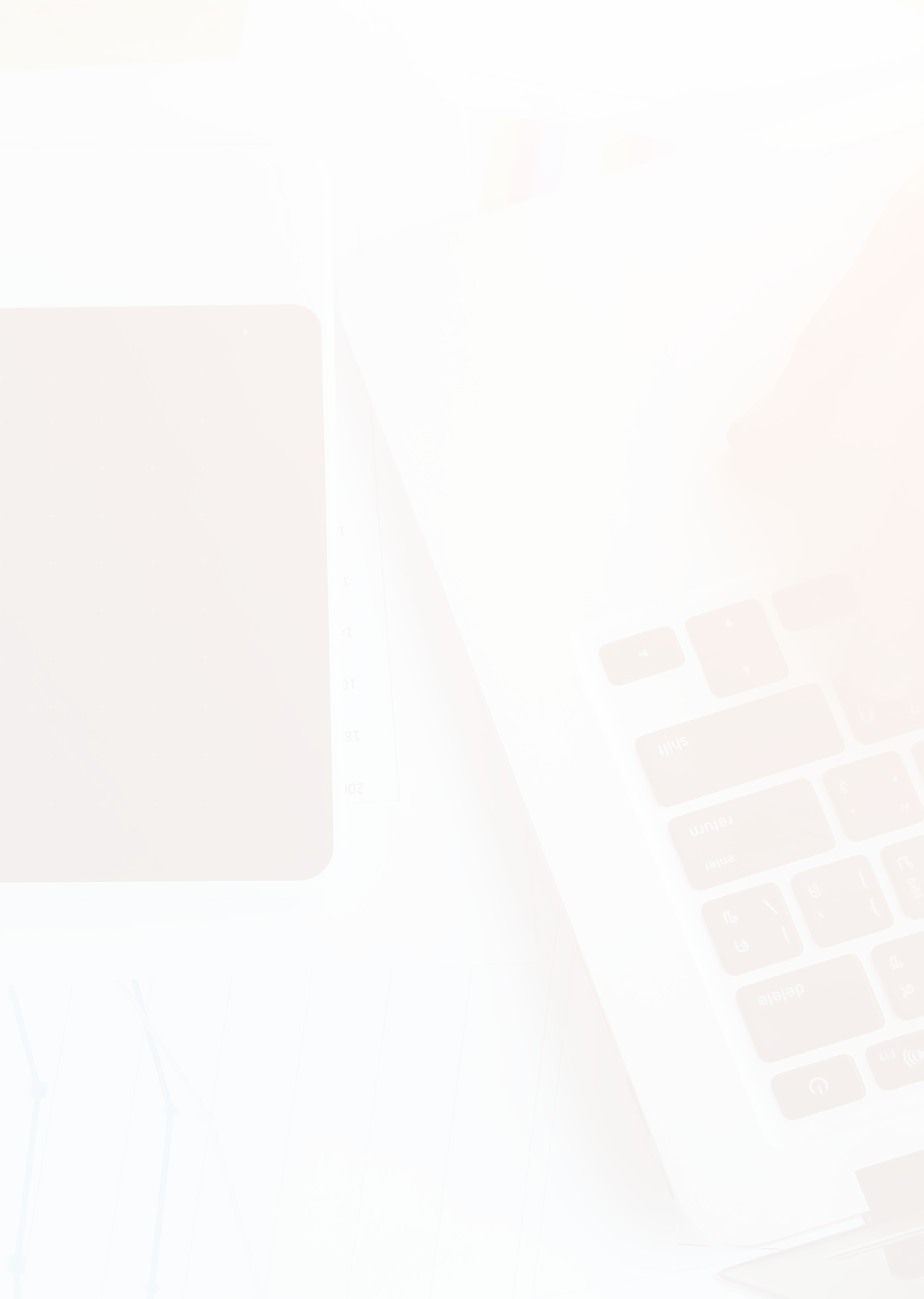
**VAGAKEY PROJETO**

**IDENTIDADE VISUAL**

Por



Vagakey

Para

FIAP

**A EMPRESA**

Facilitando a vida do motorista no dia a dia.

Um sistema de gestão para microempresas e pequenas empresas que trabalham com estacionamentos na grande São Paulo . A Vagakey é uma plataforma de software projetada para ajudar pequenos negócios a administrar suas operações de maneira eficiente. Ele pode abranger áreas como contabilidade básica, controle de vagas e sua gestão, emissão de notas fiscais, tickets de estacionamentos e acompanhamento da lotação para o cliente e usuário. Além disso, pode oferecer funcionalidades como registro de clientes, agendamento, relatórios financeiros simplificados e ferramentas para monitorar o desempenho geral do estacionamento em datas comemorativas e dias corriqueiros.

**DIVISAO DO PROJETO**

O calendário anual ajuda a empresa a manter o foco em seus objetivos, gerenciar recursos de maneira eficaz e garantir que todas as partes interessadas estejam cientes das atividades planejadas. Ele facilita a comunicação interna, o alinhamento das equipes e a tomada de decisões informadas ao longo do ano.

Usamos a ferramenta Trello para melhor visualização, planejamento e homologação de tarefas.

**Fase de Planejamento:**

Definição de Objetivos e Escopo (Semana 1): Reuniões com a equipe para entender os objetivos do aplicativo.

Definir os principais recursos e funcionalidades. Estabelecer os requisitos mínimos para o lançamento.

Pesquisa e Análise de Mercado (Semana 2): Pesquisar aplicativos similares no mercado. Identificar oportunidades e lacunas no mercado. Realizar análise de concorrência.

Design de Interface e Prototipagem (Semana 3-4): Criar esboços e wireframes da interface.

Desenvolver protótipos interativos.

Coletar feedback da equipe e fazer ajustes.

**Fase de Desenvolvimento:** Desenvolvimento Backend e API (Semana 5-8): Configurar o ambiente de desenvolvimento.

Desenvolver o backend do aplicativo.

Criar APIs necessárias para comunicação entre frontend e backend.

Desenvolvimento Frontend (Semana 9-12): Implementar a interface do usuário de acordo com os designs.

Integração com as APIs do backend. Testes de interface e interatividade.

Desenvolvimento de Recursos Específicos (Semana 13- 16):

Implementar recursos específicos, como integração de mídia social, geolocalização, etc.

Testar e otimizar esses recursos.

**Fases de Polimento e Desenvolvimento:**

Testes e Depuração (Semana 17-20):

Testar a aplicação em diferentes dispositivos e sistemas operacionais.

Identificar e corrigir bugs e problemas de desempenho.

Realizar testes de usabilidade com um grupo seleto de usuários.

Otimização e Polimento (Semana 21-24): Otimizar o desempenho da aplicação.

Ajustar a interface com base no feedback dos testes de usabilidade.

Realizar testes finais de qualidade.

Fase de Lançamento e Pós-Lançamento: Preparação para o Lançamento (Semana 25-26): Preparar materiais de marketing e promoção.

Configurar servidores e recursos para suportar o tráfego inicial.

Lançamento e Monitoramento (Semana 27): Lançar o aplicativo nas lojas de aplicativos (App Store, Google Play, etc.).

Monitorar o lançamento e responder a problemas emergentes.

Manutenção e Atualizações (Semana 28 em diante):

Continuar monitorando a aplicação após o lançamento.

Coletar feedback dos usuários e implementar melhorias.

Lançar atualizações regulares com base nas necessidades e feedback.

**DESENVOLVIMENTO**

O desenvolvimento de um aplicativo mobile envolve várias etapas, desde o planejamento até o lançamento. As linguagens de programação mais comuns para criar aplicativos móveis são:

1. Planejamento: Definir os objetivos do aplicativo, público-alvo e recursos essenciais.
2. Design: Criar a interface do usuário (UI) e a experiência do usuário (UX) do aplicativo.
3. Desenvolvimento Front-End: Utilizar linguagens como Swift (iOS) ou Kotlin (Android) para desenvolver a parte visual e interativa do aplicativo.
4. Desenvolvimento Back-End: Criar a lógica e os recursos do aplicativo usando linguagens como Python, Ruby, Node.js ou PHP.
5. Banco de Dados: Utilizar linguagens SQL ou NoSQL para gerenciar e armazenar dados do aplicativo.
6. Testes: Realizar testes de qualidade, desempenho e segurança para garantir que o aplicativo funcione corretamente.
7. Implantação: Disponibilizar o aplicativo nas lojas de aplicativos (App Store para iOS e Google Play Store para Android).
8. Manutenção: Continuar atualizando e melhorando o aplicativo com base no feedback dos usuários e nas necessidades em evolução.

As linguagens de programação usadas foram Styled Company e CSS, React IS, TypeScript, Recat RouterDom e NPM . Em banco de Dados utrilizamos o Microsoft SQL com o Azure Directory. Java para orietação aos objetos e dimensionamento.

Cada etapa do processo exige conhecimento técnico e coordenação para criar um aplicativo móvel eficaz e bem-sucedido.





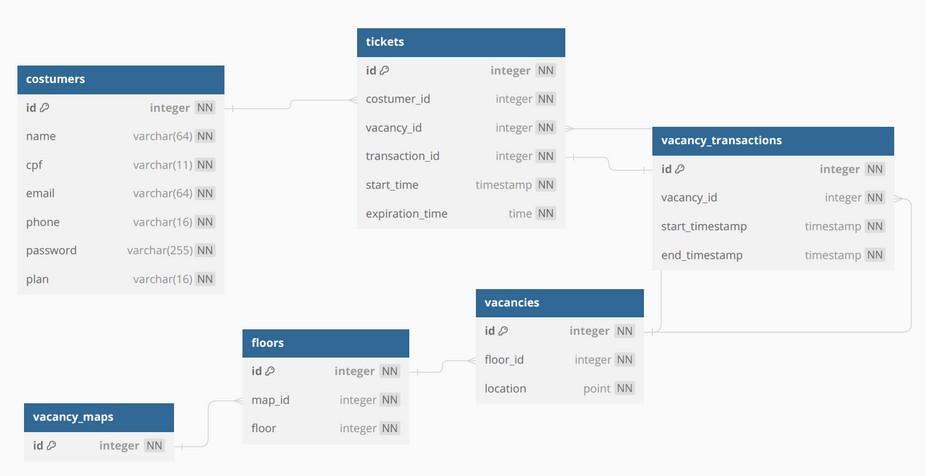
0 -Baixo

1 - Razoavel

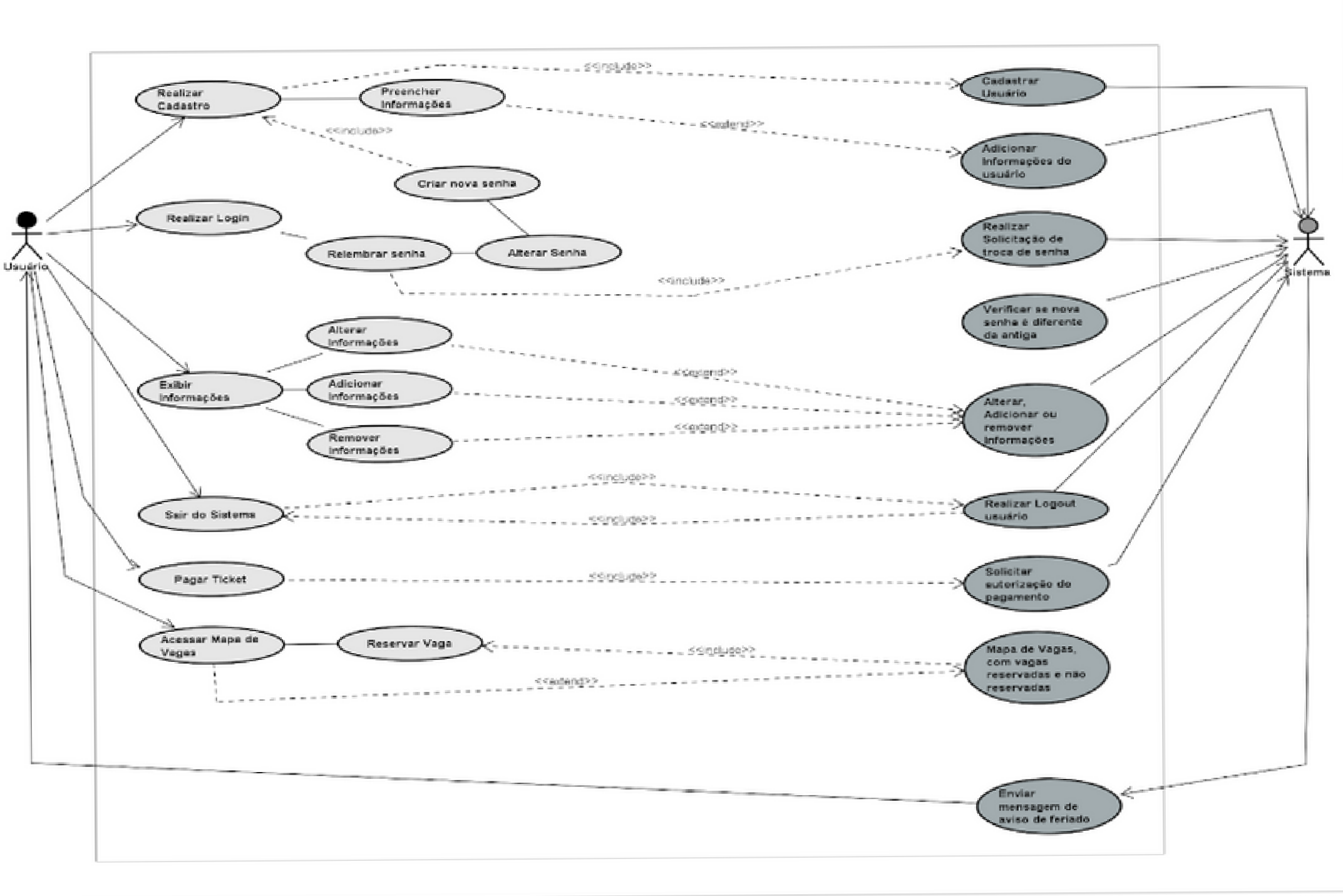
2 - Médio

3 - Alto

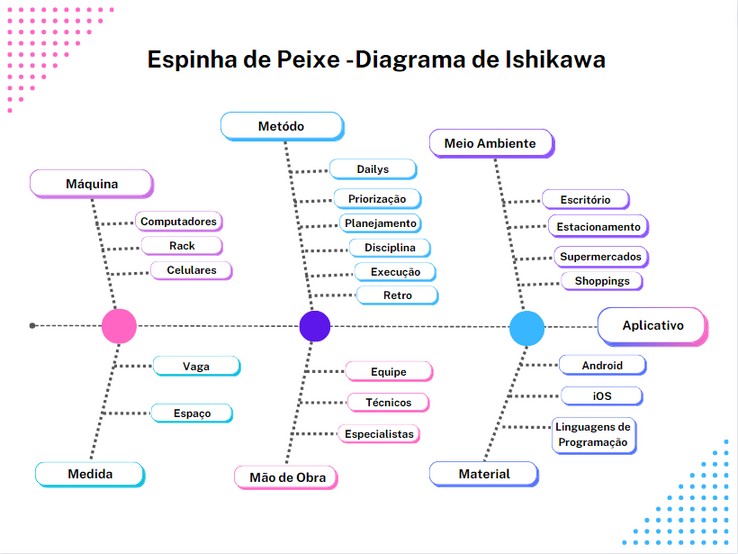
**DIAGRAMA DE BANCO DE DADOS**



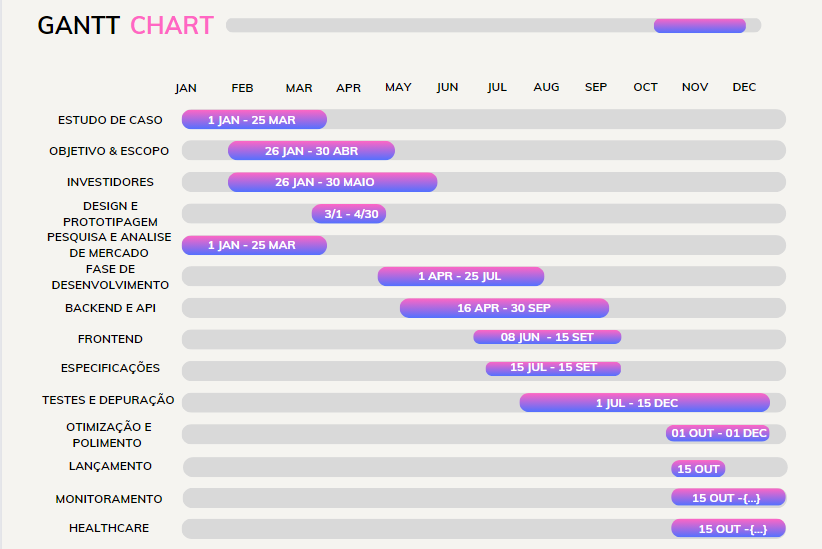
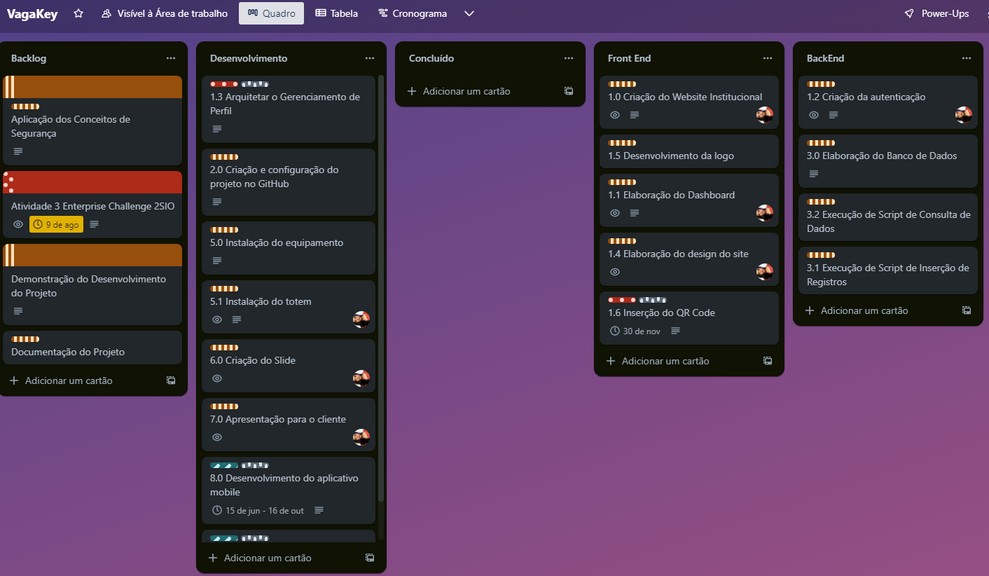
**CASO DE USO**



**PLAMILHA DE RISCOS E DIAGRAMA**



**CALENDÁRIO**



**LINK DO VÍDEO**

https://youtu.be/C-j6kN7hr3w

**LINK DA APRESENTAÇÃO**

**https://www.canva.com/design/DAFqinekKzU/QJ6Z9ypTZknm4CBM34\_VkA/edit?utm\_content=DAFqinekKzU&utm\_campaign=designshare&utm\_medium=link2&utm\_source=sharebutton**

**MEMBROS**

Gabriela Noleto Costa - RM: 86339 Luana Fumes de Aquino - RM: 93074 Gabriel Felix Bouças- RM: 93580 Bianca Hellen de Souza - RM: 93093 Isac dos Anjos França - RM93215