**Documento A&P - Edu Planner**

**Versão:** 1.0  
**Data:** 30/06/2025  
**Autores:** Pedro Dylan Fernandes, Felipe Haiashida, Isaac Alves, Rafael Cardoso, Davi Bertasone Kirisawa, João Pedro Ibiapina, Ryan Macedo, Louise Sampaio  
**Professora:** Fabiana Gomes Marinho

### **1. Objetivo**

Este documento descreve a arquitetura, os componentes, os fluxos e as principais decisões técnicas do **Edu Planner**, sistema voltado para gerenciamento de estudos, organização de tarefas e incentivo à produtividade com elementos de gamificação.

### **2. Escopo**

Aplica-se ao ciclo de vida completo do Edu Planner: requisitos, arquitetura, desenvolvimento, integração, testes e entrega.

Cobertura: - Tarefas e calendário de estudos - Gestão de ciclos Pomodoro - Gamificação com níveis e recompensas - Perfil personalizável do usuário

### **3. Arquitetura Geral**

* **Frontend:** React Native
* **Backend:** Node.js com NestJS (Camadas: Domain, Infra, App, Interface)
* **Banco de Dados:** SQLite (armazenamento local)
* **Autenticação:** JWT stateless com MFA opcional
* **Criptografia:** AES para dados em repouso e HTTPS para dados em trânsito
* **Tarefas assíncronas:** Workers Node.js com BullMQ

### **4. Componentes Principais**

* **Autenticação:** Login JWT, Refresh Token, MFA opcional
* **Gestão de Tarefas e Calendário:** CRUD de tarefas, prazos e calendário pessoal
* **Pomodoro:** Temporizador com pausa, ciclos e estatísticas
* **Gamificação:** Níveis, recompensas e loja de itens visuais
* **Perfil do Usuário:** Personalização de avatar e visual

### **5. Fluxos Importantes**

#### Login e Autenticação:

1. Usuário envia login → AuthController
2. Gera JWT e Refresh Token
3. Refresh Token permite novo JWT por 15 dias

#### Criação de Tarefa:

1. Usuário cria Tarefa → TaskController
2. Define prazo, prioridade e dificuldade
3. Vincula ao calendário

#### Execução Pomodoro:

1. Usuário inicia ciclo Pomodoro
2. Sistema contabiliza tempo e pausas
3. Pontuação é atribuída após sessão concluída

#### Gamificação:

1. Ao concluir tarefas e ciclos, o usuário ganha pontos
2. Pode trocar pontos por itens na loja e evoluir seu perfil

### **6. Decisões Arquiteturais**

* JWT stateless para reduzir complexidade de sessão
* Armazenamento local por simplicidade e foco offline
* Pomodoro e gamificação independentes dos dados sensíveis
* Modularização em camadas para facilitar manutenção e testes

### **7. Riscos e Mitigações**

| Risco | Probabilidade | Impacto | Mitigação |
| --- | --- | --- | --- |
| Inconsistência nos dados locais | Média | Médio | Backup automático e rollback |
| Falhas no Pomodoro | Alta | Médio | Retentativas automáticas e logs de erro |
| Interface pouco intuitiva | Média | Alto | Testes UX com usuários-piloto |
| Performance ruim em aparelhos antigos | Média | Médio | Otimização de renderização e processos leves |

### **8. Padrões e Práticas**

* Code Review obrigatório via GitHub PRs
* Testes unitários com Jest (> 80% de cobertura)
* ESLint e Prettier padronizando código
* Pipeline CI/CD com GitHub Actions

### **9. Anexos**

* Diagrama de Componentes (a ser incluído)
* Diagrama de Fluxo do Pomodoro (a ser incluído)

**Fim do Documento A&P - Edu Planner**