

## DOCUMENTAÇÃO TÉCNICA DE PROJETO

# LUMENBORN

2D Multi-Layer Idle/Action RPG

<b>Tipo de Documento</b>	Project Design Document (PDD) / Game Design Document (GDD)
<b>Versão</b>	1.0.0
<b>Data</b>	Janeiro 2026
<b>Status</b>	Em Desenvolvimento
<b>Classificação</b>	Portfólio Profissional

*Desenvolvido com metodologias ágeis e boas práticas de engenharia de software*

# Índice

1. Sumário Executivo
2. Escopo do Projeto
3. Requisitos Funcionais
4. Requisitos Não-Funcionais
5. Arquitetura do Sistema
6. Stack Tecnológico
7. Modelagem de Dados
8. Diagramas de Fluxo
9. Especificações de Módulos
10. Sistema de Economia
11. Infraestrutura e Deploy
12. Segurança e Compliance
13. Plano de Testes
14. Roadmap de Desenvolvimento
15. Estimativas e Métricas
16. Glossário Técnico

# 1. Sumário Executivo

## 1.1 Visão Geral

Lumenborn é um projeto de desenvolvimento de jogo mobile/desktop que combina mecânicas de idle games com elementos de Action RPG. O sistema foi arquitetado para suportar multiplataforma, economia real entre jogadores, e infraestrutura escalável para servidores dedicados e suporte a modificações da comunidade.

## 1.2 Objetivos do Projeto

- Desenvolver um jogo 2D multiplataforma com arquitetura escalável
- Implementar sistema de economia com transações reais (P2P marketplace)
- Criar infraestrutura para servidores dedicados e sistema de mods
- Garantir experiência balanceada entre jogadores pagantes e não-pagantes
- Estabelecer pipeline de CI/CD para deploys contínuos

## 1.3 Stakeholders

Stakeholder	Papel	Responsabilidade
Desenvolvedor	Full-Stack Dev	Arquitetura, código, deploy
Usuários Finais	Players	Feedback, beta testing
Comunidade	Modders	Conteúdo adicional

## 2. Escopo do Projeto

### 2.1 Dentro do Escopo (In Scope)

- Motor de jogo 2D side-scroller com sistema de fases/níveis
- Sistema de progressão multi-camada (Ação, Vila, Habilidades)
- Backend para autenticação, persistência e sincronização
- Sistema de economia com marketplace P2P
- Gateway de pagamento para transações reais
- API RESTful para comunicação cliente-servidor
- Painel administrativo para gerenciamento
- Sistema anti-fraude e anti-bot
- Suporte multiplataforma (Android, iOS, Windows, macOS)

### 2.2 Fora do Escopo (Out of Scope) - v1.0

- Realidade virtual/aumentada
- Modo multiplayer síncrono (PvP real-time)
- Versão para consoles (PlayStation, Xbox, Switch)
- Aplicativo companion separado

### 2.3 Premissas

- Usuário possui dispositivo com Android 8.0+, iOS 13+, ou Windows 10+
- Conexão à internet necessária para features online
- Modo offline disponível com sincronização posterior

### 2.4 Restrições

- Budget limitado - desenvolvimento solo/pequena equipe
- Timeline de 10-12 meses para MVP
- Conformidade com políticas das lojas (Google Play, App Store)

### 3. Requisitos Funcionais

#### 3.1 Módulo de Gameplay Core

ID	Requisito	Prioridade	Status
RF001	Sistema de movimento automático do personagem	Alta	Planejado
RF002	Deteção de input touch/click para interações	Alta	Planejado
RF003	Sistema de combate com dano, vida, morte	Alta	Planejado
RF004	Spawner de inimigos com configuração por nível	Alta	Planejado
RF005	Sistema de coleta de recursos (orbs, itens)	Alta	Planejado
RF006	Sistema de combos com multiplicador	Média	Planejado
RF007	Progressão por níveis (1-10, boss, novo bioma)	Alta	Planejado
RF008	Seleção de nível (migração livre)	Média	Planejado

#### 3.2 Módulo de Progressão

ID	Requisito	Prioridade	Status
RF009	Sistema de Eco-Prestige (reset 80%, ganho 20% permanente)	Alta	Planejado
RF010	Moedas de Prestígio e loja de upgrades permanentes	Alta	Planejado
RF011	4 árvores de habilidades com skills ativas/passivas	Média	Planejado
RF012	Sistema de classes com skills específicas	Baixa	Futuro

#### 3.3 Módulo de Inventário e Itens

ID	Requisito	Prioridade	Status
RF013	6 slots de equipamento (arma, elmo, peito, luvas, botas, amuleto)	Alta	Planejado
RF014	Sistema de raridades (Comum → Primordial)	Alta	Planejado
RF015	Itens lendários com poderes únicos	Média	Planejado
RF016	Sistema de Sets com bônus 2/4/6 peças	Média	Planejado
RF017	Sistema de upgrade +1 até +15 (estilo MU)	Baixa	Futuro

#### 3.4 Módulo de Vila

ID	Requisito	Prioridade	Status
RF018	Ferreiro - upgrade e craft de armas	Alta	Planejado
RF019	Mercador - compra/venda de itens	Alta	Planejado
RF020	Igreja - cura e buffs de luz	Alta	Planejado

RF021	Alquimista - poções e consumíveis	Alta	Planejado
RF022	Altar Angelical - extração de poderes (seguro)	Média	Planejado
RF023	Altar Demoníaco - extração de poderes (risco)	Média	Planejado

### 3.5 Módulo de Economia

ID	Requisito	Prioridade	Status
RF024	Auction House com moeda in-game (Luz)	Alta	v2.0
RF025	Auction House com dinheiro real (gateway)	Alta	v2.0
RF026	Sistema de carteira digital para jogadores	Alta	v2.0
RF027	Taxa de transação 15% (reinvestimento)	Alta	v2.0
RF028	Sistema de saque com verificação KYC	Alta	v2.0

### 3.6 Módulo Online

ID	Requisito	Prioridade	Status
RF029	Sistema de autenticação (email, OAuth)	Alta	v1.5
RF030	Save na nuvem com sincronização	Alta	v1.5
RF031	Leaderboards globais e por temporada	Média	v2.0
RF032	Separação Vanilla/Modded com validação	Alta	v2.0
RF033	API para servidores dedicados da comunidade	Baixa	v3.0
RF034	Sistema de mods com carregamento runtime	Baixa	v3.0

## 4. Requisitos Não-Funcionais

### 4.1 Performance

ID	Requisito	Métrica
RNF001	Frame rate estável em dispositivos médios	≥ 60 FPS
RNF002	Tempo de carregamento inicial	< 5 segundos
RNF003	Uso de memória RAM	< 500 MB
RNF004	Tamanho do APK/IPA	< 150 MB
RNF005	Tempo de resposta da API	< 200 ms

### 4.2 Escalabilidade

ID	Requisito	Métrica
RNF006	Usuários simultâneos suportados	10.000+ CCU
RNF007	Transações por segundo no marketplace	1.000+ TPS
RNF008	Arquitetura deve permitir scaling horizontal	Auto-scaling

### 4.3 Segurança

ID	Requisito	Métrica
RNF009	Criptografia de dados sensíveis	AES-256
RNF010	Comunicação cliente-servidor	TLS 1.3
RNF011	Autenticação de tokens	JWT + Refresh
RNF012	Conformidade com LGPD/GDPR	100%
RNF013	Sistema anti-cheat server-side	Validação total

### 4.4 Disponibilidade

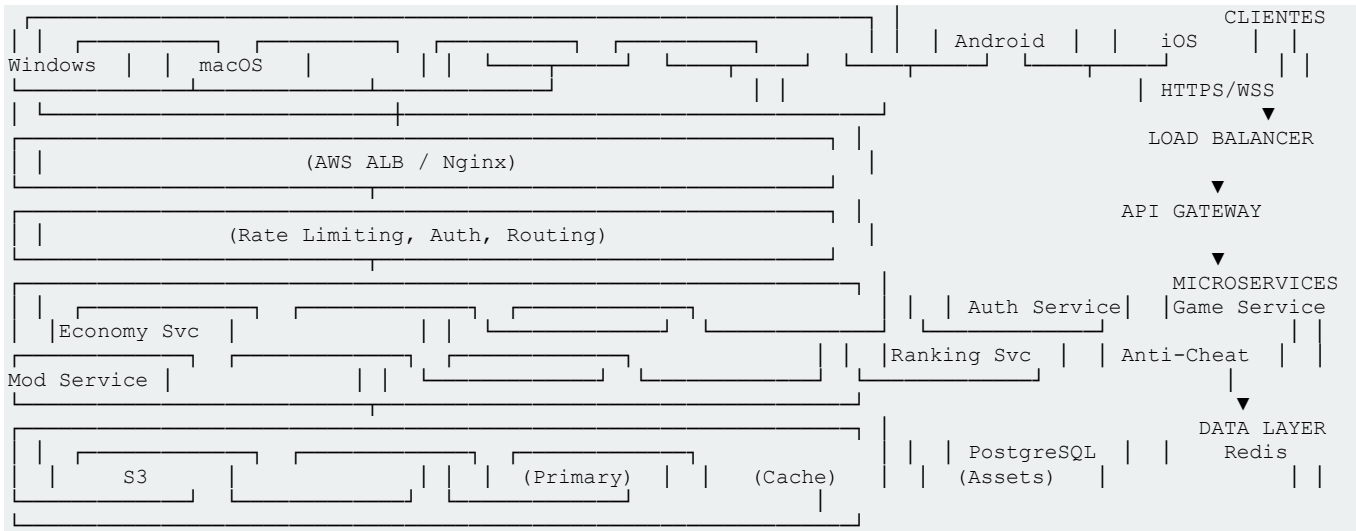
ID	Requisito	Métrica
RNF014	Uptime do serviço	99.9% SLA
RNF015	Modo offline funcional	100% gameplay
RNF016	Backup de dados	Diário + real-time

## 5. Arquitetura do Sistema

### 5.1 Visão Geral da Arquitetura

O sistema utiliza arquitetura cliente-servidor com separação clara de responsabilidades. O cliente (jogo) é responsável pela renderização e input, enquanto o servidor autoritativo valida todas as ações críticas para garantir integridade.

#### Diagrama de Arquitetura (Alto Nível):



### 5.2 Padrões Arquiteturais

- **Cliente:** Component-Based Architecture (Unity ECS-like)
- **Backend:** Microservices com API RESTful
- **Comunicação:** Request-Response (HTTP) + Event-Driven (WebSocket)
- **Persistência:** Repository Pattern com Unit of Work
- **Cache:** Cache-Aside Pattern com Redis

## 6. Stack Tecnológico

### 6.1 Cliente (Game)

Componente	Tecnologia	Justificativa
Game Engine	Unity 2022 LTS	Multiplataforma, mercado
Linguagem	C#	Nativa Unity, tipagem forte
Networking	Unity Netcode / Mirror	Sincronização, servidores
UI Framework	Unity UI Toolkit	Performance, flexibilidade
Serialização	JSON / MessagePack	Compatibilidade, performance
Analytics	Unity Analytics + Firebase	Métricas, eventos

### 6.2 Backend

Componente	Tecnologia	Justificativa
Runtime	Node.js / .NET Core	Performance, ecossistema
Framework	Express / ASP.NET	Maturidade, suporte
Database	PostgreSQL	ACID, JSON support
Cache	Redis	Performance, pub/sub
Message Queue	RabbitMQ / Redis Streams	Async processing
Payments	Stripe / PayPal	Global, confiável

### 6.3 Infraestrutura

Componente	Tecnologia	Justificativa
Cloud Provider	AWS / GCP	Escalabilidade, serviços
Containers	Docker + Kubernetes	Orquestração, scaling
CI/CD	GitHub Actions	Integração, automação
CDN	CloudFront / CloudFlare	Assets, latência
Monitoring	Prometheus + Grafana	Métricas, alertas
Logging	ELK Stack	Centralização, busca

## 7. Modelagem de Dados

### 7.1 Entidades Principais

#### Player (Jogador)

```
Player {  id: UUID (PK)  email: VARCHAR(255) UNIQUE  username: VARCHAR(50) UNIQUE
password_hash: VARCHAR(255)  created_at: TIMESTAMP  last_login: TIMESTAMP  is_banned:
BOOLEAN  ban_reason: TEXT  wallet_balance: DECIMAL(10,2)  total_playtime: INTEGER
settings: JSONB }
```

#### Character (Personagem)

```
Character {  id: UUID (PK)  player_id: UUID (FK -> Player)  name: VARCHAR(50)  class_id:
INTEGER  level: INTEGER  experience: BIGINT  current_stage: INTEGER  max_stage_unlocked:
INTEGER  is_vanilla: BOOLEAN  prestige_count: INTEGER  prestige_currency: INTEGER
base_stats: JSONB  light_dark_balance: INTEGER (-100 to 100)  created_at: TIMESTAMP
total_playtime: INTEGER }
```

#### Item (Item)

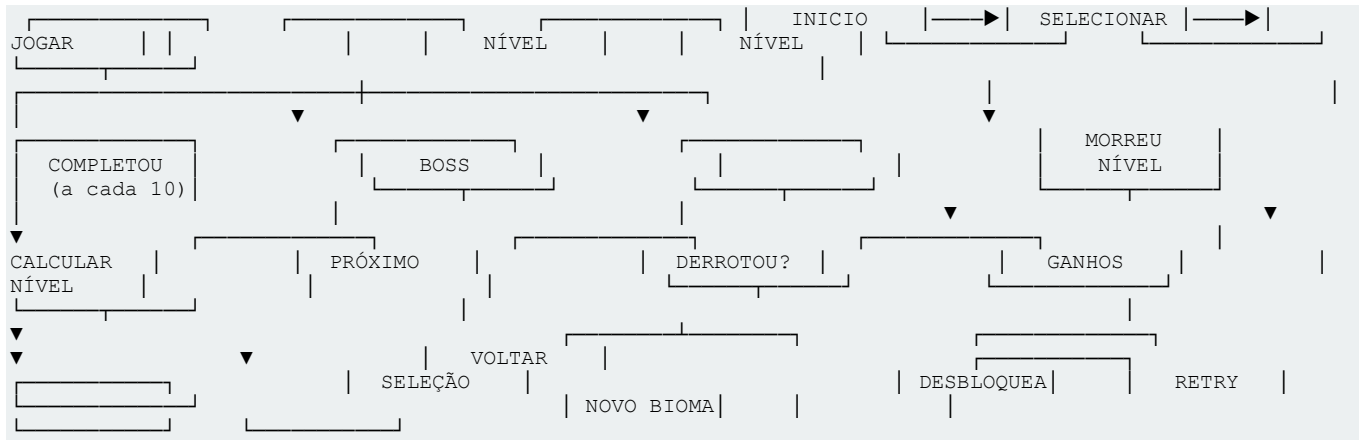
```
Item {  id: UUID (PK)  template_id: INTEGER (FK -> ItemTemplate)  character_id: UUID (FK
-> Character)  rarity: ENUM(common, magic, rare, legendary, set, ancestral, primordial)
upgrade_level: INTEGER (0-15)  stats: JSONB  unique_power_id: INTEGER  is_equipped:
BOOLEAN  equipment_slot: ENUM(weapon, helm, chest, gloves, boots, amulet)  is_soulbound:
BOOLEAN  acquired_at: TIMESTAMP  trade_cooldown_until: TIMESTAMP }
```

#### Transaction (Transação)

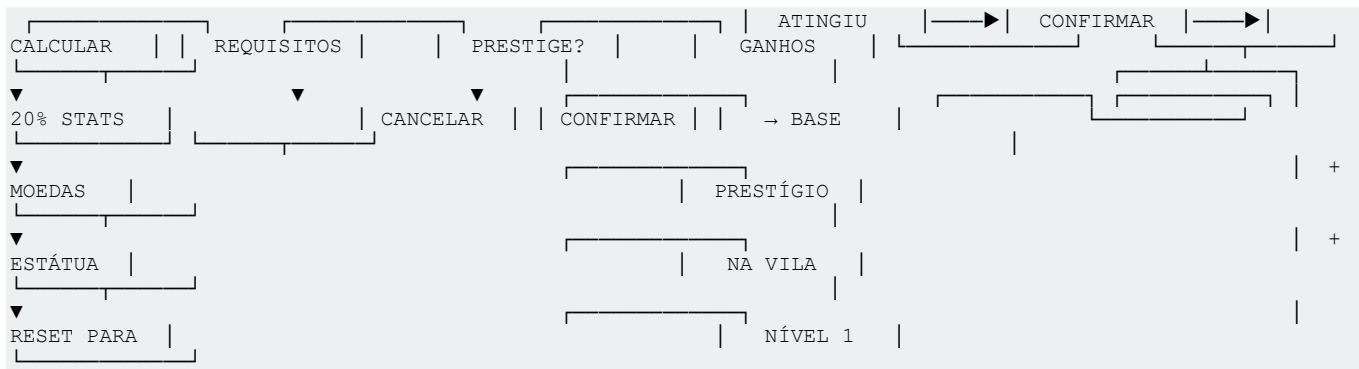
```
Transaction {  id: UUID (PK)  seller_id: UUID (FK -> Player)  buyer_id: UUID (FK ->
Player)  item_id: UUID (FK -> Item)  currency_type: ENUM(light, real_money)  amount:
DECIMAL(15,2)  fee_amount: DECIMAL(15,2)  status: ENUM(pending, completed, cancelled,
refunded)  created_at: TIMESTAMP  completed_at: TIMESTAMP  payment_gateway_id:
VARCHAR(255) }
```

## 8. Diagramas de Fluxo

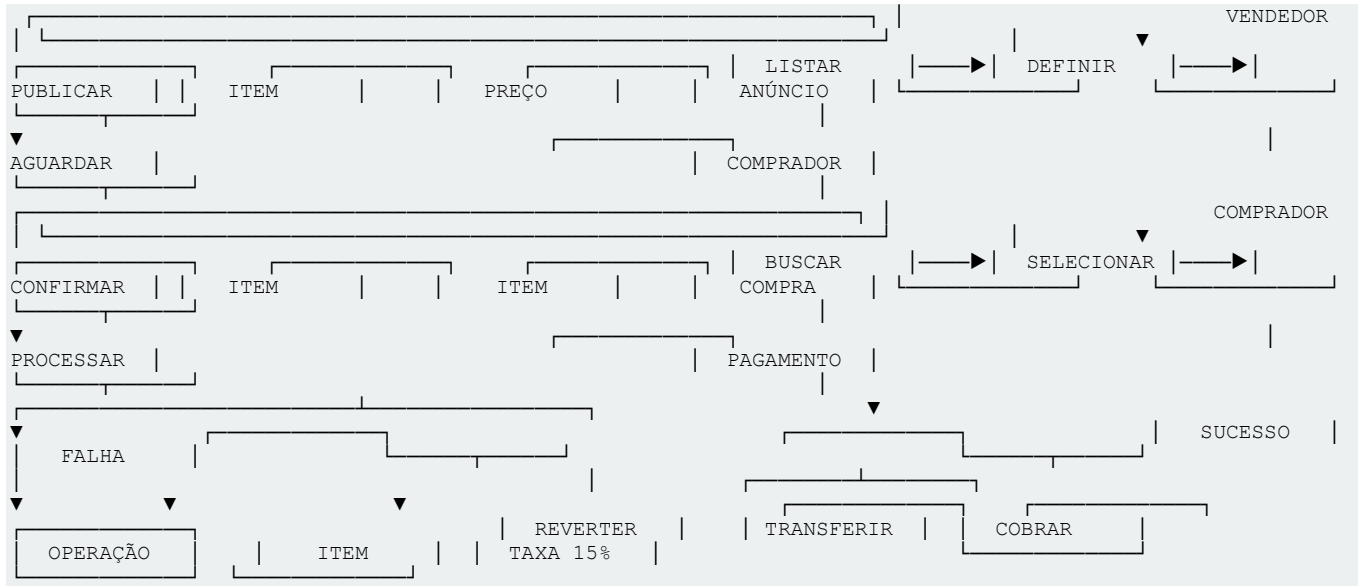
### 8.1 Fluxo de Gameplay Principal



### 8.2 Fluxo de Eco-Prestige



### 8.3 Fluxo de Transação (Auction House)



## 9. Especificações de Módulos

### 9.1 Módulo de Combate

- **Responsabilidade:** Gerenciar toda lógica de combate, dano, e morte
- **Componentes:** DamageCalculator, HealthSystem, CombatManager, ComboSystem
- **Dependências:** StatsModule, ItemModule, SkillModule
- **Eventos Emitidos:** OnDamageDealt, OnEnemyKilled, OnPlayerDeath, OnComboChanged

### 9.2 Módulo de Progressão

- **Responsabilidade:** Gerenciar XP, níveis, prestige e upgrades permanentes
- **Componentes:** ExperienceManager, PrestigeManager, SkillTreeManager
- **Dependências:** SaveModule, StatsModule
- **Eventos Emitidos:** OnLevelUp, OnPrestigeComplete, OnSkillUnlocked

### 9.3 Módulo de Inventário

- **Responsabilidade:** Gerenciar itens, equipamentos, e storage
- **Componentes:** InventoryManager, EquipmentManager, ItemFactory, LootGenerator
- **Dependências:** SaveModule, StatsModule
- **Eventos Emitidos:** OnItemAcquired, OnItemEquipped, OnItemDropped

### 9.4 Módulo de Economia

- **Responsabilidade:** Gerenciar moedas, transações, e marketplace
- **Componentes:** CurrencyManager, AuctionHouse, PaymentGateway, WalletManager
- **Dependências:** AuthModule, InventoryModule, NetworkModule
- **Eventos Emitidos:** OnTransactionComplete, OnBalanceChanged, OnListingCreated

## 10. Sistema de Economia

### 10.1 Filosofia Econômica

#### "Tempo investido > Dinheiro investido"

O sistema econômico foi projetado para que jogadores dedicados sempre tenham vantagem sobre jogadores que apenas gastam dinheiro. Isso é alcançado através de um split de poder onde a maioria vem de fontes não-compráveis.

### 10.2 Distribuição de Poder

Fonte de Poder	% do Total	Comprável?
Itens Equipados	35%	Sim (via Auction House)
Nível de Habilidades	25%	Não (apenas jogando)
Eco-Prestiges + Estátuas	20%	Não (apenas jogando)
Vila (Construções)	15%	Aceleração apenas
Maestria	5%	Não (apenas jogando)

### 10.3 Progressão Estilo MU x100

Tempo de Jogo	Progresso Esperado
30 minutos	Level 10, primeiro boss derrotado, itens azuis
2 horas	Level 30, 3 bosses, alguns itens raros
1 dia	Level 50+, build básica formada
1 semana	Level 100+, lendários, primeiro prestige
1 mês	Múltiplos prestiges, builds otimizadas

### 10.4 Estratégia Anti-Bot/Anti-P2W

- Drop generoso = itens abundantes = preços naturalmente baixos
- Melhores itens são Soulbound (não vendáveis)
- Caps de stats baseados em Prestige (não pode usar item OP sem jogar)
- Cooldown de 7 dias para revender item comprado
- Limite diário de compras (5 lendários, 2 sets, 1 ancestral)

**Resultado:** Bot não lucra porque preços são baixos. Comprar não dá vantagem real porque 65% do poder não é comprável.

## 11. Infraestrutura e Deploy

### 11.1 Ambientes

Ambiente	Propósito	Acesso
Development	Desenvolvimento local	Desenvolvedores
Staging	Testes de integração	QA + Devs
Production	Ambiente de produção	Usuários finais

### 11.2 Pipeline CI/CD

1. Push para branch → Trigger do GitHub Actions
2. Execução de testes unitários e de integração
3. Build do cliente (Unity) para plataformas alvo
4. Build dos serviços backend (Docker)
5. Deploy para ambiente de staging
6. Testes de smoke automatizados
7. Aprovação manual para produção
8. Deploy para produção com blue-green deployment

### 11.3 Estratégia de Scaling

- **Horizontal:** Auto-scaling de pods Kubernetes baseado em CPU/Memory
- **Database:** Read replicas para queries pesadas, sharding futuro
- **Cache:** Redis Cluster para alta disponibilidade
- **CDN:** Assets estáticos em edge locations globais

## 12. Segurança e Compliance

### 12.1 Autenticação e Autorização

- JWT tokens com refresh token rotation
- OAuth 2.0 para login social (Google, Apple, Facebook)
- Rate limiting por IP e por usuário
- 2FA opcional para contas com saldo real

### 12.2 Proteção de Dados

- Criptografia AES-256 para dados sensíveis em repouso
- TLS 1.3 para dados em trânsito
- Hashing bcrypt para senhas
- PCI DSS compliance para dados de pagamento (via gateway)

### 12.3 Sistema Anti-Cheat

Fase	Implementação
Fase 1	Detecção de padrões anômalos, tempo de sessão, múltiplas contas
Fase 2	Captcha inteligente durante gameplay, ML para detecção de bots
Fase 3	Fingerprinting de dispositivo, análise comportamental, honey pots

### 12.4 Compliance

- LGPD (Brasil) - Consentimento, direito ao esquecimento, portabilidade
- GDPR (EU) - Data protection, privacy by design
- COPPA (US) - Restrições para menores de 13 anos
- Políticas Google Play e App Store

## 13. Plano de Testes

### 13.1 Níveis de Teste

Tipo	Escopo	Ferramentas
Unitários	Funções e classes isoladas	NUnit, xUnit
Integração	Módulos integrados	Unity Test Framework
E2E	Fluxos completos	Automated UI Testing
Performance	Load e stress testing	Unity Profiler, k6
Segurança	Vulnerabilidades	OWASP ZAP, Burp Suite

### 13.2 Critérios de Aceitação

- Cobertura de código mínima: 70%
- Zero bugs críticos em produção
- Tempo de resposta da API < 200ms (P95)
- Frame rate estável ≥ 60 FPS em dispositivos de referência

## 14. Roadmap de Desenvolvimento

Versão	Timeline	Entregáveis
<b>v1.0</b>	Meses 1-10	Jogo single-player completo, todas mecânicas core, publicação
<b>v1.5</b>	Meses 11-13	Sistema de contas, save na nuvem, leaderboards básicos
<b>v2.0</b>	Meses 14-18	Vanilla/Modded, Auction House, rankings, temporadas, anti-bot
<b>v2.5</b>	Meses 19-21	Anti-bot avançado (ML), sistema de reports melhorado
<b>v3.0</b>	Meses 22-28	API de mods, servidores dedicados, hub de mods
<b>v3.5+</b>	Meses 29+	Expansões, novos biomas, classes, eventos sazonais

## 15. Estimativas e Métricas

### 15.1 Estimativa de Esforço (v1.0)

Módulo	Estimativa	Complexidade
Setup + Aprendizado Unity	80 horas	Média
Core Gameplay	120 horas	Alta
Sistema de Itens	80 horas	Alta
Sistema de Progressão	60 horas	Média
Vila e Construções	40 horas	Média
UI/UX	60 horas	Média
Arte e Assets	100 horas	Alta
Audio	20 horas	Baixa
Testes e Polish	60 horas	Média
<b>TOTAL</b>	<b>~620 horas</b>	-

### 15.2 KPIs do Projeto

- DAU (Daily Active Users) - Meta: 10.000+
- Retention D1/D7/D30 - Meta: 40%/20%/10%
- ARPU (Average Revenue Per User) - Meta: R\$ 2,00/mês
- Transaction Volume - Meta: R\$ 50.000/mês
- Bug Rate - Meta: < 0.1% de sessões com crash

## 16. Glossário Técnico

Termo	Definição
Eco-Prestige	Sistema de reset parcial onde jogador sacrifica 80% do progresso atual para ganhar 20% como bônus permanente
Vanilla	Modo de jogo oficial sem modificações, com ranking e economia oficial
Modded	Modo de jogo com modificações da comunidade, sem ranking oficial
Soulbound	Item que não pode ser transferido ou vendido entre jogadores
Auction House	Marketplace P2P onde jogadores podem comprar/vender itens
CCU	Concurrent Users - usuários conectados simultaneamente
KYC	Know Your Customer - verificação de identidade para transações financeiras
Idle Game	Gênero de jogo onde progressão continua mesmo sem interação ativa

## **FIM DO DOCUMENTO**

Lumenborn - Project Design Document v1.0

*Documento preparado seguindo padrões de engenharia de software*