

Universidade Federal do Tocantins

Projeto de Sistemas

Aplicativo Moovin

Data: 26/05/2025

Alunos: Victhor C. Magalhães, João Vitor Reis Dias, Mayconn Cardoso Soares, Pedro Lucas, Ana Julia

Planejamento da Sprint 4

Objetivo da Sprint

A Sprint 4 tem como objetivo implementar funcionalidades críticas no aplicativo Moovin, utilizando **Flutter** para o desenvolvimento das interfaces e interações no aplicativo mobile e **Django** (com Django REST Framework) para a implementação das APIs e lógica de negócios no backend. Esta Sprint foca na integração com serviços externos, segurança e comunicação entre usuários.

Valor Entregue pela Sprint

Com esta Sprint, o Moovin passa a oferecer funcionalidades importantes para um ambiente seguro e interativo:

- **Integração com API de pagamento:** permite a realização de transações financeiras seguras através do app.
- **Recuperação de senha ("Esqueci minha senha"):** oferece ao usuário uma forma prática e segura de recuperar o acesso à sua conta.
- **Chat de mensagens:** possibilita a comunicação entre usuários diretamente pelo aplicativo, promovendo interação e dinamismo.
- **Landing page do produto:** permite reunir informações importantes sobre o aplicativo

Cronograma

- **Início:** 26/05
- **Primeira Revisão:** 02/06
- **Segunda Revisão:** 08/06
- **Entrega:** 09/06

Atividades da Sprint

1. Integração com API de Pagamento

Responsável: Victhor

Revisores: Mayconn

Descrição:

Implementar a integração de uma solução de pagamento no aplicativo, permitindo que os usuários realizem transações diretamente no Moovin. A interface será desenvolvida em **Flutter**, enquanto o backend em **Django** será responsável por validar e registrar as transações, garantindo a segurança dos processos.

Funcionalidades:

- Integração com provedor de pagamento Stripe via SDK ou API REST.
- Interface em Web para seleção da assinatura e execução do pagamento.
- Endpoint no Django para receber e validar notificações de pagamento (webhooks).
- Armazenamento seguro de registros de transações no banco de dados.
- Feedback em tempo real ao usuário sobre o status da transação.

2. Chat de Mensagens

Responsável: Mayconn e Pedro

Revisores: João Vitor

Descrição:

Desenvolver uma funcionalidade de chat no aplicativo, permitindo que usuários

conversem entre si. O frontend será implementado com **Flutter**, utilizando sockets ou um serviço em tempo real, enquanto o backend em **Django** será responsável pela autenticação, armazenamento e gerenciamento das mensagens.

Funcionalidades:

- Interface de chat com envio e recebimento de mensagens em tempo real no Flutter.
- Backend em Django para persistência das mensagens e controle de acesso.
- Sistema de autenticação JWT para garantir segurança na troca de mensagens.
- Integrar com as notificações já implementada na sprint anterior
- Controle de status de mensagens: enviadas, entregues e lidas.
- Deve haver outras duas funcionalidades disponíveis juntas com o chat: Opção de registrar visitas(backend já implementado) e registrar aluguéis (backend já implementado). Essas funcionalidades devem aparecer apenas para usuário premium.

3. Recuperação de Senha ("Esqueci minha senha")

Responsável: João Vitor

Revisores: Ana Júlia, Pedro

Descrição:

Criar a funcionalidade de recuperação de senha via e-mail, permitindo que usuários redefinam sua senha de forma segura. O processo será coordenado entre o frontend em **Flutter** e o backend em **Django**, que gerará tokens de redefinição e enviará e-mails automáticos.

Funcionalidades:

- Tela no Flutter para solicitação de recuperação de senha, informando o e-mail cadastrado.
- Backend em Django responsável por:
 - Gerar um token seguro e temporário.
 - Enviar e-mail automático com link para redefinição.
 - Validar o token e permitir alteração segura da senha.
- Feedback claro para o usuário sobre o andamento do processo.
- Expiração automática de tokens após determinado período.

4. Landing Page do produto

Responsável: Ana Júlia

Revisores: Victhor

Descrição:

Desenvolvimento da página oficial de apresentação do produto, com objetivo de comunicar de forma clara e atrativa as principais características e diferenciais do aplicativo. A landing page deve refletir a identidade visual do app, com um estilo minimalista, moderno e responsivo, garantindo uma boa experiência tanto em desktop quanto em dispositivos móveis.

Deverá conter seções bem definidas que facilitem a navegação e incentivem a conversão, especialmente para a assinatura do plano premium e o download do aplicativo.

Funcionalidades e Seções:

1. Topo da Página:

- Nome e slogan do app.
- Imagem ou mockup do aplicativo em uso.
- Botão "Baixar APK" com destaque.
- Botão "Assine agora" que leva ao checkout de pagamento.

2. Descrição e Benefícios do Produto:

- Texto objetivo sobre o que é o Moovin e qual problema resolve.
- Lista de principais funcionalidades (em ícones ou bullet points).
- Depoimentos de usuários ou cases de sucesso (se houver).

3. Diferenciais do Moovin:

- Destaque visual para os diferenciais competitivos.
- Por exemplo: facilidade, transparência, eficiência etc. Consultar escopo do projeto para mais informações

4. Seção "Plano Premium para Proprietários":

- Explicação clara dos benefícios exclusivos da assinatura.
- Tabela comparativa: **Plano Gratuito** vs. **Plano Premium**.
- Botão "Assine agora" posicionado estrategicamente.

5. Instruções para Download:

- Link direto para download do APK.
- QR Code para facilitar o download via smartphone.
- Instruções para habilitar "fontes desconhecidas" no Android, se necessário.

6. Perguntas Frequentes (FAQ):

- Seção com respostas rápidas para dúvidas comuns:
 - Como funciona a assinatura?
 - É seguro baixar o APK?
 - Como recuperar minha senha?

Funcionalidades Técnicas:

- Link "Assine agora" → Redireciona para página de checkout de pagamento.
- Link para download do APK → Com opção de copiar link ou abrir diretamente.