

IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo

ANA PAULA MOURA MESSIAS DE SOUZA	SP3044505
GUSTAVO SANTOS COSTA SOARES	SP3044491
HENRIQUE LUIS BAESA	SP3045048
ISABELLA VALERIO MAZARÁ	SP3045463
JOSINEUDO DAS CHAGAS ARRUDA	SP3045439
PAULO KENJI YOKOTA MUNEISCHI	SP3045382

ADA - Atribuição de Aulas

São Paulo - SP - Brasil

2023

IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo

ANA PAULA MOURA MESSIAS DE SOUZA	SP3044505
GUSTAVO SANTOS COSTA SOARES	SP3044491
HENRIQUE LUIS BAESA	SP3045048
ISABELLA VALERIO MAZARÁ	SP3045463
JOSINEUDO DAS CHAGAS ARRUDA	SP3045439
PAULO KENJI YOKOTA MUNEISCHI	SP3045382

ADA - Atribuição de Aulas

Projeto apresentado no quarto ano do curso de informática integrado ao ensino médio, do Instituto Federal de Educação, Ciência e Tecnologia de São Paulo - Câmpus São Paulo, para a conclusão da disciplina de Projeto de Desenvolvimento de Sistemas.

Professor: GUSTAVO FORTUNATO PUGA

Professor: LEONARDO ANDRADE MOTTA DE LIMA

IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo

Curso Técnico em Informática Integrado ao Ensino Médio

PDS - Prática para Desenvolvimento de Sistemas

São Paulo - SP - Brasil

2023

Resumo

Esse projeto, [Sistema de Atribuição de Aulas \(ADA\)](#), como o próprio nome sugere, fornece a coordenação e a execução do processo de atribuição de aulas aos docentes, no [Instituto Federal de Educação, Ciência e Tecnologia de São Paulo \(IFSP\)](#) - Câmpus São Paulo. Como objetivo, visa automatizar o processo atual, no [Excel](#), de forma a tornar o preenchimento e a leitura dos dados do [Formulário de Preferência de Atividades \(FPA\)](#) funcional e descomplicado. Destinado a cumprir a necessidade final dos docentes, o projeto igualmente será administrado por outros funcionários, cujos cargos estarão responsáveis pela atribuição naquele ano. Para executar todos os processos, ~~no~~ será o utilizado o ~~framework open source, à linguagem Java, e no, o open source, com base na linguagem TypeScript, e detalhes implementados através do Cascading Style Sheets 3rd version (-).~~ ~~Isso usando do open source~~ Django, baseado na linguagem Python, em conjunto com o Sistema de Gerenciamento de Banco de Dados (SGBD) ~~MySQL~~ SQLite.

Palavras-chaves: Atribuição. Aulas. Automatização. Docentes. Processo. Implementação.

Abstract

This project, Assignment of Classes ([ADA](#)), as its name suggests, coordinates and executes the process of assigning classes to professors at the Federal Institute of Education, Science, and Technology of São Paulo ([IFSP](#)) - São Paulo Campus. As an objective, it aims to automate the current process, in [Excel](#), to make filling and reading the data of the Activity Preference Form ([FPA](#)) functional and uncomplicated. Destined to fulfill the final need of the professors, the project will also be managed by other employees, whose positions will be responsible for the attribution that year. To execute all the processes, ~~in the , the open-source the framework will be used in the Java language, and the , the open-source Django framework, based on the language, and details implemented through the Cascading Style Sheets 3rd version (). This is using the MySQL Database Management System ().~~ Python language, will be used together with SQLite Database Management System (DBMS).

Keywords: Assignment. Classes. Automation. Teachers. Process. Implementation.

Lista de ilustrações

Figura 1 – Fluxograma dos pré-processos de atribuição	13
Figura 2 – Caso de uso do envio de preferências	14
Figura 3 – Fluxograma da atribuição de aulas	14
Figura 4 – Caso de uso troca de Aula	15
Figura 5 – <u>Certificado SSL/TLS</u>	19
Figura 6 – <u>Diagrama de arquitetura</u>	20

Lista de abreviaturas e siglas

ADA	Sistema de Atribuição de Aulas - Citado em 2 , 3 , 9 , 11 , 12 , 14 , 16 , 23
AJAX	Asynchronous JavaScript and XML - JavaScript e XML Assíncronos - Citado em 17
Apeoesp	Sindicato dos Professores do Ensino Oficial do Estado de São Paulo - Citado em 11
API	Application Programming Interface - Interface de Programação de Aplicação - Citado em 17
AWS	Amazon Web Services - Serviços Web da Amazon - Citado em 18
CPF	Cadastro de Pessoas Físicas - Citado em 11
CSS	Folha de Estilos em Cascata - Citado em 18
DBMS	Database Management System - Citado em 3
DIT	Diretoria de Informática e Turismo - Citado em 9 , 23
DRY	Don't Repeat Yourself - Não Se Repita - Citado em 16
Etecs	Escolas Técnicas Estaduais - Citado em 10
FPA	Formulário de Preferência de Atividades - Citado em 2 , 3 , 9 , 10 , 12 , 15
HTML	HyperText Markup Language - Linguagem de Marcação de Hipertexto - Citado em 17 , 18
HTTPS	Hyper Text Transfer Protocol Secure - Protocolo Seguro de Transferência de Hipertexto - Citado em 19
IDE	Ambiente de Desenvolvimento Integrado - Citado em 16
IFSP	Instituto Federal de Educação, Ciência e Tecnologia de São Paulo - Citado em 2 , 3 , 9 , 16 , 17 , 20 , 23
JS	JavaScript - Citado em 17
JSON	JavaScript Object Notation - Notação de Objetos JavaScript - Citado em 17
MTV	Model-View-Template - Citado em 7 , 16 , 18
MVC	Model-View-Controller - Citado em 16
OO	Orientação a Objetos - Citado em 17
SED	Secretaria da Educação do Estado de São Paulo - Citado em 10 , 11
SGBD	Sistema de Gerenciamento de Banco de Dados - Citado em 2 , 18
SIG	Sistema Integrado de Gestão - Citado em 10
SPA	Aplicações de Página Única - Citado em 9

SQL	Structured Query Language - Linguagem de Consulta Estruturada - Citado em 18
SSL	Secure Sockets Layer - Camada de Soquete Seguro - Citado em 19
SUAP	Sistema Unificado de Administração Pública - Citado em 16
SVN	SubVersion - Subversão - Citado em 19
TLS	Transport Layer Security - Segurança de Camada de Transporte - Citado em 19
URH	Unidade de Recursos Humanos - Citado em 10
VSCode	Visual Studio Code - Citado em 20

Sumário

1	INTRODUÇÃO	9
1.1	Objetivo	9
1.2	Análise de concorrentes	10
2	IDEIA	12
2.1	Cadastramento dos usuários	12
2.1.1	Configuração do ambiente	12
2.2	Automatização do FPA	13
2.3	Permutação	15
2.4	Tecnologias e ferramentas aplicadas	16
2.4.1	<u>Tecnologias</u>	16
2.4.2	Desenvolvimento do sistema	16
2.4.1.1	<u>Django</u>	16
2.4.1.1.1	Model-View-Template (MTV)	16
2.4.1.2	<u>Python</u>	17
2.4.1.3	<u>AJAX</u>	17
2.4.1.4	<u>JavaScript</u>	17
2.4.1.5	<u>HTML</u>	17
2.4.1.6	<u>CSS</u>	18
2.4.1.7	<u>Bootstrap</u>	18
2.4.1.8	<u>SQLite3</u>	18
2.4.2	<u>Hospedagem</u>	18
2.4.3	<u>Criptografia</u>	19
2.4.4	Controle de Versão e Implementação <u>Ferramentas</u>	19
2.4.4.1	<u>Controle de Versão</u>	19
2.4.4.2	<u>Documentação</u>	19
2.4.4.3	<u>Programação</u>	20
2.4.5	<u>Diagrama de Arquitetura</u>	20
3	LINKS DO PROJETO	21
3.1	Fluxograma	21
3.2	Repositório GitHub	21
3.3	Repositório Subversion	21
3.4	Casos de Uso	22
3.5	Protótipo de Baixa Fidelidade	22

4	CONSIDERAÇÕES FINAIS	23
	Referências	24
	GLOSSÁRIO	25

1 Introdução

No IFSP Câmpus São Paulo, são mais de 350 docentes, com 56 na área de [Diretoria de Informática e Turismo \(DIT\)](#). Logo, é essa a quantidade que participa da atribuição de aulas semestral e/ou anual, indicando seu extenso e intrincado processo. Além da referida quantidade, o que dá essas características à atribuição são os critérios a serem seguidos, desde a ordem de prioridade dos docentes até as preferências colocadas por cada um no [FPA](#) e os regramentos presentes na resolução vigente – em 2023, ainda rege a Resolução n°109/2015, de 4 de novembro de 2015.

Todo esse processo é realizado manualmente, através da entrega do [FPA](#), da ferramenta [Excel](#) e da comunicação particular constante entre o administrador daquele ano e o docente, principalmente no caso de [permutas](#). Assim, procedem adversidades, conflitos interpessoais e atrasos, relatados semestralmente pelos docentes, principalmente pelos que ficam no final da fila de prioridade (os substitutos), e comentados, após contato da equipe, pelo administrador das atribuições da [DIT](#) atual, Evandro ..., e antigo, Leonardo Motta, os quais enfatizaram sobre a consequente sobrecarga em suas funções, ao tentar equilibrar a vontade de todos e, ao mesmo tempo, cumprir com a lei.

Em decorrência disso, surge a necessidade da automatização de parte dos processos, que tem como objetivo a aprimoração do andamento do fluxo de trabalho ([SYDLE, 2023](#)), trazendo como resultado o aumento da produtividade e a redução de custos e de erros ([TOTVS, 2022](#)).

E, à vista do que foi exposto, o projeto retratado propõe a elaboração de um sistema que automatize os principais processos da atribuição - as seleções do [FPA](#) e a [permutação](#) - e, simultaneamente, cumpra o exigido na Resolução, e nos outros critérios estabelecidos hoje (como a prioridade da escolha do docente na atribuição) e que podem ser posteriormente adicionados. Esse sistema é o [ADA](#).

1.1 Objetivo

O [ADA](#) visa apresentar uma solução e uma aprimoração às problemáticas da atribuição de aulas. Logo, oferecer um sistema Web responsivo de Single Application Page ([Aplicações de Página Única \(SPA\)](#)) aos funcionários, que automatize essa atribuição e a respectiva e consecutiva [permutação](#) (caso habilitada), sem a necessidade de organização manual e de negociações individuais e extraoficiais.

Com processos correspondentes aos problemas centrais, terá a implementação do [login](#), pelo e-mail oficial do Instituto no Google, tratando do gerenciamento geral dos

docentes e dos administradores; da automatização do [FPA](#), tratando da dificuldade de estruturação da grade horária seguindo todos os critérios e os regramentos; e da automatização das negociações à [permuta](#), tratando dos atritos e da dificuldade de comunicação gerados.

Consequentemente, o sistema proporcionará, a princípio, um ambiente em que o administrador superior e os subadministradores consigam controlar e ordenar os critérios às suas área e subáreas, respectivamente, e habilitar funções como a [permutação](#) e a desativação de um em determinada matéria. Ademais, proporcionará um ambiente em que o docente consiga selecionar todas as suas preferências e solicitar suas [permutas](#) (caso habilitadas) em um único local, sem demasiadas complicações e processos.

Uma operação antes com responsabilidades individuais e organização manual, a qual incorre de mais erros devido a subjetividade e os problemas humanos, passará a ser uma operação tecnológica mais limpa e funcional, com menos erros.

1.2 Análise de concorrentes

A Análise de Concorrência é valiosa para obter conhecimento sobre como outros sistemas – com o mesmo propósito ou um próximo – desenvolvem seu projeto, implementam seus processos, atraem clientes, apresentam sua plataforma, gerenciam seus dados, fecham parcerias, entre outros; fatores importantes a serem considerados tanto para o aprimoramento do sistema que você está realizando, quanto para saber com quem está disputando o mercado.

Nessa pesquisa, a equipe achou algumas concorrências referentes ao processo de atribuição de aulas. Entre elas, vale a pena serem citadas a [Secretaria da Educação do Estado de São Paulo \(SED\)](#) e o [Sistema Integrado de Gestão \(SIG\)](#). A SED apresenta um sistema que abrange toda a rede educacional estadual de São Paulo (SP) ([EDUCAÇÃO DO ESTADO DE SÃO PAULO, s.d.](#)). Através dele, o docente pode manifestar seu interesse em aulas vagas e livres ou em substituição das escolas, pela pesquisa por uma escola específica ou uma disciplina. Quanto à prioridade de escolha, o docente pode alterar a ordem da sua seleção de acordo com as escolas em que prefere lecionar.

E o [SIG \(SISTEMA..., s.d.\)](#) aparenta apresentar um sistema parecido, onde o docente preenche e envia a inscrição e o requerimento de ampliação de carga horária de forma digital. Entretanto, as informações são escassas, baseadas na página de divulgação da [Unidade de Recursos Humanos \(URH\) \(UNIDADE..., s.d.\)](#), onde está o sistema; o tutorial leva a uma página de erro. Contudo, uma diferença fundamental é o fato de ser voltado apenas às [Escolas Técnicas Estaduais \(Etecs\)](#).

Com o conhecimento adquirido foi possível observar pontos a serem implementados

no [ADA](#), como a disponibilidade de um tutorial no início e mensagens de ajuda ao longo da página – além do ótimo design da [SED](#) que pode servir de inspiração. E pontos semelhantes foram justamente essa escolha de aulas pelo docente, de acordo com sua subárea, e o *login* que não permite o cadastro de pessoas não autorizadas (um usa do [Cadastro de Pessoas Físicas \(CPF\)](#) e o outro do e-mail).

Todavia, foi igualmente possível observar a necessidade do sistema apresentado neste projeto, que prevê erros retratados nos concorrentes, como o atraso considerável do primeiro devido às longas filas de seleção, a ausência de verificação de componente curricular, abrindo uma brecha para qualquer docente lecionar a disciplina sem ter o nível de escolaridade necessária, e os erros no processo de pontuação para prioridade de escolha – retratados em uma matéria ([PROFESSORES... , 2020](#)) da [Globo](#) e relatados pela Inês Paz, coordenadora da subsede da [Sindicato dos Professores do Ensino Oficial do Estado de São Paulo \(Apeoesp\)](#) e vereadora de Mogi, ([ACIOLI, 2021](#)): “A classificação saiu com muitos erros e os professores não estão tendo um retorno às suas perguntas, enquanto isso, a atribuição continua acontecendo com esses profissionais correndo o risco de se prejudicarem”. Já do segundo concorrente, a falta de tutorial e dificuldade na compreensão da página de *login*.

À vista do citado acima, o sistema [ADA](#) subsidia uma série de ações que permitem o cruzamento de dados e processos que diferenciam a ideia de qualquer outra anterior; e, por ser voltado ao Instituto em específico, igualmente permite uma melhor análise dos [feedbacks](#) e tratamento dos erros. Portanto, será implementado de forma a considerar boas práticas de concorrentes e aplicações parecidas, e, principalmente, dificuldades nelas encontradas, não atendo-se a um ciclo de falhas.

2 Ideia

A fim de implementar um sistema que trate dos problemas citados e consiga atingir os objetivos propostos, são necessários processos e a utilização de determinadas tecnologias, citadas no subtópico .

Os processos principais são três, o cadastramento dos usuários, a automatização do [FPA](#) e a possibilidade de habilitação de outros processos, por parte dos administradores, com destaque à permutação dos horários já atribuídos e à desativação de um docente em determinada matéria.

2.1 Cadastramento dos usuários

Preliminar à qualquer utilização do [ADA](#), o Administrador Superior ([SuperAdmin](#)), será cadastrado pelos próprios programadores e terá o maior nível de acesso, podendo realizar quaisquer alterações e controlar quais serão os Administradores ([Admin](#)).

Então, os outros funcionários receberão um link para acessarem o [ADA](#) via Google, pelo e-mail institucional - o que evita acessos não permitidos, e serão atribuídos instantaneamente ao papel de Professor ([Professor](#)); como mencionado, a mudança desse nível de acesso para o de [Admin](#) é realizada pelo [SuperAdmin](#). E acessos posteriores poderão ser através do Google ou do prontuário e senha.

2.1.1 Configuração do ambiente

A configuração do ambiente é um subprocesso, em que o [SuperAdmin](#) será responsável por habilitar a possibilidade de [permutas](#) e de desativação do docente em uma disciplina; prazos limites à organização; e definição ou atualização dos critérios da atribuição - baseados na legislação vigente e na ordem de prioridade de escolha das disciplinas.

E o [Admin](#) será responsável pela subárea, conseqüentemente, por subir a grade horária; determinar prazos específicos; autorizar a [permutação](#) e se deseja participar da aprovação das [permutas](#); controlar os docentes desativados; e adicionar¹ os que participarão de sua subárea.

¹ Essa adição será manual e de acordo com a prioridade escolhida. Portanto, um subprocesso, onde o [Admin](#) colocará os docentes na ordem e, igualmente, poderá alterá-la em caso de erro ou modificações futuras.

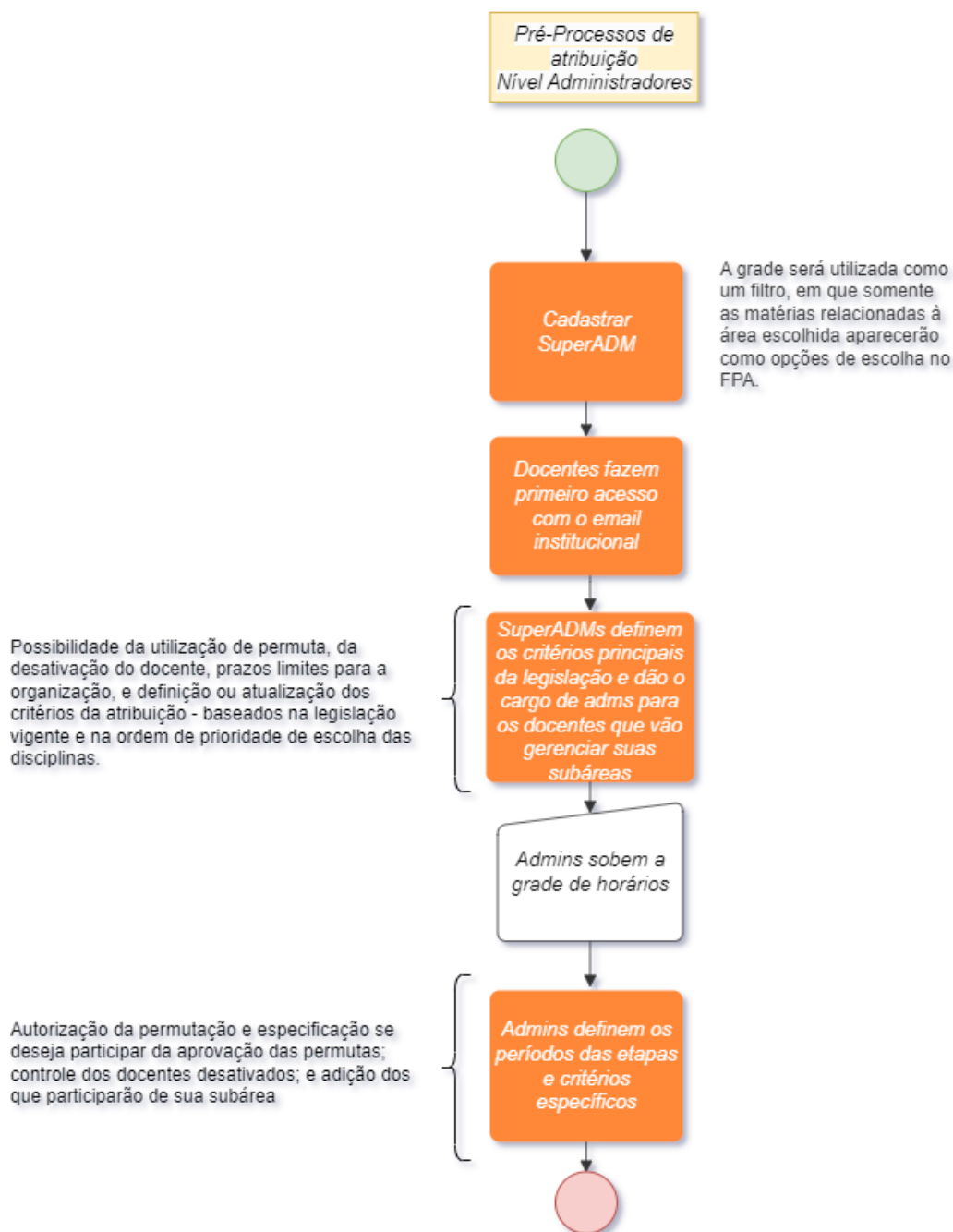


Figura 1 – Fluxograma dos pré-processos de atribuição

2.2 Automatização do FPA

Finalizada a organização do sistema pelos administradores e todos os docentes cadastrados nas subáreas, eles poderão acessar o sistema e iniciar o processo de escolha da disponibilidade de horários e da preferência de aulas (prioritária e secundária) e de atividades. Conforme é realizado esse processo, o ADA verifica se cada escolha segue os regramentos, e impossibilita a escolha de disciplinas em conflito; igualmente, informa com uma mensagem breve caso o docente selecione uma em que foi desativado.

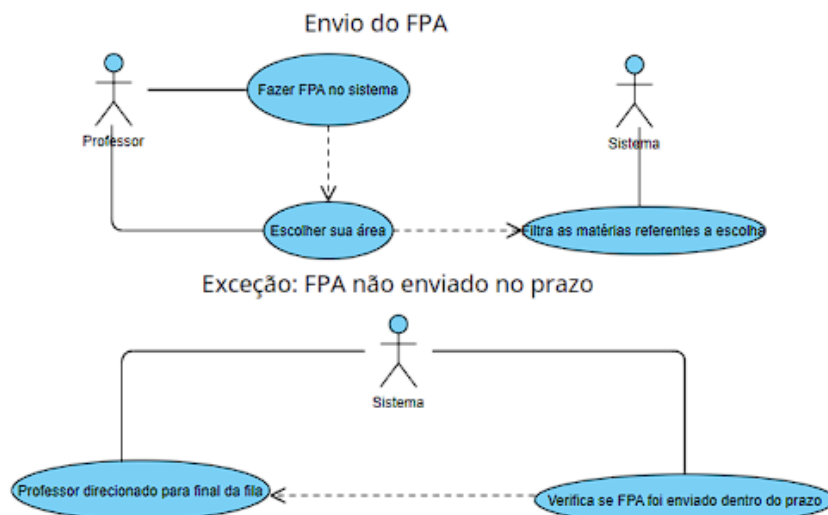


Figura 2 – Caso de uso do envio de preferências

A determinação da preferência de atividades poderá ser modificada dentro do prazo de entrega estabelecido pelo **Admin**. Porém, ao encerrar o prazo, o **ADA** percorre a lista de docentes, em ordem decrescente, e atribui as aulas de acordo com o selecionado. O processo é interrompido - e é armazenado o que já foi feito - caso haja conflito com uma disciplina já escolhida; assim, aquele docente receberá uma solicitação para alterar sua escolha dentro de determinado prazo.

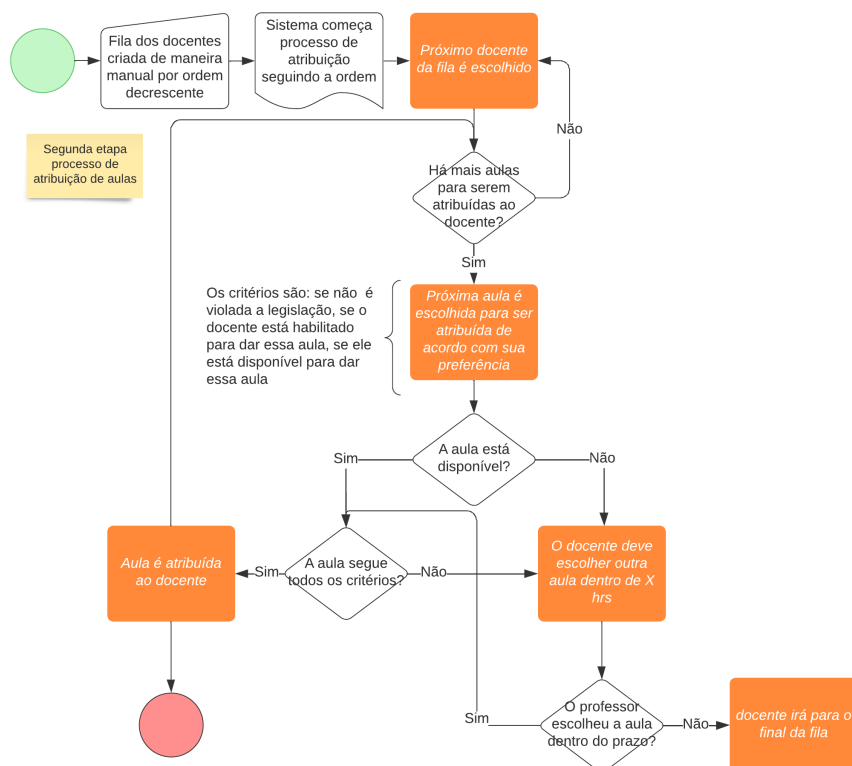


Figura 3 – Fluxograma da atribuição de aulas

2.3 Permutação

A permutação é aberta, caso habilitada com a conclusão da grade pelo sistema. De modo geral, é feita com a solicitação de um docente pela troca de sua aula por uma específica do outro, selecionada na grade. É impossibilitada mais de uma solicitação, ao mesmo tempo, para uma mesma aula; Apenas é liberada quando essa for aceita ou recusada. Igualmente é impossibilitada a solicitação de alguma que descumpra o regramento. Caso o [Admin](#) seja moderador, ele terá que aprovar a aceitação da permuta pelo segundo docente.

Por fim, é gerada a grade horária final, onde os docentes e os administradores conseguem visualizar e salvar a atribuição de aulas da subárea. Além da possibilidade de gerar o [FPA](#) com essa grade pronta.

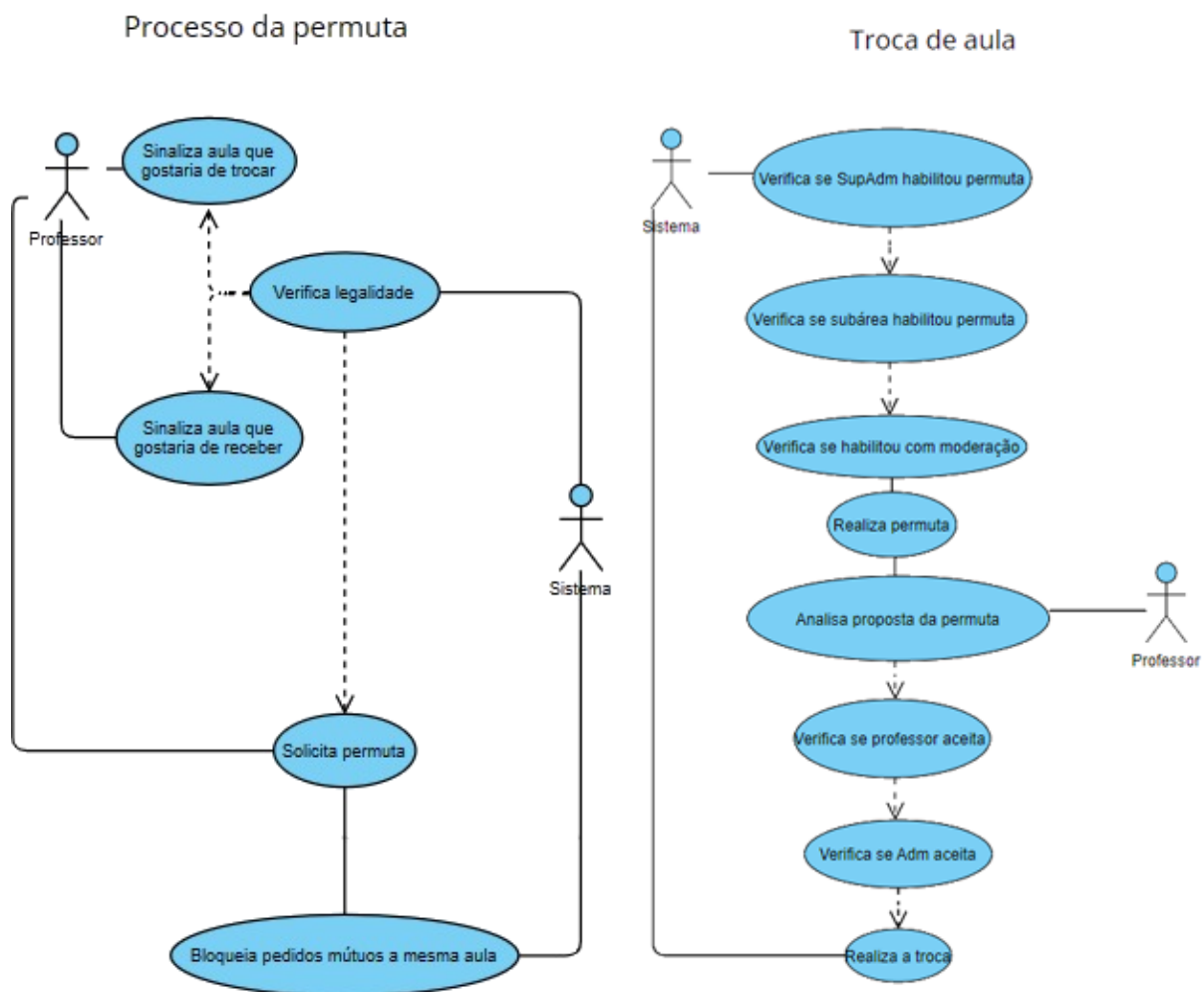


Figura 4 – Caso de uso troca de Aula

2.4 Tecnologias e ferramentas aplicadas

Em vista do desenvolvimento do [ADA](#) de maneira concisa e eficaz, a implementação de tecnologias e suas respectivas ferramentas se faz necessária. Além disso, repositórios de controle de versão e Integrated Development Environment ([Ambiente de Desenvolvimento Integrado \(IDE\)](#)) deverão, e serão, ~~utilizadas~~utilizados.

2.4.1 Tecnologias

A seguir estão as tecnologias utilizadas, suas características principais e, assim, porque foram escolhidas. A finalidade principal desse conjunto é escrever a aplicação de forma rápida e eficiente, concentrando toda a energia no desenvolvimento e aplicação da lógica, e, logo, poupando tempo em funcionalidades básicas.

2.4.2 ~~Desenvolvimento do sistema~~

2.4.1.1 Django

~~É baseado nas metodologias ensinadas no projeto, para manter uma padronização e pela maior afinidade dos integrantes com as linguagens eos. No, será utilizado o, um open source da linguagem Java. E~~framework web open source e de alto nível, desenvolvido em Python, que se baseia no padrão MTV, apresentando semelhança com o Model-View-Controller (MVC). Assim, segue o princípio Don't Repeat Yourself - Não Se Repita (DRY)², no, o open source, com base na linguagem, é moderadamente opinativo³ e apresenta suporte para erros comuns de segurança. Além desses benefícios, foi escolhido devido a sua aplicação em grandes empresas (como Mozilla e Pinterest) e, principalmente, no Sistema Unificado de Administração Pública (SUAP) do IFSP, o que permite manter o padrão de tecnologias no Instituto. (LUCAS, 2021) (ANDRADE, 2019)

2.4.1.1.1 MTV

Derivação da arquitetura de software MVC, de três camadas, altera a nomenclatura e a relação entre os arquivos. O Model permanece o mesmo, como um canal de conexão entre os tipos de dados e como serão armazenados no Banco de Dados, e a exibição ao ter requisição à View. Essa é responsável, então, pelo gerenciamento das requisições e a lógica de negócio, com a formatação dos dados enviados pelo Model. Por fim, o

² Permite que as aplicações sejam desenvolvidas com a maior quantidade de aproveitamento de código possível.

³ Flexibilidade que o framework dá aos desenvolvedores à resolução dos problemas. Opinativo, já possui uma maneira correta de resolvê-los, sem margens; não-opinativo, não possui essas regras e deixa livre para resolvê-los como quiser. Django equilíbrio entre soluções prontas e arquitetura desacoplada com liberdade na resolução de erros.

Template é a interação com o usuário, através de uma exibição estática ou inserção de sintaxe de conteúdo dinâmico, com a renderização dos dados entregues pela View. (SILVA, D. A. e., 2020)

2.4.1.2 Python

É uma linguagem de programação *open source* e de alto nível, interpretada em scripts e orientada a objetos, que apresenta tipagem dinâmica⁴ forte. Logo, prioriza a agilidade por meio de sua fácil compreensão, sintaxe menor e simplificada, sem muitas exigências gramaticais. É por isso que foi escolhida, uma ótima opção que supriu de forma excelente a necessidade de uma aprendizagem rápida e fácil codificação nos dispositivos do IFSP, além de poder ser facilmente integrada a outras linguagens de programação populares, caso seja necessário no decorrer do projeto. (AMAZON, 2023) (MELO, 2021)

2.4.1.3 AJAX

O Asynchronous JavaScript and XML - JavaScript e XML Assíncronos (AJAX) é uma técnica de desenvolvimento *web*, caracterizada pela criação de aplicações interativas através de requisições ao servidor. Uma junção das funcionalidades do JavaScript (JS) com a troca dos dados, armazenados e transmitidos, nesse caso, pelo JavaScript Object Notation - Notação de Objetos JavaScript (JSON) (mais próximo do JS). Foi escolhido justamente por servir como um canal de comunicação independente entre o cliente e o servidor. (ANDREI, 2019) (CARVALHO, 2007)

2.4.1.4 JavaScript

É uma linguagem de programação de alto nível e interpretada em scripts, com recursos de Orientação a Objetos (OO) e Application Programming Interface - Interface de Programação de Aplicação (API), que apresenta tipagem dinâmica. Assim, por meio de um funcionamento assíncrono⁵, usa trechos dos códigos HTML para renderizar funções que proporcionem uma interação dinâmica local com o conteúdo da página. Foi escolhida para, em conjunto com o AJAX, proporcionar essa dinamicidade em tempo real, recarregamento automático. (MOZILLA, 2023b) (MELO, 2021)

2.4.1.5 HTML

O HyperText Markup Language - Linguagem de Marcação de HiperTexto (HTML) é uma estrutura responsável pela exibição dos dados no navegador *web*, caracterizado

⁴ Tipo do dado é determinado no tempo de execução, de acordo com o valor do dado, não a partir da sua variável.

⁵ A programação assíncrona é uma técnica na qual o programa inicia uma tarefa e ainda é capaz de executar simultaneamente outros eventos, ao invés de bloquear processos para esperar o término da execução.

por seus elementos hierarquizados e sua marcação que abriga elementos como tags. Na aplicação ADA, é utilizada nos templates, explicados no parágrafo 2.4.1.1.1. (MOZILLA, 2023a)

2.4.1.6 CSS

O Folha de Estilos em Cascata (CSS) é uma linguagem de marcação, responsável pela estilização de elementos HTML. Foi escolhido a fim de ajudar na formatação dos templates em detalhes específicos em ~~Para o gerenciamento de dados, o Sistema de Gerenciamento de Banco de Dados (-), pelo~~ que, por vezes, não são compreendidos pelo framework, pois esse é mais genérico. (TOTVS, 2020)

~~Ademais,~~

2.4.1.7 Bootstrap

É um framework *front-end*, logo, voltado à ~~vista de promover uma~~ estilização, e *open source*. Foi escolhido devido à agilidade no desenvolvimento da página para o usuário, característica pelos frameworks, e, principalmente, devido à responsividade proporcionada. (ANDREI, 2019) (LIMA, 2021)

2.4.1.8 SQLite3

SGBD, o SQLite é uma biblioteca em linguagem C, *open source*, acoplada ao banco de dados Structured Query Language - Linguagem de Consulta Estruturada (SQL). Escolhido porque entrega o banco em conjunto com a aplicação (é embutido), sem a necessidade de um servidor, já que optamos por realizar a arquitetura MTV em vez de Cliente/Servidor. (SILVA, R. F., 2007) (CARLOS, 2019)

2.4.2 Hospedagem

A fim de ter uma instância em nuvem e conseguir fazer a hospedagem do site, foi utilizada a Amazon Web Services - Serviços Web da Amazon (AWS). É uma plataforma que disponibiliza diversos serviços de computação em uma rede de servidores remotos. Assim, é possível criar instâncias de máquinas com sistema operacional Windows ou Linux, de modo que a aplicação ~~segura~~ funcione constantemente, sem necessitar que um computador pessoal fique ligado.

Somente com ela já é disponibilizada a aplicação na *web*. Todavia, o acesso é difícil, pois aparecerá somente o endereço IPv4 público da máquina virtual criada. Para resolvê-lo, foi comprado o domínio <https://mottarios.cloud/> no *website* Hostinger.

2.4.3 Criptografia

A criptografia, para fornecer uma aplicação segura, foi configurada seguindo o protocolo Hyper Text Transfer Protocol Secure - Protocolo Seguro de Transferência de Hipertexto (HTTPS), ~~o sistema seguirá uma sequência de métodos que aplicam o formato~~, ~~utilizará o~~, ~~um padrão da Internet que visa a autenticação dos usuários no sistema.~~

De forma a verificar seu devido funcionamento, foi utilizada a certificadora *Let's Encrypt*. Assim, teve uma confirmação de que há controle sobre o domínio citado anteriormente, e, com isso, foi gerado um certificado Secure Sockets Layer - Camada de Soquete Seguro (SSL)/Transport Layer Security - Segurança de Camada de Transporte (TLS) para ele. Após adicionar o certificado, a aplicação atingiu nota A no *SSL Labs*. Para mais informações: <https://www.ssllabs.com/ssltest/analyze.html?d=mottarios.cloud>



Figura 5 – Certificado SSL/TLS

2.4.4 ~~Controle de Versão e Implementação~~Ferramentas

~~Terá a utilização do Subversion/Subversão (SVN) como repositório principal, e do~~

2.4.4.1 Controle de Versão

É utilizado o SubVersion - Subversão (SVN), uma ferramenta que armazena projetos e todas suas versões em um servidor centralizado. Escolhido devido à padronização dos projetos e devido à capacidade de armazená-los de forma segura. Igualmente, é utilizado o GitHub como repositório secundário, devido ao melhor gerenciamento de branches e o armazenamento em outro local de informações, em caso de algum erro no . Ele será atualizado quinzenalmente com o conteúdo do principal. E o código será implementado em, de forma secundária, devido à familiaridade e à melhor organização, principalmente tratando-se do uso de *branches*.

2.4.4.2 Documentação

É utilizado o Overleaf, um editor e compilador *online* de LaTeX. A fim de seguir a padronização dos projetos anteriores e para uma produção mais dinâmica dos documentos LaTeX, já que permite o compartilhamento dos arquivos entre outras pessoas e a edição simultânea.

2.4.4.3 Programação

É utilizado o Visual Studio Code (VSCode), um editor de código-fonte ~~desenvolvido pela~~ criado pela Microsoft. Foi escolhido devido a sua dinamicidade para codificação e a familiaridade que os membros da equipe já possuem com a ferramenta.

Ademais, também é utilizado o codespace, que, por sua vez, é um ambiente de desenvolvimento hospedado em nuvem, promovido pela plataforma GitHub. Assim, é possível programar através do navegador, sem precisar de uma aplicação instalada na máquina, facilitando a programação no IFSP.

2.4.5 Diagrama de Arquitetura

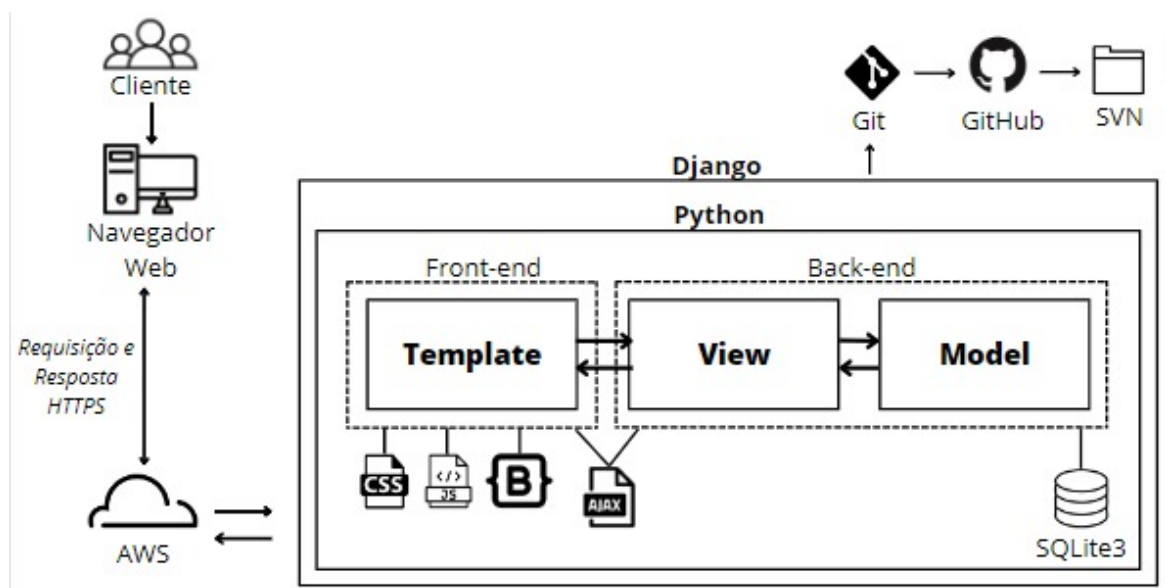


Figura 6 – Diagrama de arquitetura

\item command.

3 Links do projeto

3.1 Fluxograma



<Fluxograma>

3.2 Repositório GitHub



<<https://github.com/mottariosifsp>>

3.3 Repositório Subversion



<Repositório Subversion (SVN)>

3.4 Casos de Uso



<Casos de Uso>

3.5 Protótipo de Baixa Fidelidade



<Protótipo de Baixa Fidelidade>

4 Considerações Finais

Por meio das questões e relatos expostos ao longo da pesquisa sobre o tema do projeto, ficou cada vez mais evidenciado o grande acréscimo e consequente aprimoramento que o ADA trará à DIT, do IFSP Câmpus São Paulo; a aplicação é factível, promissora e necessária.

Quanto às dificuldades na atribuição de aulas, são comuns e recorrentes, e o sistema é projetado visando facilitar essa operação. Porém, é um projeto que pode ser expandido tanto para outras áreas quanto para outros Câmpus ou, inclusive, outras Instituições de Ensino, o que significa um grande potencial atual e, também, futuro do sistema.

Isso graças ao desenvolvimento melhor estruturado, que abriga todas as nuances dos processos e seus respectivos planos de ação, proporcionado pelas reuniões da equipe realizadas com os orientadores e demais docentes, e pela transferência das ideias para fluxogramas, protótipos e casos de uso.

Referências

- ACIOLI, Mariana. **Problemas no sistema de Atribuição de aulas prejudicam professores da rede estadual em mogi**. [S.l.]: O DIÁRIO, dez. 2021. Disponível em: <https://odiariodemogi.net.br/cidades/problemas-no-sistema-de-atribuicao-de-aulas-prejudicam-professores-da-rede-estadual-em-mogi-1.30131>. Citado 1 vez na página 11.
- AMAZON. **O que é Python?** [S.l.: s.n.], Outono 2023. Disponível em: <https://aws.amazon.com/pt/what-is/python/>. Citado 1 vez na página 17.
- ANDRADE, Ana Paula de. **O que é Django?** pt. [S.l.: s.n.], abr. 2019. Disponível em: <https://www.treinaweb.com.br/blog/o-que-e-django>. Citado 1 vez na página 16.
- ANDREI, L. **O Que é AJAX e Como Funciona?** pt. [S.l.]: Hostinger, jan. 2019. Disponível em: <https://www.hostinger.com.br/tutoriais/o-que-e-ajax>. Citado 2 vezes nas páginas 17, 18.
- CARLOS, E. **SQLite vs MySQL – Qual a Diferença e Qual Usar**. pt. [S.l.]: Hostinger, jul. 2019. Disponível em: <https://www.hostinger.com.br/tutoriais/sqlite-vs-mysql>. Citado 1 vez na página 18.
- CARVALHO, Rogério Moraes. **O que é o AJAX? Estudaremos o Asynchronous JavaScript and XML**. pt. [S.l.: s.n.], out 2007. Disponível em: <https://www.devmedia.com.br/o-que-e-o-ajax/6702>. Citado 1 vez na página 17.
- EDUCAÇÃO DO ESTADO DE SÃO PAULO, Secretaria da. **Secretaria da Educação do Estado de São Paulo**. [S.l.: s.n.]. Disponível em: <https://sed.educacao.sp.gov.br/>. Citado 1 vez na página 10.
- LIMA, Guilherme. **Bootstrap: o que é, como usar, documentação e exemplos**. pt. [S.l.: s.n.], jul. 2021. Disponível em: <https://www.alura.com.br/artigos/bootstrap>. Citado 1 vez na página 18.
- LUCAS, Eduardo. **Django: tudo que você precisa saber sobre o framework de Python**. pt. [S.l.: s.n.], jun. 2021. Disponível em: <https://blog.geekhunter.com.br/django-introducao-ao-framework/>. Citado 1 vez na página 16.
- MELO, Diego. **O que é Python? [Guia para iniciantes]**. pt. [S.l.]: Mobilon Mídia LTDA, jan. 2021. Disponível em: <https://tecnoblog.net/responde/o-que-e-python-guia-para-iniciantes/>. Citado 2 vezes na página 17.
- MOZILLA. **HTML: Linguagem de Marcação de Hipertexto**. pt. [S.l.: s.n.], abr. 2023a. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/HTML>. Citado 1 vez na página 18.

- MOZILLA. **O que é JavaScript?** pt. [S.l.: s.n.], out 2023b. Disponível em: https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First_steps/What_is_JavaScript. Citado 1 vez na página 17.
- NASCIMENTO, Anderson. **O que é open source?** [S.l.: s.n.], ago 2014. Disponível em: <https://canaltech.com.br/produtos/0-que-e-open-source/>. Citado 1 vez na página 26.
- PROFESSORES da Rede Estadual de SP Relatam Problemas E tumulto no processo de Atribuição de Aulas. [S.l.: s.n.], jan. 2020. Disponível em: <https://g1.globo.com/sp/sao-paulo/educacao/noticia/2020/01/24/professores-da-rede-estadual-de-sp-relatam-problemas-e-tumulto-no-processo-de-atribuicao-de-aulas.gh.html>. Citado 1 vez na página 11.
- SILVA, Diandra Andrade e. **Como funciona a arquitetura MTV (Django)**. en. [S.l.: s.n.], jan. 2020. Disponível em: <https://diandrasilva.medium.com/como-funciona-a-arquitetura-mtv-django-86af916f1f63>. Citado 1 vez na página 17.
- SILVA, Rafael Felix. **SQLite - O Pequeno Notável**. pt. [S.l.: s.n.], nov. 2007. Disponível em: <https://www.devmedia.com.br/sqlite-o-pequeno-notavel/7249>. Citado 1 vez na página 18.
- SISTEMA Integrado de Gestão - URH. [S.l.: s.n.]. Disponível em: <https://sigurh.cps.sp.gov.br/>. Citado 1 vez na página 10.
- SYDLE. **Automatização de Processos: Como Funciona? Quais os benefícios?** [S.l.]: Blog SYDLE, mar. 2023. Disponível em: <https://www.sydle.com/br/blog/automatizacao-de-processos-5ef257889d8f5430788fcd45/>. Citado 1 vez na página 9.
- TOTVS, Equipe. **Automação de Processos: Tipos, exemplos e como fazer**. [S.l.: s.n.], dez. 2022. Disponível em: <https://www.totvs.com/blog/gestao-para-assinatura-de-documentos/automacao-de-processos/>. Citado 1 vez na página 9.
- TOTVS, Equipe. **O que é CSS? Conheça benefícios e como funciona**. pt. [S.l.: s.n.], dez 2020. Disponível em: <https://www.totvs.com/blog/developers/o-que-e-css/>. Citado 1 vez na página 18.
- UNIDADE de Recursos Humanos. [S.l.: s.n.]. Disponível em: <https://urh.cps.sp.gov.br/wp-content/uploads/sites/10/2021/10/Inscricao-de-atribuicao-de-aulas.pdf>. Citado 1 vez na página 10.

Glossário

Admin	Administradores, enum que representa os funcionários que receberão essa funcionalidade naquele semestre/ano. - Citado em 12 , 14 , 15
Django	Framework web full stack open source baseado em Python, gratuito e de alto nível, focado em desenvolvimento rápido para web. - Citado em 2 , 3
enum	Tipo de classe especial que permite o armazenamento de um grupo de valores constantes. - Citado em 26 , 27
Excel	Editor de outras planilhas produzido pela empresa Microsoft. - Citado em 2 , 3 , 9
feedback	Avaliação de uma ação ou de uma pessoa que leva em consideração uma série de fatores sobre o processo almejado; pode ser positiva ou negativa. - Citado em 11
framework	Junção de códigos, como bibliotecas, que traz como resultado final uma funcionalidade genérica, a fim de resolver problemas recorrentes e agilizar o desenvolvimento da aplicação. - Citado em 2 , 3 , 16 , 18
GitHub	Sistema utilizado para hospedar códigos e arquivos versionados pelo Git. Disponível no endereço <https://github.com/> . - Citado em 19
Globo	Rede de televisão comercial brasileira aberta. - Citado em 11
LaTeX	Sistema/programa de marcação de documentos, com alta tipografia; padrão para textos científicos. - Citado em 20
login	Processo de acesso em um sistema através dos dados de identificação de um usuário. - Citado em 9
Mozilla	Comunidade global de software livre responsável por criar o navegador Firefox, dentre outros navegadores e aplicações. - Citado em 16
open source	Código-fonte aberto de um software, adaptável à finalidade desejada.(NASCIMENTO, 2014) - Citado em 2 , 16 , 17 , 18
permuta	Trocas; nesse caso, de aulas. - Citado em 9 , 10 , 12
permutação	Processo de troca; nesse caso, entre dois docentes, seguindo todos critérios de regramento. - Citado em 9 , 10 , 12
Pinterest	Rede social de compartilhamento de fotos - Citado em 16
Professor	Professor, enum que representa o docente no sistema. - Citado em 12

Python	Linguagem de programação de alto nível, utilizada em aplicações desktop, web, servidores e em ciência de dados. - Citado em 2 , 3
SuperAdmin	Administrador Superior, enum que representa os funcionários que receberão essa funcionalidade naquele semestre/ano. - Citado em 12