

FACULDADE DE TECNOLOGIA DE SOROCABA
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Felipe Gonelli Paes

**ADS Library – Projeto de Biblioteca Digital de Trabalhos de Graduações -
Metodologias e Ferramentas**

Sorocaba
Julho/2016



**GOVERNO DO ESTADO
DE SÃO PAULO**

**ADS Library – Projeto de Biblioteca Digital de Trabalhos de Graduações -
Metodologias e Ferramentas**

Felipe Gonelli Paes

Orientadora: Prof.^a M.^a Denilce de Almeida Oliveira Veloso

Sorocaba
Julho/2016

AGRADECIMENTOS

Agradeço a todo o apoio e auxílio que tive durante o desenvolvimento deste projeto dos meus amigos, colegas, familiares e a minha orientadora, onde no momento quando tudo parecia perdido não me deixaram desistir.

"O insucesso é apenas uma oportunidade para
recomeçar de novo com mais inteligência. "

(Ford, Henry)

RESUMO

Em uma Biblioteca Pública Digital ou uma Biblioteca Digital Comunitária, é possível o acesso universal em qualquer lugar que tenha computadores disponíveis conectados a Web, o que permite a qualquer pessoa ler, estudar, aprender e interagir com um universo literário. Permite, aos leitores de todo o mundo, o acesso direto (ou através de *download*) a livros no formato eletrônico. Desde a reestruturação das competências do Curso Superior de Processamento de Dados para Tecnologia em Análise e Desenvolvimento de Sistemas nas Faculdades de Tecnologias do Estado de São Paulo no ano de 2008, o critério de aprovação para obter o diploma de tecnólogo passou a ser a avaliação, por uma banca composta de docentes, do Trabalho de Graduação desenvolvido e entregue individualmente por cada aluno ao final do curso. A Faculdade de Tecnologia de Sorocaba “José Crespo Gonzales” conta com um grande acervo físico desses trabalhos, muitos de altíssima qualidade, e com temas interessantes que podem ser continuados/implementados. Porém, esses trabalhos não estão centralizados de forma informatizada, além da dificuldade na busca das informações, esse controle dificulta e às vezes até desestimula os alunos e professores a terem acesso aos mesmos. Dentro desse contexto, surgiu a ideia de se criar uma Biblioteca Digital de Trabalhos de Graduações chamada *ADS Library*, para disponibilizar as informações e conteúdo desses trabalhos de forma simples e organizada. O projeto está sendo realizado em equipe, porém cada membro tem as suas responsabilidades. Este relatório pretende mostrar as atividades realizadas durante projeto de iniciação científica com relação à parte da pesquisa das Metodologias e Ferramentas a serem utilizadas no projeto.

Palavras-Chave: Engenharia de Software, Biblioteca Digital, Desenvolvimento Web, Java.

ABSTRACT

In a public digital library or in a community digital library is possible the universal access everywhere that has a computer connect to the internet, so everyone can read, study, learn and interact with a whole literary universe. It enables readers around the world, direct access or by downloading books in an electronic format. Since the restructuring of the responsibilities of the undergraduate degree of the Data Processing for Technology Analysis and Systems Development in Universities of São Paulo State in 2008, the criteria for approval for the technologist diploma became an evaluation, by an evaluation board composed by teachers. The Graduate Work must be developed and delivered individually by each student at the end of the course. The University of Technology of Sorocaba "José Crespo Gonzales" has a great physical collection of these works, many of them with high quality, and interesting topics that can be continued or implemented. However, these works are not centralized in a computerized way, besides the difficulty in finding the information, this situation difficult and sometimes discourages students and teachers to have access to them. In this context, the idea of creating a Digital Library of Graduations works called ADS Library came up to make information and content of these works available in a simple and organized way. The project is being developed as a team, but each member has their responsibilities. The goal of this report is to show the activities executed during the research project regarding part of the research methodologies and tools to be used in the project.

Keywords: Software Engineering, Digital Library, Web Development, Java.

LISTA DE ILUSTRAÇÕES

Figura 1 – Java.....	18
Figura 2 – Spring.....	18
Figura 3 – MySQL	19
Figura 4 – Trello	19
Figura 5 – PrimeFaces.....	19
Figura 6 – Hibernate.....	20
Figura 7 – Maven	20
Figura 8 – Git.....	21
Figura 9 – Integração entre as Tecnologias	21
Figura 10 – Estrutura da aplicação.....	25
Figura 11 – Tela inicial do sistema	27
Figura 12 – Tela de informações do sistema.....	28
Figura 13 – Tela de termos do sistema	29
Figura 12 – Tela de busca de monografias	30
Figura 15 – Tela de busca de monografias com resultados.....	32
Figura 16 – Tela de detalhamento da monografia.....	32
Figura 17 – Tela de detalhamento da monografia – Usuário logado.....	34
Figura 18 – Tela de login.....	35
Figura 19 – Tela de registro de funcionário	36
Figura 20 – Tela de consulta de funcionário	37
Figura 21 – Tela de registro de monografia	38
Figura 22 – Tela de consulta de monografia	40
Figura 23 – Tela de edição de monografia.....	41
Figura 24 – Tela de consulta de autores	43
Figura 25 – Tela de registro de instituição	43
Figura 26 – Tela de registro de instituição	45
Figura 27 – Tela de registro de cursos.....	46
Figura 28 – Tela de consulta de cursos	47
Figura 29 – Tela de registro de orientadores.....	48
Figura 30 – Tela de consulta de orientadores	49
Figura 31 – Tela de palavras chave	50

Figura 32 – Tela de reportar problemas ou sugestões.....	51
Figura 33 – Tela de geração de relatórios.....	52
Figura 34 – Tela de geração de relatórios – Relatório gerado	53

LISTA DE ABREVIATURAS E SIGLAS

API – *Application Programming Interface*

DAO – *Data Access Object*

Java EE – *Java Enterprise Edition*

JSF – *Java Server Faces*

JSP – *Java Server Pages*

JSTL – *Java Standard Tags Library*

MVC – *Model, View, Controller*

ORM – *Object Relational Mapping*

POM – *Project Object Model*

RJI – *Regime de Jornada Integral*

SQL – *Structured Query Language*

UML – *Unified Modeling Language*

W3C – *World Wide Web Consortium*

XML – *Extensible Markup Language*

SUMÁRIO

1 INTRODUÇÃO E JUSTIFICATIVA.....	10
2 OBJETIVOS.....	12
3 MATERIAIS E MÉTODOS.....	13
4 RESULTADOS E DISCUSSÕES.....	14
4.1 Desenvolvimento de Software	14
4.1.1 Fase de Definição Planejamento do <i>software</i> :	14
4.1.2 Fase de Desenvolvimento	14
4.1.3 Fase de Verificação, Liberação e Manutenção Realização de testes ..	14
4.2 Conceitos utilizados no desenvolvimento:	14
4.2.1 Metodologias de Desenvolvimento	15
4.2.2 Método.....	15
4.2.3 Técnica	15
4.2.4 Notação	15
4.3 UML	15
5 FERRAMENTAS.....	17
5.1 Java.....	17
5.2 Spring.....	18
5.3 MySQL.....	18
5.4 Trello	19
5.5 PrimeFaces.....	19
5.6 Hibernate	20
5.7 Maven.....	20
5.8 Git.....	21
5.9 Integração entre as Ferramentas.....	21
5.10 Tecnologias	22
5.10.1 Facelets	23
5.10.2 Servlets.....	23
5.10.3 JSTL	23
5.10.4 JSP	23
5.10.5 JSF	23
6 AMBIENTES.....	24
6.1 Eclipse	24

7 PROJETO DETALHADO DE SOFTWARE	25
7.1 Arquitetura.....	25
7.2 Telas.....	26
7.2.1 Página inicial.....	27
7.2.2 Sobre	28
7.2.3 Termos	29
7.2.4 Buscar monografia.....	30
7.2.5 Detalhar monografia	32
7.2.6 Login.....	35
7.2.7 Funcionários	36
7.2.8 Monografia.....	38
7.2.9 Autores	43
7.2.10 Instituições.....	43
7.2.11 Cursos	46
7.2.12 Orientadores	48
7.2.13 Palavras-Chave	50
7.2.14 Reportar.....	51
7.2.15 Relatórios	52
8 CODIFICAÇÃO.....	54
8.1 Script.....	54
8.2 Código Fonte	57
8.3 Instalação	57
8.4 Trabalhos Futuros	57
CONCLUSÃO	59
REFERÊNCIAS.....	60
GLOSSÁRIO.....	62

1 INTRODUÇÃO E JUSTIFICATIVA

A palavra “biblioteca” tem sua origem do grego *biblíon* (livro) e *teke* (caixa, depósito), portanto um depósito de livros (HOUAISS, 2001).

O final da Segunda Guerra Mundial trouxe o computador e a informática para facilitar o trabalho nas bibliotecas. Todavia, de imediato ele não foi compreendido assim. Com o transcorrer do tempo, o computador diminuiu de tamanho e aumentou sua potência, saindo dos ambientes dos laboratórios de cientistas, passando a ter uso pessoal. O desenvolvimento da informática possibilitou a criação da internet que rompeu com a comunicação unidirecional (MILANESI, 2002).

Biblioteca Digital é aquela que contempla documentos gerados ou transpostos para o ambiente digital (eletrônico), um serviço de informação (em todo tipo de formato), no qual todos os recursos são disponíveis na forma de processamento eletrônico (aquisição, armazenagem, preservação, recuperação e acesso através de tecnologias digitais). (ROSETTO, 2002)

"Uma biblioteca digital é a coleção de serviços e de objetos de informação, com organização, estrutura e apresentação que suportam o relacionamento dos utilizadores com os objetos de informação, disponíveis direta ou indiretamente via meio eletrônico / digital." (LEINER, 1998)

A Universidade é um espaço para a produção e transferência de conhecimentos. Porém, as teses, dissertações, trabalhos de graduação e produção científica (resultado dos projetos/pesquisas) muitas vezes encontram-se disponibilizados somente aos próprios pesquisadores.

Partindo desse pressuposto, a biblioteca digital é a democratização do conhecimento e a universalização da informação.

Na Faculdade de Tecnologia de Sorocaba "José Crespo Gonzales", o foco principal para desenvolvimento deste projeto são os trabalhos de graduação dos alunos do Curso de Análise e Desenvolvimento de Sistemas. Há um grande acervo físico de trabalhos, muitos de altíssima qualidade, e com temas interessantes que podem ser continuados/implementados, porém com difícil acesso para consultas. Propõe-se nesse trabalho o desenvolvimento de uma ferramenta de Biblioteca Digital no formato *Web* para facilitar a pesquisa desses trabalhos ampliando o acesso e permitindo que os alunos possam pesquisar, consultar e fazer o *download* dos mesmos.

Nesse relatório será dada ênfase a Pesquisa de Metodologias e Ferramentas, que foi realizada em paralelo com o levantamento inicial dos requisitos. As demais fases estão contempladas nos relatórios dos outros componentes da equipe, porém devido a desistência de um integrante em vista a este projeto, a fase de desenvolvimento também será incluída neste relatório.

Este projeto faz parte do projeto de RJI da Prof.^a M^a Denilce de Almeida Oliveira Veloso chamado "Integração entre Dispositivos Móveis e Web". O livro "Engenharia de software: uma abordagem profissional" (R. Pressman), foi uma referência importante para estudo das metodologias. Quanto à parte das ferramentas, as pesquisas deram-se em sites oficiais dos produtos, ou na ausência destes, em outros sites e livros especializados sobre o assunto, a referência principal foi o livro "Programação Java para a Web" (Luckow & Melo), que oferece uma grande base para o desenvolvimento de um projeto Java, além de fornecer descrição e explicações das principais ferramentas e tecnologias utilizadas na plataforma Java.

2 OBJETIVOS

Desenvolver uma aplicação em plataforma *Web*, ampliando o acesso e permitindo que os alunos da Fatec Sorocaba, do curso de Análise e Desenvolvimento de Sistemas, possam pesquisar, consultar e fazer o *download* dos trabalhos de graduação.

3 MATERIAIS E MÉTODOS

Primeiramente o levantamento de requisitos iniciou com pesquisas sobre o assunto, foi utilizado o método de entrevista com as bibliotecárias da Fatec Sorocaba durante a fase de Análise de Requisitos e Criação dos Diagramas, que está descrita no relatório do aluno Ronaldo Akio L. Yamashita (YAMASHITA, 2016). Em paralelo foram realizadas pesquisas bibliográficas sobre metodologias, ambientes e ferramentas de desenvolvimento, responsabilidades deste autor e em seguida começa a fase de desenvolvimento que deveria estar descrita no relatório do aluno Júlio César Fortunato Filho, porém com o desligamento do mesmo no projeto, o desenvolvimento do projeto foi de responsabilidade deste autor.

4 RESULTADOS E DISCUSSÕES

Os resultados obtidos neste projeto foram positivos e dentro do esperado. Para garantir de certa forma, que tudo está foi realizado de forma correta, foram estudadas metodologias de desenvolvimento que serviram como guias, para assegurar a qualidade e produtividade de um processo de desenvolvimento de *software*, portanto foram discutidas algumas abordagens e componentes que fazem parte do desenvolvimento de *software*.

4.1 Desenvolvimento de Software

O desenvolvimento, segundo Pressman (2015) é realizado em três fases.

4.1.1 Fase de Definição Planejamento do *software*:

Descrição do escopo, análise do esforço, análise de riscos, levantamento dos recursos exigidos, estimativas de custos e de prazos. O objetivo é fornecer uma indicação da viabilidade do *software*; fase de análise e requisitos do *software*: a análise forma do domínio da informação é utilizada para estabelecer modelos de fluxo de dados e da estrutura da informação. Alternativamente pode ser feito um protótipo. Estes modelos são detalhados para se tornar uma especificação do *software*, que é o documento produzido com resultado desta fase.

4.1.2 Fase de Desenvolvimento

Descrição de estrutura modular, definição de interfaces, uma estrutura de dados é estabelecida. Uma especificação de projeto é produzida. E a codificação é realizada.

4.1.3 Fase de Verificação, Liberação e Manutenção Realização de testes

Para descobrir o máximo de erros. Faz-se a manutenção do *software* ao longo da sua vida útil.

4.2 Conceitos utilizados no desenvolvimento:

O dicionário (WEBSTERS, 1998) define metodologia como um conjunto de métodos, regras e postulados empregados por uma disciplina: um procedimento particular ou conjuntos de procedimentos.

É objetivo de uma metodologia definir de forma clara “*quem*” faz “*o que*”, “*quando*”, “*como*”, e até mesmo “*onde*”, para todos os que estejam envolvidos diretamente ou não com o desenvolvimento de *software*. Deve definir também qual o papel dos técnicos, dos usuários, e o da administração da empresa no processo de desenvolvimento. Com isso, evita-se a situação a qual o conhecimento sobre o sistema é de poucos, comumente apelidados, de “os donos do sistema”. Além disso, deve instruir um conjunto de padrões preestabelecidos, de modo a ser evitar a subjetividade na abordagem, a fim de garantir fácil integração entre os sistemas desenvolvidos. (DEVMEDIA,2015).

4.2.1 Metodologias de Desenvolvimento

Maneira de se utilizar um conjunto coerente e coordenado de métodos para atingir um objetivo. Em outras palavras, a metodologia deve definir quais as fases de trabalho previstas no desenvolvimento de sistemas.

4.2.2 Método

É um procedimento a ser adotado para se atingir um objetivo.

4.2.3 Técnica

É um modo apropriado de se investigar sistematicamente um determinado universo de interesse ou domínio de um problema. Ex: análise estruturada, análise essencial e projeto estruturado.

4.2.4 Notação

É um conjunto de caracteres, símbolos e sinais formando um sistema convencionado de representação.

4.3 UML

Os conceitos da orientação a objetos são bem difundidos, desde o lançamento da primeira linguagem orientada a objetos, a linguagem SIMULA. O desenvolvimento de sistemas de *software* suportados por métodos de análise e projeto que modelam esse sistema de modo a fornecer para toda a equipe envolvida uma compreensão completa do projeto. A UML (*Unified Modeling Language*) é a sucessora de um conjunto de métodos de análise e projeto orientados a objeto. A

UML é um modelo de linguagem, não um método. Um método pressupõe um modelo de linguagem e um processo. O modelo de linguagem é a notação que o método usa para descrever o projeto. O processo são os passos que devem ser seguidos para se construir o projeto.

A UML define uma notação e um meta-modelo. A notação representa a sintaxe da linguagem composta por elementos gráficos. Um meta-modelo é um diagrama de classe. A UML foi desenvolvida por Grady Booch, James Rumbaugh, e Ivar Jacobson que são conhecidos como "os três amigos". A UML é a junção do que havia de melhor nas três metodologias e que foi adicionado novos conceitos e visões da linguagem. (JÚNIOR, 2010).

Abstração - Permite ignorar os aspectos de um assunto não relevante para o propósito. Diminui a complexidade.

Objeto - É alguma coisa que pode ser identificada distintamente. Qualquer coisa, real ou abstrata, a respeito da qual armazenamos dados e os métodos que os manipulam (Martin/Odell, 1995). Uma entidade capaz de armazenar um estado (informação) e que oferece um número de operações (comportamento) para ou consultar ou alterar o estado (Jacobson, 1992). É algo que possui estado, comportamento e identidade (Booch, 1996).

Classe - Pode ser vista como uma fábrica de objetos idênticos. Possui atributos e métodos. Uma descrição de um conjunto de objetos que compartilham os mesmos atributos, operações, relacionamentos e semântica. (Rumbaugh, 1994). Um gabarito para diversos objetos que descreve como estes objetos são estruturados internamente (Jacobson, 1992). Um conjunto de objetos que compartilham uma estrutura comum e um comportamento comum (Booch, 2000).

Após uma pesquisa de ferramentas/linguagens com possibilidade de utilização no projeto, a equipe chegou a um comum acordo. As ferramentas que serão/estão sendo utilizadas, estarão na listagem a seguir.

5 FERRAMENTAS

Antes de aqui descrever as ferramentas utilizadas, é importante relatar que antes de consolidar a escolha das mesmas, foram cogitados outros tipos de tecnologias, como por exemplo a linguagem PHP, muito utilizada em ambientes *web*. Necessariamente este tópico não se prende ao termo “ferramentas”, em suma é mais amplo que isto, já que o que aqui se define são tecnologias que serão ferramentas para o desenvolvimento do projeto. Uma ferramenta pode ser também algum tipo de tecnologia, um *framework* ou metodologia.

A razão de cogitar o PHP como escolha seria pelo fato de que o ambiente atual da Faculdade de Tecnologia de Sorocaba oferece suporte somente a essa linguagem, simplesmente para facilitar a implementação do sistema futuramente. Para isso foram estudados alguns *frameworks* baseados na linguagem, um exemplo deles foi o CakePHP, o mesmo não será abordado neste documento. Aprofundando mais a pesquisa essa ideia logo foi abandonada, basicamente pela base de ensino do curso ser sob a linguagem Java, seguido do fato de que o desenvolvimento será feito por alunos. A conclusão é que escolhendo o Java como plataforma base do projeto, o tempo gasto no aprendizado de uma nova linguagem, um *framework* com uma base diferente foi evitado, tendo um maior aproveitamento ao usufruir os conhecimentos em uma plataforma já conhecida, onde além de aumentar o contato com a linguagem, é uma oportunidade para aprofundamento na mesma, além do aprendizado de um novo *framework*, no caso o PrimeFaces.

Essa escolha facilitou tanto para o início do projeto, quanto ao seu gerenciamento em modo geral. Também com ela foi possível garantir o bom desenvolvimento do projeto, pelo oferecimento dos inúmeros recursos que possui.

5.1 Java

Java (Figura 1) é uma linguagem de programação orientada a objeto desenvolvida na década de 90 pela empresa *Sun Microsystems*. Diferentemente das linguagens convencionais, que são compiladas para código nativo, a linguagem Java é compilada para um *byte code* que é executado por uma máquina virtual.

Não só uma ferramenta mais uma tecnologia muito popular, uma das mais conhecidas plataformas de desenvolvimento. A linguagem é orientada a objetos e tem como principais características: sua portabilidade, segurança e independência

de sistemas operacionais, ou seja, através da utilização de uma máquina virtual Java, é possível que seus códigos sejam interpretados por qualquer sistema operacional. Sua principal regra é “*Write once run anywhere*” (WORA), ou seja, é uma linguagem escrita uma vez, e executada em qualquer lugar. (JAVA, 2015).

Figura 1 – Java



Disponível em: <https://en.wikipedia.org/wiki/File:Java_logo.png>

5.2 Spring

Framework utilizado juntamente com a plataforma Java, criada com intuito de simplificar a sua programação (Figura 2). É encarregado de realizar a inversão de controle e a injeção de dependências, ou seja, é responsável por realizar a instanciação de classes Java além de definir as dependências entre elas através de um arquivo XML ou anotações Java. Assim, permite-se realizar o baixo acoplamento entre suas classes. (SPRING,2015).

Figura 2 – Spring



Disponível em: <<https://spring.io/img/spring-by-pivotal.png>>

5.3 MySQL

Sistema de gerenciamento de banco de dados livre, popular, com bom suporte e compatibilidade (Figura 3). Utiliza linguagem SQL, possui várias ferramentas e consome poucos recursos. Ideal para uso em sistemas *Web*. (MYSQL,2015).

Figura 3 – MySQL



Disponível em: <<https://www.mysql.com/about/legal/logos.html>>

5.4 Trello

O Trello (Figura 4) é um sistema de quadro virtual para gerenciamento de tarefas muito usado no desenvolvimento com métodos ágeis. Ele permite a criação de diversos quadros, facilitando a divisão de tarefas, oferecendo também um gerenciamento de prazos, equipes e atividades. (TRELLO,2015)

Figura 4 – Trello



Disponível em: <<https://trello.com/about/branding>>

5.5 PrimeFaces

O PrimeFaces (Figura 5) é um *framework* de desenvolvimento *front end* para sistemas *web*, utilizando componentes JSF seu objetivo é trazer facilidade e praticidade no desenvolvimento, com componentes em sua maioria responsivos, garante a adaptação em diferentes tipos de dispositivos melhorando assim a interação com o usuário. (PRIMEFACES,2015).

Figura 5 – PrimeFaces



Disponível em: < <https://en.wikipedia.org/wiki/File:Pf-logo.png> >

5.6 Hibernate

É um *framework* para realizar o mapeamento objeto relacional (ORM) escrito na linguagem Java (Figura 6), onde seu principal objetivo é diminuir a complexidade envolvido no desenvolvimento de aplicações que necessitam trabalhar com banco de dados relacional, onde ele realiza a intermediação entre o banco de dados e sua aplicação, poupando o desenvolvedor de ter que se preocupar com instruções SQL para recuperar ou persistir os dados do seu *software*. (HIBERNATE, 2015).

Figura 6 – Hibernate



Disponível em: < <https://en.wikipedia.org/wiki/File:Hibernate.png> >

5.7 Maven

O Apache Maven é um *software* de gerenciamento de projeto (Figura 7), baseado no conceito de POM, ele pode gerenciar as informações do projeto, repositórios, *plug-ins*, compilações, execução de testes, além de centralizar e controlar as dependências do projeto.

Figura 7 – Maven



Disponível em: < <https://maven.apache.org/images/maven-logo-black-on-white.png> >

5.8 Git

O Git (Figura 8) é uma ferramenta de controle de versão que registra as mudanças realizadas em um arquivo ou um conjunto de arquivos de um projeto, a partir deles é possível recuperá-los, voltar a uma versão anterior, mesclar arquivos com conflitos. Com essa ferramenta é possível gerenciar cada alteração feita durante o desenvolvimento do projeto, ter controle de quem está desenvolvendo o código, evitando problemas de versão e perda de código, melhorando a produtividade no desenvolvimento.

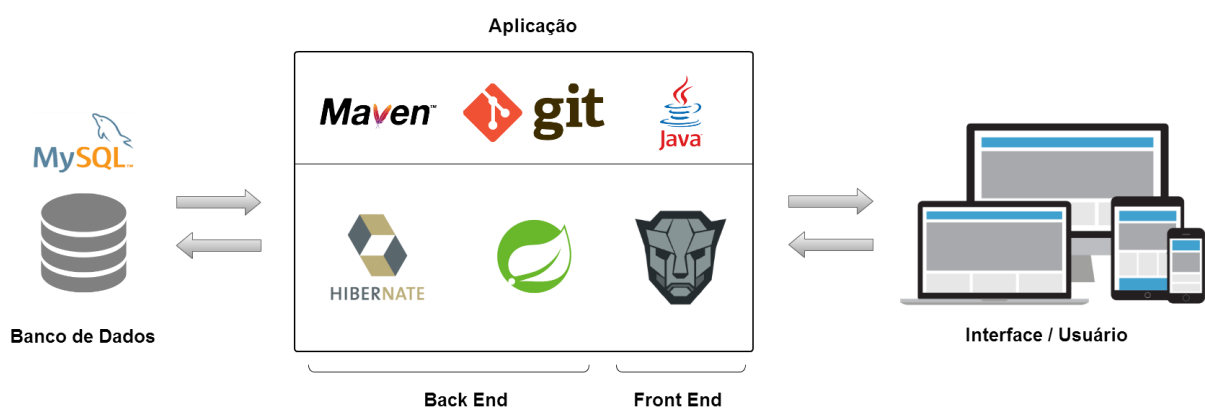


Disponível em: <<https://git-scm.com/images/logos/downloads/Git-Logo-2Color.png>>

5.9 Integração entre as Ferramentas

Após uma explicação geral de cada tecnologia escolhida, segue abaixo um diagrama (Figura 9) que exemplifica como é a integração entre elas:

Figura 9 – Integração entre as Tecnologias



Fonte: Autor.

Primeiramente temos uma Base de Dados que utiliza um Banco de Dados Relacional (MySQL), nela serão armazenadas todas as informações referentes a aplicação. Para gravar e ler essas informações utilizaremos um mediador, esse

mediador deve gerenciar a transcrição dos dados para um objeto na aplicação através do mapeamento objeto relacional, por esta razão é utilizado o Hibernate, que nada mais é que um facilitador de comunicação entre a aplicação e o banco de dados, mas muito mais que isso ele é uma poderosa ferramenta, capaz de abstrair completamente o conceito de persistência relacional, facilitando muito o desenvolvimento.

No *core* da aplicação, também chamado de *back end*, temos o Java em conjunto com o Git e o Maven. O Java disponibiliza todos os recursos do projeto, o Git gerencia os arquivos do projeto, podendo armazenar o projeto em um repositório online e administrar as alterações de código feitas por cada desenvolvedor e o Maven responsável por controlar todas as dependências utilizadas no projeto. Como *frameworks* temos o Spring, que funciona em conjunto com o Java (por tê-lo como base), processando todas as informações das camadas da aplicação, nessa camada são aplicadas todas as validações e regras de negócio, além de enviar as informações para o banco de dados. (A estrutura da aplicação será melhor descrita no relatório de desenvolvimento deste projeto).

No *front end*, temos a parte visual do sistema, responsável por receber e exibir informações na interface, basicamente o responsável por isso é o PrimeFaces, ele é executado com um conjunto de diversas tecnologias, como JSF, JSP, JSTL, Servlets e Facelets. O *front end* é uma parte mais separada da aplicação por ter a interface em seu controle, mas com o uso de *frameworks* Java para essa camada, como o já mencionado PrimeFaces, a tendência é que ela fique mais próxima do *back end*, trabalhando de forma um pouco diferente, mas ainda sim produzindo bons resultados e boa resposta ao usuário, no quesito *interface*.

O conjunto dessas tecnologias e camadas formam a aplicação, onde a indisponibilidade de algum componente pode inviabilizar o uso da mesma.

5.10 Tecnologias

Algumas tecnologias são incorporadas, interligadas ou dependentes de alguma forma, não menos importante, as descritas aqui fazem parte das ferramentas definidas.

5.10.1 Facelets

É um processador de telas, uma linguagem de declaração visual, com ele é possível construir *templates* e criar componentes, garante bom desempenho, suporta reutilização de código e é compatível com todos os componentes JSF.

5.10.2 Servlets

Servlet é uma API Java que dá a uma classe características de um servidor *web*, onde é possível processar requisições e produzir resultados, usado geralmente para criação de páginas dinâmicas.

5.10.3 JSTL

Significa *Java Server Pages Standard Tag Library*, é uma API que contém diversas tags de controles de laços, fluxos, manipulação de dados xml e controle de internacionalização.

5.10.4 JSP

É uma tecnologia que permite a criação de páginas dinâmicas, proporciona rápido desenvolvimento e fácil manutenção. Ele é independente de plataforma e separa a interface das informações exibidas, permitindo alterações sem mudança do conteúdo dinâmico.

5.10.5 JSF

Conhecido como Java Server Faces, é um conjunto de APIs com componentes de interface, suportando gerenciamento de estado, manipulação de eventos, validação de entradas, navegação de páginas, internacionalização e acessibilidade.

O JSF é usado dentro de uma página JSP no formato de tag, ele possui diversos componentes já prontos, para diminuir a preocupação de desenvolvimento com html e javascript. O modo como ele é estruturado, garante suas camadas de apresentação e aplicação bem divididas, utilizando o conceito de MVC.

6 AMBIENTES

O Ambiente de desenvolvimento muitas vezes influencia na produtividade do projeto, o *software* selecionado foi o eclipse, por fácil adaptação de uso e por ser livre.

6.1 Eclipse

IDE para desenvolvimento Java, é um ambiente completo que oferece diversos recursos para o desenvolvimento. Uma ferramenta *Open Source*, que atualmente é a mais utilizada para o desenvolvimento¹, além do Java também suporta várias outras linguagens, é personalizável, tem suporte a inúmeros *plug-ins* e extensões e possui uma grande comunidade para soluções de problemas.

¹ Java Tools and Technologies Landscape for 2014. Consultado em 22 de fevereiro de 2016.

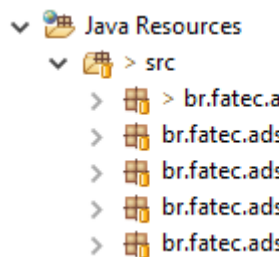
7 PROJETO DETALHADO DE SOFTWARE

A partir deste ponto será descrita a fase de desenvolvimento do projeto, o desenvolvimento tem como base o levantamento de requisitos e a escolha das ferramentas e tecnologias a serem utilizadas.

7.1 Arquitetura

O desenvolvimento do projeto é executado na utilização de alguns padrões, esses padrões garantem um grau de confiabilidade, entendimento e organização do código. Primeiramente a estrutura da aplicação segue como base o padrão MVC (Model, View e Controller), cada termo representam camadas que definem o funcionamento de toda a aplicação, essa estrutura é muito utilizada e conhecida, mas a aplicação em si utiliza uma adaptação da mesma.

Figura 10 – Estrutura da aplicação



Fonte: Autor.

A partir da figura 10 podemos observar a estrutura básica da aplicação, ela é composta pelo:

Bean: Esta camada é basicamente o equivalente ao controller do MVC, é responsável por controlar todas as interações entre a tela e os serviços da aplicação, validações e regras de negócio.

Service: Provém os serviços que a aplicação fornece, além de estabelecer a comunicação entre o bean e a camada de acesso aos dados.

DAO: Responsável por toda a comunicação com o banco de dados, fornece uma estrutura para a persistência de dados, se comunica diretamente com a camada de serviço e contém todas as funções relacionadas ao banco de dados.

Model: Camada base da aplicação, define a estrutura dos objetos principais do sistema, em que abstraem para o mundo computacional alguma representação do mundo real.

Util: Camada de utilidades, contém algumas funções gerais utilizadas pela aplicação, como funções de criptografia e tipos enumerados.

7.2 Telas

Segue uma listagem das telas do sistema divididas em nível de acesso, as páginas de acesso público também são acessadas por usuários logados, porém seu comportamento pode ser diferente.

Páginas de acesso público:

- Tela inicial
- Buscar monografia
- Detalhar monografia
- Reportar
- Sobre
- Termos de Uso
- Área Administrativa (Login)

Páginas somente de acesso privado:

- Cadastro de Monografias
- Cadastro de Funcionários (Somente Administrador)
- Cadastro de Orientadores
- Cadastro de Instituições
- Cadastro de Cursos
- Listagem de Palavras Chave
- Listagem de Monografias
- Listagem de Funcionários
- Listagem de Autores
- Listagem de Orientadores
- Listagem de Instituições
- Listagem de Cursos
- Geração de Relatórios

7.2.1 Página inicial

Figura 11 – Tela inicial do sistema

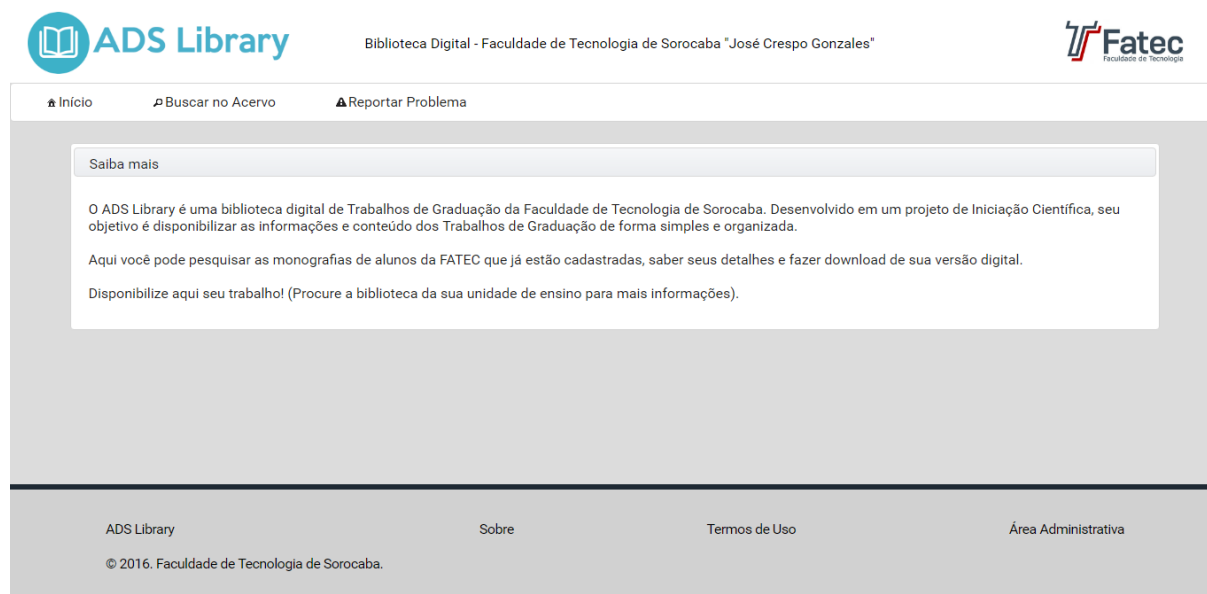


Fonte: Autor.

A tela inicial do sistema (Figura 11) exhibe um cabeçalho com logo e nome da instituição, um menu de acesso para as principais funções do sistema: tela inicial, busca e reportar problema, as últimas monografias cadastradas no sistema e o rodapé com algumas informações e atalhos para a página inicial, sobre, termos de uso e acesso a área administrativa (login). Ao clicar em um título, os detalhes da monografia selecionada serão exibidos e ao selecionar uma palavra chave, será realizada uma busca de monografias com base na palavra chave selecionados.

7.2.2 Sobre

Figura 12 – Tela de informações do sistema

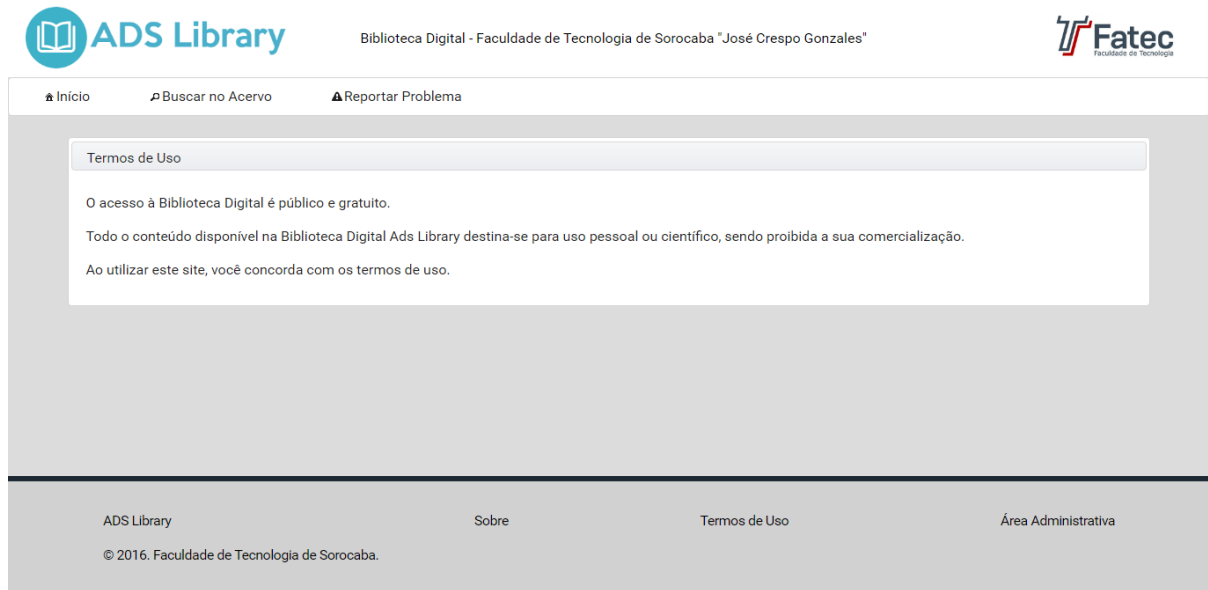


Fonte: Autor.

A tela de informações do sistema (Figura 12) pode ser acessada por um link no rodapé e exibe as informações gerais do projeto, autores e do sistema.

7.2.3 Termos

Figura 13 – Tela de termos do sistema



Fonte: Autor.

A tela de termos de uso (Figura 13) exibe informações gerais sobre as regras de utilização do sistema.

7.2.4 Buscar monografia

Figura 14 – Tela de busca de monografias

Fonte: Autor.

Tela de busca de monografia (Figura 14), através dessa tela é possível realizar a busca de monografias, podendo ser filtradas por palavras chave, autor, orientador, curso, instituição, título, ano de publicação ou todos os campos. Além de poder filtrar o tipo de busca é possível combinar mais termos, bastando apenas separá-los com vírgula. Os resultados da pesquisa são exibidos logo abaixo da caixa de busca.

A busca também pode ser realizada através de uma combinação de termos na URL, combinando o termo pesquisar com o tipo de busca e valor.

Ex.:

www.bibliotecadigital.com.br/pesquisar/autor/maria

A URL acima executa a busca de todas as monografias que possuem um autor com o nome de “maria”. Assim como no campo de pesquisa da tela de busca é possível combinar termos de busca, utilizando “&” para separar os termos.

Ex.:

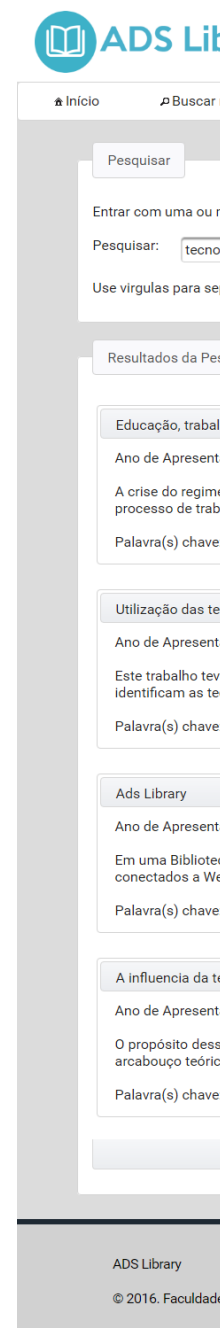
www.bibliotecadigital.com.br/pesquisar/key/matematica&formulas

A URL acima executa a busca de todas as monografias que possuem as palavras chave “matemática” e “formulas”. Obs.: *O domínio utilizado nos exemplos é fictício e utilizado para fins de exemplo.*

Segue abaixo todas opções de busca:

- .../pesquisar/**autor**/maria – Busca por autores
- .../pesquisar/**orientador**/denilce – Busca por orientadores
- .../pesquisar/**curso**/matematica – Busca por curso
- .../pesquisar/**instituição**/fatec – Busca por instituição
- .../pesquisar/**key**/tecnologia – Busca por palavra chave
- .../pesquisar/**titulo**/adslibrary – Busca por título
- .../pesquisar/**ano**/2016 – Busca por ano de apresentação
- .../pesquisar/**all**/tecnologia – Busca por todos os campos disponíveis

Figura 15 – Tela de busca de monografias com resultados

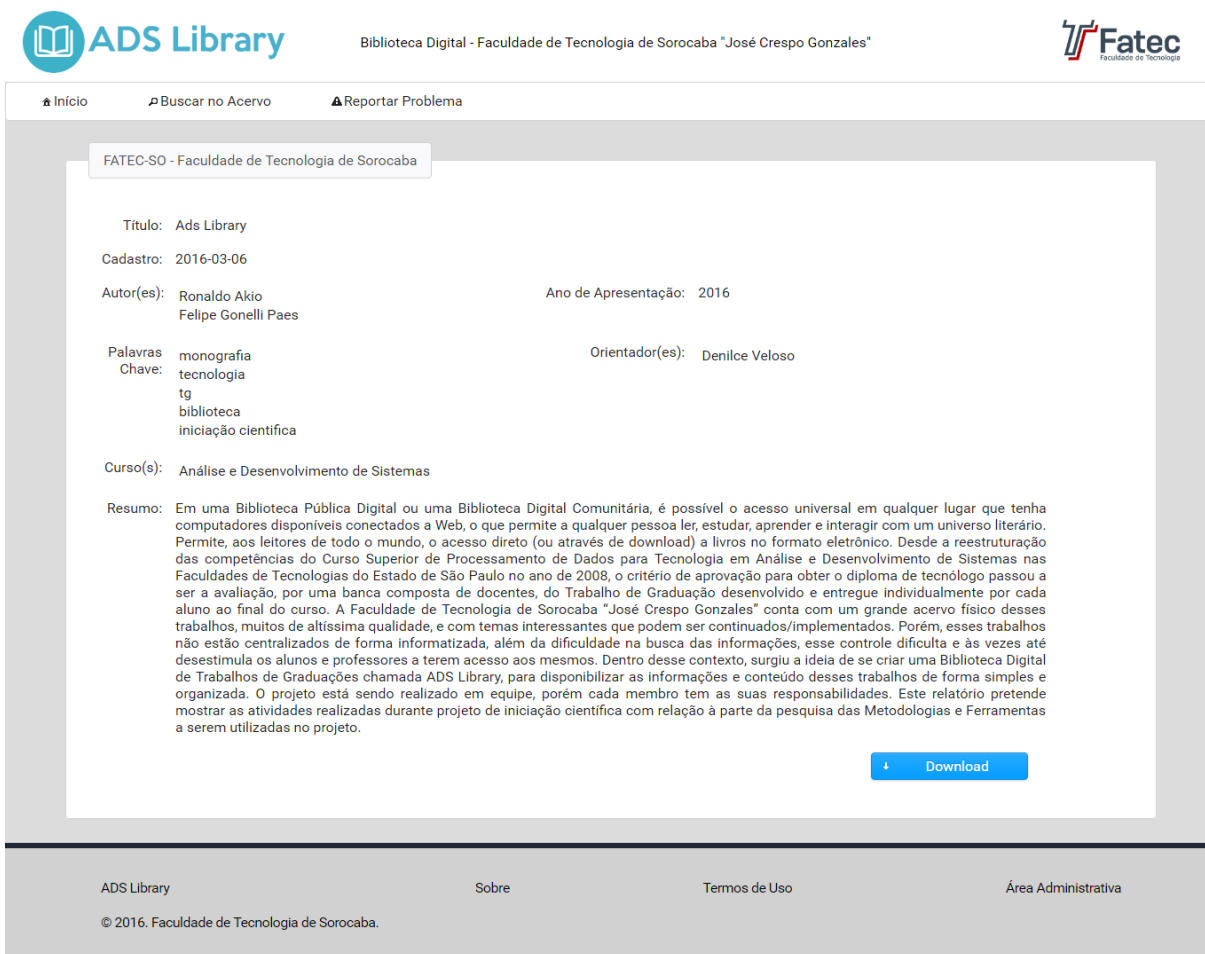


Fonte: Autor.

Assim que a busca retorna algum resultado, é possível visualizar as monografias encontradas (Figura 15), a partir desses resultados é possível visualizar detalhes da monografia selecionando o título ou pesquisar outras monografias a partir das palavras chave selecionando-as na tela.

7.2.5 Detalhar monografia

Figura 16 – Tela de detalhamento da monografia



The screenshot displays the ADS Library web interface. At the top, there are logos for ADS Library and Fatec, along with the text 'Biblioteca Digital - Faculdade de Tecnologia de Sorocaba "José Crespo Gonzales"'. Below the logos is a navigation bar with links: 'Início', 'Buscar no Acervo', and 'Reportar Problema'. The main content area shows a record for a monography titled 'Ads Library'. The record includes the following details:

- Título:** Ads Library
- Cadastro:** 2016-03-06
- Autor(es):** Ronaldo Akio, Felipe Gonelli Paes
- Ano de Apresentação:** 2016
- Palavras Chave:** monografia, tecnologia, tg, biblioteca, iniciação científica
- Orientador(es):** Denilce Veloso
- Curso(s):** Análise e Desenvolvimento de Sistemas

The **Resumo:** section contains a detailed paragraph about the project, stating that it aims to provide universal access to digital resources through a web-based library. It mentions the project's origin in 2008 and its goal of organizing and making available a large collection of student and faculty works. At the bottom right of the record, there is a blue button labeled 'Download'.



The footer of the interface includes the text 'ADS Library', 'Sobre', 'Termos de Uso', 'Área Administrativa', and '© 2016. Faculdade de Tecnologia de Sorocaba.'

Fonte: Autor.

A tela mostrada na figura 16 exibe todas as informações de uma monografia selecionada como ano de apresentação, data de cadastro, instituição, curso, orientadores, autores, palavras chave e resumo, além de disponibilizar a monografia em formato digital pela opção de download.

Caso um usuário esteja logado na aplicação será exibido um botão para editar a monografia (Figura 17).

Figura 17 – Tela de detalhamento da monografia – Usuário logado


Biblioteca Digital - Faculdade de Tecnologia de Sorocaba "José Crespo Gonzales"


Bem vindo, teste

[Início](#)
[Buscar no Acervo](#)
[Cadastrar](#)
[Consultar](#)
[Relatórios](#)
[Reportar Problema](#)
[Logout](#)

FATEC-SO - Faculdade de Tecnologia de Sorocaba

Título: Ads Library

Cadastro: 2016-03-06

Autor(es): Ronaldo Akio
Felipe Gonelli Paes

Palavras Chave: monografia
tecnologia
tg
biblioteca
iniciação científica

Curso(s): Análise e Desenvolvimento de Sistemas

Resumo: Em uma Biblioteca Pública Digital ou uma Biblioteca Digital Comunitária, é possível o acesso universal em qualquer lugar que tenha computadores disponíveis conectados a Web, o que permite a qualquer pessoa ler, estudar, aprender e interagir com um universo literário. Permite, aos leitores de todo o mundo, o acesso direto (ou através de download) a livros no formato eletrônico. Desde a reestruturação das competências do Curso Superior de Processamento de Dados para Tecnologia em Análise e Desenvolvimento de Sistemas nas Faculdades de Tecnologias do Estado de São Paulo no ano de 2008, o critério de aprovação para obter o diploma de tecnólogo passou a ser a avaliação, por uma banca composta de docentes, do Trabalho de Graduação desenvolvido e entregue individualmente por cada aluno ao final do curso. A Faculdade de Tecnologia de Sorocaba "José Crespo Gonzales" conta com um grande acervo físico desses trabalhos, muitos de altíssima qualidade, e com temas interessantes que podem ser continuados/implementados. Porém, esses trabalhos não estão centralizados de forma informatizada, além da dificuldade na busca das informações, esse controle dificulta e às vezes até desestimula os alunos e professores a terem acesso aos mesmos. Dentro desse contexto, surgiu a ideia de se criar uma Biblioteca Digital de Trabalhos de Graduações chamada ADS Library, para disponibilizar as informações e conteúdo desses trabalhos de forma simples e organizada. O projeto está sendo realizado em equipe, porém cada membro tem as suas responsabilidades. Este relatório pretende mostrar as atividades realizadas durante projeto de iniciação científica com relação à parte da pesquisa das Metodologias e Ferramentas a serem utilizadas no projeto.

[Editar](#)
[Download](#)

ADS Library

Sobre

Termos de Uso

Área Administrativa

© 2016. Faculdade de Tecnologia de Sorocaba.

Fonte: Autor.

7.2.6 Login

Figura 18 – Tela de login

ADS Library

Biblioteca Digital - Faculdade de Tecnologia de Sorocaba "José Crespo Gonzales"

Fatec
Faculdade de Tecnologia

[Início](#) [Buscar no Acervo](#) [Reportar Problema](#)

Login:

Senha:

Login

ADS Library

© 2016, Faculdade de Tecnologia de Sorocaba.

[Sobre](#) [Termos de Uso](#) [Área Administrativa](#)

Fonte: Autor.

A tela de login mostrada na figura 18 é responsável por limitar o acesso da aplicação para páginas que precisam de privilégios, para realizar o login é preciso um usuário e senha cadastrados, onde qualquer tentativa de acesso a uma página deste grupo o acesso é bloqueado.

7.2.7 Funcionários

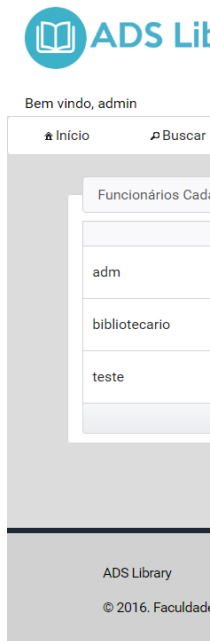


Figura 19 – Tela de registro de funcionário

Fonte: Autor.

A tela de registro de funcionários (Figura 19) permite o cadastro de um novo funcionário. O acesso a tela é permitido somente para funcionários cadastrados que possuam o nível administrador.

Figura 20 – Tela de consulta de funcionário



Fonte: Autor.

A tela de consulta de funcionários mostrada na figura 20 permite a consulta dos funcionários cadastrados, edição, exclusão e alteração de senha. O acesso ao sistema é permitido somente para funcionários cadastrados, o mesmo possui nível de acesso e conforme seu nível suas funções são limitadas. Para um usuário administrador é possível alterar a senha de qualquer funcionário e um usuário bibliotecário somente a sua própria.

7.2.8 Monografia



Figura 21 – Tela de registro de monografia

Fonte: Autor.

Sendo o principal registro do sistema, a tela de registro de monografias exibida na figura 21 realiza o cadastro de uma nova monografia no sistema.

O cadastro de monografia é diferenciado pois possui algumas considerações:

- A instituição e seus respectivos cursos e os orientadores devem ser previamente cadastrados. Seus cadastros são realizados em seus formulários específicos, que podem ser acessados no menu de cadastro.

- Uma monografia pode possuir mais de um orientador e mais um curso da mesma instituição.
- As palavras chave e autores não possuem cadastro prévios, cumprindo o princípio de que não existe uma palavra chave e um autor no sistema se uma monografia relacionada a eles também não existir. Essa restrição evita o cadastro desnecessário de palavras chaves e autores que não correspondam a qualquer trabalho de graduação.
- Os campos de adição de palavra chave e autor, sugerem os registros já existentes no banco de dados, para evitar registros duplicados.
- A partir do momento em que uma monografia é cadastrada a mesma se encontrará disponível para visualização no sistema.
- O upload da monografia deve ser feito com um arquivo em formato PDF.

Figura 22 – Tela de consulta de monografia



The screenshot displays the ADS Library interface. At the top, there are logos for ADS Library, the Digital Library of the Faculty of Technology of Sorocaba, and Fatec. A navigation bar includes links for Inicio, Buscar no Acervo, Cadastrar, Consultar, Relatórios, Reportar Problema, and Logout. The main content area, titled 'Monografias Cadastradas', contains a table with the following data:



Id	Título	Ano de Apresentação	Data de Cadastro	Arquivo	Funcionário
dGVzdGUJU2F0IE1hciAx	O conhecimento matematico e o uso de jogos na sala de aula	2000	2016-03-19	dGVzdGUJU2F0IE1hciAx.pdf	admin
dGVzdGUJV2VklE1hciAz	Educação, trabalho e tecnologia	1992	2016-03-30	dGVzdGUJV2VklE1hciAz.pdf	
dGVzdGVAMT	Utilização das tecnologias da informação e da comunicação, na pratica docente, numa instituição de ensino tecnologico	2004	2016-03-17	dGVzdGVAMT.xlsx	admin
Q29tcGFyYe	Comparação Analítica em Bancos de Dados	2015	2016-03-06	Q29tcGFyYe.pdf	admin
QWRzIEpYn	Ads Library	2016	2016-03-06	QWRzIEpYn.docx	

Below the table is a pagination control showing '1 - 5 de 9' and navigation buttons. The footer contains links for ADS Library, Sobre, Termos de Uso, and Área Administrativa, along with the copyright notice '© 2016. Faculdade de Tecnologia de Sorocaba.'

Fonte: Autor.

A tela de consulta de monografias mostrada na figura 22 exibi as monografias cadastradas no sistema, são exibidas informações da monografia, arquivo digital e do usuário que efetuou o cadastro.

Figura 23 – Tela de edição de monografia


Biblioteca Digital - Faculdade de Tecnologia de Sorocaba "José Crespo Gonzales"


Bem vindo, teste

[Início](#)
[Buscar no Acervo](#)
[Cadastrar](#)
[Consultar](#)
[Relatórios](#)
[Reportar Problema](#)
[Logout](#)

Editar Monografia - Ads Library

Título:


Ano de Apresentação:

Resumo

Em uma Biblioteca Pública Digital ou uma Biblioteca Digital Comunitária, é possível o acesso universal em qualquer lugar que tenha computadores disponíveis conectados a Web, o que permite a qualquer pessoa ler, estudar, aprender e interagir com um universo literário. Permite, aos leitores de todo o mundo, o acesso direto (ou através de download) a livros no formato eletrônico. Desde a reestruturação das competências do Curso Superior de Processamento de Dados para Tecnologia em Análise e Desenvolvimento de Sistemas nas Faculdades de Tecnologias do Estado de São Paulo no ano de 2008, o critério de aprovação para obter o diploma de tecnólogo passou a ser a avaliação, por uma banca composta de docentes, do Trabalho de Graduação desenvolvido e entregue individualmente por cada aluno ao final do curso. A Faculdade de Tecnologia de Sorocaba "José Crespo Gonzales" conta com um grande acervo físico desses trabalhos, muitos de altíssima qualidade, e com temas interessantes que podem ser continuados/implementados. Porém, esses trabalhos não estão centralizados de forma informatizada, além da dificuldade na busca das informações, esse controle dificulta e às vezes até desestimula os alunos e professores a terem acesso aos mesmos. Dentro desse contexto, surgiu a ideia de se criar uma Biblioteca Digital de Trabalhos de Graduações chamada ADS Library, para disponibilizar as informações e conteúdo desses trabalhos de forma simples e organizada. O projeto está sendo realizado em equipe, porém cada membro tem as suas responsabilidades. Este relatório pretende mostrar as atividades realizadas durante projeto de iniciação científica com relação à parte da pesquisa das Metodologias e Ferramentas a serem utilizadas no projeto.

1265/3000

Arquivo: [+ Selecionar arquivo](#)

QWRZlExpYn.docx 

Instituição:

Curso(s):

☐ Eletrônica Automotiva
☐ Logística
☐ Processos Metalúrgicos
☐ Projetos Mecânicos
☐ Sistemas Biomédicos
☒ Análise e Desenvolvimento de Sistemas

Orientador(es):

☒ Denilce Veloso
☐ Antonio Cesar de Barros Munari
☐ Maria Angélica Cardieri
☐ Jefferson Blaitt
☐ José Roberto Bordieri
☐ Maria José Cardozo
☐ Glaucio Todesco
☐ Daniela Arruda

Palavras-Chave:

[Adicionar](#)

Palavra	Opção
monografia	Remover
tecnologia	Remover
tg	Remover
biblioteca	Remover
iniciação científica	Remover

RA Autor:

Nome Autor:

E-mail Autor:

[Adicionar](#)

RA	Nome	Opção
0030481321039	Ronaldo Akio	Remover
0030481321016	Felipe Gonelli Paes	Remover

[Salvar](#)

ADS Library

Sobre

Termos de Uso

Área Administrativa

© 2016. Faculdade de Tecnologia de Sorocaba.

Fonte: Autor.

A tela de edição (Figura 23) é acessada pelo detalhamento da monografia e ela exibi os dados da monografia selecionada e são obedecidas as mesmas regras

para o cadastro da monografia, ao realizar as modificações necessárias os dados são salvos e as alterações são visualizadas imediatamente.

7.2.9 Autores



Figura 24 – Tela de consulta de autores



Fonte: Autor.

A tela de autor (Figura 24) exibe os autores cadastrados no sistema, dando a possibilidade de edição e exclusão de dados, sendo que um cadastro poderá ser excluído somente se o mesmo não estiver relacionado a uma monografia.

7.2.10 Instituições

Figura 25 – Tela de registro de instituição

Biblioteca Digital - Faculdade de Tecnologia de Sorocaba "José Crespo Gonzales"

Bem vindo, admin

[Início](#)
[Buscar no Acervo](#)
[Cadastrar](#)
[Consultar](#)
[Relatórios](#)
[Reportar Problema](#)
[Logout](#)

Instituições Cadastradas

Sigla	Nome	Cidade	Editar	
FATEC-SO	Faculdade de Tecnologia de Sorocaba	Sorocaba		
FATEC-SP	Faculdade de Tecnologia de São Paulo	São Paulo		
fdffdfdsfsjshdfjh	bbbbbb	aaaaaaaaaaaa		
UNICAMP	Universidade Estadual de Campinas	Campinas, SP		
UNESP	Universidade Estadual Paulista	Sorocaba, SP		

1 - 5 de 5

ADS Library

Sobre

Termos de Uso

Área Administrativa

© 2016, Faculdade de Tecnologia de Sorocaba.

Fonte: Autor.

A tela de registro de instituição (Figura 25) disponibiliza o cadastro de uma nova instituição de ensino.

Figura 26 – Tela de registro de instituição

Bem vindo, admin

[Início](#)
[Buscar no Acervo](#)
[Cadastrar](#)
[Consultar](#)
[Relatórios](#)
[Reportar Problema](#)
[Logout](#)

Instituições Cadastradas

Sigla	Nome	Cidade	Editar	
FATEC-SO	Faculdade de Tecnologia de Sorocaba	Sorocaba		
FATEC-SP	Faculdade de Tecnologia de São Paulo	São Paulo		
fdffdfdsfsjshdfjh	bbbbbb	aaaaaaaaaaaa		
UNICAMP	Universidade Estadual de Campinas	Campinas, SP		
UNESP	Universidade Estadual Paulista	Sorocaba, SP		

1 - 5 de 5 1 5

ADS Library Sobre Termos de Uso Área Administrativa
 © 2016. Faculdade de Tecnologia de Sorocaba.

Fonte: Autor.

A tela de consulta de instituições (Figura 26) exibe as instituições já cadastradas no sistema e possibilita a edição e exclusão, respeitando as regras de integridade, no caso de uma instituição estar relacionada a uma monografia ou curso.

7.2.11 Cursos

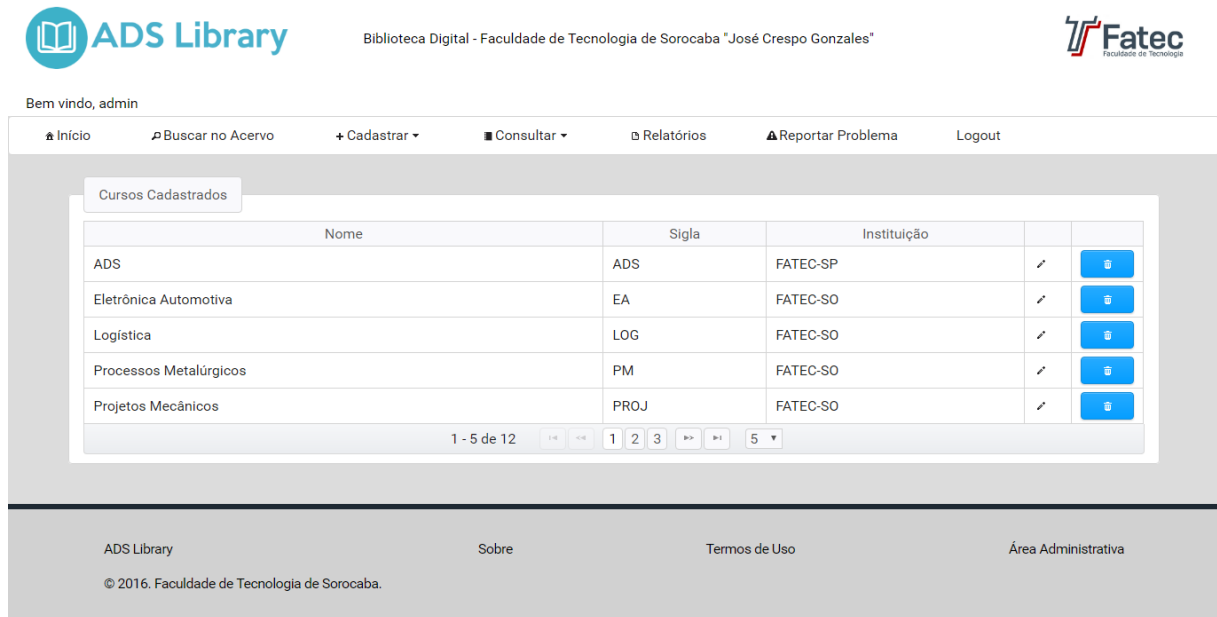
Figura 27 – Tela de registro de cursos

The screenshot displays the 'ADS Library' web interface. At the top left is the 'ADS Library' logo, and at the top right is the 'Fatec' logo with the text 'Faculdade de Tecnologia'. Below the logos, the text 'Biblioteca Digital - Faculdade de Tecnologia de Sorocaba "José Crespo Gonzales"' is visible. A navigation bar contains links: 'Início', 'Buscar no Acervo', 'Cadastrar', 'Consultar', 'Relatórios', 'Reportar Problema', and 'Logout'. The main content area is titled 'Cadastro de Curso' and contains a form with three input fields: 'Sigla:', 'Nome:', and 'Instituição:'. The 'Instituição:' field is a dropdown menu with the text 'Selecione uma Instituição'. Below these fields is a blue 'Cadastrar' button. The footer of the page includes the text 'ADS Library', 'Sobre', 'Termos de Uso', 'Área Administrativa', and '© 2016, Faculdade de Tecnologia de Sorocaba.'

Fonte: Autor.

A tela de registro de curso mostrada na figura 27 disponibiliza a possibilidade de cadastro de novos cursos no sistema, para realização do cadastro a instituição deve ser previamente cadastrada.

Figura 28 – Tela de consulta de cursos



Bem vindo, admin

[Início](#)
[Buscar no Acervo](#)
[Cadastrar](#)
[Consultar](#)
[Relatórios](#)
[Reportar Problema](#)
[Logout](#)

Cursos Cadastrados

Nome	Sigla	Instituição		
ADS	ADS	FATEC-SP		
Eletrônica Automotiva	EA	FATEC-SO		
Logística	LOG	FATEC-SO		
Processos Metalúrgicos	PM	FATEC-SO		
Projetos Mecânicos	PROJ	FATEC-SO		

1 - 5 de 12 1 2 3 5

ADS Library Sobre Termos de Uso Área Administrativa
 © 2016. Faculdade de Tecnologia de Sorocaba.

Fonte: Autor.

A tela de consulta de cursos (Figura 28) exhibe os cursos já cadastrados e permite a edição e exclusão do registro, obedecendo as regras de integridade já mencionadas.

7.2.12 Orientadores

Figura 29 – Tela de registro de orientadores

The screenshot shows the 'ADS Library' web application interface. At the top left is the 'ADS Library' logo. To its right is the text 'Biblioteca Digital - Faculdade de Tecnologia de Sorocaba "José Crespo Gonzales"'. At the top right is the 'Fatec' logo. Below the header, a navigation bar contains links: 'Início', 'Buscar no Acervo', 'Cadastrar', 'Consultar', 'Relatórios', 'Reportar Problema', and 'Logout'. The main content area is titled 'Cadastro de Orientador' and contains a form with two input fields: 'Nome:' and 'E-mail:'. Below these fields is a blue 'Cadastrar' button. The footer of the page includes the text 'ADS Library', 'Sobre', 'Termos de Uso', 'Área Administrativa', and a copyright notice '© 2016, Faculdade de Tecnologia de Sorocaba.'

Fonte: Autor.

A tela de registro de orientadores (Figura 29) possibilita o cadastro de novos orientadores no sistema.

Figura 30 – Tela de consulta de orientadores

Bem vindo, admin

[Início](#)
[Buscar no Acervo](#)
[+ Cadastrar](#)
[■ Consultar](#)
[Relatórios](#)
[▲ Reportar Problema](#)
[Logout](#)

Orientadores Cadastrados

Nome	E-mail		
aaateste	testehjk		
Ana Carolina	anamaria@email.com		
Antonio Cesar de Barros Munari	munari@fatec.com		
Daniela Arruda	daniela@email.com		
Denilce Veloso	denilce@fatec.com		

1 - 5 de 22 1 2 3 4 5 5 ▼

ADS Library
 Sobre
 Termos de Uso
 Área Administrativa

© 2016. Faculdade de Tecnologia de Sorocaba.

Fonte: Autor.

A tela de consulta (Figura 30) exibe os orientadores já cadastrados e disponibiliza a função de edição e exclusão, no caso da exclusão, obedecendo as regras de integridade.

7.2.13 Palavras-Chave



Figura 31 – Tela de palavras chave

Fonte: Autor.

A tela de palavras chave mostra da figura 31 exibi todas as palavras cadastradas nas monografias para simples conferência, também disponibiliza a opção de exclusão, porém a exclusão só é permitida para palavras chave que não possuam relação com qualquer monografia.

7.2.14 Reportar



Figura 32 – Tela de reportar problemas ou sugestões

Fonte: Autor.

A tela de reportar exibida na figura 32 disponibiliza ao usuário a opção de reportar algumas situações como: erros, melhorias, adição de recursos, arquivos corrompidos, cadastros duplicados, cadastros inconsistentes, etc.

7.2.15 Relatórios

Figura 33 – Tela de geração de relatórios

The screenshot shows the 'ADS Library' web application interface. At the top, there is a header with the 'ADS Library' logo, the text 'Biblioteca Digital - Faculdade de Tecnologia de Sorocaba "José Crespo Gonzales"', and the 'Fatec' logo. Below the header, a navigation bar contains links: 'Início', 'Buscar no Acervo', 'Cadastrar', 'Consultar', 'Relatórios', 'Reportar Problema', and 'Logout'. The main content area is titled 'Relatório' and features a form for generating reports. The form includes two date input fields labeled 'Período de' and 'até', both set to '12/06/2016'. There is a dropdown menu for 'Categoria' set to 'Monografia' and a blue 'Gerar' button. The footer of the page contains the text 'ADS Library', 'Sobre', 'Termos de Uso', 'Área Administrativa', and a copyright notice '© 2016. Faculdade de Tecnologia de Sorocaba.'

Fonte: Autor.

A tela de relatórios (Figura 33) possibilita a geração de documentos com alguns dados do sistema, a partir de um filtro são disponibilizados relatório das monografias cadastradas (inclui estatísticas), dos problemas/sugestões reportados e o log de acesso ao sistema. Para a geração de um relatório é necessário selecionar o período dos dados desejados e a seleção do tipo de relatório (Os logs de login de usuários são disponibilizados apenas para usuários administradores). Os documentos são disponibilizados em formato PDF (.pdf) e Excel (.xls).

Figura 34 – Tela de geração de relatórios – Relatório gerado

Bem vindo, admin

[Início](#)
[Buscar no Acervo](#)
[Cadastrar](#)
[Consultar](#)
[Relatórios](#)
[Reportar Problema](#)
[Logout](#)

Relatório

Período de: 01/04/2016 até: 12/06/2016 Categoria: Monografia [Gerar](#)

Título	Ano	Data	N° Downloads	Autores	Palavras Chave	Orientadores	Cursos
A influencia da tecnologia no planejamento estrategico	2003	22/05/2016	1	Lucas Buga Rodrigo Rodrigues	teste2 asd	Denilce Veloso	Eletrônica Automotiva
Um modelo de sistema de gestão da segurança da informação baseado nas normas ABNT NBR ISO/IEC 27001:2006, 27002:2005 e 27005:2008	2012	24/05/2016	1	Valdeci Otacilio dos Santos	informação tecnologia da informação - segurança redes de informação serviços de informação	Renato Baldini Filho	Faculdade de Engenharia Elétrica e de Computação

[Exportar para Excel](#)
[Exportar para PDF](#)

ADS Library
 Sobre
 Termos de Uso
 Área Administrativa

© 2016, Faculdade de Tecnologia de Sorocaba.

Fonte: Autor.

Após a geração de um relatório (Figura 34) os formatos de arquivos são disponibilizados para download.

8 CODIFICAÇÃO

A codificação envolve toda a parte de código da aplicação, desde o banco de dados até a criação de telas, bibliotecas e dependências do projeto. Também será tratado questões para execução do mesmo, ambiente e configurações.

8.1 Script

Segue abaixo o script de criação das tabelas, entretanto o mesmo não foi elaborado, pois o mesmo é gerado pelo framework de persistência Hibernate, o Hibernate se encarrega de mapear as classes Java para tabelas do banco de dados. De acordo com a estrutura estabelecida pela camada de Modelagem da aplicação o Hibernate se encarrega de gerar este script.

```
CREATE DATABASE IF NOT EXISTS `adslibrary`;
USE `adslibrary`;
--
-- Table structure for table `autor`
--
DROP TABLE IF EXISTS `autor`;
CREATE TABLE `autor` (
  `RA` varchar(255) NOT NULL,
  `email` varchar(100) NOT NULL,
  `nome` varchar(250) NOT NULL,
  PRIMARY KEY (`RA`)
);
--
-- Table structure for table `curso`
--
CREATE TABLE `curso` (
  `id` int(11) NOT NULL,
  `nome` varchar(200) DEFAULT NULL,
  `sigla` varchar(50) NOT NULL,
  `instituicao_id` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `FKb92pc49po18w3j0ukh0uim3c9` (`instituicao_id`),
  CONSTRAINT `FKb92pc49po18w3j0ukh0uim3c9` FOREIGN KEY (`instituicao_id`)
REFERENCES `instituicao` (`id`)
);
--
-- Table structure for table `funcionario`
--
CREATE TABLE `funcionario` (
  `login` varchar(250) NOT NULL,
  `senha` longtext,
  `tipo` int(11) DEFAULT NULL,
  PRIMARY KEY (`login`)
);
--
-- Table structure for table `hibernate_sequence`
--
```

```

CREATE TABLE `hibernate_sequence` (
  `next_val` bigint(20) DEFAULT NULL
);
--
-- Table structure for table `instituicao`
--
CREATE TABLE `instituicao` (
  `id` int(11) NOT NULL,
  `cidade` varchar(150) DEFAULT NULL,
  `nome` varchar(250) DEFAULT NULL,
  `sigla` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`id`)
);
--
-- Table structure for table `monografia`
--
CREATE TABLE `monografia` (
  `id` varchar(255) NOT NULL,
  `NomeArquivo` varchar(250) NOT NULL,
  `anoApresentacao` int(11) NOT NULL,
  `dataCadastro` date DEFAULT NULL,
  `numDownloads` int(11) DEFAULT NULL,
  `titulo` varchar(250) NOT NULL,
  `funcionario_login` varchar(250) DEFAULT NULL,
  `instituicao_id` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `FK2nqhv115ovjssdn92n5ed36jo` (`funcionario_login`),
  KEY `FK6dnycuwctea20sfpyl4fdvn6` (`instituicao_id`),
  CONSTRAINT `FK2nqhv115ovjssdn92n5ed36jo` FOREIGN KEY (`funcionario_login`)
REFERENCES `funcionario` (`login`),
  CONSTRAINT `FK6dnycuwctea20sfpyl4fdvn6` FOREIGN KEY (`instituicao_id`)
REFERENCES `instituicao` (`id`)
);
--
-- Table structure for table `monografia_autor`
--
CREATE TABLE `monografia_autor` (
  `Monografia_id` varchar(255) NOT NULL,
  `autores_RA` varchar(255) NOT NULL,
  KEY `FKbowdxdgjm04ciamxu85tahv0r` (`autores_RA`),
  KEY `FKaxtth5s2rgfpxe3dhqchvmfbc` (`Monografia_id`),
  CONSTRAINT `FKaxtth5s2rgfpxe3dhqchvmfbc` FOREIGN KEY (`Monografia_id`)
REFERENCES `monografia` (`id`),
  CONSTRAINT `FKbowdxdgjm04ciamxu85tahv0r` FOREIGN KEY (`autores_RA`) REFERENCES
`autor` (`RA`)
);
--
-- Table structure for table `monografia_curso`
--
CREATE TABLE `monografia_curso` (
  `Monografia_id` varchar(255) NOT NULL,
  `cursos_id` int(11) NOT NULL,
  KEY `FKgvg61x8cd9fpk3mm0eqmukmkd` (`cursos_id`),
  KEY `FK2pl6e3dfd62m2dyat7mtqqlr0` (`Monografia_id`),
  CONSTRAINT `FK2pl6e3dfd62m2dyat7mtqqlr0` FOREIGN KEY (`Monografia_id`)
REFERENCES `monografia` (`id`),
  CONSTRAINT `FKgvg61x8cd9fpk3mm0eqmukmkd` FOREIGN KEY (`cursos_id`) REFERENCES
`curso` (`id`)
);

```

```
--
-- Table structure for table `monografia_orientador`
--
CREATE TABLE `monografia_orientador` (
  `Monografia_id` varchar(255) NOT NULL,
  `orientadores_id` int(11) NOT NULL,
  KEY `FK4oispas3sw3dvc702aqwihmuwh` (`orientadores_id`),
  KEY `FKsj7nb5ynn8vi940toa9qysg0p` (`Monografia_id`),
  CONSTRAINT `FK4oispas3sw3dvc702aqwihmuwh` FOREIGN KEY (`orientadores_id`)
REFERENCES `orientador` (`id`),
  CONSTRAINT `FKsj7nb5ynn8vi940toa9qysg0p` FOREIGN KEY (`Monografia_id`)
REFERENCES `monografia` (`id`)
);
--
-- Table structure for table `monografia_palavrachave`
--
CREATE TABLE `monografia_palavrachave` (
  `Monografia_id` varchar(255) NOT NULL,
  `palavrasChave_palavraChave` varchar(250) NOT NULL,
  KEY `FKboaobu7mvjdubwdv0hhyy2py5` (`palavrasChave_palavraChave`),
  KEY `FKmgksys84k51bvqc43m3myyb` (`Monografia_id`),
  CONSTRAINT `FKboaobu7mvjdubwdv0hhyy2py5` FOREIGN KEY
(`palavrasChave_palavraChave`) REFERENCES `palavrachave` (`palavraChave`),
  CONSTRAINT `FKmgksys84k51bvqc43m3myyb` FOREIGN KEY (`Monografia_id`) REFERENCES
`monografia` (`id`)
);
--
-- Table structure for table `orientador`
--
CREATE TABLE `orientador` (
  `id` int(11) NOT NULL,
  `email` varchar(100) DEFAULT NULL,
  `nome` varchar(150) DEFAULT NULL,
  PRIMARY KEY (`id`)
);
--
-- Table structure for table `palavrachave`
--
CREATE TABLE `palavrachave` (
  `palavraChave` varchar(250) NOT NULL,
  PRIMARY KEY (`palavraChave`)
);
--
-- Table structure for table `relato`
--
CREATE TABLE `relato` (
  `id` int(11) NOT NULL,
  `assunto` varchar(255) NOT NULL,
  `dataHora` datetime DEFAULT NULL,
  `descricao` longtext NOT NULL,
  `email` varchar(100) NOT NULL,
  PRIMARY KEY (`id`)
);
--
-- Table structure for table `usuario`
--
CREATE TABLE `usuario` (
  `id` int(11) NOT NULL,
  `dataHora` datetime DEFAULT NULL,
```

```
`login` varchar(250) DEFAULT NULL,  
`status` int(11) DEFAULT NULL,  
PRIMARY KEY (`id`)  
);
```

8.2 Código Fonte

O código fonte do projeto será disponibilizado através de uma mídia física e entregue na instituição de ensino. Será composto por um projeto Java EE, com integrado com Maven, onde inclui todos os fragmentos e dependência da aplicação.

8.3 Instalação

Uns dos objetivos deste projeto é que o mesmo seja utilizado pelos membros da instituição, para isso a aplicação deve ser disponibilizada em um servidor e ser de fácil acesso a todos os alunos e professores da instituição.

A aplicação é executada em um servidor de aplicações com suporte a Java. O escolhido foi o Tomcat, por meio do servidor é possível disponibilizar a aplicação por meio de um endereço de rede. Para a instalação é necessário um servidor Linux ou Windows configurado que possua:

- Tomcat 8 ou superior
- Java Runtime Edition 7 ou superior
- MySQL 5 ou superior.

Uma vez que a aplicação esteja configurada e disponível no servidor, o primeiro passo é possuir um cadastro de funcionário para conseguir acesso ao sistema, tendo esse acesso já será possível utilizar todas as funcionalidades da aplicação.

8.4 Trabalhos Futuros

Primeiramente deve-se salientar que este projeto está em sua primeira versão, desde que sua primeira versão executável foi criada, o mesmo não passou por testes de usuário reais. Em relação a requisitos funcionais, existem alguns pontos que podem ser melhorados como:

- Envio de e-mail de confirmação de comentário recebido, assim que um usuário enviar seu comentário na tela de reportar.

- Realizar estudo de informações relevantes para o usuário de manutenção do sistema e a partir disso melhorar a área administrativa e a geração de relatórios.
- Melhorias de performance.
- Melhorias de interface: Realizar estudo de interação com o usuário e avaliar se a interface é de fácil utilização e se o sistema possui uma pequena curva de aprendizado, além de levantar pontos fracos da interface a serem melhorados.

A partir do momento em que o sistema for implantado deve se considerar os problemas e sugestões enviados pelos usuários para corrigir no sistema. Uma das ideias principais é que esse projeto esteja em constante melhoria e seu acervo fique cada vez mais completo, portanto o servidor em que estiver hospedado, tem de estar em funcionamento e a capacidade de armazenamento deve ser constante monitorada.

CONCLUSÃO

Acredita que a aplicação poderá ser de grande ajuda. Mediante restrições de acesso dos perfis via *login*, o funcionário/bibliotecário poderá cadastrar as informações dos trabalhos de graduação como autor, título, resumo, ano de apresentação, palavras-chave, orientador. Os visitantes da aplicação poderão realizar pesquisas simples ou avançadas, combinando campos conforme sua necessidade, podendo visualizar mais detalhes e fazer *download* do trabalho de graduação buscado.

A literatura utilizada foi muito importante para a pesquisa, principalmente com relação a parte de UML. A *web* também tem se apresentado como uma grande aliada para pesquisa das ferramentas que foram utilizadas no desenvolvimento do projeto, algumas destas ferramentas foram testadas e descartadas por motivos de incompatibilidade e problemas de performance no desenvolvimento. A escolha das ferramentas tem se voltado no sentido de facilitar o trabalho de desenvolvimento, aliando-se ao melhor resultado.

Essa escolha vem consolidando sua importância nos projetos de tecnologia da informação, onde as ferramentas e tecnologias são grandes pontos de influência dentro de um projeto, saber evidenciar seus aspectos bons e ruins e aproveitar todos seus recursos torna-se muito válido. Com a definição de uma boa estrutura do projeto baseado em tecnologias confiáveis, a etapa de desenvolvimento consegue um grande aproveitamento, fazendo que o projeto em sua totalidade seja concluído com sucesso.

REFERÊNCIAS

BOOCH, G. **Análisis y diseño orientado a objetos con aplicaciones**. Addison-Wesley, 1996.

_____; RUMBAUGH, J.; JACOBSON, I. **UML – Guia do usuário**. São Paulo: Editora Campus, São Paulo, 2000.

CAELUM. **Apostila Java com Testes, JSF e Design Patterns**. Curso Fj-22. Disponível em: <<https://www.caelum.com.br/apostila-java-testes-jsf-web-services-design-patterns>> Acesso em: 20.FEV.2016.

_____. **Apostila Java para Desenvolvimento Web**. Curso Fj-21. Disponível em: <<https://www.caelum.com.br/apostila-java-web>> Acesso em: 20.FEV.2016.

DEV MEDIA. **Metodologia de Desenvolvimento de Software**. Disponível em: <<http://www.devmedia.com.br/metodologia-de-desenvolvimento-desoftware/1903>> Acesso: 31.JUL.2015.

ECLIPSE, I.D.E. Disponível em: <<http://www.eclipse.org>>. Acesso: 26.FEV.2016.

GIT. Scm. Disponível em: <<https://git-scm.com>>. Acesso em: 22.FEV.2016.

HIBERNATE. Disponível em: <<http://hibernate.org>> Acesso: 20.JUL.2015.

HOUAISS, Antônio. **Dicionário Houaiss da Língua Portuguesa**. Rio de Janeiro: Objetiva, 2001.

ISAÍAS, P. **Bibliotecas Digitais**. Lisboa: Universidade Aberta. 1999.

JACOBSON, I. et al. **Object-oriented software engineering: a use case driven approach**. Reading: Addison-Wesley, 1992.

_____; GRISS, M.; JONSSON, P. **Software reuse: architecture, process and organization for business success**. Addison-Wesley, New York, 2001. 53 Addison-Wesley, New York, 2001.

JAVA. Disponível em: <<http://www.oracle.com/br/java/overview/index.html>> Acesso: 20.JUL.2015.

LUCKOW, Décio Heinzelmann; MELO, Alexandre Altair de. **Programação Java para Web**. 2ª Edição. Editora Novatec. 2015.

MARTIN, James; ODEL, James J. **Análise e Projeto Orientados a Objeto**. Makron Books, 1995.

MAVEN, Apache. Disponível em: <<http://maven.apache.org>>. Acesso: 22.FEV.2016.

MILANESI, Luís. **Biblioteca**. São Paulo: Ateliê Editorial, 2002.

MYSQL. Disponível em: <<http://www.mysql.com>> Acesso: 20.JUL.2015.

ORACLE. **Java Platform, Enterprise Edition 6**. Disponível em: <<http://docs.oracle.com/javase/6>> Acesso: 20.FEV.2016.

_____. **Java Platform, Enterprise Edition 7**. Disponível em: <<http://docs.oracle.com/javase/7>> Acesso: 20.FEV.2016.

PARREIRA JÚNIOR, Walteno Martins. **Apostila Engenharia de Software**. UEMG, Itumbiara-GO, 2010

PRESSMAN, R. S. **Engenharia de software**: uma abordagem profissional. 7. ed. Porto Alegre: AMGH, 2011. 780p.

PRIMEFACES. Disponível em: <<http://www.primefaces.org>> Acesso: 20.JUL.2015.

PROCÓPIO, Ednei. **Construindo uma biblioteca digital**. São Paulo: Edições Inteligentes, 2004. 109 p.

ROSETTO, Márcia; NOGUEIRA, Adriana Hypólito. **Aplicação de elementos metadados Dublin Core para a descrição de dados bibliográficos on-line da biblioteca digital de teses da USP**. In: SEMINÁRIO NACIONAL DE BIBLIOTECAS UNIVERSITÁRIAS, 12. 2002, Recife. Anais... Recife: UFPe, 2002. 1 CD-ROM.

RUMBAUGH, J. et al. **Modelagem e projetos baseados em objetos**. Rio de Janeiro: Editora Campus, 1994.

SPRING. Disponível em: <<http://www.devmedia.com.br/introducao-ao-springframework/26212>> Acesso: 20.JUL.2015.

TRELLO. Disponível em: <<https://trello.com>> Acesso: 20.JUL.2015.

VELOSO, Denilce de Almeida O. Veloso. **Projeto de Regime de Jornada Integral: "Integração entre Dispositivos Móveis e Web"**. Centro Paula Souza. 04/2014.

VERASZTO, Estéfano Vizconde et al. **Tecnologia: buscando uma definição para o conceito**. Revista Prisma.com, n. 7, 2010.

WEBSTERS, Ninth Neiv Collegiate Dictionary. 1998.

YAMASHITA, Ronaldo A. L. **ADS Library – Projeto de Biblioteca Digital de Trabalhos de Graduações: Análise de Requisitos**. Centro Paula Souza, 2016.

ZEROTURNAROUND. **Java Tools and Technologies Landscape for 2014**. Disponível em: <<http://zeroturnaround.com/rebellabs/java-tools-and-technologies-landscape-for-2014>>. Acesso: 22.FEV.2016.

GLOSSÁRIO

API – *Application Programming Interface*, do inglês interface de programação da aplicação, é um conjunto de funções e serviços que são disponibilizados para uso de outras aplicações externas, possibilita o fato de uma aplicação não se preocupar com as implementações feitas pela API, mas apenas utilizar os seus serviços.

Byte Code – Traduzido do inglês por código binário, é um código que possui um conjunto de combinações de zeros e uns.

Código fonte – Diz respeito a todo código de uma aplicação, onde a partir dele a mesma pode ser executada.

DAO – *Data Access Object*, do inglês objeto de acesso a dados, é um padrão de desenvolvimento para persistência de dados. Concentra a lógica respectiva ao banco de dados, como conexão e mapeamento de dados.

Framework – É uma estrutura de códigos e/ou *softwares* que juntos providenciam uma funcionalidade ou solução.

IDE – *Integrated Development Environment*, do inglês Ambiente de Desenvolvimento Integrado, é um software que possui um conjunto de recursos com o objetivo de auxiliar no processo de desenvolvimento.

Log – Dentro do contexto ao que se refere a “Log de dados”, é uma expressão utilizada para descrever o processo de registro de eventos relevantes num sistema computacional. Esse registro pode ser utilizado para restabelecer o estado original de um sistema ou para que um administrador conheça o seu comportamento no passado. Um arquivo de log pode ser utilizado para auditoria e diagnóstico de problemas em sistemas computacionais.

Open Source – Do inglês, Fonte Aberta, esse termo é utilizado para classificar os produtos de *software* (sendo programas prontos, códigos, *frameworks*) como livres para uso, *forks* e produção de códigos, ou *frameworks* que utilizem o tal produto para seu funcionamento ou o faça funcionar em outro sistema, desde que preservado os direitos autorais da fonte.

Repositório – É um local de armazenamento, que guarda, arquiva tudo que nele for colocado. Geralmente costuma armazenar todo o código e bibliotecas de um ou mais projetos.

Responsividade – Esse termo no âmbito de desenvolvimento web, ou mais conhecido como design responsivo é relacionado a adaptação, onde o desenvolvimento é estruturado para que o design da tela esteja preparado para qualquer resolução e/ou tamanho de tela, fazendo que a usabilidade da aplicação não seja prejudicada pelo dispositivo de uso.

Script – Texto contendo codificação.

SQL – *Structured Query Language*, do inglês linguagem de consulta estruturada, é linguagem base para os bancos de dados relacionais.

XML – *eXtensible Markup Language*, do inglês linguagem de marcação estendida, é uma linguagem recomendada pela W3C, utilizada na criação de documentos com dados hierarquicamente organizados, como textos, banco de dados ou desenhos vetoriais.