

Wildyson Dantas dos Santos

# **SAGH: Sistema Automatizado para Gerenciamento de Horários**

Picos - PI  
12 de Novembro de 2018

Wildyson Dantas dos Santos

## **SAGH: Sistema Automatizado para Gerenciamento de Horários**

Trabalho de Conclusão de Curso apresentado a Universidade Federal do Piauí – UFPI, Campus Senador Helvidio Nunes de Barros, como requisito parcial para obtenção de título de Bacharel em Sistemas de Informação. Orientador: Prof. Me. Francisco das Chagas Imperes Filho.

Universidade Federal do Piauí  
Campus Senador Helvídio Nunes de Barros  
Bacharelado em Sistemas de Informação

Picos - PI  
12 de Novembro de 2018

**FICHA CATALOGRÁFICA**  
**Serviço de Processamento Técnico da Universidade Federal do Piauí**  
**Biblioteca José Albano de Macêdo**

**S237s** Santos, Wildyson Dantas dos  
SAGH: Sistema Automatizado para Gerenciamento de Horários  
/ Wildyson Dantas dos Santos.– 2018.  
CD-ROM : il.; 4 ¾ pol. (45 f.)  
Trabalho de Conclusão de Curso (Curso Bacharelado em Sistemas  
de Informação) – Universidade Federal do Piauí, Picos, 2018.  
Orientador(A): Prof. Me. Francisco das Chagas Imperes Filho.

1. Sistema Web. 2. Gerenciamento de Horários. 3. Grade de  
Horários. 4. Sistema Automatizado. I. Título.

**CDD 005.7**

SAGH: SISTEMA AUTOMATIZADO PARA GERENCIAMENTO DE HORÁRIOS

WILDYSON DANTAS DOS SANTOS

Monografia aprovada como exigência parcial para obtenção do grau de Bacharel em Sistemas de Informação.

Data de Aprovação

Picos – PI, 27 de novembro de 20 18

Francisco das Chagas Imperes Filho

Prof. Msc. Francisco das Chagas Imperes Filho

Orientador

Patrícia Medyna Lauritzen de Lucena Drummond

Prof. Msc. Patrícia Medyna Lauritzen de Lucena Drummond

Membro

Patrícia Vieira da Silva Barros

Profa. Msc. Patrícia Vieira da Silva Barros

Membro

# Agradecimentos

Primeiramente agradeço a Deus por ter me guiado nessa extensa caminhada. Aos meus familiares, principalmente aos meu pais: Francisco Wildo José dos Santos e Elisa Maria Dantas dos Santos; a minha segunda mãe Maria das Mercês Da Silva; as minhas tias: Ana Joaquina, Medianeira Dantas e Rosa Dantas pelo grande apoio, incentivo e confiança que me forneceram.

Aos meus amigos mais próximos que nunca me deixaram fraquejar, por sempre me encorajar nos momentos difíceis independente da situação. Em especial ao meus amigos de infância: Raimundo Júnior e Alex William; ao meu irmão Willyson Santos pelo apoio transmitido mesmo de longe.

Aos meus colegas de curso pelos momentos de descontração e discussões sobre os mais diversos assuntos. Em especial aos colegas: Paulo Henrique, André Lucas, Tácio Moreira e João Marcos.

A todos professores do corpo docente do curso pelos conhecimentos adquiridos e, especialmente ao meu orientador, o professor Francisco Imperes, pela atenção, dedicação, paciência e transmissão de conhecimentos.

*O sucesso nasce do querer, da  
determinação e persistência em  
se chegar a um objetivo. Mesmo  
não atingindo o alvo, quem  
busca e vence obstáculos, no  
mínimo fará coisas admiráveis.*

---

José de Alencar

# Resumo

Atualmente as universidades estão cada vez mais inserindo ferramentas computacionais no meio educacional. Essas ferramentas visam facilitar a vida dos discentes, docentes e todos os envolvidos no processo de ensino aprendizagem. Essas aplicações automatizam tarefas que antes eram feitas manualmente, facilitando assim o gerenciamento de dados. Até a presente data, o Sistema Integrado de Gestão de Atividade Acadêmicas (SIGAA) da Universidade Federal do Piauí (UFPI) não possui um sistema que recomende grade horária para alunos da graduação que possuem dúvidas na escolha das disciplinas a pagar. Apesar da grande influência da tecnologia no meio educacional, sistemas para gerenciamento de horários ainda são novidade no cenário atual. Diante deste contexto, este trabalho apresenta um sistema *web* para auxiliar alunos que possuem uma ou mais reprovações no curso de Sistemas de Informação (SI) da UFPI, na etapa de escolha das disciplinas para sua grade do período. Foram utilizadas diversas etapas no desenvolvimento da aplicação e cada etapa descreve como projeto foi idealizado. Dentre outras tecnologias, o sistema utiliza técnicas de Interação Humano-Computador (IHC) para melhorar a interface e usabilidade do site. A avaliação de usabilidade obteve resultados satisfatórios, demonstrando a viabilidade de uso da aplicação dentre os alunos do curso de graduação de SI da UFPI.

**Palavras-chaves:** Sistema *Web*. Gerenciamento de Horários. Grade de Horários. Sistema Automatizado.

# Abstract

Currently universities are increasingly inserting computational tools in the educational environment. These tools aim to make life easier for students, teachers and all involved in the process of teaching learning. These applications automate tasks that were previously done manually, thus facilitating data management. To date, the Integrated System of Management of Academic Activity (SIGAA) of the Federal University of Piauí (UFPI) does not have a system that recommends hourly grading for undergraduate students who have doubts about the choice of subjects to be paid. Despite the great influence of technology in the educational environment, systems for time management are still new in the current scenario. In view of this context, this work presents a textit web system to assist students who have one or more failures in the Information Systems (SI) course at the UFPI, in the stage of choosing the subjects for their grade of the period. Several stages were used in the development of the application and each step describes how the project was designed. Among other technologies, the system uses Human-Computer Interaction (IHC) techniques to improve the interface and usability of the site. The usability evaluation obtained satisfactory results, demonstrating the feasibility of using the application among the students of the undergraduate course of SI of the UFPI.

**Key-words::** Web System. Time Management. Timetable. Automated System.

# Lista de ilustrações

Figura 1 – Crescimento da linguagem de programação <i>python</i> . . . . .	20
Figura 2 – Diagrama de caso de uso . . . . .	27
Figura 3 – Tela de cadastro . . . . .	31
Figura 4 – Tela de Login do Sistema . . . . .	31
Figura 5 – Tela da área do aluno do sistema . . . . .	32
Figura 6 – Tela de gerar grade do sistema . . . . .	33
Figura 7 – Tela de acompanhar grade do sistema . . . . .	33
Figura 8 – Estrutura de classificação das disciplinas . . . . .	34
Figura 9 – Tela de fechar período do sistema . . . . .	34
Figura 10 – Tela de atualizar informações do sistema . . . . .	35
Figura 11 – Questão 1 - Você teve dificuldade em compreender o sistema? . . . . .	36
Figura 12 – Questão 2 - O sistema é de fácil utilização? . . . . .	36
Figura 13 – Questão 3 - Você consegue compreender as informações das paginas de forma clara? . . . . .	37
Figura 14 – Questão 4 - O uso do sistema lhe passou segurança? . . . . .	37
Figura 15 – Questão 5 - Você usaria esse sistema? . . . . .	38
Figura 16 – Questão 6 - O sistema apresenta demora nas respostas ao usuário? . . . . .	38
Figura 17 – Questão 7 - Você teve dificuldade ao gerar sua grade horária? . . . . .	39
Figura 18 – Questão 8 - O sistema é preciso e eficaz? . . . . .	39

# Lista de tabelas

Tabela 1 – Requisitos Funcionais . . . . .	25
Tabela 2 – Requisitos Não Funcionais . . . . .	25
Tabela 3 – Tabela de Atores . . . . .	26
Tabela 4 – Caso de Uso Gerenciar Banco de Dados . . . . .	27
Tabela 5 – Caso de Uso Cadastro . . . . .	28
Tabela 6 – Caso de Uso Login . . . . .	28
Tabela 7 – Caso de Uso Gerar Grade . . . . .	28
Tabela 8 – Caso de Uso Fechar Período . . . . .	29
Tabela 9 – Caso de Uso Atualizar Informações . . . . .	29

# Lista de abreviaturas e siglas

SAGH	Sistema Automatizado para Gerenciamento de Horários
UFPI	Universidade Federal do Piauí
SIGAA	Sistema Integrado de Gestão de Atividades Acadêmicas
HTML	HyperText Markup Language
CSS	Style Sheet Cascading
IHC	Interação Humano-Computador
SI	Sistemas de Informação
SGBD	Sistemas Gerenciador de Banco de Dados
SQL	Structured Query Language
PHP	Hypertext Preprocessor
ASP	Active Server Pages
XHTML	Extensible HyperText Markup Language
XML	Extensible Markup Language
JS	JavaScript

# Sumário

<b>1</b>	<b>Introdução</b>	<b>13</b>
1.1	Contextualização do Problema	13
1.2	Objetivo	14
1.3	Organização do Trabalho	14
<b>2</b>	<b>Referencial Teórico</b>	<b>15</b>
2.1	Banco de Dados	15
2.2	<i>Frameworks</i>	16
2.3	Interação Humano-Computador (IHC)	18
2.4	Linguagem de Programação	19
2.5	Tecnologias <i>Web</i>	21
2.5.1	<i>Hyper Text Markup Language</i>	21
2.5.2	<i>Cascading Style Sheets</i>	22
2.5.3	<i>JavaScript</i>	22
<b>3</b>	<b>SAGH - Sistema Automatizado para Gerenciamento de Horários</b>	<b>24</b>
3.1	Requisitos do Sistema	24
3.1.1	Requisitos Funcionais	24
3.1.2	Requisitos Não Funcionais	25
3.2	Modelagem UML	26
3.2.1	Diagrama de Caso de Uso	26
3.3	Ferramentas	29
3.4	Funcionalidades do Sistema	30
3.4.1	Cadastro	30
3.4.2	Login	31
3.4.3	Área do Aluno	32
3.4.4	Gerar Grade	32
3.4.5	Fechar Período	34
3.4.6	Atualizar Informações	35
3.5	Resultados e Discussões	35
<b>4</b>	<b>Conclusão e Trabalhos Futuros</b>	<b>40</b>
	<b>Referências</b>	<b>41</b>

<b>Apêndices</b>	<b>43</b>
<b>APÊNDICE A Fomulário Utilizado nos Testes . . . . .</b>	<b>44</b>

# 1 Introdução

As tecnologias vêm evoluindo constantemente e se tornando cada vez mais essenciais para o cotidiano das pessoas, uma vez que sem os avanços tecnológicos seria muito difícil atingir grau de rapidez no processamento de informações. Uma grande vantagem das novas tecnologias é que elas podem ser aplicadas em diversas áreas como medicina, engenharia, astronomia, ecologia, educação e etc. Acompanhar as novas mudanças é essencial para que a sociedade possa crescer e se desenvolver. As novas tecnologias devem ser usadas como meios de acesso e construção de conhecimento, e não somente para a sua propagação.

É indiscutível mencionar o quanto as tecnologias estão inseridas no contexto das Instituições de Ensino Superior. Um ambiente virtual presente no âmbito acadêmico permite que todos que participam daquele meio, como professores, coordenadores e alunos, possam facilmente acessar dados relevantes e direcionados aos seus interesses. Através desse ambiente o aluno pode realizar diversas ações, como por exemplo, acessar seu histórico escolar independentemente do horário e região geográfica a qual está inserido.

Desta forma, é importante destacar a relevância do papel das ferramentas computacionais aplicadas no ambiente escolar. Esses recursos tecnológicos quando empregados nas universidades, são capazes de fornecer uma série de recursos que visam auxiliar todos os atores envolvidos no processo educacional. A partir disso, este trabalho consiste em fornecer um sistema *web* para que discentes reprovativos do curso de Sistemas de Informação possam usá-la como uma ferramenta auxiliadora para a construção da sua grade de disciplinas/horários dentro de um determinado período.

## 1.1 Contextualização do Problema

O grande número de fatores envolvidos como, a quantidade de disciplinas no semestre e carga horária excessiva fazem com que os estudantes de graduação mantenham o foco apenas em determinadas disciplinas consideradas as mais importantes e/ou difíceis de concluir. Por esse motivo, o aluno pode ter reprovações em componentes curriculares o que ocasiona alguns atrasos na sua grade curricular e, conseqüentemente, na conclusão do seu curso. No ato da realização da matrícula de cada período, o aluno pode escolher erroneamente disciplinas que retardem ainda mais a conclusão do curso.

Com isso, surgiu a ideia da construção de um sistema *web* a fim de recomendar ao aluno em cada período do curso, a melhor grade de disciplinas para cursar. O Sistema Automatizado para Gerenciamento de Horários (SAGH) é proposta como uma ferramenta que pode auxiliar os discentes a escolherem uma grade de disciplina/horários mais adequada às suas necessidades. O sistema está voltado apenas para alunos do curso de Sistemas de Informação da Universidade Federal do Piauí - Campus Picos.

## 1.2 Objetivo

O objetivo principal desse trabalho consiste em apresentar o desenvolvimento um sistema *web* a partir das informações de disciplinas ofertadas no período, visando auxiliar alunos de graduação na etapa de escolha dos componentes curriculares mais adequados para compor sua grade de disciplinas/horários.

## 1.3 Organização do Trabalho

O presente trabalho está organizado em 5 (cinco) capítulos. Após a introdução, contextualização do problema e objetivo geral, os demais capítulos estão organizados da seguinte forma:

- Capítulo 2 – Referencial Teórico: é apresentada uma revisão bibliográfica que aborda conceitos diversos, fornecendo sustentação teórica para o entendimento desse trabalho;
- Capítulo 3 – Trabalhos Relacionados: são descritos alguns trabalhos que apresentam algum grau de semelhanças com a proposta;
- Capítulo 4 – Desenvolvimento da Aplicação: discorre sobre os requisitos necessários para o desenvolvimento do sistema;
- Capítulo 5 – Conclusão: apresenta as conclusões e sugestão de trabalhos futuros.

## 2 Referencial Teórico

Para uma melhor compreensão, este capítulo apresenta o embasamento teórico dos principais conceitos utilizados no desenvolvimento deste trabalho, como a linguagem de programação, banco de dados, frameworks, tecnologias *web* e, por fim, conceitos sobre a temática Interação Humano-Computador.

### 2.1 Banco de Dados

Banco de Dados (BD) são repositórios nos quais os dados são estruturados, indexados e armazenados em arquivos. Eles permitem que as informações sejam mais fáceis de serem encontradas, armazenadas, incluídas e atualizadas (POLLO, 2017). Praticamente qualquer tipo de aplicação hoje em dia utiliza um banco de dados como repositório para o armazenamento de informações. Esse fator, torna os BDs peças essenciais para a implantação de qualquer sistema informatizado. Para manusear as informações armazenadas nos bancos de dados foram criados os Sistemas Gerenciadores de Banco de Dados (SGBDs). Esta tecnologia é capaz de fornecer um ambiente propício para realizar transações na base de dados.

Um SGBD é constituído por um conjunto de dados associados a um conjunto de programas para acesso a estes dados. O conjunto de dados, comumente chamado de banco de dados, contém, por exemplo, informações sobre uma empresa em particular. O principal objetivo de um SGBD é proporcionar um ambiente conveniente e suficiente para a recuperação e armazenamento das informações do banco de dados (KORTH; SILBERCHATZ; SUDARSHAN, 1999).

Atualmente existem diversos Sistemas Gerenciadores de Banco de Dados disponíveis no mercado. Cada um possui características próprias, tornando-os mais adequados para casos específicos. Dentre os existentes, os mais difundidos são: *MySQL*<sup>1</sup>, *PostgreSQL*<sup>2</sup>, *Firebird*<sup>3</sup>, *Oracle*<sup>4</sup>, *DB2*<sup>5</sup> e *Microsoft SQL Server*<sup>6</sup>. Os três primeiros estão disponíveis no formato *open source*. Os demais são proprietários, mas apresentam versões gratuitas com recursos limitados.

O *MySQL* foi escolhido como Sistema Gerenciador de Banco de Dados da aplicação desenvolvida por apresentar recursos adequados aos objetivos pretendidos. Através dele, é possível realizar consultas no BD com alto desempenho, característica fundamental para

<sup>1</sup> <https://www.mysql.com/>

<sup>2</sup> <https://www.postgresql.org/>

<sup>3</sup> <https://firebirdsql.org/>

<sup>4</sup> <https://www.oracle.com/br/index.HTML>

<sup>5</sup> <https://www.ibm.com/analytics/br/pt/technology/db2/>

<sup>6</sup> <https://www.microsoft.com/pt-br/sql-server/sql-server-2017>

qualquer sistema que acesse repositórios de dados. Fatores como confiabilidade, segurança, baixa exigência de processamento, compatibilidade com diferentes Sistemas Operacionais (SO) e fácil manutenção, influenciaram diretamente na escolha do SGBD do *MySQL*.

Um SGBD possui diversos recursos transparentes ao utilizador que garante que não haja inconsistências e redundâncias nos dados, dificuldade no acesso, problemas de integridade, também implementam questões de segurança relacionados ao acesso aos dados e gerenciam acessos concorrentes (KORTH; SILBERCHATZ; SUDARSHAN, 1999).

O SGBD *MySQL* é uma ferramenta que disponibiliza recursos gratuitos, porém paga para o uso comercial. Ela foi desenvolvida nas Linguagens de Programação (LP) *C* e *C++*. Funciona em diversas plataformas e possui a possibilidade de fazer conexões com várias linguagens de programação como o *PHP*<sup>7</sup>, *Java*<sup>8</sup>, *Python*<sup>9</sup>, *C*<sup>10</sup> e *C++*<sup>11</sup>. O *MySQL* é bastante usado para o estudo da linguagem *Structured Query Language* (SQL) nas universidades. Possui facilidade de integração com servidores *web* e existem várias aplicações (*YouTube*, *Facebook* e *Twitter*) que utilizam o *MySQL* como gerenciador de bases de dados devido às suas vantagens e disponibilidade de uso.

Por fim, o fator preponderante para utilizar o *MySQL* no desenvolvimento desse trabalho está relacionado à várias vantagens como, por exemplo, seu mecanismo de consulta de alto desempenho, facilidade de programação, multi-plataforma, capacidade de inserção de dados otimizados, e um forte apoio para as funções especializadas da *web*, como pesquisas rápidas em textos longos.

## 2.2 Frameworks

Segundo Jaques (2016), *framework* é um conjunto de códigos abstratos e/ou genéricos, geralmente classes, desenvolvidos em alguma linguagem de programação, que se relacionam entre si para disponibilizar funcionalidades específicas ao desenvolvedor de *software*. Os *frameworks* costumam ser divididos em módulos que podem ser interdependentes ou não. Isso quer dizer que pode-se utilizá-lo em sua totalidade, com todos os seus aparatos, ou então pode haver a possibilidade de utilizar em partes.

Os *frameworks* podem ser classificados de diversas formas. De acordo com a literatura eles pode ser classificados em dois grupos: *frameworks* de aplicação orientado a objetos e *frameworks* de componentes. A seguir são apresentadas essas duas classificações

- *Framework* de Aplicação - Segundo Fayad, Schimidt e Johnson (1999), *frameworks* de aplicação são classificados quanto ao seu escopo em *frameworks* de infra-estrutura

<sup>7</sup> <http://www.php.net/>

<sup>8</sup> [https://www.java.com/pt\\_BR/](https://www.java.com/pt_BR/)

<sup>9</sup> <https://www.python.org/>

<sup>10</sup> <http://www.open-std.org/jtc1/sc22/wg14/>

<sup>11</sup> <https://isocpp.org/std/the-standard>

de sistemas, *frameworks* de integração de *middleware* e *frameworks* de aplicações corporativas. Além da classificação por escopo, os *frameworks* podem ainda ser classificados quanto à forma usada para estendê-los. Neste caso, são conhecidos como *frameworks* caixa branca, instanciados usando herança.

- *Framework* de Componentes - De acordo com Szyperski (1997), um *framework* de componentes é uma entidade de *software* que provê suporte a componentes e segue um determinado modelo. Ele possibilita que instâncias destes componentes sejam plugadas no *framework* de componentes. Esta categoria estabelece as condições necessárias para um componente ser executado e regula a interação entre as instâncias destes componentes

Como a aplicação proposta é baseada na plataforma *web*, foram utilizados alguns *frameworks web open source* (gratuitas) disponíveis no mercado. Esses tipos de *frameworks* possuem diversas vantagens. Algumas são citadas abaixo:

- *Front-end* - parte visual do *site*;
- Bibliotecas de acesso à banco de dados;
- Gerenciamento de sessões;
- Desenvolvido dinâmico de *sites*;
- Reuso de código.

Os *softwares open source* apresenta um modelo onde, o código fonte é disponibilizado de forma livre e os serviços agregados a ele são utilizados de forma gratuita. Já o *software* proprietário é um modelo de serviço privado, cuja cópia e modificação do produto são restringidos por uma pessoa ou organização. Para a utilização do *software* proprietário é necessária autorização do proprietário.

A seguir são apresentados os *frameworks Django* e *Bootstrap*. Ambas serviram de aporte para o desenvolvimento deste trabalho.

- *Django* - O *Django* é um *framework web python* de alto nível que incentiva o rápido desenvolvimento e o *design* limpo e pragmático. Construído por desenvolvedores experientes, ele cuida de grande parte do incômodo do desenvolvimento da *web*, para que o desenvolvedor possa se concentrar exclusivamente em escrever o código do aplicativo (DJANGO, 2018). Além de todos os benefícios, o que mais influenciou na escolha desse *framework* foi a possibilidade de trabalhar com a LP *Python*.
- *Bootstrap* - *Bootstrap* é o mais popular *framework HTML (Hyper Text Markup Language)*, *CSS (Cascading Style Sheets)*, e *JS (JavaScript)* para desenvolvimento de projetos responsivos e focados para dispositivos móveis na *web*. O *Bootstrap*

oferece uma enorme variedade de *plugins*, temas compatíveis com o *framework* e possui integração com várias linguagens de programação (BATTISTELLI, 2018).

## 2.3 Interação Humano-Computador (IHC)

No decorrer da década de 1990, com o grande avanço da computação, foram criados processos para garantir um melhor desempenho do usuário com a máquina. Métodos e técnicas de usabilidade foram utilizadas para testar e projetar interfaces de sistemas interativos, garantindo uma experiência de qualidade no uso desses sistemas

Com a evolução tecnológica e diante da necessidade de foco nos estudos das novas tecnologias interativas e desafios, é adotado o termo IHC, tendo um enfoque mais amplo que a ergonomia e abordando as comunicações ou interações entre usuários e computadores (ROCHA; BARANAUSKAS, 2000).

Os processos de *design* de IHC buscam atender e servir em primeiro lugar aos usuários e aos demais envolvidos (*stakeholders*), e não às tecnologias. Boa parte desses processos é centrada no usuário. Segundo Gould e Lewis (1985) eles seguem estes princípios:

- Foco no Usuário - O *designer* deve projetar a interação e a interface de um sistema interativo para atender às necessidades dos usuários e ajudá-los a alcançarem seus objetivos. Assim, o *designer* deve estudar quem serão os usuários do sistema, seus objetivos, suas características físicas, cognitivas e comportamentais, sua formação educacional.
- *Design* Iterativo - Quando problemas forem encontrados durante os experimentos com usuários, eles deverão ser corrigidos. Isso significa que as atividades do processo de *design* devem ser iterativas, ou seja, o ciclo de projeto, avaliação com medições empíricas e reprojeção deve se repetir quantas vezes forem necessárias.
- Métricas Observáveis - O processo de *design* deve permitir a realização de experimentos (estudos empíricos) em que representantes dos usuários usem simulações ou protótipos do sistema para realizarem suas atividades e alcançarem seus objetivos.

Nielsen (2007), define o critério de usabilidade como um conjunto de fatores que qualificam quão bem uma pessoa pode interagir com um sistema interativo. Esses critérios estão relacionados com a facilidade e o esforço necessários para os usuários aprenderem e utilizarem um sistema. A seguir são apresentados os fatores definidos pelo autor.

- Facilidade de Aprendizado (*learnability*) - refere ao tempo e esforço necessário para que o usuário aprenda a utilizar o sistema com determinado nível de competência e desempenho.

- Facilidade de Recordação (*memorability*) - diz respeito ao esforço cognitivo do usuário necessário para lembrar como interagir com a interface do sistema interativo,
- Eficiência (*efficiency*) - Diz respeito ao tempo necessário para conclusão de uma atividade com apoio computacional.
- Segurança no uso (*safety*) - Se refere ao grau de proteção de um sistema contra condições desfavoráveis ou até mesmo perigosas para os usuários.
- Satisfação do usuário (*satisfaction*) - é o fator de usabilidade relacionado com uma avaliação subjetiva que expressa o efeito do uso do sistema sobre as emoções e os sentimentos do usuário.

## 2.4 Linguagem de Programação

Atualmente existem diversas linguagens de programação, cada uma com suas particularidades. A escolha de uma linguagem de programação que é usada no desenvolvimento de algum sistema, é muitas vezes relacionadas ao quão bem essas linguagens podem resolver situações particulares em determinados contextos desse *software*. As particularidades dessas linguagens estão ligadas as classificações que elas possuem. De acordo com a ACM<sup>12</sup> (*Association for Computing Machinery*) as linguagens de programação podem ser classificadas quanto a seu paradigma, nível e estrutura de tipos. A seguir são explicadas essas classificações.

- Paradigma - as linguagens de programação possuem dois tipos de paradigma, o imperativo e o declarativo. O paradigma imperativo é aquele que se preocupa em informar como as instruções ou ações serão executadas. Já o paradigma declarativo tem a preocupação em dizer ao computador o que precisa ser feito, deixando para o computador tomar a melhor decisão para essa solicitação.
- Grau - as linguagens de programação podem divididas em alto, médio e baixo nível. Linguagens de baixo nível usam instruções que somente são executadas pelo processador. As de alto nível possui um grau de abstração mais longe da máquina e mais próximo da linguagem humana. As de médio nível possui as duas características ao mesmo tempo.
- Estrutura de Tipos - possui 4 tipos de tipagem, sendo eles: fracamente, fortemente, dinamicamente e estaticamente. A fracamente muda variável conforme a situação. Na fortemente, a variável se mantém a mesma até ser descartada da memória. A dinâmica muda a variável em tempo de execução, já a estaticamente muda em tempo de compilação

---

<sup>12</sup> <https://www.acm.org/>

Foi realizado um estudo das linguagens de programação mais populares para o desenvolvimento de aplicações *web*. Dentre elas destacaram-se *PHP*, *ASP*, *Python* e *RUBY*. A linguagem escolhida para o desenvolvimento da aplicação foco desse projeto foi a LP Python. Um dos fatores que culminou em sua escolha foi a simplicidade de programação com baixa curva de aprendizagem, sintaxe clara e direta. Outra característica bastante relevante é que ela pode ser utilizada para o desenvolvimento de aplicações para ambientes multiplataforma. A seguir são ressaltados alguns motivos que levaram à escolha dessa linguagem.

Python é uma linguagem de programação orientada a objeto interativo. Ela é muito fácil de aprender, é uma linguagem de alto nível e é extremamente poderosa. Python pode ser executada em ambientes *Windows*, *linux*, *UNIX*, *Mac OS X* e é livre para usar (mesmo para fins comerciais), uma vez que é baseada em uma licença *open source*.(VELHO, 2013)

Dados indicam um crescimento diferenciado da LP Python em relação a outras linguagens populares (como Javascript, Java, C++ e PHP) nos últimos 5 anos, e consideram especificamente o comportamento dos usuários do *Stack Overflow* que estão em países desenvolvidos (pelos critérios definidos pelo Banco Mundial). A Figura 1 demonstra o crescimento da LP Python em relação a outras linguagens de programação.

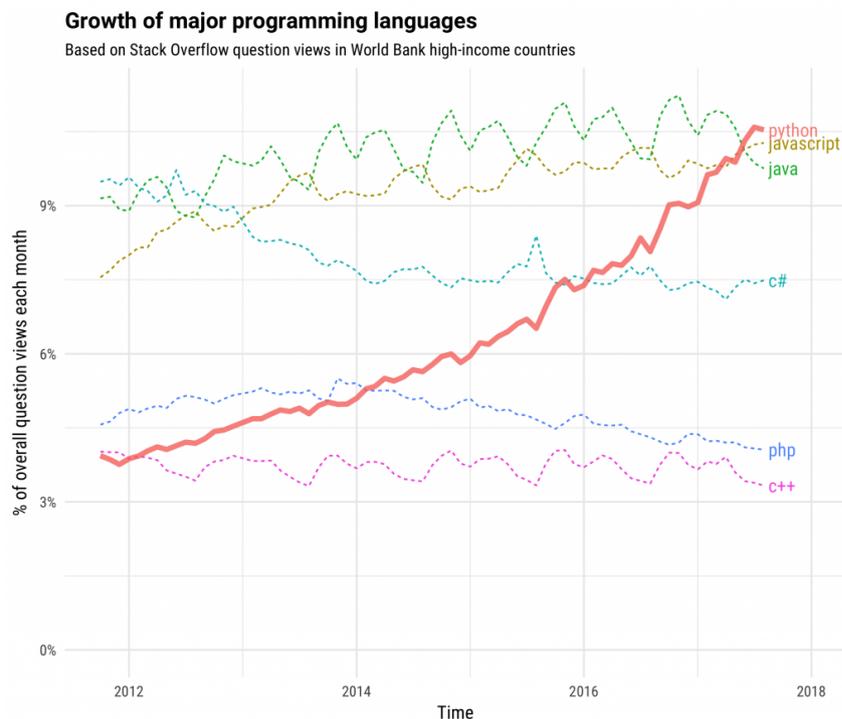


Figura 1: Crescimento da linguagem de programação *python*

Fonte: [Robinson \(2017\)](#)

Devido as características da LP *Python* mencionadas anteriormente, concluí-se que essa linguagem de programação é a mais adequada para o desenvolvimento do sistema proposto. Por fim, ela possui suporte ao SGBD e *frameworks* que são utilizados no projeto

da aplicação SAGH, funcionalidades essenciais para conclusão da proposta.

## 2.5 Tecnologias *Web*

Quando se propõe a criar um sistema *web*, vem logo na pauta dos desenvolvedores algumas tecnologias que são essenciais para execução do projeto. Estas tecnologias já estão consolidadas e envolvidas em praticamente qualquer *site* existente atualmente, como por exemplo, HTML (*Hyper Text Markup Language*), CSS (*Cascading Style Sheets*) e *JavaScript*. Para tirar toda vantagem dessas tecnologias é preciso estudá-las a fundo. A seguir são apresentados os conceitos básicos referentes às tecnologias supracitadas.

### 2.5.1 *Hyper Text Markup Language*

Grande parte do código que roda na *internet* é o *Hyper Text Markup Language* (HTML) ou Linguagem de Marcação (LM) para Hipertextos). Hipertextos podem ser considerados conjuntos de elementos (áudio, palavras, imagens, arquivos etc.) interligados. Este conjunto de ligações forma a grande Rede Mundial de Computadores. A LM HTML tem como base o uso de *tags*, que são estruturas que consistem em breve instruções a serem renderizadas no navegador do usuário final. As *tags* têm uma marca de início, representada pelo símbolo "{", e uma de fim, representada pelo símbolo "}"(SCHROEDER, 2012).

Atualmente o HTML está na versão 5, lançada em outubro de 2014, onde é possível notar diferenças em relação as versões anteriores. Algumas dessas mudanças são: padronização e acolhimento, melhor tratamento de exceção, mais *tags* para substituir *scripts*, independência de plataforma, redução da necessidade de *plugins* externos e a possibilidade de utilizar estruturas de código Orientado a Objetos (OO). A versão do HTML5 é suportado por todos os navegadores e esta versão é usado neste trabalho.

Almeida et al. (2012) define alguns recursos e características que o HTML5 oferece ao desenvolvedor, estes sendo:

- Semântica: permite descrever mais precisamente o seu conteúdo.
- Conectividade: permite uma comunicação com o servidor de formas modernas e inovadoras
- *Offline* e armazenamento: permite que páginas *web* armazenem dados localmente do lado do cliente e opere de forma *offline* mais eficientemente.
- Multimídia: capacidade de suportar elementos nativos de áudio e vídeo significa que os usuários não precisarão baixar *plugins* adicionais para exibir conteúdos multimídias.

- Gráficos e efeitos 2D/3D: viabiliza um leque diversificado de opções de representação gráfica.
- Performance e integração: fornece grande otimização de velocidade e melhor utilização do *hardware* do computador.
- Acesso ao dispositivo: viabiliza a utilização de diversos métodos e dispositivos de entrada e saída de dados.
- Estilização: permite aos autores a escrita de temas mais sofisticados.

### 2.5.2 *Cascading Style Sheets*

Quando o HTML (*Hyper Text Markup Language*) foi criado, a intenção não era formatar informação, mas apenas exibi-la. À medida que o HTML foi evoluindo, foram incluídas novas funcionalidades no intuito de controlar algumas aparências do documento. Devido a esse fato, a linguagem ficou muito mais complexa e de difícil compreensão. Observando toda essa dificuldade foi criada uma linguagem específica para formatação de texto. Essa linguagem é conhecida como *Cascading Style Sheets* (CSS) ou Folhas de Estilo em Cascata.

CSS é uma "folha de estilo" composta por "camadas" e utilizada para definir a apresentação (aparência) em páginas da *internet* que adotam para o seu desenvolvimento linguagens de marcação (como XML, HTML e XHTML). O CSS define como serão exibidos os elementos contidos no código de uma página da *internet*. Sua maior vantagem é efetuar a separação entre o formato e o conteúdo de um documento (PEREIRA, 2009).

A importância do uso do CSS em qualquer *site* é extremamente relevante. Qualquer interface de utilização de sistema deve apresentar as informações da forma simples e objetiva. Devido a isso, a interface deve estar relacionada principalmente aos processos de *designs* de IHC presente na seção 2.3. Para este projeto, foi utilizado o CSS3 que conta com uma gama de recursos que vão ajudar a deixar o sistema mais apresentável e amigável para o usuário final.

### 2.5.3 *JavaScript*

O *JavaScript* é uma linguagem de programação interpretada onde o código fonte nessa linguagem é executado por um programa de computador chamado interpretador, que em seguida é executado pelo sistema operacional ou processador. O principal papel dessa linguagem é deixar processos das páginas *web* mais dinâmicos aos programadores. O uso do JS alinhado ao CSS (*Cascading Style Sheets*) em um sistema é capaz de atender às necessidades dos clientes deixando a navegação mais agradável e simples.

O JS (*JavaScript*) é extremamente eficiente na transformação e processamento dos dados antes de enviar para o servidor. Um exemplo clássico do uso do JS é na validação de formulários, onde é possível criar um código para validar os dados antes de enviar para o

servidor. Isso ajuda muito no desempenho do sistemas *web*, pois o processamento é realizado no lado do cliente. Além disso, JS consegue interagir com elementos HTML (*Hyper Text Markup Language*) e CSS, ajudando a tornar o *site* mais dinâmico e compreensível para os usuários.

Luiz (2018) define algumas características e recursos que o JavaScript possui, estes são:

- Interações - por meio da utilização da linguagem JavaScript, é possível efetuar interações com elementos HTML, permitindo que eles sejam alterados de maneira dinâmica. Além também de interações com regras CSS.
- Sistemas - o JavaScript apresenta sistemas de autocomplete (semelhante ao que se encontra quando se digita uma palavra no Google), completando, assim, termos que podem ou não já estarem salvos no servidor.
- Validação - a linguagem JavaScript permite a validação de formulários, evitando que o usuário não necessite recarregar uma página inteira devido, por exemplo, um erro de digitação.
- Funções - o JavaScript é uma linguagem que conta com funções de primeira classe, ou seja, podem ser enviadas como argumentos, atribuídos como as variáveis, entre outras funções.
- Métodos - ao contrário de outras linguagens de programação destinadas para objetos, com o JavaScript não há diferenciação entre a função e a definição de um método na hora da programação.

Através desse estudo foi possível conhecer todas as características, vantagens e justificativas dos autores a respeito da temática que foi desenvolvida neste projeto. Essa base teórica é de grande importância, pois só por intermédio dela, foi possível extrair o conhecimento necessário para a realização deste trabalho.

## 3 SAGH - Sistema Automatizado para Gerenciamento de Horários

O sistema desenvolvido neste projeto foi exclusivamente construído para a plataforma *web*. Denominado "Sistema Automatizado para Gerenciamento de Horários (SAGH)", tem como principal objetivo auxiliar alunos que possuem uma ou mais reprovações no curso de Sistemas de Informação da Universidade Federal do Piauí, na etapa de escolha das disciplinas para sua grade do período. Os detalhes do desenvolvimento são apresentados a seguir, dentre eles pode-se destacar: levantamento dos requisitos, modelagem UML<sup>1</sup>, ferramentas, modelagem do banco de dados e funcionalidades do sistema.

### 3.1 Requisitos do Sistema

O levantamento de requisitos é uma das etapas mais importantes no desenvolvimento de sistemas. É fundamental definir os requisitos corretamente, pois a má definição dos tais requisitos podem acarretar em problemas no *software*. Através do levantamento de requisitos, é possível visualizar como o sistema se comportará e isso ajuda a idealizar qualquer sistema de *software*. Com base nas informações levantadas, as próximas seções apresentam os requisitos funcionais e não funcionais que foram avaliados para a implementação da aplicação foco deste trabalho.

#### 3.1.1 Requisitos Funcionais

Os Requisitos Funcionais (RF) descrevem o comportamento do *software* desejado em termos de tarefas e serviços. Esses requisitos funcionais definem recursos específicos que devem ser fornecidos pelo sistema. Essa etapa de levantamento de requisitos é feita geralmente através de um questionário. A Tabela 1, apresenta os detalhes de cada RF do sistema, contando com identificador, descrição e possíveis dependências a outros requisitos.

---

<sup>1</sup> <http://www.uml.org/>

Tabela 1: Requisitos Funcionais

Identificador	Descrição	Depende de
RF01	O sistema deverá ter um usuário administrador.	
RF02	Deve ter a opção de cadastrar os dados dos alunos.	
RF03	O sistema deverá ter a opção de autenticação por login.	RF02
RF04	O sistema deverá ter a opção para que os alunos possam salvar sua grade.	RF03
RF05	O sistema deverá ter uma tela para que alunos possam acompanhar o progresso de sua grade.	RF03, RF08
RF06	O sistema deverá ter a opção de alterar os dados alunos.	RF03
RF07	O sistema deverá ter a opção de concluir as disciplinas daquele semestre em que ele esta.	RF03
RF08	O sistema deverá atualizar a grade horária de acordo com as disciplinas já concluídas.	RF03

### 3.1.2 Requisitos Não Funcionais

Os Requisitos Não Funcionais (RNF) são as características de qualidade que um sistema deve ter em relação as suas funcionalidades. Na Tabela 2 são demonstrados os RNF que foram utilizados na implementação do sistema *web*.

Tabela 2: Requisitos Não Funcionais

Identificador	Descrição	Categoria
RNF01	O sistema irá manter um nível básico de segurança, impedindo que as informações dos usuários sejam acessadas indevidamente.	Segurança de Acesso
RNF02	Os sistemas de bancos de dados só poderão ser acessados e remodelados por usuários <i>ROOT</i> .	Segurança de Acesso
RNF03	O sistema possuirá uma interface simples e intuitiva, se utilizando dos padrões de IHC.	Facilidade de Aprendizado
RNF04	O <i>site</i> deve estar funcionando de maneira integral, focando na alta disponibilidade, deixando de funcionar apenas em manutenções e casos excepcionais.	Confiabilidade e Disponibilidade
RNF05	Todos os usuários cadastrados deverão conseguir acessar suas contas a qualquer hora e em qualquer lugar, mediante conexão via <i>Internet</i> .	Confiabilidade e Disponibilidade
RNF06	O sistema não apresentará aos usuários quaisquer dados de cunho privativo, atendendo às normas legais vigentes na UFPI.	Requisitos Éticos e Legais
RNF07	O sistema deve ser leve e rápido, permitindo até para as mais lentas conexões um nível mínimo de aceitabilidade de resposta.	Eficiência em relação ao Tempo
RNF08	O site deverá funcionar de maneira correta e previsível em todas as plataformas que possuírem acesso à <i>Internet</i> .	Portabilidade

Esta seção apresentou toda a documentação que foi levantada para a criação do sistema. Apresentando detalhadamente cada requisito e característica de qualidade que o sistema deve entregar ao usuário final.

## 3.2 Modelagem UML

A UML<sup>2</sup> (*Unified Modeling Language*) ou Linguagem Unificada de Modelagem é uma linguagem padrão para modelagem orientada a objetos. Essa linguagem ajuda a visualizar o roteiro de comunicação de objetos de um sistema. Ela possui diversos modelos de diagramas que ajudam o desenvolvedor de *software* a visualizar de forma ampla todos os aspectos daquele sistema. O diagrama que foi escolhido para representar as ações dos usuários no sistema desenvolvido foi o diagrama de caso uso.

### 3.2.1 Diagrama de Caso de Uso

O diagrama de caso de uso descreve a ação das atividades que um ator irá desempenhar naquele sistema. Bastante utilizados, este diagrama possui a vantagem de demonstrar do ponto de vista do usuário a sua interação com cada funcionalidade do sistema sem se aprofundar em detalhes técnicos. Conforme a Tabela 3, foram identificados dois atores que interagem com o sistemas.

Tabela 3: Tabela de Atores

<b>Ator</b>	<b>Descrição</b>
Usuário Administrador	Este usuário será responsável por manter as informações da tabela de disciplinas do banco de dados atualizadas.
Usuário do Sistema	São usuários que poderão acessar as funcionalidades do sistema, como cadastro, geração de grade, fechamento de grade no período, atualização de dados e acompanhamento da grade

Na Figura 2, é apresentada a descrição das ações que o usuário administrador e o usuário final possuem no sistema.

---

<sup>2</sup> <http://www.uml.org/>

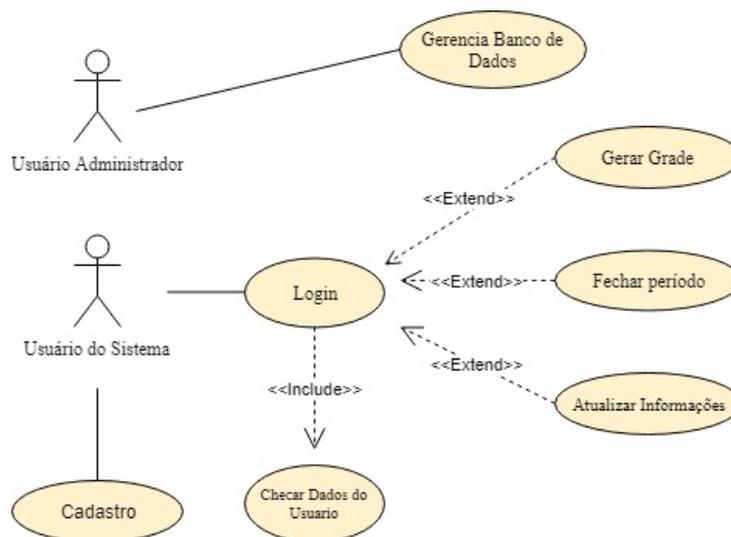


Figura 2: Diagrama de caso de uso

Os atores são representados pelos bonecos e as setas indicam o relacionamento aos balões que são os casos de uso. O vínculo de inclusão (*include*), define que sempre que um caso de uso **A** for executado, o caso de uso **B** deve ser obrigatoriamente executado em seguida. Já o o vínculo de extensão (*extend*) define, que após a execução do caso de uso **A**, o caso de uso **B** poderá ou não ser executado em seguida.

Para uma melhor compreensão da Figura 2, será feita uma análise mais detalhada com informações relevantes sobre a execução de cada caso de uso descrito.

Tabela 4: Caso de Uso Gerenciar Banco de Dados

<b>Nome do caso de uso</b>	Gerenciar Banco de Dados
<b>Objetivo</b>	Manter os dados das disciplinas atualizados,
<b>Atores</b>	Usuário Administrador
<b>Pré-Condições</b>	Deve esta logado no sistema.
<b>Pós-Condições</b>	
<b>Fluxo Principal</b>	
<b>Ações do ator</b>	<b>Ações do Sistema</b>
	1. O sistema solicita os dados do usuário.
2. Usuário insere os dados	3. O sistema verifica os dados do usuário.
	4. O sistema apresenta os dados das disciplinas.
5. Usuário Solicita a disciplina	6. O sistema solicita os dados da disciplina.
7. O usuário insere os dados	8. O sistema verifica os dados e atualiza as informações

Tabela 5: Caso de Uso Cadastro

<b>Nome do caso de uso</b>	Cadastro
<b>Objetivo</b>	Efetuar cadastro do usuário.
<b>Atores</b>	Usuário do sistema.
<b>Pré-Condições</b>	
<b>Pós-Condições</b>	
<b>Fluxo Principal</b>	
<b>Ações do ator</b>	<b>Ações do Sistema</b>
	1. O sistema solicita os dados do usuário.
2. Usuário insere os dados	3. O sistema insere os dados do usuário no banco de dados.

Tabela 6: Caso de Uso Login

<b>Nome do caso de uso</b>	Login
<b>Objetivo</b>	Efetuar login no sistema.
<b>Atores</b>	Usuário do sistema.
<b>Pré-Condições</b>	Deve possuir cadastro.
<b>Pós-Condições</b>	
<b>Fluxo Principal</b>	
<b>Ações do ator</b>	<b>Ações do Sistema</b>
	1. O sistema solicita os dados do usuário.
2. Usuário insere os dados	3. O sistema verifica os dados do usuário.
	4. O sistema apresenta a interface principal do sistema.

Tabela 7: Caso de Uso Gerar Grade

<b>Nome do caso de uso</b>	Gerar Grade
<b>Objetivo</b>	Gerar sua grade horária.
<b>Atores</b>	Usuário do sistema
<b>Pré-Condições</b>	Deve estar logado no sistema.
<b>Pós-Condições</b>	
<b>Fluxo Principal</b>	
<b>Ações do ator</b>	<b>Ações do Sistema</b>
	1. O sistema solicita as disciplinas que o usuário já concluiu.
2. Usuário insere os dados	3. O sistema verifica os dados do usuário.
	4. O sistema apresenta a grade horária gerada.

Tabela 8: Caso de Uso Fechar Período

<b>Nome do caso de uso</b>	Fechar Período
<b>Objetivo</b>	Concluir disciplinas já pagas.
<b>Atores</b>	Usuário do sistema
<b>Pré-Condições</b>	Deve estar logado no sistema.
<b>Pós-Condições</b>	
<b>Fluxo Principal</b>	
<b>Ações do ator</b>	<b>Ações do Sistema</b>
	1. O sistema apresenta as disciplinas que o usuário está pagando.
2. Usuário escolhe quais disciplinas já pagou.	3. O sistema verifica os dados do usuário.
	4. O sistema apresenta a nova grade horária gerada.

Tabela 9: Caso de Uso Atualizar Informações

<b>Nome do caso de uso</b>	Atualizar Informações
<b>Objetivo</b>	Atualizar as informações do aluno.
<b>Atores</b>	Usuário do sistema
<b>Pré-Condições</b>	Deve estar logado no sistema.
<b>Pós-Condições</b>	
<b>Fluxo Principal</b>	
<b>Ações do ator</b>	<b>Ações do Sistema</b>
	1. O sistema apresenta os dados que o usuário pode alterar.
2. Usuário insere os dados.	3. O sistema verifica os dados do usuário.
	4. O sistema atualiza os dados no banco de dados.

Esta seção apresentou as ações que os atores podem realizar no sistema. Foi relatado detalhadamente como o sistema reage a cada solicitação dos atores, exibindo o fluxo de funcionamento do sistema a cada solicitação de uma funcionalidade requisitada.

### 3.3 Ferramentas

Esta seção apresenta todas as ferramentas que foram utilizadas para o desenvolvimento da aplicação foco deste trabalho. Dentre as tecnologias utilizadas, estão: a linguagem de programação *Python*, o Sistema Gerenciador de Banco de Dados *MySQL*, a *framework Django* para o desenvolvimento *web* e a *framework Bootstrap* para formatar o *design (front-end)* do sistema. Para utilizar essas ferramentas, foram escolhidos alguns ambientes de desenvolvimento que são: *Sublime Text*<sup>3</sup>, *PyCharm*<sup>4</sup> e *MySQL Workbench*<sup>5</sup>. Vale ressaltar

<sup>3</sup> <https://www.sublimetext.com/>

<sup>4</sup> <https://www.jetbrains.com/pycharm/>

<sup>5</sup> <https://www.mysql.com/products/workbench/>

que todas as ferramentas utilizadas possuem licença gratuita, exceto o *PyCharm*.

Após a instalação de todas essas ferramentas, foi dado início a criação das tabelas do banco de dados *MySQL*. O *MySQL Workbench* foi bastante utilizado no processo de construção dessas tabelas, por ser capaz de oferecer uma gama de recursos que facilitam a administração do banco. Uma das grandes vantagens em utilizar o *MySQL Workbench*, foi exatamente a capacidade de alterar diversos aspectos na tabela do banco sem precisar digitar o código SQL (*Structured Query Language*) manualmente.

Depois da criação do banco de dados, foi iniciada a etapa mais complexa do projeto que foi a programação do sistema. A codificação foi escrita na linguagem de programação *Python* dentro do ambiente de desenvolvimento *PyCharm*. Além da sintaxe simples e limpa que o *Python* oferece ao programador de *software*, ele também possui uma variedade de bibliotecas que podem ser usadas para facilitar a criação do algoritmo.

Posteriormente a criação do banco de dados e o algoritmo base da aplicação, foi iniciado a programação *web* do sistema. A *framework Django* foi a tecnologia utilizada no desenvolvimento. O *design* do *site* teve como base algumas técnicas de IHC (Interação Humano-Computador). O *Sublime Text* foi o responsável pelo gerenciamento e criação de dos arquivos de configuração do *Django*, das páginas HTML, dos arquivos *JavaScript*, dos arquivos CSS (*Cascading Style Sheets*) e dos arquivos *Python*.

O Sistema Automatizado Gerenciador de Horários (SAGH) foi testado em um microcomputador com as seguintes especificações: sistema operacional *Windows 10 Home Single Language*, modelo *Aspire E 15 E5-571-320G*, processador *Intel Core 3*, 500 GB de armazenamento e memória ram de 4 GB.

## 3.4 Funcionalidades do Sistema

Esta seção aborda detalhadamente as principais funcionalidades do SAGH. A aplicação foi projetada para garantir que o usuário possa administrar sua grade de horários sem grandes dificuldades. O funcionamento do sistema é apresentado e dividido em interfaces (Telas) a saber: Cadastro, Login, Gerar Grade, Área do Aluno, Fechar Período e Atualizar Informações.

### 3.4.1 Cadastro

Esta funcionalidade é responsável por cadastrar as informações do aluno. O discente irá inserir seus dados pessoais no formulário para que o sistema possa registrá-lo. Este cadastro serve exclusivamente para que os alunos possam fazer login no sistema e posteriormente gerenciar sua grade horária. Todos os campos dessa página são obrigatórios e estão validados contra eventuais falhas. Essa característica é válida também para a funcionalidade *Login*. A página de cadastro pode ser visualizada na Figura 3.

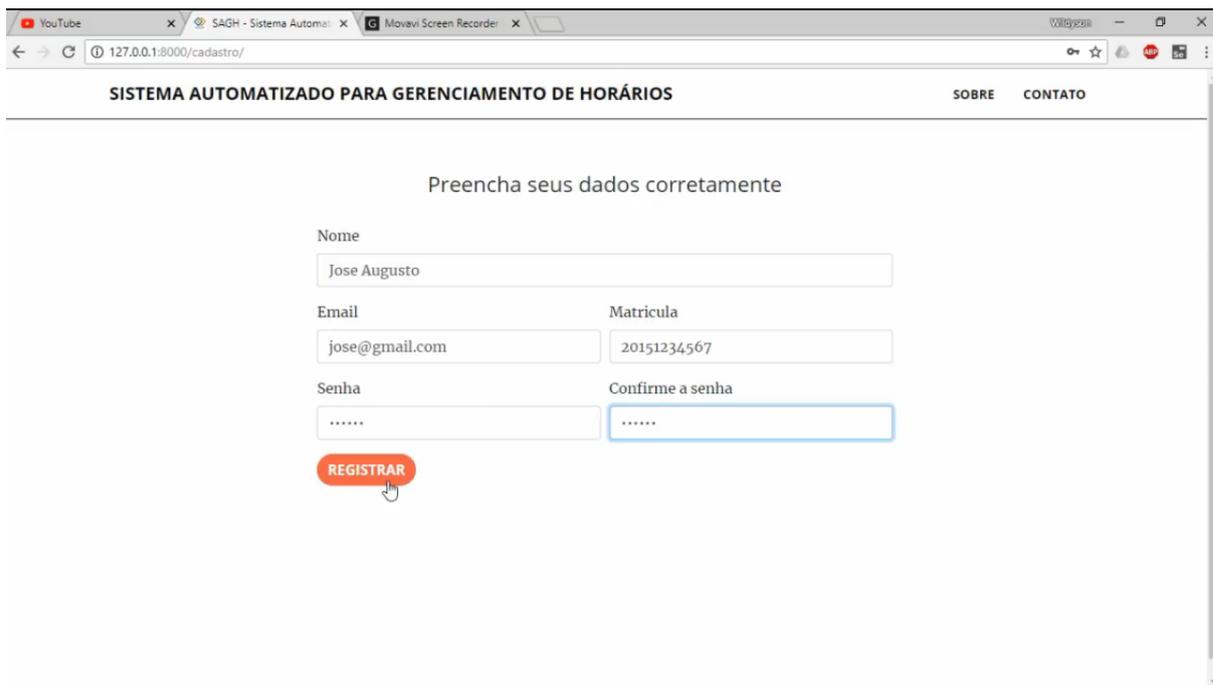


Figura 3: Tela de cadastro

### 3.4.2 Login

Esta funcionalidade é responsável por autenticar as informações do aluno para que ele possa ter acesso ao sistema. O discente irá inserir seus dados pessoais de matricula e senha para autenticação e permissão de acesso. Caso o aluno ainda não tenha cadastro, é disponibilizado um *link* na página para redirecioná-lo à página de cadastro. A página de *login* pode ser visualizada na Figura 4.

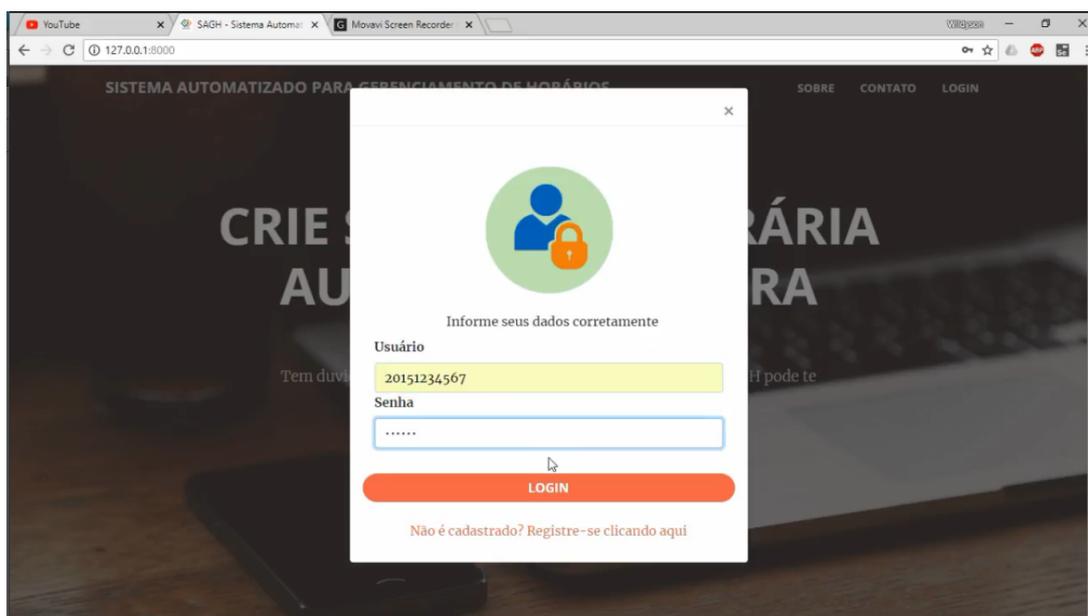


Figura 4: Tela de Login do Sistema

### 3.4.3 Área do Aluno

Esta funcionalidade é responsável por exibir a área do aluno com todas as funções que ele pode realizar. Para ter acesso a essa área o aluno deve estar devidamente autenticado. Através dessa funcionalidade o aluno pode gerenciar sua própria grade horária e acompanhar o progresso da mesma. Caso o aluno tenha alguma informação pessoal errada, é disponibilizada a função de atualizar informações. A Figura 5 apresenta uma página onde o discente ainda não criou sua grade.



Figura 5: Tela da área do aluno do sistema

### 3.4.4 Gerar Grade

Esta funcionalidade é responsável por exibir todos os componentes curriculares do curso de Sistemas de Informação da Universidade Federal do Piauí - Campus Senador Helvidio Nunes de Barros (CSHNB). Para ter acesso a essa área o aluno deve estar devidamente autenticado. Através dessa funcionalidade o aluno irá selecionar as disciplinas que ele já concluiu do curso e, ao fim deve clicar no botão gerar grade para que o sistema possa processar as informações. Após o processamento da grade, a página é redirecionada para a página inicial que pode ser visualizada na Figura 7. A página de gerar grade pode ser visualizada na Figura 6.

1º PERIODO	
<input checked="" type="checkbox"/>	ALGORITMOS E PROGRAMACAO I
<input checked="" type="checkbox"/>	LOGICA PARA COMPUTACAO
<input checked="" type="checkbox"/>	METODOLOGIA CIENTIFICA
<input checked="" type="checkbox"/>	SISTEMAS DE INFORMACAO I
<input checked="" type="checkbox"/>	GESTAO EMPRESARIAL
<input checked="" type="checkbox"/>	SEMINARIO DE INTRODUCAO AO CURSO
2º PERIODO	
<input checked="" type="checkbox"/>	ALGORITMOS E PROGRAMACAO II
<input type="checkbox"/>	CIRCUITOS DIGITAIS
<input checked="" type="checkbox"/>	MATEMATICA DISCRETA
<input checked="" type="checkbox"/>	SISTEMAS DE INFORMACAO II
<input checked="" type="checkbox"/>	CALCULO DIFERENCIAL E INTEGRAL I
<input checked="" type="checkbox"/>	PROGRAMACAO LOGICA

Figura 6: Tela de gerar grade do sistema

SUA GRADE					
CÓDIGO	DISCIPLINA	PRÉ-REQUISITO	HORÁRIO	NÍVEL	CREDITOS
20 creditos					
SINF/CSHNB016	ENGENHARIA DE SOFTWARE I	0	2M56 6M34	3	4
SINF/CSHNB020	SISTEMAS OPERACIONAIS	0	4M56 3T34	3	4
SINF/CSHNB018	ESTRUTURAS DE DADOS I	SINF/CSHNB007	35M56 3M56 5T34	3	4
SINF/CSHNB019	PROGRAMACAO ORIENTADA A OBJETOS I	SINF/CSHNB007	34T56 35T56	3	4
SINF/CSHNB017	ESTADISTICA	0	24M34	3	4
22 creditos					
SINF/CSHNB008	CIRCUITOS DIGITAIS	SINF/CSHNB003	35T34 35T56	2	4
SINF/CSHNB027	REDES DE COMPUTADORES I	SINF/CSHNB020	24M56	4	4
SINF/CSHNB015	BANCO DE DADOS I	0	24T56	3	4
SINF/CSHNB025	PROGRAMACAO ORIENTADA A OBJETOS II	SINF/CSHNB010	4T34 5M34	4	4
SINF/CSHNB024	INTERACAO HUMANO COMPUTADOR	0	2M34 3M56	4	4

Figura 7: Tela de acompanhar grade do sistema

Uma dos pontos mais importantes dessa funcionalidade é a rapidez que o algoritmo processa os dados e, conseqüentemente, gera a grade do aluno. Essa funcionalidade classifica as disciplinas de acordo com seu grau de importância, definindo pesos para todas as disciplinas. A Figura 8 apresenta o fluxo de classificação das disciplinas.

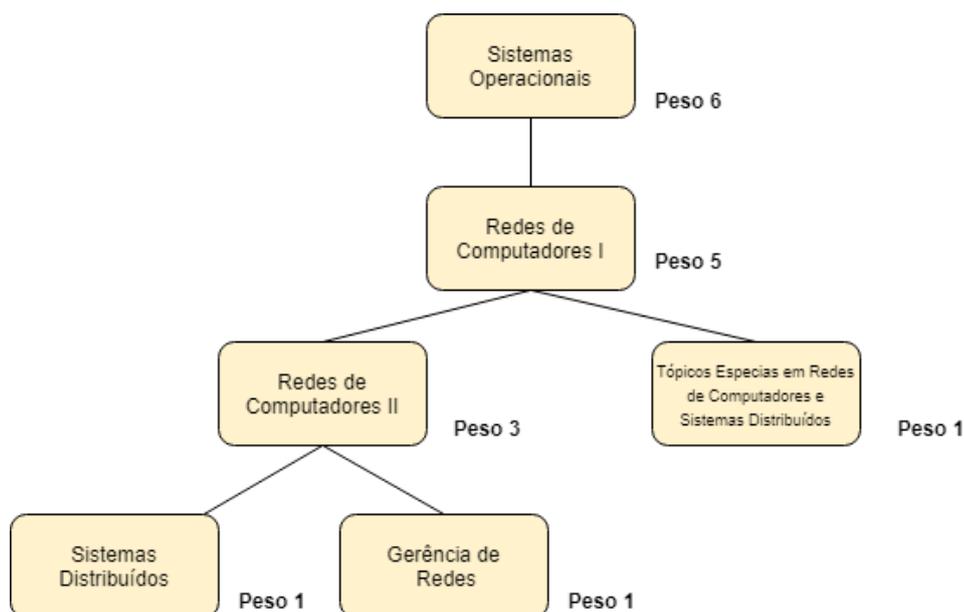


Figura 8: Estrutura de classificação das disciplinas

### 3.4.5 Fechar Período

Esta funcionalidade é responsável por exibir os componentes curriculares daquele período que o discente está cursando. Através dessa funcionalidade o aluno irá selecionar as disciplinas que ele já concluiu da sua grade e ao fim deve pressionar o botão "OK" para processar as informações. Após o processamento da grade, a página é redirecionada para a página inicial. A página de fechar período pode ser visualizada na Figura 9.

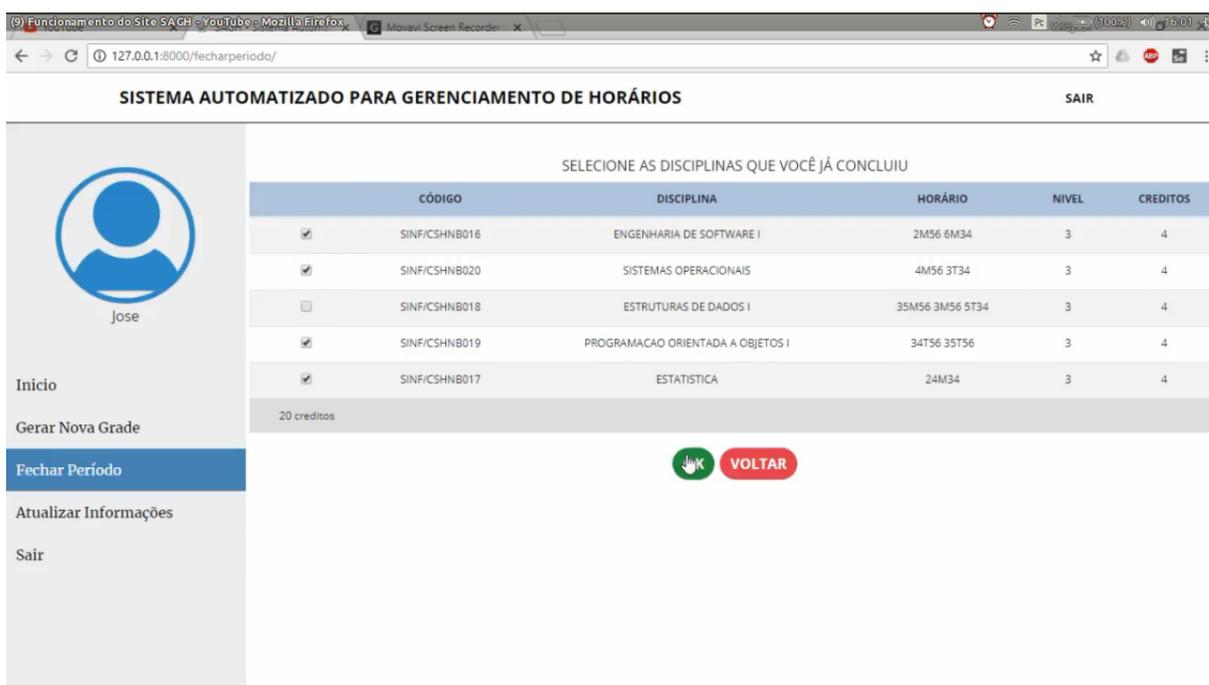


Figura 9: Tela de fechar período do sistema

### 3.4.6 Atualizar Informações

Esta funcionalidade é responsável por atualizar as informações do discente. Através dessa funcionalidade o aluno irá alterar os campos que possam conter erros e em seguida pressionar o botão "Salvar" para atualizar as informações. Vale ressaltar que a informação do campo matrícula não pode ser alterada. Após o processamento dessa funcionalidade, a página é redirecionada para a página inicial. A página de Atualizar Informações pode ser visualizada na Figura 10.

A imagem mostra a interface de usuário do sistema SAGH. No topo, há uma barra de navegação com o título "SISTEMA AUTOMATIZADO PARA GERENCIAMENTO DE HORÁRIOS" e o botão "SAIR". Abaixo, há uma barra lateral com um menu contendo as opções: "Início", "Gerar Nova Grade", "Fechar Período", "Atualizar Informações" (destacado em azul) e "Sair". O conteúdo principal da tela é o formulário "Altere os dados que desejar". O formulário contém os seguintes campos: "Nome" (preenchido com "Jose Augusto"), "Email" (preenchido com "jose@gmail.com"), "Nova Senha" (preenchido com "informe a nova senha") e "Confirme a nova senha" (preenchido com "confirme a nova senha"). Um botão "SALVAR" em vermelho está localizado abaixo dos campos de senha.

Figura 10: Tela de atualizar informações do sistema

Este capítulo descreveu como o projeto foi desenvolvido, apresentando a documentação do sistema, as ferramentas utilizadas e as principais funcionalidades do sistema. A partir desse contexto, foi possível ter uma compreensão detalhada de como o sistema foi criado e como suas funções funcionam.

## 3.5 Resultados e Discussões

Esta seção apresenta os resultados e discussões obtidas ao final do trabalho. Foram aplicados questionários aos estudantes do curso de Sistemas de Informação da Universidade Federal do Piauí - Campus Senador Helvidio Nunes de Barros, no período de 30/10/2018 a 01/11/2018. O teste foi realizado da seguinte maneira: cada aluno testava o sistema e em seguida respondia as questões da avaliação. Os resultados foram obtidos através da análise das respostas dadas a cada pergunta.

Para a análise avaliativa foram levadas em consideração amostras de 15 alunos aleatórios do curso de Sistemas de Informação. Cada avaliação é composta por oito perguntas que são baseadas em aspectos visuais, intuitivos e coerentes do sistema. As perguntas são objetivas e possuem escala de 1 até 3. A seguir são apresentados os resultados obtidos através das respostas dos questionários avaliativos constantes no Apêndice A.

A primeira pergunta foi: "Você teve dificuldade em compreender o sistema?". Como pode ser visto na figura 11, 93,3% dos alunos, responderam que não possuíram dificuldades em compreender o sistema. Apenas 1 aluno que corresponde a 6,7% do total, respondeu que possuiu um pouco de dificuldade. Analisando esses dados, é possível concluir que o sistema é de fácil compreensão, mas pode ser melhorado.

### Você teve dificuldade em compreender o sistema?

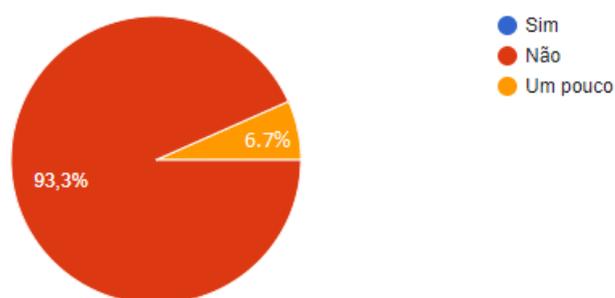


Figura 11: Questão 1 - Você teve dificuldade em compreender o sistema?

Com relação a presteza do sistema, a segunda pergunta foi: "O sistema é de fácil utilização?". Como pode ser visto na figura 12, 93,3% dos alunos responderam que acreditam que o sistema é de fácil utilização. Apenas 6,7% dos estudante acreditam que o sistema é um pouco fácil de utilizar. É possível concluir que o sistema é de fácil utilização, mas é possível melhorar alguns aspectos.

### O sistema é de fácil utilização?

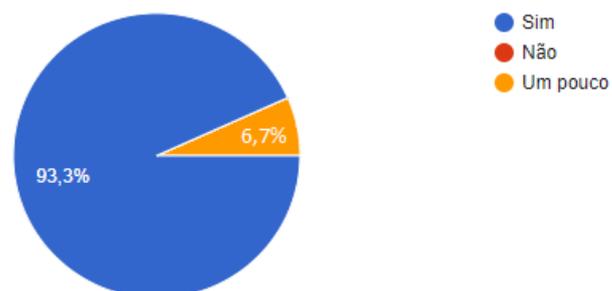


Figura 12: Questão 2 - O sistema é de fácil utilização?

Para verificar a legibilidade das informações, a terceira pergunta foi: "Você consegue compreender as informações das páginas de forma clara?". Como pode ser visto na figura 13, 100% dos alunos responderam que acreditam que as informações apresentadas pelas páginas do sistema são claras. Como não houve discordâncias dessa pergunta, conclui-se então que as informações apresentadas pelas páginas são intuitivas e claras.

Você consegue compreender as informações das páginas de forma clara?

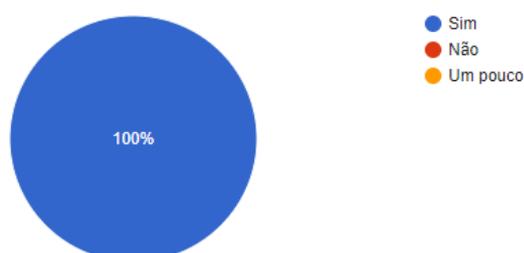


Figura 13: Questão 3 - Você consegue compreender as informações das páginas de forma clara?

Com intuito de verificar a confiabilidade que o sistema passa, a quarta pergunta foi: "O uso do sistema lhe passou segurança?". Como pode ser visto na figura 14, 100% dos alunos responderam que o sistema passa segurança. Como não houve discordâncias dessa pergunta, foi possível concluir que o sistema é confiável e consegue passar total segurança durante seu uso.

O uso do sistema lhe passou segurança?

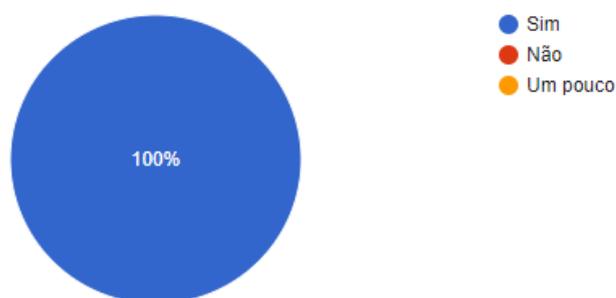


Figura 14: Questão 4 - O uso do sistema lhe passou segurança?

Para verificar a desejabilidade do sistema, a quinta pergunta foi: "Você usaria esse sistema?". Como pode ser visto na figura 15, 80% dos alunos responderam que usariam o sistema apresentado. 20% dos estudantes responderam talvez, que possivelmente usariam o sistema. É possível concluir com porcentagem satisfatória que o sistema SAGH está pronto para ser utilizados pelos discentes.

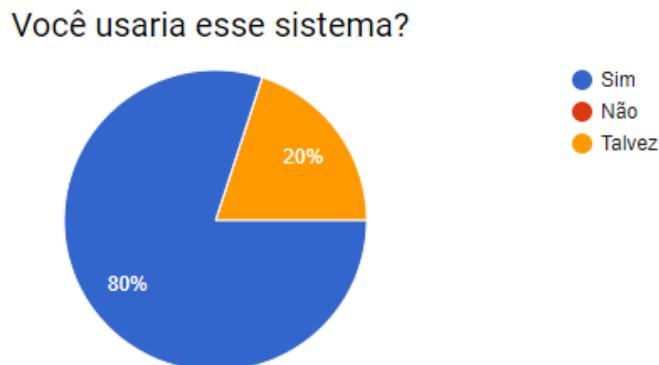


Figura 15: Questão 5 - Você usaria esse sistema?

Para verificar o tempo de resposta, a sexta pergunta foi: "O sistema apresenta demora nas respostas ao usuário?". Através das informações na figura 16, 100% dos alunos responderam que não, o sistema não apresenta demora nas respostas ao usuário. Como não houve discordâncias dessa pergunta, foi possível concluir que o sistema tem um bom tempo de respostas aos usuários e isso garante a satisfação dos mesmos.

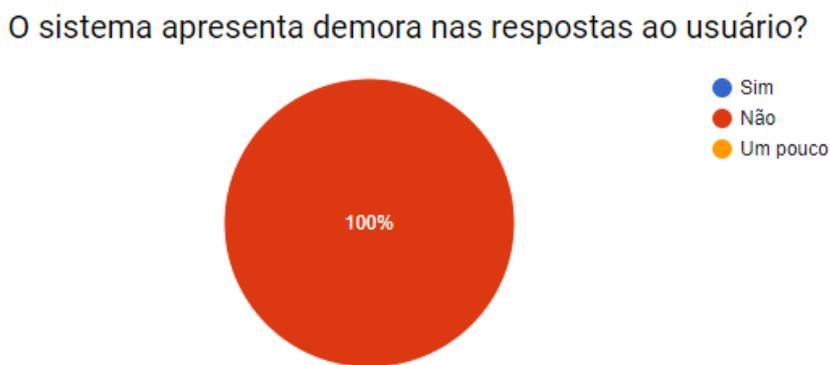


Figura 16: Questão 6 - O sistema apresenta demora nas respostas ao usuário?

Com objetivo de compreensão de como gerar a grade no sistema, a sétima pergunta foi: "Você teve dificuldade ao gerar sua grade horária?". Como pode ser visto na figura 17, 86,7% dos alunos responderam não tiveram dificuldade em gerar a grade. 13,3% dos estudantes responderam um pouco. Conclui-se então, que há pouca dificuldade em gerar a grade horaria do sistema.

## Você teve dificuldade ao gerar sua grade horária?

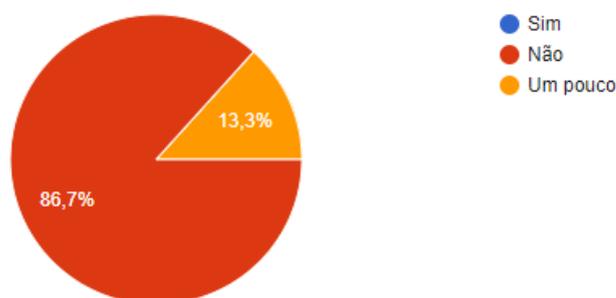


Figura 17: Questão 7 - Você teve dificuldade ao gerar sua grade horária?

Com relação a usabilidade do sistema, a oitava e última pergunta foi: "O sistema é preciso e eficaz?". De acordo com os dados dispostos na figura 18, 93,3% dos alunos responderam que sim, acreditam que o sistema é preciso e eficiente. Apenas 6,7% dos estudantes responderam um pouco. Com uma ótima margem de porcentagem e com algumas divergências, é possível concluir que o sistema é bastante eficaz.

## O sistema é preciso e eficaz?

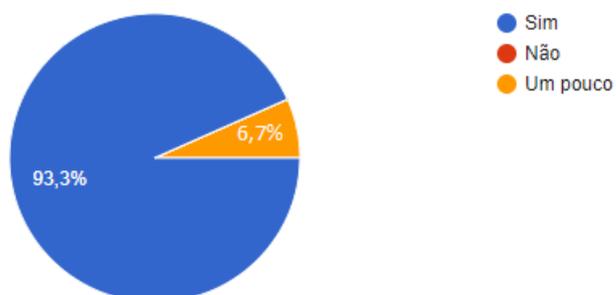


Figura 18: Questão 8 - O sistema é preciso e eficaz?

Diante do exposto, foram analisados diversos aspectos de usabilidade de *software*, permitindo compreender pontos relevantes do sistema projetado. Através dessa avaliação, foi possível identificar necessidades de ajustes que devem ser implementados em relação a intuitividade da aplicação. Através do questionário respondido pelos entrevistados que utilizaram o Sistema Automático para Gerenciamento de Horários, foi possível observar que a aplicação *web* desenvolvida possui características relevantes ao seu propósito e está pronta para ser utilizada no contexto do Curso de Sistemas de Informações do Campus Senador Helvidio Nunes de Barros.

## 4 Conclusão e Trabalhos Futuros

Com o grande avanço das ferramentas tecnológicas no meio educacional, surgiu a necessidade de um sistema que conseguisse auxiliar os alunos que possuem reprovação na tomada de decisão de quais disciplinas escolherem para sua grade de horários. Com essa necessidade, surgiu então o Sistema Automatizado para Gerenciamento de Horários (SAGH). Este sistema consegue não só oferecer a melhor opção, mas também consegue ajudar o aluno a visualizar e administrar toda a sua grade de horários.

Os resultados da avaliação comprovaram que o sistema alcançou seu principal objetivo de fornecer assistência aos discentes na escolha de sua grade horária. O objetivo principal só foi alcançado devido a fato de que o sistema foi fundamentado de forma clara e adequada através de um bom embasamento teórico, levantamento de requisitos e modelagens UML. A avaliação apontou que o sistema alcançou ótimas porcentagens em todas as características que foram analisadas.

Uma das grandes dificuldades em desenvolver este sistema, foi devido ao fato de que o atual ambiente virtual que recebe o nome de Sistema Integrado de Gestão de Atividades Acadêmicas (SIGAA) da Universidade Federal do Piauí (UFPI), não possui *API* (Interface de Programação de Aplicações) para fornecer os dados dos componentes curriculares. Todos os dados das disciplinas foram coletados manualmente. Isso dificulta muito o manuseio dos dados, pois a tarefa de atualizar os dados é preciso ser feita de forma manual.

Através da avaliação feita no capítulo anterior, foi possível concluir que o Sistema Automático para Gerenciamento de Horários está apto a conseguir prover auxílio aos alunos do curso de Sistemas de Informação da Universidade Federal do Piauí - Campus Picos na etapa de escolha dos componentes curriculares. Desenvolvido para alinhar presteza e eficiência, o sistema consegue oferecer uma interface interativa para os usuários administrarem sua grade horária com facilidade.

Para trabalhos futuros, algumas melhorias podem complementar o sistema. Tais como: melhorar a interface para aumentar a intuitividade com o usuário, expandir o sistema para alcançar mais cursos, acrescentar mais informações para ajudar os alunos a compreenderem melhor o sistema, possibilidade do aluno salvar sua grade em PDF e por fim, conseguir integrá-lo com alguma *API* (Interface de Programação de Aplicações) que o Sistema Integrado de Gestão de Atividades Acadêmicas (SIGAA) venha disponibilizar no futuro.

# Referências

- ALMEIDA, R. et al. *HTML5*. [S.l.], 2012. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Web/HTML/HTML5>>. Acesso em: 25 de Setembro de 2018. Citado na página 21.
- BATTISTELLI, J. *O que é o Bootstrap e porque você precisa dele na sua programação front-end*. [S.l.], 2018. Disponível em: <<https://blog.mastertech.tech/tecnologia/o-que-e-o-bootstrap-e-porque-voce-precisa-dele-na-programacao-front-end/>>. Acesso em: 20 de Setembro de 2018. Citado na página 18.
- DJANGO. *Meet Django*. [S.l.], 2018. Disponível em: <<https://www.djangoproject.com/>>. Acesso em: 20 de Setembro de 2018. Citado na página 17.
- FAYAD, M. E.; SCHIMIDT, D. C.; JOHNSON, R. E. *Implementing Application Frameworks: Object-Oriented Frameworks at Work*. EN: Wiley, 1999. Citado na página 16.
- GOULD, J. D.; LEWIS. Design for usability: Key principles and what designers think. *Communications of the ACM*, n. 28, p. 300–311, 1985. Citado na página 18.
- JAQUES, R. *O que é um Framework? Para que serve?* [S.l.], 2016. Disponível em: <<http://www.phpit.com.br/artigos/o-que-e-um-framework.phpit>>. Acesso em: 19 de Setembro de 2018. Citado na página 16.
- KORTH, H. F.; SILBERCHATZ, A.; SUDARSHAN, S. *Sistema de bancos de dados*. São Paulo, SP, Brasil: Makron Books, 1999. Citado 2 vezes nas páginas 15 e 16.
- LUIZ, A. *O que é JavaScript? Conheça essa linguagem de programação*. [S.l.], 2018. Disponível em: <<https://www.weblink.com.br/blog/programacao/o-que-e-javascript/>>. Acesso em: 25 de Setembro de 2018. Citado na página 23.
- NIELSEN, J. *Usability Engineering*. New York, NY, EUA: Academic Press, 2007. Citado na página 18.
- PEREIRA, A. P. *O que é o CSS*. [S.l.], 2009. Disponível em: <<https://www.tecmundo.com.br/programacao/2705-o-que-e-css-.htm>>. Acesso em: 25 de Setembro de 2018. Citado na página 22.
- POLLO, I. *Banco de dados: definição, manipulação de dados, tabelas e outras dicas*. [S.l.], 2017. Disponível em: <<http://polo-it.com.br/blog/banco-de-dados-definicao-manipulacao-de-dados-tabelas-e-outras-dicas/>>. Acesso em: 09 de Outubro de 2018. Citado na página 15.
- ROBINSON, D. *The Incredible Growth of Python*. [S.l.], 2017. Disponível em: <<https://stackoverflow.blog/2017/09/06/incredible-growth-python/>>. Acesso em: 17 de Outubro de 2018. Citado na página 20.
- ROCHA, H. V. da; BARANAUSKAS, M. C. C. *Design e Avaliação de Interfaces Humano-computador*. São Paulo, SP, Brasil: ime-usp, 2000. Citado na página 18.

SCHROEDER, R. *Html5, Um Novo Desenvolvimento Para A Web*. Dissertação (Trabalho de Conclusão de Curso) — UFSC - RS, 2012. Citado na página 21.

SZYPERSKI, C. *Component Software: Beyond Object-Oriented Programming*. Menlo Park, CA, EUA: Addison-Wesley, 1997. Citado na página 17.

VELHO, D. S. *A importância da linguagem Python para profissionais de segurança*. [S.l.], 2013. Disponível em: <<https://deividvelho.wordpress.com/2013/10/29/a-importancia-da-linguagem-python-para-profissionais-de-seguranca/>>. Acesso em: 20 de Setembro de 2018. Citado na página 20.

# Apêndices

# APÊNDICE A – Fomulário Utilizado nos Tes- tes

1. Você teve dificuldade em compreender o sistema?

Sim

Não

Um pouco

2. O sistema é de fácil utilização?

Sim

Não

Um pouco

3. Você consegue compreender as informações das paginas de forma clara?

Sim

Não

Um pouco

4. O uso do sistema lhe passou segurança?

Sim

Não

Um pouco

5. Você usaria esse sistema?

Sim

Não

Talvez

6. O sistema apresenta demora nas respostas ao usuário?

Sim

Não

Um pouco

7. Você teve dificuldade ao gerar sua grade horária?

Sim

Não

Um pouco

8. O sistema é preciso e eficaz?

Sim

Não

Um pouco



**TERMO DE AUTORIZAÇÃO PARA PUBLICAÇÃO DIGITAL NA BIBLIOTECA  
“JOSÉ ALBANO DE MACEDO”**

**Identificação do Tipo de Documento**

- ( ) Tese  
( ) Dissertação  
( X ) Monografia  
( ) Artigo

Eu, **Wildyson Dantas dos Santos**, autorizo com base na Lei Federal nº 9.610 de 19 de Fevereiro de 1998 e na Lei nº 10.973 de 02 de dezembro de 2004, a biblioteca da Universidade Federal do Piauí a divulgar, gratuitamente, sem ressarcimento de direitos autorais, o texto integral da publicação **SAGH: Sistema Automatizado para Gerenciamento de Horários** de minha autoria, em formato PDF, para fins de leitura e/ou impressão, pela internet a título de divulgação da produção científica gerada pela Universidade.

Picos-PI 04 de dezembro de 20 16.

Wildyson Dantas dos Santos  
Assinatura