**Documento de Backlog do Produto e da Sprint – EcoTask**

1. **Sequenciador (Waves / Increments)**

Este sequenciador define o que é o **Produto Mínimo Viável (MVP)** (Onda 1) e o que será construído nos incrementos subsequentes.

**Estrutura da Avaliação:** Funcionalidade (Esforço em E/EE/EEE, Valor para o Negócio em $/$$/$$$, Valor para o Usuário em ♡/♡♡/♡♡♡).

| Incremento | Funcionalidade | Esforço | Negócio | UX |
| --- | --- | --- | --- | --- |
| **Onda 1: MVP (Core do EcoTask)** | Cadastro de Usuário e Login | E | $$$ | ♡♡♡ |
|  | Check-in do Hábito (Registro) | E | $$$ | ♡♡♡ |
|  | Criação e Edição de Hábitos | EE | $$$ | ♡♡♡ |
|  | Cálculo de Eco-Pontos | EE | $$$ | ♡♡♡ |
|  | Tela de Perfil Básico | E | $$ | ♡♡ |
|  | Contador de *Streaks* (Consistência) | EE | $$ | ♡♡♡ |
| **Onda 2: Foco em Retenção e Qualidade** | Estatísticas Avançadas (Gráficos) | EEE | $$ | ♡♡ |
|  | Notificações e Lembretes | EE | $$ | ♡♡♡ |
|  | Sistema de *Badges* e Títulos | EE | $$ | ♡♡ |
| **Onda 3: Foco em Expansão e Diferenciação** | Leaderboards (Placares de Líderes) | EEE | $$ | ♡♡ |
|  | Sugestões de Hábito por IA | EEE | $$$ | ♡♡♡ |
|  | Integração com *Smart Home* / Wearables | EEE | $$ | ♡ |

**2. Backlog do Produto (product-backlog.pdf)**

O *backlog* é detalhado em níveis (Épicos, Features, Histórias de Usuário e Enablers).

| Tipo | Chave | Título | Épico Associado | Prioridade | Story Points |
| --- | --- | --- | --- | --- | --- |
| **ÉPICO** | **E1** | Core do Produto (MVP) |  | P1 (MVP) | 24 |
| **FEATURE** | F1.1 | Autenticação e Onboarding | E1 | P1 | Média |
| **US** | US1.1.1 | Cadastro de Usuário (E-mail/Senha) | E1/F1.1 | Alta | **3** |
| **US** | US1.1.2 | Login de Usuário | E1/F1.1 | Alta | **2** |
| **FEATURE** | F1.2 | Gerenciamento de Hábitos | E1 | P1 | Alta |
| **US** | US1.2.1 | Criação e Edição de Hábito | E1/F1.2 | Alta | **5** |
| **US** | US1.2.2 | Registro de Conclusão (Check-in) | E1/F1.2 | Alta | **5** |
| **FEATURE** | F1.3 | Gamificação Básica | E1 | P1 | Alta |
| **US** | US1.3.1 | Visualização de Eco-Pontos e Perfil | E1/F1.3 | Alta | **3** |
| **US** | US1.3.2 | Contador de *Streaks* (Consistência) | E1/F1.3 | Média | **3** |
| **ENABLER** | E1.4 | Configuração da Arquitetura e CI/CD | E1 | Alta | **3** |
| **ÉPICO** | E2 | Features Futuras (Incremento 1) |  | P2 (Futuro) | N/A |
| **FEATURE** | F2.1 | Estatísticas Avançadas | E2 | P2 | N/A |
| **FEATURE** | F2.2 | Notificações e Lembretes | E2 | P2 | N/A |

**3. Backlog da Sprint 1 (sprint-backlog.pdf)**

**Total de Story Points:** 24 SP.

| Chave | Título | Descrição | Story Points | Critérios de Aceitação (CAs) |
| --- | --- | --- | --- | --- |
| **US1.1.1** | **Cadastro de Usuário** | Como um novo Usuário, eu quero me cadastrar usando e-mail e senha, para ter acesso ao App e salvar meu progresso. | **3** | 1. DADO QUE sou um novo usuário, QUANDO preencho E-mail e Senha, ENTÃO sou cadastrado e logado automaticamente. 2. O sistema deve armazenar a senha do usuário utilizando **hashing** (RNF de Segurança). 3. O sistema deve validar se o e-mail já está em uso e exibir uma mensagem de erro clara. |
| **US1.1.2** | **Login de Usuário** | Como Usuário, eu quero fazer login com minhas credenciais, para retomar minha sessão e continuar a acumular Eco-Pontos. | **2** | 1. DADO QUE sou um usuário cadastrado, QUANDO insiro credenciais válidas, ENTÃO sou direcionado para a tela principal. 2. O sistema deve exibir uma mensagem de erro genérica se as credenciais forem inválidas. 3. O sistema deve gerar e usar um **token de sessão** para manter o usuário logado (RNF de Segurança). |
| **US1.2.1** | **Criação e Edição de Hábito** | Como Usuário, eu quero cadastrar e editar um novo Hábito Sustentável (ex: Banho Curto), para que ele apareça na minha lista de tarefas diárias. | **5** | 1. O usuário pode inserir um nome, selecionar frequência (Diária/Semanal) e categoria (Água/Lixo/Energia). 2. O sistema deve salvar o novo hábito no banco de dados, atribuindo uma pontuação padrão. 3. O usuário deve conseguir retornar à tela de edição para alterar o nome ou a frequência do hábito. |
| **US1.2.2** | **Registro de Conclusão (Check-in)** | Como Usuário, eu quero marcar um Hábito como concluído, para que o sistema me dê Eco-Pontos e eu possa rastrear minha consistência. | **5** | 1. DADO QUE estou na tela principal, QUANDO clico no botão *check-in* de um hábito, ENTÃO o status do hábito muda para concluído. 2. O sistema deve impedir um segundo *check-in* no mesmo dia (se for diário) ou período (se for semanal). 3. O tempo de resposta para o *check-in* (gravação no DB) deve ser **inferior a 2 segundos** (RNF de Performance). |
| **US1.3.1** | **Visualização de Eco-Pontos e Perfil** | Como Usuário, eu quero ver minha pontuação total de Eco-Pontos no meu Perfil, para medir meu progresso e sentir que estou contribuindo. | **3** | 1. O sistema deve exibir a pontuação total de Eco-Pontos atualizada no Perfil. 2. A tela de Perfil deve carregar o nome de usuário e a pontuação sem falhas. 3. Deve haver um botão claro de navegação para acessar a Tela de Perfil. |
| **US1.3.2** | **Contador de *Streaks*** | Como Usuário, eu quero que o sistema me mostre quantos dias consecutivos fiz *check-ins*, para me motivar a não quebrar a sequência. | **3** | 1. O sistema deve calcular e exibir o número de dias consecutivos de *check-ins* (o *streak*) no Perfil. 2. O *streak* deve ser resetado para zero se o usuário falhar em fazer *check-in* em todos os hábitos diários por um dia. |
| **E1.4** | **Configuração da Arquitetura e CI/CD** | Como Time de Desenvolvimento, eu preciso configurar o ambiente de backend, para podermos implementar as histórias da Sprint. | **3** | 1. O ambiente de desenvolvimento do Backend (API) deve estar configurado e funcional. 2. O modelo de dados inicial (tabelas Usuário e Hábito) deve estar criado no banco de dados. 3. Deve existir um pipeline de *deploy* básico configurado para o ambiente de *staging*. |

**4. Definições do Projeto**

**Definition of Ready (DoR)**

Um item do *backlog* está **pronto** para entrar na *Sprint* quando:

1. **Clareza da Necessidade:** A História de Usuário (HU) está escrita no formato "Como [Persona], eu quero [Ação] para que [Benefício]" (Modelo INVEST atendido).
2. **Critérios de Aceitação Definidos:** Os Critérios de Aceitação (CAs) estão claros, não ambíguos e são testáveis.
3. **Prioridade Definida:** O Product Owner definiu a prioridade do item no *backlog*.
4. **Estimativa Realizada:** O Time de Desenvolvimento forneceu uma estimativa relativa (em *Story Points*).
5. **Entendimento Compartilhado:** O Time confirmou com o PO que todos entendem o propósito e o escopo da HU.

**Definition of Done (DoD)**

Um incremento do produto está **Concluído** quando:

1. **Código Implementado e Revisado:** O código-fonte está implementado, passou por **revisão de código** (*code review*) e está devidamente versionado.
2. **Testes Unitários:** Todos os testes unitários foram escritos e **passaram** (cobertura mínima de 80%).
3. **Testes Funcionais Aprovados:** Todos os **Critérios de Aceitação (CAs)** foram testados e **validados** pelo Tester/PO.
4. **Requisitos Não Funcionais (RNFs) Atendidos:**
   * **Performance (RNF):** O tempo de resposta de qualquer *endpoint* crítico (e.g., *check-in*) é **inferior a 2 segundos**.
   * **Segurança (RNF):** A autenticação está utilizando **criptografia** (*hashing* de senhas) e tokens de sessão válidos.
   * **Usabilidade (RNF):** O *wireframe* de baixa fidelidade foi seguido e a interface atende à consistência visual básica.
5. **Documentação Atualizada:** A documentação técnica relevante foi atualizada (se necessário).
6. **Product Owner Aceita a História:** O item foi demonstrado ao Product Owner e formalmente aceito.