

Gabriel Trevisan
Guilherme Henrique Ventura
João Frederico Delgado Benedito
Matheus Henrique Castilho
Pedro Franco Guidolin
Henry Calle

WikiFam

Tema: Uma proposta de sistema para melhorar interação entre alunos

Ciência da computação
Faculdade de Americana

Americana – SP

2024

Gabriel Trevisan
Guilherme Henrique Ventura
João Frederico Delgado Benedito
Matheus Henrique Castilho
Pedro Franco Guidolin
Henry Calle

WikiFam

Tema: Uma proposta de sistema para melhorar interação entre alunos

Documentação apresentada ao programa
de graduação em ciência da computação
da Faculdade de Americana, como projeto
integrador

Orientador: Prof. Lucas Serafim Parizotto

Americana – SP

2024

Sumário

WikiFam.....	0
Sumário.....	2
Lista de Figuras/Imagens.....	3
Resumo.....	4
Introdução.....	4
Apresentação do Projeto.....	5
Figura 1. Diretório.....	6
Figura 2. CSS.....	7
Requisitos e Viabilidades.....	8
Requisitos e Relacionamentos - BD.....	9
Aplicação do Scrum.....	11
Análise de Riscos do Projeto.....	12
Cronograma.....	13
Figura 3. Cronograma.....	13
Gráfico de Gantt.....	14
Protótipo de tela com aplicação dos requisitos.....	15
Figura 4. Tela de Login.....	15
Figura 5. Homepage.....	16
Figura 6. Perfil de Usuário.....	16
Figura 7. Mural de novidades.....	17
Figura 8. Manual dos calouros.....	18
Figura 9. Quiz de boas vindas.....	18
Figura 10. Dicas e truques.....	19
Figura 11. Dúvidas.....	19
Figura 12. Mural de eventos.....	20
Figura 13. Lista de departamentos.....	21
Figura 14. Mapa Fam.....	22
Figura 15. Dica de mapa.....	22
Figura 16. Lista de Tarefas.....	23
Figura 17. Sugestões.....	23
Figura 18. Atendimento.....	24
Diagramas.....	25
Figura 19. Diagrama de Uso.....	25
Figura 20. Diagrama Conceitual.....	26
Figura 21. Diagrama de Classe.....	27
Figura 22. Diagrama de Sequência.....	29
Figura 23. Diagrama de Atividade.....	30
Figura 24. Diagrama Lógico.....	31
Tabelas e seus Campos:.....	32
Relacionamentos.....	34
Conclusão.....	35
Referência.....	36

Lista de Figuras/Imagens

Figura 1. Diretório.....	6
Figura 2. CSS.....	7
Figura 3. Cronograma.....	12
Figura 4. Tela de Login.....	13
Figura 5. Homepage.....	14
Figura 6. Perfil de Usuário.....	14
Figura 7. Mural de novidades.....	15
Figura 8. Manual dos calouros.....	16
Figura 9. Quiz de boas vindas.....	16
Figura 10. Dicas e truques.....	17
Figura 11. Dúvidas.....	17
Figura 12. Mural de eventos.....	18
Figura 13. Lista de departamentos.....	19
Figura 14. Mapa Fam.....	20
Figura 15. Dica de mapa.....	20
Figura 16. Lista de Tarefas.....	21
Figura 17. Sugestões.....	21
Figura 18. Atendimento.....	22
Figura 19. Diagrama de Uso.....	23
Figura 20. Diagrama Conceitual.....	24
Figura 21. Diagrama de Classe.....	25
Figura 22. Diagrama de Sequência.....	27
Figura 23. Diagrama de Atividade.....	28
Figura 24. Diagrama Lógico.....	29

Resumo

Este trabalho integrador tem a proposta de melhorar a interação entre alunos e também auxiliar em dúvidas e situações comuns apresentadas dentro da instituição. Nós temos como intuito criar um site onde os alunos possam acessar e interagir de uma maneira mais prática e intuitiva tanto para os alunos calouros quanto para os veteranos.

Para os calouros, o Site oferecerá recursos como o Manual do Calouro com dicas e orientações sobre como se adaptar em nosso ambiente acadêmico, Indicadores que facilitarão que os alunos encontrem suas salas, e quizzes para melhorar a interação e entendimento sobre a faculdade. Já para os veteranos, haverá funcionalidades que facilitarão o acompanhamento de novos eventos, Contato com todas as áreas da instituição e uma agenda para facilitar o controle de entregas e datas dos afazeres.

Com essa iniciativa, esperamos não só melhorar a comunicação entre os alunos, mas também criar uma comunidade acadêmica mais coesa e solidária.

Introdução

A WikiFam é um Projeto inovador dedicado a transformar a educação por meio de tecnologia de ponta, nosso sistema acadêmico multifuncional integra uma ampla gama de ferramentas avançadas, incluindo plataformas de aprendizado interativas, gestão acadêmica eficiente e uma rede social exclusiva para alunos e professores, Com o objetivo de promover um ambiente colaborativo e dinâmico, a WikiFam conecta educadores e estudantes, facilitando a troca de conhecimento e a inovação.

Acreditamos que a educação deve ser acessível, envolvente e adaptada às nossas necessidades modernas, e trabalhamos continuamente para aprimorar nossa plataforma e oferecer soluções que empoderam a comunidade acadêmica ao redor do mundo.

Apresentação do Projeto

Visão Geral do WikiFam

O software WikiFam é uma aplicação *web* que foi desenvolvida na FAM-Faculdade de Americana, pela equipe integrada por alunos do 5º Semestre de CC, com o propósito de evoluir os sistemas Acadêmicos e acrescentar algo inovador e novo para método de comunicação dentro e fora das Instituições. A aplicação que inclui inúmeras ferramentas tem como propósito transformar o processo de graduação mais fácil e agradável para alunos, professores e administradores da instituição.

1. Projeto Lógico

Para o desenvolvimento da ferramenta pensamos em utilizar linguagens que se comunicam de forma harmônica, assim foi utilizada a seguinte arquitetura/ estrutura.

BackEnd: Foram utilizados códigos e scripts em NodeJs para a comunicação com MySQL e com o FrontEnd(ReactJs).

Dados: Os Dados são todos armazenados no banco de dados Sql e são retornados para suas devidas funcionalidades.

Apresentação: Após a junção das seguintes funcionalidades os dados são incorporados na estruturação HTML que fica codificada por ReactJs, assim utilizando os Bancos prontos da linguagem.

Nas subseções iremos ver mais detalhadamente os elementos das ferramentas.

3.1 Estrutura de diretórios e subdiretórios

A estrutura de diretórios e subdiretórios contém cada pasta e arquivo utilizado na ferramenta, como é apresentado na *Figura 1*. A arquitetura de pastas contém cada arquivo do sistema WikiFam, são informados nas seguintes subseções a seguir.

3.2 Descrição do Arquitetura

3.2.1. Diretório WikiFam

1. **Sever:** Pasta onde fica toda a configuração do BackEnd
 - a. **node_modules:** Pasta que armazena todos os arquivos prontos do Node Js.
 - b. **Routes:** Pasta que armazena os arquivos *AdminRoute.js* que administra as rotas de autenticação do login e o arquivo *controller.js* que administra as rotas de criação dos comentários.
 - c. **Utils:** Pasta que contém o arquivo *db.js* que serve para fazer a comunicação com o banco de dados(MySQL).
 - d. **index.js:** Arquivo que contém todas as rotas do sistema.
 - e. **package-lock.json:** Arquivo que contém os registros de todas as versões que foram introduzidas na ferramenta.

- f. **package.json:** Arquivo que contém os registros de todas as versões que foram introduzidas na ferramenta e que informa em qual arquivo que ele irá iniciar o BackEnd.
- g. **Wiki_Fam:** Pasta que contém todos os arquivos do FrontEnd.
- h. **node_modules:** Pasta que contém todos os módulos do node.
- i. **public:** Pasta que pode ser utilizada para armazenar imagens, ícones....
- j. **src:** Pasta onde se encontra todos os arquivos que dá vida ao FrontEnd.
- k. **assents:** Pasta utilizada para armazenar imagens e ícones.
- l. **components:** Aqui se encontram todas as pastas do nosso sistema, cada pasta contém um arquivo *Exemplo.jsx*, cada arquivo contém o código das páginas, junto a eles existem arquivos *Exemplo.css* que são utilizados para a estilização das páginas.
- m. **LayOut:** Pasta onde contém um arquivo *LayOut.jsx* que cuida da navegação entre as páginas.
- n. **Pages:** Pasta utilizada para a criação de páginas.

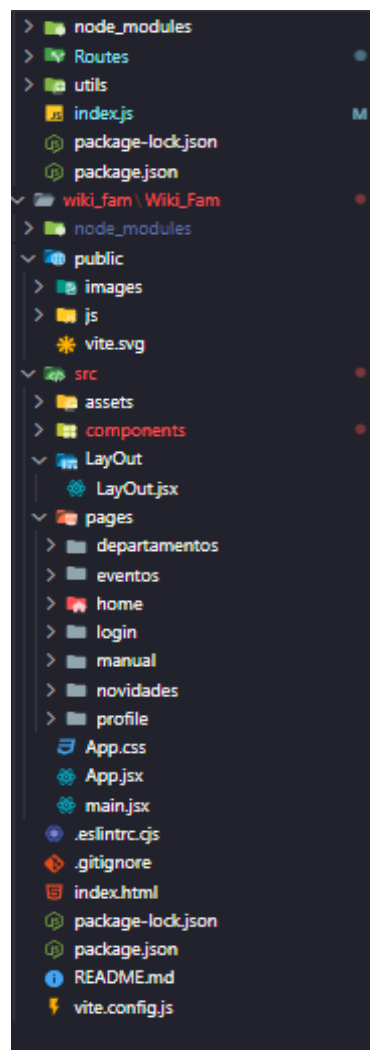


Figura 1. Diretório

3.2.2. CSS

Todos os arquivos `.jsx` estão conectados com arquivos `.css`, cada arquivo tem sua estilização diferente, com o intuito de criar algo mais bonito e limpo ao mesmo tempo. Criamos variáveis com as cores principais para facilitar na hora de codificar, como mostra na *Figura 2*.

```
:root{
  --text-color: #000;
  --text-color-input: #c3c3c3;
  --color-bg: rgb(236, 236, 236);
  --color-card: #fff;
  --color-soft: #aaaaaa;
  --color-forms: #f2f2f2;
  --color-boder: lightgray;
  --color-red: red;
  --border-sala: #000;
  --text-color-input: #949494;
}
.darkmood{
  --text-color: white;
  --color-bg: black;
  --color-card: #111111;
  --border-sala: #fff;
  --text-color-input: #434343;
  --color-forms: #222222;
  --text-color-input: #676767;
}
```

Figura 2. CSS

Requisitos e Viabilidades

RF	Requisito	Prioridade
RF01	O usuário deve ser matriculado na instituição	Alta
RF02	O usuário deve realizar o preenchimento dos dados obrigatórios	Alta
RF03	Permitir o contato com departamentos da instituição	Médio
RF04	Permitir que os alunos se comuniquem pelo fórum	Alta
RF05	Permitir que alunos se cadastrem como voluntários de seu curso	Médio
RF06	Mostrar os eventos	Baixo
RF07	Auxiliar o aluno sobre os pontos de interesse, mapeamento de salas	Baixo
RF08	Auxiliar com dúvidas de calouros e usuários já estudam	Baixo

RNF	Requisito
RNF01	O sistema deve permitir o usuário alterar sua senha
RNF02	O sistema deve ser de fácil usabilidade e objetivo
RNF03	O sistema deve ser íntegro

Esses requisitos, embora não estabeleçam os serviços a serem disponibilizados pela ferramenta, orientaram na criação da arquitetura e tecnologias adquiridas durante o desenvolvimento do sistema.

Assim, o desenvolvimento foi realizado para agilizar o processo de arquitetura deixando a aplicação limpa e robusta, com as seguintes tecnologias:

- a lógica para a criação do BackEnd foi codificada na linguagem Node Js e JavaScript,
- o conteúdo foi armazenado em um banco de dados MySQL,
- apresentação do projeto foi arquitetada e desenvolvida pela linguagem ReactJs

O sistema foi desenvolvido inicialmente com ideias de melhorias envolvendo alunos da própria instituição, em seguida foi utilizado o modelo de prototipação, assim juntamos todas as ideias e criamos os *requisitos funcionais*. A partir da criação dos requisitos, podemos dar o primeiro passo para a criação do sistema e assim podendo dar enfim a estrutura da ferramenta.

Requisitos e Relacionamentos - BD

1. **wiki_fam_new_posts:**
 - Armazena posts com conteúdo, data de criação e ID do usuário autor (pode ser um funcionário ou aluno).
 - Relacionamento: Cada post é criado por um usuário (funcionário ou aluno).
2. **wiki_fam_new_comments:**
 - Armazena comentários vinculados a posts, com texto, data de criação e ID do usuário autor.
 - Relacionamentos: Cada comentário pertence a um post (ligação com [wiki_fam_new_posts](#)).
 - Cada comentário é feito por um usuário (funcionário ou aluno).
3. **wiki_fam_new_funcionarios:**
 - Armazena dados dos funcionários, incluindo email, senha e RA (registro acadêmico).
 - Relacionamento: Relacionado com [wiki_fam_new_posts](#) e [wiki_fam_new_comments](#) (pode ser autor de posts e comentários).
4. **wiki_fam_new_professores:**
 - Armazena dados dos professores, similares aos funcionários, mas sem acesso a atendimentos.
 - Relacionamento: Independente, mas alinhado aos demais funcionários.
5. **wiki_fam_new_alunos:**
 - Armazena dados dos alunos, incluindo nome, email institucional, senha e RA.
 - Relacionamentos:
 - Cada aluno pode criar posts e comentários.
 - Pode fazer sugestões e ser atendido em [wiki_fam_new_sugestoes](#) e [wiki_fam_new_atendimentos](#).
6. **wiki_fam_new_sugestoes:**
 - Registra sugestões feitas por alunos.
 - Relacionamento: Cada sugestão é associada a um aluno.
7. **wiki_fam_new_atendimentos:**
 - Armazena atendimentos feitos para alunos, com ID do aluno, curso, data, descrição, paciente e link de reunião.
 - Relacionamentos: Cada atendimento é relacionado a um aluno ([wiki_fam_new_alunos](#)), curso ([wiki_fam_new_curso](#)) e paciente ([wiki_fam_new_pacientes](#)).
8. **wiki_fam_new_curso:**
 - Registra informações sobre cursos.
 - Relacionamento: Associado a [wiki_fam_new_atendimentos](#) (cada atendimento pode ser vinculado a um curso específico).
9. **wiki_fam_new_referencias_sala e wiki_fam_new_dicas_sala:**
 - Armazenam referências e dicas associadas às salas de aula.
 - Relacionamento: Ligadas às [wiki_fam_new_salas_de_aula](#) (cada referência e dica pertence a uma sala).
10. **wiki_fam_new_blocos:**
 - Representa blocos nos quais as salas de aula estão localizadas.
 - Relacionamento: Ligado a [wiki_fam_new_salas_de_aula](#) (cada sala pertence a um bloco).

11. wiki_fam_new_salas_de_aula:

- Armazena dados das salas de aula, como nome e bloco.
- Relacionamento: Associada a [wiki_fam_new_blocos](#), [wiki_fam_new_referencias_sala](#) e [wiki_fam_new_dicas_sala](#).

12. wiki_fam_new_todo_list:

- Gerencia uma lista de tarefas com título, descrição, data de vencimento, status de conclusão e data de criação.
- Relacionamento: Não possui ligação direta com outras tabelas, podendo servir como funcionalidades independentes.

13. wiki_fam_new_pacientes:

- Armazena dados dos pacientes, com nome, telefone, email e data de criação do registro.
- Relacionamento: Associado a [wiki_fam_new_atendimentos](#) (paciente é vinculado a um atendimento específico).

14. wiki_fam_new_departamentos:

- Contém informações dos departamentos, como nome, área, contato, horário de atendimento e descrição.
- Relacionamento: Pode ser usado para categorizar ou identificar funcionários específicos, se houver uma expansão futura.

Aplicação do Scrum

A metodologia Scrum foi aplicada para organizar e agilizar o desenvolvimento do WikiFam:

Product Owner (PO): Definiu os requisitos e funcionalidades necessárias, Priorizou o backlog com base no valor e nas necessidades dos usuários e Manteve comunicação constante com os stakeholders.

Scrum Master: Garantiu o seguimento dos princípios e práticas do Scrum, Removeu impedimentos que poderiam atrasar o progresso e Facilitou reuniões diárias , revisões de sprint e retrospectivas de sprint

Equipe de Desenvolvimento: Implementou as funcionalidades definidas no backlog, Dividiu o trabalho em sprints de curta duração e entregou incrementos funcionais ao final de cada sprint.

Benefícios da Aplicação de Scrum: Flexibilidade e Adaptação: Permite ajustes rápidos a mudanças de requisitos ou prioridades.

Transparência e Comunicação: Melhora a comunicação entre todos os envolvidos.

Entrega Contínua de Valor: Funcionalidades utilizáveis são entregues ao final de cada sprint.

Com a implementação do Scrum, o desenvolvimento do WikiFam foi mais organizado e eficiente, resultando em um produto final que atende melhor às expectativas e necessidades dos alunos, professores e administradores da instituição.

Análise de Riscos do Projeto

Durante o planejamento e desenvolvimento do projeto WikiFam, foram identificados alguns riscos potenciais que poderiam comprometer a entrega ou a qualidade final do sistema. A seguir, apresenta-se uma análise desses riscos, com sua probabilidade, impacto e o plano de mitigação correspondente:

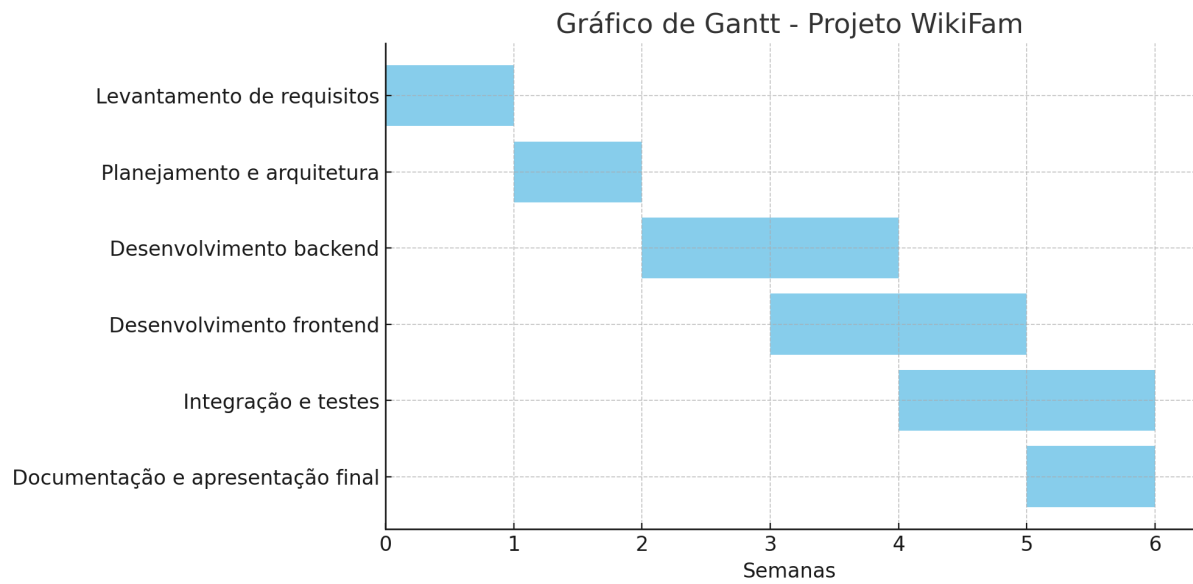
Risco Identificado	Probabilidade	Impacto	Plano de Mitigação
Atraso na entrega das funcionalidades	Média	Alto	Utilização rigorosa da metodologia Scrum com sprints bem definidos e reuniões frequentes.
Falta de conhecimento técnico específico	Alta	Médio	Incentivar pesquisas, divisão de tarefas conforme habilidades e ajuda mútua entre membros.
Falha de comunicação entre os membros	Média	Alto	Uso de ferramentas como Discord, Trello e reuniões presenciais para alinhamento constante.
Erros de integração entre frontend e backend	Média	Alto	Realização de testes contínuos com Postman e revisão de código em equipe.
Problemas com deploy e hospedagem	Baixa	Médio	Simulações em ambiente local com XAMPP e controle de versões via GitHub.

Cronograma



Figura 3. Cronograma

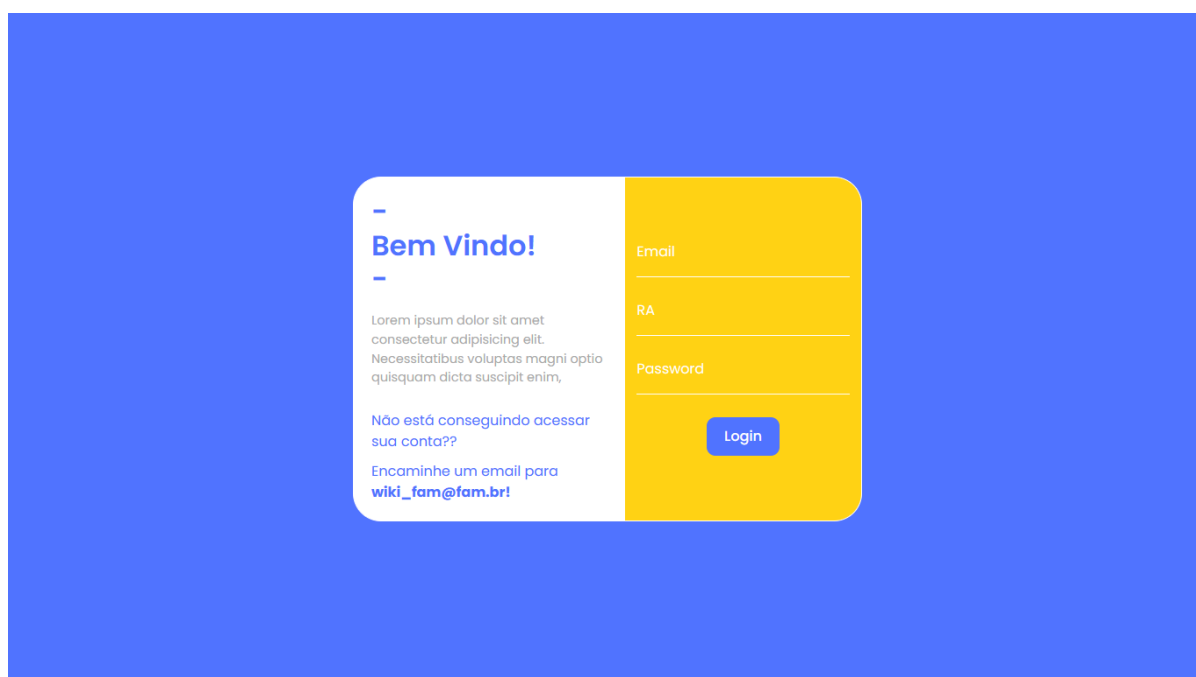
Gráfico de Gantt



Protótipo de tela com aplicação dos requisitos

Tela de Login

A tela de login servirá para realizar a autenticação de que o usuário é um aluno, funcionário ou professor da instituição, ele terá que informar seu e-mail, RA\RE e senha para conseguir acessar o site.



The login screen features a solid blue background. Centered on the screen is a white card with rounded corners and a yellow right half. On the white side, the text reads: "Bem Vindo!" followed by placeholder text "Lorem ipsum dolor sit amet consectetur adipiscing elit. Necessitatibus voluptas magni optio quisquam dicta suscipit enim,". Below this, it says "Não está conseguindo acessar sua conta??" and "Encaminhe um email para wiki_fam@fam.br!". The yellow side contains three input fields labeled "Email", "RA", and "Password", with a blue "Login" button at the bottom.

Figura 4. Tela de Login

HomePage

O homepage seria o centralizador ou menu principal do site, exibindo o feed e dando acessos às outras funcionalidades do sistema.

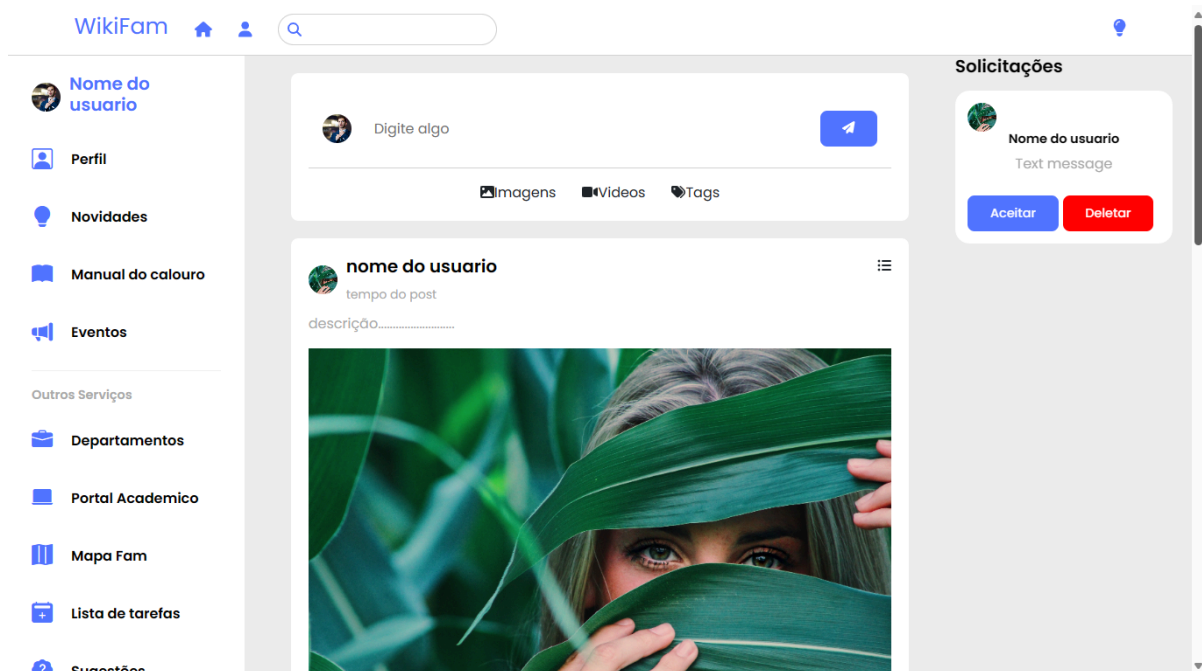


Figura 5. Homepage

Perfil

A Tela de Perfil será onde o usuário irá editar suas informações e também realizar publicações no feed.

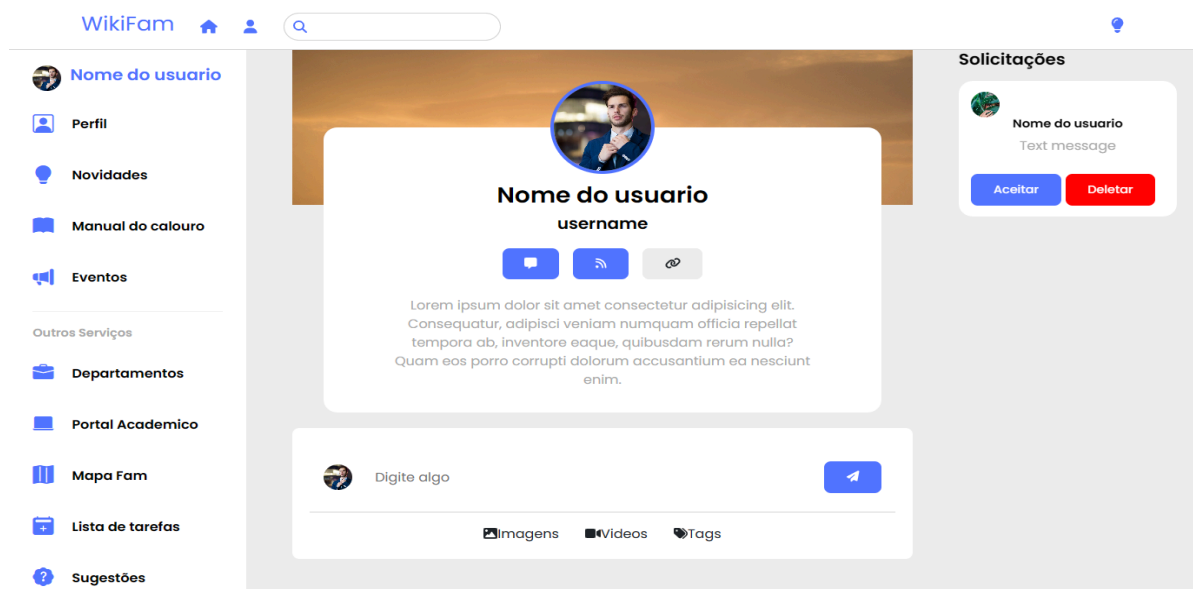


Figura 6. Perfil de Usuário

Novidades

A tela de novidades será onde irá ser divulgado melhorias e alterações na instituição.

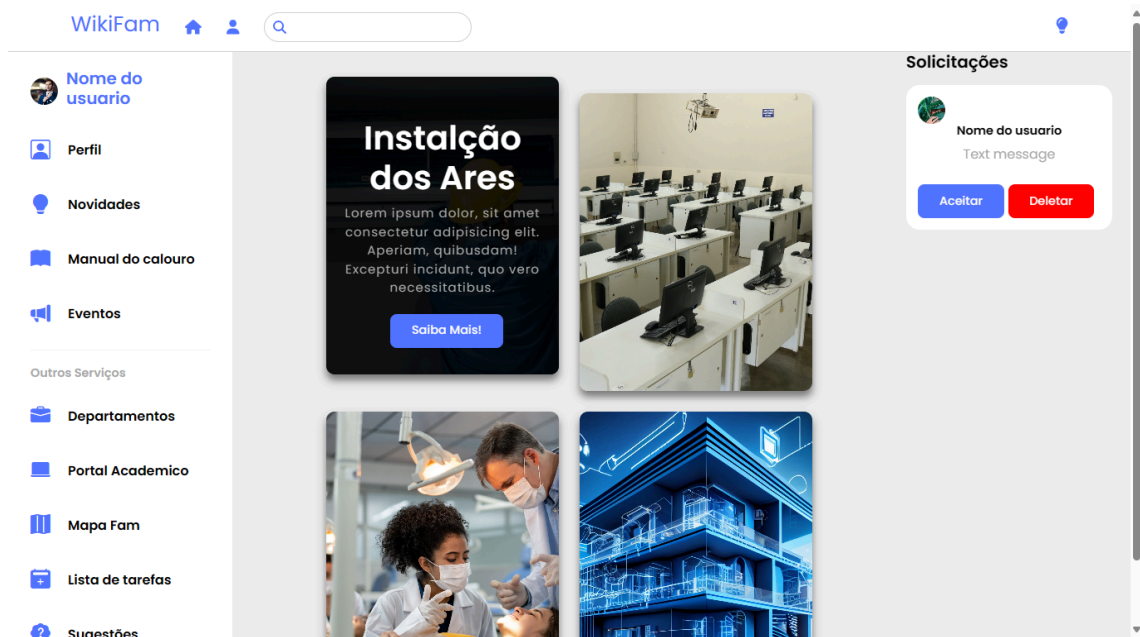


Figura 7. Mural de novidades

Manual do Calouro

O manual do calouro servirá para auxiliar e suprir eventuais dúvidas dos alunos.

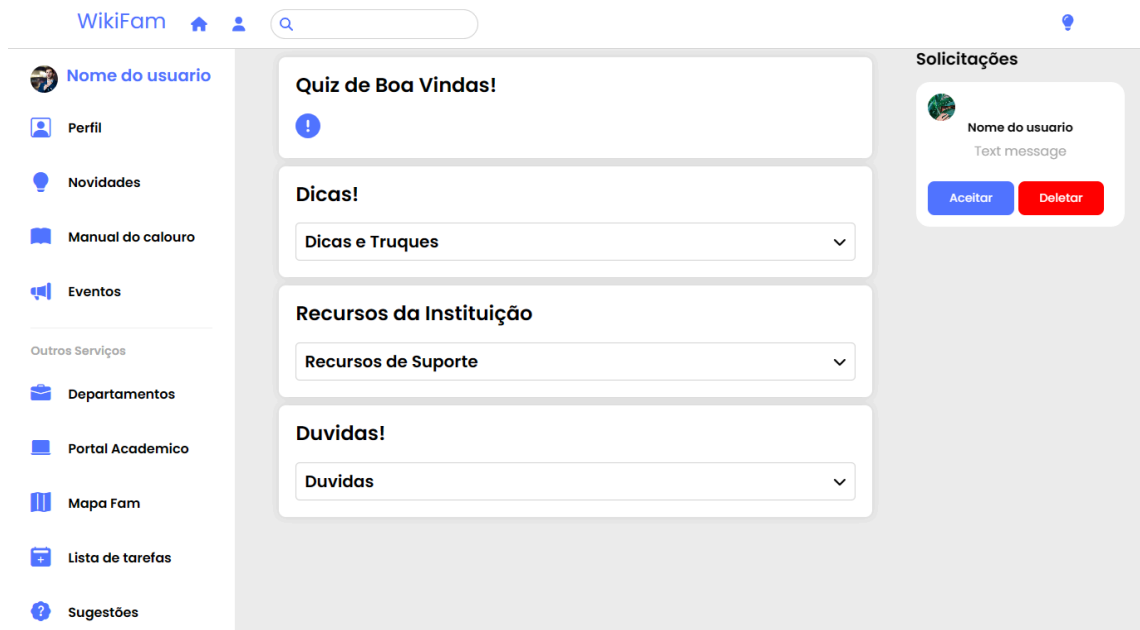


Figura 8. Manual dos calouros

Quiz de boas vindas

O Quiz de boas vindas é uma maneira de melhorar a interação com os alunos.

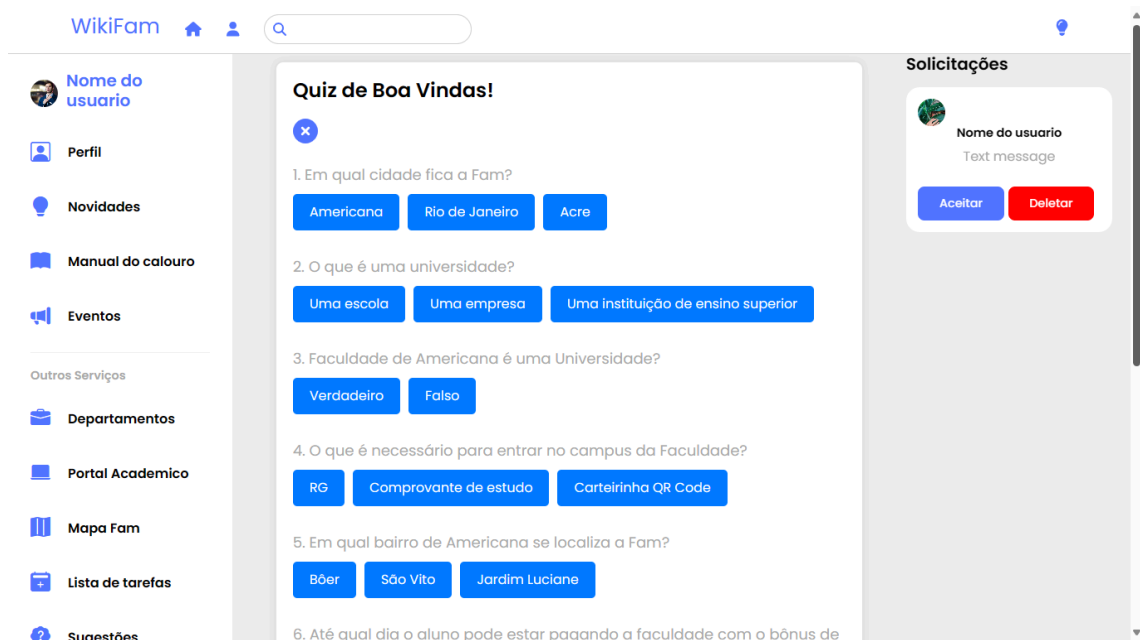


Figura 9. Quiz de boas vindas

Dicas e truques

A aba de dicas servirá como um tópico com dicas importantes para quem está iniciando seu ciclo acadêmico

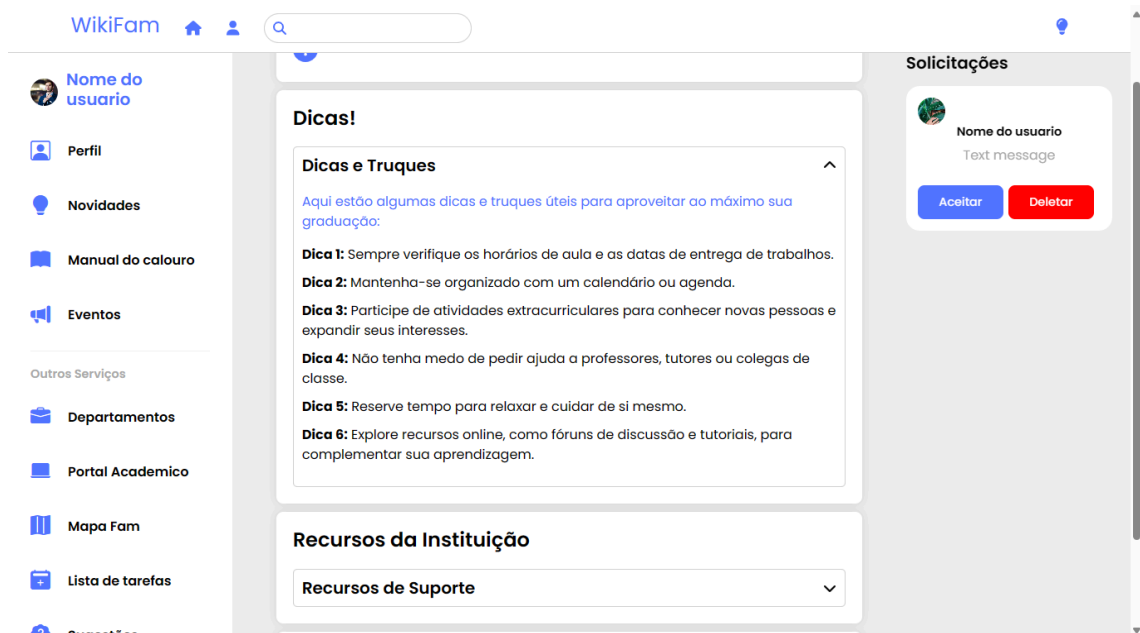


Figura 10. Dicas e truques

Dúvidas

A Aba Dúvidas tem como objetivo auxiliar o aluno em coisas do dia a dia do aluno, como entrar no campus, conseguir descontos na de adiantamento no boleto financeiro e etc.

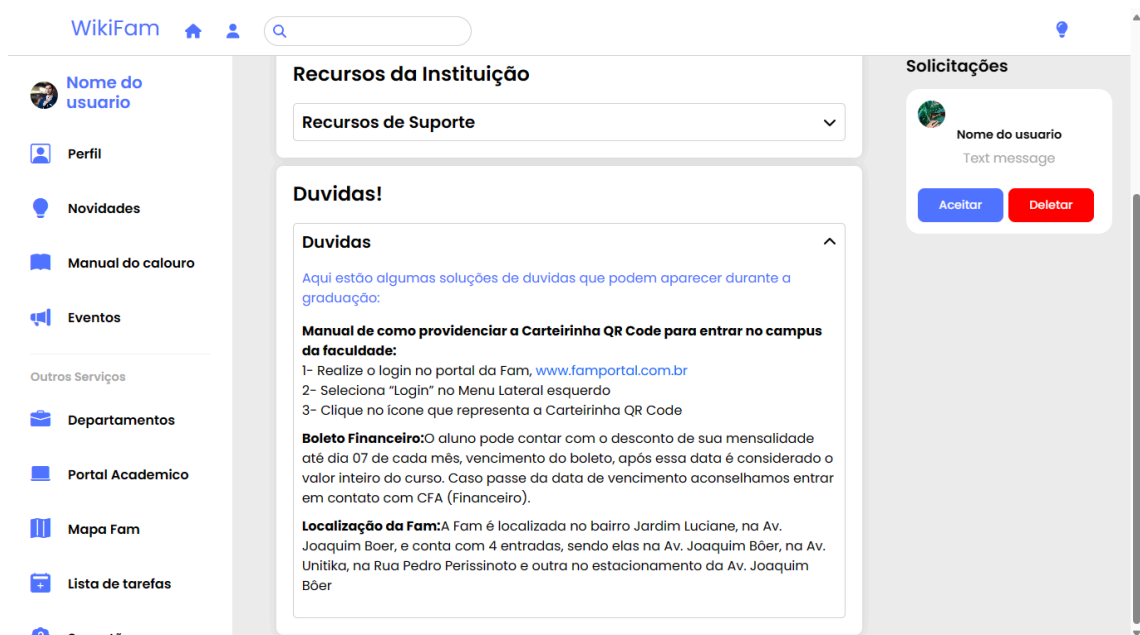


Figura 11. Dúvidas

Eventos

Na tela de eventos será divulgado apresentações, campanhas e demais atividades que serão realizadas na instituição.

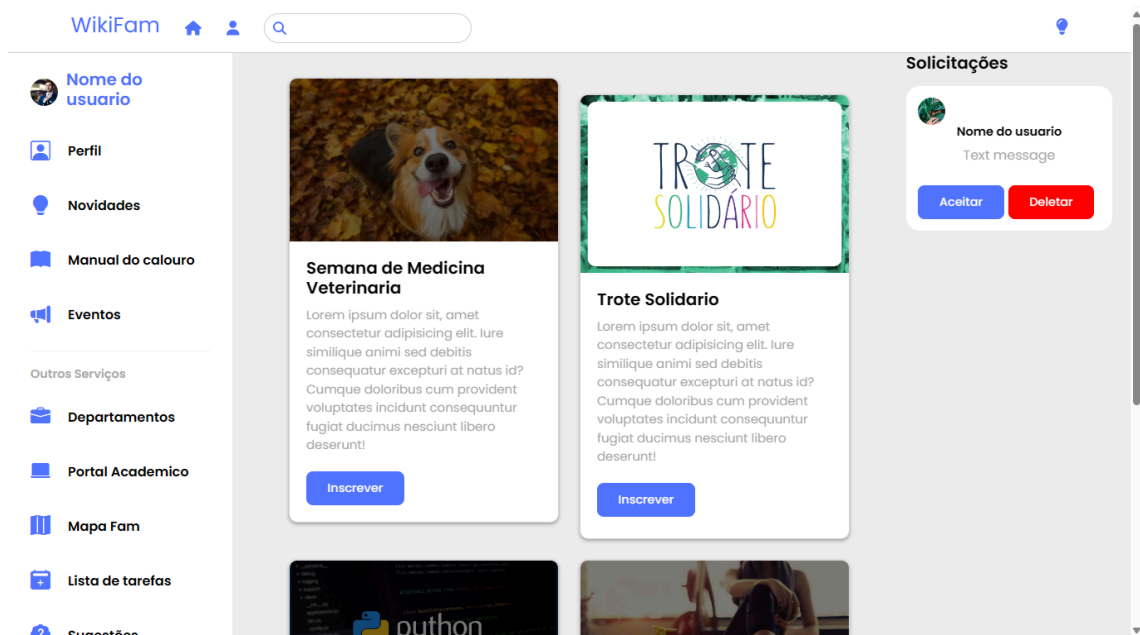
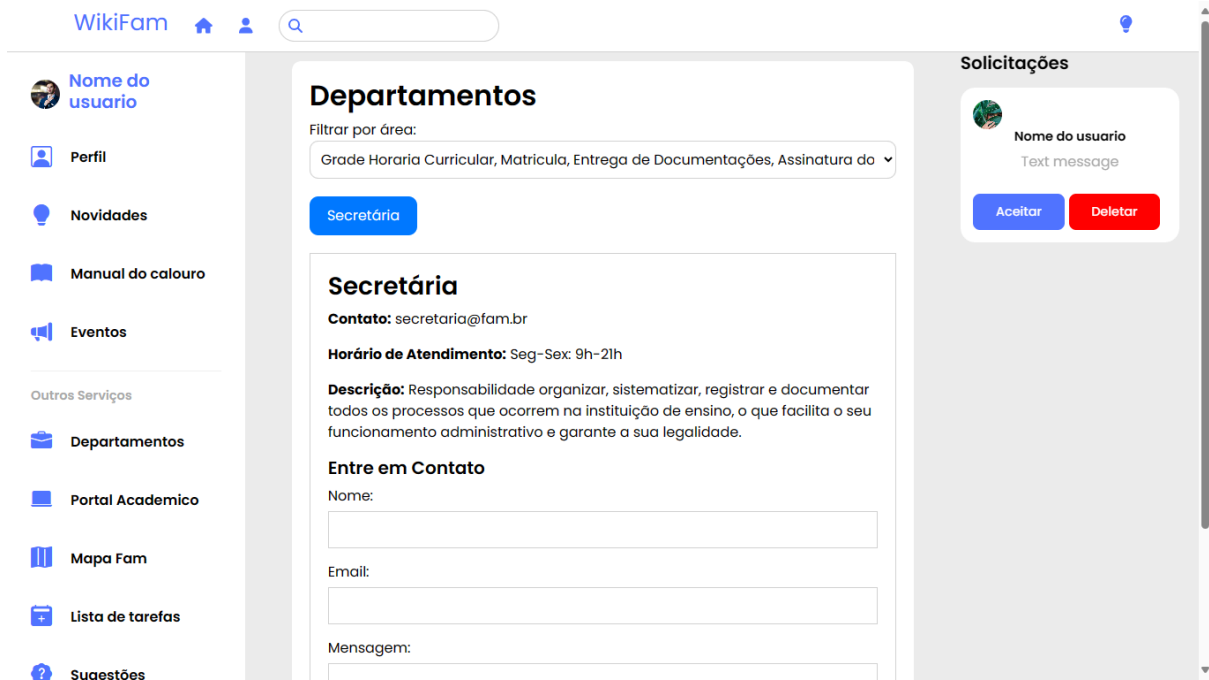


Figura 12. Mural de eventos

Departamentos

Terá a função de direcionar o aluno aos canais de comunicação da instituição.



The screenshot displays the 'WikiFam' web application interface. The top navigation bar includes the 'WikiFam' logo, a home icon, a user profile icon, and a search bar. The left sidebar contains a user profile section with the name 'Nome do usuário' and a list of navigation links: 'Perfil', 'Novidades', 'Manual do calouro', 'Eventos', and 'Outros Serviços'. Under 'Outros Serviços', there are links for 'Departamentos', 'Portal Acadêmico', 'Mapa Fam', 'Lista de tarefas', and 'Suaestões'. The main content area is titled 'Departamentos' and includes a filter dropdown menu labeled 'Filtrar por área:' with the selected option 'Grade Horária Curricular, Matricula, Entrega de Documentações, Assinatura do'. Below the filter is a blue button labeled 'Secretária'. The 'Secretária' section provides contact information: 'Contato: secretaria@fam.br', 'Horário de Atendimento: Seg-Sex: 9h-21h', and a 'Descrição' of the department's responsibilities. It also includes a contact form with fields for 'Nome:', 'Email:', and 'Mensagem:'. The right sidebar, titled 'Solicitações', shows a user profile and a 'Text message' input field, with 'Aceitar' and 'Deletar' buttons.

Figura 13. Lista de departamentos

Mapa FAM

O mapa servirá para localizar a sala do aluno, lhe informando a direção e um ponto de referência para seguir. Separamos cada bloco com a cor correspondente às que a faculdade disponibiliza no mural.

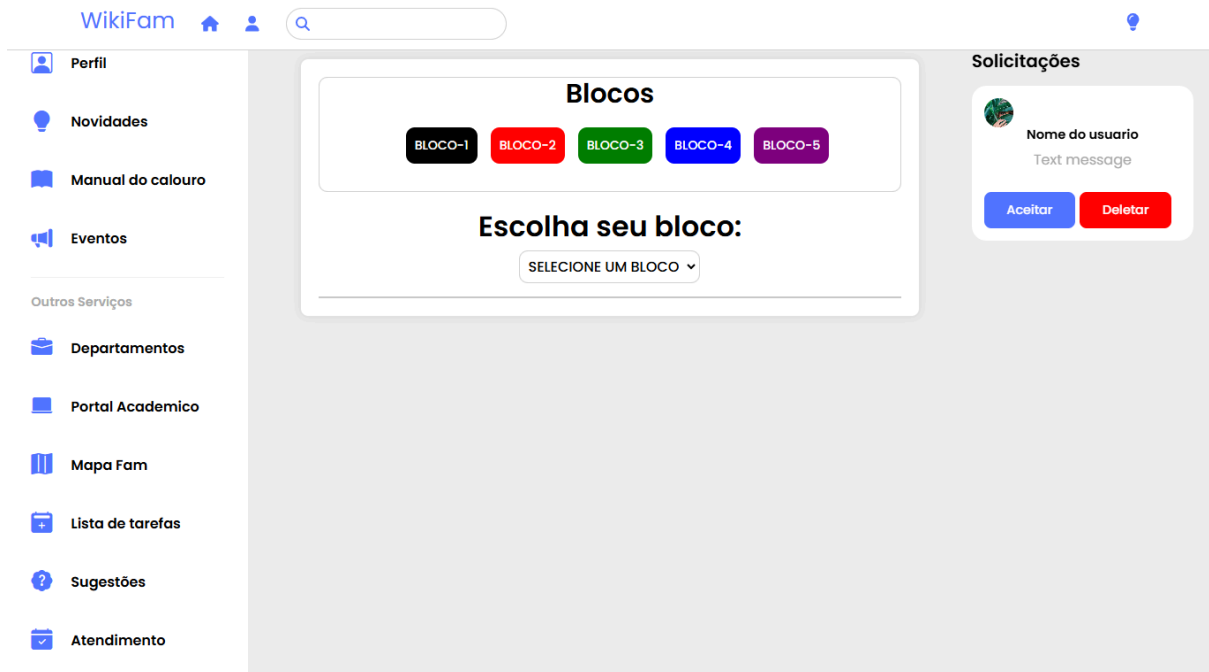


Figura 14. Mapa Fam



Figura 15. Dica de mapa

Lista de Tarefas

Terá a funcionalidade de marcar alguma tarefa ou atividade do aluno.

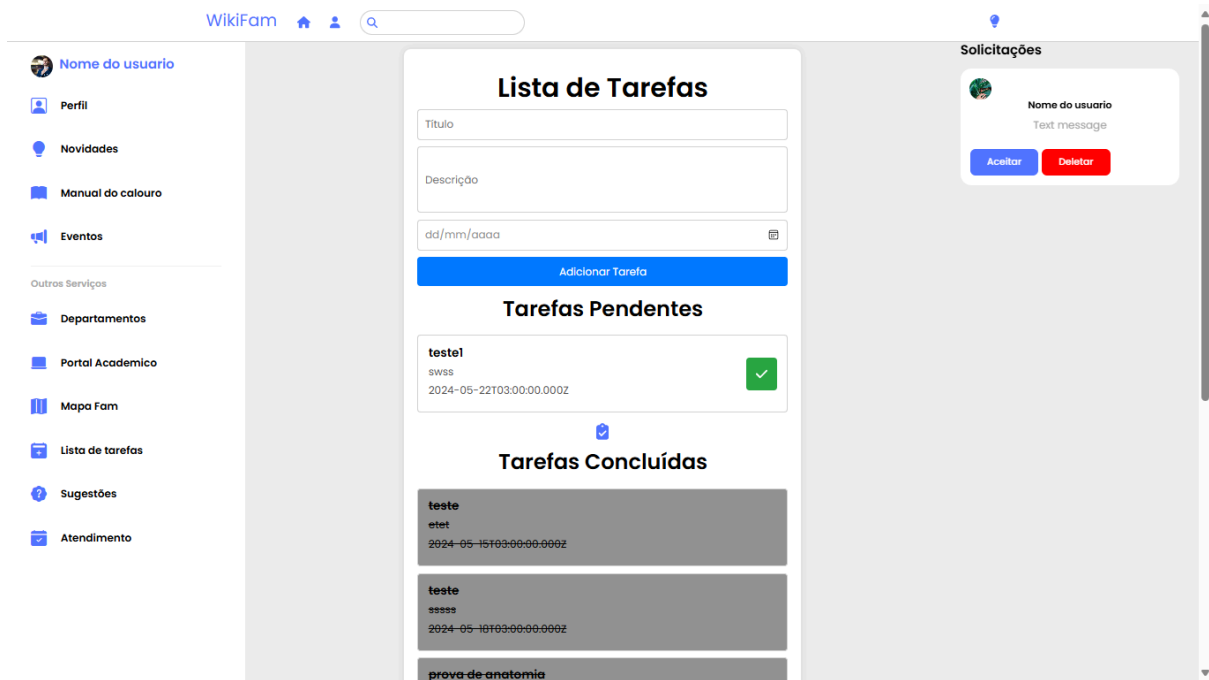


Figura 16. Lista de Tarefas

Sugestões

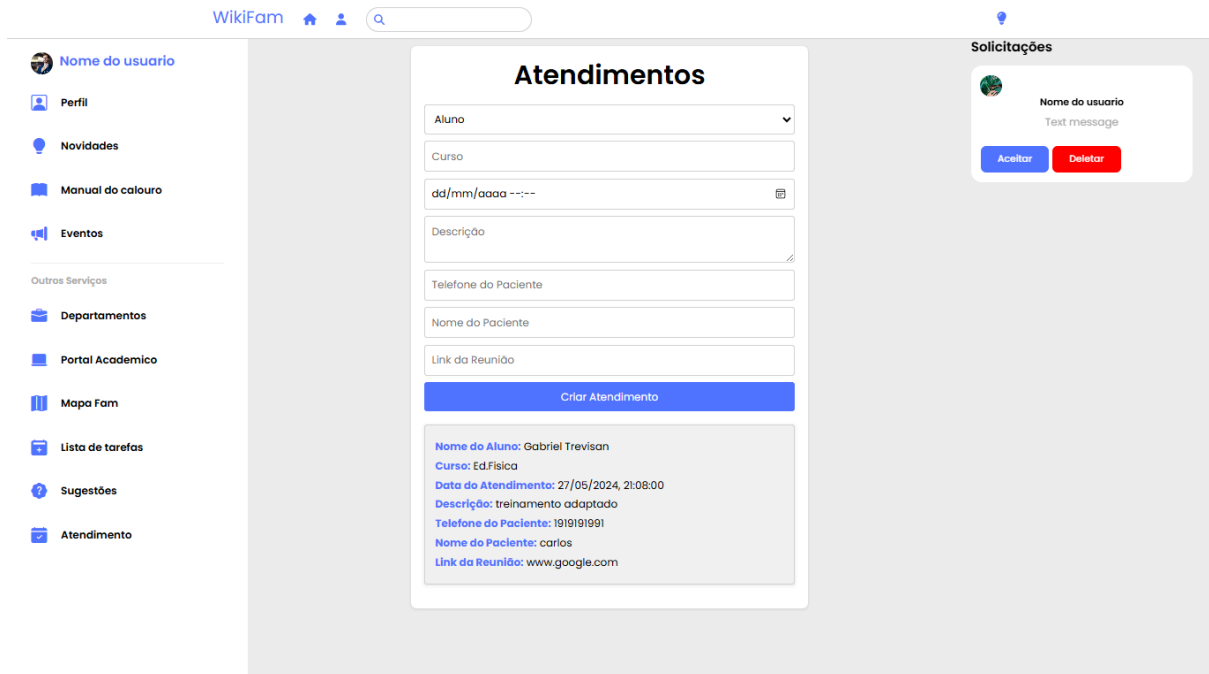
Será uma forma do aluno dar sugestões de melhoria para a instituição.



Figura 17. Sugestões

Atendimentos

Tem a função de marcar atendimentos entre alunos e cursistas de outras matérias, por exemplo, o aluno poderá marcar uma consulta com um cursista de nutrição.



The screenshot displays the 'Atendimentos' (Appointments) section of the WikiFam application. The interface is divided into three main areas: a left sidebar, a central content area, and a right sidebar.

Left Sidebar: Contains navigation links for 'Nome do usuário', 'Perfil', 'Novidades', 'Manual do calouro', 'Eventos', and 'Outros Serviços'. Under 'Outros Serviços', there are links for 'Departamentos', 'Portal Acadêmico', 'Mapa Fam', 'Lista de tarefas', 'Sugestões', and 'Atendimento'.

Central Content Area: Features a form titled 'Atendimentos' with the following fields: 'Aluno' (dropdown), 'Curso' (text), 'dd/mm/aaaa --:--' (date and time), 'Descrição' (text area), 'Telefone do Paciente' (text), 'Nome do Paciente' (text), and 'Link da Reunião' (text). A blue button labeled 'Criar Atendimento' is positioned below the form. A summary box at the bottom of the form displays the following information: 'Nome do Aluno: Gabriel Trevisan', 'Curso: Ed.Física', 'Data do Atendimento: 27/05/2024, 21:08:00', 'Descrição: treinamento adaptado', 'Telefone do Paciente: 1919191991', 'Nome do Paciente: carlos', and 'Link da Reunião: www.google.com'.

Right Sidebar: Contains a section titled 'Solicitações' (Requests) with a user profile 'Nome do usuário' and a 'Text message' field. Below this are two buttons: 'Aceitar' (Accept) and 'Deletar' (Delete).

Figura 18. Atendimento

Diagramas

Diagrama de Caso

A criação do diagrama de caso foi realizado pensando em como o sistema seria utilizado pelos alunos e colaboradores da instituição.

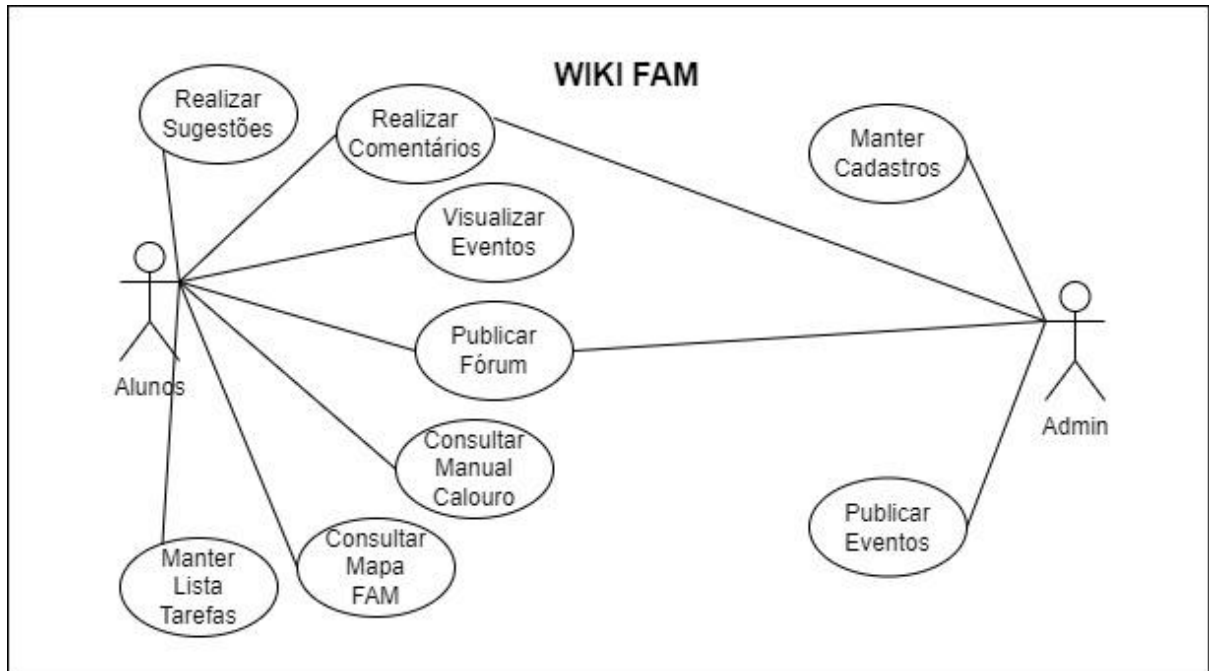


Figura 19. Diagrama de Uso

Diagrama Conceitual

Com base no diagrama de uso, decidimos criar o conceitual para termos uma base mais elaborada e definir os atributos das entidades, assim facilitar o processo de criação do banco.

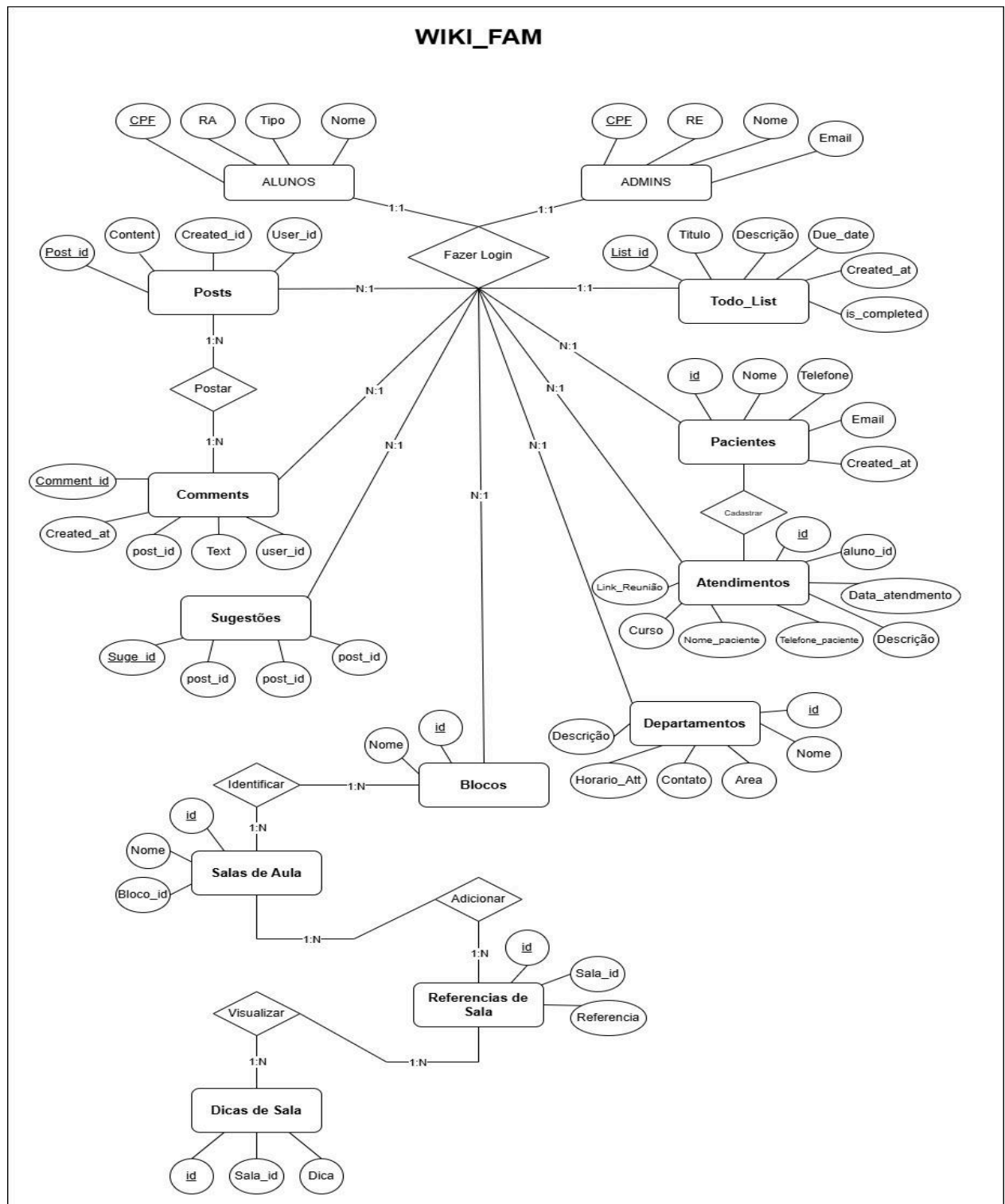


Figura 20. Diagrama Conceitual

Diagrama de Classe

Logo após a criação do diagrama conceitual decidimos criar o de classe assim poderíamos definir melhor qual tipo de dado os campos iriam receber

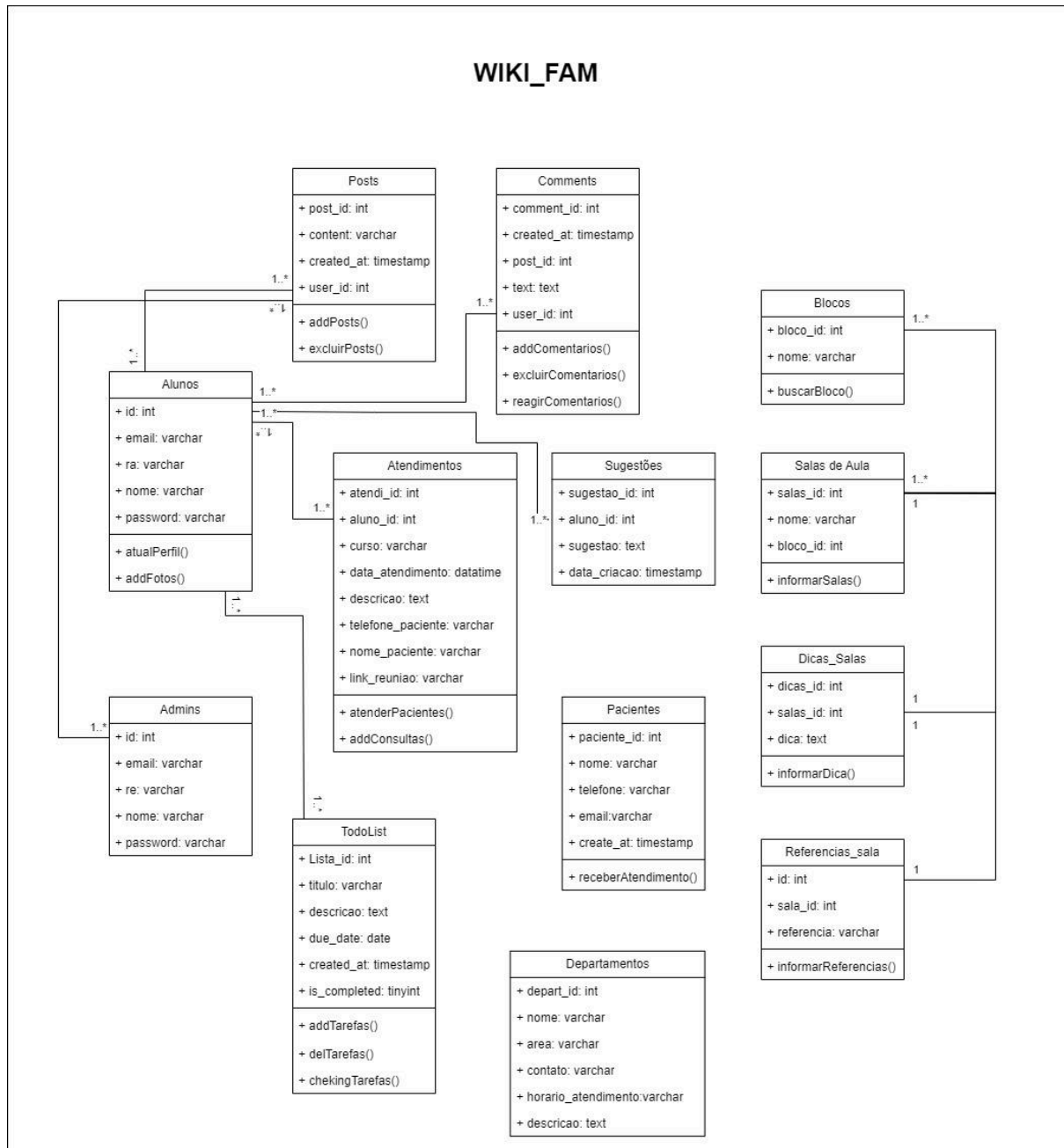
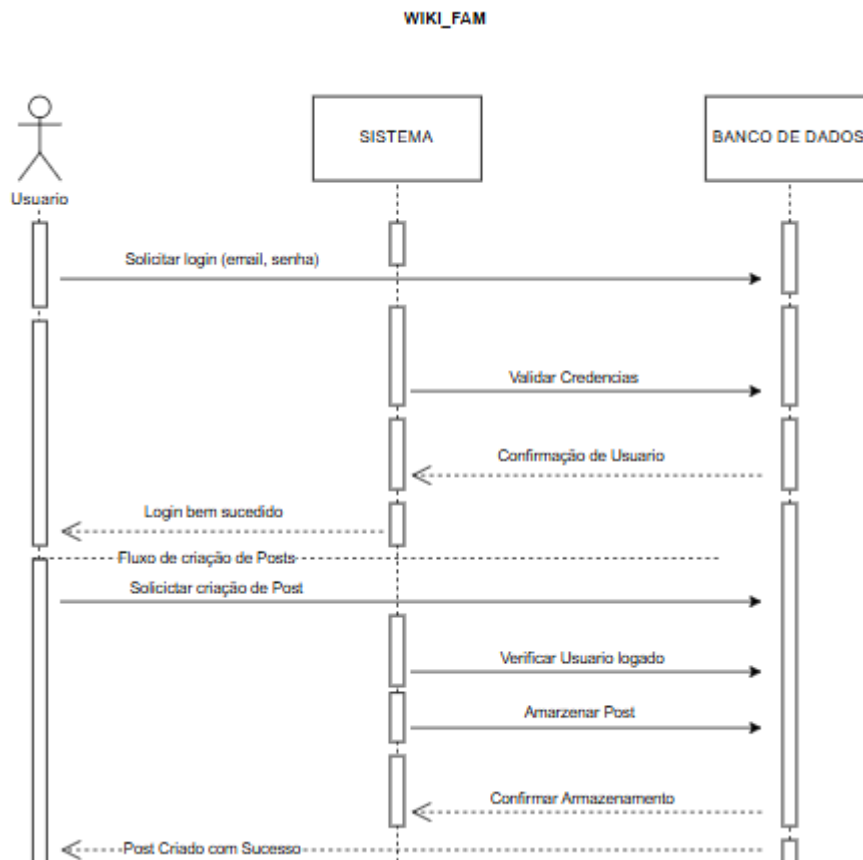


Figura 21. Diagrama de Classe

Diagrama de Sequência

O diagrama de sequência foi criado para termos uma visão mais detalhada da duração de cada ação e do que poderia ser retornado dentro do nosso sistema.



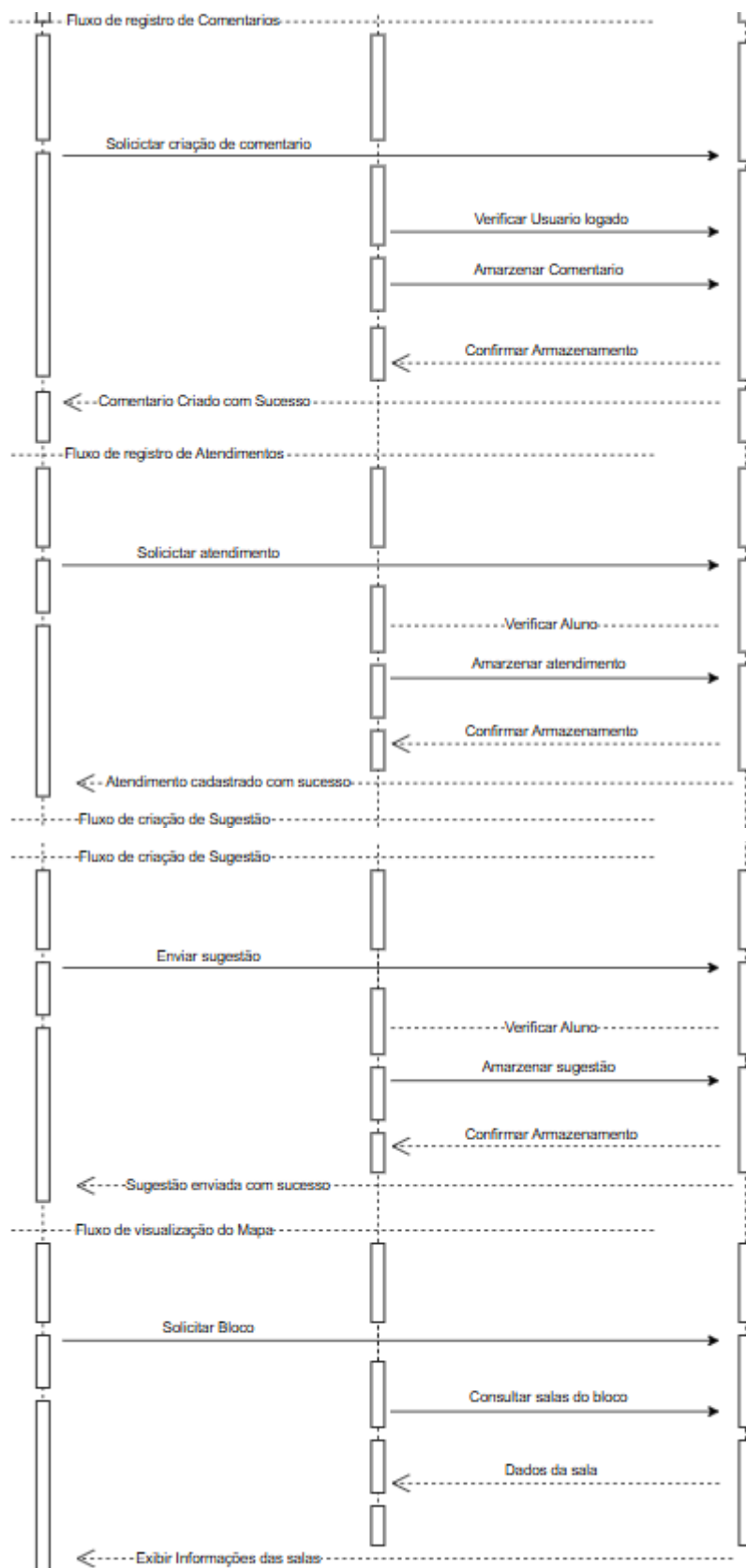


Figura 22. Diagrama de Sequência

Diagrama de Atividade

Após o diagrama de sequência decidimos entender um pouco mais como cada atividade do sistema ficará disponível para o aluno, então criamos o diagrama de atividade de forma resumida.

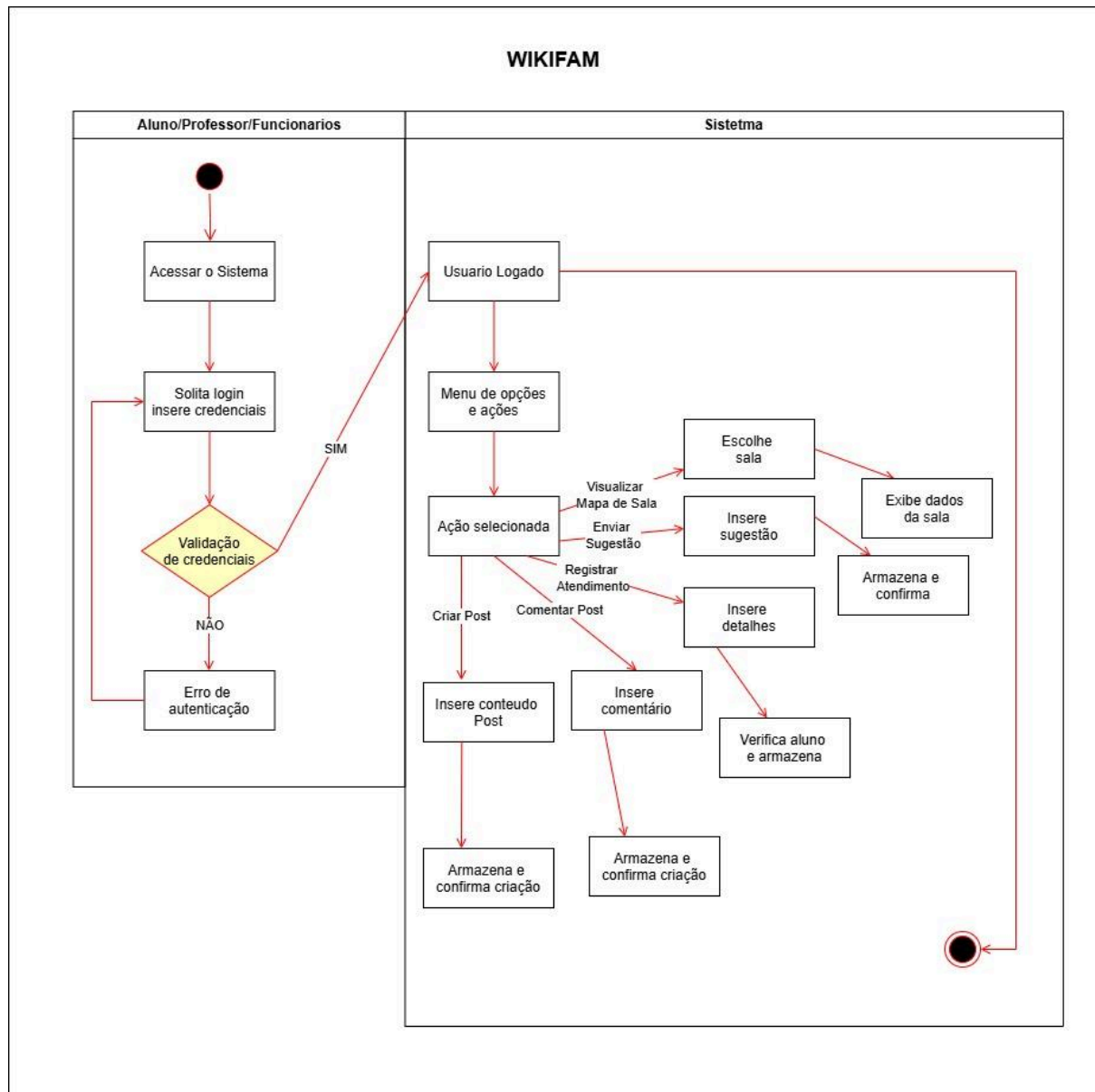


Figura 23. Diagrama de Atividade

Na etapa final do projeto, decidimos alterar alguns dados de tabelas e também modificar algumas delas. Por exemplo, a tabela 'admin' foi transformada em 'funcionários' e 'professores'. Além disso, criamos a tabela 'cursos' para facilitar o cadastro de todos os cursos que se relacionam com os alunos. Também ajustamos alguns campos, como na tabela 'atendimentos', para que ela possa se relacionar com a tabela 'pacientes'.



Tabelas e seus Campos:

1. **wiki_fam_new_posts**
 - post_id (int): Identificador único do post.
 - content (varchar(255)): Conteúdo do post.
 - created_at (timestamp): Data e hora de criação do post.
 - user_id (int): ID do usuário que criou o post (relacionado com funcionarios ou alunos).
2. **wiki_fam_new_comentarios**
 - comentarios_id (int): Identificador único do comentário.
 - created_at (timestamp): Data e hora de criação do comentário.
 - post_id (int): ID do post ao qual o comentário pertence (relacionado com posts).
 - text (text): Texto do comentário.
 - user_id (int): ID do usuário que fez o comentário (relacionado com funcionarios ou alunos).
3. **wiki_fam_new_funcionarios**
 - funcio_id (int): Identificador único do funcionário.
 - email (varchar(255)): Email do funcionário.
 - password (varchar(255)): Senha do funcionário.
 - ra (varchar(255)): Registro acadêmico do funcionário.
4. **wiki_fam_new_professores**
 - profs_id (int): Identificador único do professor.
 - email (varchar(255)): Email do professor.
 - password (varchar(255)): Senha do professor.
 - ra (varchar(255)): Registro acadêmico do professor.
5. **wiki_fam_new_alunos**
 - alunos_id (int): Identificador único do aluno.
 - email (varchar(255)): Email institucional do aluno.
 - password (varchar(255)): Senha do aluno.
 - ra (varchar(255)): Registro acadêmico do aluno.
 - nome (varchar(255)): Nome do aluno.
6. **wiki_fam_new_sugestoes**
 - sugestoes_id (int): Identificador único da sugestão.
 - aluno_id (int): ID do aluno que fez a sugestão (relacionado com alunos).
 - sugestao (text): Texto da sugestão.
 - data_criacao (timestamp): Data e hora de criação da sugestão.
7. **wiki_fam_new_atendimentos**
 - atendi_id (int): Identificador único do atendimento.
 - aluno_id (int): ID do aluno atendido (relacionado com alunos).
 - curso_id (int): ID do curso (relacionado com curso).
 - data_atendimento (datetime): Data e hora do atendimento.
 - descricao (text): Descrição do atendimento.
 - paciente_id (int): ID do paciente atendido (relacionado com pacientes).
 - link_reuniao (varchar(255)): Link da reunião.

8. **wiki_fam_new_curso**
 - curso_id (int): Identificador único do curso.
 - nome (varchar(255)): Nome do curso.
9. **wiki_fam_new_referencias_sala**
 - referencias_id (int): Identificador único da referência.
 - sala_id (int): ID da sala relacionada (relacionado com salas_de_aula).
 - referencia (varchar(255)): Texto da referência.
10. **wiki_fam_new_dicas_sala**
 - dicas_id (int): Identificador único da dica.
 - sala_id (int): ID da sala relacionada (relacionado com salas_de_aula).
 - dica (text): Texto da dica.
11. **wiki_fam_new_blocos**
 - blocos_id (int): Identificador único do bloco.
 - nome (varchar(50)): Nome do bloco.
12. **wiki_fam_new_salas_de_aula**
 - salas_id (int): Identificador único da sala de aula.
 - nome (varchar(50)): Nome da sala de aula.
 - bloco_id (int): ID do bloco relacionado (relacionado com blocos).
13. **wiki_fam_new_todo_list**
 - lista_id (int): Identificador único do item de lista de tarefas.
 - titulo (varchar(255)): Título do item.
 - descricao (text): Descrição do item.
 - due_date (date): Data de vencimento.
 - created_at (timestamp): Data e hora de criação do item.
 - tf_completada (tinyint): Indica se o item está completo.
14. **wiki_fam_new_pacientes**
 - pacientes_id (int): Identificador único do paciente.
 - nome (varchar(100)): Nome do paciente.
 - telefone (varchar(15)): Telefone do paciente.
 - email (varchar(100)): Email do paciente.
 - created_at (timestamp): Data e hora de criação do registro.
15. **wiki_fam_new_departamentos**
 - depart_id (int): Identificador único do departamento.
 - nome (varchar(100)): Nome do departamento.
 - area (varchar(150)): Área do departamento.
 - contato (varchar(20)): Contato do departamento.
 - horario_de_atendimento (varchar(100)): Horário de atendimento do departamento.
 - descricao (text): Descrição do departamento.

Relacionamentos

- **Posts e Comentários:** Um post pode ter vários comentários, cada comentário pertence a um post.
- **Posts e Usuários (Funcionários/Alunos):** Um post ou comentário pode ser feito por um funcionário ou aluno.
- **Sugestões e Alunos:** Um aluno pode fazer várias sugestões, cada sugestão pertence a um aluno.
- **Atendimentos e Alunos/Pacientes:** Cada atendimento é relacionado a um aluno e a um paciente, ambos podendo ter múltiplos atendimentos.
- **Atendimentos e Curso:** Cada atendimento está associado a um curso específico.
- **Referências/Dicas de Sala e Salas de Aula:** Cada sala pode ter múltiplas referências e dicas.
- **Salas de Aula e Blocos:** Um bloco contém várias salas de aula, cada sala pertence a um bloco.

Conclusão

O projeto WikiFam demonstra um compromisso significativo com a inovação e a melhoria do ambiente acadêmico da Faculdade de Americana. Por meio do desenvolvimento de uma plataforma multifuncional, que integra ferramentas avançadas para interação, gestão acadêmica e uma rede social exclusiva, buscamos transformar a educação e facilitar a comunicação entre alunos e professores.

A aplicação é estruturada de maneira robusta, utilizando uma combinação eficaz de Node.js para o backend, MySQL para o gerenciamento de dados e React.js para o frontend. Esta arquitetura garante uma comunicação harmoniosa entre as diversas partes do sistema, proporcionando uma experiência de usuário fluida e eficiente. A detalhada organização das pastas e arquivos, junto com a estilização aprimorada, assegura que o sistema seja não apenas funcional, mas também esteticamente agradável e fácil de navegar.

A metodologia Scrum desempenhou um papel crucial no desenvolvimento do WikiFam, garantindo flexibilidade, transparência e uma entrega contínua de valor. A divisão clara de responsabilidades entre o Product Owner, Scrum Master e a equipe de desenvolvimento facilitou um progresso ágil e alinhado com as necessidades dos usuários. Cada sprint resultou em incrementos funcionais, permitindo um aprimoramento constante e uma resposta rápida a qualquer mudança de requisitos.

Além de melhorar a comunicação e interação entre calouros e veteranos, o WikiFam oferece uma gama de funcionalidades que atendem a diferentes necessidades, como o acompanhamento de eventos, acesso a manuais e dicas, e gestão de tarefas. Com isso, espera-se não apenas facilitar a adaptação dos novos alunos, mas também criar uma comunidade acadêmica mais integrada e solidária.

Em suma, o WikiFam representa uma evolução significativa nos sistemas acadêmicos da instituição, trazendo benefícios tangíveis para toda a comunidade acadêmica. A dedicação da equipe de desenvolvimento e a aplicação de tecnologias modernas resultaram em uma ferramenta que não só atende às expectativas, mas também estabelece um novo padrão de excelência na gestão acadêmica.

Referência

Este projeto, WikiFam, foi desenvolvido por uma equipe de alunos do 5º semestre de Ciência da Computação da Faculdade de Americana (FAM). Gostaríamos de reconhecer as contribuições de cada membro da equipe e os recursos que foram fundamentais para o sucesso deste projeto.

Equipe de Desenvolvimento:

- **João Frederico:** Responsável pela criação do banco de dados em MySQL e ajudou no desenvolvimento do backend.
- **Guilherme Ventura:** Responsável pela arquitetura do backend utilizando Node.js e integração com o banco de dados MySQL.
- **Matheus Castilho:** Desenvolvimento do frontend com React.js e criação de componentes interativos e o design e estilização utilizando CSS.
- **Henry Calle:** Responsável pela arquitetura do backend utilizando Node.js e integração com o banco de dados MySQL.
- **Pedro Franco:** Coordenação do projeto e aplicação da metodologia Scrum, atuando como Scrum Master.
- **Gabriel Trevisan:** Arquitetou e organizou toda a documentação do projeto.

Ferramentas e Tecnologias Utilizadas:

- **Node.js:** Utilizado para o desenvolvimento do backend e scripts de comunicação.
- **MySQL:** Banco de dados utilizado para armazenar e gerenciar informações.
- **React.js:** Framework utilizado para a construção do frontend e interface de usuário.
- **CSS:** Estilização e design das páginas do projeto.
- **Bootstrap:** Framework web utilizado para importação de ícones.
- **Font-Awesome:** Biblioteca de ferramentas utilizadas para importação de ícones.
- **Scrum:** Metodologia ágil aplicada para organização e gerenciamento do desenvolvimento do projeto.
- **GitHub:** Plataforma de hospedagem utilizada para armazenar os códigos.
- **Postman:** API utilizada para fazer testes da funcionalidade dos códigos.