

**UNIVERSIDADE CATÓLICA DE SANTOS**  
**Centro de Ciências Exatas, Arquitetura e**  
**Engenharia**  
**Ciência da Computação e Sistemas de Informação**

**Augusto Emanuel Batista Novaes Santos**

**Gabriel dos Santos Bello Botelho**

**Guilherme Marques Tavares**

**Jean dos Santos Gomes da Silva**

**Pablo Luis dos Santos Alves**

**Rafaela Mendes Lomba Pinho**

**Desenvolvimento de Plataforma ALUMNI – Unisantos**

Santos, SP  
2023

## **Desenvolvimento de Plataforma ALUMNI – Unisantos**

Projeto de extensão para o desenvolvimento de um sistema de informação para uma plataforma ALUMNI da comunidade UNISANTOS, apresentado como projeto de extensão dos cursos de Ciência da Computação e Sistemas de Informação, com início em março de 2023 e previsão de conclusão da primeira etapa do projeto em dezembro de 2024.

**Project Owner: Profª Me. Lilian Matheus Marques**

## **Sumário**

<b>1. METODOLOGIA DE DESENVOLVIMENTO.....</b>	<b>4</b>
1.1 Tecnologias.....	4
1.2 AUTENTICAÇÃO .....	6
1.3 ARQUITETURA.....	7
1.4 CONCEITOS .....	8
<b>2 RESULTADOS OBTIDOS .....</b>	<b>11</b>
<b>3 APRENDIZADO DOS ALUNOS .....</b>	<b>18</b>
<b>4 VALOR ENTREGUE A SOCIEDADE .....</b>	<b>19</b>
<b>5 PROXIMAS ETAPAS.....</b>	<b>20</b>
<b>6 CONCLUSÃO .....</b>	<b>21</b>
<b>7 FEEDBACKS.....</b>	<b>22</b>

## 1. METODOLOGIA DE DESENVOLVIMENTO

Após um amplo estudo referente aos recursos que mais se adequam as limitações do projeto, concluímos que para o desenvolvimento da ferramenta, será necessário utilizar os seguintes recursos:

### 1.1 Tecnologias

Para o desenvolvimento do projeto de criação da plataforma ALUMNI da Universidade Católica de Santos, serão utilizadas várias tecnologias. A seguir, vou detalhar cada uma delas:

PHP (versão 8.0): O PHP é uma linguagem de programação amplamente utilizada para o desenvolvimento de aplicações web. A versão 8.0 é a mais recente, trazendo diversas melhorias e recursos. O PHP será a base do desenvolvimento da plataforma, permitindo a criação da lógica de negócio, manipulação de dados e interação com o banco de dados.

Composer: É uma ferramenta de gerenciamento de dependências para o PHP. Ele permite que você especifique as bibliotecas e pacotes necessários para o seu projeto e lida automaticamente com a instalação e atualização dessas dependências. Com o Composer, será possível versionar a aplicação e gerenciar as bibliotecas utilizadas no desenvolvimento da plataforma.

Doctrine: É uma biblioteca de persistência de dados para o PHP, que fornece um conjunto de ferramentas para mapear objetos PHP para estruturas de banco de dados relacionais. Com o Doctrine, será mais fácil e eficiente trabalhar com o banco de dados da aplicação, neste caso, o Postgres.

React: É uma biblioteca JavaScript amplamente utilizada para a criação de interfaces de usuário (UI). Ele permite o desenvolvimento de interfaces dinâmicas e interativas, facilitando a construção do front-end da aplicação ALUMNI. Além do React, também serão utilizados conceitos de HTML5 e CSS3 para estruturar e estilizar os elementos da interface. Além disso, o uso do TypeScript, uma linguagem superset do JavaScript, trará recursos adicionais de tipagem estática e aumento da produtividade no desenvolvimento.

Symfony (para o PHP 8): É um framework PHP que fornece um conjunto de componentes e ferramentas para desenvolvimento web. Ele facilita a criação de aplicações robustas e escaláveis, seguindo as melhores práticas de desenvolvimento. O Symfony será utilizado para aproveitar seus recursos e funcionalidades específicas para o PHP 8, que ainda estão em discussão.

PHPUnit: É um framework para realização de testes unitários no PHP. Ele fornece uma série de ferramentas e recursos para criar e executar testes automatizados, garantindo a qualidade do código e facilitando a identificação de possíveis erros e problemas. Com o PHPUnit, será possível criar testes para verificar o correto funcionamento das diversas partes do código do projeto.

Selenium: É uma ferramenta para automação de testes em navegadores web. Ele permite criar testes automatizados que simulam a interação de um usuário com a aplicação em um navegador real. Com o Selenium, será possível criar testes que verifiquem o correto funcionamento da plataforma ALUMNI em diferentes navegadores, garantindo a funcionalidade e usabilidade em diferentes ambientes. Isso é especialmente importante considerando a variedade de navegadores utilizados pelos usuários.

O Selenium oferece recursos poderosos para interagir com elementos da página, preencher formulários, clicar em botões e validar resultados. Com essa ferramenta, será possível automatizar testes complexos, reduzindo a necessidade de testes manuais repetitivos e aumentando a eficiência do processo de desenvolvimento.

Em resumo, o Selenium será utilizado no projeto ALUMNI para elaborar testes automatizados que verifiquem a correta funcionalidade da plataforma em diferentes navegadores web. Isso permitirá identificar possíveis problemas e garantir uma experiência consistente e confiável para os usuários.

Dessa forma, com a utilização das tecnologias mencionadas, como PHP 8.0, Composer, Doctrine, React (com conceitos de HTML5, CSS3 e TypeScript), Symfony, PHPUnit, Selenium e Postgres, a plataforma ALUMNI da Universidade Católica de Santos será desenvolvida de maneira eficiente, robusta e com alta qualidade. Essas tecnologias oferecem recursos e ferramentas poderosas para atender aos requisitos do projeto, proporcionando uma experiência positiva tanto para os usuários atuais quanto para os ex-alunos e demais membros da comunidade acadêmica.

Para a base de dados, utilizaremos o Postgres, que é um sistema de gerenciamento de banco de dados relacional (RDBMS) de código aberto. Ele será utilizado como a base de dados da aplicação ALUMNI, armazenando todas

as informações necessárias, como cadastros de usuários, informações dos cursos, entre outros dados relevantes. O Postgres é conhecido por sua confiabilidade, desempenho e recursos avançados, tornando-o uma escolha sólida para a plataforma.

Em resumo, a plataforma ALUMNI da Universidade Católica de Santos será desenvolvida utilizando PHP 8.0 como linguagem de programação, com o suporte do Composer para gerenciamento de dependências. A persistência de dados será feita com o auxílio do Doctrine, facilitando a interação com o banco de dados Postgres.

No lado do front-end, será utilizado o React para a criação da interface de usuário, aproveitando os conceitos de HTML5 e CSS3 para estruturação e estilização. Além disso, o TypeScript será utilizado para trazer recursos adicionais de tipagem estática.

Para garantir a qualidade do código, serão elaborados testes unitários com o PHPUnit, verificando o correto funcionamento de cada parte do código. Além disso, o Selenium será utilizado para a criação de testes automatizados, simulando a interação do usuário com a aplicação em diferentes navegadores.

Por fim, o Postgres será o banco de dados escolhido para armazenar todas as informações da plataforma, garantindo a confiabilidade e desempenho necessários.

Essas tecnologias foram selecionadas visando criar uma plataforma robusta, escalável e de alta qualidade, atendendo aos requisitos do projeto ALUMNI da Universidade Católica de Santos.

## 1.2 AUTENTICAÇÃO

Pensando nos aspectos de segurança da informação na aplicação, iremos utilizar alguns recursos para a elaboração da autenticação da aplicação, sendo eles:

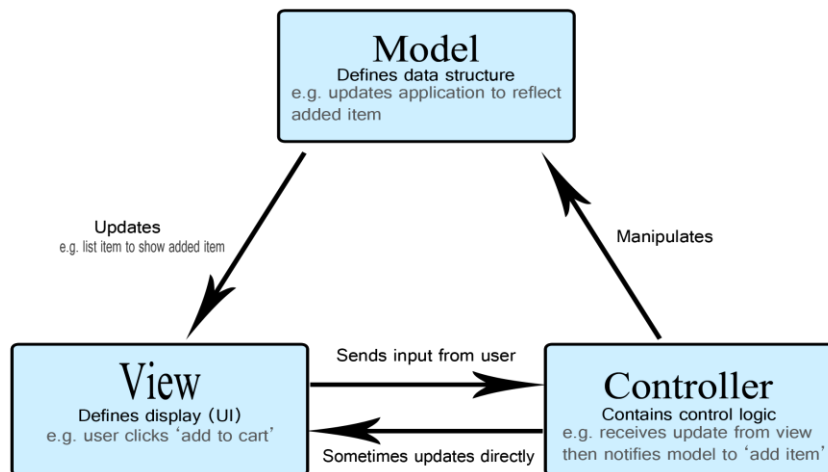
- **Persistência de dados:** Será utilizado o PDO (PHP Data Object) para estabelecer uma conexão segura com o banco de dados. Além disso, iremos utilizar um ORM denominado Doctrine para efetuar a persistência

de dados, de maneira com que possamos efetuar o controle e todo o tratamento de informações que serão direcionadas diretamente para nossa base de dados;

- **Criptografias:** Serão utilizados algoritmos de hash (como a própria criptografia MD5 do PHP) para criptografar os dados sensíveis do usuário, de maneira com que possamos fornecer uma proteção mais forte para a aplicação;
- **Json Web Token (JWT):** É um padrão aberto (RFC 7519) para a criação de tokens de acesso baseados em JSON. É uma forma compacta e autônoma de transmitir informações entre partes de forma segura por meio de objetos JSON. Os tokens JWT são usados principalmente para autenticação e autorização em aplicativos web e serviços.

### 1.3 ARQUITETURA

O padrão MVC (Model-View-Controller): é amplamente utilizado em projetos PHP. Ele separa as preocupações do aplicativo em três componentes principais: Model (modelo), View (visão) e Controller (controlador). O modelo é responsável pela manipulação dos dados e regras de negócio, a visão é responsável pela apresentação dos dados ao usuário e o controlador gerencia a interação entre o modelo e a visão, tratando as requisições do usuário e coordenando as ações apropriadas.



## 1.4 CONCEITOS

Clean Code: refere-se a um estilo de escrita de código que é fácil de entender, manter e evoluir. Um código limpo segue princípios e boas práticas que visam tornar o código legível, conciso e de fácil compreensão. Isso inclui dar nomes significativos às variáveis, funções e classes, escrever funções curtas e bem estruturadas, evitar repetições desnecessárias, entre outros aspectos. O objetivo é produzir um código que seja de alta qualidade, fácil de dar manutenção e que possa ser compreendido por outros desenvolvedores. Como dito por Robert Cecil Martin “Tornar seu código legível é tão importante quanto torná-lo executável”.

Clean Architecture: É um estilo arquitetural que visa separar as preocupações e tornar o código fonte independente de frameworks, bancos de dados e tecnologias externas. A ideia central é definir camadas ou níveis de abstração dentro da aplicação, onde as camadas internas contêm a lógica de negócio e as camadas externas são responsáveis pela integração com o mundo externo, como interfaces de usuário, bancos de dados, APIs, entre outros. A Clean



Architecture promove a manutenção do código, a testabilidade e a independência tecnológica.

Domain-Driven Design (DDD) :É uma abordagem de desenvolvimento de software que coloca o domínio da aplicação no centro do projeto. Ela enfatiza a colaboração próxima entre especialistas do domínio e desenvolvedores para criar um modelo de domínio rico e bem compreendido. O DDD propõe a criação de uma linguagem ubíqua compartilhada entre os especialistas do domínio e os desenvolvedores, e a utilização de padrões de design como Agregados, Entidades, Serviços de Domínio e Repositórios para representar e gerenciar o domínio de forma eficiente.

Test-Driven Development (TDD): É uma prática de desenvolvimento de software que enfatiza a escrita de testes automatizados antes da implementação do código. O ciclo do TDD consiste em escrever um teste que falhe, implementar a funcionalidade mínima para que o teste passe e, em seguida, refatorar o código para melhorar sua qualidade. Essa abordagem auxilia na criação de um código mais limpo, com maior cobertura de testes e menos bugs, além de promover um design de software mais modular e coeso.:

SOLID: É um conjunto de princípios de design de software que visam criar código de qualidade, flexível e de fácil manutenção. Cada letra do acrônimo SOLID representa um princípio específico:

- **Single Responsibility Principle** (Princípio da Responsabilidade Única): Uma classe deve ter apenas uma responsabilidade.
- **Open-Closed Principle** (Princípio Aberto-Fechado): Entidades de software devem ser abertas para extensão, mas fechadas para modificação.
- **Liskov Substitution Principle** (Princípio da Substituição de Liskov): Subtipos devem ser substituíveis por seus tipos base sem afetar a integridade do sistema.

- **Interface Segregation Principle** (Princípio da Segregação de Interfaces): Clientes não devem ser forçados a depender de interfaces que não utilizam.
- **Dependency Inversion Principle** (Princípio da Inversão de Dependência): Módulos de alto nível não devem depender de módulos de baixo nível. Ambos devem depender de abstrações.

Esses princípios SOLID fornecem diretrizes para escrever código limpo, modular e extensível. Eles promovem a separação de preocupações, o baixo acoplamento entre os componentes, a reutilização de código e a facilidade de manutenção.

Ao aplicar esses princípios, os desenvolvedores podem criar sistemas mais robustos, flexíveis e fáceis de dar manutenção. Eles ajudam a evitar a criação de código redundante, complexo e difícil de entender. Além disso, a adesão aos princípios SOLID facilita a realização de testes automatizados, pois o código se torna mais modular e isolado.

Em resumo, a adoção dos princípios SOLID, juntamente com as práticas de Clean Code, Clean Architecture, DDD e TDD, contribui para o desenvolvimento de software de alta qualidade, que atende aos requisitos de negócio, é facilmente mantido, testável e evoluível ao longo do tempo

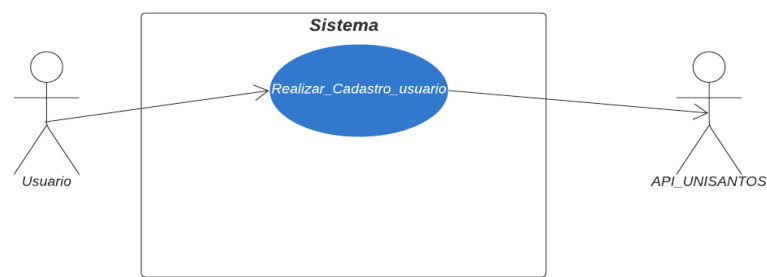
## 2 RESULTADOS OBTIDOS

Até o presente momento, não foi efetuado o desenvolvimento propriamente dito de nenhuma funcionalidade referente a plataforma ALUMNI, porém os protótipos de casos de uso desenvolvidos explicando as funcionalidades iniciais de cadastro e login foram validados e estão com previsão de início de desenvolvimento no primeiro semestre de 2024. Abaixo, podemos identificar os casos de uso com a explicação referente as funcionalidades básicas de login e cadastro, assim como um protótipo de tela:

RQ 1: Cadastro do usuário.

Descrição: O sistema deve ser capaz de armazenar o nome, senha, e-mail, data de nascimento, RM (registro de matrícula), cursos, e o tipo de usuário.

Nome	Descrição	Categoria	Desejável	Permanente
NF 1.1: Tempo de cadastro	O cadastro deve ser feito em até 3 segundos	Performance	(X)	



**Descrição:** O usuário deverá inserir seu RM. Com isso, o sistema irá efetuar uma busca para retornar e-mail institucional, nome, cursos e o telefone do usuário.

**Ator:** Usuário;

**Pré-Condições:** Possuir um vínculo com a universidade (Aluno, Ex-Aluno, Professor, Funcionário...);

**Cenário Principal:**

1. O usuário entrará com o RM;
2. O sistema irá retornar: e-mail institucional, nome do usuário, cursos do usuário e o telefone do usuário;
3. O usuário irá validar se as informações retornadas estão atualizadas;
4. O usuário irá informar a senha para login na plataforma;
5. O usuário clicará em cadastrar.

**Cenário Alternativo:**

- 1A. RM não localizado. O sistema deverá informar que o RM não foi localizado;
- 2A. As senhas não coincidem. O sistema informará que as senhas digitadas não coincidem;
- 3A. E-mail inválido. Caso o usuário altere o e-mail retornado ao informar o RM para um e-mail que não seja @unisantos, o sistema deverá informar que o e-mail é inválido;
- 4A. Algum campo em branco. O sistema não deverá cadastrar o usuário, informando que existem campos em branco e que os mesmos devem ser preenchidos;

**5A.** Alguma informação inválida. O sistema não deverá cadastrar o usuário, informando qual informação se encontra inválida;

**6A.** Termos de uso não aceitos. O sistema não deverá cadastrar o usuário, informando que os termos de uso não foram aceitos.



**Figura:** Processo 1, pesquisa de RM na API.



**Figura:** RM não localizado.

The image shows a web browser window with a 'CADASTRO' (Registration) page. The page has a header with a logo and the title 'CADASTRO'. Below the header, there is a form with the following fields:

- RM (Registration Number)
- Email
- Nome Completo (Full Name)
- Celular (Cellular Phone Number)
- Situação (Status)
- Cursos (Courses): A list box containing 'Ciência da Computação', 'Sistemas de Informação', 'Análise e Desenvolvimento de Sistemas', and 'Ciência de Dados'.

At the bottom of the form is a 'Cadastrar' (Register) button.

**Figura:** Processo 2, formulário de cadastro.

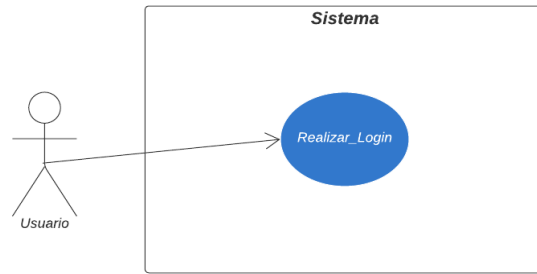


**Figura:** Processo 2, cadastro realizado com sucesso.

RQ 2: Login do usuário.

Descrição: O sistema deve possibilitar o login de usuários já cadastrados.

Nome	Descrição	Categoria	Desejável	Permanente
NF 2.1: Tela de login	A tela de login deve conter apenas dois campos	Usabilidade	(X)	
NF 2.1: Tempo de login	O login deve ocorrer em até 3 segundo	Desempenho	(X)	



**Descrição:** Após o usuário efetuar seu cadastro o sistema irá permitir que ele faça seu login para poder utilizar a ferramenta;

**Ator:** Usuário

**Pré-Condições:** Possuir um cadastro

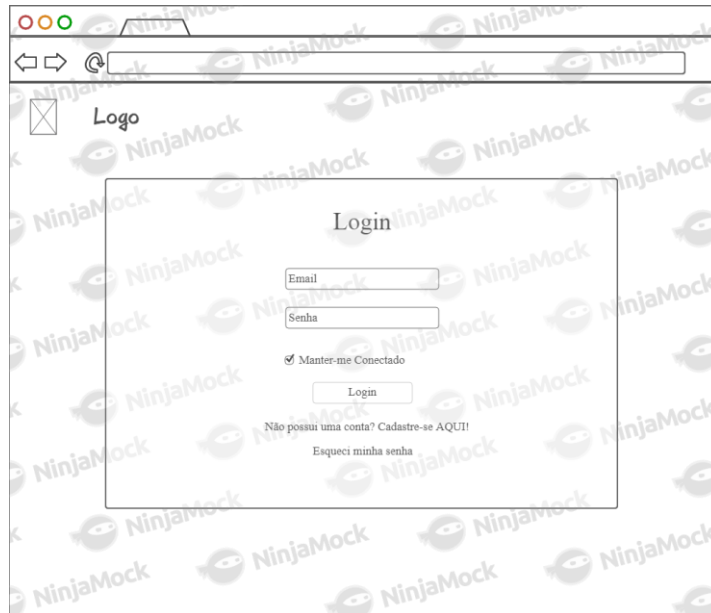
**Cenário Principal:**

1. O usuário irá informar seu e-mail institucional
2. O usuário irá digitar a sua senha
3. O usuário irá clicar no botão login

**Cenário Alternativo:**

**3A.** Dados inválidos. O sistema informará mostrará a seguinte mensagem: “E-mail e/ou senha inválidos”.





A mockup of a web browser window showing a login form. The browser's address bar is empty. The page header includes a placeholder for a logo and the word "Logo". The main content area is a white box with a black border containing the following elements:

- The word "Login" centered at the top.
- An "Email" input field.
- A "Senha" (Password) input field.
- A checked checkbox labeled "Manter-me Conectado" (Keep me connected).
- A "Login" button.
- Two links at the bottom: "Não possui uma conta? Cadastre-se AQUI!" (Don't have an account? Register here!) and "Esqueci minha senha" (Forgot my password).

**Figura:** Tela de Login.

### **3 APRENDIZADO DOS ALUNOS**

Durante o desenvolvimento inicial do projeto, os alunos tiveram a oportunidade de aprimorar habilidades técnicas e práticas, além de desenvolver competências interpessoais. Foi necessário que se envolvessem ativamente na codificação, design, testes e implementação da plataforma, o que resultou em um aprendizado significativo em programação, gerenciamento de banco de dados e arquitetura de sistemas.

Além disso, a futura aplicação prática dos conceitos de Clean Code, Clean Architecture, Domain-Driven Design, Test-Driven Development e princípios SOLID irá proporcionar aos alunos uma compreensão aprofundada destas metodologias.

Importante destacar também o desenvolvimento de habilidades em trabalho em equipe, comunicação e gestão de projetos, fundamentais para o sucesso no ambiente de trabalho. A interação com o Product Owner (PO) e os stakeholders do projeto, permitindo assim entender melhor as necessidades dos usuários e a importância de criar soluções que atendam a esses requisitos de forma eficaz.

Finalmente, o projeto proporcionou uma compreensão clara da importância da conformidade com normas e regulamentações, como a Lei Geral de Proteção de Dados (LGPD), enfatizando a necessidade de desenvolver soluções tecnológicas responsáveis e seguras.

## **4 VALOR ENTREGUE A SOCIEDADE**

A plataforma ALUMNI irá reforçar muito mais o relacionamento de alunos, ex-alunos, professores e todos aqueles que em algum momento possuíram algum vínculo com a Universidade Católica de Santos, onde será possível que eles conversem sobre os mais variados assuntos pertinentes aos seus cursos, áreas e temas que lhe agradem, sendo um ambiente de total interação daqueles que estão ou já estiveram presentes no “universo Unisantos”.

5 PROXIMAS ETAPAS

Para as próximas etapas, teremos o início do desenvolvimento da plataforma, com previsão de conclusão da primeira etapa em dezembro de 2024. Abaixo é possível visualizar algumas estimativas realizadas pelos alunos (estimativas essas feitas em horas):

1	Esforço das Tarefas										
	Orçamento	Análise e Desenho da Solução	Desenvolvimento e Testes Automatizados	Testes de Qualidade	Suporte à Homologação	Configuração e Publicação	Total				
2											
13	1.1.1.2.1.1 Validar Fluxo Feliz	8	8	8	8	8	40				
14	1.1.1.2.1.2 Validar Cenários Alternativos	8	8	8	8	8	40				
15	1.1.1.2.2 Integrar com API Unisantos	8	8	8	8	8	40				
16	1.1.1.2.2.1 Validar Fluxo Feliz	8	8	8	8	8	40				
17	1.1.1.2.2.2 Validar Cenários Alternativos	8	8	8	8	8	40				
18	1.1.1.2 Requisito Não Funcional						0				
19	1.1.2.1 Garantir Performance De Cadastro	8	8	8	8	8	40				
20	1.1.2.2 Garantir Disponibilidade da API Unisantos	8	8	8	8	8	40				
21	1.1.2.3 Garantir Estabilidade do Servidor	8	8	8	8	8	40				
22	2.0 APLICAÇÃO DAS TECNOLOGIAS						0				
23	2.1 PHP						0				
24	2.1.1 Desenvolver API de Cadastro	8	8	8	8	8	40				
25	2.1.2 Desenvolver de API de Login	8	8	8	8	8	40				
26	2.1.3 Integrar com sistema Unisantos	8	8	8	8	8	40				
27	2.2 Composer						0				
28	2.2.1 Instalar e Configurar o gerenciador de pacotes	8	8	8	8	8	40				
29	2.3 Doctrine						0				
30	2.3.1 Instalar e Configurar a ferramenta	8	8	8	8	8	40				
31	2.3.2 Desenvolver Entidades Relacionais	16	8	8	8	8	48				
32	2.4 React						0				
33	2.4.1 Instalar o node (versão LTS mais recente)	8	8	8	8	8	40				
34	2.4.2 Instalar o NPM (versão LTS mais recente)	8	8	8	8	8	40				
35	2.4.3 Instalar o React (versão LTS mais recente)	8	8	8	8	8	40				
36	2.4.3.1 Criar Projeto e Instalar pacotes de desenvolvimento	8	8	8	8	8	40				
37	2.5 Symfony						0				
38	2.5.1 Instalar e Configurar a ferramenta	8	8	8	8	8	40				
39	2.6 PHPUnit						0				
40	2.6.1 Instalar e Configurar a ferramenta	8	8	8	8	8	40				
41	2.7 Selenium						0				
42	2.7.1 Instalar e Configurar a ferramenta	8	8	8	8	8	40				

< >

EAP\_Complexidade

Resumo\_Orcamento

Parametros

Esforco\_Horas

Cluster\_T-ShirtSize

+

:

2.6.1 Instalar e Configurar a ferramenta	8	8	8	8	8	40	
2.7 Selenium						0	
2.7.1 Instalar e Configurar a ferramenta	8	8	8	8	8	40	
2.8 PostgresSQL						0	
2.8.1 Instalar e Configurar a ferramenta	8	8	8	8	8	40	
3.0 ARQUITETURA						0	
3.1 Estruturar Arquitetura Base do Sistema							
3.1.1 Desenvolver camada Model	16	16	8	8	8	56	
3.1.2 Desenvolver camada View	16	16	8	8	8	56	
3.1.3 Desenvolver camada Controller	16	16	8	8	8	56	
4.0 ESTRUTURA FISICA						0	
4.1 Verificar disponibilidade do Servidor						0	
4.1.1 Criar repositório do Projeto	8	8	8	8	8	40	
4.1.2 Configurar CI/CD	16	8	8	8	16	56	
4.1.3 Configurar o ambiente de deploy	16	8	8	8	8	48	
4.1.3.1 Configurar o ambiente Local	8	8	8	8	8	40	
4.1.3.2 Configurar o ambiente Produção	8	8	8	8	8	40	

## 6 CONCLUSÃO

Com base nas pesquisas realizadas até o momento e nas conversas com o PO e com todos os envolvidos, seja direta ou indiretamente com o projeto, concluímos que o projeto será extremamente benéfico para a Universidade como um todo, estreitando os laços de todos aqueles que já passaram pela Unisantos, além de claro, ser um grande desafio para os alunos em questão “técnica”, pois será necessário utilizar muitos conceitos de desenvolvimento que não estão presentes na grade do curso, o que será um desafio extra de buscar o conhecimento teórico e aplicá-lo na prática em um projeto tão robusto quanto o sistema ALUMNI UNISANTOS.

## 7 FEEDBACKS

Até o presente momento, não foi desenvolvido nenhuma das funcionalidades iniciais do sistema, porém foram realizadas algumas conversas com os envolvidos do projeto para que possamos realizar uma integração com o sistema da Unisantos, conforme transcrição das ATA's abaixo:

### ATA

**Data:** 17 de agosto de 2023

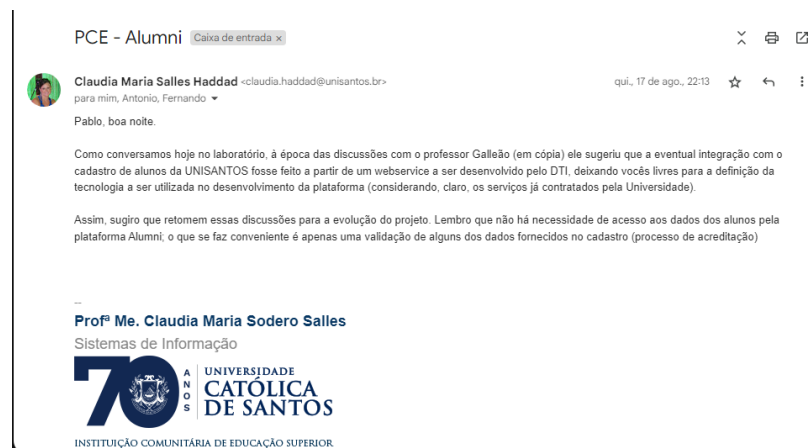
**Local:** Laboratório 5 (Unisantos)

**Presentes:** Claudia Sodero e Pablo luis

**Assunto:** Integração com o cadastro de alunos da UNISANTOS

No dia 17/08/2023, reuniram-se no laboratório 5 os participantes Claudia Sodero e Pablo Luis para discutir a integração do cadastro de alunos da UNISANTOS com a plataforma Alumni.

Durante a reunião, foi destacada a sugestão do professor Galleão de que a integração fosse realizada por meio de um webservice a ser desenvolvido pelo DTI. Esta abordagem permitiria que a equipe de desenvolvimento tivesse liberdade na escolha da tecnologia a ser utilizada, levando em consideração os serviços já contratados pela Universidade.



Ficou estabelecido que a finalidade da integração não seria o acesso direto aos dados dos alunos pela plataforma Alumni, mas sim a validação de determinadas informações fornecidas no cadastro, no processo de acreditação.

Dessa forma, sugere-se que as discussões sejam retomadas para a evolução do projeto, levando em consideração as orientações e sugestões apresentadas durante a reunião.

Atenciosamente,

Pablo Luis

## ATA

**Data:** 18 de agosto de 2023

**Local:** Gmail

**Presentes:** Antonio Galleao, Claudia Salles, Fernando Macedo, Pablo Luis, Rosângela Ballego

**Assunto:** Lei Geral de Proteção de Dados (LGPD) e validação de dados para a plataforma Alumni

No dia 18/08/2023, foi discutida a aplicação da Lei Geral de Proteção de Dados (LGPD) no contexto do projeto da plataforma Alumni. O DTI esclareceu que, de acordo com a LGPD, qualquer dado de aluno só pode ser disponibilizado mediante autorização expressa da reitoria.

Foi ressaltado que a LGPD permite a utilização dos dados coletados apenas para serviços acadêmicos, no caso da Universidade, e para processos de cobrança, no caso da Mantenedora.

Para solicitar a disponibilização de dados para a plataforma Alumni, será necessário encaminhar formalmente um pedido ao reitor, informando quais dados serão necessários, como serão tratados e qual a segurança adotada no processo.

A Professora Rosângela Campanhã se comprometeu a encaminhar a solicitação ao reitor na segunda-feira, 18/08, aproveitando a reunião de reitoria prevista para esse dia.

Foi esclarecido que a plataforma Alumni terá a função de permitir que ex-alunos e alunos cadastrem-se e construam uma rede de relacionamentos pessoais e profissionais. Ao se cadastrar, o interessado fornecerá suas informações, que serão enviadas para o sistema acadêmico para verificação. O sistema acadêmico fornecerá apenas um SIM ou NÃO dessa verificação. Em caso de SIM, a conta será autorizada e receberá um "selo" como conta verificada. Em caso de NÃO, o "selo" não será disponibilizado e a conta precisará passar por um processo de acreditação manual.

Ficou estabelecido que não haverá acesso direto aos dados do sistema acadêmico nem compartilhamento dos mesmos.

A sugestão da Professora Claudia Maria Salles Haddad de verificar se esse procedimento atende à LGPD foi considerada válida. Em caso de necessidade, será avaliada uma alternativa que esteja em conformidade com a legislação.



## Seguem evidências abaixo:



**Claudia Maria Salles Haddad** <claudia.haddad@unisantos.br>

sex., 18 de ago., 14:03



para Rosângela, Antonio, mim, Fernando ▾

Prezados, apenas para esclarecer a questão, e fornecer mais informações para análise:

A questão não é de fornecimento dos dados, e sim de validação. Explico:

A plataforma Alumni permitirá que ex-alunos e alunos cadastrem-se e componham uma rede de relacionamentos pessoais e profissionais. Ao se cadastrar, o interessado fornecerá suas informações. A ideia é que estas informações sejam enviadas para o sistema acadêmico para serem verificadas, e o sistema acadêmico fornecerá apenas um SIM ou NÃO desta verificação. Caso retorne um SIM, a conta é autorizada e recebe um "selo" como conta verificada. Caso retorne um NÃO, este "selo" não é disponibilizado, e a conta precisará, se quiser ser certificada, passar por um processo de acreditação manual.

Ou seja, não haverá acesso aos dados do sistema acadêmico ou mesmo compartilhamento dos mesmos.

Acho importante verificar se este procedimento atende à LGPD, e em caso negativo, podemos avaliar uma outra alternativa que seja adequada.

Obrigada!

\*\*\*



**Rosângela Ballego Campanhã** <rosangelabcampanha@unisantos.br>

sex., 18 de ago., 09:46



para Antonio, Claudia, mim, Fernando ▾

Bom dia,

Encaminharei a solicitação ao reitor na segunda-feira, 18/08 pois temos reunião de reitoria prevista.

Obrigada,

Profa, Rosângela

**Rosângela Ballego Campanhã**

**Pró-Reitora de Graduação**

Universidade Católica de Santos - SP/Brasil

Tel.: +55-13-3205.5555, ramal 1204 - Direto: +55-13-3228-1204

[www.unisantos.br](http://www.unisantos.br)

[rosangelabcampanha@unisantos.br](mailto:rosangelabcampanha@unisantos.br)



**Antonio Miranda Galleao** <galleao@unisantos.br>

sex., 18 de ago., 09:29



para Claudia, mim, Fernando ▾

Bom dia.

Em função da lei de proteção de dados (LGPD), conforme dito anteriormente quando do início deste projeto, o DTI só poderá disponibilizar algum dado de aluno mediante autorização expressa da reitoria.

Lembro que a LGPD só autoriza a utilização dos dados que coletamos para, no caso da Universidade, serviços acadêmicos e, no caso da Mantenedora, para cobrança.

Para tanto, penso que o pedido deve ser encaminhado formalmente ao reitor, informando quais os dados serão necessários, como eles serão tratados, qual a segurança adotada etc.

para conhecimento, copio a profa Rosângela neste email.

obrigado

Atenciosamente,

Pablo Luis

## ATA

**Data:** 22 de agosto de 2023

**Local:** Gmail

**Presentes:** Antonio Galleao, Claudia Salles, Fernando Macedo, Pablo Luis, Rosângela Ballego

**Assunto:** API de Consulta para validação de CPF de ex-alunos

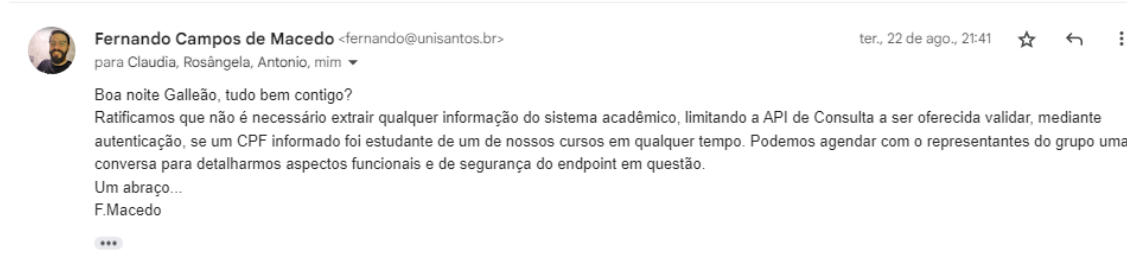
No dia [inserir data], foi discutida a implementação da API de Consulta para validação de CPF de ex-alunos, conforme sugerido pelo Sr. Fernando Macedo.

Ficou esclarecido que não será necessário extrair informações do sistema acadêmico. A API de Consulta terá a finalidade de validar, mediante autenticação, se um CPF informado pertenceu a um estudante de um dos cursos da Universidade em qualquer momento.

Ficou acordado que será agendada uma conversa com representantes do grupo para detalhar os aspectos funcionais e de segurança do endpoint da API em questão.

O Professor Fernando Macedo ratificou o entendimento e manifestou disponibilidade para colaborar no desenvolvimento dessa funcionalidade.

Segue evidências abaixo:



Atenciosamente,

Pablo Luis

## ATA

**Data:** 23 de agosto 2023

**Local:** Gmail

**Presentes:** Antonio Galleao, Claudia Salles, Fernando Macedo, Pablo Luis, Rosângela Ballego

**Assunto:** Escopo da validação de CPF e considerações sobre a obrigatoriedade do CPF na matrícula

No dia 23/08/2023 foi discutida a definição do escopo para a validação do CPF de ex-alunos, conforme abordado na mensagem do Professor Antonio Miranda Galleao.

Ficou esclarecido que a Reitoria concordou em disponibilizar a informação necessária para a validação do CPF.

Foi levantada a questão do escopo da verificação, se deve se restringir a cursos de graduação (licenciatura, bacharelado, tecnológico) ou se deve abranger também pós-stricto sensu, pós lato sensu e extensão. Esta decisão será discutida em reunião posterior com representantes do grupo.

O Professor Galleao ressaltou que o CPF pessoal se tornou obrigatório na matrícula em um período recente, e anteriormente não era necessário que o CPF fosse da própria pessoa, pois eram aceitos CPF do pai, da mãe ou de outro responsável financeiro. Além disso, foi mencionado que no histórico passado existem vários estudantes sem informação de CPF, o que pode impactar na validação.

Foi destacado que a resposta "Não" baseada apenas no CPF pode não ser totalmente correta, levando em consideração essas variáveis.

Seguem abaixo as evidências:



**Antonio Miranda Galleao** <galleao@unisantos.br>  
para Fernando, Claudia, Rosângela, mim ▾

qua., 23 de ago., 09:48 ☆ ↶ ⋮

Bom dia Fernando

Conversei com a Reitoria e eles concordaram que passemos essa informação.

Eu só preciso saber o escopo: temos que verificar apenas se o cpf foi estudado em cursos superior (licenciatura, bacharelado, tecnológico), ou devemos considerar também pós -stricto sensu, pós lato sensu e extensão.

Outra observação é que o cpf pessoal tornou-se obrigatório na matrícula de um tempo para cá - anteriormente ele não era obrigatório que o cpf fosse da própria pessoa - aceitavam cpf do pai, da mãe ou de um outro responsável financeiro.

Temos também, no histórico passado, vários estudantes sem informação de cpf.

Assim, a resposta Não baseada apenas no CPF pode não ser totalmente correta.

Bom aguardo suas considerações para fazermos o endpoint

obrigado  
galleao

\*\*\*

Atenciosamente,

Pablo Luis