



Análise e Desenvolvimento de Sistemas (ADS)
Centro de Educação a Distância (CEAD)

PROJETO INTEGRADO II [ADS0013]

Prof. Allysson Allex Araújo
allysson.araujo@ufca.edu.br

Entregável Parcial 1 (EP1) - Relatório

Nome completo: Joenio Borges de Araújo

Matrícula: 2025017084

Nome completo: Maria Fernanda Sousa Silva

Matrícula: 2025019580

Nome completo: Kauê Oliveira Fernandes

Matrícula: 2025019070

1) Explique de forma robusta e aprofundada como a equipe compreendeu e atendeu aos requisitos do entregável. Apresente evidências (prints, gifs, etc).

Com base em trabalhos anteriores de busca e conceituação da situação que envolvia o objetivo de um painel de acompanhamento escolar para pais/responsáveis, podemos nos nortear no que diz respeito ao entendimento de como seguir com os requisitos do trabalho atual.

1. Título do Projeto

Painel de Acompanhamento Escolar Digital Integrado (PAEDI)

2. Descrição do problema

O cenário da educação brasileira na atualidade enfrenta desafios multifacetados que transcendem a esfera pedagógica. Incidentes críticos em instituições de ensino, evasão escolar e diagnósticos tardios de dificuldades de aprendizagem deixam em

evidência a necessidade de mecanismos de monitoramento e comunicação entre família e escola.

3. Contexto e Justificativa.

O ambiente escolar transformou-se em espaço complexo onde convergem desafios educacionais, sociais e de segurança. Tendo em vista que em muitos casos, uma grande parcela de estudantes passa boa parte do dia em um ambiente escolar - seja por conta de uma modalidade de ensino regular ou integral - torna-se de extrema importância trazer essa questão de como os pais ou responsáveis podem acompanhar a rotina de seus filhos, mesmo que não estejam presentes fisicamente com a frequência esperada. Estar ciente de todos os aspectos que rodeiam o ambiente escolar, contribuem para uma boa relação de segurança, desenvolvimento e bem-estar dos alunos.

4. Público-alvo.

O público-alvo principal do projeto são pais e responsáveis com rotinas profissionais exigentes, que possuem acesso a smartphones, mas têm disponibilidade limitada para o acompanhamento presencial da vida escolar. Sua necessidade específica central é superar a barreira física e de tempo para obter informações transparentes, em tempo real e de forma consolidada. Para atender essa necessidade, o projeto irá fornecer um aplicativo móvel e uma interface web centralizada. Uma necessidade específica deste grupo é a comunicação ágil e direta com a escola, sem depender de telefonemas para a secretaria ou da agenda física, que pode se extraviar. Para atender essa necessidade, o projeto integrará um canal de mensagens direto e segmentado, permitindo que o responsável entre em contato com professores, coordenadores ou a direção de forma assíncrona, recebendo respostas e notificações “push” no celular, similar a aplicativos de mensagens populares, o que se adequa ao seu hábito digital.

Para o grupo dos professores e gestores escolares, a necessidade específica é evitar a duplicação de trabalho e a sobrecarga burocrática. Eles precisam de uma ferramenta que se integre à sua rotina existente. Para atender essa necessidade, o

projeto será desenvolvido com um sistema de gestão integrado, onde o lançamento de uma nota, falta ou ocorrência será feita uma única vez e automaticamente disponibilizada para os responsáveis, eliminando a necessidade de preencher planilhas separadas ou agendas físicas.

5. Proposta de solução.

Nesse contexto, a criação de um software surge como uma oportunidade estratégica para fortalecer essa aproximação, promovendo não apenas a comunicação, mas também o acompanhamento do estudante em todo o seu percurso — da saída de casa ao retorno do convívio familiar.

Esta plataforma oferecerá um painel de controle com acesso imediato a frequência, notas, ocorrências, horário de aulas, calendário de avaliações, cronogramas gerais e específicos, opções para contato direto com núcleo escolar e professores, rotina de transporte escolar, acesso a quem são os funcionários e funções, permitindo que o responsável monitore a rotina do aluno de qualquer lugar e a qualquer hora.

Sendo assim, este relatório tem como objetivo analisar a viabilidade e relevância de um Painel de Acompanhamento Escolar Digital Integrado, fundamentando sua necessidade com base em dados educacionais, marcos legais e demandas sociais contemporâneas.

Por fim, a necessidade específica de todos os envolvidos, incluindo os alunos, é um ambiente escolar mais seguro e integrado. Para atender essa necessidade, o projeto incluirá funcionalidades de alertas proativos. Os responsáveis poderão ser notificados automaticamente sobre eventos críticos, como múltiplas faltas consecutivas, baixo rendimento em uma disciplina ou ocorrências disciplinares, permitindo uma intervenção rápida e coordenada entre família e escola, fortalecendo a rede de apoio em torno do estudante.

Referencias:

<https://www.gazetadopovo.com.br/vida-e-cidadania/entra-e-sai-nas-escolas-expoe-alunos-a-violencia-do-transito-afu6k20r7dle99y232mwi5z66/amp/>

<https://rmmg.org/artigo/detalhes/1400>

<https://novaescola.org.br/conteudo/7355/pais-que-seguem-de-perto-a-rotina>

OCDE. Programa Internacional de Avaliação de Estudantes (PISA) 2022: Resultados Brasileiros. Paris, 2023.

BRASIL. Lei nº 9.394, de 20 de dezembro de 1996. Estabelece as diretrizes e bases da educação nacional.

FUNDAÇÃO LEMANN. Pesquisa sobre Participação Parental na Educação Brasileira. São Paulo, 2024.

Cartão de história de usuário:

Expectativa de uso:

Como professor, gostaria de um ambiente centralizado, no qual pudesse estar a um clique, de fácil acesso, que eu pudesse usar tanto no meu computador, quanto no celular, facilitando o dia a dia de registro de frequência, provas, notas e planejamento.

Como aluno, gostaria de ter um ambiente seguro no qual seja possível autenticar os usuários do sistema, para garantir segurança no acesso autorizado e controlado.

Como Pai/responsável, gostaria de ter acesso à informações pertinentes ao aluno no qual sou responsável, como acompanhamento de notas, atividades, frequências e diagramas. Além disso, uma parte que eu pudesse conversar diretamente com o professor.

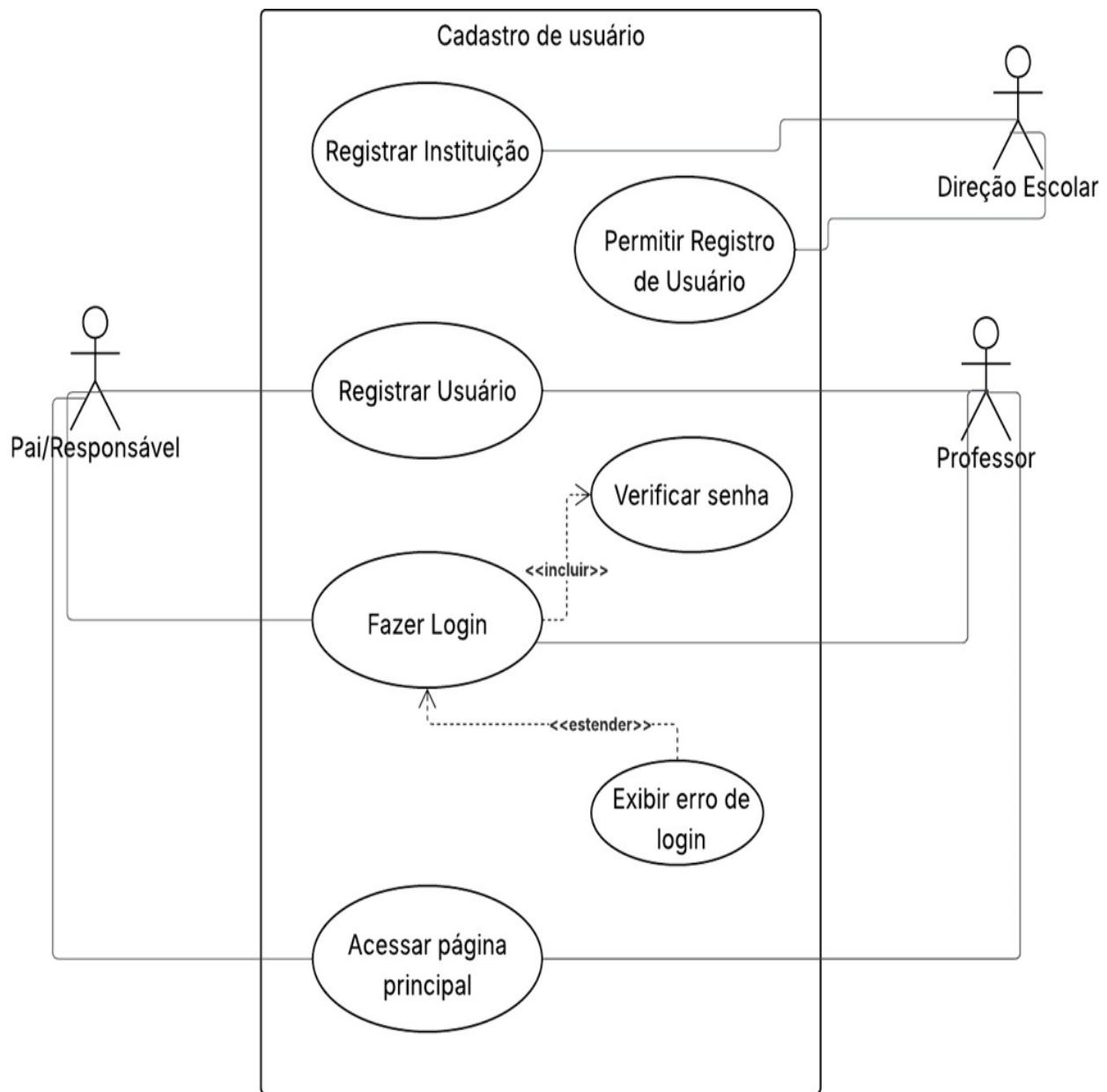
Painel de Acompanhamento Escolar Digital Integrado (PAEDI)

Esses diagramas têm como objetivo exemplificar e trazer de maneira formal as estruturas e funcionalidades principais do projeto de desenvolvimento de uma

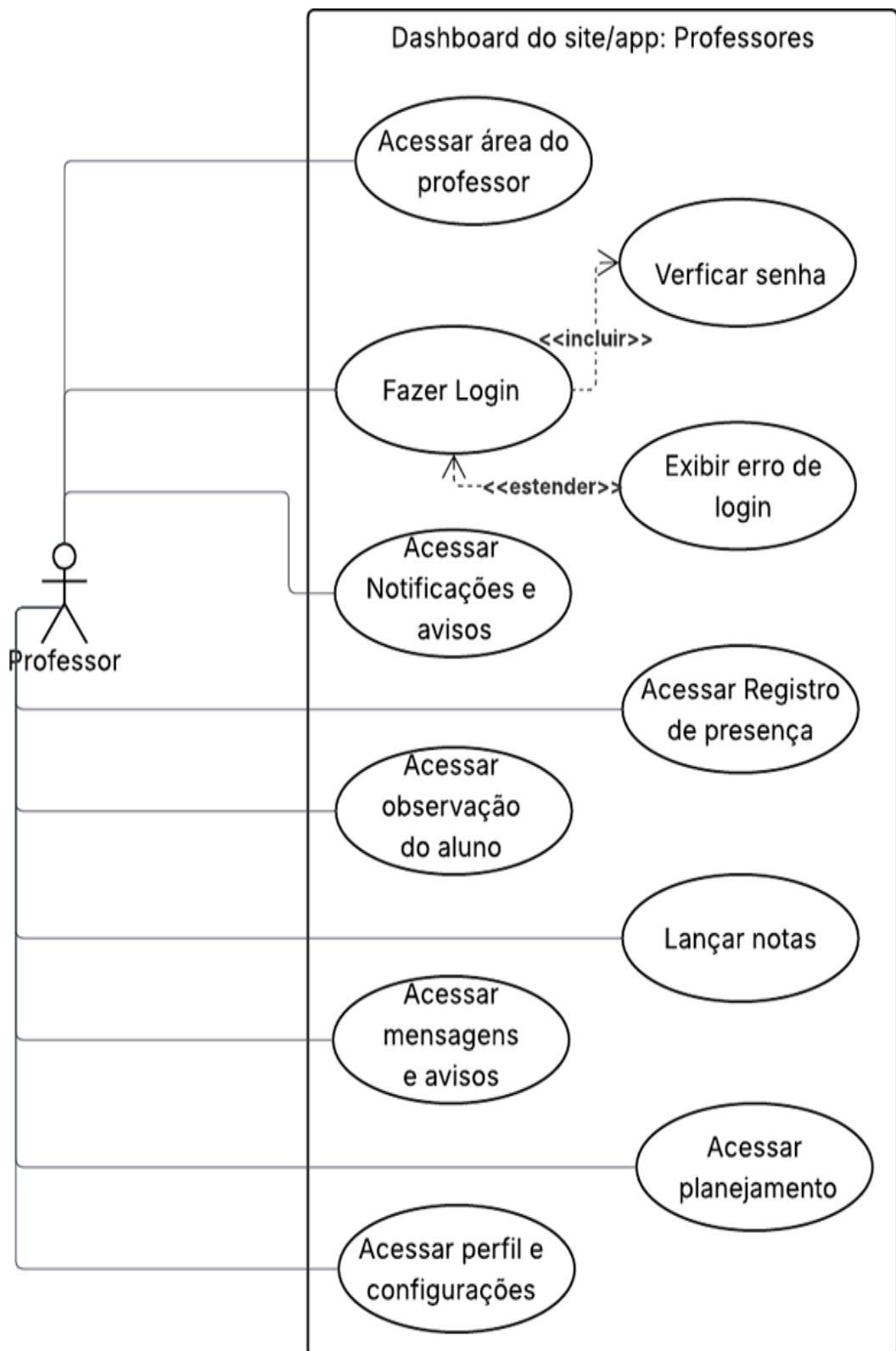
aplicação para auxiliar e trazer uma solução para os pais/responsáveis que desejam ter um maior acesso à informação e contato centralizado, seja pela falta de disponibilidade de acompanhamento assíduo ao ambiente escolar, ou pelo desejo de ter uma opção de poder ter um maior controle da vida escolar como um todo, daqueles que estão sob sua responsabilidade. Como também um ambiente onde possa ter contato através de chat com os professores, na qual também farão uso dessa ferramenta.

Diagrama de Caso de Uso (UML)

Objetivo	Descrever uma sequência de ações realizadas para cadastro e acesso ao sistema, dos atores que utilizarão a aplicação.
Pré-condições	Direção escolar permitir acesso ao sistema.
Fluxo básico	Autor principal cadastra a instituição, permite acesso aos professores e pais/responsáveis para acessar o sistema.



Objetivo	Descrever uma sequência de etapas para acesso e uso da página de trabalho do professor.
Pré-condições	Que o ator realmente desempenhe a função pré-definida para acesso.
Fluxo básico	Início com login da área destinada ao professor e acesso às funções pré-definidas do ambiente de trabalho.
Resultado esperado	Um ambiente de trabalho reservado ao professor com ferramentas de uso para serviço e comunicação com os pais/responsáveis.



Objetivo	Descrever uma sequência de etapas para acessar a área pré-definida para pais/responsáveis.
Pré-condições	Que o ator realmente desempenhe a função pré-definida para acesso.
Fluxo básico	Iniciando com a primeira etapa de validação de usuário adequado para acesso, juntamente com o controle e permissão da direção da instituição.
Resultado esperado	Cadastro e acesso dos pais/responsáveis, para utilização da área do aluno(s) que é responsável.

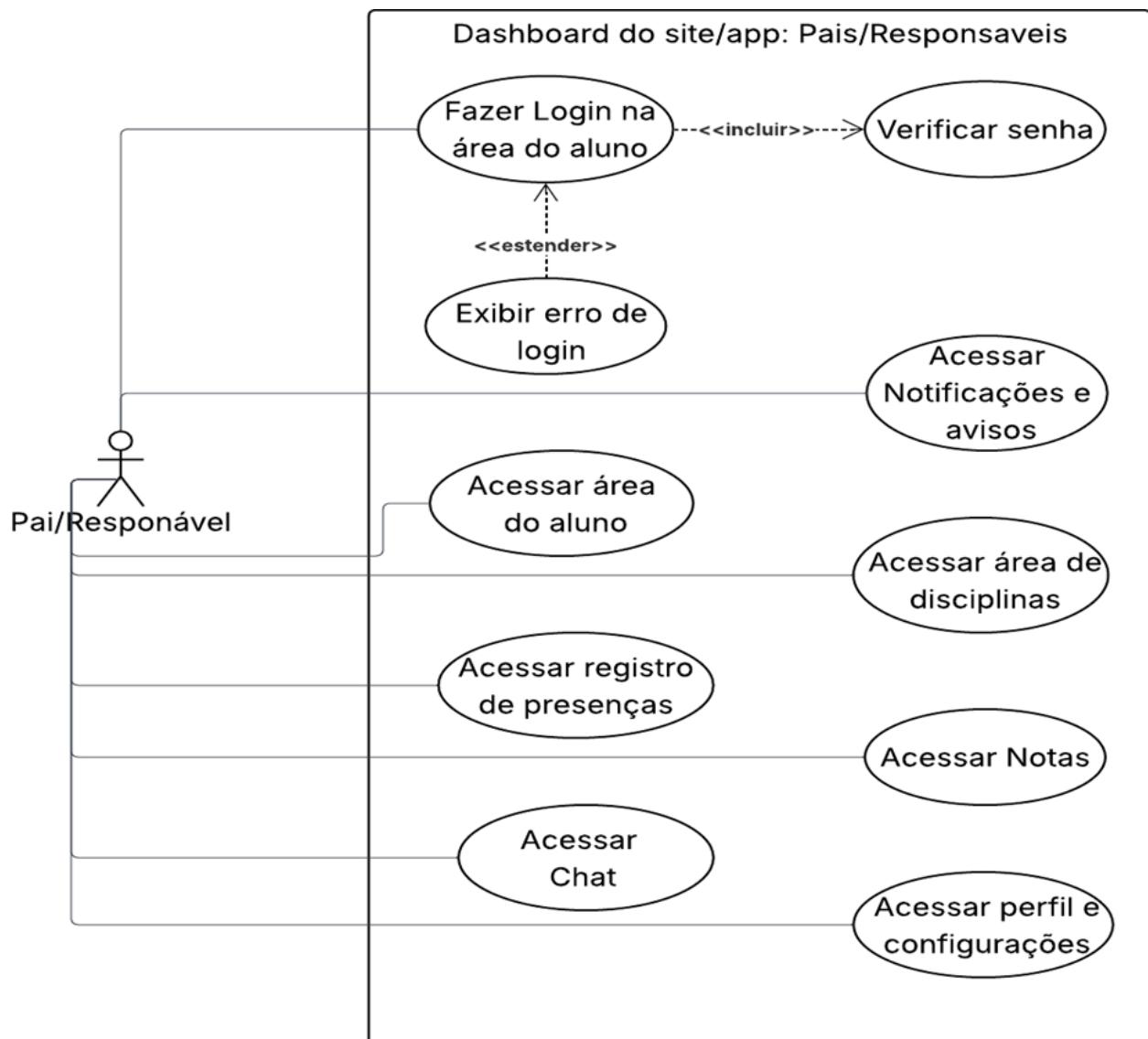
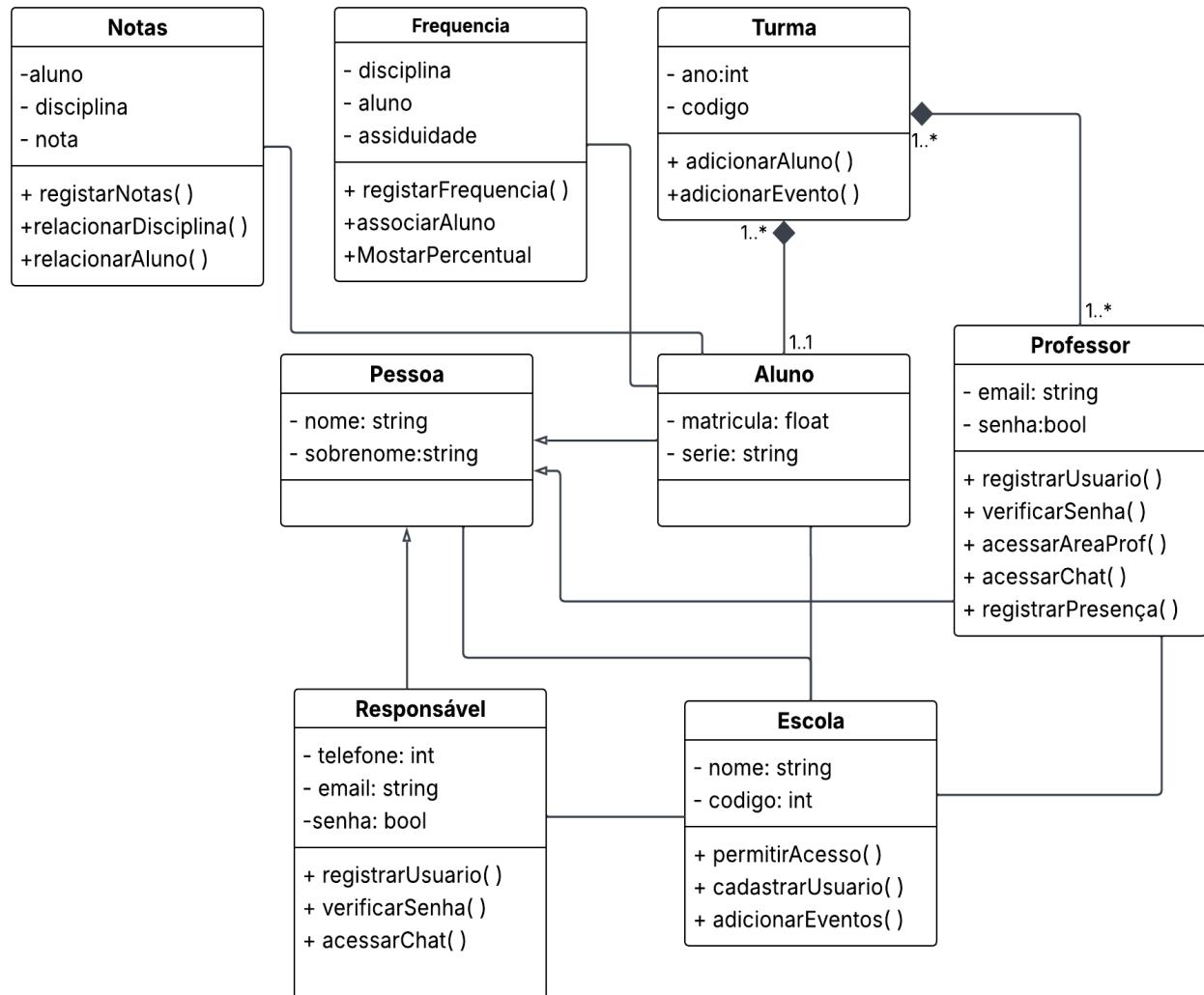


Diagrama de classes:



a) Como as classes principais foram implementadas, com explicação dos código-fonte e de processos de software adotados. O projeto, incluindo código-fonte, deve estar disponível no GitHub;

As classes principais foram implementadas com base no fluxo de informações e desenvolvimento do trabalho, chegando em um diagrama de classes simples, que se baseia na ideia central do projeto. Além disso, foi adotado o Scrum como processo de software, uma metodologia ágil baseada em ciclos curtos de desenvolvimento para implementação de um Produto Mínimo Viável (MVP).

A classe Pessoa foi implementada como classe base do sistema, contendo atributos comuns a todos os usuários, como nome e sobrenome. Essa classe serve como base para reaproveitamento de código, evitando duplicação e facilitando a manutenção do sistema.

A partir da classe Pessoa, foram criadas as classes Aluno, Professor e Responsável, utilizando o conceito de herança. Cada uma dessas classes herda os atributos da classe Pessoa e adiciona características específicas.

A classe Aluno possui atributos como matrícula e série, representando informações acadêmicas.

A classe Professor inclui email e senha, além de métodos que simulam ações como acessar a área do professor, acessar o chat e registrar presença. A senha foi definida como um atributo privado, utilizando o encapsulamento, garantindo maior segurança e controle de acesso.

A classe Responsável também utiliza encapsulamento para o atributo senha e permite acesso ao chat do sistema.

A classe Turma foi implementada para representar a organização dos alunos e professores. Nela, contém listas para armazenar múltiplos alunos e professores, caracterizando um relacionamento de composição. Métodos foram criados para adicionar alunos, professores e eventos à turma, retornando mensagens que confirmam as ações realizadas.

As classes Notas e Frequência foram desenvolvidas para gerenciar o desempenho acadêmico dos alunos. A classe Notas associa um aluno a uma disciplina e permite o registro de notas. Já a classe Frequência controla o percentual de presença do aluno em determinada disciplina, retornando esse valor formatado para exibição no sistema.

A classe Escola representa a entidade principal do sistema, armazenando informações institucionais e uma lista de usuários cadastrados. Ela permite o cadastro de pessoas, a simulação de acesso ao sistema e o registro de eventos gerais da escola.

Como processo de software adotado, o sistema separa claramente a lógica de negócio da exibição de informações. Os métodos das classes retornam valores utilizando a instrução `return`, enquanto a exibição no terminal é realizada por meio da função `print()` na parte final do código. Essa abordagem melhora a organização, facilita testes e torna o código mais reutilizável.

Por fim, a execução do sistema ocorre de forma sequencial, criando os objetos necessários, registrando alunos, professores, notas e frequência, e exibindo os resultados no terminal. Essa estrutura permite simular o funcionamento básico de um sistema escolar, mantendo o código simples e legível.

b) Quais princípios e práticas de POO foram utilizadas para implementar o projeto.

Nessa etapa inicial, o Sistema PAEDI foi desenvolvido utilizando a linguagem de programação Python, aplicando os conceitos fundamentais da Programação Orientada a Objetos (POO), tais como classes, herança, encapsulamento e composição. O objetivo foi modelar, de forma simples e organizada, os principais elementos de um ambiente escolar.

Em algumas situações, algumas classes dependem da outra, chamado assim de “acoplamento”, no caso implementado, ficou com baixo acoplamento, além do método construtor `__init__`, encapsulamento: para proteger os dados internos de uma classe, permitindo acesso controlado por métodos.

c) [Componente Extensionista] No README, criar uma seção com o título "Possíveis usos da nossa solução" e escrever sobre como esse projeto poderia ajudar uma pessoa ou negócio do mundo real. Objetivo: incentivar a conexão entre o projeto e problemas reais, mesmo de forma hipotética e breve.

Muitas escolas, principalmente de pequeno e médio porte, enfrentam diversos problemas no âmbito da instituição, como por exemplo, falta de comunicação eficiente entre escola, professores e responsáveis pelos alunos. Além disso, dificuldade em acompanhar o desempenho do aluno, falta de centralização das informações, esses problemas geram desorganização, retrabalho e insatisfação de ambos que fazem parte.

O projeto proposto deverá atuar como um sistema de gestão escolar, trazendo benefícios para a instituição, como: centralização de dados, redução de erros administrativos, melhora a organização interna, cria base para o crescimento do sistema.

Para os professores, o objetivo é um registro organizado de informações dos alunos, menor dependência de papel ou mensagens informais, histórico dos alunos acessível.

Os pais/responsáveis representam um papel fundamental nesse sistema, pois a ideia principal foi baseada no contexto no qual, esses atores teriam essa preocupação em acompanhar melhor a rotina e todo um meio que acompanha a vida escolar do filho, mesmo em uma rotina cada vez mais conflitante com a disponibilidade de estar acompanhando pessoalmente, pois muitos enfrentam alguma barreira de emprego, geográficas e em alguns contextos, econômico. Tendo em vista o fato de que cada vez mais cresce a quantidade de pessoas com acesso a um smartphone, estar conectado, tornou-se uma realidade bastante acessível, assim podendo ser acessado em qualquer dispositivo nesse contexto, bem como pelo computador.

3) Link do repositório no GitHub ([exemplo](#) de referência).

https://github.com/Joenio97/PI2_PAEDI

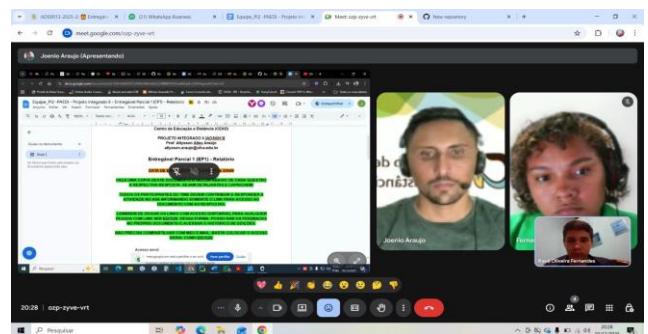
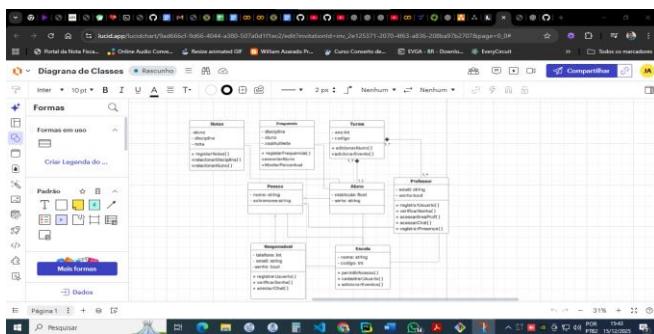
4) Detalhe como cada membro da equipe contribuiu para o desenvolvimento do entregável?

Desde que nós abordamos esse tema sobre o painel de acompanhamento escolar para os pais, buscamos avaliar de que maneira os pais poderiam ter acesso ao desenvolvimento de seus filhos dentro do ambiente escolar sem que precisassem ir na escola ou receber uma ligação misteriosa da coordenação. Depois de várias pesquisas e também pela nossa própria experiência como estudantes, elaboramos o PAEDI (painel de acompanhamento escolar digital para pais/responsáveis) - software para dispositivos móveis - visando fortalecer não só a comunicação, como também a aproximação entre os pais e a escola. Sendo atualizados sobre o desempenho e comportamento dos filhos em tempo real para que os pais monitorem seus filhos de onde estiverem de forma remota e sem preocupação.

O objetivo do nosso projeto é auxiliar os pais em relação a comunicação com a escola, vigilância e controle dos filhos no ambiente escolar. Todos os membros da equipe pensaram de forma conjunta para que esse resultado fosse encontrado, mas podemos destacar os 03 pontos principais para esse projeto onde cada um se desenvolveu melhor. Sendo eles: elaboração, conceito(escrita) e lógica do projeto

- Joenio: Lógica do projeto
- Fernanda: Conceitual do projeto
- Kauê: Elaboração do projeto

5) Insira abaixo evidências das contribuições coletivas (fotos de reuniões, etc) e individuais (que demonstrem o que cada membro fez).



```

class Pessoa:
    def __init__(self, nome, sobrenome, matricula, serie):
        self.nome = nome
        self.sobrenome = sobrenome
        self.matricula = matricula
        self.serie = serie
        self.email = email
        self._senha = senha

    def verifyPassword(self, senha):
        return self._senha == senha

    def accessGranted(self):
        return "Acesso à área do professor concedido"
    def accessGranted(self):
        return "Acesso ao chat concedido"

class Aluno(Pessoa):
    def __init__(self, nome, sobrenome, matricula, serie):
        super().__init__(nome, sobrenome, matricula, serie)
        self.disciplinas = []
        self.notas = {}

    def matricularAluno(self, disciplina):
        self.disciplinas.append(disciplina)

    def registrarNota(self, nota):
        self.notas[disciplina] = nota

    def relatorioAluno(self):
        return f"Relatório de {self.nome} {self.sobrenome}\nMatrícula: {self.matricula}\nSérie: {self.serie}\nDisciplinas: {', '.join(self.disciplinas)}\nNotas: {self.notas}\nFrequência: {self.frequencia}\nFrequência em Matemática: {self.notas.get('Matemática')}\nFIM DA LISTA!\nESCOLAR PAÍS!"

    def estruturaDoCírculo():
        print("ESTRUTURA DO CÍRCULO")

class Professor(Pessoa):
    def __init__(self, nome, sobrenome, matricula, serie):
        super().__init__(nome, sobrenome, matricula, serie)
        self.turmas = []

    def registrarTurma(self, turma):
        self.turmas.append(turma)

    def relatorioProfessor(self):
        return f"Relatório do professor concedido\nNome: {self.nome}\nSobrenome: {self.sobrenome}\nMatrícula: {self.matricula}\nSérie: {self.serie}\nTurmas: {', '.join(self.turmas)}\nPresença registrada"

    def acessarChat(self):
        return "Acesso ao chat concedido"

class Responsavel(Pessoa):
    def __init__(self, nome, sobrenome, matricula, serie):
        super().__init__(nome, sobrenome, matricula, serie)
        self.alunos = []

    def matricularAluno(self, aluno):
        self.alunos.append(aluno)

    def relatorioResponsavel(self):
        return f"Relatório do responsável\nNome: {self.nome}\nSobrenome: {self.sobrenome}\nMatrícula: {self.matricula}\nSérie: {self.serie}\nAlunos: {', '.join([aluno.nome for aluno in self.alunos])}"

```

Nessa etapa inicial, o Sistema PAEDI foi desenvolvido utilizando a linguagem de programação Python, aplicando os conceitos fundamentais da Programação Orientada a Objetos (POO), uso de classes, herança, encapsulamento e composição. O objetivo foi: modelar de forma simples e organizada, os principais elementos de um ambiente escolar.

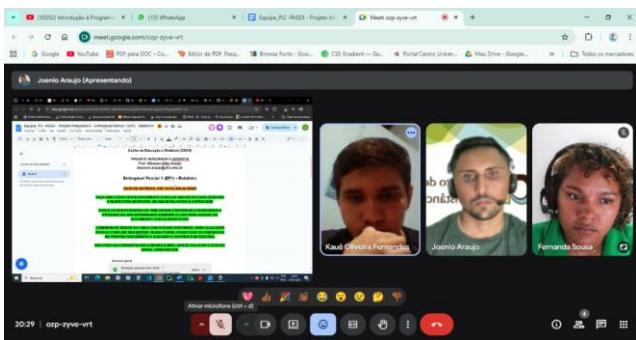
A classe Pessoa é a implementação concreta base do sistema, contendo atributos comuns a todos os usuários: nome, sobrenome. Esta classe serve como base para o reaproveitamento de código, evitando duplicação e facilitando a manutenção do sistema.

A parte da classe Pessoa, foram criadas as classes Aluno, Professor e Responsável, utilizando o conceito de herança. Cada uma dessas classes herda os atributos da classe Pessoa e adiciona novos.

A classe Aluno possui atributos, como matrícula e série, representando informações sobre o estudante.

A classe Professor inclui email e senha, além de métodos que simulam ações como acessar a área do professor, acessar o chat e registrar presença. A senha é definida como um atributo privado, utilizando o encapsulamento, garantindo maior segurança e controle de acesso.

O código é estruturado de forma modular, utilizando pacotes para organizar o código e facilitar a manutenção.



```

class Notas:
    def __init__(self, aluno):
        self.disciplina = disciplina
        self.aluno = aluno
        self.notas = {}

    def registrarNota(self, nota):
        self.notas[disciplina] = nota

    def relatorioAluno(self):
        return f"Relatório de {self.aluno.nome} {self.aluno.sobrenome}\nMatrícula: {self.aluno.matricula}\nSérie: {self.aluno.serie}\nDisciplinas: {', '.join(self.aluno.disciplinas)}\nNotas: {self.notas}\nFrequência: {self.aluno.frequencia}\nFrequência em Matemática: {self.notas.get('Matemática')}\nFIM DA LISTA!\nESCOLAR PAÍS!"

    def estruturaDoCírculo():
        print("ESTRUTURA DO CÍRCULO")

class Professor:
    def __init__(self, nome, sobrenome, matricula, serie):
        self.nome = nome
        self.sobrenome = sobrenome
        self.matricula = matricula
        self.serie = serie
        self.turmas = []

    def registrarTurma(self, turma):
        self.turmas.append(turma)

    def relatorioProfessor(self):
        return f"Relatório do professor concedido\nNome: {self.nome}\nSobrenome: {self.sobrenome}\nMatrícula: {self.matricula}\nSérie: {self.serie}\nTurmas: {', '.join(self.turmas)}\nPresença registrada"

    def acessarChat(self):
        return "Acesso ao chat concedido"

class Responsavel:
    def __init__(self, nome, sobrenome, matricula, serie):
        self.nome = nome
        self.sobrenome = sobrenome
        self.matricula = matricula
        self.serie = serie
        self.alunos = []

    def matricularAluno(self, aluno):
        self.alunos.append(aluno)

    def relatorioResponsavel(self):
        return f"Relatório do responsável\nNome: {self.nome}\nSobrenome: {self.sobrenome}\nMatrícula: {self.matricula}\nSérie: {self.serie}\nAlunos: {', '.join([aluno.nome for aluno in self.alunos])}"

```

