



SPRINT REPORT — SPRINT 1

Equipe: Deivison Luiz, Francisco Lima, João Filipe Vinicius Neves, Márcio Vinicius, Ronilson Rocha e Thiago Moitinho

1. Informações Gerais

Nome do Projeto	Sombras do conhecimento
Período da Sprint 0	Um semana
Data de Início	17/04/25
Data de Término	01/05/25
Responsável pelo Preenchimento (Scrum Master ou outro)	Ronilson Rocha e João Filipe
Membros do Time	Deyvson Luiz, João Filipe, Francisco Lima, Márcio Vinicius, Ronilson, Thiago Moutinho e Vinicius

2. Objetivo Principal da Sprint 1

Descrição:	Criação do repositório no Github, capacitação das tecnologias back-end/ front-end e criação do designer inicial do App.
------------	--

3. Atividades-chave e resultados

(1) Definição da visão e objetivos do produto

Visão do Produto	Nosso aplicativo consiste em um jogo de perguntas e respostas voltado para a interpretação e teste de conhecimento em torno do livro “Mito da Caverna” de Platão, de uma forma intuitiva, mas que também abre espaço para que o usuário emita sua opinião.
Objetivos de negócio e indicadores (KPIs)	Nossos principais objetivos com esse projeto são: Teste de conhecimento sobre o livro. Promover o estudo da filosofia de uma forma mais lúdica. Criar um produto que possa ser usado em ambientes educacionais. Esses tópicos serão medidos através de feedbacks sobre o sistema.

Principais decisões tomadas	<p>Foi decidido inicialmente que o jogo terá um quiz de perguntas com quatro alternativas, onde a cada acerto o usuário avança de fase.</p> <p>Foi decidido também que o próprio sistema emitirá uma forma a qual o jogador tenha acesso ao livro, caso nunca tenha lido.</p> <p>Além disso, o jogo ter uma interface amigável e simples também foi algo debatido</p>
Pendências/Ações futuras	<p>Ainda existem algumas coisas que estão sendo pensadas, como:</p> <p>Como será organizado a base de perguntas?</p> <p>As respostas mudaram de lugar sempre que a fase for acessada?</p> <p>Quantas fases serão ao total?</p>

(2) Formação e alinhamento do time

Papéis e responsabilidades definidos	<p>Os papéis foram divididos e ficamos assim:</p> <p>Scrum Master: João Filipe</p> <p>Devs Front-End: João Filipe, Vinicius e Deyvson Luiz</p> <p>Devs Back-End: Márcio Vinicius, Thiago Moutinho e Lima</p> <p>Banco de Dados: Ronilson</p>
Working agreements (acordos internos)	<p>Foi combinado em sala que a duração de cada Sprint será de 2 semanas e usaremos a sala de aula como meio de encontro presencial e o Google Meet como forma de nos encontrarmos para debater online.</p> <p>Os dias para as reuniões serão às quintas em sala das 7:30 às 11h e ao decorrer da semana em dias e horários que fiquem mais apropriados para os membros.</p>
Nivelamento em Scrum (se necessário)	A ser realizado na próxima sprint
Pendências/Ações futuras	<p>Cada membro irá avaliar seu conhecimento na parte a qual foi designado e procurar formas de melhorar, caso necessário.</p> <p>De maneira geral, cada um tentará evoluir individualmente para a ajuda do coletivo, para que o projeto seja concluído de forma adequada.</p>

(3) Identificação e priorização do backlog

Backlog inicial criado	<p>Foi elaborado um backlog inicial contendo os principais épicos e histórias de usuário relacionadas à criação do software. Essas histórias estão registradas e serão detalhadas no Jira ao longo das próximas sprints. Abaixo, seguem alguns exemplos de histórias em formato misto para representar o início do nosso backlog:</p> <ul style="list-style-type: none">• Como jogador iniciante, quero criar uma conta e fazer login no sistema para salvar meu progresso no jogo.• Como usuário que nunca leu o livro, quero ter acesso ao texto “Mito da Caverna” para entender melhor o conteúdo do quiz.• Como estudante, quero responder perguntas de múltipla escolha para testar meu conhecimento sobre o “Mito da Caverna”.• Como jogador, quero receber um feedback visual a cada acerto ou erro para entender minha evolução no jogo.• Como professor, quero acompanhar o desempenho dos jogadores para avaliar o uso educacional do sistema. <p>Essas histórias estão ligadas à épicos como Autenticação de Usuário, Sistema de Perguntas e Respostas, Interface Interativa, e Gamificação com Feedback Visual.</p>
Técnica de priorização utilizada	<p>A priorização foi realizada com base no valor de negócio atribuído pelo Product Owner, levando em consideração a importância das funcionalidades para a entrega mínima viável do jogo. A técnica utilizada foi a decisão direta do PO, priorizando funcionalidades essenciais como autenticação, fluxo de perguntas e interface inicial.</p>
Estimativa iniciada	<p>Foi iniciado o processo de estimativas utilizando a técnica de Planning Poker, considerando complexidade, esforço técnico e interdependência de funcionalidades. Algumas histórias ainda precisam ser refinadas para estimativas mais precisas.</p>

Pendências/Ações futuras	Refinamento de histórias mais complexas; Validação das prioridades com stakeholders; Ajuste de estimativas após novas definições técnicas (ex.: API a ser utilizada no Frontend); Inclusão de critérios de aceitação detalhados.

(4) Planejamento de arquitetura e infraestrutura

Arquitetura definida	Front end- Monolito Modular Back end- MVC
Ferramentas de desenvolvimento/ versionamento	GitHub
Configuração de ambientes	Repositório com seção de Frontend e Backend.
Pendências/Ações futuras	Definir API a ser utilizada para fazer requisições HTTP no Frontend.

(5) Planejamento de qualidade e testes

Estratégia de testes	<p>Ferramentas definidas(FrontEnd):</p> <ul style="list-style-type: none"> ● Jest – motor principal de testes unitários e de integração. ● React Testing Library – para testes baseados na interação do usuário. ● Vitest - poderá ser avaliado futuramente como alternativa mais integrada ao Vite, se necessário.
Métricas de qualidade planejadas	<p>Métricas de Qualidade Planejadas</p> <p>Serão utilizadas as seguintes métricas para monitoramento e garantia de qualidade:</p> <ul style="list-style-type: none"> ● Cobertura de Testes: Gerada pelo jest --coverage, com relatórios em HTML. A

	<p>cobertura mínima esperada será de 80% para linhas e funções.</p> <ul style="list-style-type: none"> • Análise Estática de Código: Ferramentas como ESLint e Prettier já integradas ao projeto para garantir padronização e evitar más práticas. • Verificação de Tipagens: Utilização do TypeScript no projeto, com validação automática no build e via CI.
Pipeline de CI/CD inicial	<p>A integração contínua será configurada utilizando GitHub Actions, com os seguintes passos básicos:</p> <ol style="list-style-type: none"> 1. Instalação de dependências (<code>npm ci</code>) 2. Execução de testes unitários (<code>npm test</code>) 3. Geração de cobertura (<code>npm test -- --coverage</code>)
Pendências/Ações futuras	Garantir que todas as histórias de usuário tenham cenários de teste documentados e automatizados.

(6) Riscos e gestão de dependências

Riscos Identificados	Desafios tecnológicos: Integração do frontend (HTML/CSS/JS) com o banco de dados, principalmente no que diz respeito à autenticação de login e à busca dinâmica das perguntas do quiz.
Plano de mitigação	<p>Design do banco de dados: Planejar bem o esquema antes de começar a inserir dados. Fazer testes com poucos dados antes de escalar.</p> <p>Controle de versão e tarefas: Utilizar Git é alguma ferramenta de gerenciamento de tarefas (Jira) para manter organização e clareza do que falta fazer.</p>

Dependências externas	<p>Banco de Dados: Banco de dados utilizado para armazenar perguntas e respostas.</p> <p>Link para o PDF do livro: Certificar-se de que o link seja de uma fonte confiável.</p>
Pendências/Ações futuras	<p>Escolher e configurar o backend: Decidir se usará Python, Node.js ou outro e configurar o ambiente.</p> <p>Modelar o banco de dados: Criar estrutura de tabelas (usuários, perguntas, respostas, feedback).</p> <p>Implementar sistema de login com segurança mínima.</p> <p>Definir perguntas por nível de dificuldade.</p> <p>Testar integração entre front e back antes de inserir muitos dados.</p> <p>Implementar botão de feedback e definir onde ele será armazenado (banco, e-mail, etc.).</p>

(7) Planejamento de lançamentos e *roadmap* macro

Visão geral das 5 Sprints Finais (Planejamento Macro)	<p>- Sprint 1: Implementação da tela de login e autenticação de usuário - Sprint 2: Criação do banco de dados com perguntas e respostas - Sprint 3: Desenvolvimento da interface do quiz e lógica de múltipla escolha - Sprint 4: Sistema de pontuação e níveis de dificuldade - Sprint 5: Testes, correções e publicação da versão final</p>
Marcos (milestones) importantes	<p>Finalização da tela de login, Banco de dados completo com 15 perguntas, Demo funcional para feedback interno, Validação do sistema por usuários teste.</p> <p>Datas serão definidas nas atualizações posteriores.</p>
Alinhamento com stakeholders	<p>- Reunião inicial em 08/04 para aprovação do escopo</p> <p>- Feedback de professor/orientador sobre nível das</p>

	perguntas em 24/04 - Acompanhamento semanal com equipe de desenvolvimento
Pendências/Ações futuras	- Revisar conteúdo das perguntas com especialista em filosofia - Ajustar responsividade para dispositivos móveis - Adicionar sistema de dicas ou ajuda para perguntas difíceis

(8) Critérios de pronto (Definition of Done - DoD) e critérios de aceite

Definition of Done (DoD)	<ul style="list-style-type: none"> ● Código implementado, testado e revisado por outro membro do time. ● Cobertura de testes igual ou superior a 80%, conforme métricas de qualidade planejadas. ● Código aprovado na branch principal via Pull Request. ● Testes unitários automatizados com Jest e/ou React Testing Library executados com sucesso. ● Documentação atualizada (código comentado, Wiki ou README, se aplicável). ● Deploy funcional no ambiente de testes/homologação. ● Validação do PO com base nos critérios de aceite
Critérios de aceitação (Gerais) adotados	<p>Cada história de usuário deve:</p> <ul style="list-style-type: none"> ● Cumprir integralmente o que está descrito na descrição da história. ● Apresentar comportamento esperado em casos normais e de exceção. ● Estar integrada à aplicação de forma funcional. ● Ter feedback positivo do PO e/ou stakeholder envolvido.

	<ul style="list-style-type: none"> Ter evidências visíveis (exemplo: prints, vídeos, protótipos, se solicitado).
Pendências/Ações futuras	<ul style="list-style-type: none"> Refinar os critérios de pronto após o primeiro ciclo completo de desenvolvimento. Alinhar critérios de aceite específicos por história com o PO nas reuniões de planning. Incorporar feedbacks de testes futuros ao DoD.

(9) Prototipagem ou Prova de Conceito - PoC (se aplicável)

Objetivo do protótipo/PoC	Ainda não foi criado um protótipo. A criação do protótipo visual do sistema será realizada nas próximas sprints para validar o fluxo de experiência do usuário (UX), especialmente no que diz respeito à navegação entre as telas de login, perguntas e sistema de pontuação.
Resultados obtidos	Não aplicável, pois o protótipo ainda não foi desenvolvido.
Pendências/Ações futuras	<p>Desenvolvimento do protótipo visual para validação da interface do jogo e fluxo de interação.</p> <p>A prototipagem será feita com base nas histórias de usuário e feedbacks iniciais da equipe, com foco na experiência do usuário.</p>

(10) Organização de ferramentas e fluxos de trabalho

Ferramenta de gestão de projetos	Jira para gestão de projeto com sprints e backlogs.
Repositório de código	GitHub: https://github.com/ronilsonrocha/sombras_do_conhecimento.git
Documentação e <i>Wiki</i>	Toda a documentação será feita pelo GitHub.

Pendências/Ações futuras	Ajustar a arquitetura para se adaptar a uma futura expansão.

4. Conclusão e *status* final da *Sprint 1*

O que foi concluído com sucesso?	Treinamento com as ferramentas, definição dos cargos, configuração do Jira e GitHub.
Desvios ou itens não finalizados	Cumprimos com o projetado.
Lições aprendidas	Aprendemos novas ferramentas e funcionalidades que levaremos para a carreira de Devs. Poderia ter sido melhor, mas devido a vários feriados e alguns membros morarem em outras cidades, eles aproveitaram para ficar com a família.
Próximos passos e preparação para a <i>Sprint 2</i>	Na Sprint 2 de fato começaremos a desenvolver o App. Pois o time já está alinhado com as ferramentas e com seus cargos.

5. Aprovações e revisão

- ***Product Owner:***

Ronilson Rocha Santos

- ***Scrum Master:***

João felipe

- **Data de Revisão e Aprovação:**

01/05/2025

Anexos (se forem necessários)

- **Documentos referenciados:**

(Links ou anexos de Planos de Teste, Arquitetura, Roadmap detalhado etc.)

- **Quadro de riscos detalhado:**

(Se houver registro mais extenso de riscos e plano de mitigação.)

- **Lista detalhada do *backlog*:**

(Para casos em que haja muitas histórias de usuário, pode-se anexar ou referenciar uma planilha/breakdown completo.)

Observações finais:

1. A **Sprint** 1 foi criada. A equipe definiu os marcos iniciais (repositório, capacitação e design) como pré-requisitos para a implementação das funcionalidades prioritárias nas próximas sprints.