**UNIVERSIDADE PAULISTA**

**Diego Felipe Martins Mendes RA N1268J0**

**Jayme Rossini Júnior RA D3840A4**

**UniSearch**

Sistema de localização de salas e cadastro de dados dos cursos.

**SOROCABA – SP**

**2018**

**Diego Felipe Martins Mendes RA N1268J0**

**Jayme Rossini Júnior RA D3840A4**

**UniSearch**

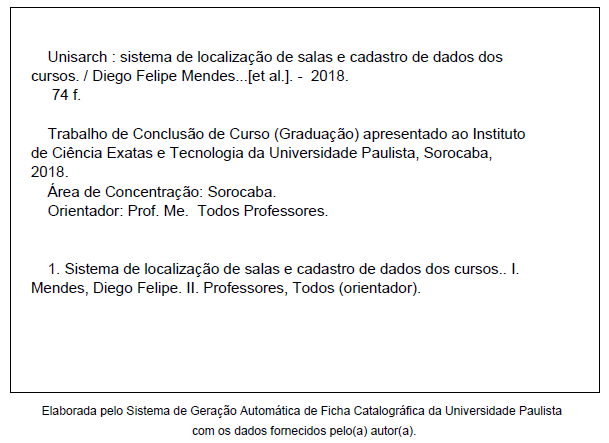
Sistema de localização de salas e cadastro de dados dos cursos.

Trabalho de conclusão de semestre de curso de Analise de desenvolvimento de sistema apresentado à Universidade Paulista

Orientadores: Todos os professores do semestre.

**SOROCABA – SP**

**2018**



**Diego Felipe Martins Mendes RA N1268J0**

**Jayme Rossini Júnior RA D3840A4**

**UniSearch**

Sistema de localização de salas e cadastro de dados dos cursos.

Trabalho de conclusão de semestre de curso de Analise de desenvolvimento de sistema apresentado à Universidade Paulista

Orientadores: Todos os professores do semestre.

Aprovado em:

BANCA EXAMINADORA  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_/\_\_\_  
Prof: Richardson Kennedy Luz  
Universidade Paulista  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_/\_\_\_

Prof: Eduardo Martins Simi  
Universidade Paulista  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_/\_\_\_

Prof: Marcos Domingues da Cruz  
Universidade Paulista  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_/\_\_\_

Prof: Randal Gasparini  
Universidade Paulista

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_/\_\_\_

Prof: Reverdan Almeida Sparinger

Universidade Paulista

**RESUMO**

O propósito da UniSearch é desenvolver um projeto, que proverá de um aplicativo mobile de busca de salas e uma página administrativa web, para cadastramento de cursos e consulta via banco de dados. Devido a dificuldades de localização, esse aplicativo terá como público alvo os estudantes calouros e visitantes, já a página web será destinada para o uso de profissionais da instituição. Para a composição do software, foi utilizado a padronização NBR ISO 9126, uma norma que tem como objetivo avaliar a qualidade, bem como as suas características, sub características e atributos. O aplicativo foi desenvolvido utilizando as linguagens de marcação de texto, HTML5, CSS3 e JavaScript e as linguagens de programação PHP e banco de dados em MySQL.

Palavra-Chave: Aplicativo. Usuário. Linguagens.

**ABSTRACT**

The UniSearch purpose is to develop a project to provide a mobile application to search the rooms and a webpage administrative, for registration of courses and consultation by database. This application market share are the freshman students and visitors due to difficulties of location. Already the webpage will be destined to the institution professionals. The software composition was based on the NBR ISO 9126, a norm that has as objective to evaluate the quality, as well as its characteristics, sub characteristics and attributes. The application was developed using text markup, HTML5, CSS3 and JavaScript languages and programming languages PHP and MySQL database.

Keyword: Application. User. Languages.

**SUMÁRIO**

[1.0 INTRODUÇÃO 15](#_Toc514600752)

[2.0 UNISEARCH 17](#_Toc514600753)

[2.1 Aplicativo 17](#_Toc514600754)

[2.2 Página administrativa web 18](#_Toc514600755)

[3.0 FERRAMENTAS UTILIZADAS PARA O DESENVOLVIMENTO 21](#_Toc514600756)

[3.1 Notepad ++ 21](#_Toc514600757)

[3.2 Wampserver 21](#_Toc514600758)

[3.3 PhoneGap 21](#_Toc514600759)

[4.0 LINGUAGENS UTILIZADAS 23](#_Toc514600760)

[4.1 HTML 23](#_Toc514600761)

[4.2 CSS 23](#_Toc514600762)

[4.3 JAVASCRIPT 23](#_Toc514600763)

[4.4 PHP 24](#_Toc514600764)

[4.5 SQL 24](#_Toc514600765)

[5.0 MODELO DE PROCESSO DE ENGENHARIA DE SOFTWARE 25](#_Toc514600766)

[5.1 Objetivos da prototipação evolucionária e descartável 25](#_Toc514600767)

[5.2 Pontos negativos da prototipação 25](#_Toc514600768)

[5.3 Pontos positivos da prototipação 25](#_Toc514600769)

[6.0 CASO DE USO FORMAL 27](#_Toc514600770)

[6.1 Casos de uso 27](#_Toc514600771)

[7.0 DIAGRAMA DE CASO DE USO 31](#_Toc514600772)

[8.0 ISO 9126 33](#_Toc514600773)

[8.1 Funcionalidade 33](#_Toc514600774)

[8.1.1 Adequação 33](#_Toc514600775)

[8.1.2 Acurácia 35](#_Toc514600776)

[8.1.3 Segurança 35](#_Toc514600777)

[9.0 BANCO DE DADOS 37](#_Toc514600778)

[9.1 Modelo conceitual 38](#_Toc514600779)

[9.2 Modelo lógico 38](#_Toc514600780)

[9.3 Modelo físico 39](#_Toc514600781)

[10.0 HEURÍSTICAS DE NIELSEN 41](#_Toc514600782)

[10.1 As 10 heurísticas de Nielsen no projeto UniSearch 41](#_Toc514600783)

[10.1.1.1 Visibilidade de qual estado estamos no sistema 41](#_Toc514600784)

[10.1.1.2 Correspondência entre o sistema e o mundo real 42](#_Toc514600785)

[10.1.1.3 Liberdade de controle fácil para o usuário 42](#_Toc514600786)

[10.1.1.4 Consistência e padrões 42](#_Toc514600787)

[10.1.1.5 Prevenções de erros 43](#_Toc514600788)

[10.1.1.6 Reconhecimento em vez de memorização 43](#_Toc514600789)

[10.1.1.7 Flexibilidade e eficiência de uso 43](#_Toc514600790)

[10.1.1.8 Estética e design minimalista 44](#_Toc514600791)

[10.1.1.9 Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros 44](#_Toc514600792)

[10.1.1.10 Ajuda e documentação 45](#_Toc514600793)

[11.0 CONCLUSÃO 47](#_Toc514600794)

[12.0 REFERÊNCIAS 49](#_Toc514600795)

[APÊNDICE 51](#_Toc514600796)

[APÊNCIDE A – Código do aplicativo 53](#_Toc514600797)

[Index.html 53](#_Toc514600798)

[Funções.js 55](#_Toc514600799)

[APÊNDICE B – Código página web administrativa 59](#_Toc514600800)

[Connect.php 59](#_Toc514600801)

[Function.php 59](#_Toc514600802)

[Form-login.php 60](#_Toc514600803)

[Login.php 61](#_Toc514600804)

[Addusuario.php 62](#_Toc514600805)

[Logout.php 63](#_Toc514600806)

[Indexx.php 63](#_Toc514600807)

[Check.php 64](#_Toc514600808)

[Cadastrocurso.php 64](#_Toc514600809)

[Addinfo.php 66](#_Toc514600810)

[Consulta.php 67](#_Toc514600811)

[Form-edit.php 70](#_Toc514600812)

[Edit.php 73](#_Toc514600813)

# 1.0 INTRODUÇÃO

UniSearch é um projeto que fornece aplicativo mobile e uma página administrativa.

O aplicativo mobile que será utilizado para localização de salas dentro da universidade, para orientar os alunos e visitantes, a se localizar, de forma ágil e eficaz.

A página administrativa web terá a função alimentar o aplicativo com dados sobre a localização de salas. Há uma grande dificuldade para os alunos calouros, que não tem o conhecimento do ambiente da universidade, para localizar as salas que compõem o campus.

Além de minimizar a dificuldade o software também proporcionará uma maior autonomia e liberdade dos alunos e visitantes, dentro da universidade.  
 O aplicativo utilizado, será disponibilizado para smartphones e tablets, com acesso de fácil compreensão, através de uma linguagem simples.

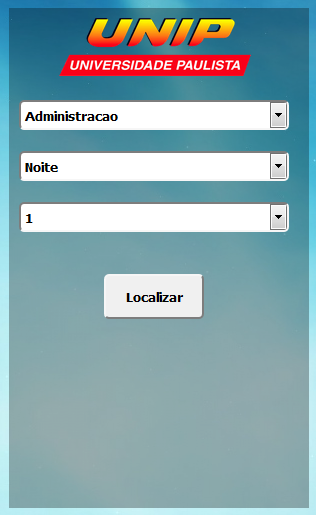
# 2.0 UNISEARCH

UniSearch é um projeto que tem por objetivo auxiliar os usuários a receber informações fornecidas pelos profissionais encarregados pela administração do sistema, assim viabilizando de forma rápida e concisa informações claras e precisas sobre as salas.

Para o desenvolvimento do projeto foram utilizadas as linguagens de marcação de texto HTML, CSS e JAVASCRIPT, em conjunto com as linguagens de programação PHP e SQL, os frameworks usados foram SUBLIME TEXT 3, NOTEPAD++, WAMPSERVER e PHONEGAP.

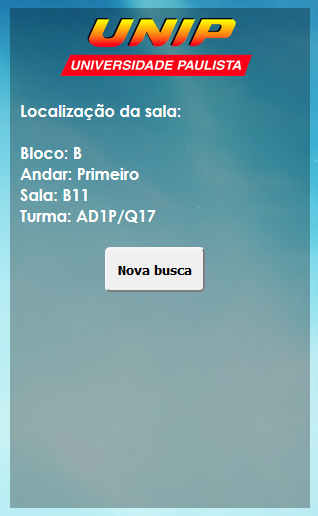
## 2.1 Aplicativo

O objetivo de aplicativo mobile visa ajudar o usuário a localizar sua sala dentro da universidade, devido à dificuldade dos calouros em início de semestre e a constante mudança de sala, fundamentalmente dá acesso a informações necessárias aos usuários para que consigam encontrar suas salas.



Fonte– Criação própria 2018.

Imagem 1 – Tela inicial (Aplicativo mobile).



Fonte –Criação própria 2018.

Imagem 2 – Tela Resultado (Aplicativo mobile).

## 2.2 Página administrativa web

Foi desenvolvida a página web com o intuito de cadastrar dados das localizações de sala por curso, período e semestre para, alimentar o aplicativo mobile.

O mesmo foi desenvolvido para auxiliar os administradores na entrada, saída e consulta dos dados.

Para a segurança dessas informações tem implementado em si o sistema, um login para o usuário administrador.





Fonte –Criação própria 2018

Imagem 4 – Tela Inicial (Página web).

Imagem 3 – Tela Login (Página web).

Fonte –Criação própria 2018



Imagem 3 – Tela Cadastramento de cursos (Página web).

Fonte –Criação própria 2018

Imagem 3 – Tela Consulta de cursos cadastrados (Página web).



Fonte –Criação própria 2018

# 3.0 FERRAMENTAS UTILIZADAS PARA O DESENVOLVIMENTO

O desenvolvimento do layout para aplicativo mobile foi baseado em um formulário simples (Dynamic dependent select box) de busca por filtragem, sendo assim de fácil compreensão para o usuário que utilizará o sistema, o usuário seleciona seu curso, período e semestre para obter a localização de sua sala.

De mesmo modo o layout da página web comporta um design com cores suaves e que facilitam a visualização do usuário influenciando diretamente na praticidade de acesso.

## 3.1 Notepad ++

É um editor de texto e de código fonte de código aberto sob a licença GPL. Suporta várias linguagens de programação rodando sob o sistema Microsoft Windows (possível utilização no Linux via Wine). Foi utilizado para o desenvolvimento do código fonte do front-end da página web que será utilizada pela secretária para cadastro de alunos no software e para desenvolvimento do layout do aplicativo e suas funcionalidades.

## 3.2 Wampserver

É um software publicado sob a GNU General Public License desenvolvido pela PHP Team. É usado para instalar rapidamente no computador os softwares PHP 5, MySQL e Apache, disponibilizando suporte ao uso de scripts PHP localmente no Windows. Foi utilizado para gerar um servidor local para suportar um banco de dados com as informações de usuários e cursos cadastrados para a apresentação do projeto.

## 3.3 PhoneGap

Framework para o desenvolvimento de aplicações nativas de sistemas operativos móveis, fazendo uso de tecnologias web como HTML5, CSS3 e JavaScript. Utilizado para a compilação dos códigos de forma que gere o software a ser desenvolvido.

# 4.0 LINGUAGENS UTILIZADAS

Linguagens de programação são **padrões de codificação binária**, com sintaxe e semânticas específicas. Desta forma, capazes de criar instruções para máquinas. Graças a esses conjuntos de códigos e recursos, é possível criar programas e sistemas para resolver os mais diversos problemas do cotidiano.

## 4.1 HTML

HTML é uma das linguagens que utilizamos para desenvolver websites e aplicativos. A sigla HTML vem do inglês e significa Hypertext Markup Language ou em português Linguagem de Marcação de Hipertexto.

## 4.2 CSS

O Cascading Style Sheets (CSS) é uma "folha de estilo" composta por “camadas” e utilizada para definir a apresentação (aparência) em páginas da internet que adotam para o seu desenvolvimento linguagens de marcação (como XML, HTML e XHTML). O CSS define como serão exibidos os elementos contidos no código de uma página da internet e sua maior vantagem é efetuar a separação entre o formato e o conteúdo de um documento.

## 4.3 JAVASCRIPT

JavaScript é uma linguagem de programação interpretada, foi originalmente implementada como parte dos navegadores web para que scripts pudessem ser executados do lado do cliente e interagissem com o usuário sem a necessidade deste script passar pelo servidor, controlando o navegador, realizando comunicação assíncrona e alterando o conteúdo do documento exibido. É atualmente a principal linguagem para programação client-side em navegadores web.

## 4.4 PHP

O PHP (um acrônimo recursivo para PHP: Hypertext Preprocessor) é uma linguagem de script open source de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento web e que pode ser embutida dentro do HTML.

## 4.5 SQL

A linguagem SQL (Struct Query Language) é uma linguagem padrão para a manipulação de dados dentro de um SGBD. Desenvolvida pela IBM, possibilita o armazenamento, organização, atualização e exclusão de informações dentro do banco de dados.

# 5.0 MODELO DE PROCESSO DE ENGENHARIA DE SOFTWARE

O modelo de processo de engenharia de software utilizado para o desenvolvimento do software é o modelo de prototipação, pois o foco é compreender e ajudar os clientes e desenvolvedores entender os requisitos para o sistema. Os benefícios da prototipação é a melhoria na facilidade de uso do sistema com uma maior aproximação do sistema com as necessidades dos usuários que possua uma melhoria da qualidade do projeto e na facilidade de manutenção assim reduzindo o esforço no desenvolvimento.

## 5.1 Objetivos da prototipação evolucionária e descartável

O objetivo da prototipação evolucionária é fornecer aos usuários finais um sistema funcionando, o desenvolvimento começa com aqueles requisitos que são melhores compreendidos.

O objetivo da prototipação descartável é validar ou derivar os requisitos do sistema.

## 5.2 Pontos negativos da prototipação

O cliente não considera o software, pois pode haver uma confusão do protótipo com a versão final do produto.

Baixa qualidade do desenvolvimento, retrabalho e auto custo de desenvolvimento.

## 5.3 Pontos positivos da prototipação

Auxilia no entendimento dos requisitos que estão mal compreendidos, pelo cliente e tem uma taxa de desenvolvimento rápido de um projeto.

Ainda que possam ocorrer problemas, a prototipação é um ciclo muito eficiente, o cliente e o desenvolvedor ambos devem concordar que o protótipo seja construído para servir como um mecanismo a fim de definir os requisitos.

# 6.0 CASO DE USO FORMAL

Os casos de uso são os veículos de captura de requisitos e servem de base para a definição de requisitos funcionais. Além de facilitarem a visualização da aplicação, ajudam na delimitação do sistema. Servem como meio de comunicação com usuários finais e clientes por oferecerem uma visão dinâmica e fechada do sistema.

Os casos de uso ajudam a dimensionar a capacidade de processamento necessária. São utilizados também para a definição dos testes (case test), como base para os testes de integração, e como fonte para a documentação de usuários e manuais.

## 6.1 Casos de uso

|  |  |
| --- | --- |
| **CASO DE USO CONSULTA DOS CURSOS** | |
| **Nome do caso:** | Consulta dos cursos. |
| **Atores:** | Funcionários da secretaria |
| **Oque os atores esperam:** | Os funcionários esperam, realizar o login na página web, e visualizar os dados dos cursos. |
| **Descrição do processo normal:** | Funcionário loga na página web, consultar os dados do curso. |
| **Descrição dos possíveis erros:** | Não ter o curso cadastrado no sistema. |
| **Caso alternativos:** | Realizar o cadastro do curso desejado. |
| **Resultado final do caso:** | Consulta realizada com sucesso. |

|  |  |
| --- | --- |
| **CASO DE USO ALTERAÇÃO DOS DADOS DO CURSO** | |
| **Nome do caso:** | Alteração dos dados do curso |
| **Atores:** | Funcionários da secretaria |
| **Oque os atores esperam:** | Os funcionários esperam, realizar o login na página web, visualizar e alterar os dados dos cursos. |
| **Descrição do processo normal:** | Funcionário loga na página web, consultar os dados do curso, realizar a alteração do mesmo. |
| **Descrição dos possíveis erros:** | Aplicar dados errados no processo de alteração, deixar campos vazios |
| **Caso alternativos:** | Realizar nova consulta para editar. |
| **Resultado final do caso:** | Alteração de dados concluída. |

|  |  |
| --- | --- |
| **CASO DE USO ALTERAÇÃO DOS DADOS DO CURSO** | |
| **Nome do caso:** | Alteração dos dados do curso |
| **Atores:** | Funcionários da secretaria |
| **Oque os atores esperam:** | Os funcionários esperam, realizar o login na página web, visualizar e alterar os dados dos cursos. |
| **Descrição do processo normal:** | Funcionário loga na página web, consultar os dados do curso, realizar a alteração do mesmo. |
| **Descrição dos possíveis erros:** | Aplicar dados errados no processo de alteração, deixar campos vazios |
| **Caso alternativos:** | Realizar nova consulta para editar. |
| **Resultado final do caso:** | Alteração de dados concluída. |

|  |  |
| --- | --- |
| **CASO DE USO CADASTRO DOS DADOS CURSOS** | |
| **Nome do caso:** | Cadastro dos dados cursos |
| **Atores:** | Funcionários da secretaria |
| **Oque os atores esperam:** | Os funcionários esperam, realizar o login na página web, para cadastra dados de acordo com o curso desejado. |
| **Descrição do processo normal:** | Funcionário loga na página web, seleciona as informações do curso, período, bloco, andar e adiciona manualmente as informações sala e turma. |
| **Descrição dos possíveis erros:** | Cadastra informações incorretas, campos vazios. |
| **Caso alternativos:** | Realizar alterações. |
| **Resultado final do caso:** | Cadastro realizado com sucesso. |

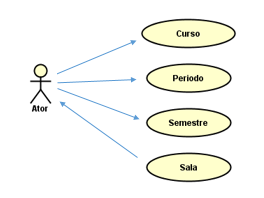
|  |  |
| --- | --- |
| **CASO DE USO LOCALIZAÇÃO DE SALA** | |
| **Nome do caso:** | Localização de sala |
| **Atores:** | Alunos |
| **Oque os atores esperam:** | Os alunos esperam, realizar uma busca de sala utilizando a aplicativo mobile. |
| **Descrição do processo normal:** | O aluno selecionar curso, período e semestre desejado, para exibir informações do curso. |
| **Descrição dos possíveis erros:** | Informa dados errados. |
| **Caso alternativos:** | Realizar nova busca. |
| **Resultado final do caso:** | Informação do curso informado com sucesso. |

# 7.0 DIAGRAMA DE CASO DE USO

Diagrama de Casos de Uso tem o objetivo de auxiliar a comunicação entre os analistas e o cliente.

O diagrama de caso de uso descreve um cenário que mostra as funcionalidades do sistema do ponto de vista do usuário é uma excelente ferramenta para o levantamento dos requisitos funcionais do sistema.

O diagrama abaixo representa o caso de uso de localização de sala.



Fonte: Criação própria, 2018.

Imagem 1 – Diagrama caso de uso localização de sala.

# 8.0 ISO 9126

ISO 9126, uma norma que tem como objetivo avaliar a qualidade, bem como as suas características, sub características e atributos. O objetivo é padronizar a avaliação da qualidade do software, um produto final com qualidade é uma necessidade, uma vez que desde o levantamento de requisitos até a fase de produção, as necessidades do cliente devem estar em evidência, bem como o feedback para melhoria contínua dos processos da organização.

Foram realizadas alterações nas características e sub características com base na ISO 9126.

## 8.1 Funcionalidade

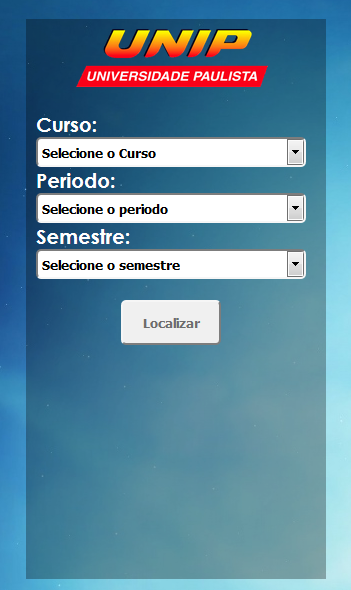
Por definição funcionalidade é a capacidade do produto de software de prover funções que atendam às necessidades explícitas e implícitas, quando o software estiver sendo utilizado sob condições especificadas.

### 8.1.1 Adequação

Com a base na ISO 9126, foi alterado o modo busca de sala do aplicativo que estava como “turma”, pois não estava compatível com a necessidade do usuário, devido os alunos não lembrarem a turma que cursava. Novo modo de busca de sala, o aluno seleciona curso, período e semestre, conforme imagem 1.

Alterado a página inicial de (cadastro curso), pois anteriormente o usuário era direcionado direto para pagina onde realiza os cadastros das informações dos cursos, foi acrescentada na página dois links de redirecionamento, para acessar página de cadastro ou consulta de cadastro, conforme imagem 2.

Alteração na tela de consulta de cursos, inserido filtros para cada coluna de dados cadastrados na página, modificações no design geral, conforme imagem 3.



Fonte: Criação própria, 2018.

Imagem 1 – Tela inicial (aplicativo).



Imagem 2 – Tela inicial (página web).

Fonte: Criação própria, 2018.



Imagem 3 – Tela consulta de cursos filtro (página web).

Fonte: Criação própria, 2018.

### 8.1.2 Acurácia

O software apresentava informações precisas, porem devido a mudança de busca por sala e à falta de adequação do público (usuário), tivemos que alterar o mesmo, ainda assim a acurácia permanece precisa.

### 8.1.3 Segurança

Foi inserido uma tela com um formulário de login e de cadastro de usuários, na página web da secretária (cadastro do curso), pois o mesmo não possuía nenhum bloqueio para impedir pessoas não autorizadas realizar a utilização do mesmo, conforme imagem 4.



Imagem 4 – Tela de login e cadastro (página web).

Fonte: Criação própria, 2018.

# 9.0 BANCO DE DADOS

Bancos de dados, são coleções de informações que se relacionam de forma que crie um sentido. São de vital importância para empresas, e há duas décadas se tornaram a principal peça dos sistemas de informação.

O grande objetivo de um banco de dados é oferecer dados abstratos aos usuários. A forma como estes dados estão armazenados e mantidos não interessa aos usuários, mas a disponibilidade eficiente destes dados é que são fundamentais.

Existem vários SGBD (Sistema Gerenciador de Banco de Dados), no mercado, alguns são pagos, outros gratuitos, possui programas que criam uma interface gráfica, gerando um código em SQL automaticamente.

Alguns SGBD:

* SQLServer: Um dos maiores SGBD do mundo, sob licença da Microsoft, tem versões pagas e gratuitas.
* MySQL: O MySQL é um software livre, com código fonte aberto e uso gratuito.
* FirebirdSQL:  Roda na maioria dos sistemas Unix, e tem código fonte aberto.
* mSQL:  Criado pela Hughes Technologies Pty Ltd., trabalha mais com o uso eficiente da memória, e é um sistema pequeno. Sua licença é altamente controlada pela empresa dona do produto.
* Microsoft Access: é um SGBD da Microsoft que acompanha o pacote Office. É muito utilizado para a aprendizagem e tem poucas atribuições profissionais, devido a sua limitação muito grande em armazenamento.

Para armazenar um dado, é necessário criar tabelas, dentro das tabelas são criadas colunas, onde as informações são armazenadas. Para os dados da base de dados ficarem organizados, devem ser criadas tabelas que não misturem informações.

## 9.1 Modelo conceitual

Refere-se ao desenvolvimento de um modelo inicial da base de dados que reflitam as necessidades do usuário.

Essa modelagem preocupa-se em descrever quais dados serão armazenados na base de dados e quais dados se relacionam.

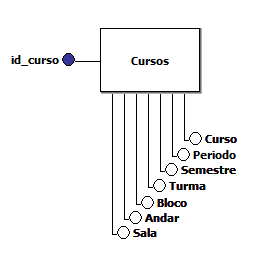
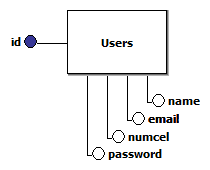
Para fazer o modelo conceitual é necessário entender que dados o usuário final espera que o sistema disponibilize.

A modelagem conceitual fornece uma visão mais próxima do modo como os usuários visualizam os dados realmente.

## 9.2 Modelo lógico

O projeto lógico consiste no mapeamento do esquema conceitual para o modelo de dados do SGBD adotado.

Isso significa transformar o modelo conceitual obtido na primeira fase num modelo mais próximo da implementação, ou seja, fornece uma visão mais detalhada do modo como os dados são armazenados no computador.



Fonte: Criação própria, 2018.

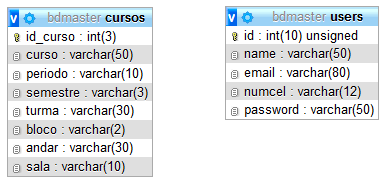
Imagem 1 – Esquema lógico banco de dados.

## 9.3 Modelo físico

O projeto físico inicia-se a partir do modelo lógico e resulta em um modelo físico. O armazenamento físico do banco de dados, o nível mais baixo de abstração, como os dados estão realmente armazenados.

Um esquema físico é uma descrição da implementação do bando de dados e é direcionado para um SGBD específico.

Imagem 1 – Esquema físico banco de dados.



Fonte: Criação própria, 2018.

# 10.0 HEURÍSTICAS DE NIELSEN

É um método de avaliação de usabilidade onde um avaliador procura problemas de usabilidade numa interface com o usuário através da análise e interpretação de um conjunto de princípios ou heurísticas. Este método de avaliação é baseado no julgamento do avaliador.

Nestas avaliações, um pesquisador age como o usuário-alvo e tenta completar uma série de tarefas pré-determinadas no site desenvolvido. Por exemplo, tentar comprar um produto, tentar fazer um pagamento de um produto ou até mesmo encontrar informações importantes (como telefone, endereço ou e-mail para contato).

Pensando nesses problemas o Jakob Nielsen, criou o que chamamos de: 10 heurísticas de Nielsen.

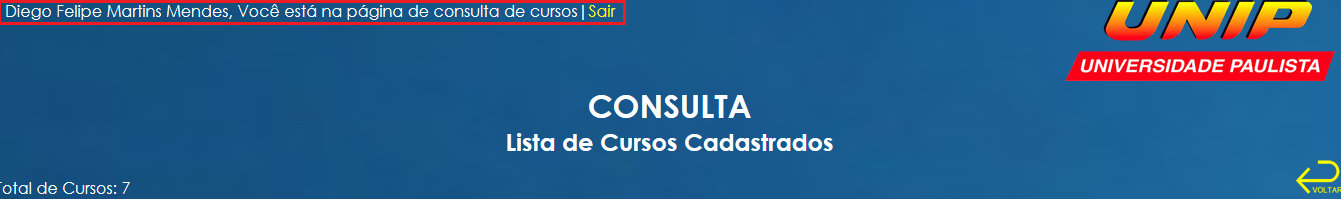
## 10.1 As 10 heurísticas de Nielsen no projeto UniSearch

### 10.1.1.1 Visibilidade de qual estado estamos no sistema

No geral nossos clientes são muito dependentes da visão, ainda assim, quando estamos em um ambiente físico temos outros sentidos que nos informam o que está acontecendo. Em ambientes digitais a dependência da visão é ainda maior e por conta disso é essencial que a interface forneça ao usuário o status em relação à sua posição dentro do sistema, ou seja, informar o usuário sobre qual ambiente ele estava, em qual ele está e para quais outros ambientes ele poderá se dirigir a partir de sua localização.

Na página administrativa, após o login o usuário é informado por uma mensagem na parte superior a esquerda, em qual pagina ele se encontra.

Imagem 1 – Tela de consulta (página web).



Fonte: Criação própria, 2018.

### 10.1.1.2 Correspondência entre o sistema e o mundo real

Um sistema precisa falar a mesma linguagem do usuário, não estou falando apenas do idioma em si, mas da linguagem que o usuário utiliza no dia-a-dia.

Outro aspecto da compatibilidade entre sistema e mundo real são os símbolos utilizados dentro de uma interface. Parece algo sem muita importância mas escolher corretamente quais ícones serão utilizados para colocar em uma interface pode facilitar ou prejudicar a compreensão das informações.

### 10.1.1.3 Liberdade de controle fácil para o usuário

No momento em que o usuário esquece de preencher todos os dados do curso e acaba clicando em “cadastra curso”, nesse momento o sistema identificara que ainda a dados a serem preenchido, dessa forma o sistema exibira uma mensagem: “Volte e preencha todos os campos, aguarde 5 segundos”.

O ponto aqui é que quando o usuário realiza ações por engano, algo que é frequente, o sistema deve apresentar ao usuário uma “saída de emergência”, essa possibilidade de reverter ações remove a insegurança do usuário ao utilizar a página.

### 10.1.1.4 Consistência e padrões

Manter consistência entre as telas de uma aplicação é essencial para que não seja necessário o entendimento de vários padrões e formas de interações diferentes para cada tela, uma vez aprendido será algo replicável em outros contextos. Além disso a experiência de uso se torna muito mais interessante pois não existirá aquela sensação de estar perdido. Muitas vezes o motivo pelo qual usuários não interagem com as aplicações é essa sensação causada pela falta de consistência e padronização.

### 10.1.1.5 Prevenções de erros

Os usuários normalmente cometem um deslize, é quando um usuário pretende realizar uma ação, mas acaba realizando outra, isso acontece tipicamente quando ele não está totalmente focado em sua ação dentro da aplicação.

Portanto melhor do que mensagens dizendo que o usuário cometeu algum erro é prevenir que o mesmo não cometa esse erro. Caixas de confirmação, como as que aparecem quando você deleta um curso, são um exemplo de como evitar erros.

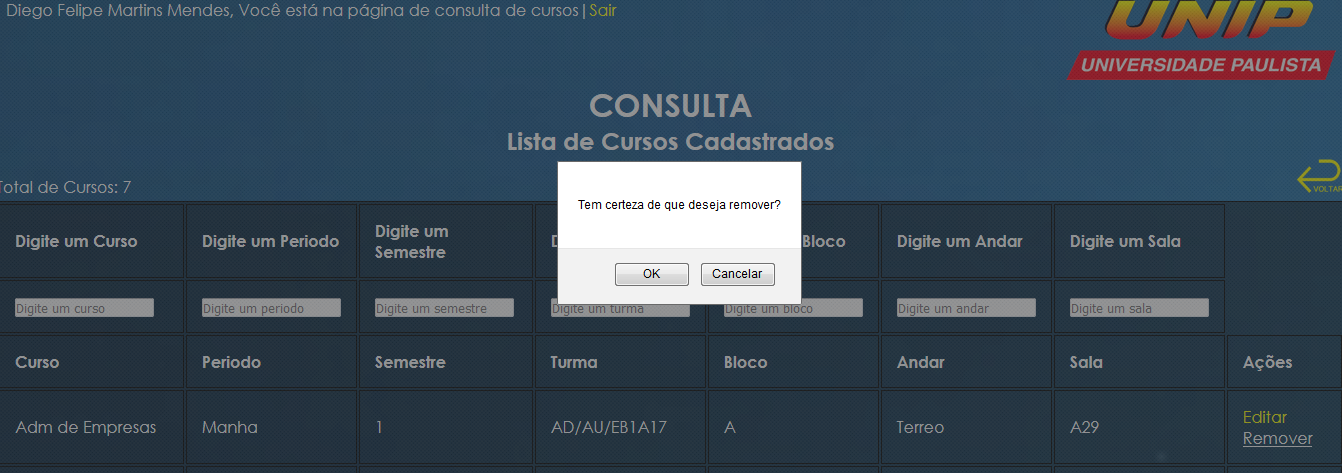


Imagem 1 – Tela de consulta (página web).

Fonte: Criação própria, 2018.

### 10.1.1.6 Reconhecimento em vez de memorização

A grande diferença entre reconhecer e memorizar é a quantidade de dicas que são fornecidas para que um conhecimento seja acessado, reconhecer padrões fornece muito mais dicas do que tentar acessar memórias.

### 10.1.1.7 Flexibilidade e eficiência de uso

O ideal é que a interface seja útil tanto para usuários leigos como para experientes. Leigos precisam de informações mais detalhadas para poderem realizar tarefas, mas na medida que crescem em conhecimento sobre a interface, a necessidade de formas mais rápidas de interação para realizar uma tarefa começa a surgir. UniSearch possui informações detalhadas, portanto não haverá duvidas pois sabemos as necessidades dos mesmos.

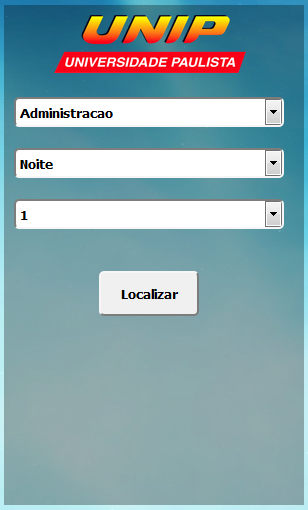
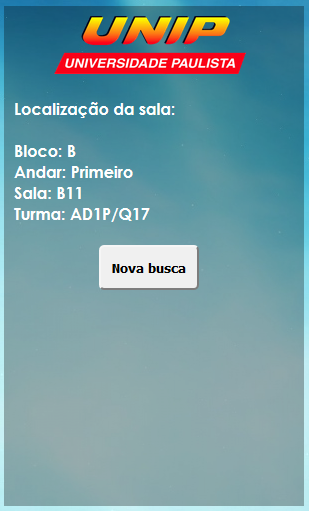
### 10.1.1.8 Estética e design minimalista

Quanto maior a quantidade de informação maior será a quantidade de informações que serão analisadas e decisões que o usuário precisará tomar, por isso, é crucial manter apenas as informações que são realmente necessárias.

O aplicativo mobile possui limitação de telas, quando o usuário busca a informação do curso, exibe as informações na mesma tela e somente as informações necessárias para a localização.

Imagem 2 – Resultado (aplicativo mobile).

Imagem 1 – Busca de sala (aplicativo mobile).



Fonte: Criação própria, 2018.

Fonte: Criação própria, 2018.

### 10.1.1.9 Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros

Um exemplo são aqueles avisos de formulários nos campos que não foram preenchidos corretamente. É uma forma simples de mostrar para o usuário que ele cometeu um erro, onde errou e o que precisa ser feito para corrigir tal erro.

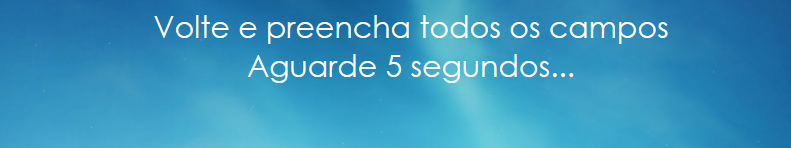


Imagem 1 – Aviso de campos vazios (Página Web).

Fonte: Criação própria, 2018.

### 10.1.1.10 Ajuda e documentação

Normalmente essas são as áreas menos acessadas, mas ainda assim elas são importantes dentro de um sistema pois nunca se sabe quando um usuário irá precisar de uma ajudinha.

UniSearch possui informações detalhadas, portanto não haverá duvidas, não possuímos documentos de autoajuda.

# 11.0 CONCLUSÃO

Diante de pesquisas realizadas, foi feito um levantamento quanto à opinião dos universitários que frequentam a universidade, indicando que aplicativo será de grande utilidade, pois, de acordo com os mesmos, a dificuldade de localizar as salas dentro da universidade, tanto nos períodos de início de semestre quanto para aqueles que começam a cursar um curso na universidade (os alunos calouros) é grande.

Ao adentrar no ambiente da universidade, percebe – se que grande parte dos estudantes ficam perdidos em meio à tantas pessoas buscando o mesmo tipo de informação “Onde é minha sala? ”, muitos deles acabam se frustrando, outros acham sua sala após uma longa procura e mesmo assim se encontram atrasados para a aula prejudicando assim sua aprendizagem.

Com o aplicativo os universitários veteranos, calouros e até mesmo visitantes da universidade, adquiram maior autonomia, pois o software proporcionará as informações necessárias para a localização de sala ao usuário da maneira desejada a qualquer momento.

# 12.0 REFERÊNCIAS

Carregar dados JSON com jQuery, PHP e MySQL:

<https://www.electrictoolbox.com/json-data-jquery-php-mysql/>  
Acesso em: 14/03/2018.

Utilizando jQuery mobile:

<http://coenraets.org/blog/2011/10/sample-application-with-jquery-mobile-and-phonegap/>

Acesso em: 15/03/2018.

Sistema de cadastro:

<http://blog.ultimatephp.com.br/sistema-de-cadastro-php-mysql-pdo/>

Acesso em: 13/03/2018.

Sistema de cadastro PHP:

<https://www.youtube.com/channel/UCeDGEGnfVRBvSZN9UHVfO-A/>

Acesso em: 10/02/2018.­­

Pesquisar com dois campos:

<https://celke.com.br/material-gratuito/pesquisar-com-dois-campos?divul=youcon203/>

Acesso em 04/03/2018.

Estrutura de funções JavaScript:

<https://stackoverflow.com/questions/35102464/disable-button-if-no-selected-value-in-dropdown/>

Acesso em 02/04/2018.

# APÊNDICE

# APÊNCIDE A – Código do aplicativo

## Index.html

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<meta name="format-detection" content="telephone=no" />

<meta name="msapplication-tap-highlight" content="no" />

<meta name="viewport" content="user-scalable=no, initial-scale=1, maximum-scale=1, minimum-scale=1, width=device-width" />

<meta http-equiv="Content-Security-Policy" content="default-src \* 'unsafe-inline'; style-src 'self' 'unsafe-inline'; media-src \*" />

<link rel="stylesheet" type="text/css" href="css/styleindex.css">

<title>Encontre sua Sala</title>

</head>

<body>

<!--Inicio do formulario -->

<form>

<!--logo do app -->

<div class="logop">

<img class="logo" src="img/unisearchlogo.png" >

</div>

<!--Ocultar select após pressionar "Localizar"-->

<div id="Selects">

<!-- Cursos -->

<select name="curso" id="curso">

<option value="" selected hidden>Selecione o curso</option>

<option disabled>::|Cursos Tradicionais|::</option>

<option value="Administracao">Administracao</option>

<option value="Adm de Empresas">Adm de Empresas</option>

<option value="Arquitetura e Urbanismo">Arquitetura e Urbanismo</option>

<option value="Biomedicina">Biomedicina</option>

<option value="Ciencias Biologicas">Ciencias Biologicas</option>

<option value="Ciencia da Computacao">Ciencia da Computacao</option>

<option value=" ">Ciencias Contabeis</option>

<option value=" ">Com.Social(Prop.e Public. e Jornalismo)</option>

<option value="Direito">Direito</option>

<option value="Educacao Fisica">Educacao Fisica</option>

<option value="Enfermagem">Enfermagem</option>

<option value="Engenharia Basica">Engenharia Basica</option>

<option value="Engenharias">Engenharias</option>

<option value="Estetica e Cosmetica">Estetica e Cosmetica</option>

<option value="Farmacia">Farmacia</option>

<option value="Fisioterapia">Fisioterapia</option>

<option value="Letras">Letras</option>

<option value="Matematica">Matematica</option>

<option value="Nutricao">Nutricao</option>

<option value="Pedagogia">Pedagogia</option>

<option value="Psicologia">Psicologia</option>

<option value="Radiologia Medica">Radiologia Medica</option>

<option value="Serviço Social">Serviço Social</option>

<option value=" ">Fotografia e Design Grafico</option>

<option disabled>::|Cursos Superiores Tecnológicos|::</option>

<option value=" ">Analise e Desenvolvimento de Sistemas</option>

<option value="Automacao Industrial">Automacao Industrial</option>

<option value="Gestao Qualidade">Gestao Qualidade</option>

<option value="Gestao Financeiro">Gestao Financeiro</option>

<option value="Logistica">Logistica</option>

<option value="Processos">Processos</option>

<option value=" ">Gestao Recursos Humanos</option>

</select><br><br>

<!-- Periodos serão exibidos após selecionar curso-->

<select name="periodo" id="periodo">

</select><br><br>

<!--Semestre serão exibisdo após selecionar periodo-->

<select name="semestre" id="semestre">

</select><br><br><br>

</div>

<!--Botão Corrigir-->

<div class="btnCorrigel">

<button id="btnCorrige" onclick="SwitchButtons('btnCorrige');" window.location.href = 'http://index.html/' hidden="hidden"><img class="corrige" src="img/corrige2.png"></button>

</div>

<!--Botão Localizar-->

<button name="bntPesq" class="btnPesq" onclick="SwitchButtons('btnPesquisar');" disabled="disabled" type="button" id="btnPesquisar">Localizar</button>

<!--Exibi resultado do após clicar em "Localizar"-->

<div class="resultado">

</div>

<!--Botão para nova consulta-->

<button name="btnRef" class="btnRef" onclick="SwitchButtons('btnRefresh');" window.location.href = 'http://index.html/' id="btnRefresh" style='display:none;'>Nova busca</button><br>

</form>

<!--Fim do formulario-->

<!--Inicio das funções em JS-->

<script src="js/jquery.min.js"></script>

<script type="text/javascript" src="js/funcoes.js"></script>

</body>

</html>

## Funções.js

/\* Inicio da função para desabilitar os select option após seleção de valores \*/

$(function() {

$("#curso").change(function() {

$("#curso").prop("disabled", true);

});

$("#periodo").change(function() {

$("#periodo").prop("disabled", true);

});

$("#semestre").change(function() {

$("#semestre").prop("disabled", true);

});

});

/\* Inicio da função para habilitar o botão corrigir após o primeiro \*/

$(document).ready(function(){

$('#curso').change(function(){

var val = $('#curso').val();

if (val == '') {

$('#btnCorrige').Attr('hidden', 'hidden');

}else{

$('#btnCorrige').removeAttr('hidden');

}

});

});

/\* Inicio da função para carregar os option após a seleção do curso \*/

function buscaPeriodo() {

$.getJSON('http://191.252.96.136/plesk-site-preview/www.diego-pim.com.br/191.252.96.136/unisearchapp/periodo.php', {curso: $('#curso').val()}, function (data) {

var select = $('#periodo');

var options = select.prop('options');

$('option', select).remove();

$("#periodo").html("<option hidden >Selecione o periodo</option>");

$.each(data, function (index, array) {

options[options.length] = new Option(array['periodo']);

});

});

}

buscaPeriodo();

$('#curso').change(function () {

buscaPeriodo();

});

/\* Inicio da função para carregar os option após a seleção do periodo \*/

function buscaSemestre() { $.getJSON('http://191.252.96.136/plesk-site-preview/www.diego-pim.com.br/191.252.96.136/unisearchapp/semestre.php', {curso: $('#curso').val()}, function (data) {

var select = $('#semestre');

var options = select.prop('options');

$('option', select).remove();

$("#semestre").html("<option hidden >Selecione o semestre</option>");

$.each(data, function (index, array) {

options[options.length] = new Option(array['semestre']);

});

});

}

buscaSemestre();

$('#curso').change(function () {

buscaSemestre();

});

/\* Inicio da função para carregar resultado após clicar em localizar \*/

$(function(){

$("#btnPesquisar").click(function(){

var semestre = $("#semestre").val();

var periodo = $("#periodo").val();

var curso = $("#curso").val();

$.post("http://191.252.96.136/plesk-site-preview/www.diego-pim.com.br/191.252.96.136/unisearchapp/ajax.php", {curso:curso, periodo:periodo, semestre:semestre}, function(retorno){

$(".resultado").html("<b>Localização da sala:<br><br> " + retorno + "</b>");

if (retorno == false) {

$(".resultado").html("<b>Dados do curso não localizados !<br><br>");

}

});

})

});

/\* Inicio da função para trocar Botão de localizar para nova busca e vice versa \*/

function SwitchButtons(buttonId) {

var hideBtn, showBtn, hideCo, showCo;

$() if (buttonId == 'btnPesquisar') {

showBtn = 'btnRefresh';

hideBtn = 'btnPesquisar';

hideCo = 'btnCorrige';

} else {

showBtn = 'btnPesquisar';

showCo = 'btnCorrige';

hideBtn = 'btnRefresh';

}

document.getElementById(hideBtn).style.display = 'none';

document.getElementById(hideCo).style.display = 'none';

document.getElementById(showBtn).style.display = '';

document.getElementById(showCo).style.display = '';

}

/\* Inicio da função para habilitar botao de localização após os tres valores do select serem preenchido \*/

$(document).ready(function(){

$('#semestre').change(function(){

var val = $('#semestre').val();

if (val == '') {

$('#btnPesquisar').Attr('disabled', 'disabled');

}else{

$('#btnPesquisar').removeAttr('disabled');

}

});

});

/\* Inicio da função para ocultar a div "Selects" \*/

$(document).ready(function(){

$('#btnPesquisar').click(function(){

$('#Selects').children().attr('hidden', 'hidden');

});

});

/\* Inicio da função para ocultar os select option \*/

$(function() {

$('#btnPesquisar').click(function() {

$('#curso').hide();

$('#periodo').hide();

$('#semestre').hide();

});

});

/\* Inicio da função para aparecer o option a partir do preimeiro select \*/

$(function() {

$('#periodo').hide();

$('#curso').change(function(){

$('#semestre').hide();

if($('#curso').val()) {

$('#periodo').show();

} else {

$('#periodo').hide();

}

});

});

$(function() {

$('#semestre').hide();

$('#periodo').change(function(){

if($('#periodo').val()) {

$('#semestre').show();

} else {

$('#semestre').hide();

}

});

});

# APÊNDICE B – Código página web administrativa

## Connect.php

<?php

// Constantes com as credenciais de acesso ao banco MySQL

define('DB\_HOST', 'localhost');

define('DB\_USER', 'root');

define('DB\_PASS', '');

define('DB\_NAME', 'bdmaster');

// Habilita todas as exibições de erros

ini\_set('display\_errors', true);

error\_reporting(E\_ALL);

// Inclui o arquivo de funçõees

require\_once 'functions.php';

?>

## Function.php

<?php

//Conecta com o MySQL usando PDO

function db\_connect()

{

$PDO = new PDO('mysql:host=' . DB\_HOST . ';dbname=' . DB\_NAME . ';charset=utf8', DB\_USER, DB\_PASS);

return $PDO;

}

//Cria o hash da senha, usando MD5 e SHA-1

function make\_hash($str)

{

return sha1(md5($str));

}

// Verifica se o usuário está logado

function isLoggedIn()

{

if (!isset($\_SESSION['logged\_in']) || $\_SESSION['logged\_in'] !== true)

{

return false;

}

return true;

}

?>

## Form-login.php

<?php

session\_start();

require 'connect.php';

?>

<!doctype html>

<html>

<head>

<meta charset="utf-8">

<title>Unisearch|Login</title>

<link rel="stylesheet" type="text/css" href="css/estilologin.css"/>

</head>

<body>

<div class="logop"> <!-- divisão para logo do app -->

<img class="logo" src="img/unisearchlogo.png" ></div>

<?php if (isLoggedIn()): ?>

<p>Olá, <?php echo $\_SESSION['user\_name']; ?>. <a href="indexx.php">Página Principal</a> | <a onclick="return confirm('Tem certeza de que deseja trocar de usuário?');" href="logout.php">Trocar de Usuário</a></p>

<?php else: ?> <p>Olá, visitante.</p> <?php endif; ?>

<div>

<form class="formlogin" action="login.php" method="post" <?php if (isLoggedIn()){?> hidden <?php }?>>

<h1 <?php if (isLoggedIn()){?> hidden <?php }?>> Login</h1>

<label for="email">Email: </label> <br>

<input type="text" name="email" id="email"> <br>

<label for="password">Senha: </label> <br>

<input type="password" name="password" id="password"> <br><br>

<input class="inpentra" type="submit" value="Entrar">

</form>

</div>

<div class="formcad">

<form method="post" enctype="multipart/form-data" action="addusuario.php" <?php if (isLoggedIn()){?> hidden <?php }?>>

<h1 <?php if (isLoggedIn()){?> hidden <?php }?>>Cadastro</h1>

<label>Nome:</label> <br>

<input type="text" name="nome"/> <br>

<label>Email:</label> <br>

<input type="email" name="email"/> <br>

<label>Tel/Cel:</label> <br>

<input type="number\_format" name="numcel"/> <br>

<label>Senha:</label> <br>

<input type="password" name="senha"/><br><br>

<input class="inpcad" type="submit" value="Cadastrar Usuário"/>

</form>

</div>

</body>

</html>

## Login.php

<!DOCTYPE HTML>

<html>

<head>

<meta charset="utf-8">

<title>Unisearch|Login</title>

<link rel="stylesheet" type="text/css" href="css/estilologinerro.css"/>

</head>

<body>

</body>

</html>

<?php

// Inclui o arquivo de inicialização

require 'connect.php';

// Resgata variáveis do formulário

$email = isset($\_POST['email']) ? $\_POST['email'] : '';

$password = isset($\_POST['password']) ? $\_POST['password'] : '';

if (empty($email) || empty($password)){

echo "Erro: Informe email e senha <br>";

echo "<br> Aguarde 5 segundos...";

echo "<meta HTTP-EQUIV='refresh' CONTENT='5;URL=form-login.php'>";

exit;

}

//Cria o hash da senha

$passwordHash = make\_hash($password);

$PDO = db\_connect();

$sql = "SELECT id, name FROM users WHERE email = :email AND password = :password";

$stmt = $PDO->prepare($sql);

$stmt->bindParam(':email', $email);

$stmt->bindParam(':password', $passwordHash);

$stmt->bindParam(':password', md5($password));

$stmt->execute();

$users = $stmt->fetchAll(PDO::FETCH\_ASSOC);

if (count($users) <= 0)

{

echo "Erro: Email ou senha incorretos <br>";

echo "<br> Aguarde 5 segundos...";

echo "<meta HTTP-EQUIV='refresh' CONTENT='5;URL=form-login.php'>";

exit;

}

// Pega o primeiro usuário

$user = $users[0];

session\_start();

$\_SESSION['logged\_in'] = true;

$\_SESSION['user\_id'] = $user['id'];

$\_SESSION['user\_name'] = $user['name'];

header('Location: indexx.php'); ?>

## Addusuario.php

<!doctype html>

<html>

<head>

<meta charset="utf-8">

<title>Unisearch|Login</title>

<link rel="stylesheet" type="text/css" href="css/estilologinerro.css"/>

</head>

<body>

<?php

require\_once 'connect.php';

// Pega os dados do formuário

$nome = isset($\_POST['nome']) ? $\_POST['nome'] : null;

$email = isset($\_POST['email']) ? $\_POST['email'] : null;

$numcel = isset($\_POST['numcel']) ? $\_POST['numcel'] : null;

$senha = isset($\_POST['senha']) ? hash('md5',$\_POST['senha']) : null;

// validação (bem simples, só pra evitar dados vazios)

if (empty($nome) || empty($email) || empty($senha) || empty($numcel))

{

echo "Favor: Preencha todos os campos<br>";

echo "<br> Aguarde 5 segundos...";

echo "<meta HTTP-EQUIV='refresh' CONTENT='5;URL=form-login.php'>";

exit;

}

// Insere no banco

$PDO = db\_connect();

$sql = "INSERT INTO users(name, email, numcel, password) VALUES(:nome, :email, :numcel, :senha)";

$stmt = $PDO->prepare($sql);

$stmt->bindParam(':nome', $nome);

$stmt->bindParam(':email', $email);

$stmt->bindParam(':numcel', $numcel);

$stmt->bindParam(':senha', $senha);

if ($stmt->execute())

{

echo "Cadastro realizado com sucesso<br>";

echo "<br> Aguarde 5 segundos...";

echo "<meta HTTP-EQUIV='refresh' CONTENT='5;URL=form-login.php'>";

}

else

{

echo "Erro ao cadastrar";

print\_r($stmt->errorInfo());

echo "<meta HTTP-EQUIV='refresh' CONTENT='5;URL=form-login.php'>";

}

?>

## Logout.php

<?php

// inicia a sessão

session\_start();

// muda o valor de logged\_in para false

$\_SESSION['logged\_in'] = false;

// finaliza a sessão

session\_destroy();

// retorna para a index.php

header('Location: form-login.php');

?>

## Indexx.php

<?php

session\_start();

require\_once 'connect.php';

require 'check.php';

?>

<html>

<head>

<title>Unisearch|Cadastro e Consulta</title>

<meta charset="UTF-8"/>

<link rel="stylesheet" type="text/css" href="css/index.css"/>

</head>

<body>

<?php if (isLoggedIn()): ?>

<p>Bem-vindo à página de cadastro e consulta, <?php echo $\_SESSION['user\_name']; ?> | <a onclick="return confirm('Tem certeza de que deseja sair?');" href="logout.php">Sair</a></p>

<?php endif; ?>

<div class="logop"> <!-- divisão para logo do app -->

<img class="logo" src="img/unisearchlogo.png" >

</div>

<div class="ientra"> <!-- divisão para btn consulta -->

<a href="cadastrocursos.php"><img class="ientraimg" src="img/entra.png"></a>

</div>

<div class="iconsulta"> <!-- divisão para btn consulta -->

<a href="http://localhost/CadastroCurso/consulta.php"><img class="iconsultaimg" src="img/consulta.png"></a>

</div>

</body>

</html>

## Check.php

<?php

// Inclui o arquivo de conexão

require\_once 'connect.php';

if (!isLoggedIn())

{

header('Location: form-login.php');

}

?>

## Cadastrocurso.php

<?php

session\_start();

require 'connect.php';

require 'check.php';

?>

<html>

<head>

<title>Unisearch|Cadastro de Cursos</title>

<meta charset="UTF-8"/>

<link rel="stylesheet" type="text/css" href="css/estilopagweb.css"/>

</head>

<body>

<?php if (isLoggedIn()): ?>

<p class="pp"><?php echo $\_SESSION['user\_name'];?>, Você está na página de cadastro de cursos|<a class="btnsair" onclick="return confirm('Tem certeza de que deseja sair?');" href="logout.php">Sair</a></p>

<?php endif; ?>

<form name="entrar" method="post" action="addinfo.php">

<div class="logop"> <!-- divisão para logo do app -->

<img class="logo" src="img/unisearchlogo.png" >

</div>

<div class="back">

<a href="http://localhost/CadastroCurso/indexx.php"><img class="backimg" src="img/voltar.png"></a>

</div>

<h2>Dados do Curso</h2></br>

<h4>Cursos</h4>

<select name="curso" id="curso">

<option hidden selected>Selecione o Curso</option>

<option disabled>::|Cursos Superiores Tradicionais|::</option>

<option>Administracao</option>

<option>Adm de Empresas</option>

<option>Arquitetura e Urbanismo</option>

<option>Biomedicina</option>

<option>Ciencias Biologicas</option>

<option>Ciencia da Computacao</option>

<option>Ciencias Contabeis</option>

<option>Com.Social(Prop.e Public. e Jornalismo)</option>

<option>Direito</option>

<option>Educacao Fisica</option>

<option>Enfermagem</option>

<option>Engenharia Basica</option>

<option>Engenharias</option>

<option>Estetica e Cosmetica</option>

<option>Farmacia</option>

<option>Fisioterapia</option>

<option>Letras</option>

<option>Matematica</option>

<option>Nutricao</option>

<option>Pedagogia</option>

<option>Psicologia</option>

<option>Radiologia Medica</option>

<option>Serviço Social</option>

<option>Fotografia e Design Grafico</option>

<option disabled>::|Cursos Superiores Tecnológicos|::</option>

<option>Analise e Desenvolvimento de Sistemas</option>

<option>Automacao Industrial</option>

<option>Gestao Qualidade</option>

<option>Gestao Financeiro</option>

<option>Logistica</option>

<option>Processos</option>

<option>Gestao Recursos Humanos</option>

</select></br>

<h4>Períodos</h4>

<select name="periodo" id="periodo">

<option hidden selected>Selecione o periodo</option>

<option>Manha</option>

<option>Noite</option>

</select></br>

<h4>Semestres</h4>

<select name="semestre" id="semestre">

<option hidden selected>Selecione o semestre</option>

<option>1</option>

<option>2</option>

<option>3</option>

<option>4</option>

<option>5</option>

<option>6</option>

<option>7</option>

<option>8</option>

<option>9</option>

<option>10</option>

</select></br>

<h4>Turmas</h4>

<input type="text" name="turma" id="turma" placeholder="Digite o Cod. da turma"/></br>

<h4>Blocos</h4>

<select name="bloco" id="bloco">

<option hidden selected>Selecione o Bloco</option>

<option>A</option>

<option>B</option>

<option>C</option>

<option>D</option>

<option>E</option>

<option>F</option>

<option>G</option>

<option>H</option>

</select></br>

<h4>Andares</h4>

<select name="andar" id="andar">

<option hidden selected>Selecione o andar</option>

<option>1 Subsolo</option>

<option>2 Subsolo</option>

<option>Terreo</option>

<option>Primeiro</option>

<option>Segundo</option>

</select></br>

<h4>Salas</h4>

<input type="text" name="sala" id="sala" placeholder="Digite o Cod. da sala"/></br>

<input class="button" type="submit" value="Cadastrar Curso"/>

<a href="consulta.php"><strong>CONSULTA DE CURSOS</strong></a>

</form>

</body>

</html>

## Addinfo.php

<!doctype html>

<html>

<head>

<meta charset="utf-8">

<title>Unisearch|Cadastro Curso</title>

<link rel="stylesheet" type="text/css" href="css/estilologinerro.css"/>

</head>

<body>

<?php

require\_once 'connect.php';

// pega os dados do formuário

$curso = isset($\_POST['curso']) ? $\_POST['curso'] : null;

$periodo = isset($\_POST['periodo']) ? $\_POST['periodo'] : null;

$semestre = isset($\_POST['semestre']) ? $\_POST['semestre'] : null;

$turma = isset($\_POST['turma']) ? $\_POST['turma'] : null;

$bloco = isset($\_POST['bloco']) ? $\_POST['bloco'] : null;

$andar = isset($\_POST['andar']) ? $\_POST['andar'] : null;

$sala = isset($\_POST['sala']) ? $\_POST['sala'] : null;

// validação (bem simples, só pra evitar dados vazios)

if (empty($curso) || empty($periodo) || empty($semestre) || empty($turma) || empty($bloco) || empty($andar) || empty($sala))

{

echo "Volte e preencha todos os campos";

echo "<br> Aguarde 5 segundos...";

echo "<meta HTTP-EQUIV='refresh' CONTENT='5;URL=cadastrocursos.php'>";

exit;

}

// insere no banco

$PDO = db\_connect();

$sql = "INSERT INTO cursos(curso, periodo, semestre, turma, bloco, andar, sala) VALUES(:curso, :periodo, :semestre, :turma, :bloco, :andar, :sala)";

$stmt = $PDO->prepare($sql);

$stmt->bindParam(':curso', $curso);

$stmt->bindParam(':periodo', $periodo);

$stmt->bindParam(':semestre', $semestre);

$stmt->bindParam(':turma', $turma);

$stmt->bindParam(':bloco', $bloco);

$stmt->bindParam(':andar', $andar);

$stmt->bindParam(':sala', $sala);

if ($stmt->execute())

{

header('Location: consulta.php');

}

else

{

echo "Erro ao cadastrar";

print\_r($stmt->errorInfo());

}

?>

## Consulta.php

<?php

session\_start();

require\_once 'connect.php';

require 'check.php';

// abre a conexão

$PDO = db\_connect();

// SQL para contar o total de registros

// A biblioteca PDO possui o método rowCount(), mas ele pode ser impreciso.

// É recomendável usar a função COUNT da SQL

// Veja o Exemplo 2 deste link: http://php.net/manual/pt\_BR/pdostatement.rowcount.php

$sql\_count = "SELECT COUNT(DISTINCT curso) AS total FROM cursos ORDER BY curso ASC";

// SQL para selecionar os registros

$sql = "SELECT \* FROM cursos ORDER BY curso ASC";

// conta o total de registros

$stmt\_count = $PDO->prepare($sql\_count);

$stmt\_count->execute();

$total = $stmt\_count->fetchColumn();

// seleciona os registros

$stmt = $PDO->prepare($sql);

$stmt->execute();

?>

<!doctype html>

<html>

<head>

<meta charset="utf-8">

<link rel="stylesheet" type="text/css" href="css/estiloconsulta.css"/>

<script type="text/javascript" src="js/jquery-1.7.2.min.js"></script>

<script type="text/javascript" src="js/script.js"></script>

<title>Unisearch|Consulta</title>

</head>

<body>

<?php if (isLoggedIn()): ?>

<p class="pcons" ><?php echo $\_SESSION['user\_name']; ?>, Você está na página de consulta de cursos|<a onclick="return confirm('Tem certeza de que deseja sair?');" href="logout.php">Sair</a></p>

<?php endif; ?>

<div class="logop"> <!-- divisão para logo do app -->

<img class="logo" src="img/unisearchlogo.png" >

</div>

<h1>CONSULTA</h1>

<h2>Lista de Cursos Cadastrados</h2>

<p>Total de Cursos: <?php echo $total ?></p>

<?php if ($total > 0): ?>

<table id="tabela">

<thead>

<tr>

<th>Digite um Curso</th>

<th>Digite um Periodo</th>

<th>Digite um Semestre</th>

<th>Digite um Turma</th>

<th>Digite um Bloco</th>

<th>Digite um Andar</th>

<th>Digite um Sala</th>

</tr>

<tr>

<th><input placeholder="Digite um curso" type="text" id="txtColuna1"/></th>

<th><input placeholder="Digite um periodo" type="text" id="txtColuna2"/></th>

<th><input placeholder="Digite um semestre" type="text" id="txtColuna3"/></th>

<th><input placeholder="Digite um turma" type="text" id="txtColuna4"/></th>

<th><input placeholder="Digite um bloco" type="text" id="txtColuna5"/></th>

<th><input placeholder="Digite um andar" type="text" id="txtColuna6"/></th>

<th><input placeholder="Digite um sala" type="text" id="txtColuna7"/></th>

</tr>

</thead>

<thead>

<tr>

<th>Curso</th>

<th>Periodo</th>

<th>Semestre</th>

<th>Turma</th>

<th>Bloco</th>

<th>Andar</th>

<th>Sala</th>

<th>Ações</th>

</tr>

</thead>

<div class="back">

<a href="http://localhost/CadastroCurso/indexx.php"><img class="backimg" src="img/voltar.png"></a>

</div>

<tbody>

<?php while ($courses = $stmt->fetch(PDO::FETCH\_ASSOC)): ?>

<tr>

<td><?php echo $courses['curso'] ?></td>

<td><?php echo $courses['periodo'] ?></td>

<td><?php echo $courses['semestre'] ?></td>

<td><?php echo $courses['turma'] ?></td>

<td><?php echo $courses['bloco'] ?></td>

<td><?php echo $courses['andar'] ?></td>

<td><?php echo $courses['sala'] ?></td>

<td>

<a href="form-edit.php?id\_curso=<?php echo $courses['id\_curso'] ?>">Editar</a>

<a href="delete.php?id\_curso=<?php echo $courses['id\_curso'] ?>" onclick="return confirm('Tem certeza de que deseja remover?');">Remover</a>

</td>

</tr>

<?php endwhile; ?>

</tbody>

</table>

<?php else: ?>

<p>Nenhum curso registrado</p>

<?php endif; ?>

</body>

</html>

## Form-edit.php

<?php

session\_start();

require\_once 'connect.php';

require 'check.php';

// pega o ID da URL

$id\_curso = isset($\_GET['id\_curso']) ? (int) $\_GET['id\_curso'] : null;

// valida o ID

if (empty($id\_curso))

{

echo "ID para alteração não definido";

exit;

}

// busca os dados do usuário a ser editado

$PDO = db\_connect();

$sql = "SELECT curso, periodo, semestre, turma, bloco, andar, sala FROM cursos WHERE id\_curso = :id\_curso";

$stmt = $PDO->prepare($sql);

$stmt->bindParam(':id\_curso', $id\_curso, PDO::PARAM\_INT);

$stmt->execute();

$courses = $stmt->fetch(PDO::FETCH\_ASSOC);

// se o método fetch() não retornar um array, significa que o ID não corresponde a um usuário válido

if (!is\_array($courses))

{

echo "Nenhum curso encontrado";

exit;

}

?>

<!doctype html>

<html>

<head>

<meta charset="utf-8">

<link rel="stylesheet" type="text/css" href="css/estiloedit.css"/>

<title>Unisearch|Editor de dados</title>

</head>

<body>

<?php if (isLoggedIn()): ?>

<p class="pp"><?php echo $\_SESSION['user\_name']; ?>, Você está na página de edição de dados do curso|<a class="ppa" onclick="return confirm('Tem certeza de que deseja sair?');" href="logout.php">Sair</a></p>

<?php endif; ?>

<form action="edit.php" method="post">

<div class="logop"> <!-- divisão para logo do app -->

<img class="logo" src="img/unisearchlogo.png" >

</div>

<div class="back">

<a href="http://localhost/CadastroCurso/consulta.php"><img class="backimg" src="img/voltar.png"></a>

</div>

<h2>Editar dados do Curso</h2></br>

<h4>Cursos</h4>

<select name="curso" id="curso" value="">

<option hidden selected><?php echo $courses['curso'] ?></option>

<option disabled>::|Cursos Superiores Tradicionais|::</option>

<option>Administracao</option>

<option>Adm de Empresas</option>

<option>Arquitetura e Urbanismo</option>

<option>Biomedicina</option>

<option>Ciencias Biologicas</option>

<option>Ciencia da Computacao</option>

<option>Ciencias Contabeis</option>

<option>Com.Social(Prop.e Public. e Jornalismo)</option>

<option>Direito</option>

<option>Educacao Fisica</option>

<option>Enfermagem</option>

<option>Engenharia Basica</option>

<option>Engenharias</option>

<option>Estetica e Cosmetica</option>

<option>Farmacia</option>

<option>Fisioterapia</option>

<option>Letras</option>

<option>Matematica</option>

<option>Nutricao</option>

<option>Pedagogia</option>

<option>Psicologia</option>

<option>Radiologia Medica</option>

<option>Serviço Social</option>

<option>Fotografia e Design Grafico</option>

<option disabled>::|Cursos Superiores Tecnológicos|::</option>

<option>Analise e Desenvolvimento de Sistemas</option>

<option>Automacao Industrial</option>

<option>Gestao Qualidade</option>

<option>Gestao Financeiro</option>

<option>Logistica</option>

<option>Processos</option>

<option>Gestao Recursos Humanos</option>

</select></br>

<h4>Períodos</h4>

<select name="periodo" id="periodo" value="">

<option hidden selected><?php echo $courses['periodo'] ?></option>

<option>Manha</option>

<option>Noite</option>

</select></br>

<h4>Semestres</h4>

<select name="semestre" id="semestre" value="">

<option hidden selected><?php echo $courses['semestre'] ?></option>

<option>1</option>

<option>2</option>

<option>3</option>

<option>4</option>

<option>5</option>

<option>6</option>

<option>7</option>

<option>8</option>

<option>9</option>

<option>10</option>

</select></br>

<h4>Turmas</h4>

<input type="text" name="turma" id="turma" placeholder="Digite o Cod. da turma" value="<?php echo $courses['turma'] ?>"/></br>

<h4>Blocos</h4>

<select name="bloco" id="bloco" value="">

<option hidden selected><?php echo $courses['bloco'] ?></option>

<option>A</option>

<option>B</option>

<option>C</option>

<option>D</option>

<option>E</option>

<option>F</option>

<option>G</option>

<option>H</option>

</select></br>

<h4>Andares</h4>

<select name="andar" id="andar" value="">

<option hidden selected><?php echo $courses['andar'] ?></option>

<option>1 Subsolo</option>

<option>2 Subsolo</option>

<option>Terreo</option>

<option>Primeiro</option>

<option>Segundo</option>

</select></br>

<h4>Salas</h4>

<input type="text" name="sala" id="sala" placeholder="Digite o Cod. da sala" value="<?php echo $courses['sala'] ?>"/></br>

<input type="hidden" name="id\_curso" value="<?php echo $id\_curso ?>">

<input class="button" type="submit" value="Alterar">

</form>

</body>

</html>

## Edit.php

<?php

require\_once 'connect.php';

// resgata os valores do formulário

$curso = isset($\_POST['curso']) ? $\_POST['curso'] : null;

$periodo = isset($\_POST['periodo']) ? $\_POST['periodo'] : null;

$semestre = isset($\_POST['semestre']) ? $\_POST['semestre'] : null;

$turma = isset($\_POST['turma']) ? $\_POST['turma'] : null;

$bloco = isset($\_POST['bloco']) ? $\_POST['bloco'] : null;

$andar = isset($\_POST['andar']) ? $\_POST['andar'] : null;

$sala = isset($\_POST['sala']) ? $\_POST['sala'] : null;

$id\_curso = isset($\_POST['id\_curso']) ? $\_POST['id\_curso'] : null;

// validação (bem simples, mais uma vez)

if (empty($curso) || empty($periodo) || empty($semestre) || empty($turma) || empty($bloco) || empty($andar) || empty($sala))

{

echo "Volte e preencha todos os campos";

exit;

}

// atualiza o banco

$PDO = db\_connect();

$sql = "UPDATE cursos SET curso = :curso, periodo = :periodo, semestre = :semestre, turma = :turma, bloco = :bloco, andar = :andar, sala = :sala WHERE id\_curso = :id\_curso";

$stmt = $PDO->prepare($sql);

$stmt->bindParam(':curso', $curso);

$stmt->bindParam(':periodo', $periodo);

$stmt->bindParam(':semestre', $semestre);

$stmt->bindParam(':turma', $turma);

$stmt->bindParam(':bloco', $bloco);

$stmt->bindParam(':andar', $andar);

$stmt->bindParam(':sala', $sala);

$stmt->bindParam(':id\_curso', $id\_curso, PDO::PARAM\_INT);

if ($stmt->execute())

{

header('Location: consulta.php');

}

else

{

echo "Erro ao alterar";

print\_r($stmt->errorInfo());

}

?>