

# PLANO DE DESENVOLVIMENTO DETALHADO - MONEY QUEST

## PLANO DE DESENVOLVIMENTO DETALHADO MONEY QUEST

### 1. Planejamento Inicial

#### 1.1. Configuracao do Ambiente

- Instale Unity LTS (recomendo 2022.x ou mais recente).
- Instale Visual Studio ou VS Code.
- Crie um projeto Unity 2D.
- Configure o repositório no GitHub (.gitignore para Unity).
- Separe pastas para: Assets, Scripts, Scenes, Art, UI, Audio, Plugins.

#### 1.2. Requisitos & Prototipagem

- Revise e liste todos os requisitos funcionais e não funcionais.
- Defina as principais telas e fluxos no Figma (se ainda não estiverem prontos).
- Detalhe os fluxos de usuário: Login Home Guilda/Missoes Batalha Investimento Loja/Inventario/Quiz Status/Personagem.

### 2. Estrutura Basica do Projeto

#### 2.1. Criacao das Scenes principais

- Login, Home (fogueira), Guilda (missoes), Batalha, Investimento, Loja, Quiz, Inventario/Personagem, Pop-ups

#### 2.2. Organize Prefabs de UI

- Crie prefabs para botoes, paineis, barras de progresso, pop-ups etc.
- Importe ou crie assets em pixel art de acordo com o tema.

### 3. Sistema de Usuario e Autenticacao

#### 3.1. Tela de Login/Cadastro

- Implemente UI com Canvas, campos de texto, botoes.
- Integre Firebase Auth para registro e login por e-mail/senha.
- Valide login/logout e redirecionamento para a Home.

### 4. Estrutura dos Dados

#### 4.1. Firebase Firestore

- Modele as colecoes: jogadores, personagens, missoes, investimentos, quizzes, inventario.
- Implemente scripts para salvar e recuperar dados do progresso do jogador.

# PLANO DE DESENVOLVIMENTO DETALHADO - MONEY QUEST

## 4.2. ScriptableObjects

- Use ScriptableObjects para definir personagens, inimigos, itens, tipos de investimento.

## 5. Mecanica Central de Jogo

### 5.1. Party Management

- Sistema para criar/editar grupo de 3 personagens (Tank, DPS, Healer).
- UI de selecao e visualizacao dos status (vida, fome, moedas, experiencia).

### 5.2. Sistema de Missoes

- Gerencie missoes da Guilda (nome, recompensa, dificuldade).
- Adicione logica para sorteio ou atualizacao diaria das missoes.

### 5.3. Sistema de Batalha (Turn-based)

- Implemente logica de turnos: jogador escolhe acao, inimigo responde.
- Adicione skills, efeitos visuais e sons.
- Gerencie recompensas ao final (moedas, itens, XP).

## 6. Sistemas Educacionais

### 6.1. Investimentos

- Telas e scripts para simulacao de Poupanca, Tesouro Direto e Acoes.
- Interface para inserir valor, tempo, taxa e exibir rendimento.
- Use formulas de juros simples e compostos.

### 6.2. Quiz Educativo

- Tela para perguntas de multipla escolha, com banco de questoes salvo no Firestore ou local.
- Scripts para verificar respostas e dar recompensas.

## 7. Sistema de Loja e Inventario

- UI para listar itens a venda.
- Logica de compra (checa saldo, adiciona ao inventario, debita moedas).
- Inventario visual, com slots para armas, comida, pocoes.

## 8. Navegacao e UX

- Implemente transicoes de tela com SceneManager.
- Crie pop-ups para missoes, quiz, confirmacoes de acao, erro de login etc.
- Ajuste o layout para varias resolucoes (pelo menos mobile 390x844 e 1080x2400).

# PLANO DE DESENVOLVIMENTO DETALHADO - MONEY QUEST

## 9. Polimento e Feedback

- Adicione sons, trilhas, efeitos visuais para acoes principais.
- Animacoes simples (personagens andando, atacando, comemorando).
- Feedback instantaneo (ganho de moedas, missoes completas, respostas certas/erradas).

## 10. Testes

- Testes funcionais: cada funcionalidade separada (login, batalha, investir, quiz).
- Testes de usabilidade: peça para colegas/jovens jogarem e darem feedback.
- Testes de integracao: fluxo completo, salvando e recuperando progresso.
- Testes de desempenho: rode em aparelhos reais Android.

## 11. Ajustes Finais e Documentacao

- Corrija bugs relatados nos testes.
- Faça ultimos ajustes de UX/UI, traducoes, textos.
- Atualize documentacao: README, manual de uso, comentarios nos scripts.

## 12. Deploy

- Gere o APK de producao.
- Faça testes finais no APK em varios dispositivos Android.
- Prepare a apresentacao final e documentacao para entrega.

## Resumo Visual (Exemplo de Ordem de Implementacao)

1. Login/Cadastro
2. Home + Navegacao
3. Party e Missoes
4. Batalha
5. Investimento
6. Quiz
7. Loja/Inventario
8. Polimento/Feedback
9. Testes
10. Deploy

## Dicas de Ouro

- Trabalhe sempre em pequenas tarefas (Trello/Kanban).
- Integre e teste tudo frequentemente para evitar bugs acumulados.
- Use o GitHub para controle de versao nunca trabalhe so local!

## **PLANO DE DESENVOLVIMENTO DETALHADO - MONEY QUEST**

- Foque primeiro no fluxo central: login home missao/batalha recompensa.
- Quando travar, volte aos tutoriais de cada etapa (nao pule etapas essenciais).