 9: Página do Aluno na aba de mensagens filtradas e lidas.	255
Figura 10: Página de leitura da mensagem.	266
Figura 11: Página do professor e administrador.	266
Figura 12: Fluxograma do sistema.	277
Figura 13: Inserção de dados na página do professor/administrador.	288
Figura 14: Conferindo os dados no banco na tabela de mensagens.	299
Figura 15: Conferindo os dados no banco na tabela de destinatário.	29
Figura 16: Mensagem nova na página principal do aluno.	30
Figura 17: Leitura da mensagem.	30
Figura 18: Conferindo se a mensagem foi lida.	31
Figura 19: Mensagem lida não aparece na página principal.	311

SUMÁRIO

1 INTRODUÇÃO	8
1.1 OBJETIVOS	9
1.1.1 Objetivos específicos.....	9
1.2 JUSTIFICATIVAS	10
1.3 METODOLOGIA	10
1.3.1 Desenvolvimento da interface	11
1.3.2 <i>Webapp</i> compatível	11
1.3.3 Fácil integração	11
1.4 ESTRUTURA DO TRABALHO	11
2 SOFTWARES DE APLICAÇÃO	13
2.1 HISTÓRIA DOS SMARTPHONES	13
2.2 SOFTWARES DE APLICAÇÃO	14
2.2.1 Softwares de aplicação para Celulares	14
2.3 <i>WEBAPP</i>	15
2.4 LINGUAGEM DE DESENVOLVIMENTO	15
2.5 SISTEMAS DE GESTÃO EDUCACIONAL	16
3 DESENVOLVIMENTO DA PESQUISA	17
3.1 BANCO DE DADOS	17
3.2 PLANEJAMENTO DAS TELAS	21
3.2.1 <i>Wireframe</i>	21
3.2.2 Desenvolvimento da Interface.	23
3.3 FLUXOGRAMA	27
4 ANÁLISE DE RESULTADOS	28
5 CONCLUSÃO	32
REFERÊNCIAS	33
ANEXOS	34

1 INTRODUÇÃO

Com o avanço de tecnologias e serviços, a qualidade e a praticidade são fundamentais, ainda mais em se tratando do atendimento personalizado ao usuário, apresentando informações e dados de interesse.

A internet está presente no dia-a-dia das pessoas. Seja em aparelhos mobile ou nas residências a conectividade permite que os usuários recebam notícias e informações em tempo real e interajam com o conteúdo e com outros usuários.

Hoje é possível contratar sistemas de gerenciamento educacional que controlam e auxiliam a gestão de uma instituição em quase todas as áreas, e a parte acessível ao seu cliente é restrita às comunicações, e em alguns casos cuida também das matrículas de alunos (ex. UTFPR). Porém o acesso aos seus portais, ou sistemas, são limitados a web sites que, em sua maioria, não tem a estrutura para visualizações em aparelhos celulares, tablets e smartphones.

Durante o ano letivo a comunicação entre professores e alunos, geralmente, se dá durante as aulas, ou seja, dificilmente se ultrapassa a quantidade de horas que a matéria tem para uma discussão de ideias ou de eventos extracurriculares. Também se observa que se poderia ter alguma ferramenta que auxiliasse na divulgação de notas parciais, informativos de trabalhos e mensagens de alterações de datas (ex. entrega de trabalhos, provas, eventos, recessos). Na UTFPR existe um sistema para o aluno acessar, apenas aos inícios e finais de semestre, onde é possível verificar se a matrícula está feita e quais são as notas finais obtidas. Percebe-se, portanto, que a comunicação entre os professores e os alunos é limitada e, em muitos casos, insuficiente. Outras alternativas buscadas pelos professores são os usos de páginas pessoais e a comunicação através de emails, que misturam com os emails de uso pessoal do aluno.

Para melhorar a comunicação deve-se criar um vínculo entre as duas partes de maneira ágil e precisa, de modo que os professores, e a universidade, possam interagir com o aluno de forma mais direta, mandando mensagens personalizadas e com o conteúdo que aquele aluno tem interesse de acordo com o curso em que está matriculado. Estas mensagens recebidas pelo aluno poderão ser acessadas de qualquer lugar através de um aplicativo para celulares.

No início de um semestre os professores fazem listas com os correios eletrônicos dos alunos para ter uma forma de contato, porém alguns alunos têm mais de uma conta de e-mail, ou tem apenas um e-mail corporativo, de modo que o próprio aluno pode se atrapalhar na hora de organizar suas informações pessoais, corporativas e acadêmicas.

Até mesmo os professores podem ter algum tipo de problema ao não entender a caligrafia de algum aluno e não conseguir enviar alguma informação ou mesmo, ter o trabalho de criar grupos de e-mails distintos para as diversas turmas que leciona.

A instituição não conseguirá atingir os alunos caso durante o curso, estes mudem a conta de e-mail, pois esta alteração deveria ser informada à instituição e, muitas vezes, o aluno não é organizado a este ponto. Estas situações podem levar à perda de informações, mas também é possível que a informação não exista, como quando o aluno falta a aula onde receberia a nota de uma prova que realizou.

1.1 OBJETIVOS

O objetivo deste trabalho de conclusão de curso é apresentar uma solução de acesso aos sistemas corporativos da universidade, o “MobAluno”. Esta solução é portátil e de fácil integração com serviços existentes.

1.1.1 Objetivos específicos

- a) Desenvolver uma interface leve e compacta com a visualização adequada para *smartphones*.
- b) Transformar a página em *webapp* compatível com IOS e Android.
- c) Desenvolver código fonte de fácil implementação para sistemas de gestão educacionais do mercado.

1.2 JUSTIFICATIVAS

Com o avanço da tecnologia dos aparelhos *smartphones*, cada vez mais estudantes estão conectados e aptos a receber informações a qualquer instante. Desta forma propõe-se neste trabalho o desenvolvimento de um sistema *smartphone* que leve as informações acadêmicas para o mundo *mobile*, de forma organizada e personalizada.

Pretende-se com essa proposta, acompanhar a tendência da computação móvel e *cloud computing*, que está presente na maioria dos países através da disseminação dos dispositivos *smartphones*. Para atender ao público que frequenta os centros acadêmicos, as instituições teriam que desenvolver novas interfaces e as vezes até alterar o modelo computacional aplicado para a viabilização de uma interface *mobile* aos seus alunos e colaboradores.

Aplicativos *mobile* permitem uma comunicação de baixo custo entre provedores e consumidores de informação. O MobAluno criará um dialogo triangular, envolvendo Instituição, professores e aluno. O aplicativo otimizará a organização das informações.

Por exemplo, a Instituição precisa informar aos alunos de engenharia mecânica que terá uma palestra agendada para a semana seguinte. Através do uso de filtros, a universidade poderá encaminhar a mensagem apenas ao público de interesse (a engenharia mecânica, no exemplo) sem necessitar escolher grupos de e-mails, que são facilmente perdidos. Esse sistema de filtragem estará conectado às informações dos alunos do sistema acadêmico, de forma a apresentar os dados atualizados dos alunos, que para acessarem suas informações deverão utilizar o seu código de aluno.

1.3 METODOLOGIA

1.3.1 Desenvolvimento da interface

Possuindo uma interface leve e simplificada, com apenas as informações que o usuário necessita, a usabilidade e a navegabilidade da página serão melhor aproveitadas pelo usuário. Nesta fase será elaborado um protótipo do layout, analisando a melhor interação da página simulando as localizações dos objetos e suas disposições em diferentes resoluções de tela.

1.3.2 *Webapp* compatível

Os sistemas operacionais de aparelhos móveis dominantes no mercado atualmente são: o IOS da Apple e o Android, do Google. Visando atender a estes dois sistemas, será feito um desenvolvimento que tenha uma estrutura independente da arquitetura.

1.3.3 Fácil integração

A facilidade de integração com sistemas já existentes dar-se-á através de parâmetros comuns aos sistemas de gestão educacionais. Nesta fase serão estipuladas as variáveis chaves que servirão de referências para uma integração a fim de deixar o aplicativo versátil.

1.4 ESTRUTURA DO TRABALHO

Esta monografia está organizada em 5 capítulos, sendo que o primeiro apresenta os aspectos introdutórios, a descrição da proposta e os objetivos. Os

demais capítulos estão organizados da maneira descrita a seguir.

No capítulo 2, são apresentados os temas: história dos *smartphones*, softwares de aplicação, *webapp*, linguagem PHP, MYSQL e sistemas de gestão educacionais.

O processo do desenvolvimento do trabalho é apresentado no capítulo 3, onde primeiramente mostra-se a formação do banco de dados e as ligações entre as tabelas. Em seguida é apresentado o desenvolvimento das telas, desde o planejamento em *wireframe*, o HTML desenvolvido e a programação PHP inserida. Por fim, serão apresentadas o levantamento de custos para o desenvolvimento do sistema e de manutenção e o custo para colocar o sistema em atividade.

Apresenta-se no capítulo 4 uma análise dos resultados obtidos e finalmente no capítulo 5 as conclusões do trabalho.

2 SOFTWARES DE APLICAÇÃO

2.1 HISTÓRIA DOS SMARTPHONES

Assim como nos computadores, os aparelhos celulares também foram beneficiados com novas tecnologias. O tamanho dos aparelhos com o tempo foi diminuindo, as resoluções de tela aumentaram, novas funções foram acrescentadas e a memória foi expandida. A alteração no design foi rápida e está em constante evolução, mudanças foram feitas nas tecnologias de transmissão dos sinais e novos conceitos de comunicação surgiram, entre eles está o serviço de mensagens instantâneas.

Os celulares começaram a agregar dispositivos na sua composição, como câmera, GPS, rádio e capacidade de execução de arquivos MP3.

Um *smartphone* é um aparelho celular que contém um sistema operacional que permite expandir suas funcionalidades e permite que desenvolvedores criem programas ampliando a utilidade do aparelho.

As principais características de um *smartphone* são: possibilidade de conexão com uma rede de dados, conexão com um computador e o uso de uma memória externa, se necessário.

Com o avanço da tecnologia os computadores foram diminuindo de tamanho e ganhando novas funções, e a necessidade de ter a informação a todo o momento fez com que os PDA's surgissem. Seu significado literal é assistente pessoal digital mas o mercado conheceu esses aparelhos pelo nome de Palmtop. Nele se encontram funções como: agenda, editores de texto, planilhas eletrônicas, conexão com a internet via *wi-fi*, acesso à conta de e-mail.

“Os *smartphones* são o resultado de um longo processo de evolução e convergência de dispositivos” (MORIMOTO, 2009).

O avanço tecnológico dos aparelhos celulares fez com que estes incorporassem os recursos dos computadores de mão e, conseqüentemente, os substituíssem. Os novos aparelhos foram chamados de *smartphones*, e não mais de telefones celulares.

Os *smartphones* cresceram no mercado de acordo com a agência de notícias AFP sendo que, no segundo semestre de 2013 foram vendidos 225 milhões de aparelhos, significando 51,8% de todos os celulares vendidos no mundo, ou seja, pela primeira vez na história a venda de *smartphones* superou a de celulares convencionais.

Como atualmente existem diversas marcas de smartphones e também os sistemas operacionais disponíveis, o consumidor final tem que escolher com cautela qual aparelho adquirir.

2.2 SOFTWARES DE APLICAÇÃO

Os *softwares* se dividem em dois tipos: de sistema e de aplicação.

Os *softwares* de sistema interagem com a máquina e, por isso, são chamados de programas de baixo nível. Exemplos deles são os Sistemas Operacionais, que possibilitam o uso do aparelho e gerenciam seus dispositivos e periféricos (através de *drivers*).

Diferente dos *softwares* de sistema, os de aplicação interagem com o usuário proporcionando o uso efetivo da máquina pelos seus programas, já que permitem que o usuário execute tarefas de seu interesse. Exemplo: um programa editor de texto.

Dentro da categoria de programas de aplicação, pode-se ter diversos tipos de *softwares* sendo eles gerenciais, de infraestrutura, simulação, desenvolvimento de mídia, *mobile*, educacionais, etc.

2.2.1 Softwares de aplicação para Celulares

Os softwares de aplicação para celulares popularmente conhecidos por Aplicativos ou App, são programas concebidos para serem executados em dispositivos móveis, tais como *smartphones*, *tablets*, *e-books* e *players* de músicas.

Os aplicativos têm muitas funcionalidades. É possível, em alguns casos modificar as propriedades de alguns dispositivos, visando criar novas aplicações (exemplo, alterar a programação da lâmpada de *flash* do celular para ter uma lanterna) entre os tipos de aplicativos, pode-se citar os aplicativos de gerenciamento de aparelhos domésticos, gerenciamento pessoal, jogos, aplicativos de serviços de mensagem instantânea, de voz, de vídeo, etc.

Para o desenvolvimento de um aplicativo deve-se levar em conta fatores como, qual dispositivo irá usá-lo e qual plataforma será usada, pois atualmente no mercado há diversos tipos de dispositivos e seus sistemas operacionais, e a escolha do dispositivo e da plataforma restringe o uso do aplicativo.

2.3 WEBAPP

AS Webapp's são aplicações que executam em servidores e são acessíveis através de navegadores de internet. Essas aplicações são populares porque qualquer computador ou smartphone que possua acesso a internet via browser pode fazer uso delas.

“A fluidez cada vez maior de interfaces de usuário em aplicativos criará uma demanda por um tipo mais fluido de servidor web também” (KOSNER, 2012).

Os serviços prestados pelos aplicativos de web são tão diversificados quanto os aplicativos que devem ser instalados nos dispositivos, sendo que a principal diferença é a tecnologia de processamento, que ou se estabelece no servidor ou no dispositivo.

As linguagens mais comuns de programação para desenvolver os aplicativos web são: Java, JavaScript, DHTML, Flash, Silverlight e PHP.

2.4 LINGUAGEM DE DESENVOLVIMENTO

O *Personal Home Page (PHP)* permite o desenvolvimento de *sites* dinâmicos

na internet, em que o usuário possa interagir através de formulários, links e parâmetros passados através da URL.

O PHP é executado no servidor em resposta às solicitações do usuário feitas através de páginas HTML. Um exemplo de aplicação deste tipo é o preenchimento de um campo de senha em uma janela no navegador, que gerará a demanda de validação de senha a um programa que roda no servidor.

O MYSQL é um sistema de gerenciamento de banco de dados (SGBD), desenvolvido pela Oracle, sob a forma de software livre. Este SGBD é utilizado em muitas aplicações *web* por ser de fácil integração com linguagens para *web* como o PHP.

2.5 SISTEMAS DE GESTÃO EDUCACIONAL

Para auxiliar várias áreas da administração de uma instituição de ensino, surgiram os Sistemas de Gestão Educacional que se desenvolveram a partir de modelos de CRM (*Customer Relationship Management*) que são programas de gerenciamento entre empresa e cliente. Esses softwares também auxiliam no controle e comunicação de vários departamentos de uma instituição.

Os sistemas contam com restrições de ações, permitindo somente usuários com um certo privilégio possam visualizar ferramentas para administrar algum departamento. Exemplo: contadores podem fazer a gestão do sistema financeiro da instituição e de visualização exclusiva desses dados. O mesmo se aplica para o cliente final de uma instituição de ensino, o aluno. Sua forma de acesso é a mais restrita dentre todos os usuários, pois apenas visualiza, sem alterar, as informações. E na grande maioria dos softwares a conta do aluno no sistema funciona como um fórum, onde o professor coloca notícias ou faz a solicitação e a correção de trabalhos escolares.

3 DESENVOLVIMENTO DA PESQUISA

3.1 BANCO DE DADOS

Para que o sistema funcione corretamente o banco de dados deve ser construído de forma a prever todas as possibilidades de integração a sistemas pré-estabelecidos, assim devem ser criados tabelas e campos com relacionamentos de fácil entendimento e manutenção.

A estrutura foi desenvolvida com 10 tabelas, que podem ser agrupadas em duas áreas ligadas pela tabela central, onde estão presentes as informações do usuário (User).

Essas áreas são tabelas relacionadas à mensagem e estrutura dos cursos.

A primeira área consiste nas mensagens que aparecem ao aluno provenientes dos professores ou da instituição como demonstra a figura 1:

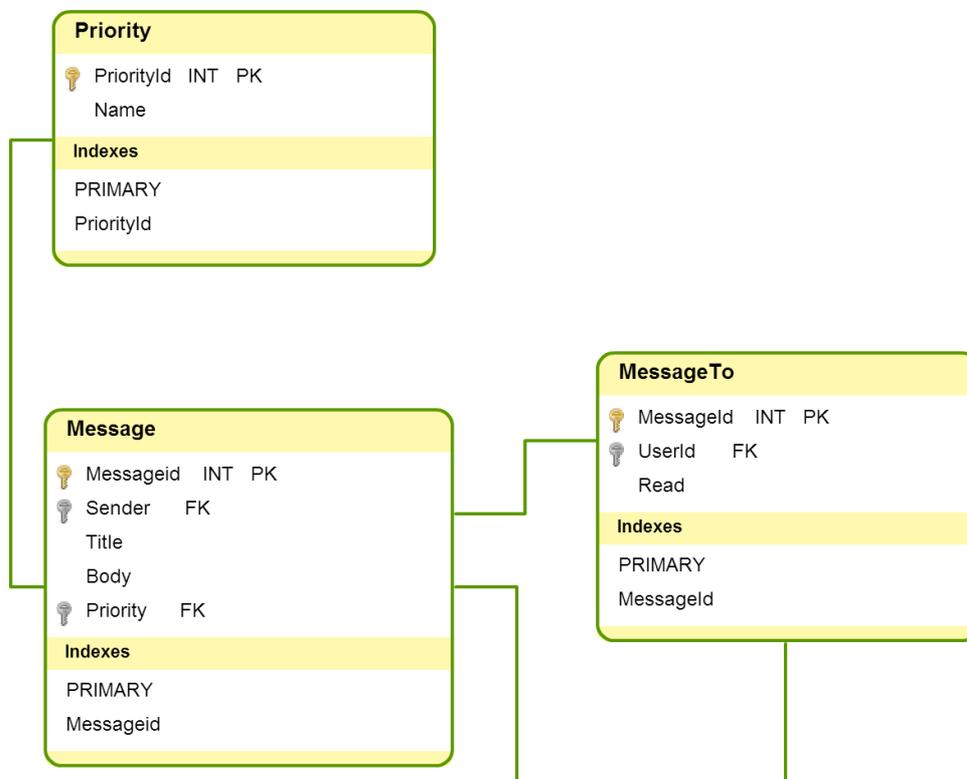


Figura 1: Tabelas relacionadas a mensagens.
Fonte: Autoria própria.

A tabela Message contém informações de quem enviou a mensagem, o assunto, o texto da mensagem e qual a prioridade dela. Através da chave primária MessageId é feito o relacionamento com a tabela MessageTo onde se guarda o destinatário e uma *flag* que determina se a mensagem foi lida. Ainda na tabela Message a chave estrangeira (*foreign key*) Priority faz relacionamento com a tabela Prority, que contém o código e a que se refere, como no exemplo do quadro 1:

PriorityId	Name
1	Baixa
2	Alta

Quadro 1: Prioridades da mensagem.
Fonte: Autoria própria.

Como pode se observar na figura:

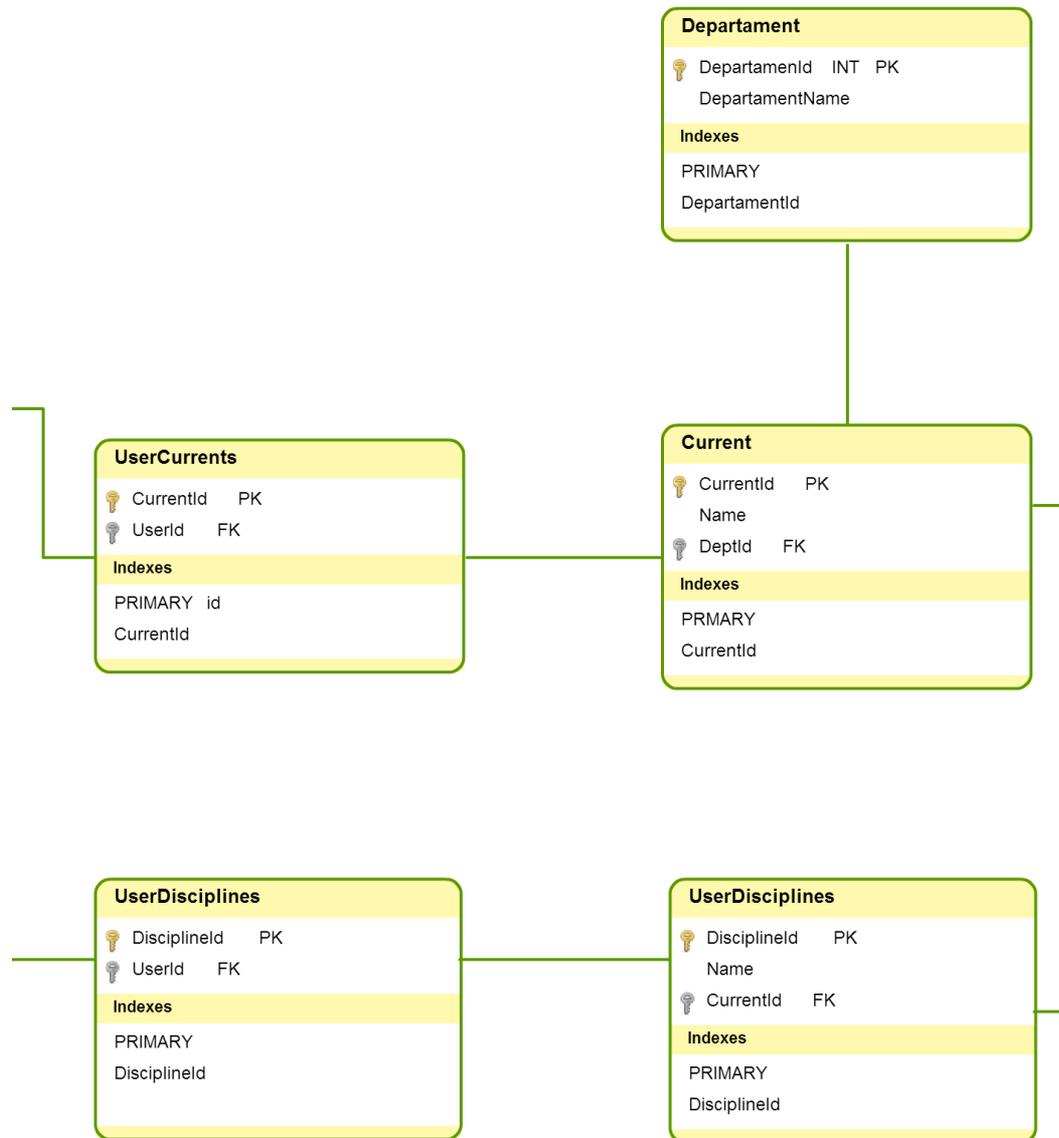


Figura 2: Tabelas relacionadas à estrutura da instituição.
Fonte: Autoria própria.

A figura 2 demonstra a segunda área que contém a estrutura de cursos, disciplinas e departamentos de uma instituição

Os relacionamentos realizados entre estas tabelas formulam então a estrutura e as dependências do sistema, por exemplo, nas tabelas UserCurrents e UserDisciplines é feita a ligação do aluno com a turma que está e em quais disciplinas está matriculado, da mesma forma a lógica se aplica ao professor ligando a turma e disciplinas que leciona.

A figura 3 mostra a parte central do banco de dados:

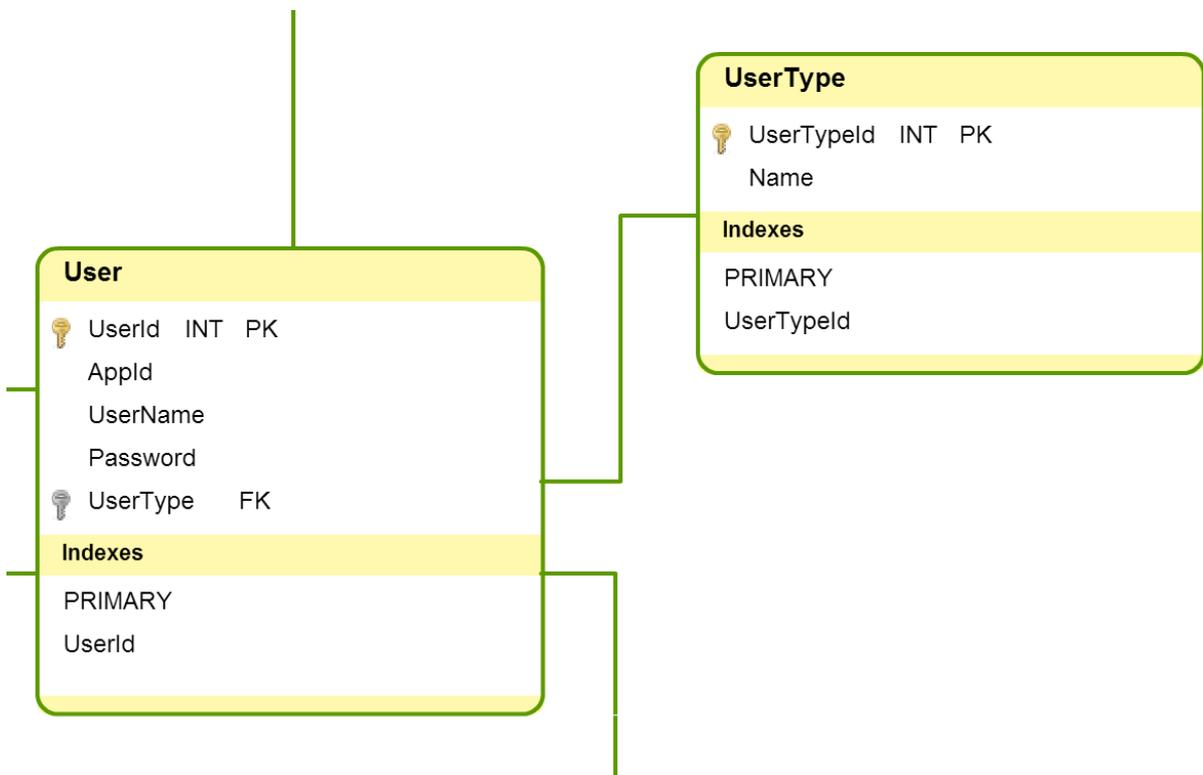


Figura 3: Tabelas relacionadas ao usuário.
Fonte: Autoria própria.

Como se pode observar existem duas tabelas para configurar os dados do usuário: User e UserType.

Na tabela User determina-se qual o código de acesso do usuário (UserId) que pode ser o código de registro da instituição, tem-se o campo UserName contendo o nome do usuário, *Password* onde a senha de acesso ao sistema é armazenada, AppId é um campo que pode ser usado caso o sistema a ser integrado possua algum tipo de identificação própria do usuário e o campo UserType que é uma chave que faz a ligação com a tabela UserType para determinar que tipo de usuário é. A tabela 2 mostra os tipos possíveis de usuário:

UserTypeId	Name
1	Aluno
2	Professor
3	Administrador

Quadro 2: Tipos de usuários.
Fonte: Autoria própria.

O tipo de usuário Administrador é aquele que dentro da instituição tem acesso a todas as mensagens de todas as turmas, disciplinas e departamentos. Tendo como filtro apenas as disciplinas que leciona, o usuário Professor visualiza apenas turmas dessas disciplinas, e finalmente o usuário Aluno visualiza as mensagens enviadas tanto pela conta de Administrador quanto de Professor.

3.2 PLANEJAMENTO DAS TELAS

3.2.1 *Wireframe*

Ao se pensar em qual forma de interação o usuário terá com o sistema, planeja-se como será a interface e o primeiro passo é arquitetar a disposição dos objetos nas janelas da interface. Para desenhar as janelas, utilizou-se o software SmartDraw.

A figura 4 mostra o que o aluno irá visualizar logo após o login:

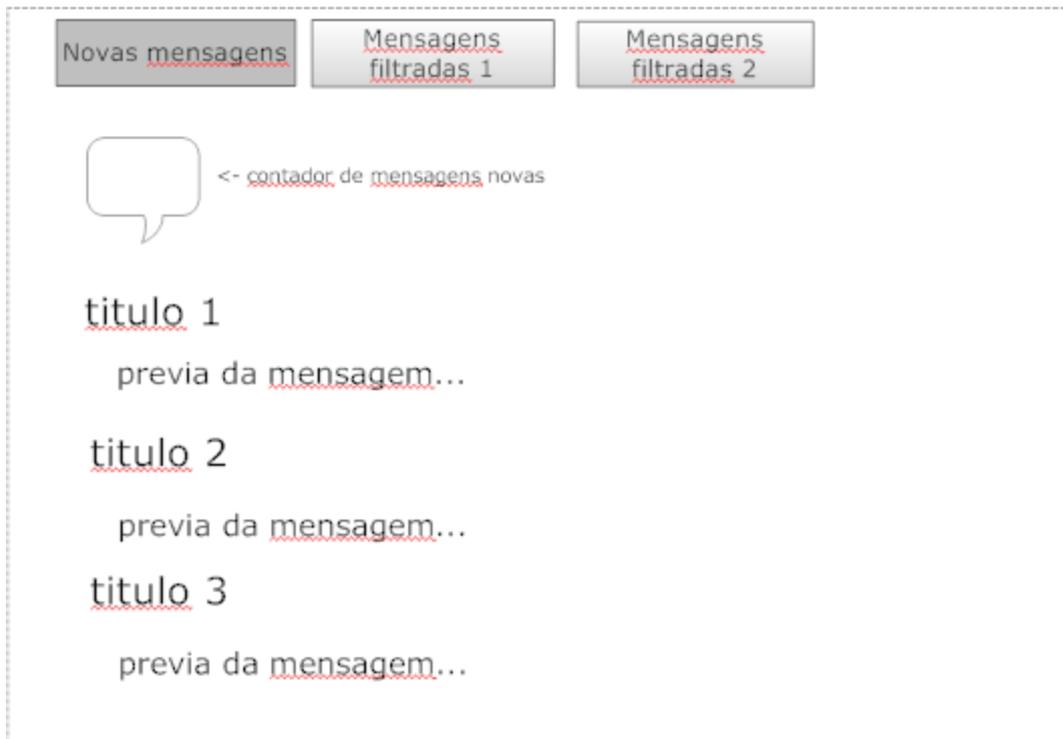


Figura 4: planejamento da página de aluno.
Fonte: Autoria própria.

Como pode ser observado na figura 4, buscou-se a simplicidade, dando-se prioridade às informações de interesse do aluno.

Na figura 5 pode-se ver o planejamento da interface com a qual tanto o usuário do tipo professor, quanto do tipo administrador irá interagir:



Figura 5: planejamento da página de professor ou administrador.
Fonte: Autoria própria.

3.2.2 Desenvolvimento da Interface.

Para o desenvolvimento da tela após o planejamento, existem algumas maneiras de montar o HTML antes de inserir os comandos de funcionamento, que no caso deste projeto será feito em PHP.

Foi utilizado o plugin DivShot para navegador *web* que constrói as divisões da página e insere os objetos que serão utilizados gerando o HTML básico e suas bibliotecas.

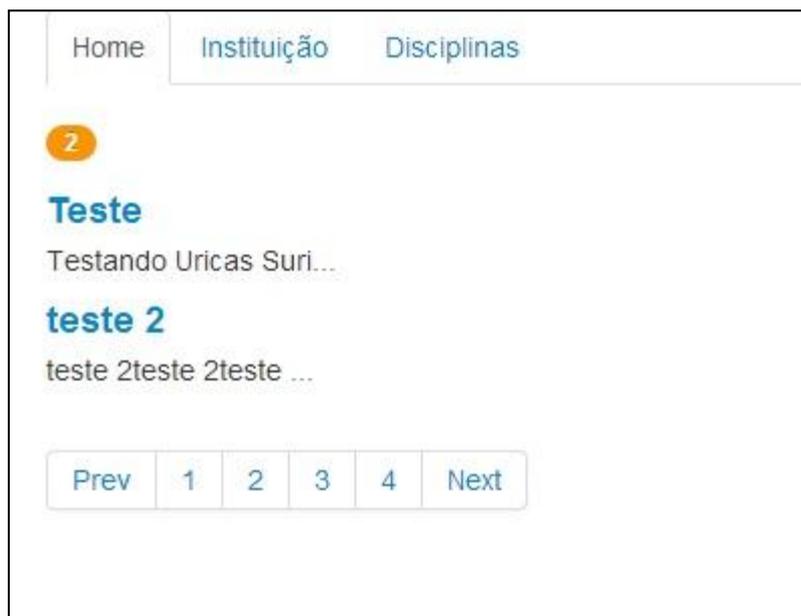
A partir do planejamento do *wireframe* as telas foram montadas de acordo com as figuras a seguir:



MobAluno

Figura 6: Página de login.
Fonte: Autoria própria.

Ao acessar o sistema o usuário se depara com a tela de *login* como demonstra a figura 6. Após garantir o acesso com o código e senha o usuário é encaminhado a tela principal:



Home **Instituição** Disciplinas

2

Teste
Testando Uricas Suri...

teste 2
teste 2teste 2teste ...

Prev 1 2 3 4 Next

Figura 7: Página do Aluno com mensagens novas.
Fonte: Autoria própria.

As figuras 7 e 8 representam respectivamente a visão do usuário na tela principal com ou sem mensagens novas.

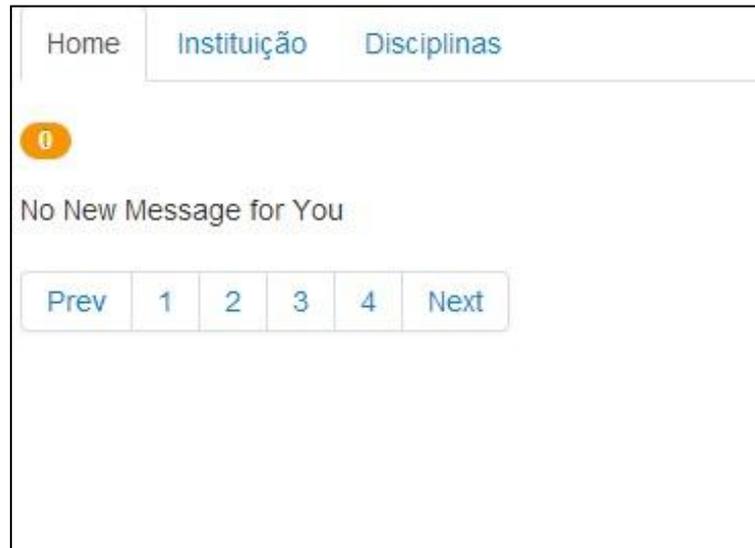


Figura 8: Página do Aluno sem mensagens novas.
Fonte: Autoria própria.

Como há opções de mensagem de remetentes variados, é possível filtrar as mensagens através do menu superior de qual remetente as mensagens serão trazidas, conforme exemplifica a figura 9.



Figura 9: Página do Aluno na aba de mensagens filtradas e lidas.
Fonte: Autoria própria.

A figura 10 demonstra como o usuário visualiza o conteúdo da mensagem.



Figura 10: Página de leitura da mensagem.

Fonte: Autoria própria.

Caso o usuário tenha conta de administrador ou de professor, a tela de envio de mensagens é liberada conforme mostra a figura 11:



Figura 11: Página do professor e administrador.

Fonte: Autoria própria.

A partir da página HTML estruturada, o desenvolvimento dos objetos inseridos é o próximo passo.

Utilizando o software livre PHP Editor foi trabalhado o comportamento dos objetos e quais resultados deveriam ser mostrados a partir das informações armazenadas no banco de dados.

3.3 FLUXOGRAMA

Para que o sistema seja considerado funcional, deve constar no banco de dados os valores assumidos e enviados através da página na internet. A figura 12 mostra o fluxo de funcionamento do sistema:

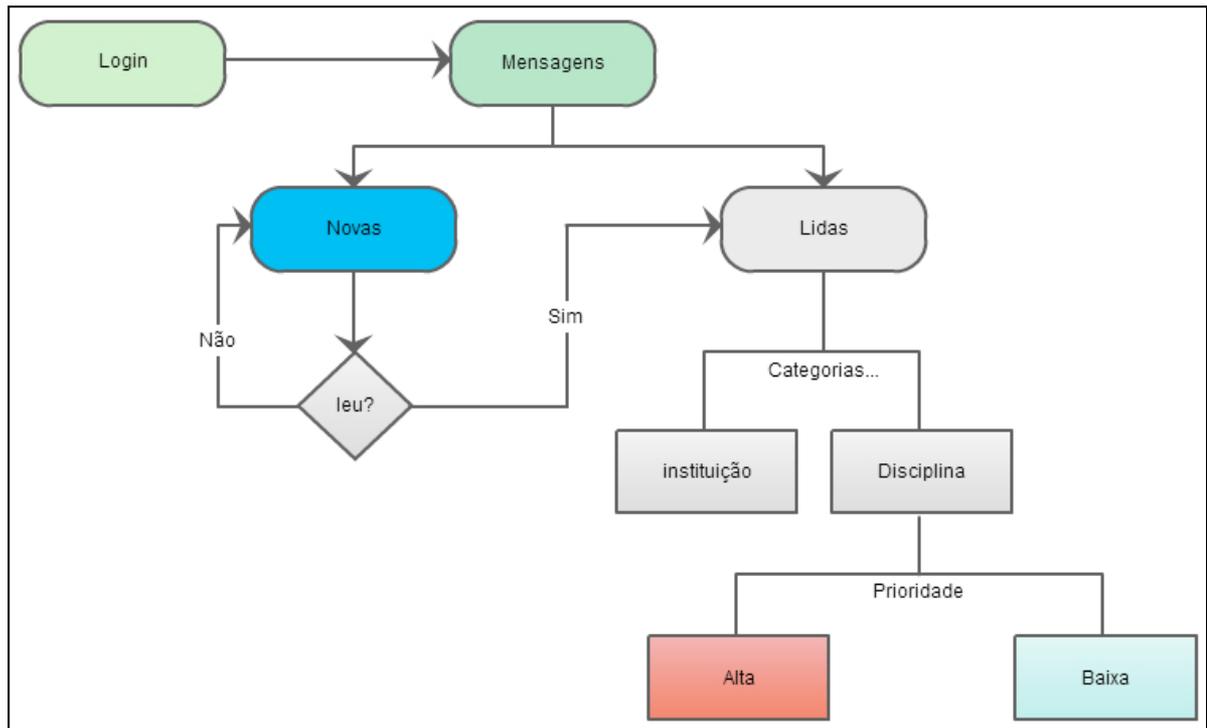
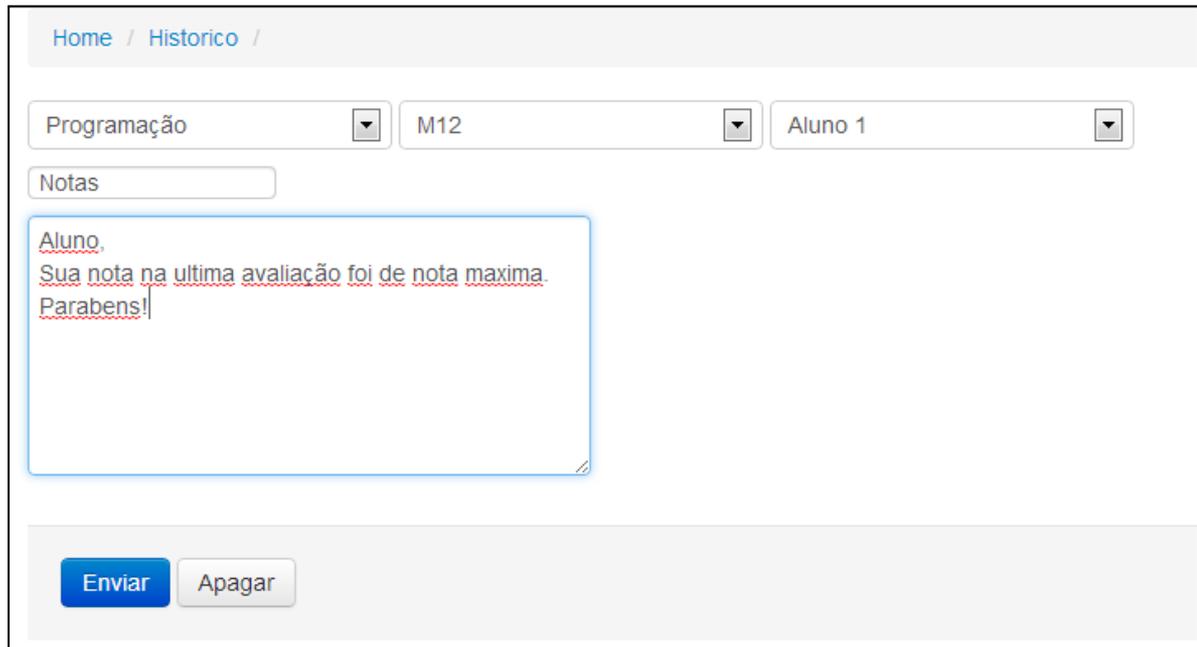


Figura 12: Fluxograma do sistema.
Fonte: Autoria própria.

Através do fluxograma pode-se entender a lógica de operação do sistema. No Login tem-se o início do funcionamento do software e, após a validação do login, o sistema apresentará as mensagens, que podem ser divididas em duas categorias: Novas ou Lidas. Se as mensagens foram novas, tem-se o questionamento se a mensagem nova foi lida e, em não sendo lida, o sistema permanece no estado de mensagens novas até que a validação seja positiva. Se as mensagens foram novas, ou se o sistema já tiver mensagens lidas, ele demonstra a gama de possibilidades destas mensagens separando-as por categorias, e finalmente as mensagens dentro de suas categorias são agrupadas em níveis de prioridade, alta ou baixa.

4 ANÁLISE DE RESULTADOS

Pode-se testar o funcionamento do sistema utilizando-se as figuras a seguir que simulam a inserção de dados pelo usuário.



Home / Historico /

Programação ▼ M12 ▼ Aluno 1 ▼

Notas

Aluno.
Sua nota na ultima avaliação foi de nota maxima.
Parabens!

Enviar Apagar

Figura 13: Inserção de dados na página do professor/administrador.
Fonte: Autoria própria.

Após a inserção do texto (figura 13) se escolhe o destinatário da mensagem, podendo ser para um aluno específico ou escolher grupos de destinatários, sendo uma turma ou uma matéria em que o professor leciona.

Para conferir se a mensagem foi salva corretamente pode-se conferir no banco de dados como, em destaque, mostra a mensagem enviada na figura 14:

MessageId	Sender	Title	Body	PriorityId
16	6	Notas	Aluno. Sua nota na ultima avaliação foi de nota maxima, Pz	1
15	9	teste 3	ja lida	1
6	5	teste 2	teste 2teste 2teste 2teste 2teste 2teste 2teste 2teste 2	1
5	5	Teste	Testando...	1

Figura 14: Conferindo os dados no banco na tabela de mensagens.
Fonte: Autoria própria.

Sabendo que a mensagem foi salva, o próximo passo é conferir se o número de identificação da mensagem em questão foi endereçada para o código identificador do destinatário (aluno), conforme mostra a figura 15, onde se pode observar também um campo de controle no banco de dados chamado Read que utiliza o sistema binário para informar se a mensagem foi lida.

MessageId	UserId	Read
16	4	0
15	4	1
6	4	0
5	4	0

Figura 15: Conferindo os dados no banco na tabela de destinatário.
Fonte: Autoria própria.

Tendo em vista que no banco de dados todas as informações estão salvas e configuradas corretamente, deve-se averiguar se a página de aluno está mostrando corretamente as informações que se encontram no banco de dados. Na figura 16 observa-se que a mensagem aparece na página principal do aluno já que é considerada uma mensagem nova (controlador Read = 0) e, portanto não-lida.

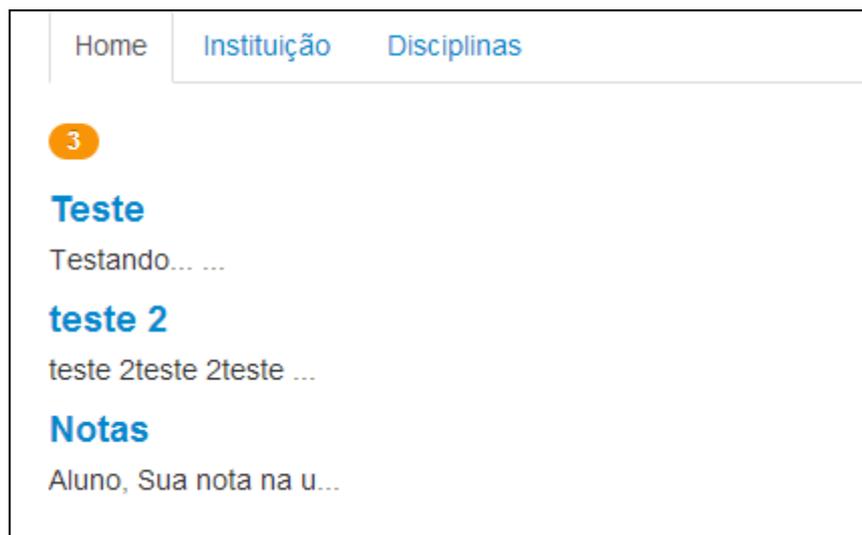


Figura 16: Mensagem nova na página principal do aluno.
Fonte: Autoria própria.

Quando o aluno acessar a mensagem em questão, o sistema deverá mostrar o conteúdo completo da mensagem e o remetente, conforme a figura 17:



Figura 17: Leitura da mensagem
Fonte: Autoria própria.

Quando a mensagem é aberta ao aluno o sistema altera no banco de dados o campo de controle *Read* demonstrando que a mensagem foi lida, como demonstra a figura 18:

The screenshot shows the MySQL-Front interface with the 'MessageTo' table selected in the Object Browser. The table data is as follows:

MessageId	UserId	Read
16	4	1
15	4	1
6	4	0
5	4	0

Figura 18: Conferindo se a mensagem foi lida.
Fonte: Autoria própria.

E na página principal as mensagens lidas não aparecem mais, como mostra a figura 19:



Figura 19: Mensagem lida não aparece na página principal.
Fonte: Autoria própria.

5 CONCLUSÃO

O principal objetivo estabelecido deste trabalho foi apresentar uma solução de acesso aos sistemas corporativos da universidade que fosse portátil e de fácil integração com serviços existentes.

Ao longo do desenvolvimento deste projeto, foi priorizada a maneira mais simples de captação de dados para que uma possível integração com sistemas educacionais complexos (com várias funções), pois na construção do banco de dados as tabelas e campos contêm informações que são usuais em sistemas dessa natureza e, portanto, podem ser exportadas de um banco de um sistema existente.

A flexibilidade do HTML permitiu construir um sistema cuja tela se adequasse às informações que estavam sendo exibidas, ou seja, o uso do navegador de internet do smartphone tornou o software mutável para se adequar à tela que está sendo exibida, assim utilizando o navegador do aparelho independente do sistema operacional e da marca do aparelho utilizado.

A estrutura desenvolvida no HTML permitiu também adaptações e formas de personalizar o sistema desenvolvido para que este tenha a identidade visual da instituição que virá a utilizá-lo, com logos, fontes, fotos do aluno ou professor.

Então, o projeto foi desenvolvido de maneira que possa ser utilizado em diversas instituições e que se integre tanto em forma operacional quanto visual.

Este projeto pode servir de ponto de partida para futuros trabalhos acadêmicos e, por isso, o código fonte foi disponibilizado nos anexos.

REFERÊNCIAS

Agence France-Presse (AFP) . Disponível em:

<<http://entretenimento.uol.com.br/noticias/afp/2013/04/26/venda-de-smarts-ja-supera-a-de-celulares-basicos-samsung-lidera.htm>>. Acesso em: 05 jun. 2013.

CARMO, Ana Lídia Lopes de. Gestão Educacional. Disponível em:

<<http://www.infoescola.com/educacao/gestao-educacional/>> Acesso em: 15 Jul. 2013

MOREIRA, Eduardo. O primeiro smartphone da história completou 20 anos do seu nascimento. Disponível em: <<http://targethd.net/o-primeiro-smartphone-da-historia-completou-20-anos-do-seu-nascimento/>> Acesso em: 03 Jun. 2013

MORIMOTO, Carlos. Smartphones, guia prático. Disponível em:

<<http://www.hardware.com.br/livros/smartphones/capitulo-plataformas.html>> Acesso em: 03 jun. 2013.

Revista da Educação. Informatização e Gestão Educacional –GED. Disponível em:

<<http://www.ici.curitiba.org.br/noticia.aspx?idf=556>> Acesso em: 15 Jul. 2013

W3SCHOOLS.COM. PHP 5 Tutorial. Disponível em:

<<http://www.w3schools.com/php/default.asp>> Acesso em: 16 Jul. 2013

WIKIPEDIA. Mobile App. Disponível em: <http://en.wikipedia.org/wiki/Mobile_app>

Acesso em: 10 Jul. 2013

_____. Software Aplicativo. Disponível em:

<http://pt.wikipedia.org/wiki/Software_aplicativo> Acesso em: 05 Jun. 2013.

_____. Web Application. Disponível em:

<http://en.wikipedia.org/wiki/Web_application> Acesso em: 10 Jul. 2013

ZAMBARDA. Pedro. História dos smartphones. Disponível em:

<<http://livetouch.blogspot.com.br/2010/06/historia-dos-smartphones.html>> Acesso em: 03 Jun. 2013

ANEXOS

ANEXO A - Index.php

```

<?session_start();?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Login</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta content="text/html; charset=UTF-8" http-equiv="Content-Type">
  <link rel="stylesheet" href="https://app.divshot.com/css/bootstrap.css">
  <link rel="stylesheet" href="https://app.divshot.com/css/bootstrap-responsive.css">
  <script src="https://app.divshot.com/js/jquery.min.js"></script>
  <script src="https://app.divshot.com/js/bootstrap.min.js"></script>
</head>
<body >
<?$_SESSION['autorizado']= 'nao';?>
<!--funÃ§Ã£o de verificar campos-->
<script language="javascript">
function Verifica()
{
  var login=document.inicial.login.value;
  if(login=="")
  {
    alert("Digite seu login!!");
    return false;
  }
}

```

```

    }

    var senha=document.inicial.senha.value;
    if(senha=="")
    {
        alert("Digite sua senha!!");
        return false;
    }
}
</script>
<center>
<br /><br /><br /><br />
<table>
<tr>
<td>
<br/>
<font size="12" color="#cc9900">
<center><b>Mob Aluno</b></center>
</font>
<br/>
</td>
</tr>
<tr>
<td>
<center>
<form name="inicial" method="post" action="home.php" onsubmit="return Verifica(this)">
<div class="well">
<form method="post" class="form-vertical">
<input type="text" class="input-medium" placeholder="Codigo do aluno"

```

```
id="login" name="login">  
<br>  
<input type="password" class="input-medium" placeholder="Senha" id="senha"  
name="senha">  
<br>  
<input type="submit" class="btn" id="confirma">  
<br>  
</button>  
</form>  
</div>  
</html>
```

ANEXO B - Conecta.php

```
<? $link = mysql_connect();  
if (!$link) {  
    die('Não conseguiu conectar: ' . mysql_error());  
}  
$db_selected = mysql_select_db( $link);  
if (!$db_selected) {  
    die ('Não pode selecionar o banco: ' . mysql_error());  
}?>
```

ANEXO C - Home.php

```

<?session_start();?>
<HTML>
<HEAD>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta content="text/html; charset=iso-8859-1" http-equiv="Content-Type">
  <link rel="stylesheet" href="https://app.divshot.com/css/bootstrap.css">
  <link rel="stylesheet" href="https://app.divshot.com/css/bootstrap-responsive.css">
  <script src="https://app.divshot.com/js/jquery.min.js"></script>
  <script src="https://app.divshot.com/js/bootstrap.min.js"></script>
</HEAD>
<BODY>
<!--funÃ§Ão de verificar campos-->
<script language="javascript">
function Clear()
{
    var login=document.inicial.login.value;
    if(login=="")
    {
        alert("Digite seu login!!");
        return false;
    }
    var senha=document.inicial.senha.value;
    if(senha=="")
    {
        alert("Digite sua senha!!");
        return false;
    }
}

```

```

    }
}
</script>
<?
include "conecta.php";
$conta_linha=0;
if ($_SESSION['autorizado']== 'nao')
{
    $login=$_POST['login'];
    //echo $login."<br/>";
    $senha=$_POST['senha'];
    //echo $senha."<br/>";
    $senha = addslashes($senha);
    $resultado=mysql_query("select UserName, UserType, UserId from User where
AppId='$login' and Password='$senha'") or die (mysql_error());
    $conta_linha=mysql_num_rows($resultado);
    // conta o total de linhas recuperadas
    $linha=mysql_fetch_array($resultado);
    $nome=$linha['UserName'];
    $_SESSION['tipo']=$linha['UserType'];
    $_SESSION['userid']=$linha['UserId'];
}
if( ($conta_linha == 1)||($_SESSION['autorizado']=='sim') )
{
    $_SESSION['autorizado']='sim';
    if ($_SESSION['tipo'] == 1) {
        $msgresult = mysql_query("select MessageId, Title, SUBSTRING(Body, 1,20) as Body
from Message where MessageId in (select MT.MessageId from MessageTo MT where
MT.UserId = '".$_SESSION['userid']."' and MT.Lido = 0)") or die (mysql_error());
        $msgrows = mysql_num_rows($msgresult);
    }
}
}
}

```

```

?>
<!doctype html>
<html>

<head>
  <title>Home</title>
  <meta name="viewport" content="width=device-width">
  <link rel="stylesheet" href="https://netdna.bootstrapcdn.com/twitter-bootstrap/2.3.2/css/bootstrap-combined.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/2.0.3/jquery.min.js" type="text/javascript"></script>
  <script type="text/javascript" src="https://netdna.bootstrapcdn.com/twitter-bootstrap/2.3.2/js/bootstrap.min.js"></script>
</head>
<body>
  <div class="container-fluid">
    <ul class="nav nav-tabs">
      <li class="active">
        <a href="/home.php">Home</a>
      </li>
      <li>
        <a href="/instituicao.php">Instituição</a>
      </li>
      <li>
        <a href="#">Disciplinas</a>
      </li>
    </ul>
    <span class="badge pull-left badge-warning"><? echo $msgrows; ?></span>
    <div class="row"></div>
  </div></div>

```

```

<div class="media">
<?
  If($msgrows == 0)
  {
    echo "No New Message for You";
  }
  Else
  {
    While ($msgarray = mysql_fetch_array($msgresult)){
?>
      <div class="media-body">
        <a href="/message.php?messageId=<? echo $msgarray['MessageId']?>"><h4
class="media-heading"><? echo $msgarray['Title']; ?></h4></a>
        <p><? echo $msgarray['Body']; ?>...</p>
      </div>
    <?
    }
  }
?>

</div>
</div>
<ul class="pager"></ul>
</body>
</html>
<?
}else{
?>
<!doctype html>

```

```

<html>

<head>
  <title>Home</title>
  <meta name="viewport" content="width=device-width">
  <link rel="stylesheet" href="https://netdna.bootstrapcdn.com/twitter-bootstrap/2.3.2/css/bootstrap-combined.min.css">
  <script type="text/javascript" src="https://ajax.googleapis.com/ajax/libs/jquery/2.0.3/jquery.min.js"></script>
  <script type="text/javascript" src="https://netdna.bootstrapcdn.com/twitter-bootstrap/2.3.2/js/bootstrap.min.js"></script>
</head>
<body>
  <div class="container">
    <ol class="breadcrumb">
      <li>
        <a href="#">Home</a> <span class="divider"></span>
      </li>
      <li>
        <a href="#">Historico</a> <span class="divider"></span>
      </li>
    </ol>
    <form name="send" method="post" action="home.php" onsubmit="Send(this)">
      <? $discipline = mysql_query("Select DisciplineId, Name from Discipline where DisciplineId in (select UD.DisciplineId from UserDisciplines UD where UD.UserId =".$linha["UserId"].")"); ?>
      <select name="Disciplina" value="Disciplina">
        <option selected value="">Disciplina</option>
        <? while ($disrow = mysql_fetch_array($discipline)) { ?>
          <option value="<?=$disrow['DisciplineId']?>"><? echo $disrow['Name']; ?></option>

```

```

<?
}
?>
</select>

<? $student = mysql_query("Select UserId, Name from User where UserId in (select
UC.CurrentId from UserCurrent UC where UD.UserId =".$linha['UserId']."); ?>

<select name="Disciplina" value="Disciplina">
  <option selected value="">Disciplina</option>
  <? while ($disrow = mysql_fetch_array($discipline)) { ?>
    <option value="<?=$disrow['DisciplineId']?>"><? echo $disrow['Name']; ?></option>
  <?
  }
?>
</select>

<div>
  <input type="text" class="input-medium" id="title" placeholder="Assunto:" value="">
</div>

<div>
  <textarea style="margin: 0px 0px 10px; width: 340px; height: 158px;" id="body"
placeholder="Texto:"></textarea>
</div>

</form>

<div class="form-actions">
  <button type="submit" class="btn btn-primary">Enviar</button>
  <input type="reset" class="btn" value="Apagar">
</div>

</div>

</body>

</html>

<?

```

```
}  
}  
else  
{  
?>  
<span style="color:#CC0000">Acesso não autorizado!</span>  
<?>  
?>  
</BODY>  
</HTML>
```

ANEXO D - Message.php

```

<?session_start();

include "conecta.php";

$msgresult = mysql_query("select Title, Body, Sender from Message where MessageId
='".$_GET['messageId'].'"") or die (mysql_error());

$query = mysql_query("UPDATE MessageTo SET Lido='1' WHERE MessageId
='".$_GET['messageId'].'"");

$msggarray = mysql_fetch_array($msgresult);

$msggsender = mysql_query("select UserName from User where UserId
='".$_msggarray['Sender'].'"");

$msggfrom = mysql_fetch_array($msggsender);?>

<!doctype html>

<html>

<head>

<title>Mensagem</title>

<meta name="viewport" content="width=device-width">

<link rel="stylesheet" href="https://netdna.bootstrapcdn.com/twitter-
bootstrap/2.3.2/css/bootstrap-combined.min.css">

<script type="text/javascript"
src="https://ajax.googleapis.com/ajax/libs/jquery/2.0.3/jquery.min.js"></script>

<script type="text/javascript" src="https://netdna.bootstrapcdn.com/twitter-
bootstrap/2.3.2/js/bootstrap.min.js"></script>

</head>

<body>

<h3 class="text-center"><? echo $msggarray['Title']; ?></h3>

<h4 class="text-center">Remetente: <? echo $msggfrom['UserName']; ?></h4>

<div class="container">

<div class="well"><? echo $msggarray['Body']; ?></div>

</div>

<a class="btn" href="/home.php">Voltar</a>

```

```
</body>
```

```
</html>
```