|  |  |
| --- | --- |
| Descrição: ssc.PNG | **UNIVERSIDADE DE SÃO PAULO**  **Instituto de Ciências Matemáticas e de Computação**  **Departamento de Sistemas de Computação** |

Um Sistema Web para o Museu da Fauna e Flora do ICMC/USP

***Paulo Sérgio Poli Junior***

***[Nome do Aluno]***

São Carlos - SP

Um Sistema Web para o Museu da Fauna

e Flora do ICMC/USP

***Paulo Sérgio Poli Junior***

##### Supervisor: Prof.ª Drª Elisa Yumi Nakagawa

|  |
| --- |
| Monografia referente ao estágio realizado na empresa **Instituto de Ciências Matemáticas e Computação - USP** dentro do escopo da disciplina **SSC0573 ­ Projeto Supervisionado ou de Graduação I** do Departamento de Sistemas de Computação do Instituto de Ciências Matemáticas e de Computação – ICMC-USP. |
| Área de Concentração: **Engenharia de Software** |

**USP – São Carlos**

**9 de Junho de 2014**

Resumo

O objetivo desta monografia é abordar os trabalhos desenvolvidos durante o período de estágio no ICMC – Instituto de Ciências Matemáticas e Computação, da Universidade de São Paulo, para o desenvolvimento de um sistema web, para o Museu da Fauna e Flora. As atividades realizadas incluem todo o desenvolvimento do software, desde a coleta de requisitos, modelagem do sistema, da interface do usuário, codificação e avaliação do sistema.

Sumário

[CAPÍTULO 1: INTRODUÇÃO 1](#_Toc356935316)

[1.1 A empresa onde o estágio foi realizado 1](#_Toc356935317)

[1.2 Objetivos do Trabalho 1](#_Toc356935318)

[1.3 Organização da Monografia 1](#_Toc356935319)

[CAPÍTULO 2: DESENVOLVIMENTO DO TRABALHO 2](#_Toc356935320)

[2.1. Descrição do Problema 2](#_Toc356935321)

[2.2 Métodos, Técnicas e Tecnologias Utilizadas 2](#_Toc356935322)

[2.3. Descrição das Atividades Realizadas 3](#_Toc356935323)

[2.4. Resultados Obtidos 3](#_Toc356935324)

[2.5. Dificuldades e Limitações 3](#_Toc356935325)

[CAPÍTULO 3: CONCLUSÃO 4](#_Toc356935326)

[3.1. Contribuições 4](#_Toc356935327)

[3.2. Relacionamento entre o Curso e o Projeto de Estágio 4](#_Toc356935328)

[3.3. Trabalhos Futuros 4](#_Toc356935330)

[REFERÊNCIAS 5](#_Toc356935331)

# CAPÍTULO 1: INTRODUÇÃO

## 1.1 A empresa onde o estágio foi realizado

O estágio está ligado ao Projeto do Museu de Fauna e Flora do ICMC [[1]](#footnote-1)foi realizado no Instituto de Ciências Matemáticas e Computação que é uma das cinco unidades de ensino e pesquisa da USP instaladas em São Carlos, juntamente com o Gabinete de Gestão e Planejamento, vinculado à Diretoria do ICMC, principal idealizadora do projeto. O projeto do Museu de Fauna e Flora tem o apoio do grupo de pesquisa Memória Virtual[Nakagawa,2014], vinculado ao Laboratório de Engenharia de Software do ICMC, que desenvolve o Memória Virtual, um sistema livre que possibilita a catalogação e disseminação de informações sobre acervos históricos utilizando tecnologias livres para implementação e implantação do sistema.

## 1.2 Objetivos do Trabalho

O projeto do Museu da Fauna e Flora está vinculado ao programa ICMC-USP Gestão Socioambiental, que tem como objetivo integrar as ações socioambientais do ICMC, sendo de extrema importância estarem aliados para atender seus objetivos em conjunto.

Nesse cenário, os objetivos deste projeto são:

* O desenvolvimento completo de um sistema web para ser acessado publicamente por todos na comunidade, ajudando assim na difusão da Fauna e Flora presentes no Instituto, bem como a sustentabilidade ambiental, como parte do Programa ICMC-USP de Gestão SocioAmbiental.
* Obtenção de ser um sistema de fácil personalização e implantação a fim de que sua replicação, junto com o projeto do Museu da Fauna e Flora seja acessível para outras unidades da USP e outras instituições, contribuindo e promovendo a sustentabilidade ambiental.
* Alinhamento e trabalho conjunto com o sistema Memória Virtual, pois o sistema web do Museu da Fauna e Flora será o primeiro a utiliza-lo. Assim, um dos objetivos principais do desenvolvimento do sistema web do Museu inclui a utilização do sistema Memória Virtual, procurando utilizar seus métodos e padrões de desenvolvimento de software livre.

## 1.3 Organização da Monografia

Os próximos capítulos da monografia descrevem como foram realizadas as atividades para o desenvolvimento do sistema, a descrição do projeto em que o estagiário se encontra inserido, as técnicas e métodos utilizados, as tecnologias que foram utilizadas, a descrição das atividades realizadas, desde a coleta de dados, até a versão final do sistema, e as dificuldades e limitações encontradas durante o processo de desenvolvimento.

# CAPÍTULO 2: DESENVOLVIMENTO DO TRABALHO

## 2.1. Descrição do Projeto

O projeto do Museu da Fauna e Flora foi proposto em 2013, com o intuito de exibir para a comunidade e incentivar à preservação dos recursos ambientais existentes do campus e da região desses recursos, como parte das ações da superintendência de Gestão Ambiental da USP.

Como definido, foi então criado o Museu Virtual da Fauna e Flora, para ser um sistema web online vinculado ao ICMC, que teria informações sobre toda a fauna e flora da região do ICMC, por meio da catalogação das espécies de fauna e flora, utilizando o sistema Memória Virtual, um sistema web para catalogação de acervos históricos. Em sua essência o Memória Virtual é um sistema disponibilizado em uma plataforma web, e de que possibilita o armazenamento de informações e sua disseminação (através da mesma e de módulos de serviços web) sobre acervos históricos, incluindo bens naturais, onde se encaixa o contexto do Museu. Sendo assim o Sistema Memória Virtual desenvolvido no contexto de projetos de pesquisa financiados principalmente pela FAPESP, Pró-reitoria de Cultura e Extensão da USP e Pró-reitoria de Graduação da USP, mostra-se bastante adequado para ser usado também no contexto do Museu da Fauna e Flora do ICMC.

O cadastro seria feito nesse sistema, e então o sistema do Museu é uma interface, que através de comunicação via serviços web, consulta a base de dados da Memória Virtual e exibe aos visitantes as informações que estão procurando. Por exemplo, um visitante poderia querer informações de uma determinada árvore localizada bem em frente à sala em que ele tem aula no período da manhã, então ao acessar o sistema seriam exibidas algumas informações da árvore, tais como sua história, quem plantou, quanto tempo estimado ela tem de vida, fotos, entre outras.

Algumas questões de início já foram levantadas: quais informações seriam cadastradas sobre um exemplar, como seria o cadastro da fauna, como um exemplar poderia ser encontrado no sistema, e como seria a comunicação do sistema com o Memória Virtual, e como definir um sistema genérico, para poder ser utilizado por outras entidades, não sendo apenas um sistema, mas uma aplicação personalizável, e distribuível para toda a comunidade.

## 2.2 Métodos, Técnicas e Tecnologias Utilizadas

### 2.2.1 Métodos

A equipe de desenvolvimento foi composta apenas pelo autor desta monografia, pois por se tratar de um projeto recém-criado, o Museu da Fauna e Flora, contava apenas com o apoio do Gabinete de Gestão e Planejamento e pela docente orientadora e responsável pelo Memória Virtual, sendo o autor desta monografia alocado para trabalhar junto às instalações do projeto Memória Virtual, onde haveria maior envolvimento com a equipe de desenvolvimento das atividades do Memória Virtual para inclusive facilitar a utilização e integração com o Projeto do Museu da Fauna e Flora. Assim, a melhor técnica encontrada para o desenvolvimento do sistema foi a entrega parcial de partes do sistema, seguindo um modelo de processo de software incremental[[2]](#footnote-2), onde foram coletados requisitos, desenvolvido uma parte funcional do sistema, reavaliada, testada, revisão dos requisitos, coleta de novos, mudança, e novo desenvolvimento das demais partes do sistema.

### 2.2.2 Técnicas

O desenvolvimento do sistema segue o padrão arquitetural *MVC* ( Modelo, Visão, Controlador), facilitando o desenvolvimento, e por ser um sistema de código aberto, a reutilização do código. O *MVC*, visa separar o sistema em três camadas, sendo: a) camada de modelo, que contém as entidades, regras de negócio, comunicação com sistemas externos; b) camada de visão, que se refere à apresentação do sistema, a saída dos dados, e por onde são feitas as requisições por parte dos usuários; c) o controlador por meio do qual as requisições passam para a camada de modelo, e os resultados da camada de modelo vão para a camada de visão.

Para auxiliar no desenvolvimento do sistema e garantir um software de qualidade foram utilizadas técnicas de desenvolvimento seguindo os padrões de engenharia de software descritas a seguir:

* Coleta de requisitos com o cliente (que no caso são o Gabinete de Planejamento e Gestão do ICMC e a docente responsável pelo estagiário), gerando um documento de requisitos básicos para o sistema.
* Criação de um diagrama de casos de uso utilizando a UML *(Unified Modeling Language)* do sistema com base no documento de requisitos, e, em seguida, a criação e descrição dos casos de uso do sistema.
* Construção do modelo conceitual que exibe as entidades existentes na definição do contexto da aplicação.
* Construção do diagrama de implantação em UML, que identifica os elementos de software e hardware existentes para a implantação do sistema.
* Escrita do documento de arquitetura de software, que visa a identificar os componentes, técnicas e padrões utilizados para o desenvolvimento do sistema suas restrições e apoio aos requisitos.
* Testes de software, modelo de processo de desenvolvimento incremental, entre outras...

### 2.2.3 Linguagens de Programação

As Linguagens de programação que foram utilizadas para o desenvolvimento do sistema web são:

**XHTM** [JEANDROCK, ERIC]**,** linguagem utilizada para desenvolvimento das páginas do sistema. Quando um cliente faz a requisição de uma página do sistema, seu pedido é interpretado em tempo real pelo servidor e a página anotada com a linguagem é carregada utilizando o controlador (camada de controle) e a partir dos modelos é exibida ao usuário no formato de uma página web.

**HTML** [JEANDROCK, ERIC], as páginas são criadas seguindo o padrão da ferramenta Facelets**,** criando uma anotação que suporta reuso de componentes, declaração de taglibs da linguagem Java, entre outras facilidades do Facelets, criando a camada de **Visão** da aplicação.

**JavaScript** [JEANDROCK, ERIC]**,** linguagem interpretada pelos navegadores, isto é, pelo cliente/visitantes do sistema web em tempo de execução, a fim de deixar a navegação mais interativa, através de requisições assíncronas e efeitos visuais (utilização de slides, mapas…), foi utilizada para efeitos nos menus do sistema, slides de imagens, exibição de fotos, iteração com mapa de busca e nas configurações do administrador, entre outros locais do sistema.

**Java** [JEANDROCK, ERIC]**,** linguagem orientada a objetos independente da plataforma que está executando, pois utiliza uma máquina virtual, oque assegura o requisito de portabilidade. Foi selecionada como linguagem do sistema por possuir a tecnologia JEE7, que segue o modelo MVC.

### 2.2.4 Tecnologias Utilizadas

Tecnologias utilizadas que auxiliaram no desenvolvimento do sistema, são conceitos e pacotes de códigos desenvolvidos para auxiliar no desenvolvimento de software:

**JAVA EE 7** [[3]](#footnote-3)é a mais nova implementação da plataforma para aplicações java web que oferece uma implementação em sistemas distribuídos e multicamada. Essa foi a plataforma a ser escolhida, pois é tolerante a falhas e oferece uma ótima perspectiva para implementação seguindo o modelo MVC (Modelo , Visão, Controle). As tecnologias a seguir fazem parte da especificação Java EE.

**JPA 2.0** é a API padrão java para persistências de dados independente do banco de dados a ser utilizado, sendo necessária a anotação e mapeamento da linguagem java. JPA realiza o acesso ao banco de dados do sistema ficando dentro da camada de **Modelo** do sistema, agilizando e facilitando o desenvolvimento do sistema.

**EJB 3.1** componente utilizado na camada de modelo do servidor, é uma aplicação a ser criada que cuida somente das regras de negócio do sistema, fazendo comunicação com banco de dados e serviços externos, ficando responsável pela maior carga. Essa tecnologia permite a execução da aplicação em um contêiner transacional, distribuídos e seguro, sendo acessível pela aplicação web, ou por qualquer outra aplicação a ser criada utilizando tecnologia Java.

**JSF 2.2(Java Server Faces)** é um framework que segue o padrão MVC para aplicações web, que possui inúmeros componentes para a construção da parte visual da aplicação, é faz parte da camada de **Controle** da aplicação, que é acessado pela camada de **Visão**, anotada em XHTML utilizando **Facelets**.

**Facelets,** tecnologia utilizada para criar a camada de visão, utilizando de marcações no formato XML, pode-se utilizar componentes baseados em JSF, podendo criar visões, como tabelas, entradas de texto, templates, baseados nos controladores criados utilizando JSF, a partir de páginas XHTML.

**Ajax** é uma tecnologia que deixa a navegação com o usuário mais iterativa e agradável. utiliza requisições assíncronas com o servidor através de JavaScript e XML[[4]](#footnote-4), não sendo necessário a total requisição de uma página pelo visitante do sistema, e sim somente uma parte da página, utilizada em quase todo o sistema, garantindo uma navegação iterativa para os visitantes.

**ApacheCommons**[[5]](#footnote-5), biblioteca da Apache Foundation de código aberto, escrita em java, visa à distribuição e utilização de códigos muito utilizados pela comunidade Java, utilizada na camada de modelo do sistema para conversão de dados vindos dos serviços web.

**jQuery[[6]](#footnote-6),** framework de código aberto e desenvolvido utilizando JavaScript. Facilita o desenvolvimento, de componentes web, sendo responsivo e garantindo o desenvolvimento padronizado. É muito utilizado para criação de plug-ins por outros frameworks e componentes, como o BootStrap.

**BootStrap[[7]](#footnote-7),** conjunto de componentes escritos utilizando a linguagem Less, que quando compilada gera componentes em CSS, utilizados para a criação de páginas Web, contento templates para formulários, botões, tipografia, navegação, menus, etc.… Pode ser usado junto com JavaScript criando componentes agradáveis ao usuário, utilizado para desenvolvimento da camada de *front-end²* do sistema juntamente com jQuery.

**WebServices** [JEANDROCK, ERIC]são serviços disponíveis na web que auxiliam na integração entre sistemas, concedendo um “serviço” do servidor (fornecedor) para o cliente (requisitante), no qual o cliente faz requisições do serviço e o servidor devolve uma resposta. É é a tecnologia que permite a utilização de dados de sistemas externos, no caso os dados da flora e fauna cadastrados no sistema externo Memória Virtual, e do conteúdo das páginas no outro sistema externo Wordpress.

### 2.2.5 Ferramentas Utilizadas

As ferramentas a seguir são software que auxiliaram no desenvolvimento do trabalho do sistema web:

**PostgreSQL 9.2[[8]](#footnote-8),** sistema gerenciador de banco de dados (SGBD) utilizado para desenvolvimento do sistema, além de ser um sistema livre de gerenciamento de banco de dados é muito utilizado no mercado atual como alternativa a sistemas fechados.

**Eclipse** [[9]](#footnote-9)é a IDE [[10]](#footnote-10)de desenvolvimento adotado para desenvolvimento do sistema. É uma interface de desenvolvimento com código livre, e muito utilizado no mercado, desenvolvida em java, suporta vários plug-ins para interagir com diversas ferramentas e outros softwares.

**GlassFish 4.0[[11]](#footnote-11),** servidor de aplicação utilizado para a implantação do sistema, por ser um software de código aberto e seguir as especificações Java EE, seguindo assim os padrões do sistema do Museu.

**SVN[[12]](#footnote-12),** sistema para controle de versão, foi utilizado para desenvolvimento do sistema a fim de ter um maior controle sobre o código e garantir sua integridade, o servidor da aplicação encontra-se em um servidor do **Google Code [[13]](#footnote-13)** ferramenta online para armazenamento de projetos de código aberto, que oferece o SVN como ferramenta de controle de versão, necessitando apenas de um plug-in no Eclipse para comunicação.

**Corel Draw** [[14]](#footnote-14) utilizado para criação do logo do Museu de Fauna e Flora, através da vetorização das imagens, a fim de criar uma identidade que poderia ser usada também em outros locais.

**Astah Community** [[15]](#footnote-15)software para modelagem UML , utilizado para criação dos diagramas UML do sistema.

## 2.3. Descrição das Atividades Realizadas

Para o desenvolvimento do projeto foram especificados alguns requisitos básicos: que o sistema fosse reutilizável, tanto o código como reutilizado em outras instituições; que fosse de fácil uso, e que utilizasse o sistema de catalogação Memória Virtual.

Tendo essas premissas, deu-se inicio ao trabalho de desenvolvimento do sistema, iniciando pela criação do documento de requisitos para tentar capturar melhor esses e outros requisitos que o software teria que seguir, passando por outras documentações, pela codificação do sistema, criação do logo do Museu e sistema Web, trabalhos para lançamento do projeto junto ao sistema de Gestão Ambiental do Instituto que incluía a criação de um logo oficial para o Museu, e a integração com placas contendo códigos QR para sua catalogação no sistema web.

### 2.3.1 Levantamento de requisitos

Para fazer o levantamento de requisitos do sistema, foram entrevistados os envolvidos com a idealização e desenvolvimento do projeto: Paulo Ernesto Celestini do Gabinete de Planejamento e Gestão do ICMC, e a Prof.ª Drª Elisa Yumi Nakagawa. Durante a consulta e levantamento dos requisitos foi criado o documento de requisitos do sistema com base nas entrevistas e padrões estabelecidos no documento de arquitetura de software, garantindo um sistema web completo para os visitantes.

#### 2.3.1.1 Documento de Requisitos

A seguir nesta seção é exibido o desenvolvimento do documento de requisitos do sistema:

##### Propósito

O propósito deste documento é definir os requisitos do sistema web do Museu da Fauna e Flora que tem como objetivo disseminar o acervo da Fauna e Flora presente no ICMC, sendo um sistema personalizável e que possa ser implantado em outras instituições.

##### Visão Geral

O sistema será uma aplicação Web, que visitantes poderão acessar e encontrar informações sobre a Fauna e Flora catalogadas, como informações, eventos e notícias sobre o Museu, podendo ser totalmente personalizável, a fim de ser implementada em outras instituições, de modo a incentivá-las a ter cada uma seu próprio museu. Além disso devendo ser um software livre.

O sistema usará como meio de catalogação o sistema Memória Virtual, que é um fruto de pesquisa do Laboratório de Engenharia de Software (Labes) do ICMC. Esse sistema permite a catalogação de qualquer bem histórico material ou imaterial, e possui um módulo que fornece buscas através de um WebService. Os eventos e notícias do sistema também ficarão armazenados em uma aplicação em PHP chamada Wordpress, uma aplicação que fornece ao usuário a montagem de um blog. Existe um Plug-in sobre licença livre, que adiciona um modulo de WebService no Wordpress, assim as páginas, eventos e notícias serão armazenados na aplicação no formato de blog, sendo acessadas por meio de um WebService para o sistema do Museu.

##### Perspectiva do Sistema

O sistema será implementado utilizando tecnologia java para Web, mais especificamente a especificação Java EE 7, que tem como servidor de aplicação o Glassfish, sobre licença de software livre da Oracle.

O sistema poderá ser acessado por qualquer usuário desde que o mesmo possua acesso a URL onde estará implementada a aplicação do sistema.

O sistema acessara dois WebServices, um que conterá os exemplares do Acervo de Fauna e Flora, e outro que conterá o conteúdo das páginas, eventos e notícias.

O Logo do sistema, seu slide principal, a localização dos WebService, as mensagens (todo o texto exibido no sistema) e entre outras coisas, serão totalmente personalizáveis, a fim de que o sistema seja usado por outras instituições.

##### Características dos Usuários

O Sistema possui ao todo 2 tipos de usuários(visitantes do sistema, e administrador), e um usuário adicional responsável por gerenciar o conteúdo dos WebServices que não estão totalmente acoplados ao sistema. Assim , os usuários são o administrador e visitantes, e o adicional gerenciador de conteúdo/biólogo.

##### Suposições e Dependências

O sistema poderá ser instalado em qualquer sistema em que o servidor de aplicação Glassfish consiga ser instalado, e que possua também um servidor de banco de dados PostgreSQL usado para armazenar dados de Administração e fazer cache de imagens dos WebServices.

##### Funções básicas do Sistema

O Sistema permitirá ao visitante

* Acessar o sistema.
* Buscar por notícias e eventos.
* Buscar por exemplares no acervo.
* Navegar entre as páginas.
* Visualizar um exemplar do acervo.

O Sistema permitirá ao administrador

* Personalizar totalmente o sistema.
* Alterar todo o texto presente no sistema.

Interfaces com os Usuários

O sistema terá uma página inicial que conterá o conteúdo atualizado com as últimas notícias, eventos e catalogações do acervo, e também uma barra superior com todas as principais páginas de acesso do sistema, e um local para busca.

Interfaces de Comunicação

O sistema possui comunicação com banco de dados PostgreSQL, e com Webservices via Socket HTTP.

***Requisitos Funcionais***

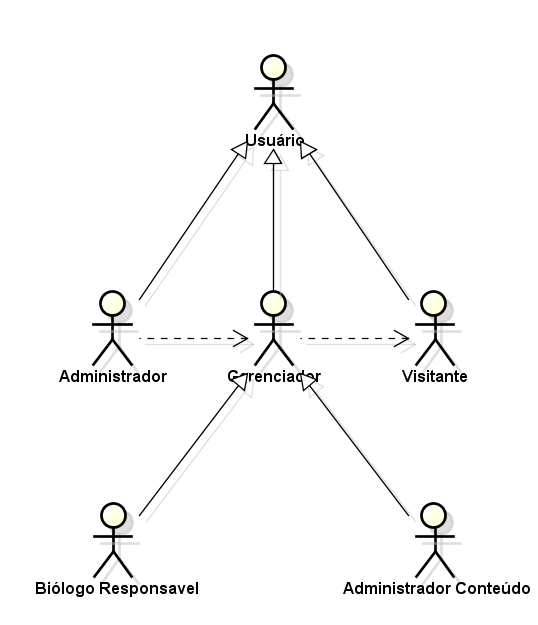
1. O sistema deve permitir a navegação para as páginas principais através de qualquer página do sistema, isto é, o usuário poderá ir para qualquer página do sistema através de outra página qualquer, não ficando preso e necessitando voltar a páginas anteriores.
2. O sistema deve conter informações sobre eventos e notícias e permitindo a busca deste conteúdo.
3. O sistema deve armazenar páginas sobre conteúdo institucional do Museu.
4. O sistema deve exibir as últimas catalogações cadastradas no sistema Memória Virtual e permitir sua busca.
5. O sistema terá um sistema de busca iterativo que usará um sistema de mapas para busca de exemplares do acervo.
6. O sistema deverá exibir um item do acervo e todas as informações relevantes , ainda a serem definidas, cadastradas no sistema Memória Virtual.
7. O sistema deve ser personalizável, a fim de outras instituições usarem o mesmo para seus próprios Museus.
8. O sistema a administração do sistema deve possuir um sistema de login para personalização do sistema.
9. O sistema deverá armazenar informações dos visitantes do sistema, a fim de ter dados estatísticos e o perfil dos visitantes para o administrador do sistema ter um controle sobre o fluxo de visitantes do sistema.

***Requisitos Não Funcionais***

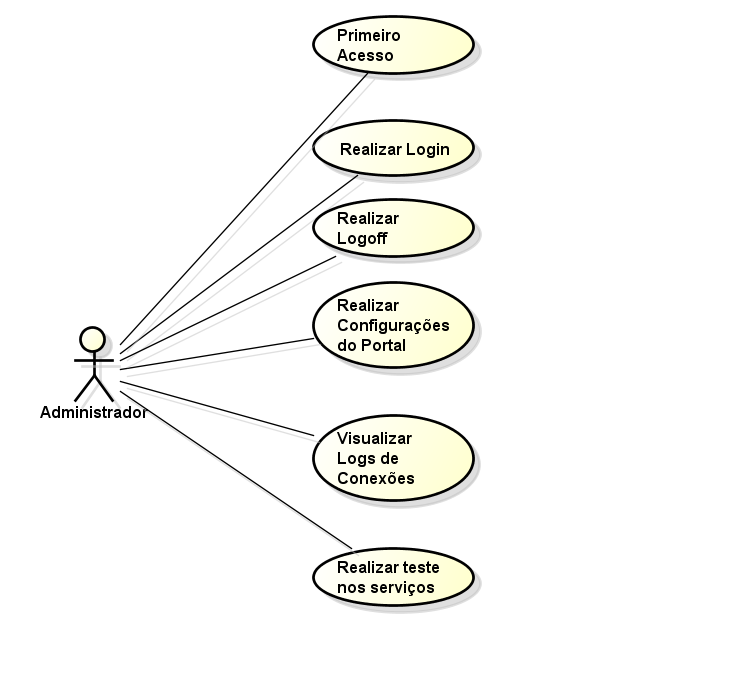
1. Desempenho, o sistema deve ser ágil e de carregamento rápido.
2. Confiabilidade, o sistema deve ser confiável, e trazer informações completas e sem erros para os visitantes do sistema.
3. Disponibilidade, o sistema deve ficar online o maior tempo possível, prevenindo ataques e sobrecarga de usuários.
4. Segurança, o sistema de login do administrador deve ser seguro e não conter falhas de segurança e autenticação.
5. Manutenibilidade, por seguir o padrão Java EE e o Modelo MVC, a manutenibilidade do sistema deverá ser fácil e rápida.
6. Portabilidade, o sistema trabalhará usando linguagem java e o servidor de aplicação Glassfish(baseado em java) sendo possível trabalhar em todas as plataformas que suportam a máquina virtual Java.
7. Compatibilidade com navegadores web, o sistema web deve ser compatível com vários clientes, sendo compatível com todos os navegadores mais usados no mercado atual.
8. Armazenamento de conteúdo, o conteúdo referente a catalogação ficará armazenado em um sistema externo, o Memória Virtual, e o conteúdo do sistema referente ao conteúdo de notícias, eventos e páginas, será armazenado em outro sistema externo, Wordpress, ambos serão comunicáveis via interfaces de webService seguras, e serão gerenciáveis.

#### 2.3.1.2 Diagrama de Casos de Uso

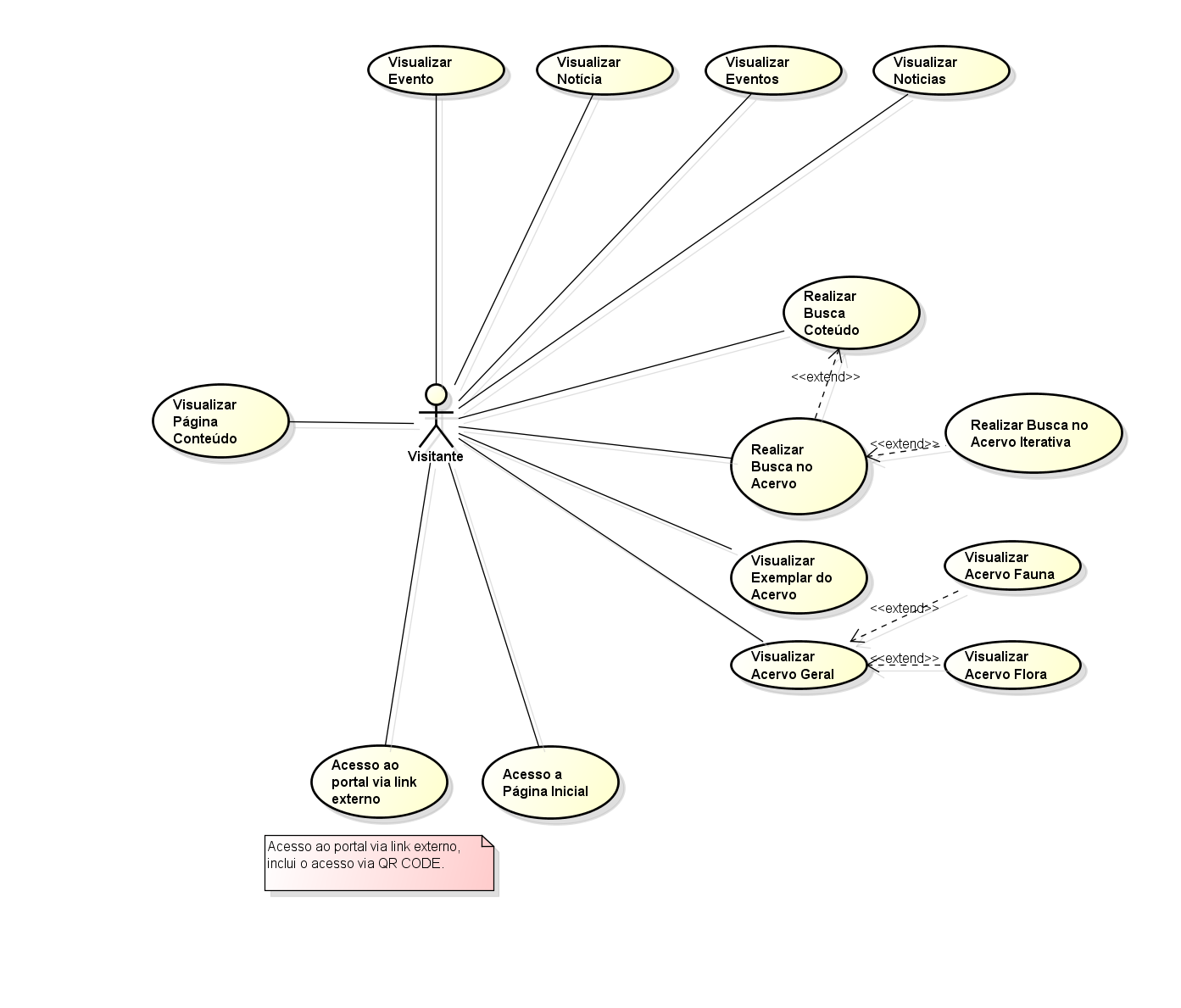
A partir do documento de requisitos do sistema, foi criado um diagrama de casos de uso, procurando abranger todos os usuários do sistema e seus respectivos usos do sistema, gerando a partir do diagrama de casos de uso todos os documentos dos casos de uso primários e essenciais para o sistema, contendo sua descrição, atores envolvidos, fluxo básico e alternativo. A seguir (figuras 1,2,3 e 4) são exibidas algumas imagens do diagrama de casos de uso do sistema.



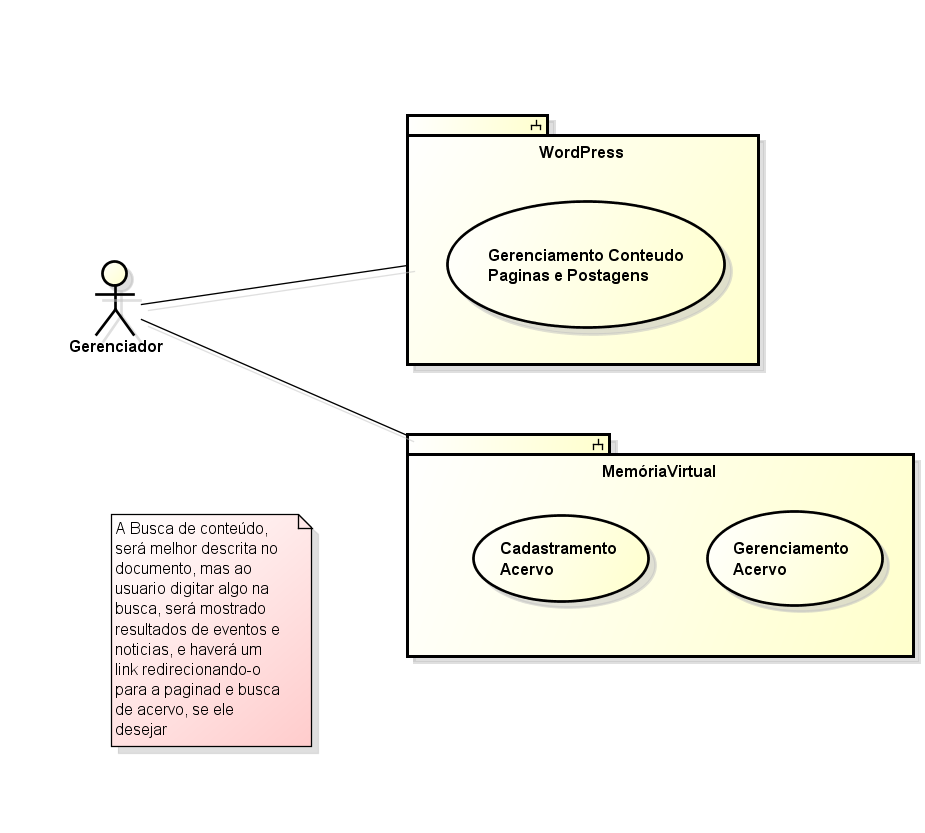
**Figura 1: Diagrama de casos de uso, os atores.**



**Figura 2: Diagrama de casos de uso, os casos de uso relacionado ao administrador do sistema.**

****

**Figura 3: Diagrama de casos de uso, os casos de uso relacionado ao visitante do sistema web.**

****

**Figura 4: Diagrama de casos de uso, os casos de uso relacionado ao gerenciador de conteúdo do sistema.**

**2.3.2 Desenvolvimento e codificação**

#### 2.3.2.1 Protótipos e primeira versão

Como o estagiário não tinha muito envolvimento com técnicas de desenvolvimento web, decidiu-se inicialmente criar um protótipo inicial em HTML a fim de aprender mais sobre a linguagem de marcação, e aprofundar-se mais em CSS juntamente, gerando o primeiro protótipo do sistema, exibida na figura 5.

Procurando capturar todas as características de um sistema web, procurando exibir componentes como caixa de busca, localização do logo, menu de administração, conteúdo, menu para o usuário, procurando também atender os requisitos de navegação do sistema.

#### 

#### Figura 5: Primeiro protótipo do sistema

A partir do esqueleto criado deu-se início ao desenvolvimento da primeira versão, em que foram criados mais protótipos e foram sendo aprimorados, até chegar ao final da primeira versão, que atendia aos requisitos do visitante ao sistema, e comunicações externas.

#### 2.3.2.2 Comunicação com serviços externos

Para o desenvolvimento das interfaces de comunicação externa, foi estudado o protocolo de troca de mensagens para serviços web SOAP[[16]](#footnote-16). Os serviços web que utilizam o protocolo SOAP possuem um WSDL[[17]](#footnote-17). Assim para a utilização dos serviços utilizados no sistema, foi utilizada uma ferramenta da IDE Eclipse, para geração de código na linguagem java para utilização do serviço, através do WSDL dos dois serviços do sistema (Memória Virtual e Wordpress), já que ambos utilizam o protocolo de troca de mensagens SOAP, ficando esse código gerado pela ferramenta acoplado na camada de modelo do sistema.

#### 

**Figura 6: código fonte gerado a partir do arquivo WSDL para comunicação com serviço Web**

#### 2.3.2.4 Segunda versão do sistema e limitações

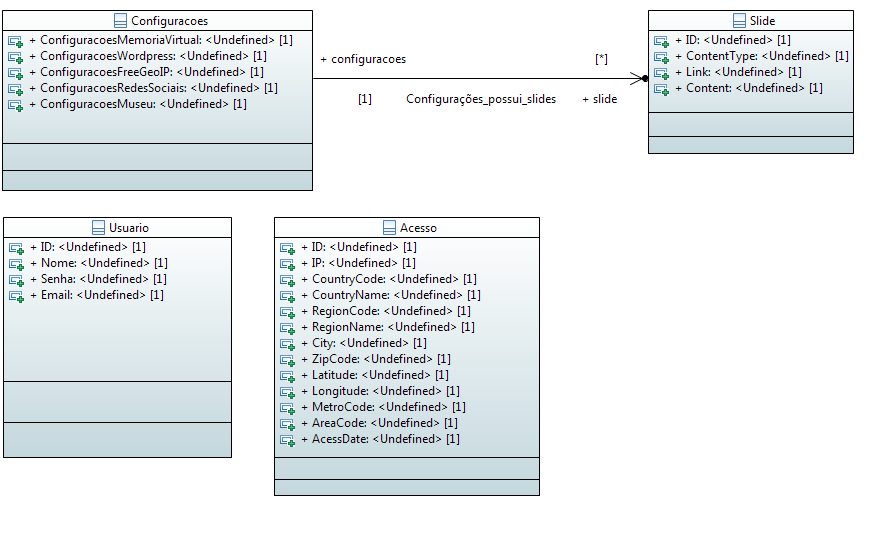
Com a entrega da primeira versão do sistema, com quase todas as funcionalidades dos visitantes já codificadas, surgiu uma limitação perante o Memória Virtual, que impedia a catalogação de inúmeros itens no acervo do Museu, pois a busca que era realizada no sistema externo era uma busca geral, no qual eram retornados todos os resultados da busca em uma resposta SOAP (em formato XML). Isto gerava uma sobrecarga de dados e até indisponibilidade no serviço algumas vezes, pois o serviço ficava indisponível com tamanha carga sendo requisitada. Assim necessitando desta e outras funcionalidades, o estagiário foi requisitado para auxiliar o grupo de pesquisa a fim de programar essas e outras funcionalidades do sistema memória virtual, descrita na seção seguinte.

Simultaneamente a esse processo, foi desenvolvida a segunda versão (figura 7) do sistema que teria uma interface renovada, utilizando o framework de *front-end* web BootStrap, que auxilia na criação de interfaces utilizando da linguagem estilizada CSS juntamente com HTML e JavaScript, para criação de interfaces responsivas e que atendem dispositivos de vários tamanhos, assim atendendo requisitos de compatibilidade com múltiplos navegadores e plataformas. Assim, restou apenas a interface de administrador a ser desenvolvida.

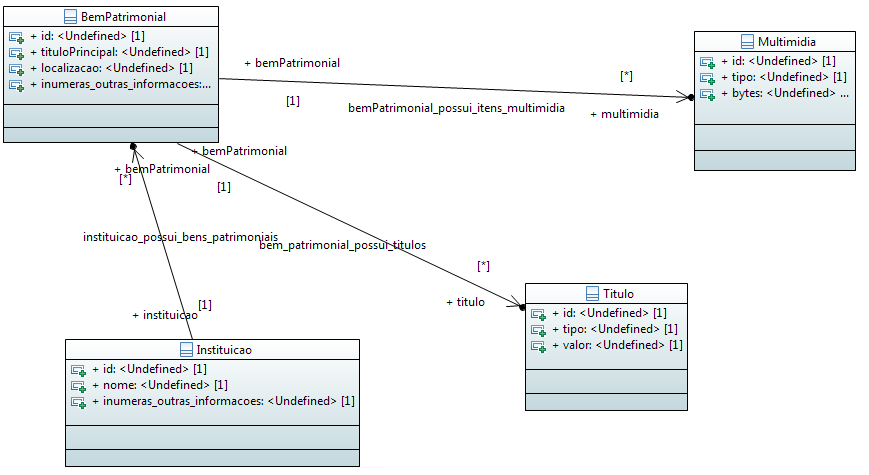
**Figura 7: Interface da Versão 2 do sistema do Museu para os visitantes.**

#### 2.3.2.5 Desenvolvimento Interface de administração

Para o desenvolvimento da interface de administração, seria levado em conta seis casos de uso que, embora sejam poucos, são cruciais para o funcionamento, personalização e configuração do sistema. Para a interface de administração, foi preciso criar um modelo de domínio para suportar as configurações do sistema (figuras 8 e 9).



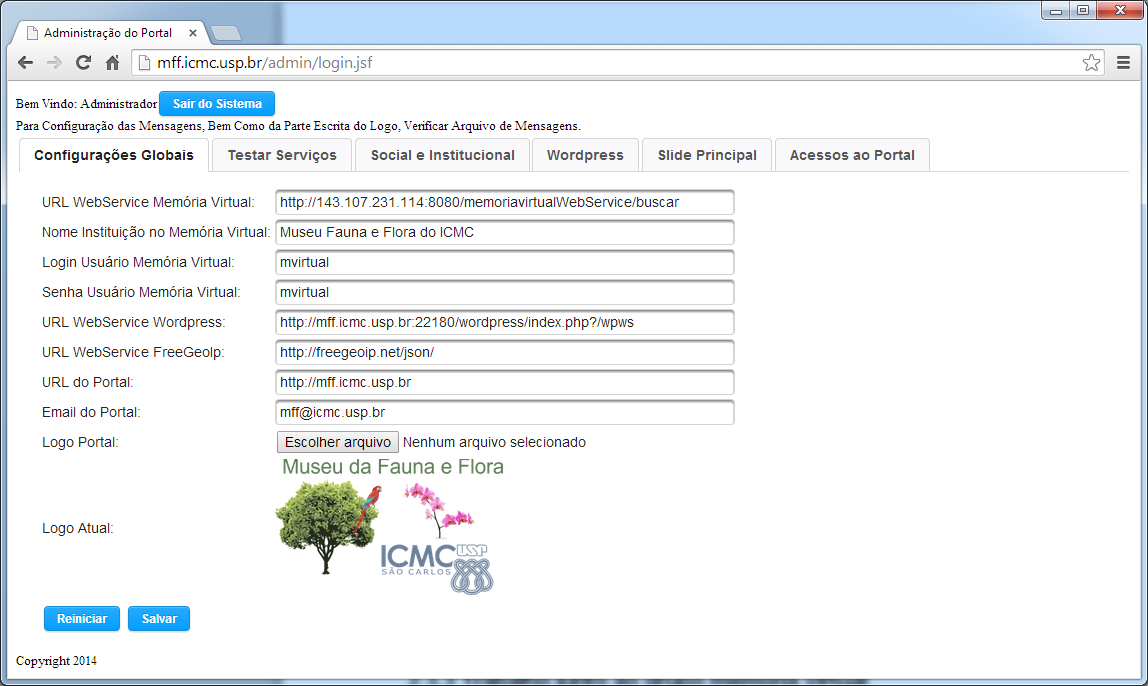
**Figura 8: Modelo conceitual do sistema.**



**Figura 9: Modelo conceitual do sistema relacionado ao Memória Virtual.**

A partir do modelo conceitual foram criadas as classes na linguagem java, utilizando-se das marcações da API JPA para persistência dos dados na base de dados.

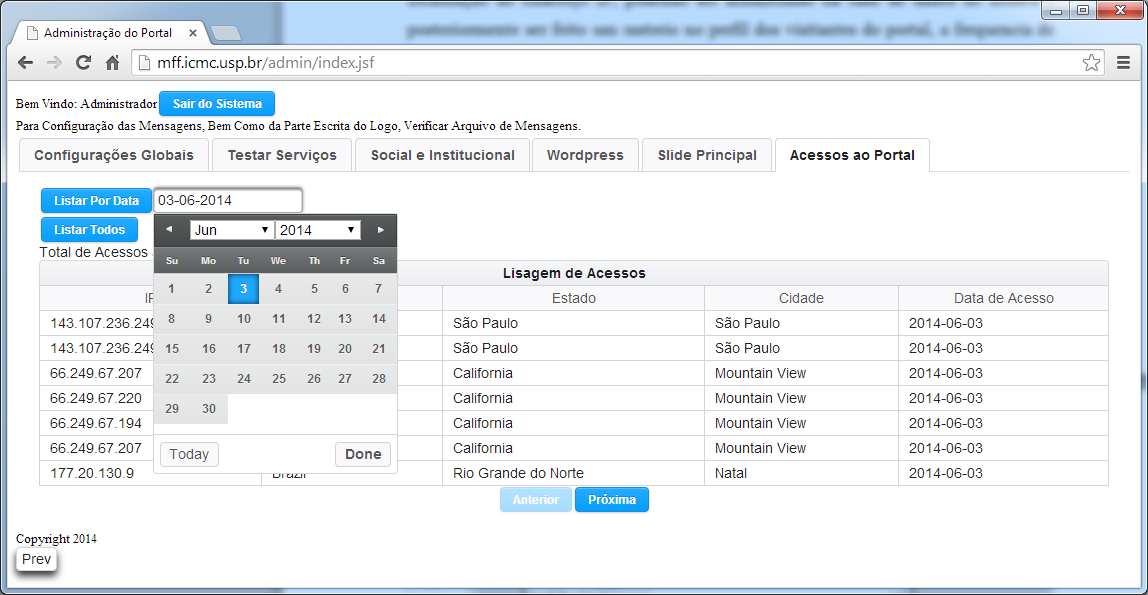
A partir da criação da base de dados pelo framework de persistência de dados, foi programada a área de administração do sistema web, sendo uma interface de simples acesso, e de fácil configuração, mostrada a seguir na figura 10.



**Figura 10: Interface de Administração do sistema**

Para o desenvolvimento da funcionalidade de log de conexões (exibido na figura 11) foi utilizado um serviço externo chamado FreeGeoIP, um software livre, que possui uma instancia executando de livre acesso ao limite de 100.000 requisições por hora. Esse serviço ao receber como parâmetro um endereço IP retorna um objeto no formato JSON [[18]](#footnote-18)com informações sobre a localização do endereço IP, podendo ser armazenado na base de dados do museu e posteriormente ser feito um rastreio no perfil dos visitantes do sistema, a frequência de visitas por dia, entre outras informações.

Juntamente com o desenvolvimento da interface de administração, foi iniciado o desenvolvimento da funcionalidade para atender múltiplos idiomas, não sendo apenas uma tradução em tempo real, possuindo todo um mapeamento das mensagens do sistema, podendo ser portado para diversos idiomas. Atualmente há suporte para o Português do Brasil e Inglês Americano. Esta funcionalidade é programada utilizando uma função do framework Java Server Faces, que detecta a requisição da linguagem do navegador e carregar o arquivo de mensagens da língua requisitada.



**Figura 11: Log de conexões de visitantes ao sistema**

**2.3.3 Trabalho junto ao grupo memória virtual**

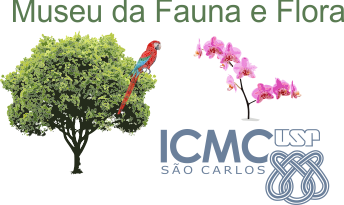
Como citado na Seção 2.3.2.4, o serviço externo utilizado para o sistema para catalogação dos exemplares de fauna e flora continha algumas limitações, e como o estagiário já estava inserido no projeto, foi pedido auxílio para implementação e melhoramento das interfaces de serviços web do sistema Memória Virtual, o que contribuiu muito para ambas as partes, pois ao utilizar o serviço web do sistema Memória Virtual para o desenvolvimento do sistema do Museu, pode encontrar problemas e falhas no sistema Memória Virtual e pode contribuir para o desenvolvimento do projeto, agilizando o desenvolvimento de ambos os projetos, tanto do Museu e do Memória Virtual.

Algumas das atividades desempenhadas foram: desenvolvimento e modificação de serviços no sistema, observação de defeitos e resolução de defeitos no sistema, implementação de novas funcionalidade.

**2.3.4 Desenvolvimento do logotipo do Museu**

Como parte das atividades a serem desempenhadas pelo estagiário, estava a criação de uma identidade visual para o Museu, que deveria ser criada usando uma ferramenta que obtivesse alta resolução, e que fosse uma identidade forte para o Museu, pois o logo estaria presente em placas, cartazes, no sistema, pôsteres, entre outros meios de mídias existentes. A melhor ferramenta encontrada foi o editor gráfico Corel Draw, além de ser um editor que gera imagens vetorizadas, já era utilizado para o logo do Instituto (que poderia ou não estar presente no logo do Museu) no formato da ferramenta.

Após o desenvolvimento de cinco logotipos, alguns reprovados pois afetavam a identidade visual do ICMC, e outros por não serem tão impactantes e representativos, chegou-se ao resultado final, como visto na Figura 12.



**Figura 12: Logotipo do Museu desenvolvido durante o estágio.**

## 2.4. Resultados Obtidos

Com todas as funcionalidades sendo atendidas, o sistema, encontra-se atualmente em fase de testes e coleta de requisitos para novas funcionalidades, e em testes para resolução de defeitos de interface, navegação e funcionalidades.

Com o padrão de desenvolvimento utilizado, o MVC, o sistema possui uma fácil manutenibilidade, sendo separado em camadas, o desenvolvimento de novos requisitos se torna ágil, bem como o reuso do código, por ser um software livre, fica bem acessível.

Aproveitando-se das últimas tecnologias utilizadas no mercado, o sistema funciona de forma ágil, por ter uma base de dados minimizada por conta de quase toda a carga de dados ficar em serviços externos, oque o torna uma aplicação leve, não necessitando de grande quantidade de poder computacional, definindo-se uma aplicação de fácil implantação e personalização em outras Instituições, sendo apenas necessário a instalação do ambiente de produção: banco de dados, e servidor de aplicação baseado na plataforma Java EE 7, lembrando que junto a distribuição do sistema será incluído um manual de instalação.

## 2.5. Dificuldades e Limitações

Algumas das dificuldades encontradas foram pertinentes às informações que seriam cadastradas no sistema Memória Virtual, e seriam exibidas no sistema do Museu. Como o sistema Memória Virtual permite a catalogação de um bem patrimonial de qualquer origem, isto é, ele possui inúmeros campos, mas nem todos seriam usados pelo Museu, então, sem uma pessoa especialista da área de biológicas, seria difícil fazer esse mapeamento. Mesmo com essa dificuldade, o mapeamento foi realizado, tendo ainda de ser aprovado, por um especialista, sendo que no sistema ainda são exibidas informações mínimas dos exemplares, e faltando ainda o término do mapeamento da Fauna a ser realizado.

Outra dificuldade encontrada, foi que quando solicitado o desenvolvimento do sistema e a utilização do sistema Memória Virtual, pensou-se ser um sistema já finalizado em pleno funcionamento e sem problemas, mas ao iniciar o uso do mesmo, o estagiário encontrou inúmeras dificuldades, e um sistema com problemas de desenvolvimento e ainda não finalizado, mas isto acabou sendo uma contribuição para o estagiário, pois ajudando o projeto Memória Virtual, contribuiu para seus estudos e desenvolvimento do sistema, mas por outro lado, acabou atrasando, o término do desenvolvimento do sistema.

# CAPÍTULO 3: CONCLUSÃO

## 3.1. Contribuições

O desenvolvimento do sistema do Museu da Fauna e Fauna trouxe e acredita-se que ainda vai trazer muito mais contribuições para o programa socioambiental, divulgando e a rica fauna e flora presente no ICMC, fazendo os visitantes do Instituto e do sistema, pararem para refletir que não é somente um local arborizado, mas um local que possui uma rica fauna e flora, que cada árvore possui uma história por trás, que deve ser preservada, criando um ambiente de aprendizado e socialização muito mais agradável para todos.

Como pessoa, o estágio proporcionou ao estagiário o contato com diversas ferramentas, padrões, métodos que não conhecia, expandindo seu conhecimento ainda mais na área . Uma grande contribuição presente foi o contato com diversas pessoas, de várias outras áreas, tanto para construção do sistema, como para o aprendizado mais profundo sobre a área ambiental, que incluem a participação de diversas palestras, curso para a criação da brigada de arboristas do instituto e o curso de fotografia de aves.

## 3.2. Relacionamento entre o Curso e o Projeto de Estágio

A formação em Sistemas de Informação contribui praticamente 100% para o desenvolvimento do estágio, considerando que o estagiário não detinha conhecimento algum sobre a área de informática ao ingressar no curso. Cada disciplina contribui para o desenvolvimento do projeto, algumas pouco, outras muito, como as disciplinas básicas de computação, que ajudaram a codificação do sistema, possibilitando a implementação dos casos de uso para a linguagem Java; já a Disciplina de Base de dados contribuiu para a criação e modelagem da base de dados do sistema. A disciplina de, Laboratório de Base de Dados teve uma contribuição enorme, pois nela que foi desenvolvido um projeto web, que permitiu ter uma base de um sistema simples, mas completo. A Engenharia de Software contribui para criação do documento de requisitos, dos modelos do sistema, ou seja, de toda a documentação correspondente ao sistema.

## 3.3. Trabalhos Futuros

Neste momento o desenvolvimento da versão atual da aplicação está terminado, ficando apenas correção de defeitos e pequenas modificações.

Para versões futuras, serão captados novos requisitos, que serão inseridos no diagrama de casos de uso e então detalhados para seu desenvolvimento.

Restam ainda o término da personalização do sistema, que possibilitarão uma interação melhor do visitante com o conteúdo.

Por último nos próximos passos será feita a definição de todas as informações da flora e fauna a serem cadastradas e exibidas para o visitante. Inicialmente elas já foram definidas, mas podem sofrer alterações, pois não foram definitivamente concretizadas, pois a coleta de informações sobre a flora (somente foram coletadas informações básicas da flora existente no ICMC, como nome popular, espécie, nome cientifico e origem) e fauna (não coletada nenhuma informação no momento) são mínimas.

Assim o desenvolvimento do projeto será finalizado, podendo ser distribuído e melhorado com o passar do tempo.

# REFERÊNCIAS

JEANDROCK, ERIC. et al. *The Java EE 7 Tutorial, Release 7 for Java EE Platform.*Oracle. Disponível em: < <http://docs.oracle.com/javaee/7/tutorial/doc/javaeetutorial7.pdf>> acesso em junho 2014.

Nakagawa, E. Y.; Bianchi, T.; Oliveira, M. C. F.; Sousa, E. P. M.; Andery, G. F.; Maldonado, J. C.; *Sistema Web Livre para Automatização do Padrão de Descrição da Informação*, Revista Digital de Biblioteconomia e Ciência da Informação, v. 12, n. 2, maio/agosto, 2014, p.173-192, ISSN 1678-765X.

Vilela Fonseca, M.;Luiz Alves, A..*MVC & FRAMEWORK*. Disponivel Em:  
<<http://www.cpgls.ucg.br/ArquivosUpload/1/File/V%20MOSTRA%20DE%20PRODUO%20CIENTIFICA/EXATAS/2-.PDF>> Acesso em junho 2014.

W3C.*W3C. Extensible Markup Language (XML)***.**  2014. Disponível em: <<http://www.w3.org/XML/>> acesso em junho 2014.

1. ICMC – Instituto de Ciências Matemáticas e Computação – Universidade de São Paulo, campus de São Carlos [↑](#footnote-ref-1)
2. http://www.inf.ufpr.br/lmperes/ciclos\_vida/ModeloIncremental.pdf [↑](#footnote-ref-2)
3. *Java Enterprise Edition Platform* [JEANDROCK, ERIC] [↑](#footnote-ref-3)
4. eXtensible Markup Language -Recomendação da W3C para escrita de documentos baseados em marcação. [↑](#footnote-ref-4)
5. <http://commons.apache.org/> [↑](#footnote-ref-5)
6. <http://jquery.com/> [↑](#footnote-ref-6)
7. <http://getbootstrap.com/> [↑](#footnote-ref-7)
8. <http://www.postgresql.org/> [↑](#footnote-ref-8)
9. <http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/keplersr2> [↑](#footnote-ref-9)
10. Integrated Development Environment [↑](#footnote-ref-10)
11. <https://glassfish.java.net/> [↑](#footnote-ref-11)
12. <http://www.eclipse.org/subversive/> [↑](#footnote-ref-12)
13. <https://code.google.com/> [↑](#footnote-ref-13)
14. <http://www.corel.com> [↑](#footnote-ref-14)
15. <http://astah.net/editions/community> [↑](#footnote-ref-15)
16. Simple Object Access Protocol[JEANDROCK, ERIC] **-** um protocolo de comunicação de serviços web, padronizado que utiliza mensagens no formato XML através do protocolo HTTP. [↑](#footnote-ref-16)
17. Web Services Description Language [JEANDROCK, ERIC]– um arquivo escrito na forma de uma espécie de contrato, fornecendo a descrição de todas as operações, que tipos de dados retornam, e como utiliza-las, facilitando o uso dos serviços. [↑](#footnote-ref-17)
18. <http://www.json.org/> [↑](#footnote-ref-18)