SENAI – SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL

CURSO TÉCNICO EM INFORMÁTICA

**CAMPUS DA INDÚSTRIA**

**GABRIEL BRAZ E SANTOS**

**GUILHERME AUGUSTO DE SENES**

### CHRONOSCHOOL

# CURITIBA

**2018**

GABRIEL BRAZ E SANTOS

GUILHERME AUGUSTO DE SENES

**CHRONOSCHOOL**

Projeto de conclusão apresentado ao Curso Técnico em Informática do SENAI – Serviço Nacional de Aprendizagem Industrial.

**Professor orientador: Mary Anne da Cruz Siqueira de Oliveira**

**CURITIBA**

**2018**

**GABRIEL BRAZ E SANTOS**

**GUILHERME AUGUSTO DE SENES**

**PROJETO FINAL DE CONCLUSÃO DE CURSO**

**SENAI – CENTRO INTERNACIONAL DE INOVAÇÃO**

**CURSO TÉCNICO EM INFORMÁTICA**

**CHRONOSCHOOL**

**APROVADO EM:\_\_\_\_\_/\_\_\_\_\_\_/ 2018**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Orientador:

SENAI

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Prof.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**SUMÁRIO \***

[1. INTRODUÇÃO 7](#_Toc517893520)

[1.1. PROBLEMATIZAÇÃO 7](#_Toc517893521)

[2. OBJETIVOS 9](#_Toc517893522)

[2.1 OBJETIVO GERAL 9](#_Toc517893523)

[2.2 OBJETIVOS ESPECÍFICOS 9](#_Toc517893524)

[3. JUSTIFICATIVA 10](#_Toc517893525)

[4. FUNDAMENTAÇÃO TEÓRICA \* 11](#_Toc517893526)

[4.1 A Internet 11](#_Toc517893527)

[4.2 O Cronograma 11](#_Toc517893528)

[4.3 A Escola 12](#_Toc517893529)

[4.4 Ferramentas 12](#_Toc517893530)

[4.4.1 Internet 12](#_Toc517893531)

[4.4.2 HTML 5 13](#_Toc517893532)

[4.4.3 CSS3 13](#_Toc517893533)

[4.4.4 C# 14](#_Toc517893534)

[4.4.5 Visual Studio 14](#_Toc517893535)

[4.4.6 SQL Server 14](#_Toc517893536)

[4.4.6 FTP 15](#_Toc517893537)

[4.4.7 Linguagem SQL 16](#_Toc517893538)

[4.4.8 Banco de Dados 16](#_Toc517893539)

[5 DESENVOLVIMENTO DO PROJETO 17](#_Toc517893540)

[5.1 Estudo de Caso 17](#_Toc517893541)

[5.2 REQUISITOS 17](#_Toc517893542)

[5.3 Principais Requisitos do Sistema\* 18](#_Toc517893543)

[5.3.1 Tela Inicial 19](#_Toc517893544)

[5.2.2 Login ao Sistema 19](#_Toc517893545)

[5.3.3 Cadastro Usuários 19](#_Toc517893546)

[5.3.4 Cadastro de Gerentes 20](#_Toc517893547)

[5.3.5 Geração de Cronogramas 20](#_Toc517893548)

[5.3.6 Envio de Notificação para Alunos 20](#_Toc517893549)

[5.3.7 Segurança 21](#_Toc517893550)

[5.3.8 Banco de Dados 21](#_Toc517893551)

[5.3.9 Portabilidade 21](#_Toc517893552)

[5.3.10 Disponibilidade 22](#_Toc517893553)

[5.3.11 Eficiência 22](#_Toc517893554)

[5.3.12 Usabilidade 22](#_Toc517893555)

[5.3 Diagrama UML\* 23](#_Toc517893556)

[5.3.1 Diagrama de Caso e Uso\* 23](#_Toc517893557)

[5.3.2 Diagrama de Classes\* 23](#_Toc517893558)

[5.3.3 Diagrama de Atividades\* 24](#_Toc517893559)

[5.3.4 Diagrama de Sequência\* 35](#_Toc517893560)

[5.3.5 Diagrama de Estados\* 40](#_Toc517893561)

[6 RESULTADOS\* 41](#_Toc517893562)

[7 ANÁLISE E INTERPRETAÇÃO DOS DADOS\* 42](#_Toc517893563)

[8 SUGESTÕES PARA MELHORIA DO PROCESSO\* 43](#_Toc517893564)

[9 CONSIDERAÇÕES FINAIS\* 44](#_Toc517893565)

[10 REFERÊNCIAS BIBLIOGRÁFICAS\* 45](#_Toc517893566)

[ANEXOS\* 46](#_Toc517893567)

Lista de imagens \*

Figura 01 – Diagrama de Caso e Uso

Figura 02 – Diagrama de Classes

Figura 03 – Diagrama de Atividades – Cadastro de Usuários

Figura 04 – Diagrama de Atividades – Login

Figura 05 – Diagrama de Atividades – Buscar

Figura 06 – Diagrama de Atividades – Cadastrar Disco/Artista

Figura 07 – Diagrama de Atividades – Troca de mensagens

Figura 08 – Diagrama de Atividades – Comentar/Classificar

Figura 09 – Diagrama de Atividades – Adicionar disco ao perfil

Figura 10 – Diagrama de Atividades – Ver discos adicionados ao perfil

Figura 11 – Diagrama de Atividades – Alterações nos discos

Figura 12 – Diagrama de Atividades – Validar alterações

Figura 13 – Diagrama de Sequência – Buscar disco

Figura 14 – Diagrama de Sequência – Buscar artista/gênero

Figura 15 – Diagrama de Sequência – Interações do usuário

Figura 16 – Diagrama de Sequência – Interações entre usuários

Figura 17 – Diagrama de Sequência – Interações ADM

Figura 18 – Diagrama de Estados – Interações ADM

1. INTRODUÇÃO

O tema central deste trabalho é a criação de um sistema web capaz de organizar automaticamente cronogramas, de forma que não colida com os horários dos professores e ainda notifique os alunos das mudanças que forem realizadas.

Este trabalho contém as principais informações da aplicação ChronoSchool, que direcionarão sua construção, assim como orientam sua manutenção. Neste é possível encontrar as principais funcionalidades baseado nos requisitos levantados, a metodologia de desenvolvimento, seu funcionamento e as características principais da aplicação. É através deste documento que podemos compreender melhor seu conceito, sua construção e sua aplicação, uma vez que é abordado desde o público-alvo até os resultados de sua utilização.

Sendo que tanto instituições de ensino privadas quanto publicas se beneficiam de um método de organização eficiente, é necessário que o gerenciamento de horários seja capaz de desempenhar sua função de forma eficiente e com o mínimo de conflitos, assim diminuindo as dificuldades no dia-a-dia de professores e alunos.

* 1. PROBLEMATIZAÇÃO

Conforme Jerônimo (2001, p. 51) “A escola é uma instituição socual, regida por normas, que dizem respeito à obrigação escolar, aos horários, ao emprego do tempo e outras”. Nesse sentido, o cronograma escolar é uma ferramenta essencial para a organização de uma instituição de ensino, possibilitando a fácil visualização do conteúdo de um dia letivo, permitindo que tantos os alunos quanto os professores possam se organizar, evitando perdas de tempo e diminuindo surpresas pois todos os envolvidos estão cientes das mudanças de horário.

Em muitos locais a criação e manutenção dos cronogramas escolares são feitos manualmente, por exemplo, utilizando tabelas do Excel ou feitos com papel e caneta, sendo esses processos desorganizados e complexos. Tal dificuldade faz com que a manutenção seja demorada e propensa a erros pois o responsável deve equilibrar tanto os horários que professores e alunos tem disponíveis quanto os horários de aula necessários.

Com estes problemas em mente decidimos criar um sistema web que amenize esta dificuldade na organização e falta de agilidade para criar cronogramas que ao mesmo tempo facilitem mudanças e a visualização por alunos/responsáveis, fazendo com que os usuários economizem tempo e esforços.

1. OBJETIVOS

2.1 OBJETIVO GERAL

Desenvolver um sistema web para instituições de ensino capaz de gerar e editar cronogramas sem causar contratempos nos horários de professores e notificar os alunos de possíveis mudanças.

2.2 OBJETIVOS ESPECÍFICOS

* Facilitar a criação, o gerenciamento e a visualização de cronogramas;
* Ser capaz de armazenar os dados relevantes para a aplicação;
* Notificar alunos envolvidos em caso de mudança;
* Construir um sistema baseado na web;
* Documentar o desenvolvimento de acordo com as técnicas de Engenharia de Software;

1. JUSTIFICATIVA

O desenvolvimento deste software é importante para auxiliar o dia a dia do corpo docente e os administradores na área de ensino. Baseado na necessidade demonstrada por professores de uma escola pública, decidimos criar um sistema web para facilitar esta manutenção dos cronogramas escolares, com objetivo de atender tanto instituições de ensino públicas quanto as privadas, utilizando técnicas para desenvolvimento e engenharia de software obtidas no curso Técnico em Informática.

1. FUNDAMENTAÇÃO TEÓRICA \*

4.1 A Internet

A rede mundial de computadores, ou internet, surgiu no final do ano de 1960, durante a guerra fria. Foi inventada para objetivos militares, além de ser utilizada para estes fins, a internet foi importante meio de comunicação acadêmico. Nas décadas de 1970 e 1980, professores e estudantes trocavam mensagens e descobertas pelas linhas da rede mundial.

A internet começou a alcançar a população, em geral no ano de 1990. Neste ano o engenheiro inglês Tim Bernes- Lee desenvolveu a World Web, possibilitando que se utilizasse uma interface gráfica e a criação de sites visualmente interessantes e dinâmicos. Passou então, a ser utilizada por diversos segmentos sociais, tornando-se muito útil em diversas áreas, como por exemplo, para buscar informações em pesquisas escolares, diversão em sites de jogos e salas de bate-papo virtual. ("Internet", 2015)

4.2 O Cronograma

O cronograma é uma ferramenta básica para gerenciamento de tempo, consiste de uma lista de momentos em que atividades, eventos ou ações devem acontecer, ou uma sequência de eventos em ordem cronológica em qual estas atividades devem ocorrer. O processo para criação de um cronograma, decidir como ordenar os eventos e como gerenciar recursos para a variedade de atividades, é denominado como agendamento. ("Schedule", 2015, tradução nossa)

4.3 A Escola

A escola é uma instituição concebida para o ensino de alunos sob a direção de professores. A maioria dos países tem sistemas formais de educação, que geralmente são obrigatórios. Nestes sistemas, os estudantes progridem através de uma série de níveis escolares e sucessivos. Os nomes para esses níveis nas escolas variam por país, mas geralmente incluem o ensino fundamental para crianças e o ensino médio para os adolescentes que concluíram o fundamental. Uma instituição onde o ensino superior é ensinado, é comumente chamada de faculdade ou universidade.

Além destas, os alunos também podem frequentar outras instituições escolares, antes e depois do ensino fundamental. A pré-escola fornece uma escolaridade básica para as crianças mais jovens. As profissionalizantes, faculdades ou seminários podem estar disponíveis antes, durante ou depois do ensino médio. A escola também pode ser dedicada a um campo particular, como uma escola de economia ou de música, por exemplo. ("Escola", 2016)

4.4 Ferramentas

4.4.1 Internet

O termo internet, como um sistema global específico de redes de IPs interconectados, é um nome próprio. A Internet também é muitas vezes referida como Net. A palavra "internet" foi utilizado historicamente, com inicial minúscula, logo em 1883 como um verbo e adjetivo para se referir a movimentos interligados. No início dos anos 1970, o termo internet começou a ser usado como uma forma abreviada do conjunto de redes técnicas, o resultado da interligação de redes de computadores com gateways especiais ou roteadores. Ele também foi usado como um verbo que significa "conectar", especialmente redes. ("Internet", 2015)

4.4.2 HTML 5

HTML5 (Hypertext Markup Language, versão 5) é uma linguagem para estruturação e apresentação de conteúdo para a World Wide Web e é uma tecnologia chave da Internet originalmente proposto por Opera Software. É a quinta versão da linguagem HTML. Esta nova versão traz consigo importantes mudanças quanto ao papel do HTML no mundo da Web, através de novas funcionalidades como semântica e acessibilidade.

Esta linguagem possibilita o uso de novos recursos antes possíveis apenas com a aplicação de outras tecnologias. Sua essência tem sido melhorar a linguagem com o suporte para as mais recentes multimídias, enquanto a mantém facilmente legível por seres humanos e consistentemente compreendida por computadores e outros dispositivos. O HTML5 será o novo padrão e está em fase de esboço, porém diversos navegadores já implementam algumas de suas funcionalidades. ("HTML5", 2015)

4.4.3 CSS

Cascading Style Sheets (CSS) é uma linguagem usada para descrever a apresentação de um documento feito com uma linguagem de marcação como HTML. CSS é um dos pilares da tecnologia dentro do World Wide Web, juntamente com o HTML e Javascript. As especificações do CSS são gerenciadas pelo World Wide Web Consortium (W3C). CSS foi projetado para possibilitar a separação da apresentação e o conteúdo, incluindo layout, cores e fontes. Esta separação é capaz de melhorar a acessibilidade do conteúdo, provendo mais flexibilidade e controle dentro das propriedades para a apresentação de características, reduzindo a complexidade e repetição dentro da estrutura do conteúdo.

O nome “*cascading*” vem do esquema de prioridades especificadas para determinar qual regra deve aplicar à um estilo, caso mais de uma regra seja aplicável em um elemento, sendo que este esquema de prioridade é previsível. ("Cascading Style Sheets", 2017, tradução nossa)

4.4.4 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.

O que distingue o PHP de algo como o JavaScript no lado do cliente é que o código é executado no servidor, gerando o HTML que é então enviado para o navegador. O navegador recebe os resultados da execução desse script, mas não sabe qual era o código fonte. Você pode inclusive configurar seu servidor web para processar todos os seus arquivos HTML com o PHP, e então não há como os usuários dizerem o que você tem em sua página. ("O que é o PHP?", 2018)

4.4.5 XAMPP

XAMPP é um servidor independente de plataforma, software livre, que consiste principalmente na base de dados MySQL, o qual foi substituído pelo MariaDB, o servidor web Apache e os interpretadores para linguagens de script: PHP e Perl. O nome vem da abreviação de X (para qualquer dos diferentes sistemas operativos), Apache, MySQLDB, PHP, Perl. É um método que torna extremamente fácil para os desenvolvedores a criar um servidor web local para fins de teste. ("XAMPP", 2009)

4.4.6 MySQL

O MySQL é um sistema de gerenciamento de banco de dados (SGBD), que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês Structured Query Language) como interface. É atualmente um dos sistemas de gerenciamento de bancos de dados mais populares[1], com mais de 10 milhões de instalações pelo mundo.

O MySQL foi criado na Suécia por suecos e um finlandês: David Axmark, Allan Larsson e Michael "Monty" Widenius, que têm trabalhado juntos desde a década de 1980. Hoje seu desenvolvimento e manutenção empregam aproximadamente 400 profissionais no mundo inteiro, e mais de mil contribuem testando o software, integrando-o a outros produtos, e escrevendo a respeito dele. ("MySQL", 2016)

4.4.7 Servidor Apache

O Servidor HTTP Apache (do inglês Apache HTTP Server) ou Servidor Apache ou HTTP Daemon Apache ou somente Apache, é o servidor web livre criado em 1995 por Rob McCool. É a principal tecnologia da Apache Software Foundation, responsável por mais de uma dezena de projetos envolvendo tecnologias de transmissão via web, processamento de dados e execução de aplicativos distribuídos.

É um servidor do tipo HTTPD, compatível com o protocolo HTTP versão 1.1. Suas funcionalidades são mantidas através de uma estrutura de módulos, permitindo inclusive que o usuário escreva seus próprios módulos, utilizando a API do software. ("Servidor Apache", 2012)

4.4.8 Linguagem SQL

Structured Query Language, ou Linguagem de Consulta Estruturada ou SQL, é a linguagem de pesquisa declarativa padrão para banco de dados relacional (base de dados relacional). Muitas das características originais do SQL foram inspiradas na álgebra relacional.

A linguagem é um grande padrão de banco de dados. Isto decorre da sua simplicidade e facilidade de uso. Ela se diferencia de outras linguagens de consulta a banco de dados no sentido em que uma consulta SQL especifica a forma do resultado e não o caminho para chegar a ele. Ela é uma linguagem declarativa em oposição a outras linguagens procedurais. Isto reduz o ciclo de aprendizado daqueles que se iniciam na linguagem. ("SQL", 2015)

4.4.9 Banco de Dados

Bancos de dados são um conjunto de arquivos relacionados entre si com registros sobre pessoas, lugares ou coisas. São coleções organizadas de dados que se relacionam de forma a criar algum sentido e dar mais eficiência durante uma pesquisa ou estudo. São de vital importância para empresas e há duas décadas se tornaram a principal peça dos sistemas de informação. Normalmente existem por vários anos sem alterações em sua estrutura. ("Banco de Dados", 2017)

4.4.10 Bootstrap

Bootstrap é um framework web com código-fonte aberto para desenvolvimento de componentes de interface e front-end para sites e aplicações web usando HTML, CSS e JavaScript, baseado em modelos de design para a tipografia, melhorando a experiência do usuário em um site amigável e responsivo. O Bootstrap é um dos projetos mais bem avaliado no site GitHub, com mais de 111 600 estrelas e 51.500 forks. ("Bootstrap", 2018)

4.4.11 Notepad++

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.

O Notepad++ é distribuído como um Software livre. O projeto foi hospedado no SourceForge.net, onde foi baixado mais de 27 milhões de vezes e ganhou duas vezes o prêmio "SourceForge Community Choice Award" por melhor ferramenta de desenvolvimento. O projeto foi hospedado no TuxFamily desde 2010 até 2015 depois foi movido para o GitHub. O objetivo do Notepad++ é oferecer um esguio e eficiente binário com uma interface gráfica totalmente modificável. ("Notepad++", 2011)

5 DESENVOLVIMENTO DO PROJETO

5.1 Estudo de Caso

Aplicação foi desenvolvida para instituições de ensino estaduais e para particulares, partindo do problema cenário, afim de atender à necessidade pedagógica de organização dos quadros de horários para professores e alunos, juntamente com a e um organização de toda a grade curricular.

5.2 REQUISITOS

**5.2.1 Requisitos funcionais**

Os requisitos funcionais são a descrição das diversas funções que clientes e usuários querem ou precisam que o software ofereça. Eles definem a funcionalidade desejada do software. O termo função é usado no sentido genérico de operação que pode ser realizada pelo sistema, sejam através comandos dos usuários, ou seja, pela ocorrência de eventos internos ou externos ao sistema.

• Na tela inicial terão apenas as informações de acesso e cadastro de usuários;

• O site deve ter contas de usuários, gerentes e administrador;

• O site deve ter uma área de acesso online para seus usuários;

• O site deve ter uma área de cadastro de novos discos ou artistas, para serem avaliados pelo adm.;

• O site deve gerar cronogramas automaticamente assim que solicitado;

• O site deve notificar os alunos caso ocorram mudanças em cronogramas;

• O sistema deve conter um banco de dados funcional;

**5.2.2 Requisitos não funcionais\***

Requisitos não funcionais são as qualidades globais de um software, como manutenibilidade, usabilidade, desempenho, custos e várias outras. Normalmente estes requisitos são descritos de maneira informal, de maneira controversa (por exemplo, o gerente quer segurança, mas os usuários querem facilidade de uso) e são difíceis de validar.

• Aplicação Web;

• Será usada a HTML/CSS e Javascript

• Segurança de dados;

• Utilizar o Banco de Dados SQL server;

• Tempo de resposta aceitável;

• Devem ser efetuadas atualizações constantes, para uma melhor performance;

• Backup de dados periódicos definidos pelo administrador do sistema;

• Fácil migração para diversos dispositivos, exemplo, Desktop e Android;

5.3 Principais Requisitos do Sistema\*

Os requisitos do sistema definem e apresentam a função do sistema. Um requisito mostra características e funcionalidade que o sistema deve ter ou uma restrição que deve satisfazer ao usuário. O profissional que abrange esta área é o engenheiro de requisitos que lista e faz uma análise de funcionalidade de quais estarão no sistema.

5.3.1 Tela Inicial

Essa tela mostrará as informações de acesso e como os usuários se cadastram no site.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| REQ001 – Tela inicial | | | | |
| **PRIORIDADE** | Alta | **ESTABILIDADE** | Alta | |
| **SOLICITANTE:** | Usuário | **REQ. ORIGEM:** |  | |
| **TIPO DO REQUISITO:** | Funcional | **IMPACTO NA ARQUITETURA:** | | Alta |
| **DESCRIÇÃO:** | Na tela inicial terá as informações de acesso e cadastro de usuários. | | | |

5.2.2 Login ao Sistema

Será feito uma busca de registros no banco de dados.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| REQ002 – Login ao Sistema | | | | |
| **PRIORIDADE** | Alta | **ESTABILIDADE** | Alta | |
| **SOLICITANTE:** | Usuário | **REQ. ORIGEM:** |  | |
| **TIPO DO REQUISITO:** | Funcional | **IMPACTO NA ARQUITETURA:** | | Alta |
| **DESCRIÇÃO:** | Busca dos registros no banco de dados. | | | |

5.3.3 Cadastro Usuários

Área onde os alunos/professores vão se cadastrar para ter acesso a página web.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| REQ003 – Cadastro Usuário | | | | |
| **PRIORIDADE** | Alta | **ESTABILIDADE** | Alta | |
| **SOLICITANTE:** | Usuário | **REQ. ORIGEM:** |  | |
| **TIPO DO REQUISITO:** | Funcional | **IMPACTO NA ARQUITETURA:** | | Alta |
| **DESCRIÇÃO:** | Área para cadastro de usuários | | | |

5.3.4 Cadastro de Gerentes

Área onde o administrador vai cadastrar um novo usuário gerente.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| REQ004 – Cadastro novos itens | | | | |
| **PRIORIDADE** | Alta | **ESTABILIDADE** | Alta | |
| **SOLICITANTE:** | Admin | **REQ. ORIGEM:** |  | |
| **TIPO DO REQUISITO:** | Funcional | **IMPACTO NA ARQUITETURA:** | | Alta |
| **DESCRIÇÃO:** | Área de cadastro de novos itens a ser avaliado pelo Administrador | | | |

5.3.5 Geração de Cronogramas

Área onde é solicitar a geração dos cronogramas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| REQ005 – Geração de Cronogramas | | | | |
| **PRIORIDADE** | Alta | **ESTABILIDADE** | Alta | |
| **SOLICITANTE:** | Gerente | **REQ. ORIGEM:** |  | |
| **TIPO DO REQUISITO:** | Funcional | **IMPACTO NA ARQUITETURA:** | | Alta |
| **DESCRIÇÃO:** | Área para solicitar a geração dos cronogramas | | | |

5.3.6 Envio de Notificação para Alunos

Ação do sistema para enviar notificações de mudanças no cronograma aos alunos.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| REQ006 – Envio de Notificação para Alunos | | | | |
| **PRIORIDADE** | Alta | **ESTABILIDADE** | Alta | |
| **SOLICITANTE:** | Sistema | **REQ. ORIGEM:** |  | |
| **TIPO DO REQUISITO:** | Funcional | **IMPACTO NA ARQUITETURA:** | | Média |
| **DESCRIÇÃO:** | Ação do sistema para enviar notificações de mudanças no cronograma aos alunos que estiverem relacionados com determinado cronograma. | | | |

5.3.7 Segurança

Site deve ser protegido para impedir ações de usuários mal-intencionados.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| REQ007 – Segurança | | | | |
| **PRIORIDADE** | Alta | **ESTABILIDADE** | Alta | |
| **SOLICITANTE:** |  | **REQ. ORIGEM:** |  | |
| **TIPO DO REQUISITO:** | Não Funcional | **IMPACTO NA ARQUITETURA:** | | Alta |
| **DESCRIÇÃO:** | A aplicação deve ter um tratamento de segurança, para impedir ações de usuários mal intencionados. | | | |

5.3.8 Banco de Dados

Local onde ficam armazenados os dados de usuários, seus acervos e todo sistema do site.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| REQ008 – Banco de Dados | | | | |
| **PRIORIDADE** | Alta | **ESTABILIDADE** | Alta | |
| **SOLICITANTE:** |  | **REQ. ORIGEM:** |  | |
| **TIPO DO REQUISITO:** | Não Funcional | **IMPACTO NA ARQUITETURA:** | | Alta |
| **DESCRIÇÃO:** | O banco de dados deve ser protegido para inserção de dados apenas por usuários autorizados. | | | |

5.3.9 Portabilidade

O site deve rodar em qualquer navegador.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| REQ009 – Portabilidade | | | | |
| **PRIORIDADE** | Alta | **ESTABILIDADE** | Alta | |
| **SOLICITANTE:** |  | **REQ. ORIGEM:** |  | |
| **TIPO DO REQUISITO:** | Não Funcional | **IMPACTO NA ARQUITETURA:** | | Alta |
| **DESCRIÇÃO:** | O site deverá rodar em qualquer navegador. | | | |

5.3.10 Disponibilidade

O site deve estar com disponibilidade total de tempo.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| REQ010 – Disponibilidade | | | | |
| **PRIORIDADE** | Alta | **ESTABILIDADE** | Alta | |
| **SOLICITANTE:** |  | **REQ. ORIGEM:** |  | |
| **TIPO DO REQUISITO:** | Não Funcional | **IMPACTO NA ARQUITETURA:** | | Alta |
| **DESCRIÇÃO:** | O site deverá ter alta disponibilidade do tempo. | | | |

5.3.11 Eficiência

O sistema deve ser ágil com os processos.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| REQ011 – Eficiência | | | | |
| **PRIORIDADE** | Alta | **ESTABILIDADE** | Alta | |
| **SOLICITANTE:** |  | **REQ. ORIGEM:** |  | |
| **TIPO DO REQUISITO:** | Não Funcional | **IMPACTO NA ARQUITETURA:** | | Alta |
| **DESCRIÇÃO:** | A aplicação deve retornar os resultados de buscas com rapidez. | | | |

5.3.12 Usabilidade

O site deve ser simples e de fácil uso.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| REQ012 – Usabilidade | | | | |
| **PRIORIDADE** | Alta | **ESTABILIDADE** | Alta | |
| **SOLICITANTE:** |  | **REQ. ORIGEM:** |  | |
| **TIPO DO REQUISITO:** | Não Funcional | **IMPACTO NA ARQUITETURA:** | | Alta |
| **DESCRIÇÃO:** | A aplicação deve ser intuitiva e de fácil uso | | | |

5.3 Diagrama UML\*

UML (Unifiedd Modeling Language) é uma linguagem que define uma série de métodos que auxilia a tarefa de modelar e documentar os sistemas e softwares. Nesta documentação, definimos o modelo ideal, de modo a diminuir, se não erradicar os possíveis erros que o projeto poderá apresentar. Apresentaremos os seguintes diagramas: diagrama de caso e uso, diagrama de classes, diagrama de sequência, diagrama de atividades e diagrama de estados.

5.3.1 Diagrama de Caso e Uso\*

Este diagrama descreve a funcionalidade proposta para um novo sistema que será projetado. O diagrama de caso e uso é composto por um cenário, que basicamente é a sequência de eventos quando um usuário interage com o sistema.

Figura 01 – Diagrama de Caso e Uso

5.3.2 Diagrama de Classes\*

O diagrama de classes é uma representação da estrutura e relações das classes que servem de modelo para objetos. É uma modelagem muito útil para o desenvolvimento de sistemas, pois define todas as classes que o sistema necessita possuir e é a base para a construção dos diagramas de comunicação, sequência e estados.



Figura 02 – Diagrama de Classes

5.3.3 Diagrama de Atividades\*

O Diagrama de atividade é um diagrama definido pela Linguagem de Modelagem Unificada (UML), e representa os fluxos conduzidos por processamentos. É essencialmente um gráfico de fluxo, mostrando o fluxo de controle de uma atividade para outra. Comumente isso envolve a modelagem das etapas sequenciais em um processo computacional.Os diagramas de atividade não são importantes somente para a modelagem de aspectos dinâmicos de um sistema ou um fluxograma, mas também para a construção de sistemas executáveis por meio de engenharia de produção reversa.



Figura 03 – Diagrama de Atividades – Cadastro de Usuários



Figura 04 – Diagrama de Atividades – Login



Figura 05 – Diagrama de Atividades – Buscar



Figura 06 – Diagrama de Atividades – Cadastrar Disco/Artista



Figura 07 – Diagrama de Atividades – Troca de mensagens



Figura 08 – Diagrama de Atividades – Comentar/Classificar



Figura 09 – Diagrama de Atividades – Adicionar disco ao perfil



Figura 10 – Diagrama de Atividades – Ver discos adicionados ao perfil



Figura 11 – Diagrama de Atividades – Alterações nos discos



Figura 12 – Diagrama de Atividades – Validar alterações

5.3.4 Diagrama de Sequência\*

Diagrama de sequência (ou Diagrama de Sequência de Mensagens) é um diagrama usado em UML (Unified Modeling Language), representando a sequência de processos (mais especificamente, de mensagens passadas entre objetos) num programa de computador. Como um projeto pode ter uma grande quantidade de métodos em classes diferentes, pode ser difícil determinar a sequência global do comportamento. O diagrama de sequência representa essa informação de uma forma simples e lógica.



Figura 13 – Diagrama de Sequência – Buscar disco



Figura 14 – Diagrama de Sequência – Buscar artista/gênero



Figura 15 – Diagrama de Sequência – Interações do usuário



Figura 16 – Diagrama de Sequência – Interações entre usuários



Figura 17 – Diagrama de Sequência – Interações ADM

5.3.5 Diagrama de Estados\*

Um diagrama de transição de estados é uma representação do estado ou situação em que um objeto pode se encontrar no decorrer da execução de processos de um sistema. Com isso, o objeto pode passar de um estado inicial para um estado final através de uma transição.

Estado: Condição ou situação durante a vida de um objeto na qual ele satisfaz algumas condições, executa algumas atividades ou espera por eventos.

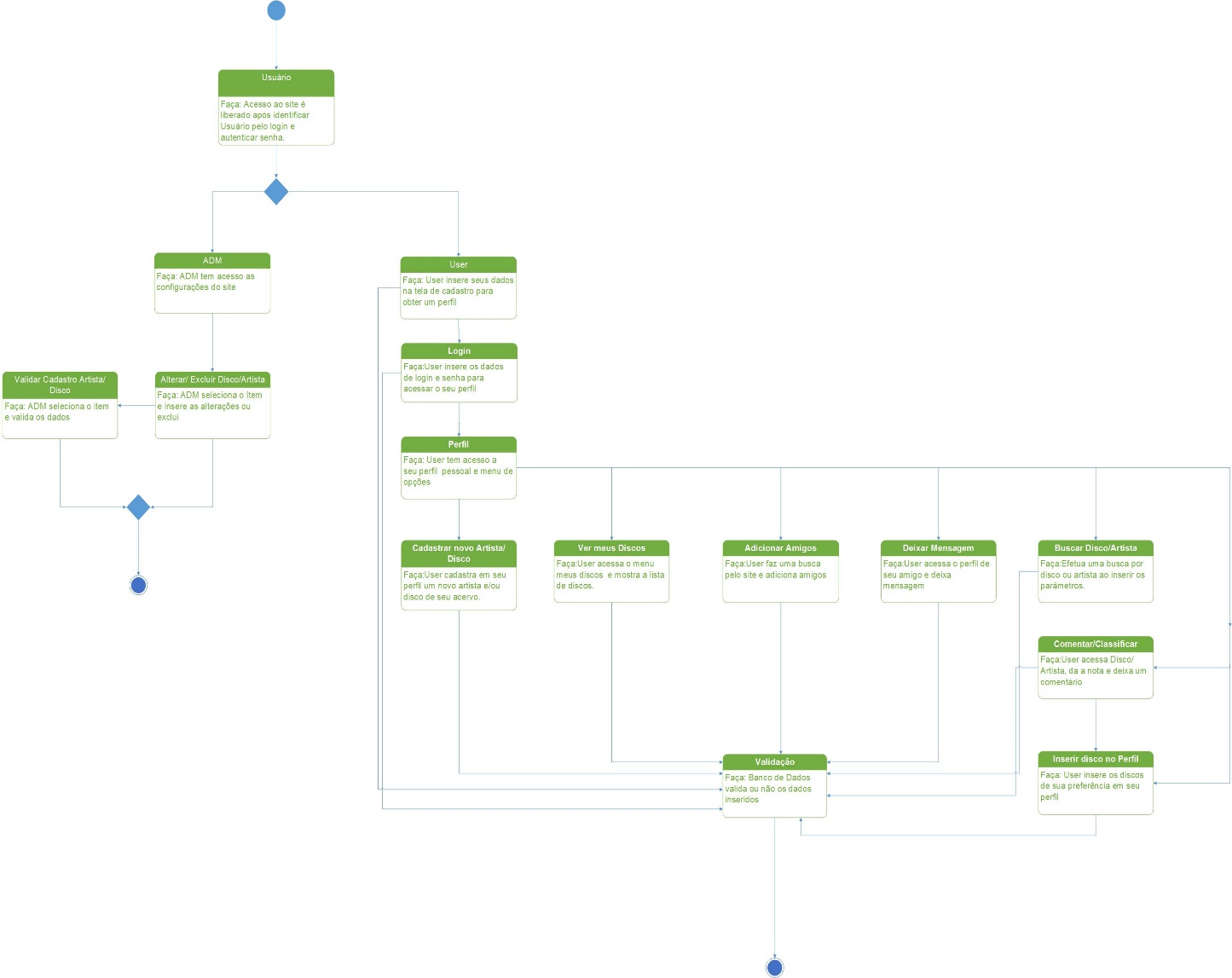


Figura 18 – Diagrama de Estados – Interações ADM

6 RESULTADOS\*

Apresente aqui os resultados finais obtidos. Utilize tabelas, gráficos, fotos ou ilustrações que facilitem a apresentação ordenada dos resultados. Aqui você deve apresentar todos os objetivos alcançados e que servem como embasamento para as considerações finais de seu projeto. Cuidado! Amontoar dados sem nenhuma organização ou relevância é uma das formas mais comuns de “enrolar” em trabalhos que não evoluíram.

7 ANÁLISE E INTERPRETAÇÃO DOS DADOS\*

À luz do marco teórico faça a relação entre os resultados obtidos e o que a teoria propõe. Ou seja, apresente os dados encontrados e vá fazendo um paralelo com a teoria existente.

Exemplo:

No estudo observou-se que este trabalho proporcionou aos funcionários da empresa uma melhor realização de suas tarefas. Esta situação está de acordo com o proposto por MORAES (1998, p. 58) de acordo com este, a qualidade de trabalho se deve a “colocar o pensamento deste autor que confirma a sua idéia (coloque uma frase ou um parágrafo da obra em questão)”.

8 SUGESTÕES PARA MELHORIA DO PROCESSO\*

O site futuramente vai operar com usuários fazendo negociações de compra, venda e troca de seus vinis com outros usuários.

Usuários de redes como Facebook, Twitter, também poderão se interagir com o nosso site.

9 CONSIDERAÇÕES FINAIS\*

Para concluirmos este projeto, utilizamos vários métodos de pesquisa, incluindo consultas de páginas web, pesquisa com o público e referências. Cada um dos itens neste projeto contribuiu com informações suficientes para que pudéssemos amadurecer sobre a rede social de discos de vinil B-Side e chegar a um resultado que agradou a dupla.

Um dos desafios do projeto foi lidar com escasso tempo que tínhamos para encontrar um caminho, para não deixar o trabalho repleto de conceitos e ideias superficiais. Gastamos muito tempo na execução do projeto realizando pesquisas, fazendo e refazendo várias vezes até chegar a um produto bem sucedido. É algo complexo, mas gratificante. A dupla sente que o trabalho apresentado expressa as nossas opiniões e tem a marca da dupla no relatório final.

Como trabalho final do curso a dupla concorda que agregou conhecimento e muitas das técnicas aprendidas durante o curso foram aplicadas aqui, não descartamos nenhum aprendizado e nossa mente esta cheia de novas ideias e conceitos sobre desenvolvimento web.

10 REFERÊNCIAS BIBLIOGRÁFICAS

"ESCOLA". **Wikipedia**, 2016. Disponivel em: <https://pt.wikipedia.org/wiki/Escola>. Acesso em: 26 junho 2018.

"INTERNET". **Wikipedia**, 2015. Disponivel em: <https://pt.wikipedia.org/wiki/Internet>. Acesso em: 26 jun. 2018.

"SCHEDULE". **Wikipedia**, 2015. Disponivel em: <https://en.wikipedia.org/wiki/Schedule>. Acesso em: 26 junho 2018.

PICHETTI, J. G. B. **Sistema Web Para Gerenciamento de Bancas**. Pato Branco. 2013.

SILVA, J. J. C. Gestão escolar participada e clima organizacional. **Gestão em Ação**, Salvador, dez. 2001. 108. Disponivel em: <http://www.gestaoemacao.ufba.br/revistas/gav4n201.PDF>. Acesso em: 21 junho 2018.

SOUZA, R. S. C. D. **Sistema Web para Departamentos**. Ouro Preto. 2010. (Universidade Federal de Ouro Preto).

# 

# ANEXOS\*

Cronograma

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CRONOGRAMA B-Side | | | | |
|
|
| **AÇÕES** | **INÍCIO ESTIMADO** | **INÍCIO REAL** | **FIM ESTIMADO** | **FIM REAL** |
| Escolha do tema | 28/07/2014 | 28/07/2014 | 01/08/2014 | 27/08/2014 |
| Descrever Situação Problema | 31/07/2014 | 31/07/2014 | 31/07/2014 | 31/07/2014 |
| Descrever Requisitos | 31/07/2014 | 31/07/2014 | 31/07/2014 | 06/08/2014 |
| Descrição de Cenários | 06/07/2014 | 06/07/2014 | 13/08/2014 | 17/08/2014 |
| Tabela de Requisitos | 06/07/2014 | 06/07/2014 | 13/08/2014 |  |
| Plano de Testes | 02/09/2014 | 02/09/2014 | 30/09/2014 | 05/10/2014 |
| Diagrama de Caso e Uso | 03/09/2014 | 02/09/2014 | 08/10/2014 | 08/10/2014 |
| Diagrama de Classes | 04/09/2014 | 02/09/2014 | 08/10/2014 | 08/10/2014 |
| Diagrama de Atividades | 05/09/2014 | 02/09/2014 | 08/10/2014 | 25/10/2014 |
| Diagrama de Sequência | 06/09/2014 | 25/10/2014 | 08/10/2014 | 25/10/2014 |
| Diagrama de Estados | 07/09/2014 | 25/10/2014 | 08/10/2014 | 28/10/2014 |
| UML | 19/09/2014 | 19/09/2014 | 19/09/2014 | 19/09/2014 |
| Banco de Dados | 19/09/2014 | 19/09/2014 | 08/11/2014 |  |
| Diagramação das Páginas | 13/10/2014 | 13/10/2014 | 10/11/2014 |  |
| Criar Classes | 29/10/2014 | 29/10/2014 | 10/11/2014 |  |
| Programar as Páginas | 10/11/2014 |  | 15/11/2014 |  |
| Trabalho Impresso | 01/08/2014 | 01/08/2014 | 15/11/2014 |  |
| Ajustar Interface | 22/10/2014 | 23/10/2014 | 29/11/2014 |  |
| Testar | 15/11/2014 |  | 19/11/2014 |  |