



---

**FACULDADE DE TECNOLOGIA DE AMERICANA**

**Curso Superior de Tecnologia em Análise e Desenvolvimento de**  
**Sistemas**

Rafael Garcia de Lima

**PROCESSO DE DESENVOLVIMENTO: SOFTWARE DE**  
**GERENCIAMENTO ADMINISTRATIVO**

Americana, SP

2018

Rafael Garcia de Lima

**PROCESSO DE DESENVOLVIMENTO: SOFTWARE DE  
GERENCIAMENTO ADMINISTRATIVO**

Trabalho de Conclusão de Curso desenvolvido em  
cumprimento à exigência curricular do Curso Superior de  
Tecnologia em Análise e Desenvolvimento de Sistemas.

Área de concentração: Engenharia de Software

**Americana, SP.**

**2018**

**FICHA CATALOGRÁFICA – Biblioteca Fatec Americana - CEETEPS**  
**Dados Internacionais de Catalogação-na-fonte**

AL71p LIMA, Rafael Garcia de

Processo de desenvolvimento: software para gerenciamento de documentos administrativos. / Rafael Garcia de Lima. – Americana, 2018.

95f.

Monografia (Curso de Tecnologia em Análise e Desenvolvimento de Sistemas) -  
- Faculdade de Tecnologia de Americana – Centro Estadual de Educação Tecnológica  
Paula Souza

Orientador: Prof. Ms. Clerivaldo Jose Roccia

1 Desenvolvimento de software 2. Engenharia de software. I. ROCCIA,  
Clerivaldo Jose II. Centro Estadual de Educação Tecnológica Paula Souza – Faculdade  
de Tecnologia de Americana

CDU: 681.3.05

**PROCESSO DE DESENVOLVIMENTO: SOFTWARE DE  
GERENCIAMENTO ADMINISTRATIVO**

Trabalho de graduação apresentado como exigência parcial para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas pelo CEETEPS/Faculdade de Tecnologia – FATEC/ Americana.

Área de concentração: Engenharia de Software.

Americana, 26 de junho de 2018

**Banca Examinadora:**



---

Clerivaldo José Roccia (Presidente)

Mestre

Fatec de Americana



---

Eduardo Antônio Vicentini (Membro)

Mestre

Fatec de Americana



---

Antônio Alfredo Lacerda (Membro)

Especialista

Fatec de Americana

## **Agradecimentos**

Agradeço minha família pelo apoio durante todos estes anos em que estivemos vivendo juntos. Agradeço aos professores que tiveram paciência e me ajudaram, orientaram e ensinaram, seja direta ou indiretamente, ao longo do curso.

## RESUMO

Trabalho de conclusão do curso superior tecnólogo de Análise e Desenvolvimento de Sistemas da FATEC de Americana, utilizando-se de todas as habilidades desenvolvidas ao longo do curso para o desenvolvimento de um software, junto da documentação do mesmo, visando auxiliar na gerência administrativa dos protocolos, quanto aos documentos, de instituições como um todo, independente do grau de ensino da instituição ser primário, fundamental, médio ou superior.

Tendo o início do desenvolvimento na parte de engenharia de requisitos, indo do levantamento até a especificação dos requisitos não funcionais, passando pela engenharia de software em si, consistindo na transformação dos requisitos dos clientes em diagramas UML, do mesmo modo que a modelagem e criação do banco de dados. Finalizando, conseqüentemente, no desenvolvimento e na documentação, fazendo uso das normas específicas das linguagens para criação da documentação. Utilizando-se, para o *front-end*, das linguagens *web* HTML e CSS, enquanto que o *back-end* mantêm-se na linguagem PHP, utilizando-se de MySQL e Phpmyadmin para o gerenciamento dos dados.

**Palavras-Chave:** Banco de dados, Desenvolvimento de software, Engenharia de software, SQL, Software Web, UML.

## **ABSTRACT**

Monography required to complete the bachelor degree in System Analysis and Development at the Technology College of Americana, making use of all the developed abilities to create a software, together with its documentation, focusing in improving the process of creating and managing protocols used by schools, independent of the school teaching level being elementary, middle-school, high school or college.

The first stage of the development was the requirement engineering, going from interviews with the user to specify functional and non-functional requirements to the software, transforming those requirements in UML diagrams and creating and modelling a database. Past that initial phase comes the software development and software documentation, using programming language rules and standards to develop a precise documentation as well. Making use of, for the front-end, web languages, such as HTML and CSS, having PHP as the back-end programming language and MySQL for the database, together with Phpmyadmin for the database administration.

**Keywords:** Database, Software development, Software engineering, SQL, Software web, UML.

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>10</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA.....</b>	<b>12</b>
2.1	Modelo de Desenvolvimento .....	12
2.2	Metodologia de desenvolvimento e engenharia de software .....	15
2.2.1	Levantamento de requisitos.....	16
2.2.2	UML: Diagramas e introdução.....	18
2.3	Ambiente de operação.....	20
2.4	Ambiente de manutenção .....	20
2.5	Design .....	21
2.5.1	UI (User interface) .....	21
2.5.2	UX (User experience) .....	22
2.6	Validação .....	23
2.7	Segurança .....	23
2.7.1	Tratamento de exceções.....	23
2.7.2	Variáveis globais e visibilidade das variáveis .....	23
2.7.3	Acesso ao software.....	24
2.8	Banco de dados.....	24
2.8.1	Queries SQL.....	25
2.8.2	Dicionário de dados.....	25
2.8.3	Modelo relacional de entidade lógica .....	25
2.9	Testes de software .....	25
2.10	Testes para aplicações web e fluxogramas da aplicação.....	25
<b>3</b>	<b>DESENVOLVIMENTO.....</b>	<b>27</b>
3.1	Requisitos funcionais .....	28
3.1.1	Requisitos de domínio .....	29
3.1.2	Requisitos de usuário .....	30
3.2	Requisitos não funcionais .....	41
3.2.1	Requisitos de produto .....	41
3.2.2	Requisitos Organizacionais .....	42
3.2.3	Requisitos Externos .....	43



3.3	Diagramas UML.....	44
3.4	Segurança .....	48
3.4.1	Tratamento de exceções.....	48
3.4.2	Variáveis globais e visibilidade das variáveis .....	48
3.4.3	Acesso ao software.....	50
3.5	Banco de dados.....	51
3.5.1	Queries SQL.....	51
3.5.2	Dicionário de dados.....	52
3.5.3	Modelo Relacional de Entidade Lógica.....	56
3.6	Testes de funcionalidade .....	57
3.7	Testes para aplicação web .....	60
<b>4</b>	<b>RESULTADOS DE DISCUSSÕES.....</b>	<b>62</b>
4.1	Software e usabilidade .....	62
<b>5.</b>	<b>CONSIDERAÇÕES FINAIS.....</b>	<b>73</b>
<b>6.</b>	<b>REFERÊNCIAS .....</b>	<b>74</b>
<b>7.</b>	<b>GLOSSÁRIO .....</b>	<b>80</b>

# 1 INTRODUÇÃO

Atualmente, grande parte das tarefas realizadas nas instituições de ensino são feitas de modo manual e arcaico; junto de pouco, ou nenhum, controle das versões dos documentos, acompanhamento dos mesmos – em termos de aprovação, andamento etc. – e registros além do documento em si. Esse modo administrativo gera tanto sobrecarga de trabalho logístico e administrativo, quanto gastos em demasia para a instituição, mesmo que essa tenha excelentes funcionários cuidando de ditos processos. Gastos com tinta, papel, manutenção dos equipamentos por sobrecarga de uso e energia são alguns dos problemas que, unidos ao tempo, tornam tais modos de realização das tarefas em ineficientes, além do trabalhoso processo (que acaba por ser, também, custoso) de se partilhar tais documentos com os demais setores.

O estudo se baseia no desenvolvimento de um software que visa solucionar tais problemas, relativos às tarefas realizadas na instituição. A instituição, em específico, utiliza de meios externos ou manuais para gestão dos documentos criados; onde mesmo de médio porte, possui diversos setores, tendo tanto cursos curriculares quanto extracurriculares em sua grade organizacional. Deste modo, a organização dos editais, certificados, solicitações, pedidos e demais documentos, requisitados tanto por alunos quanto por professores, acaba por demandar maior carga de trabalho para os funcionários da secretaria, coordenação, patrimônio e diretoria, responsáveis pela gestão de tais documentos, onde estes, muitas vezes, necessitam de passar por diversos dos setores citados a priori. Atualmente, os documentos criados, ou gerados por meio de processos, precisam ser compartilhados manualmente, por parte dos funcionários do setor, para que os demais funcionários tenham acesso a ditos documentos. Precisando, assim, salvar várias versões em diferentes máquinas, muitas vezes com versões desatualizadas, incompletas ou inválidas por meio das regras que devem ser seguidas. Este modo organizacional de protocolos acaba por gerar gastos em demasia com papel e tinta, por conta das versões incorretas que são impressas, além da sobrecarga de arquivos nas máquinas, onde a desorganização atrasa os demais processos.

A ideia principal, utilizando-se da tecnologia disponível, é facilitar tais processos de criação e gestão dos documentos, que estão mais vinculados com a área administrativa, para que os mesmos sejam automatizados e inovados. Fazendo com que tanto o trabalho passe por exigir menos dos funcionários, quanto a logística para criação e manutenção dos mesmos seja facilitada. Visando, também, reduzir a quantidade tanto de papel quanto de tinta gastos para a impressão, ao fazer com que a visualização final do documento possa ser em formato PDF, diminuindo os gastos e o espaço físico necessário para o armazenamento de documentos e consequentes papéis de rascunho (gerados por alterações após impressão do documento, onde a validade do mesmo é perdida e seu uso passa a ser apenas para rascunho).

Assim sendo, o desenvolvimento visa criar um software específico às necessidades do cliente, atendendo tanto a requisições por parte dos usuários finais, quanto a regulamentação interna, e externa, da instituição; tendo o uso, junto do compartilhamento, em tempo real (utilizando-se de um banco de dados) para que os usuários possam realizar alterações a bel-prazer, permitindo editar, imprimir, criar, excluir e visualizar os documentos sem necessitar de meios, externos ao software em si, para realizar tais ações. Logo, além de agilizar os processos de criação e gestão dos documentos, a economia quanto a impressões e tempo é igualmente visada.

Para realizar tal proposta, além da aplicação da engenharia de software como um todo, conta-se também com a programação em si. O processo de desenvolvimento parte de entrevistas com os usuários finais, assim como os clientes, e do entendimento (no possível) das normas internas, visando a realização do levantamento de requisitos, em todas as suas formas (requisitos de usuário, requisitos de domínio, requisitos funcionais, requisitos não funcionais etc). Chegando, então, na modelagem em UML (*Unified Modeling Language*), fazendo uso dos diagramas: sequência, caso de uso, classe e atividade. Fazendo uso, finalmente, da metodologia de desenvolvimento Agile, com o framework Scrum; unidos com as ferramentas específicas para o desenvolvimento, manutenção, e escalação do software.

## 2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo aborda conceitos relativos ao desenvolvimento (métodos e tecnologias), programas e linguagens utilizadas no desenvolvimento, ambiente de produção, ambiente de manutenção e engenharia de software de modo geral.

### 2.1 Modelo de Desenvolvimento

O ambiente de desenvolvimento visa criar um framework completamente novo, sem utilizar-se de recursos externos, como códigos, para sua criação; possibilitando maior customização pelos *stakeholders*, assim como maior leveza pela especificação do *front-end*. Junto de variadas formas de desenvolvimento manutenção do software; sem prender-se em softwares pagos.

Dentre as ferramentas, e metodologias, utilizadas durante o desenvolvimento, estão:

- **Notepad++ e Visual Studio Code:** Editores de texto usados para a programação geral do software. Tanto dos arquivos em PHP, XHTML e CSS quanto do arquivo, de texto, para armazenamento das queries SQL utilizadas. Fazendo uso destes editores simples e leves, sem a necessidade de uma IDE específica, onde as alterações são verificadas por meio do *output* final nos navegadores utilizados (HO, Don, 2018. Microsoft, 2018);
- **Linux, Apache, MySQL e PHP (LAMP server):** Alpine Linux 3.6.2 (Alpine Linux Development Team, 2018) para o Sistema Operacional, utilizado na máquina virtual onde o servidor se encontra, este que utiliza um servidor HTTP Apache (The Apache Software Foundation, 2018) utilizando-se da versão 2.4.10, junto de MySQL (Oracle Corporation and/or its affiliates, 2018) para o banco de dados e PHP para realização do *back-end*;

- **PHP:** PHP de versão 7.1.9 no servidor LAMP, com os recursos de sistema habilitados para uso do PDO (The PHP group, 2018) para o tratamento das queries. Tendo, durante a fase de desenvolvimento, os logs de erros ativados no php.ini (arquivo de configuração da linguagem), possibilitando, assim, que os erros tenham seu output diretamente na página, específica, do navegador utilizado;
- **Filezilla:** O software Filezilla é utilizado para a realização de update dos arquivos gerados, durante o desenvolvimento, para o servidor; isto quando estes não forem criados, alterados, ou salvos diretamente na pasta htdocs do server LAMP; respeitando os privilégios de acesso dos usuários, em conformidade com a segurança imposta pela própria organização (demais informações de segurança no capítulo 8). O Filezilla disponibiliza, ainda, a possibilidade de se fazer download dos arquivos após conectar-se ao servidor, auxiliando, assim, na disponibilização remota de tais arquivos, tanto para edição, quanto para o gerenciamento em caso de exclusões e backups. A versão utilizada do Filezilla é a 3.27.1. (KOOSE, Tim, 2018);
- **phpMyAdmin:** Ferramenta de gerenciamento de banco de dados, criada em PHP. Utilizada para criação do banco de dados, das tabelas e para executar as demais queries SQL utilizadas. Fazendo uso da versão 4.2. e tendo como foco de utilização o acesso e gerenciamento do banco de dados por meio remoto, via browsers. Assim como a amostragem dos dados, alterações rápidas na estrutura e visualização das relações entre as tabelas. (phpMyAdmin contributors, 2018);
- **HTML5:** Utilização da HTML (Hyper Text Markup Language) 5 para construção do software. Realizando, assim, a validação de todo o código para o modelo de maior portabilidade, e padronização, entre os navegadores (W3C, 2018);
- **XAMPP (Cross-platform, Apache, MariaDB, PHP and PERL) 3.2.2:** O programa XAMPP tem por objetivo iniciar um servidor apache local para uso do

PHP - linguagem que funciona no back-end – quando não for possível utilizar diretamente um servidor próprio. Fazendo com que seu uso seja mais focado para manutenções, e testes, em qualquer lugar que se tenha acesso aos arquivos do software. Assim sendo, o XAMPP acaba por ser utilizado em emergências e como um servidor de testes antes que qualquer grande alteração seja realizada no sistema (Apache Friends, 2018);

- **CSS (*Cascading Style Sheet*) 3:** CSS para o design geral das páginas da web. Seguindo os novos padrões de formatação e utilizando-se de classes CSS invocadas dentro das *tags* HTML necessárias. Não utilizando, segundo os padrões HTML5, de código CSS dentro do código HTML, mas sim de um arquivo CSS contendo todas as configurações e classes (W3C, 2018);
- **MySQL Workbench 5.2 CE:** Software para criação de bancos de dados, queries SQL e gerência de acesso aos bancos de dados em geral. Utilizado, no entanto, apenas para geração automática do diagrama MREL a partir das queries já existentes de SQL, usadas na criação das tabelas (Oracle Corporation and/or affiliates, 2018);
- **FPDF (*Free PDF*):** “FPDF is a PHP class which allows to generate PDF files with pure PHP [...]” (FPDF Library, 2018). Biblioteca PHP para geração de documentos no formato PDF (*Portable Document Format*), facilitando a edição dos documentos para o padrão já utilizado na instituição, assim como a edição dos mesmos, em qualquer momento, através do software;
- **Astah Community version 6.9:** “[...] an UML editor integrated with Mind Mapping features [...]” (Hiranabe, Kenji, 2018). Software para criação e edição dos diagramas UML de caso de uso, sequência, classe e atividade. Utilizando a versão 6.9, por conta das permissões comerciais;

- **InnoDB:** “InnoDB is a general-purpose storage engine that balances high reliability and high performance.” (Oracle Corporation and/or its affiliates, 2018). Padrão para armazenamento do banco de dados utilizado, junto do formato das tabelas criadas pelo mesmo;
- **Agile:** Metodologia de desenvolvimento de software criada após a fusão de diversas outras metodologias mais antigas. Possui, nos meios de desenvolvimento, diversos *frameworks*. É adotada em grande parte das empresas de engenharia e desenvolvimento de software. Tendo, também, aplicação no meio dos projetos em si, e não somente em desenvolvimento de software (WELLS, Don, 2009);
- **Scrum:** *Framework* de desenvolvimento da metodologia Agile, utilizada em mais de 50% dos projetos que tem por base a Agile (Scrum.org, 2018).

## 2.2 Metodologia de desenvolvimento e engenharia de software

A metodologia de desenvolvimento utilizada foi a Agile, junto do *framework* de desenvolvimento Scrum. Focando em entregar o software funcional para o cliente o quanto antes, assim como acompanhamento do cliente no desenvolvimento em si, sempre realizando as alterações necessárias com base na análise do cliente. Com o uso de *sprints* semanais, onde as reuniões com os clientes, e usuários, davam-se nas segundas-feiras (ou conforme disponibilidade dos mesmos), o software foi desenvolvido com a capacidade de escalonamento de funções, podendo englobar outros tipos de protocolos, assim como funções de gerenciamento. Por meio do *framework* Scrum, foi possível disponibilizar versões funcionais do software para uso do cliente, mesmo que alguma parte ainda não estivesse completa (como *back-end* funcional, mas *front-end* incompleto), fazendo com que estivesse utilizável, mesmo não atendendo aos requisitos não funcionais.

## 2.2.1 Levantamento de requisitos

Levantamento de todos os tipos de requisitos, seja por meio de entrevista com o cliente e o usuário final, ou mesmo por análise do ambiente interno, seu regimento, normas a serem seguidas e demais demandas padrão, específicas da empresa em si.

### 2.2.1.1 Requisitos funcionais

“[...] os requisitos funcionais descrevem a função do sistema detalhadamente, suas entradas e saídas, exceções etc.” (SOMMERVILLE, 2007, p. 81).

Requisitos ligados, diretamente, a completa e total funcionalidade do software. Sendo definidos em conformidade com as solicitações e requisições dos usuários. Tais requisitos são de suma importância, afetando o usuário final e o funcionamento correto do sistema; do mesmo modo que estabelecem as medidas para avaliação do software pelos *stakeholders*.

#### 2.2.1.1.1 Requisitos de domínio

“[...] são derivados do domínio de aplicação do sistema, em vez das necessidades específicas dos usuários do sistema” (SOMMERVILLE, 2007, p. 83).

Requisitos referentes ao domínio de operação do software, como as partes mais específicas.



#### 2.2.1.1.2 Requisitos de usuário

Especificação dos requisitos funcionais e não funcionais voltados para o usuário final, de maneira compreensiva. “[...] devem descrever os requisitos funcionais e não funcionais, de modo que eles sejam compreensíveis pelos usuários do sistema que não possuem conhecimento técnico detalhado.” (SOMMERVILLE, 2007, p. 85).

##### 2.2.1.1.2.1 Requisitos de sistema

Requisitos específicos sobre as principais funcionalidades do sistema, transformando os requisitos de usuário numa linguagem mais formal e técnica. “[...] os requisitos de sistema devem simplesmente descrever o comportamento externo do sistema e suas restrições operacionais.” (SOMMERVILLE, 2007, p. 87).

##### 2.2.1.2 Requisitos não funcionais

“[...] são aqueles não diretamente relacionados às funções específicas fornecidas pelo sistema.” (SOMMERVILLE, 2007, p. 82).

Requisitos que não interferem diretamente, ou significativamente, na funcionalidade, ou operabilidade, do sistema em sua ideia inicial; mas sim que fornecem certas características, e padrões, que devem ser seguidos, seja para atender as requisições do cliente, seja para o desenvolvimento, ou mesmo para adequar-se às características do meio no qual ele está inserido. Demais requisitos no escopo são:

#### 2.2.1.2.1 Requisitos de produto

Requisitos que definem as características físicas, como operabilidade, velocidade e desempenho geral ao se criar, editar e salvar arquivos. “[...] especificam o comportamento do produto” (SOMMERVILLE, 2007, p. 83).

#### 2.2.1.2.2 Requisitos organizacionais

“[...] requisitos derivados de políticas e procedimentos da organização do cliente e do desenvolvedor.” (SOMMERVILLE, 2007, p. 83).

Requisitos relativos as regras internas da organização onde o software será utilizado, adaptando o mesmo para seguir os cronogramas e processos internos. Do mesmo modo que visa propor os prazos de entrega tanto para a documentação, quanto para o produto final.

#### 2.2.1.2.3 Requisitos externos

“[...] abrange todos os requisitos derivados de fatores externos ao sistema e seu processo de desenvolvimento.” (SOMMERVILLE, 2007, p. 83).

Requisitos referentes as demais necessidades, mas fora do escopo do software e do desenvolvimento do mesmo; englobando conceitos legislativos, jurídicos e geográficos.

### 2.2.2 *UML: Diagramas e introdução*

Quatro diagramas principais foram utilizados para a modelagem do software, sendo eles: diagrama de caso de uso, diagrama de classe, diagrama de atividades e

diagrama de sequência. Os demais diagramas UML (oito, além dos quatro citados) encontram-se em anexo.

- **Diagrama de Caso de uso:** Diagrama para representação dos requisitos funcionais, como eles interagem entre si, suas dependências e relações com os atores. Como descrito pelo Object Management Group (OMG): “[...] describe a set of actions (use cases) that some system or systems (subject) should or can perform in collaboration with one or more **external users** of the system (actors).” (OMG™ Unified Modeling Language™, 2018);
- **Diagrama de Classe:** Diagrama que mostra uma estrutura mais detalhada do diagrama de caso de uso, focando mais no esclarecimento técnico voltado para o desenvolvimento do software em si. Mostrando, assim, os relacionamentos dentre as classes e entidades, de modo que seria feito no software. Como explicado pelo Object Management Group (OMG): “shows structure of the designed system at the level of classes and interfaces, shows their features, constraints and relationships - associations, generalizations, dependencies, etc.” (OMG™ Unified Modeling Language™, 2018);
- **Diagrama de Atividade:** Descreve as ações, divididas em partes, realizadas no diagrama de caso de uso. Segundo descrição do Object Management Group: “shows **flow of control** or **object flow** with emphasis on the sequence and conditions of the flow.” (OMG™ Unified Modeling Language™, 2018);
- **Diagrama de Sequência:** Diagrama para descrição das sequências de mensagens entre instâncias, classes, componentes e atores. Focado na visualização dos eventos que ocorrem durante a utilização do software. Como ditto pelo Object Management Group: “most common kind of interaction diagram, which focuses on the message interchange between a number of lifelines.” (OMG™ Unified Modeling Language™, 2018);

### 2.3 Ambiente de operação

O desenvolvimento do software é realizado utilizando-se das ferramentas citadas anteriormente; no constante que, como todas possuem suporte para diversos sistemas operacionais, assim como licença gratuita, fica ao critério do desenvolvedor a escolha de uma plataforma de desenvolvimento. Prática, esta, que visa gerar melhor operabilidade no software, já que o mesmo é desenvolvido, e testado, nas diversas versões do sistema operacional Windows (Windows 7, Windows 8.1, Windows 10 etc.), além de ser testado nos vários navegadores de internet existentes, tais como:

- Mozilla Firefox (Mozilla.org contributors, 1998-2018);
- Google Chrome (Google, 2018);
- Opera (Opera software, 2018);
- Vivaldi (Vivaldi Technologies™, 2015-2018);
- Internet Explorer (Microsoft, 2018).

Ademais, o servidor principal de host para o software será mantido na distribuição Alpine Linux 3.6.2 (Alpine Linux Development Team, 2018), utilizando-se de um banco de dados de 10MB (dez megabytes) inicialmente.

### 2.4 Ambiente de manutenção

Assim como o ambiente de operação, o sistema operacional base para manutenção poderá ser qualquer distribuição Linux ou Windows, a partir do Windows 7, por conta de conexão remota com o servidor. Em caso de pequenas manutenções, para alteração dos arquivos da página web, ou dos arquivos de configuração do servidor de hospedagem. Qualquer outro tipo de manutenção que exija maior atenção ocorrerá diretamente no servidor, por meio de *CLI*.

## 2.5 Design

*Front-end* geral do software, tendo como base os requisitos não funcionais dos usuários e clientes, assim como os requisitos de domínio, que acabam por fazer um papel por conta das normas internas da instituição.

### 2.5.1 UI (User interface)

Componentes do *front-end* que fazem parte do software, sua distribuição e intuição geral de uso. Ademais, uma lista dos tipos de componentes do tipo *input*, *output*.

#### 2.5.1.1 Input

Componentes do software para entrada de dados. Sendo eles:

- **Campos de texto:** Designados para a entrada de letras, números e símbolos. Utilizados no programa para realizar pesquisas, inserção dos dados de protocolo, alteração dos dados de protocolo;
- **Campo de senha:** Campo de texto único, utilizado para realizar *logon* tanto dos usuários, quanto do administrador. Possui codificação própria da linguagem HTML para não mostrar os caracteres que estão sendo digitados;
- **Botões:** Botões utilizados para converter as informações fornecidas pelos usuários, por meio dos campos de texto, para dados no software. Possibilitando futuros *outputs*;
- **Checkboxes:** Caixas de seleção que permitem que o usuário informe dados baseando-se nos textos (*outputs*) fornecidos pelo próprio programa. Aceitam diversos valores de *input* para conversão;

- **Select - Option:** Círculos de seleção, com funcionalidade semelhante as *checkboxes*, porém aceitando apenas um valor de *input*.

#### 2.5.1.2 Output

Componentes do software para saída de dados. Sendo eles:

- **Campos de texto estáticos:** Campos de texto que não podem ser alterados pelo usuário; sendo, muitas vezes, gerados a partir do *input* fornecido pelo mesmo, durante a etapa inserção dos dados de documentos;
- **Tabelas:** Tabelas com as informações de dados no sistema. Tendo o *output* chamado após uma pesquisa, ou por padrão da página específica que o usuário acessou;
- **Menus e scrollbars:** Botões e design geral do software, que visam a navegação fluída do usuário durante o uso.

#### 2.5.2 UX (User experience)

Design do software voltado para as necessidades funcionais do cliente, junto das preferências dos usuários. Em outros termos: *“UX design is the art and science of generating positive emotions among people who interact with products or services.”* (SHARON, Tomer. 2013).

O processo de UX foi realizado com acompanhamento direto do cliente/usuário, seguindo a simplicidade pedida, junto da usabilidade requisitada. Usando, como ferramentas para amostra, esboços conforme descrição dada pelo usuário/cliente.

## 2.6 Validação

Avaliação do software por parte dos usuários finais, após o mesmo já estar funcionando no ambiente de operação. Tema abordado, mais profundamente, nos dois últimos capítulos. Baseia-se, basicamente, na verificação dos requisitos funcionais, e dos não funcionais, por parte do usuário ao utilizar o software (em seu funcionamento) em si.

## 2.7 Segurança

Informações gerais relacionadas ao tratamento de erros e quanto ao acesso ao software em si. Baseando-se no *input*, *output* e demais informações geridas pelo software; assim como seus métodos de controle e registro das ações realizadas no software pelo usuário.

### 2.7.1 *Tratamento de exceções*

Especificação das partes do programa que possuem código mais delicado, utilizando-se de um log de erros para casos críticos. O usuário é, desta forma, informado, de maneira compreensível, sobre o ocorrido e o erro é enviado para os desenvolvedores e tratado logo após.

### 2.7.2 *Variáveis globais e visibilidade das variáveis*

Por se tratar de um software para web, a entrada de dados por parte dos usuários deve ser tratada de forma adequada, por conta dos inúmeros riscos existentes com dados vindos de locais cujo acesso pode ser remoto. Desta forma, as variáveis globais usadas para tratamento de tais dados são criadas somente durante o chamado PHP, recebem um tipo específico para que não fujam de seu

parâmetro de dados (String, int, date) e, logo após seu uso, são esvaziadas e destruídas.

Por padrão de acesso, as variáveis são instanciadas como *public*. Tais variáveis não são utilizadas fora do seu local de criação, de forma que a segurança quanto ao acesso, visualização e edição dos dados contidos nas mesmas fica por parte do *reset* em seu valor logo após ser utilizada, assim como o valor é passado para variáveis globais temporárias, estas que são destruídas após o uso, fazendo com que, deste jeito, exista maior restrição para quanto ao acesso, ou mesmo alteração, de tais variáveis. Do mesmo modo que as constantes de conexão são puramente estáticas e não podem ser alteradas sem uma edição direta do código, por parte do acesso de seu arquivo no servidor onde está armazenada.

### *2.7.3 Acesso ao software*

Todas as páginas do software têm seu acesso inicial restrito, tendo apenas a página de login com seu acesso permitido. Impedindo, assim, a navegação, ou visualização, das demais páginas do software. Deste modo, somente após a confirmação do usuário, e sua respectiva senha, é que a página inicial, junto das demais, passa a ter seu acesso liberado. Tais acessos são perdidos ao se sair da página do software, de qualquer maneira.

## **2.8 Banco de dados**

Informações referentes ao banco de dados e armazenamento geral do software. Contendo o tamanho disponível de armazenamento, escalabilidade e demais informações.



### *2.8.1 Queries SQL*

*Querys* SQL utilizadas para criação e gerência do banco de dados e suas demais funcionalidades, quanto a criação de relatórios referentes as ações realizadas no mesmo.

### *2.8.2 Dicionário de dados*

Tabelas que expressam as tabelas existentes no banco de dados. Contendo os tipos dos campos, tamanho dos campos e a relação da tabela com as demais; assim como as restrições de cada campo existente.

### *2.8.3 Modelo relacional de entidade lógica*

Diagrama que expressa a relação das tabelas, sendo uma abstração do dicionário de dados (ou do banco de dados em si).

## **2.9 Testes de software**

“O sistema é tratado como uma caixa-preta, cujo comportamento pode ser somente determinado por meio de estudo de suas entradas e saídas relacionadas.” (SOMMERVILLE, 2007, p. 359). Testes de entrada e saída dos dados no sistema, tendo certos quesitos que devem ser atendidos.

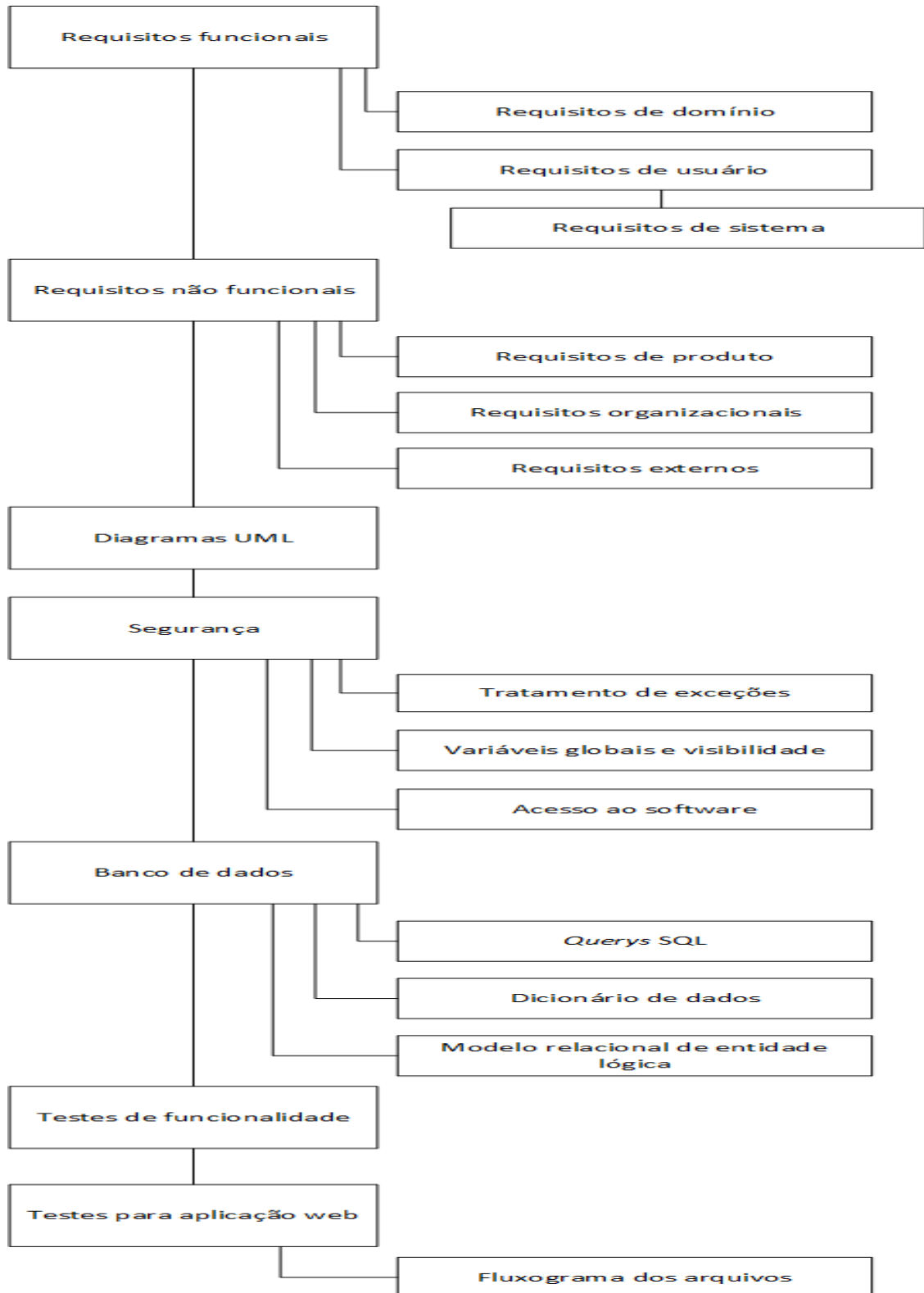
## **2.10 Testes para aplicações web e fluxogramas da aplicação**

“O teste dinâmico permite a revelação de defeitos relacionados aos fluxos de controle e de dados entre as páginas da aplicação *web*.” (Introdução ao Teste de Software, 2007, p. 218). Testes com o intuito de verificar e validar o acesso das

páginas do software, e como as mesmas interagem entre si, em termos de navegação.

### 3 DESENVOLVIMENTO

Figura 1 – Fluxograma do capítulo



Fonte: Autoria própria.

### 3.1 Requisitos funcionais

- Criar protocolos;
- Salvar protocolos no banco de dados;
- Alterar e atualizar protocolos;
- Salvar alterações;
- Excluir protocolos;
- Acompanhamento de protocolos por tipo;
- Acompanhamento de protocolos por departamento ou curso;
- Acompanhamento de protocolos por professor ou responsável;
- Possibilidade de conversão dos dados de acompanhamento para PDF;
- Converter protocolos para .PDF;
- Converter protocolos para .DOCX;
- Converter protocolos para .XLSX;
- Registrar todas as ações feitas nos protocolos em uma tabela de logs do banco de dados, contendo: ação, usuário, horário, data e ID do documento;
- Conta de administrador, estática, para gerenciamento e visualização dos logs gerados;
- Verificação dos dados inseridos pelo usuário conforme as normas da escola;
- Verificação das datas e horários inseridos conforme disponibilidade de horários de serviço;
- Pesquisa dos protocolos com *status* aberto;
- Pesquisa dos protocolos com *status* finalizados;
- Acesso aos protocolos por meio de uma rede interna, sem dependência com provedores de internet para a utilização do software;
- Log out do sistema que finalize a sessão e bloqueie o acesso ao software;
- Criação do protocolo somente com todos os campos necessários preenchidos;
- Os protocolos de HAE precisam seguir três diferentes tipos. Sendo eles: Estágio Supervisionado, Projeto de Graduação (TCC/TG) e Projeto livre;
- As HAEs do tipo projeto livre necessitam de ter uma breve descrição de seu tipo, além da descrição das atividades.

### 3.1.1 *Requisitos de domínio*

- O software necessita de ter verificação específica de caracteres e extensão de caracteres para os campos que são, de certo modo, vitais para a criação, validação e manutenção dos protocolos gerados, como: telefone aceitando apenas números, RA dos alunos tem um máximo de caracteres, ID dos professores e funcionários seguindo as especificações da empresa, datas e horários apenas em formato padrão;
- As HAEs precisam seguir datas específicas, relativas a duração do semestre, os horários de funcionamento da instituição e a quantidade máxima e mínima de HAEs;
- O software necessita de ter um registro com as ações realizadas (criar, alterar e excluir), associando as mesmas aos funcionários e aos protocolos que sofreram tais ações, por meio do ID de ambos.

### 3.1.2 *Requisitos de usuário*

- O programa deve criar protocolos de estágio e HAE;
- Os protocolos podem ser criados, editados, impressos, visualizados e excluídos;
- A cor do software deve seguir o padrão oficial de cores da instituição (cinza e vermelho);
- Os protocolos finalizados devem estar liberados para conversão em PDF;
- O PDF gerado precisa estar disponível para impressão;
- O software deve funcionar na grande maioria dos navegadores;
- O PDF gerado deverá seguir os padrões de formatação já utilizados pela instituição nos demais documentos;
- Pesquisa dos protocolos em aberto;
- Pesquisa dos protocolos fechados;
- Filtragem dos dados de acompanhamento por setor, professor e tipo do protocolo;
- Pesquisa dos dados de acompanhamento.

### 3.1.2.1 Requisitos de sistemas

- **Criar protocolos:**

Tabela 1 – Requisitos para criação de protocolos.

<b>Função</b>	Adicionar protocolos ao banco de dados.
<b>Descrição</b>	Adiciona um protocolo de tipo escolhido pelo usuário e o adiciona ao banco de dados do sistema.
<b>Entradas</b>	Dados referentes ao protocolo, como data, nomes, número de documentos.
<b>Origem</b>	Dados específicos ao protocolo selecionado pelo usuário. O ID é gerado, automaticamente, banco de dados e a data utilizada para os logs é fornecida pelo sistema operacional.
<b>Saídas</b>	Um protocolo pronto, com os campos preenchidos conforme as informações de entrada do usuário.
<b>Destino</b>	Banco de dados do sistema, após protocolo completamente preenchido.
<b>Requer</b>	Forma padrão pré-estabelecida de protocolo.
<b>Pré-condição</b>	Exibição dos formulários com os campos nomeados seguindo o padrão.
<b>Pós-condição</b>	O protocolo pode ser alterado após completado e salvo no banco de dados.
<b>Efeitos colaterais</b>	Geração de uma entrada na tabela de relatórios após a ação ser concluída.

Fonte: Autoria própria.

- **Alterar protocolos.**

Tabela 2 – Requisitos para alteração de protocolos.

<b>Função</b>	Alterar protocolos no banco de dados.
<b>Descrição</b>	Altera um protocolo, de tipo escolhido pelo usuário, e realiza a atualização do mesmo no banco de dados.
<b>Entradas</b>	Dados referentes ao protocolo, como data, nomes, número de documentos etc.
<b>Origem</b>	Dados específicos ao protocolo selecionado pelo usuário. O ID não sofre alterações e, assim como a criação, a data é fornecida automaticamente pelo sistema.
<b>Saídas</b>	A alteração dos dados do protocolo selecionado.
<b>Destino</b>	Banco de dados do sistema, após protocolo alterado ou não.
<b>Requer</b>	Forma padrão pré-estabelecida de protocolos e disponibilidade do documento.
<b>Pré-condição</b>	Exibição dos formulários com os campos nomeados, seguindo o padrão, e preenchidos com as informações do protocolo em específico, para alteração.
<b>Pós-condição</b>	O protocolo não pode ser alterado após completado e salvo no banco de dados; com a exceção de alguns protocolos específicos. Estando aberto apenas para visualização e download (formato PDF).
<b>Efeitos colaterais</b>	Geração de uma entrada na tabela de relatórios após a ação ser concluída.

Fonte: Autoria própria.



- **Excluir protocolos.**

Tabela 3 – Requisitos para excluir protocolos.

<b>Função</b>	Excluir protocolos do banco de dados.
<b>Descrição</b>	Excluir um protocolo, de tipo escolhido pelo usuário, do banco de dados.
<b>Entradas</b>	Nenhuma
<b>Origem</b>	Protocolo no banco de dados.
<b>Saídas</b>	Remoção do protocolo do banco de dados.
<b>Destino</b>	Nenhum.
<b>Requer</b>	Disponibilidade(existir) do protocolo no banco de dados.
<b>Pré-condição</b>	O protocolo ter sido criado.
<b>Pós-condição</b>	Nenhuma
<b>Efeitos colaterais</b>	Geração de uma entrada na tabela de relatórios após a ação ser concluída.

Fonte: Autoria própria.

- **Exibir protocolos.**

Tabela 4 – Requisitos para exibir protocolos.

<b>Função</b>	Exibir os protocolos cadastrados.
<b>Descrição</b>	Seleciona um protocolo, já criado, no banco de dados e exibe suas informações na tela do browser, para o usuário.
<b>Entradas</b>	Tipo de protocolo para busca.
<b>Origem</b>	Protocolo no banco de dados.
<b>Saídas</b>	Exibição de todas as informações do protocolo.
<b>Destino</b>	Tela de exibição.
<b>Requer</b>	Disponibilidade(existir) do protocolo no banco de dados.
<b>Pré-condição</b>	O protocolo ter sido criado.
<b>Pós-condição</b>	Exibir os dados de maneira editável, ou consulta.
<b>Efeitos colaterais</b>	Nenhum.

Fonte: Autoria própria.

- Gerar PDF do protocolo.

Tabela 5 – Requisitos para gerar PDF dos protocolos.

<b>Função</b>	Converter os dados do banco para o formato de visualização PDF.
<b>Descrição</b>	Seleciona um protocolo, já criado, no banco de dados e realiza a conversão do mesmo para PDF. Seguindo os padrões de formatação da instituição.
<b>Entradas</b>	Tipo de protocolo para busca.
<b>Origem</b>	Protocolo no banco de dados.
<b>Saídas</b>	Protocolo PDF com as informações do protocolo
<b>Destino</b>	Visualização online ou computador do usuário, por meio de download.
<b>Requer</b>	Disponibilidade (existir) do protocolo no banco de dados.
<b>Pré-condição</b>	O protocolo ter sido criado.
<b>Pós-condição</b>	Disponibilidade para visualização e download.
<b>Efeitos colaterais</b>	Nenhum.

Fonte: Autoria própria.

- Realizar *log on*.

Tabela 6 – Requisitos para acessar o sistema.

<b>Função</b>	Fazer o log in do usuário.
<b>Descrição</b>	Verifica a validade do usuário e senha fornecido pelo usuário, e permite o acesso ao software caso ambos sejam aceitos, redirecionando para a tela inicial.
<b>Entradas</b>	Nome de usuário e a senha.
<b>Origem</b>	Página PHP.
<b>Saídas</b>	Confirmação, ou falha, de login e redirecionamento específico.
<b>Destino</b>	Página inicial, caso o nome de usuário e senha estejam corretos; ou página de login caso incorretos.
<b>Requer</b>	Conexão com o banco de dados para confirmação dos dados de log on do usuário.
<b>Pré-condição</b>	O usuário estar cadastrado.
<b>Pós-condição</b>	Permitir acesso ao software, ou pedir as informações de login novamente.
<b>Efeitos colaterais</b>	Acesso ao software, caso login válido.

Fonte: Autoria própria.

- Realizar *log out*.

Tabela 7 – Requisitos para sair do sistema.

<b>Função</b>	Fazer o log out do usuário.
<b>Descrição</b>	Elimina a sessão do usuário e redireciona para a página de log on.
<b>Entradas</b>	Dados da sessão do usuário.
<b>Origem</b>	Sessão do navegador.
<b>Saídas</b>	Eliminação da sessão atual, junto dos seus dados específicos.
<b>Destino</b>	Página de log on do sistema.
<b>Requer</b>	Código de <i>reset</i> da sessão.
<b>Pré-condição</b>	O usuário estar conectado e a sessão não ter expirado.
<b>Pós-condição</b>	Nova sessão liberada para uso.
<b>Efeitos colaterais</b>	O usuário deverá realizar log on novamente para acessar o sistema.

Fonte: Autoria própria.

- **Alimentar tabela de log.**

Tabela 8 – Requisitos para gravação dos dados de log.

<b>Função</b>	Gravar as ações feitas pelos usuários, automaticamente, em uma tabela de logs.
<b>Descrição</b>	As ações de criar, alterar e excluir geram entradas na tabela de logs.
<b>Entradas</b>	Ação realizada, data e informações do usuário que realizou tais ações, como nome, log on e ID.
<b>Origem</b>	Banco de dados.
<b>Saídas</b>	Tabela com informações que podem ser visualizadas e impressas.
<b>Destino</b>	Computador do usuário (formato PDF), ou visualização online.
<b>Requer</b>	Que a tabela não esteja vazia e que algum usuário tenha realizado alguma ação.
<b>Pré-condição</b>	Alguma ação, além de visualizar e gerar PDF, ter ocorrido.
<b>Pós-condição</b>	Armazenar e exibir os dados.
<b>Efeitos colaterais</b>	Registro.

Fonte: Autoria própria.

- Gerar PDF dos logs.

Tabela 9 – Requisitos para gerar PDF dos logs do sistema.

<b>Função</b>	Exibir os logs registrados.
<b>Descrição</b>	Seleciona um log no banco de dados e exibe suas informações no formato PDF na tela do browser apenas para o administrador.
<b>Entradas</b>	Nenhuma.
<b>Origem</b>	Log no banco de dados.
<b>Saídas</b>	Exibição das informações no log.
<b>Destino</b>	Tela de exibição.
<b>Requer</b>	Algum protocolo ter sido manipulado.
<b>Pré-condição</b>	Existência dos registros no banco de dados.
<b>Pós-condição</b>	Exibir os dados.
<b>Efeitos colaterais</b>	Nenhum.

Fonte: Autoria própria.

- **Listagem de protocolos**

Tabela 10 – Requisitos para exibir os protocolos do sistema.

<b>Função</b>	Exibe informações sobre protocolos relativos aos professores, departamentos e tipos.
<b>Descrição</b>	A quantidade de protocolos é exibida com os filtros de protocolo por curso, professore e departamento. Possibilitando controle dos mesmos.
<b>Entradas</b>	Dados dos protocolos nos bancos de dados e pesquisa do usuário.
<b>Origem</b>	Banco de dados.
<b>Saídas</b>	Informações de quantidade e conversão para PDF.
<b>Destino</b>	Tela de exibição.
<b>Requer</b>	Protocolos no banco de dados.
<b>Pré-condição</b>	Usuário de grupo com permissão.
<b>Pós-condição</b>	Exibir os dados.
<b>Efeitos colaterais</b>	Nenhum.

Fonte: Autoria própria.



## 3.2 Requisitos não funcionais

Requisitos não necessários para o funcionamento do software, mas também importantes para o uso do mesmo. Composto pelos requisitos de produto, organização e externos.

### 3.2.1 Requisitos de produto

- O software deverá ser funcional em qualquer *browser* (Firefox, Google Chrome, Vivaldi, Internet Explorer etc.). Tendo compatibilidade suficiente para que as diferenças de uso, em todos os *browsers* utilizados na instituição, sejam mínimas, se existentes;
- A criação de arquivos não deverá tomar mais do que 3 (três) segundos, o mesmo para a operação de atualizar, pesquisar e excluir tais arquivos. Fazendo, neste meio tempo, tanto a conexão com o banco de dados quanto a execução das *queries* necessárias para tais ações e, depois, retornando a página inicial do processo;
- Após sucesso no *commit* das *queries*, mostrar uma mensagem de sucesso, com tempo de redirecionamento de 3 (três) segundos para a página de início do processo em específico;
- Em caso de fracasso, realizar o *rollback* dos dados e redirecionar, após a mensagem de fracasso, para a página anterior após 3 (três) segundos;
- O tamanho total do software, somando-se todos os arquivos HTML, CSS e PHP, não ultrapassará 5 (cinco) megabytes;
- Juntando-se todos os ícones e imagens, o tamanho final do software ficará em, aproximadamente, 10 (dez) megabytes;
- A ação de log on não deverá demorar mais do que 3 (três) segundos após a inserção dos dados por parte do usuário; realizando tanto a conexão com o banco de dados, quanto a verificação do usuário, seu grupo e, por fim, a mensagem de log on válido ou inválido, redirecionando para a página inicial.

### 3.2.2 *Requisitos Organizacionais*

- Hospedagem interna (intranet), não sendo dependente de terceiros (como provedores de internet);
- Divisão do protocolo de estágio em três partes; onde a segunda parte só é acessível após o preenchimento da primeira parte, enquanto que a terceira só estará disponível após preenchimento da segunda parte. Cada parte é relacionada a um setor;
- O software deve ser entregue até o dia 31/12/2018 (trinta e um de dezembro de dois mil e dezoito);
- Restrição de acesso tanto para as informações da tabela de logs, quanto da página que exibe tais informações. Tendo um usuário estático para acesso e gerenciamento de tal página;
- As informações de log devem ter a opção de conversão para PDF, tanto para organização fora do software, quanto para impressão das mesmas;
- A documentação do software deve ser entregue até o dia 31/12/2018 (trinta e um de dezembro de dois mil e dezoito);
- Uso de um banco de dados interno, e de licença gratuita, para armazenamento dos documentos criados;
- O software requer, para seu desenvolvimento, seis horas diárias, no mínimo, durante cinco dias da semana;
- O acesso para criação de certos documentos deverá seguir o grupo/departamento do usuário;
- O acesso ao acompanhamento dos protocolos deverá ser restrito para o grupo da coordenação, direção e administração.

### 3.2.3 *Requisitos Externos*

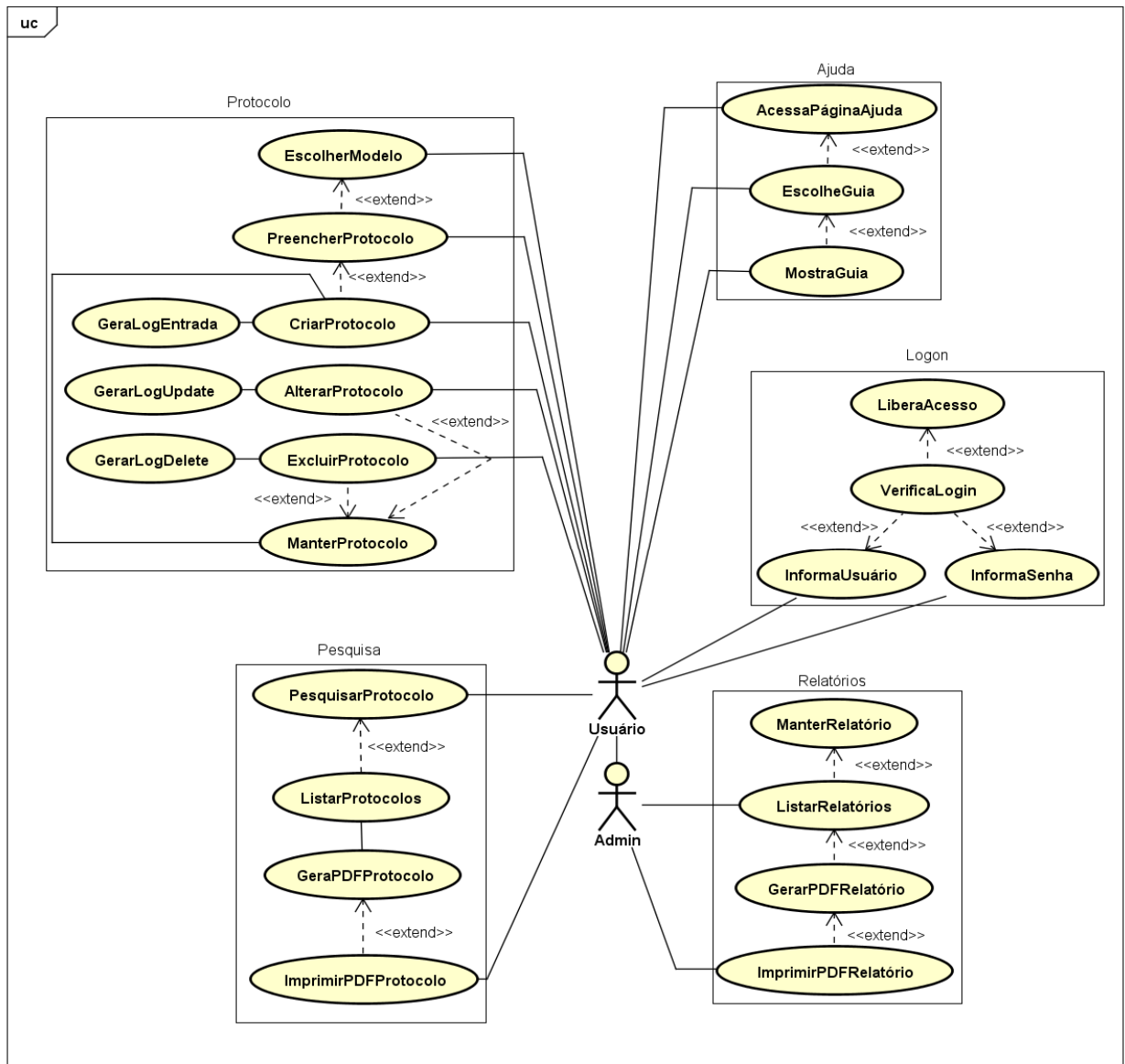
- A interface do usuário deve acompanhar a língua vernácula dos usuários finais. Português, no caso;
- Por questões legais e jurídicas, o software não pode conter código provindo de uma organização privada, que não tenha o código como aberto;
- Os padrões de auditoria do software devem acompanhar o horário local (horário de Brasília);
- A disponibilidade de todo o código do programa fica referente as legislações seguidas pela instituição;
- Os horários disponíveis para escolha devem obedecer ao horário de funcionamento comercial e, se for o caso, específico da organização, estes que seriam: 7h40min até 22h;
- Os professores não podem ultrapassar as horas específicas de trabalho por mês;
- A página de alterações feitas no banco de dados deve ter uma versão gráfica (*GUI*) para consulta.

### 3.3 Diagramas UML

Imagens dos diagramas UML criados para o desenvolvimento do software. Seguindo os padrões pré-estabelecidos pela UML (versão 2.5), para os diagramas, respectivamente, de caso de uso, classe, atividade e sequência.

- **Diagrama de Caso de Uso:**

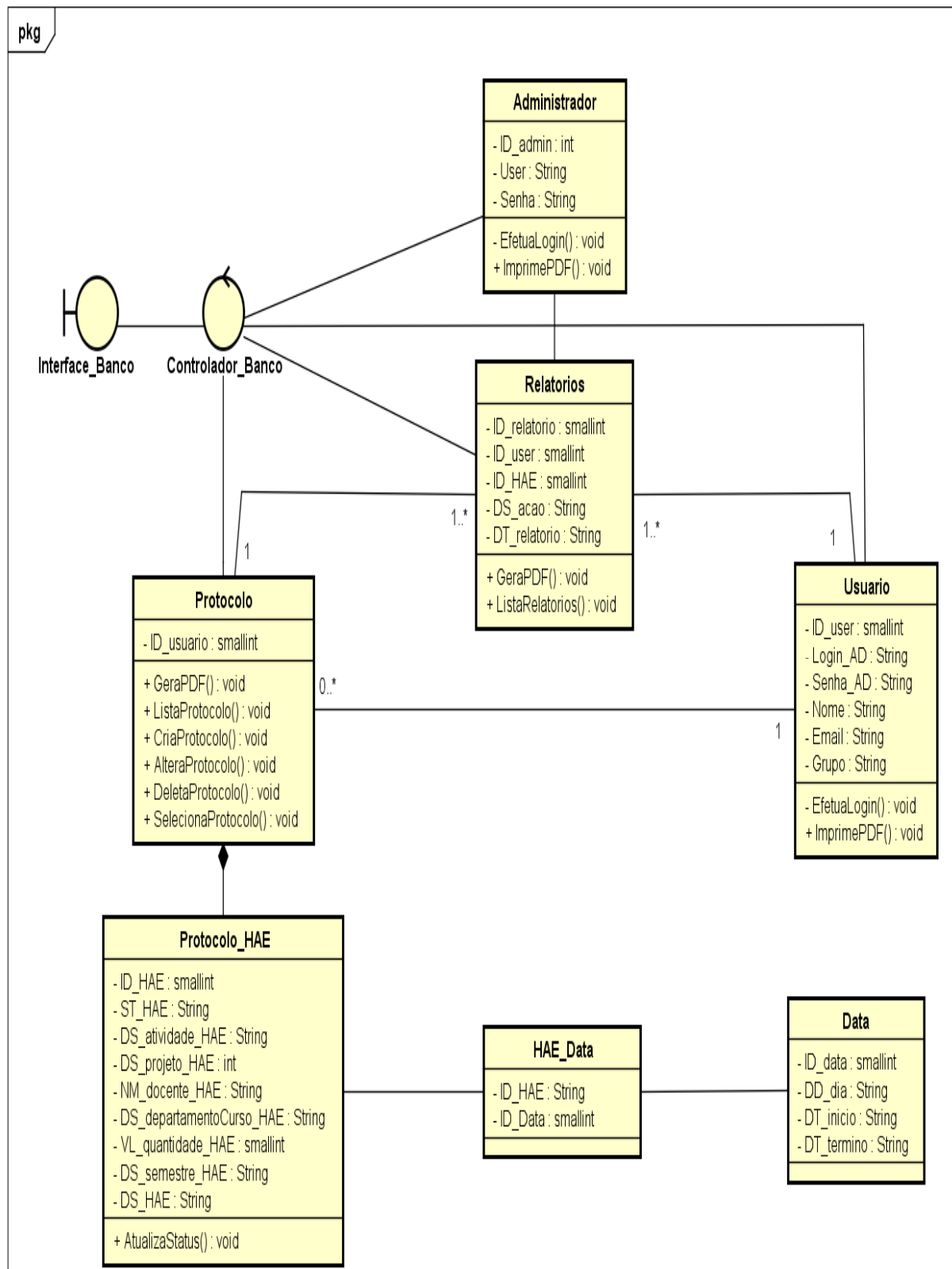
Figura 2 – Diagrama de caso de uso.



Fonte: Autoria própria.

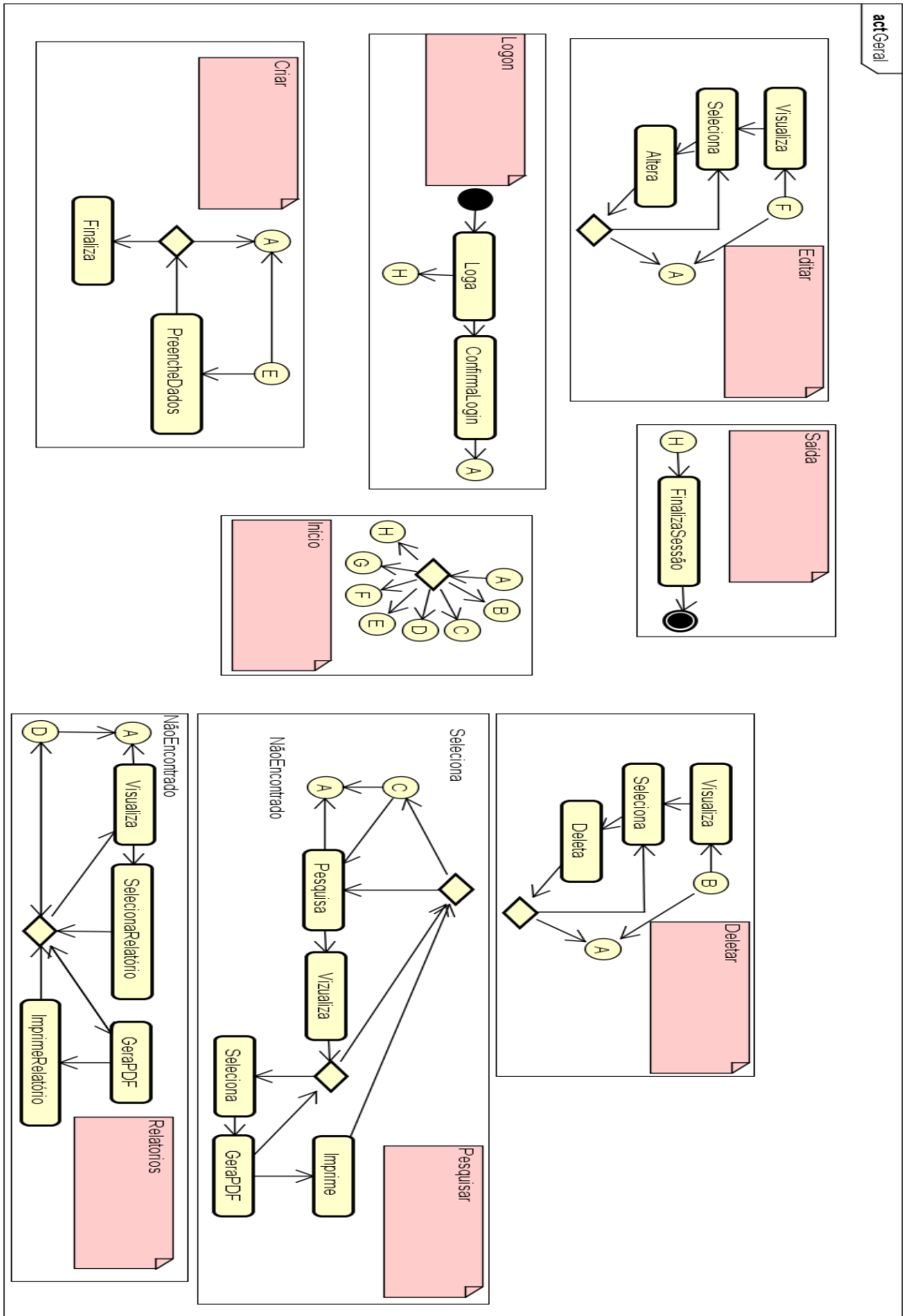
- **Diagrama de Classe:**

Figura 3 – Diagrama de classe.



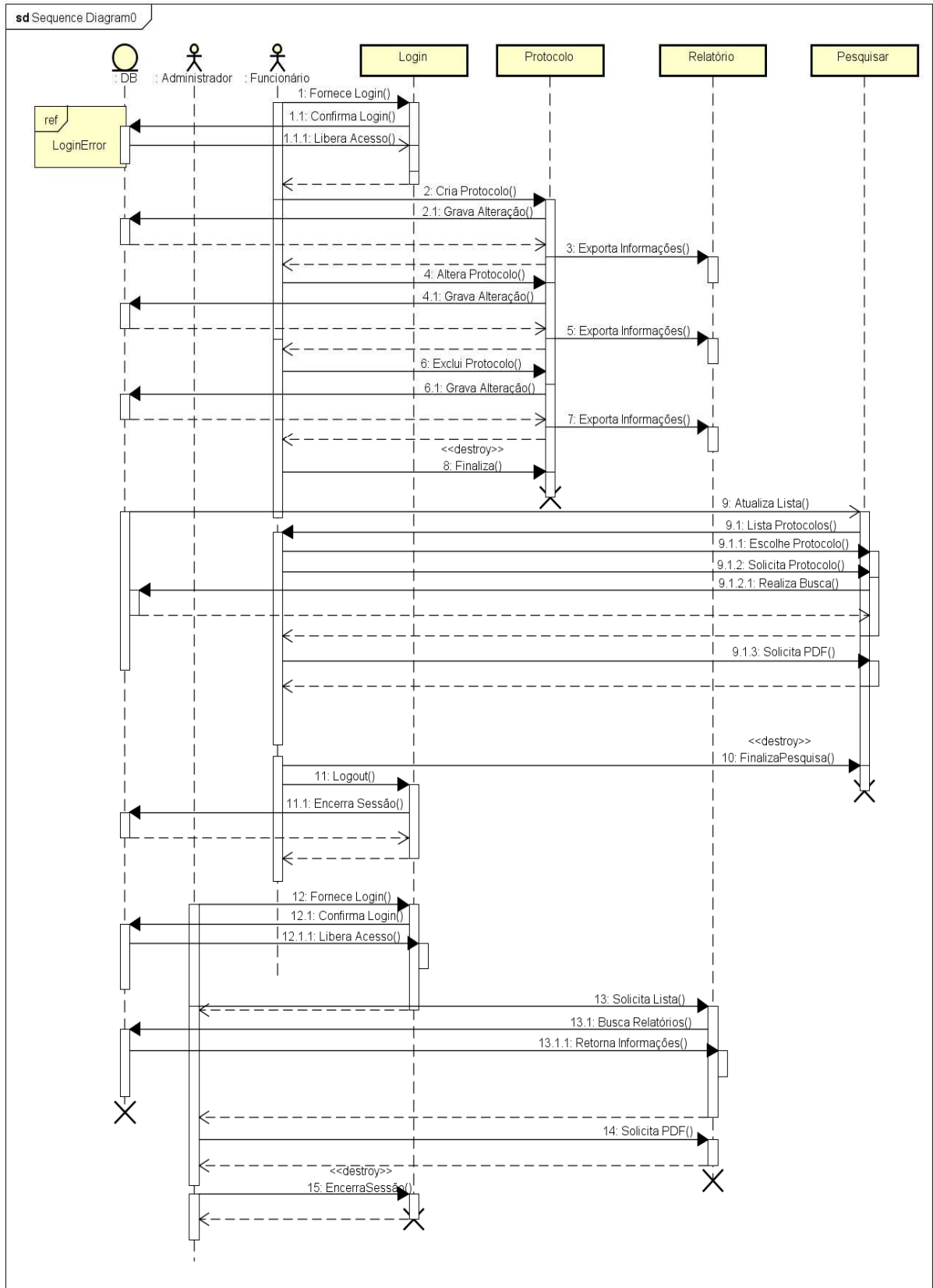
• Diagrama de Atividade:

Figura 4 – Diagrama de atividade.



• Diagrama de Sequência:

Figura 5 – Diagrama de sequência.



### 3.4 Segurança

Métodos, padrões e técnicas empregadas para manter a segurança do software. Tanto no seu desenvolvimento e implementação, quanto na sua utilização diária na instituição.

#### 3.4.1 Tratamento de exceções

O método de tratamento de exceções empregado consiste na criação de uma variável (`$errorCaught`) que armazena o tipo do erro, após o mesmo ser obtido e tratado por um método *try/catch*, e o disponibiliza para os desenvolvedores por meio de um aviso por e-mail, para que a devida manutenção seja realizada. Exibindo, também, uma mensagem mais amena do erro para o usuário. Em exemplo:

```
try{
    $connection = new PDO(servername, username, password);
    $connection->exec("set names utf8");
    $connection->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
    return $connection;
}catch(PDOException $errorCaught){
    var_dump($errorCaught);
    error_log($errorCaught, 1, Admin_Mail, "Content-Type: text/html;
charset=utf8\r\nFrom: ".Admin_Mail."\r\nTo: ".Admin_Mail);
    echo "Não conectado";
}
```

#### 3.4.2 Variáveis globais e visibilidade das variáveis

- `$stHAE->bindValue(':dia2', $dia2, PDO::PARAM_STR);` (THE PHP GROUP, 2001-2018).



Faz com que a variável `$dia2` aceite somente dados no formato String, por conta da parametrização feita pelo comando `PDO::PARAM_STR`, assim como faz a atribuição do tipo String para o tipo SQL CHAR, ou VARCHAR.

- `$_SESSION['name'] = $_POST['NM_nome_estagio'];` (THE PHP GROUP, 2001-2018).

A variável global de sessão *name* é inicializada e recebe o valor vindo de um campo preenchido pelo usuário.

- `bindValue(':name', $_SESSION['name'], PDO::PARAM_STR);` (THE PHP GROUP, 2001-2018).

O tipo String é atribuído a variável *name* como requerimento de validação, impedindo, assim, que seja utilizada com um tipo de valor diferente.

- `unset($_SESSION['name']);` (THE PHP GROUP, 2001-2018).

O método *unset()* é chamado para desinstanciar o valor atribuído as variáveis específicas, destruindo tal variável de sessão.

- `const Admin_Mail = "lifaieons@outlook.com";` (THE PHP GROUP, 2001-2018).

O tipo *const* faz com que o valor seja completamente estático (constante), de modo que a constante não possa ser alterada em momento algum após sua declaração. Utilizado nas variáveis de conexão para que sua String de conexão não seja alterada.

`const servername = "mysql:host=localhost;dbname=tabfa4_protocolo";` (THE PHP GROUP, 2001-2018).

Todas as variáveis de conexão, assim como as de *queries*, possuem o formato de constantes, fazendo com que seu valor seja inalterável.

### 3.4.3 Acesso ao software

Apenas a página `login.php` tem acesso permitido, inicialmente. Criando a sessão do usuário após o login do mesmo. Ao clicar no botão de sair, a sessão respectiva ao usuário atual é completamente destruída, fazendo com que o acesso as outras páginas, a parte da de logon, sejam restringidos novamente, o mesmo acontece ao se fechar o navegador, ou retornar para a página de login por outro meio, como uso da tecla *backspace*, ou digitando o endereço da página de login manualmente. Junto disso, o comando `unset()` é utilizado para o *reset* das variáveis globais de sessão utilizadas; mesmo caso o usuário permaneça na página do software após ter realizado uma ação, as variáveis utilizadas para a criação ou edição de um protocolo são destruídas.

Da mesma forma, o software terá uma máquina virtual dedicada, tendo seu acesso somente dentro da intranet da faculdade.

### 3.5 Banco de dados

Tamanho inicial de 1 megabyte, tendo apenas escalonamento disponível suficiente para o armazenamento dos protocolos e logs, conforme ambos são gerados.

#### 3.5.1 Queries SQL

Queries SQL utilizadas no banco de dados, seja para criação das tabelas utilizadas, para as *triggers* ou mesmo para os comandos gerais de *select*, *update* e *delete*. Consistem nas *queries* para criação da tabela de HAE, tabela de data das HAEs e a tabela relatório. Todas as *queries* se encontram no Anexo A.

- **Trigger de insert relacionado a tabela de HAE:** *Trigger* que insere os dados específicos do usuário, assim como o horário, na tabela de *logs*, ao realizar a ação de inserir dados na tabela de HAE.
- **Trigger de update relacionado a tabela de HAE:** *Trigger* que insere os dados específicos do usuário, assim como o horário, na tabela de *logs*, ao realizar a ação de alterar os dados na tabela de HAE.
- **Trigger de delete relacionado a tabela de HAE:** *Trigger* que insere os dados específicos do usuário, assim como o horário, na tabela de *logs*, ao realizar a ação de excluir dados da tabela de HAE. DROP TABLE TABFA4\_S\_Protocolo\_HAE;

### 3.5.2 Dicionário de dados

- **TABFA4\_JN\_Protocolo\_Relatorio**

Tabela 11 – Tabela de relatórios

Coluna	Tipo	Nulo	Predefinido	Ligações para
ID_relatorio ( <i>Primária</i> )	smallint(6)	Não		
ID_user	char(12)	Não		
ID_HAE	smallint(6)	Sim	<i>NULL</i>	tabfa4_s_protocolo_hae -> ID_HAE
DT_relatorio	date	Não		
DS_acao	char(7)	Não		

Fonte: Autoria própria.

- **Índices TABFA4\_JN\_Protocolo\_Log**

Tabela 12 – Tabela de índice dos logs

Nome da chave	Tipo	Único	Pacote	Coluna	Quantidade	Agrupamento (Collation)	Nulo
PRIMARY	BTREE	Sim	Não	ID_relatorio	0	A	Não
ID_HAE	BTREE	Não	Não	ID_HAE	0	A	Sim

Fonte: Autoria própria.

- **TABFA4\_S\_Protocolo\_HAE**

Tabela 13 – Tabela de HAE

Coluna	Tipo	Nulo	Predefinido	Ligações para
ID_HAE ( <i>Primária</i> )	smallint(6)	Não		
ST_HAE	varchar(7)	Não		
DS_atividade_HAE	varchar(32)	Não		
DS_projeto_HAE	varchar(30)	Sim	<i>NULL</i>	
NM_docente_HAE	varchar(50)	Não		
DS_departamentoCurso_HAE	varchar(100)	Não		
VL_quantidade_HAE	char(1)	Não		
DS_semestre_HAE	char(2)	Não		
DS_HAE	varchar(50)	Não		
DS_User_HAE	varchar(12)	Não		

Fonte: Autoria própria.

- **Índices**

Tabela 14 – Tabela de índices da HAE

Nome da chave	Tipo	Único	Pacote	Coluna	Quantidade	Agrupamento (Collation)	Nulo
PRIMARY	BTREE	Sim	Não	ID_HAE	0	A	Não

Fonte: Autoria própria.

- **TABFA4\_S\_data**

Tabela 15 – Tabela de data das HAEs

Coluna	Tipo	Nulo	Predefinido	Ligações para
ID_data ( <i>Primária</i> )	smallint(6)	Não		
DD_dia	date	Sim	NULL	
DD_inicio	time	Sim	NULL	
DD_termino	time	Sim	NULL	

Fonte: Autoria própria.

- **Índices**

Tabela 16 – Tabela de índices da data

Nome da chave	Tipo	Único	Pacote	Coluna	Quantidade	Agrupamento (Collation)	Nulo
PRIMARY	BTREE	Sim	Não	ID_data	0	A	Não

Fonte: Autoria própria

- **TABFA4\_S\_HAE\_data**

Tabela 17 – Tabela de relação HAE e Data

Coluna	Tipo	Nulo	Predefinido	Ligações para
ID_data ( <i>Primária</i> )	smallint(6)	Não		tabfa4_s_data -> ID_data
ID_HAE ( <i>Primária</i> )	smallint(6)	Não		tabfa4_s_protocolo_hae -> ID_HAE

Fonte: Autoria própria.

- Índices

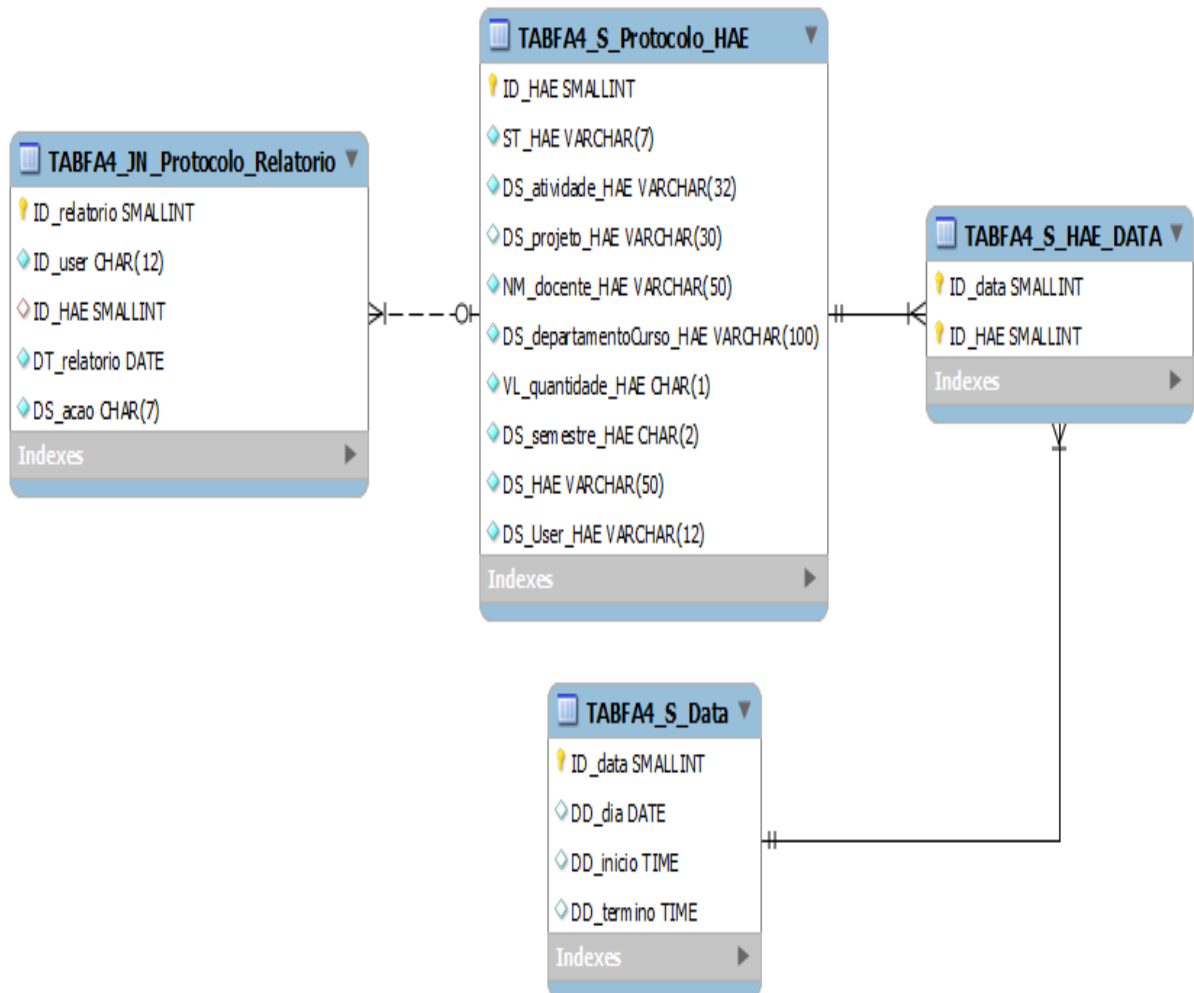
Tabela 18 – Tabela de índices da tabela de relação HAE e Data

Nome da chave	Tipo	Único	Pacote	Coluna	Quantidade	Agrupamento (Collation)	Nulo
PRIMARY	BTREE	Sim	Não	ID_data	0	A	Não
				ID_HAE	0	A	Não
ID_HAE	BTREE	Não	Não	ID_HAE	0	A	Não

Fonte: Autoria própria.

### 3.5.3 Modelo Relacional de Entidade Lógica

Figura 6 - MREL



Fonte: Autoria própria.



### 3.6 Testes de funcionalidade

- **Criação de protocolos**

Tabela 19 – Teste de criação de protocolos

<b>Entrada:</b>	Strings para nomes, status, tipos de protocolos, inteiros para criação dos ID, estes que estarão com incremento automático por padrão, e variáveis do tipo date para as datas utilizadas.
<b>Testes:</b>	Tipos de entrada. Verificação de String. Verificação de int. Verificação de data. Tamanho mínimo para os campos. Tamanho máximo para os campos. Tamanho máximo de caracteres de entrada. Tamanho mínimo de caracteres de entrada. Conexão com o banco de dados.
<b>Saída:</b>	Protocolo salvo e mensagem de protocolo cadastrado com sucesso no sistema.

Fonte: Autoria própria.

- **Alteração de protocolos**

Tabela 20 – Teste de alteração de protocolos

<b>Entrada:</b>	Strings para nomes, status, tipos de protocolos, inteiros para criação dos ID, estes que estarão com incremento automático por padrão, e variáveis do tipo date para as datas utilizadas.
<b>Testes:</b>	<p>Tipos de entrada.</p> <p>Verificação de String.</p> <p>Verificação de int.</p> <p>Verificação de data.</p> <p>Tamanho mínimo para os campos.</p> <p>Tamanho máximo para os campos.</p> <p>Tamanho máximo de caracteres de entrada.</p> <p>Tamanho mínimo de caracteres de entrada.</p> <p>Conexão com o banco de dados.</p> <p>Existência do protocolo no banco de dados.</p> <p>ID válido.</p>
<b>Saída:</b>	Protocolo alterado e mensagem de protocolo alterado com sucesso no sistema.

Fonte: Autoria própria.

- **Excluir protocolos**

Tabela 21 – Teste de exclusão de protocolos

<b>Entrada:</b>	ID específica do protocolo selecionado.
<b>Testes:</b>	Conexão com o banco de dados.  Existência do protocolo no banco de dados.  ID válido.
<b>Saída:</b>	Protocolo deletado do sistema e mensagem de protocolo excluído com sucesso, ou mensagem de falha e <i>rollback</i> da ação.

Fonte: Autoria própria.

- **Pesquisar protocolos**

Tabela 22 – Teste de pesquisa de protocolos

<b>Entrada:</b>	String fornecida por meio de <i>input</i> do usuário.
<b>Testes:</b>	Validação dos caracteres fornecidos pelo usuário no <i>input</i> .  Conexão com o banco de dados.  Existência do protocolo no banco de dados.  ID válido.
<b>Saída:</b>	Protocolo deletado do sistema e mensagem de protocolo excluído com sucesso, ou mensagem de falha e <i>rollback</i> da ação.

Fonte: Autoria própria.

### 3.7 Testes para aplicação web

Acessos por parte da página de login:

- Página de administrador;
- Página inicial.

Acesso por parte da página de administrador:

- PDF dos relatórios.

Acesso por parte da página inicial:

- Página de criação;
- Página de edição;
- Página de pesquisa;
- Página de ajuda;
- Página sobre;
- Página de log on.

Acesso por parte da página de criação:

- Página inicial;
- Página de edição;
- Página de pesquisa;
- Página de ajuda;
- Página sobre;
- Página de log on.

Acesso por parte da página de edição:

- Página de criação;
- Página de inicial;
- Página de pesquisa;
- Página de ajuda;

- Página sobre;
- Página de log on.

Acesso por parte da página de pesquisa:

- Página de criação;
- Página de edição;
- Página de inicial;
- Página de ajuda;
- Página sobre;
- Página de log on.

Acesso por parte da página ajuda:

- Página de criação;
- Página de edição;
- Página de pesquisa;
- Página de inicial;
- Página sobre;
- Página de log on.

Acesso por parte da página de sobre:

- Página de criação;
- Página de edição;
- Página de pesquisa;
- Página de ajuda;
- Página inicial;
- Página de log on.

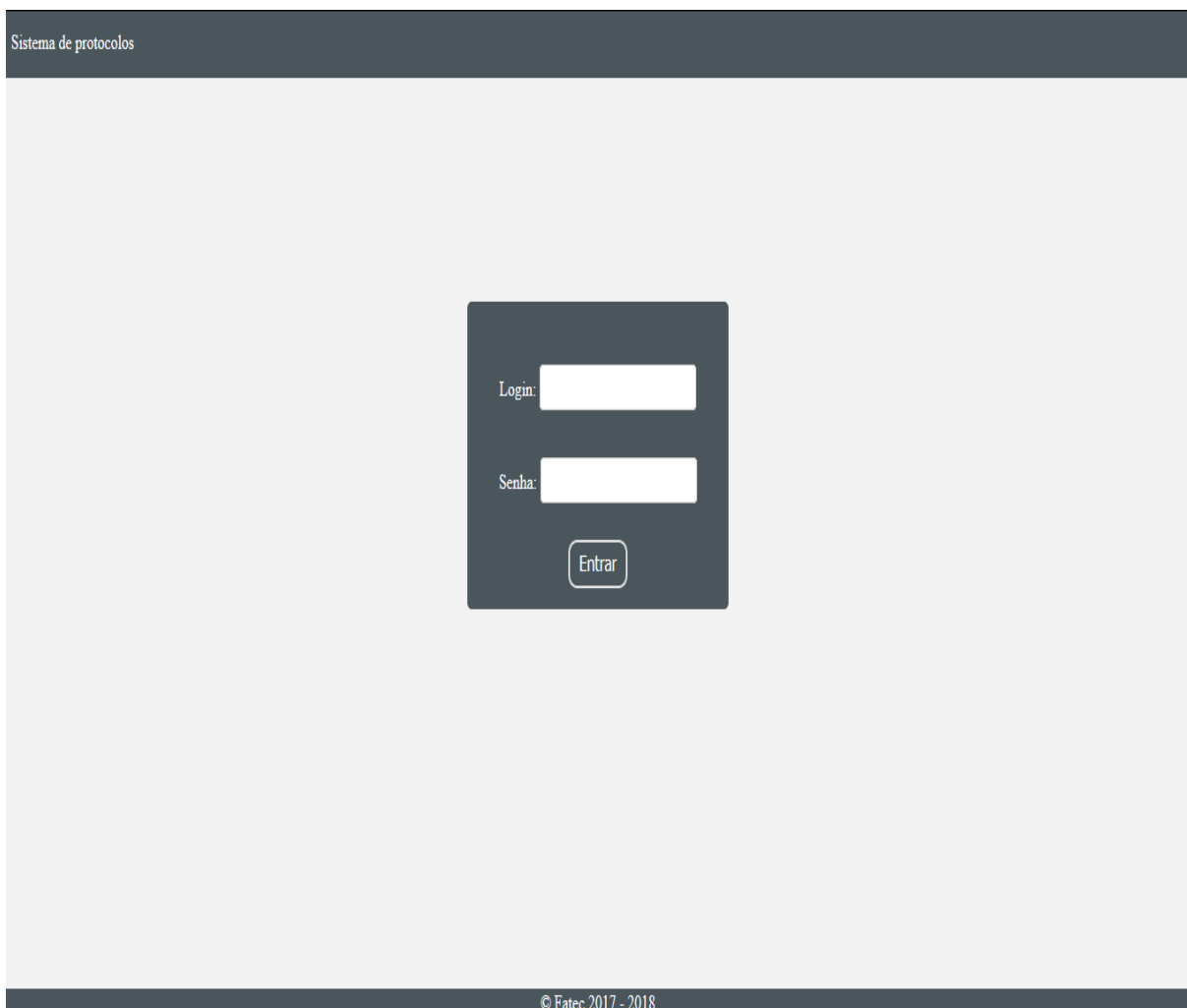
## 4 RESULTADOS DE DISCUSSÕES

### 4.1 Software e usabilidade

- **Tela de logon:**

Tela principal do software, exibindo dois campos de *input* para captura, e verificação, do usuário e senha (após acionar o botão). Página acessada, inicialmente, independente da entrada feita pelo usuário quanto à URL.

Figura 7 – Tela de logon



A imagem mostra a tela de login de um sistema. No topo, há uma barra escura com o texto "Sistema de protocolos" em branco. O fundo principal é cinza claro. Centralizado na tela há um formulário escuro com dois campos de entrada brancos: "Login:" e "Senha:". Abaixo dos campos está um botão "Entrar" com um ícone de seta para a direita. Na base da tela, uma barra escura contém o texto "© Fatec 2017 - 2018" em branco.

Fonte: Autoria própria.

- **Tela de início:**

Tela inicial do software, liberada após confirmação do cadastro ser referente ao usuário do tipo professor. Possui, apenas, uma apresentação e a imagem do calendário acadêmico referente ao semestre atual.

Figura 8 – Tela inicial

The screenshot displays the initial interface of the software. At the top, there is a navigation menu with options: Início, Criar, Editar, Concluídos, Ajuda, Sobre, and Sair. The date 'Sexta-feira, 13 De Abril De 2018' is shown in the top right corner. The main content area is divided into three sections, each representing a month: JULHO, AGOSTO, and SETEMBRO. Each month section contains a calendar grid on the left and a list of events on the right. The events are listed with their dates and descriptions.

JULHO						
D	S	T	Q	Q	S	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

JULHO		
2	Término das aulas do 1º Semestre Letivo de 2016.	
8	Prazo Final para entrega de médias e faltas do 1ºS/2016 dos Sistemas HYPNOS e SIGA.	
9	Dia da Revolução Constitucionalista de 1932.	
11 a 14	Período para solicitar REVISÃO de notas e faltas nos Sistemas HYPNOS e SIGA no 1ºS/2016.	
18 a 22	Semana de Rematrícula aos alunos incluídos no Sistema SIGA.	
19 e 20	Divulgação do RESULTADO e ACEITE de vagas de remanejamento INTERNO e entre FATEC's.	
20	Divulgação da classificação geral e desempenho dos candidatos (Vestibular 2ºS/2016).	
21 e 22	Matrícula da 1ª lista de convocação dos ingressantes (Vestibular 2ºS/2016).	
25	Divulgação da 2ª lista de convocação dos ingressantes (Vestibular 2ºS/2016).	
25 e 26	Atendimento na secretaria acadêmica aos alunos que não efetuaram matrícula no Sistema SIGA.	
26	Matrícula da 2ª lista de convocação dos ingressantes (Vestibular 2ºS/2016).	

AGOSTO						
D	S	T	Q	Q	S	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

AGOSTO		
1	Início das Aulas do 2º Semestre Letivo 2016 para alunos INGRESSANTES presencial e EaD.	
1 a 5	Inscrições para transferências EXTERNAS, portadores de DIPLOMA e Ensino a Distância (EaD).	
1 a 5	Período de AJUSTES de Matrícula para alunos veteranos.	
4	Início da aplicação do exame de Proficiência de Língua Inglesa para alunos ingressantes.	
5	Prazo final para solicitar Aproveitamento de estudos realizado em Curso Superior (Ingressantes).	
8	Início das Aulas do 2º Semestre Letivo 2016 para alunos VETERANOS.	
11 e 12	Matrícula para transferências EXTERNAS, portadores de DIPLOMA e Ensino a Distância (EaD).	
13	Reunião Geral de Planejamento.	
16	Matrícula de alunos por ocasião de cancelamento de matrícula e ausência nas 2 primeiras semanas de aula (artigo 36).	
31	Colação de Grau Oficial - Clube Veteranos de Americana (18h30).	

SETEMBRO						
D	S	T	Q	Q	S	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

SETEMBRO		
7	Feriado - Independência do Brasil (1822).	
8	Comemoração dos 30 anos da Fatec Americana.	

© Fatec 2017 - 2018

Fonte: Autoria própria.

- **Tela para criação de protocolos:**

Tela principal do software, contendo campos para *input*, por parte do usuário, das informações referentes à HAE. Mostrando os campos de *input* citados nos capítulos anteriores, junto do botão de criar HAE, este que, por sua vez, realiza a verificação dos campos, valida os dados e responde com uma mensagem de erro ou sucesso.

Figura 9 - Tela de criação de protocolos

A interface de usuário para a criação de protocolos (HAE) apresenta uma barra de menu superior com as opções: Início, Criar, Editar, Concluídos, Ajuda, Sobre e Sair. No canto superior direito, a data "Sexta-feira, 13 De Abril De 2018" é exibida. O formulário principal contém os seguintes elementos:

- Um botão "Criar HAE" centralizado no topo.
- O campo "Atividade:" com um menu suspenso.
- O campo "Docente:" com um campo de texto.
- O campo "Departamento ou Curso:" com uma lista de opções, cada uma precedida por um botão de seleção (checkbox):
  - Análise e Desenvolvimento de Sistemas
  - Direção
  - Gestão EAD
  - Segurança da Informação
  - Gestão Empresarial
  - Têxtil e Moda
  - Produção Têxtil

Na base da interface, há uma barra de rodapé com o texto "© Fatec 2017 - 2018".

Fonte: Autoria própria.



- **Tela de pesquisa de HAEs em aberto:**

Tela do software que lista todos os protocolos criados, que possuem *status* de abertos, por meio do *output* dos dados do banco de dados em uma tabela HTML. Junto aos dados da tabela, encontram-se dois botões, por HAE, com a finalidade de editar ou deletar a HAE (respectivamente). Assim como, um campo de *input* para pesquisar HAEs por qualquer uma das colunas (ID, data, docente etc.), onde a pesquisa é iniciada por meio do botão “pesquisar” e esvaziada por meio do botão “limpar”

Figura 10 – Tela de pesquisa de HAEs

Ação	ID	Status	Atividade	Docente	Departamento	Data de início	Descrição
	50	Aberta	Projeto de Graduação (TCC/TG)	ação	ADS, Direção, Gestão EAD, SI	2018-05-05	qwdaqwd
	51	Aberta	Estágio Supervisionado	Anna	ADS, Direção, Gestão EAD, SI	2018-05-05	qweqwe
	52	Aberta	Estágio Supervisionado	rrr	ADS, Direção, Gestão EAD, SI	2018-05-05	qwetr
	53	Aberta	Estágio Supervisionado	aç	ADS, Direção, Gestão EAD, SI	2018-05-05	qweqwe
	54	Aberta	Projeto de Graduação (TCC/TG)	awd	ADS, Direção, Gestão EAD	2018-05-05	aaaa

© Fatec 2017 - 2018

Fonte: Autoria própria.

- **Tela para edição de HAE:**

Tela iniciada ao acionar o botão de edição na tela de edição de protocolos. Possui *outputs* baseados nos campos da HAE escolhida, onde os mesmos podem ser editados pelo usuário, ou mesmo o *status* atualizado para “Fechado”, fazendo com que a HAE passa para o armazenamento final. Os botões encontrados nesta tela são, respectivamente, de voltar para a tela anterior e para confirmar a edição dos dados, realizando a atualização dos dados no banco de dados.

Figura 11 – Tela de edição das HAEs

A imagem mostra a interface de usuário para a edição de HAEs (Histórico de Atividades Escolhidas) em um sistema de protocolos. No topo, há uma barra de título com o texto "Sistema de protocolos". Abaixo, um botão "Voltar" está centralizado. O formulário contém os seguintes elementos:

- Um campo "Status:" com um menu suspenso selecionando "Aberta".
- Um campo "Atividade:" com um menu suspenso selecionando "Projeto de Graduação (TCC/TG)".
- Um campo "Docente:" com o texto "ação" inserido.
- Uma seção intitulada "Departamento ou Curso:" com uma lista de opções de seleção:

Departamento ou Curso	Seleção
Análise e Desenvolvimento de Sistemas	<input checked="" type="checkbox"/>
Direção	<input checked="" type="checkbox"/>
Gestão EAD	<input checked="" type="checkbox"/>
Segurança da Informação	<input checked="" type="checkbox"/>
Gestão Empresarial	<input type="checkbox"/>
Têxtil e Moda	<input type="checkbox"/>
Produção Têxtil	<input type="checkbox"/>
Jogos Digitais	<input type="checkbox"/>
Logística	<input type="checkbox"/>

Na base da interface, há uma barra de rodapé com o texto "© Fatec 2017 - 2018".

Fonte: Autoria própria.

- **Tela de pesquisa de HAEs prontas:**

Tela semelhante à tela de pesquisa de HAEs em aberto, onde a tabela HTML é preenchida pelo *output* dos dados no banco de dados, mas apenas as HAEs com *status* “Fechada” são listadas. Ademais, cada HAE possui um botão para transformar os dados do banco de dados no formato de protocolo HAE utilizado pela instituição. Junto de, finalmente, o campo de *input* para texto de pesquisa, com os botões para pesquisar e limpar tais informações providas do usuário.

Figura 12 – Tela de pesquisa de HAEs prontas

Ação	ID	Status	Atividade	Docente	Departamento	Data de inicio	Descrição
	19	Fechada	TCC	Anna	ADS,GE,SI,Têxtil e Moda,JD,Logistica	0000-00-00	
	20	Fechada	TCC	Anna	ADS,GE,SI,Têxtil e Moda,JD	0000-00-00	
	21	Fechada	TCC	Anna	ADS,GE,SI,Têxtil e Moda	0000-00-00	
	22	Fechada	TCC	Anna	ADS,GE,Têxtil e Moda,JD,Logistica	0000-00-00	
	23	Fechada	TCC	Anna	ADS,GE,Têxtil e Moda,JD	0000-00-00	
	24	Fechada	TCC	Anna	ADS,GE,Têxtil e Moda	0000-00-00	
	25	Fechada	TCC	Anna	ADS,Têxtil e Moda,JD,Logistica	0000-00-00	

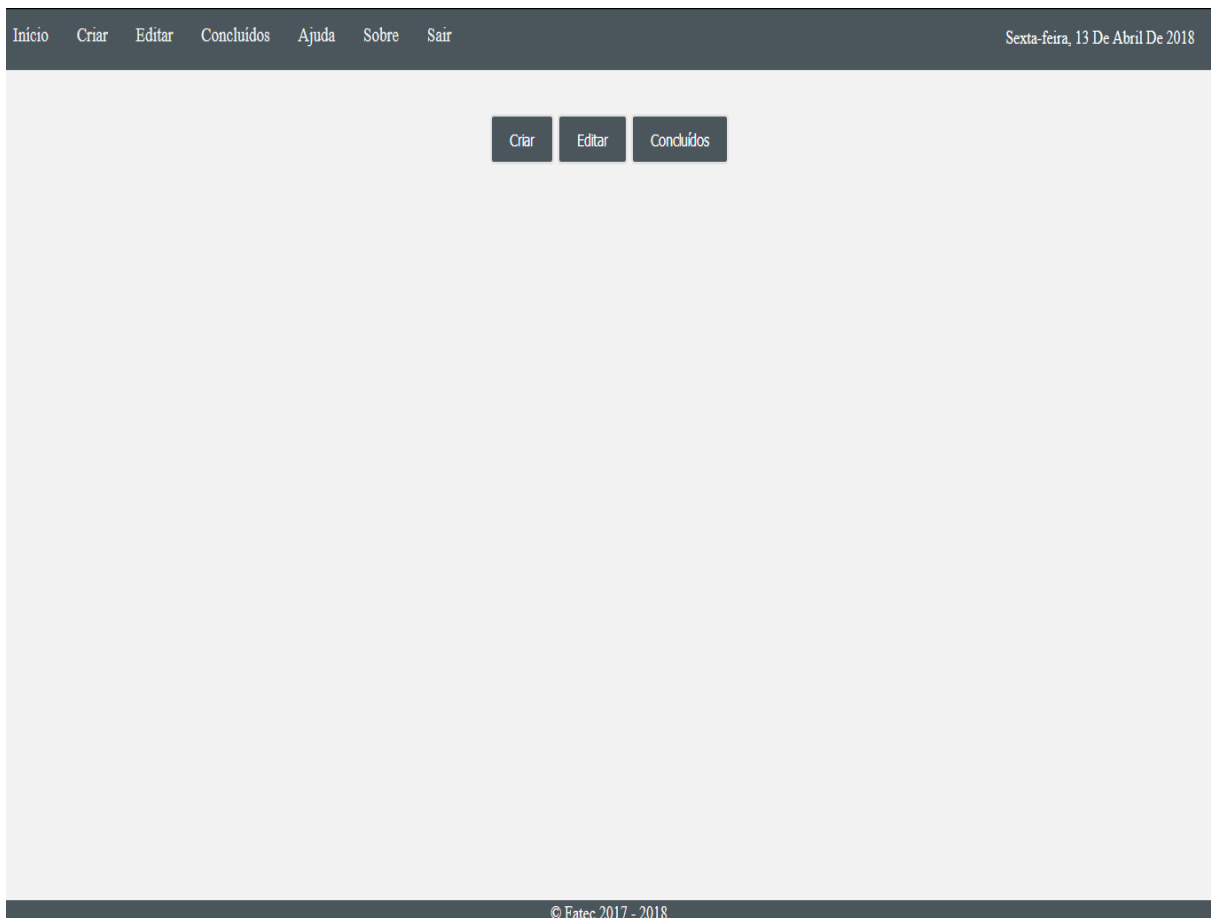
© Fatec 2017 - 2018

Fonte: Autoria própria.

- **Tela de ajuda do software:**

Tela destinada em ajudar o usuário em suas dúvidas corriqueiras quanto à utilização do software em seu dia-a-dia. Possuindo, apenas, um botão respectivo em uma função do software, onde, ao ser acionado, exibe um *output* em texto tutorial para o usuário.

Figura 13 – Tela de ajuda



Fonte: Autoria própria.

- **Tela de relatórios de ações:**

Tela que exibe um relatório geral das ações realizadas no software por parte do usuário. Consiste, basicamente, de um *output* dos dados numa tabela HTML da ação realizada, junto do ID do usuário ator e ID da HAE alvo da ação. A tela também possui três botões: um para gerar o PDF das informações descritas a priori; um botão para pesquisar um relatório específico, tendo *input* por parte do administrador e um botão para limpar a pesquisa, retornando ao estado inicial da página. Por fim, o menu desta página específica aos administradores possui um botão de *log out*.

Figura 14 – Tela de relatórios

ID	ID HAE	Data	Ação	Usuário
5	16	2017-11-16	Insert	
6	18	2017-11-16	Insert	:user
31	22	2017-11-16	Insert	User
32	22	2017-11-16	Update	User
34	22	2017-11-16	Delete	User
35	23	2017-11-16	Insert	User
36	23	2017-11-16	Update	User
37	23	2017-11-16	Delete	User
38	21	2017-11-22	Insert	User
39	21	2017-11-22	Update	User
40	21	2017-11-22	Update	User
41	22	2017-11-22	Insert	User
42	22	2017-11-22	Update	User


© Fatec 2017 - 2018

Fonte: Autoria própria.

- **PDF de uma HAE pronta:**

Resultado em formato PDF, da geração dos dados do banco de dados de uma HAE finalizada.

Figura 15 – Tela de PDF de uma HAE


**GOVERNO DO ESTADO DE SÃO PAULO**

**Faculdade de Tecnologia de Americana**

**Relatório de Atividade  
(Anexo I)**

<b>Tipo/Atividade</b>	TCC				
<b>Docente</b>	Anna				
<b>Departamento/Curso</b>	ADS, GE, SI, T?xtile Moda, J.D., Log?stica				
<b>Quantidade HAEs</b>	1	<b>Semestre/Ano</b>	22017	<b>Data de Início</b>	2017

**Descrições das atividades a serem desenvolvidas no semestre**

\_\_\_\_\_

**Assinatura do Docente**

\_\_\_\_\_

**Assinatura do(a) Coordenador(A)**

_____ Wladimir da Costa Coordenador (ADS)	_____ Reydner Furtado Garbero Coordenadora(Gestão Empresarial)	_____ Maria Cristina Aranda Coordenadora(SI)
_____ Agnaldo Pescelaro Prezoto Coordenador (PT e TM)	_____ Cleberson Eugenio Forte Coordenador (Jogos Digitais)	_____ Marco Anselmo de Godoi Prezoto Coordenador (Logística)

**Assinatura do(a) Diretor(a)**

\_\_\_\_\_

Fonte: autoria própria.

- **PDF dos relatórios:**

Resultado em PDF da tabela de relatórios, mostrado em formato específico para os relatórios de HAE.

Figura 16 – Tela de PDF dos relatórios



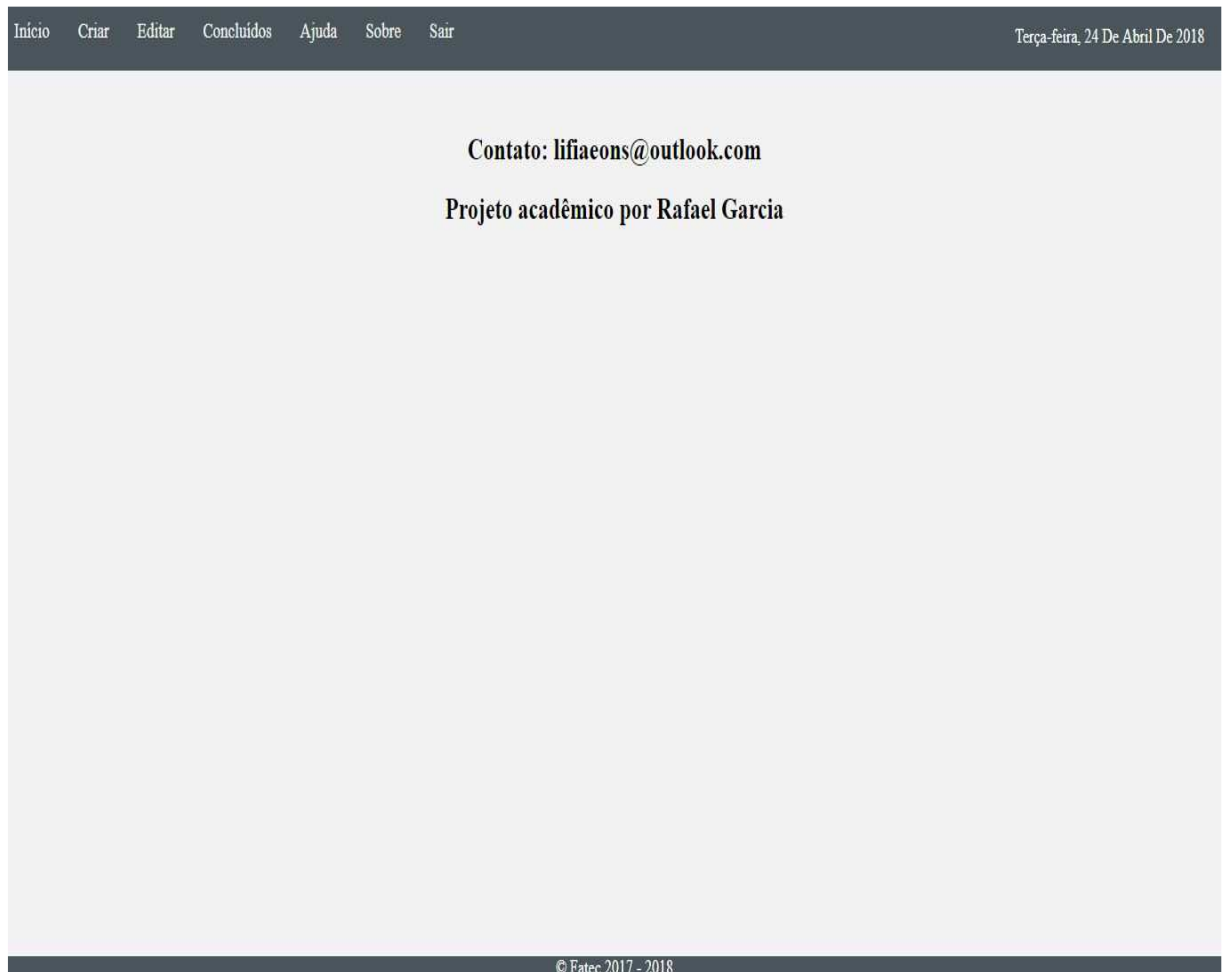
ID relatório	Usuário	ID HAE	Data da ação	Ação
5		16	2017-11-16	Insert
6	:user	18	2017-11-16	Insert
31	User	22	2017-11-16	Insert
32	User	22	2017-11-16	Update
34	User	22	2017-11-16	Delete
35	User	23	2017-11-16	Insert
36	User	23	2017-11-16	Update
37	User	23	2017-11-16	Delete
38	User	21	2017-11-22	Insert
39	User	21	2017-11-22	Update
40	User	21	2017-11-22	Update
41	User	22	2017-11-22	Insert
42	User	22	2017-11-22	Update
43	User	23	2017-11-22	Insert
44	User	23	2017-11-22	Update
45	User	24	2017-11-30	Insert
46	User	24	2017-11-30	Update
47	User	23	2017-12-02	Update
48	User	25	2017-12-02	Insert

Fonte: Autoria própria.

- **Tela com informações sobre o autor:**

Tela com a única funcionalidade de informar, brevemente, informações sobre o autor. Com intuito de contato e padronização de software.

Figura 17 – Tela sobre o autor



Fonte: Autoria própria.



## 5. CONSIDERAÇÕES FINAIS

Ao final do presente trabalho, considerando-se a parte de engenharia como um todo, a criação, gerência e manutenção do banco de dados, junto do desenvolvimento em si, acompanhado das métricas de boa programação, e seguindo-se as aplicações de segurança, tanto no código, quanto no banco de dados. Conclui-se que, considerando os levantamentos escritos anteriormente e a documentação em si do trabalho, o software atingiu os objetivos propostos pelo usuário final, cliente e demais *stakeholders* participantes. Sendo uma ferramenta pronta para uso e gerenciamento.

## 6. REFERÊNCIAS

ALPINE LINUX DEVELOPMENT TEAM. **About**. 2008-2017. Disponível em: <<https://alpinelinux.org/about/>>. Acesso em: 26 de outubro de 2017, às 15:04.

ALPINE LINUX DEVELOPMENT TEAM, 2008-2018. **Setting Up Apache with PHP**. 21 de setembro de 2017. Disponível em: <[https://wiki.alpinelinux.org/wiki/Setting\\_Up\\_Apache\\_with\\_PHP](https://wiki.alpinelinux.org/wiki/Setting_Up_Apache_with_PHP)>. Acesso em: 26 de outubro de 2017, às 11:07.

SHARON, Tomer. **Lean, mean UX**: the mindset you need to develop products that people desire. 9 de julho de 2013. Disponível em: <<https://speakerdeck.com/tsharon/mean-lean-ux>>. Acesso em: 09 de março de 2018, às 09h08min.

FPDF Library. **What is FPDF?**. 2018. Disponível em: <<http://www.fpdf.org/>>. Acesso em: 01 de março de 2018, às 09h35.

MICROSOFT, 2018. **Visual Studio Code**. Disponível em: <<https://code.visualstudio.com/>>. Acesso em: 16 de abril de 2018, às 08h50min.

HO, Don, 2016. **About**. Disponível em: <<https://notepad-plus-plus.org/>>. Acesso em: 16 de abril de 2018, às 08h51min.

OMG UML®, 2018. **UML Class and Object Diagram Overview**. Disponível em: <<https://www.uml-diagrams.org/class-diagrams-overview.html>>. Acesso em: 28 de fevereiro de 2018, às 11h30min.

OMG UML®, 2018. **UML Use Case Diagram**. Disponível em: <<https://www.uml-diagrams.org/use-case-diagrams.html>>. Acesso em: 28 de fevereiro de 2018, às 11h33min.

OMG UML®, 2018. **UML Activity Diagram**. Disponível em: <<https://www.uml-diagrams.org/activity-diagrams.html>>. Acesso em: 28 de fevereiro de 2018, às 11h35min.

OMG UML®, 2018. **UML Sequence Diagram**. Disponível em: <<https://www.uml-diagrams.org/sequence-diagrams.html>>. Acesso em: 28 de fevereiro de 2018, às 11h37min.

OBJECT MANAGEMENT GROUP, 2018. **About the Unified Modeling Language Specification Version 2.5**. 2015. Disponível em: <<http://www.omg.org/spec/UML>>. Acesso em: 23 de nov. 2017, às 11h.

OBJECT MANAGEMENT GROUP, 2018. **What is UML: Introduction**. 2005. Disponível em: <<http://www.uml.org/what-is-uml.htm>>. Acesso em: 23 de novembro, às 11h01.

Apache Friends, 2018. **What is XAMPP?**. 2018. Disponível em: <<https://www.apachefriends.org/index.html>>. Acesso em: 16 de abril de 2018, às 09h53min.

W3C. **HTML 5.2**. 2018. Disponível em: <<https://www.w3.org/TR/html52/introduction.html#introduction>>. Acesso em: 16 de abril de 2018, às 09h49min.

BOS, Bert. **WHAT IS CSS?**. 2018. Disponível em: <<https://www.w3.org/Style/CSS/Overview.en.html>>. Acesso em: 16 de abril de 2018, às 09h49min.

Mozilla.org contributors, 1998-2018. **The new Firefox**. 2018. Disponível em: <<https://www.mozilla.org/en-US/firefox/new/>>. Acesso em: 16 de abril de 2018, às 10h37min.

Google. **Browse fast**. 2018. Disponível em: <<https://www.google.com/chrome/>>. Acesso em: 16 de abril de 2018, às 10h38min.

Vivaldi Technologies™, 2015-2018. **Vivaldi: A browser for our friends**. 2018. Disponível: <<https://vivaldi.com/>>. Acesso em: 16 de abril de 2018, às 10h39min.

Opera software, 2018. **Fast, secure, easy-to-use browser**. 2018. Disponível: <<https://www.opera.com/>>. Acesso em: 16 abril de 2018, às 10h43min.

MICROSOFT, 2018. **Internet Explorer Downloads**. 2018. Disponível em: <<https://support.microsoft.com/en-us/help/17621/internet-explorer-downloads>>. Acesso em: 16 de abril de 2018, às 10h41min.

THE PHP GROUP. **Unset**. 2001-2018. Disponível em: <<http://php.net/manual/en/function.unset.php>>. Acesso em: 23 nov. 2017, às 16h05min.

THE PHP GROUP. **PDOStatement::bindValue**. 2001-2018. Disponível em: <<http://php.net/manual/en/pdostatement.bindvalue.php>>. Acesso em: 23 nov. 2017, às 16h05min.

THE PHP GROUP. **\$\_SESSION.** 2001-2018. Disponível em: <<http://php.net/manual/en/reserved.variables.session.php>>. Acesso em: 23 nov. 2017, às 16h05min.

THE PHP GROUP. **Sanitize Filters.** 2001-2018. Disponível em: <<http://php.net/manual/en/filter.filters.sanitize.php>>. Acesso em: 24 nov. 2017, às 09h23min.

THE PHP GROUP. **Predefined Constants.** 2001-2018. Disponível em: <<http://php.net/manual/en/pdo.constants.php>>. Acesso em: 23 nov. 2017, às 16h05min.

THE PHP GROUP. **PHP Data Objects.** 2001-2018. Disponível em: <<http://php.net/manual/en/book.pdo.php>>. Acesso em: 23 nov. 2017, às 16h05min.

THE PHP GROUP. **Header.** 2001-2018. Disponível em: <<http://php.net/manual/en/function.header.php>>. Acesso em: 23 nov. 2017, às 16h05min.

THE PHP GROUP. **Securing Session INI Settings.** 2001-2018. Disponível em: <<http://php.net/manual/en/session.security.ini.php>>. Acesso em: 25 nov. 2017, às 11h53min.

Asta Community. Version 6.9. [s.l.]: Change Vision, Inc, 2006-2017.

Asta. **About Astah.** 2018. Disponível em: <<http://astah.net/about-us>>. Acesso em: 22 de abril de 2018, às 22h57min.

REFSNES DATA. **PHP 5 Form Validation**. 1999-2018. Disponível em: <[https://www.w3schools.com/php/php\\_form\\_validation.asp](https://www.w3schools.com/php/php_form_validation.asp)>. Acesso em: 23 de nov. 2017, às 16h53min.

phpMyAdmin Contributors, **Bringing MySQL to the WEB**. 2003 – 2018. Disponível em: <<https://www.phpmyadmin.net/>>. Acesso em: 01 de março de 2018, às 11h05min.

FileZilla, **Overview**. 2018. Disponível em: <<https://filezilla-project.org/>>. Acesso em: 16 de abril de 2018, às 09h26min.

WELLS, Don. **Agile Software Development: A gentle introduction**. 2009. Disponível em: <<http://www.agile-process.org/>>. Acesso em: 16 de abril de 2018, às 09h28min.

JAMES, Michael. **An Empirical Framework For Learning (Not a Methodology)**. 2018. Disponível em: <<http://scrummethodology.com/>>. Acesso em: 16 de abril de 2018, às 09h31min.

Oracle Corporation and/or its affiliates. **Introduction to InnoDB**. 2018. Disponível em: <<https://dev.mysql.com/doc/refman/5.7/en/innodb-introduction.html>>. Acesso em: 01 de março de 2018, às 11h03min.

Oracle Corporation and/or its affiliates. **MySQL Workbench**. 2018. Disponível em: <<https://dev.mysql.com/doc/workbench/en/>>. Acesso em: 16 de abril de 2018, às 09h24min.

THE OPEN WEB APPLICATION SECURITY PROJECT. **Cross-site Scripting (XSS)**. 2001-2018. Disponível em: <[https://www.owasp.org/index.php/Cross-site\\_Scripting\\_%28XSS%29](https://www.owasp.org/index.php/Cross-site_Scripting_%28XSS%29)>. Acesso em: 23 de nov. 2017, às 17h10min

CANTOR, Murray. **Software Leadership**: A Guide to successful Software Development. 1<sup>o</sup> ed. Indianapolis: Addison-Wesley 2001.

SOMMERVILLE, Ian. **Engenharia de Software**. 8<sup>o</sup> edição, 2007.

VERGILIO, Silvia R.; EMER, Maria C. F.; JINO, Mario. Teste de Aplicações Web. *In*: DELAMARO, Márcio E.; MALDONADO, José C.; JINO, Mario. **Introdução ao Teste de Software**. Rio de Janeiro/RJ: Elsevier, 2007. p. 218 (Capítulo 8<sup>o</sup>).

SILBERCHATZ, Abraham; Korth, Henry F.; Sudarshan, S. **Sistema de Banco de Dados**. 3<sup>o</sup> ED. São Paulo: Pearson Makron Books, 1999.

Delamaro, Márcio E.; Maldonado, José C.; Jino, Mario. **Introdução ao teste e software**. Rio de Janeiro: Elsevier, 2007.

## 7. GLOSSÁRIO

- *CLI (Command Line Interface)*: Interface de linha de comando, controle do sistema operacional, ou programas, por meio de linhas de comando, sem interação com guias ou demais botões de auxílio;
- *Back-end*: parte voltada ao servidor e a operabilidade do software;
- *Front-end*: parte voltada ao usuário e a entrada e saída de dados;
- *Output*: Saída de dados;
- *Input*: Entrada de dados;
- *Sprint*: Arrancada, ou processo contínuo de desenvolvimento, com um prazo visando a contemplação do desenvolvimento, junto de uma reunião com os chefes e/ou clientes;
- *Scrollbar*: Barras de rolagem. Presentes em diversas páginas da web, ou em softwares com um todo;
- *Log on*: autenticação no sistema;
- *GUI (Graphic User Interface)*: Parte gráfica dos softwares, onde o usuário final interage com as funcionalidades do software. Possui guias, botões de auxílio e navegação intuitiva;
- *Private*: Variável privado, tendo seu acesso restrito a classe da qual pertence;
- *EER (Enhanced Entity-relationship model)*: Diagrama de entidade relacional lógico das tabelas;
- *UML (Unified Modeling Language)*: Linguagem de Modelagem Unificada;
- *Queries*: Linhas de código específicas de linguagens de modelagem de banco de dados, SQL;
- *Stakeholders*: Usuários finais, clientes e demais pessoas, e meios em geral, relacionados ao software.



## **Apêndice A**

*Queries SQL*

- **Criação do banco de dados:**

```
CREATE DATABASE TABFA4;
```

- **Criação da tabela de HAEs:**

```
USE TABFA4;
```

```
CREATE TABLE TABFA4_S_Protocolo_HAE(
```

```
-- Primary key da tabela HAE.
```

```
ID_HAE SMALLINT NOT NULL AUTO_INCREMENT,
```

```
-- Status da HAE, aberta/fechada
```

```
ST_HAE VARCHAR(7) NOT NULL,
```

```
-- Descrição das atividades da HAE.
```

```
DS_atividade_HAE VARCHAR(32) NOT NULL,
```

```
-- Descrição do projeto livre, disponível somente em caso de escolha
```

```
DS_projeto_HAE VARCHAR(30),
```

```
-- Nome do docente responsável pela HAE.
```

```
NM_docente_HAE VARCHAR(50) NOT NULL,
```

```
-- Descrição do departamento ou curso da HAE.
```

```
DS_departamentoCurso_HAE VARCHAR(100) NOT NULL,
```

```
-- Quantidade de HAEs
```

```
VL_quantidade_HAE CHAR(1) NOT NULL,
```

```
-- Descrição do semestre relacionado a HAE.
```

```
DS_semestre_HAE CHAR(2) NOT NULL,
```

```
-- Descrição das atividades a serem desenvolvidas.
```

```
DS_HAE VARCHAR(50) NOT NULL,
```

```
-- Login (AD) do usuário relacionado a HAE.
```

```
DS_User_HAE VARCHAR(12) NOT NULL,
```

```
-- Definição da primary key.
```

```
PRIMARY KEY (ID_HAE)
```

);

- **Criação da tabela de data:**

USE TABFA4;

```
CREATE TABLE TABFA4_S_Data(
  -- Primary key da tabela de datas.
  ID_data SMALLINT NOT NULL AUTO_INCREMENT,
  -- Data da HAE.
  DD_dia DATE,
  -- Horário inicial da HAE.
  DD_inicio TIME,
  -- Horário de término da HAE
  DD_termino TIME,

  -- Definição da primary key
  PRIMARY KEY (ID_data)
);
```

- **Criação da tabela de HAE\_DATA**

USE TABFA4;

```
CREATE TABLE TABFA4_S_HAE_DATA(
  -- Foreign keys que formam a primary key composta.
  ID_data SMALLINT NOT NULL,
  ID_HAE SMALLINT NOT NULL,

  -- Definição da primary key composta.
  PRIMARY KEY (ID_data, ID_HAE),
  -- Definição e referência da foreign key da tabela de HAE.
```

```

FOREIGN KEY (ID_HAE) REFERENCES TABFA4_S_Prococolo_HAE (ID_HAE),
-- Definição e referência da foreign key da tabela de Data.
FOREIGN KEY (ID_data) REFERENCES TABFA4_S_Data (ID_data)
);

```

- **Criação da tabela de relatório:**

```

USE TABFA4;

```

```

CREATE TABLE TABFA4_JN_Prococolo_Relatorio(
-- Primary key da tabela de relatorios
ID_relatorio SMALLINT NOT NULL AUTO_INCREMENT,
-- ID do usuário AD.
ID_user CHAR (12) NOT NULL,
-- ID da HAE.
ID_HAE SMALLINT,
-- Data do relatório.
DT_relatorio DATE NOT NULL,
-- Ação realizada pelo usuário.
DS_acao CHAR(7) NOT NULL,

-- Definição da primary key.
PRIMARY KEY (ID_relatorio),
-- Definição da foreign key, vinda da tabela de HAE.
FOREIGN KEY (ID_HAE) REFERENCES TABFA4_S_Prococolo_HAE(ID_HAE)
);

```

- **Trigger de insert relacionado a tabela de HAE:**

```
CREATE TRIGGER `HAE_insert` AFTER INSERT ON
  `TABFA4_S_Protocolo_HAE` FOR EACH ROW INSERT INTO
  TABFA4_JN_Protocolo_Relatorio (ID_HAE,DT_relatorio, DS_acao,
  UserLog) VALUES (new.ID_HAE, now(), 'Insert', new.DS_User_HAE);
```

- **Trigger de update relacionado a tabela de HAE:**

```
CREATE TRIGGER `HAE_update` AFTER UPDATE ON
  `TABFA4_S_Protocolo_HAE` FOR EACH ROW INSERT INTO
  TABFA4_JN_Protocolo_Relatorio (ID_HAE, DT_relatorio, DS_acao,
  UserLog) VALUES (new.ID_HAE, now(), 'Update', new.DS_User_HAE);
```

- **Trigger de delete relacionado a tabela de HAE:**

```
CREATE TRIGGER `HAE_delete` AFTER DELETE ON
  `TABFA4_S_Protocolo_HAE` FOR EACH ROW INSERT INTO
  TABFA4_JN_Protocolo_Relatorio (ID_HAE, DT_relatorio, DS_acao, UserLog)
  VALUES (old.ID_HAE, now(), 'Delete', old.DS_User_HAE);
```

- **Query para exclusão da tabela de HAE:**

```
DROP TABLE TABFA4_S_Protocolo_HAE;
```

- **Query para exclusão da tabela de relatórios:**

```
DROP TABLE TABFA4_JN_Protocolo_Relatorio;
```

- **Query para realizar pesquisa dos dados na tabela de HAE, utilizando um ID específico:**

```
SELECT * FROM TABFA4_S_Protocolo_HAE WHERE ID_HAE = :id;
```

- **Query para excluir dados da tabela de HAE, usando um ID específico:**

```
DELETE FROM `TABFA4_S_Procolo_HAE` WHERE ID_HAE = :id_hae;
```

- **Query para inserir dados na tabela de HAE:**

```
INSERT INTO `TABFA4_S_Procolo_HAE` (`ST_HAE`, `DS_atividade_HAE`,  
  `NM_docente_HAE`, `DS_projeto_HAE`, `DS_departamentoCurso_HAE`,  
  `VL_quantidade_HAE`, `DS_semestre_HAE`, `DS_HAE`, `DS_User_HAE`)  
VALUES (:status, :atividade, :docente, :dsProjeto, :departamento,  
  :quantidade, :semestre, :descricao, :user);
```

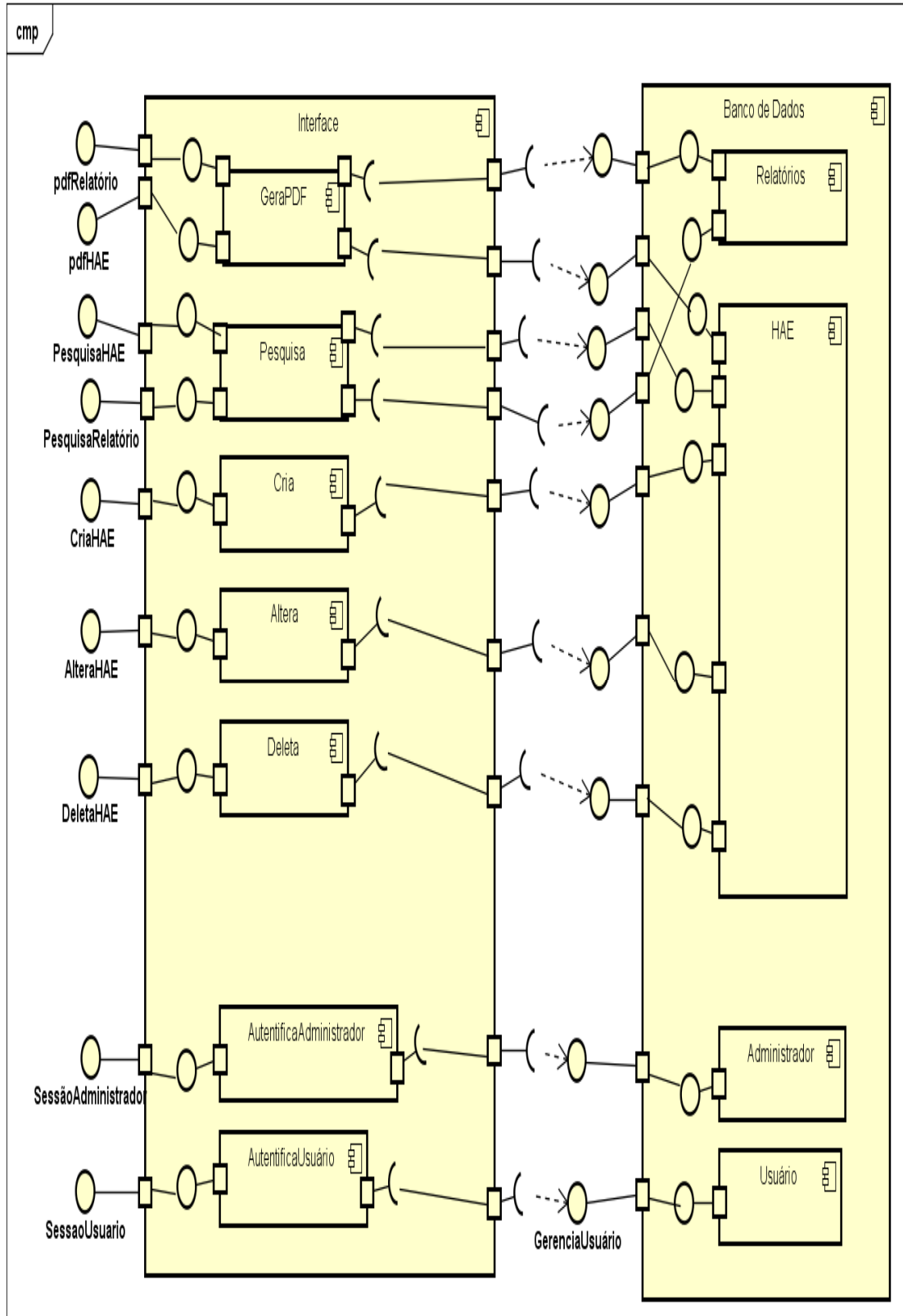
- **Query para adicionar uma nova coluna, exemplo utilizando a tabela de HAE:**

```
ALTER TABLE `TABFA4_S_Procolo_HAE` ADD `DS_User_HAE` VARCHAR(12)  
NOT NULL;
```

## **Apêndice B**

UML

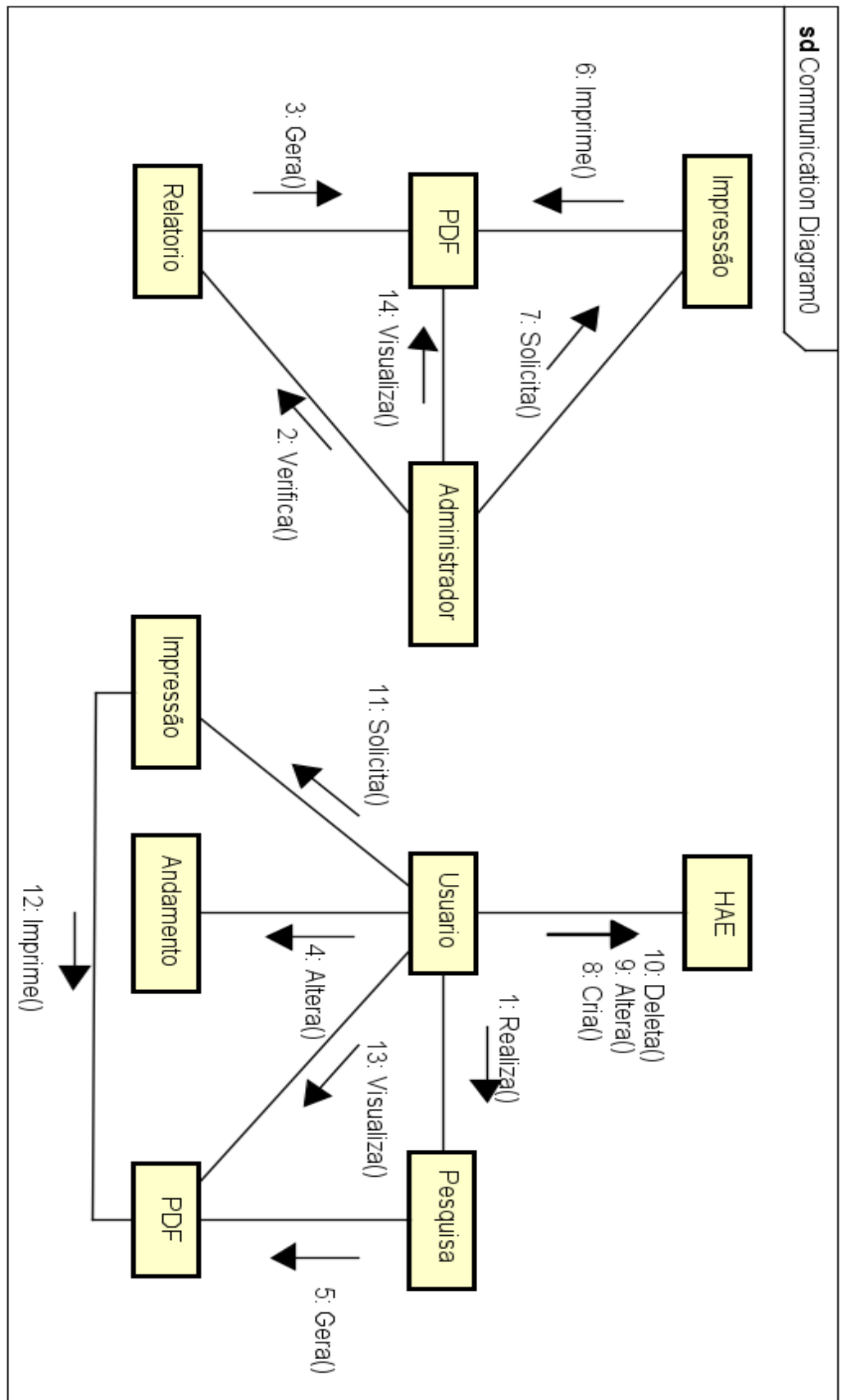
Figura 18 – Diagrama de componentes



Fonte: Autoria Própria.

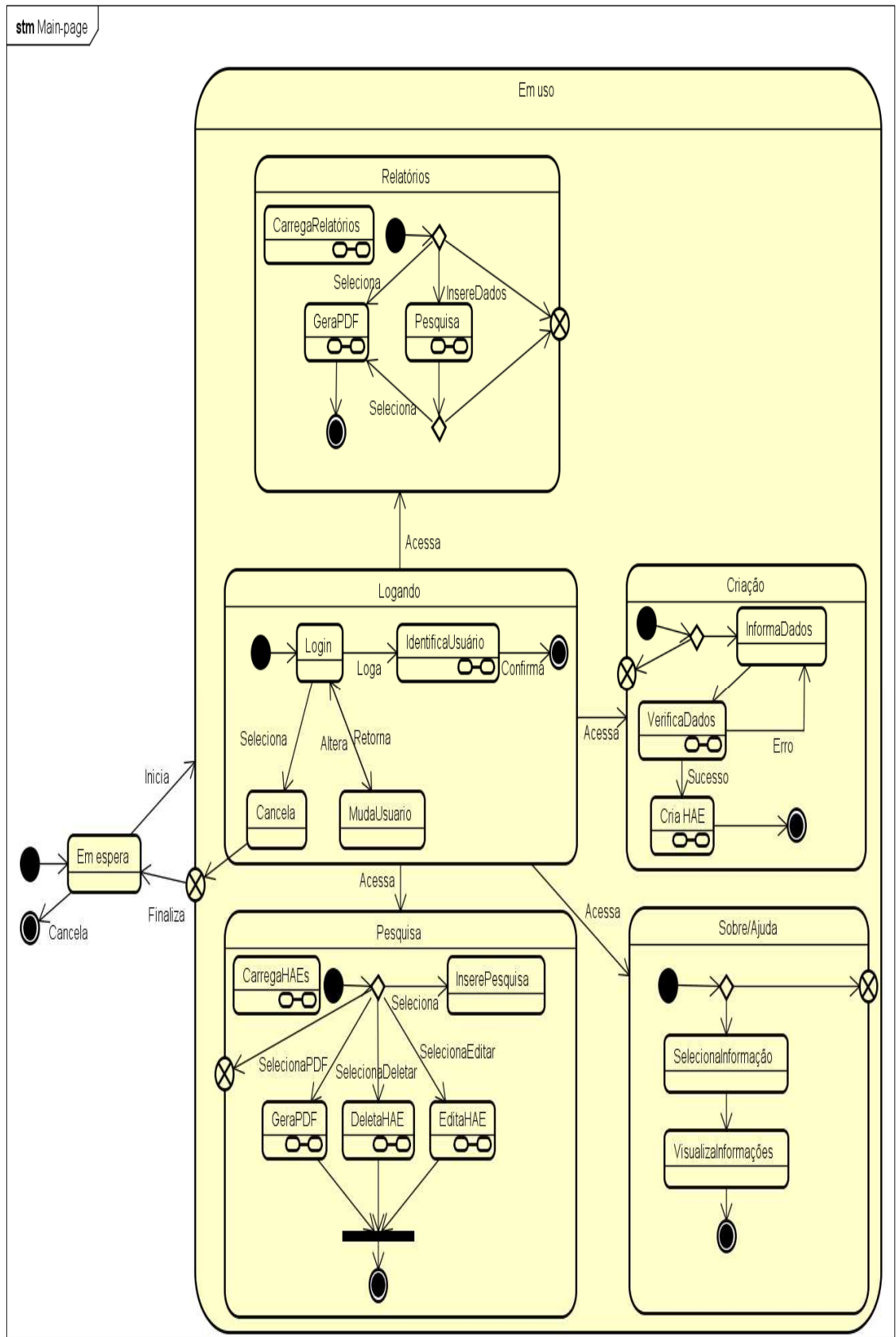


Figura 19 – Diagrama de comunicação



Fonte: Autoria própria.

Diagrama 20 – Diagrama de estado de máquina



Fonte: Autoria própria.

Diagrama 21 – Diagrama de estrutura composta

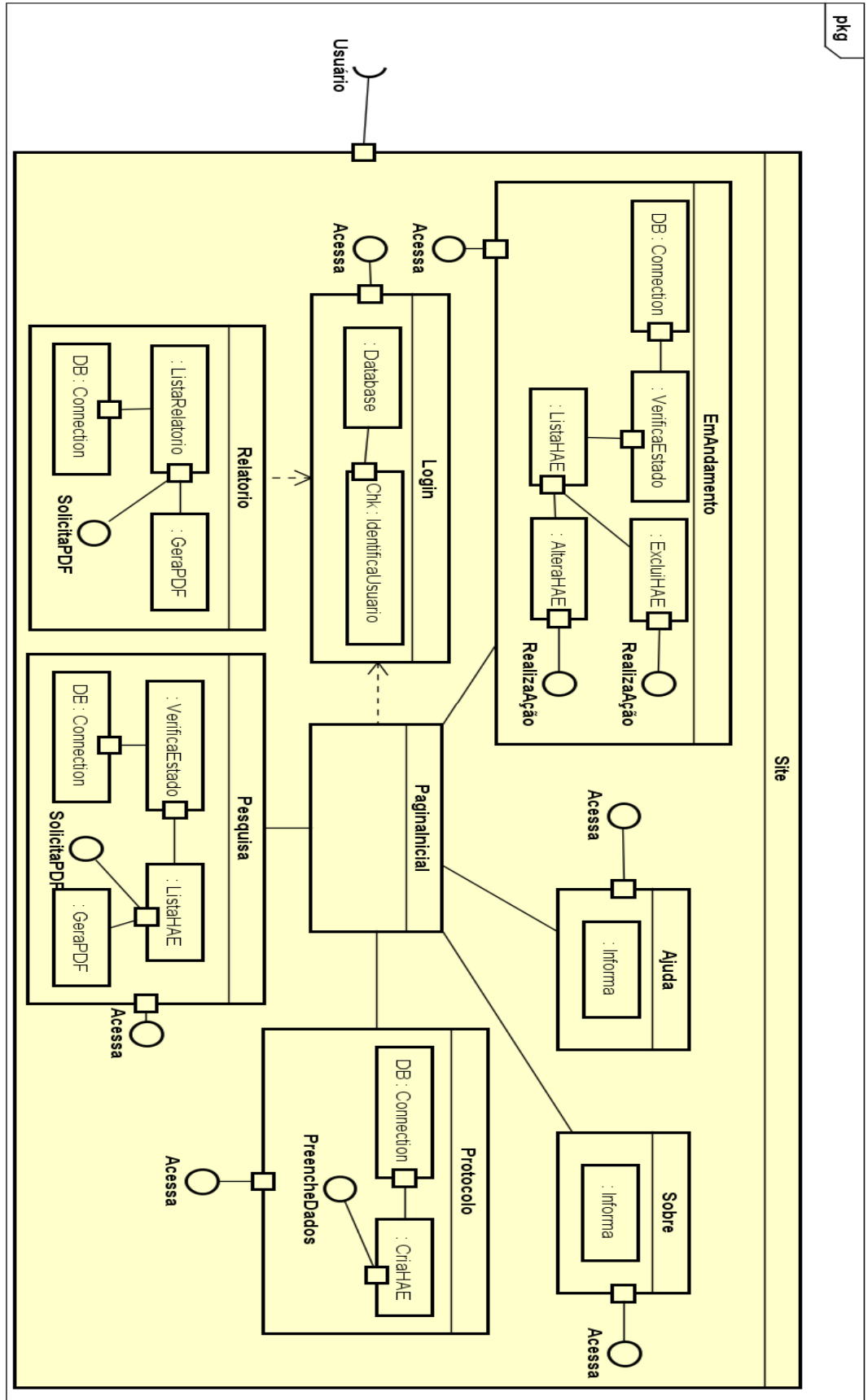
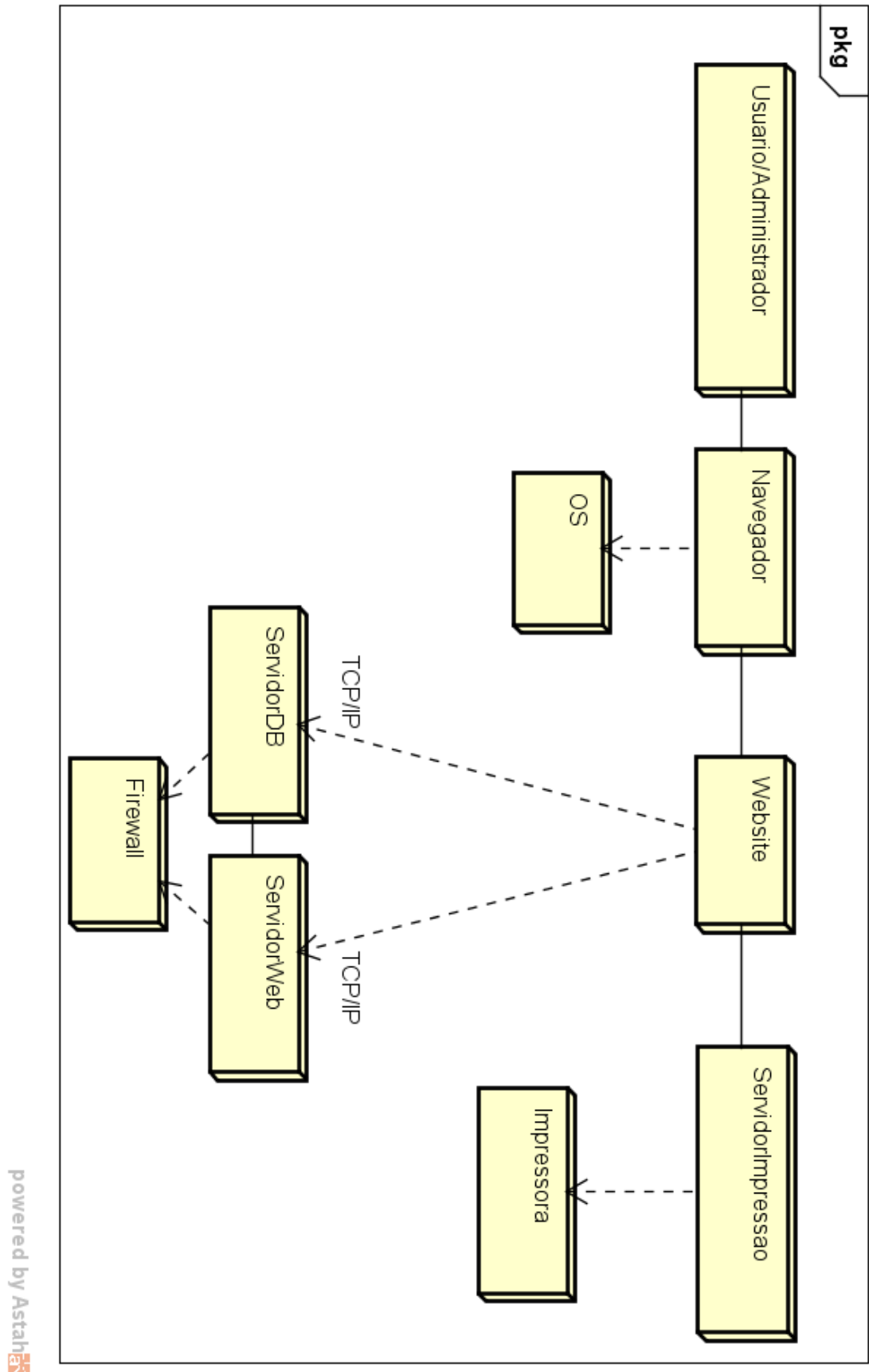


Diagrama 22 – Diagrama de instalação



Fonte: Autoria própria.

Diagrama 23 – Diagrama de objetos

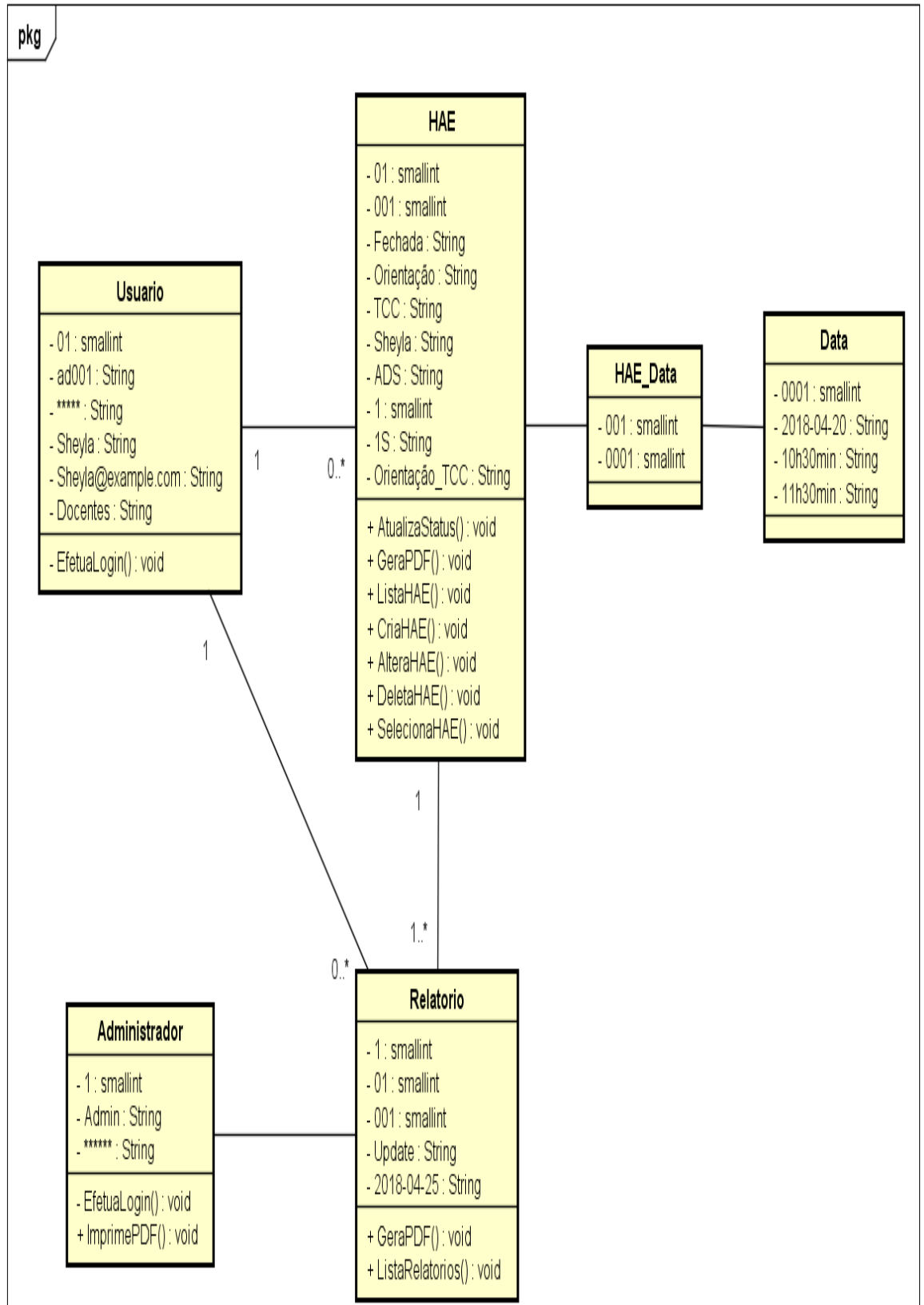
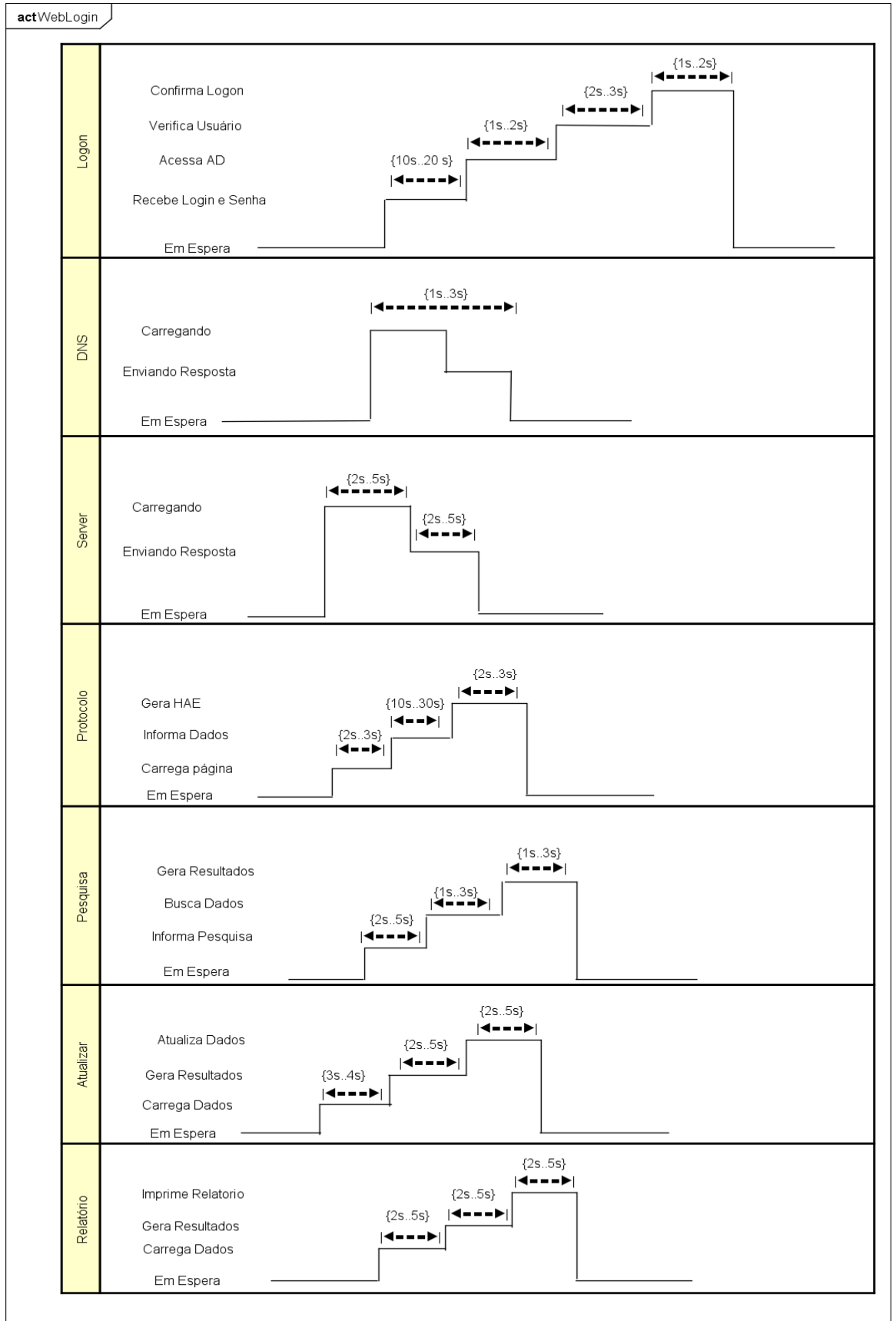


Figura 24 – Diagrama de tempo



Fonte: Autoria própria.