



Disciplina:	Projeto Integrado 1	Semestre:	4º
Professor:	Jeferson Kenedy		
Alunos:	Elenildo de Vasconcelos Kairo Matheus Sales Barbosa Luís Estevam Rosa Chaves Luis Fernando Batista Lima	Matrícula:	508287 539328 536699 538134

Documento de Visão e Escopo plataforma QXCode

Introdução

O presente documento contém dados de visão e escopo sobre o projeto QXcode, em progresso por alunos da UFC - *Campus* Quixadá. Ademais, as especificações detalhadas do sistema em construção estarão divididas no presente documento nas seguintes sessões:

- 1. Visão;**
- 2. Escopo;**
- 3. Stakeholders;**
- 4. Processos, Tecnologias e Ferramentas;**
- 5. Equipe do Projeto.**

1. Visão

Esta seção apresenta a problemática de modo sucinto e eficaz, elucidando o propósito e especificando o contexto de surgimento da idealização do projeto, fornecendo uma visualização de alto nível das capacidades do sistema em pauta.

1.1 Instrução do Problema/Contexto

Este projeto surge como resposta à crescente demanda por programas que possam oferecer suporte na resolução de questões relacionadas à programação e linguagens de programação. Em 2023, a Olimpíada Brasileira de Informática (OBI) atingiu um marco notável, com mais de 100.000 alunos inscritos para a competição. A OBI é uma competição de programação altamente competitiva destinada a alunos do ensino médio e estudantes que estão no primeiro semestre de sua graduação.

Além disso, há também a Maratona de Programação, uma competição que abrange várias etapas e apresenta um nível de dificuldade substancialmente maior, demandando um treinamento intensivo por parte dos competidores. Um software destinado a auxiliar programadores é fundamental para o desenvolvimento constante no campo da programação, pois empresas de renome, como Google e Microsoft, demandam proficiência em uma ampla gama de algoritmos durante seus processos de seleção.

Para atender a essa necessidade crescente, é imprescindível a criação de uma plataforma que ofereça conteúdo de alta qualidade em língua portuguesa, além de disponibilizar a opção de exercícios offline.

1.2 Propósito

O propósito deste projeto é promover a prática da programação tanto para programadores intermediários e experientes, proporcionando desafios complexos que são valiosos em cenários como competições, entrevistas de emprego e olimpíadas, quanto para iniciantes e entusiastas que desejam iniciar sua jornada na programação. Através do nosso software, almejamos oferecer aos usuários acesso a uma ampla variedade de problemas de programação, abrangendo diferentes níveis de complexidade e passíveis de resolução nas linguagens mais amplamente utilizadas no

mercado. Além disso, pretendemos fornecer recursos que permitam aos usuários monitorar seu progresso em cada questão e esclarecer dúvidas relacionadas às soluções propostas.

1.3 Resumo das capacidades

Os usuários terão acesso a um amplo banco de questões, organizadas por categorias de dificuldade estimada com base no nível de conhecimento necessário para resolvê-las. Além disso, terão acesso a dicas e soluções para cada questão. Para acompanhar seu progresso e manter um ambiente de estudo dinâmico, os usuários poderão visualizar um histórico de todas as suas tentativas anteriores. Os administradores terão a capacidade de adicionar, editar e excluir questões.

2. Escopo

2.1 Resumo do projeto

Um software de prática de programação com um banco de questões, o software seria capaz de armazenar e auxiliar na resolução de diversos problemas envolvendo programação, que serviria tanto para programadores mais experientes treinarem para entrevistas, competições e olimpíadas, quanto para programadores iniciantes praticarem e solidificarem a base de seus conhecimentos.

2.2 Objetivos:

- 1. Conteúdo Multimídia:** Oferecer conteúdo diversificado, incluindo vídeos explicativos, tutoriais interativos e exemplos de código prontos para uso.
- 2. Personalização de Experiência:** Permitir que os usuários personalizem suas experiências, escolhendo desafios com base em seu nível de habilidade, interesses e metas de aprendizado.
- 3. Relatórios de Progresso:** Fornecer relatórios detalhados de progresso para que os usuários possam acompanhar seu desenvolvimento ao longo do tempo e identificar áreas que precisam de mais atenção.

4. **Acesso Offline Aprimorado:** Melhorar a capacidade de os usuários acessarem conteúdo e realizar exercícios offline, tornando o aprendizado mais flexível.
5. **Parcerias Educacionais:** Colaborar com instituições de ensino e organizações educacionais para integrar o software em currículos de programação e fornecer recursos educacionais adicionais.
6. **Feedback Aperfeiçoamento Contínuo:** Coletar feedback dos usuários para aprimorar constantemente o conteúdo, a usabilidade e a qualidade geral da plataforma.

2.3 Principais funcionalidades:

1. Lista de questões divididas por nível de complexidade..
2. Capacidade de edição e execução códigos dentro da interface.
3. Capacidade de avaliar a resposta do usuário baseado em casos de testes previamente escritos.
4. Registrar e exibir o histórico de cada questão realizada pelo usuário.
5. capacidade de armazenar links que direcionam para conteúdos relevantes à questão selecionada.

2.4 Limitações e exclusões:

1. **Linguagens Específicas:** Embora o software ofereça suporte a várias linguagens de programação populares, algumas linguagens menos comuns podem não estar disponíveis, o que pode limitar a escolha dos usuários.
2. **Hardware e Sistema Operacional:** O software pode ser projetado para funcionar em sistemas operacionais específicos ou com requisitos de hardware mínimos. Mesmo assim, isso pode excluir usuários com sistemas diferentes.

3. Stakeholders

- **Estudantes de Programação :** programadores especializados em competições que envolvem resolver problemas de programação complexos em um ambiente de corrida

contra o tempo, geralmente usando linguagens de programação como C++, Java ou Python.

3.1 Personas

- **Andrei Piovezan**

Idade: 17 anos

Ocupação: Aluno novato no curso de Ciência da computação

Desafios: Equilibrar suas responsabilidades acadêmicas, morar sozinho e aprimorar seus conhecimentos em programação.

Descrição: Andrei está focado em aprimorar seus conhecimentos em programação, uma habilidade essencial para sua carreira futura. Ele reconhece a importância da prática constante e enfrenta desafios de programação para melhorar suas habilidades. No entanto, ele também entende que, devido à sua agenda lotada, precisa de um software que lhe permita praticar programação de forma conveniente e que possa ser acessado sem a necessidade de uma conexão de internet constante.

Necessidades: Um software que possa auxiliar na prática de problemas envolvendo programação e que possa ser acessado sem uma conexão de internet.

- **Marcelo Cespedes**

Idade: 30 anos

Ocupação: Engenheiro sênior de uma Multinacional

Desafios: Gerenciar sua carreira enquanto treina programação competitiva e resolução de problemas de programação.

Descrição: Marcelo Cespedes é um engenheiro sênior com 30 anos de idade, trabalhando em uma posição de destaque em uma empresa multinacional. Ele tem uma carreira de sucesso na engenharia, mas recentemente desenvolveu uma paixão por programação competitiva e resolução de problemas de programação.

Apesar de suas responsabilidades profissionais desafiadoras, Marcelo está determinado a se destacar na programação competitiva e deseja aprimorar suas habilidades de resolução de problemas em programação. Ele entende que, para alcançar sucesso nesse campo, precisa de uma prática consistente e de exposição a problemas complexos.

As competições de programação exigem pensamento rápido, criatividade e soluções eficientes para desafios algorítmicos, e Marcelo está comprometido em atingir um alto nível nessa área.

Necessidades: Uma plataforma que o auxilie a manter uma prática consistente em problemas complexos envolvendo programação.

4. Processos, Tecnologias e Ferramentas

4.1 Processo

As metodologias utilizadas serão Scrum, uma metodologia incremental baseada em intervalos de tempo de desenvolvimento chamados de sprints, as sprints terão duração de uma semana e durante o projeto serão realizadas dailys, que são reuniões curtas (15 minutos) que acontecem diariamente, para monitoramento do progresso do projeto. O quadro Kanban é uma técnica de gerenciamento de responsabilidades e progresso utilizando cartões em um quadro. Essas normas foram escolhidas pela sua eficiência e podem melhor ajudar na organização e acompanhamento do projeto, além disso as duas metodologias ágeis selecionadas são incrementais e faz com que se tenha resultados parciais que podem receber feedbacks e ajudar na melhoria contínua, além disso entregas frequentes e incrementais cria um senso de progresso e realização na equipe sendo uma ótima motivação. Outro grande ponto positivo é que essas metodologias promovem uma colaboração ativa dos membros da equipe, isso também é resultado da comunicação clara e a definição de papéis e responsabilidades, ajudando a evitar conflitos.

4.2 Tecnologias

Serão utilizadas as tecnologias JavaFX para construir as partes interativas e estrutura da aplicação, CSS como tecnologia de design para definir o estilo visual da aplicação. O JavaFX foi o escolhido pois conta com diversas ferramentas para a criação da parte gráfica, além de ser uma biblioteca com uma base sólida e já estar consolidado no mercado, o CSS chega para estilizar a interface. O PostgreSQL é um banco de dados relacionais, que tem uma alta compatibilidade com diversos padrões de linguagem, e conta com praticidade, sua praticidade faz com que ele seja ideal para o armazenamento de dados do sistema.

4.3 Ferramentas

Usaremos como ferramenta para organização o Trello, uma ferramenta visual que possibilita ao time o gerenciamento de qualquer tipo de projeto, fluxo de trabalho ou monitoramento de tarefas. Figma foi a ferramenta escolhida para criar uma representação que simula o funcionamento do futuro software, também foi escolhido o Balsamiq, uma plataforma de prototipação de baixa fidelidade, que serve para criar a primeira versão do protótipo. SceneBuilder é uma ferramenta para desenvolver aplicações com aparências modernas, isso sendo utilizado junto com o CSS e JavaFX serão os responsáveis pela criação de toda a interface do sistema. O Git é um sistema de controle de versão amplamente utilizado na gestão de projetos de desenvolvimento de software e outras colaborações, outro grande benefício é os backups para não se ter nenhuma grande perda do que já foi feito, seu alto poder de gerenciamento de ramificações é outro ponto que foi levado em consideração para a escolha do GIT. Notion é uma plataforma de produtividade multifuncional que oferece uma abordagem inovadora para a organização, colaboração e gestão de projetos, será uma peça fundamental para a utilização da metodologia Scrum para a documentação dos artefatos e atividades a serem produzidas e também monitorar o progresso de cada participante da equipe do projeto.

5. Equipe do Projeto

- **Tech Lead - Luís Estevam Rosa Chaves**
 - É uma posição dentro da equipe de desenvolvimento que tem a função de gerenciar e conduzir a parte técnica do projeto. É quem define as tecnologias e a arquitetura do produto.
- **Product Owner - Elenildo de Vasconcelos**
 - Ele é responsável por manter a integridade conceitual das novas funcionalidades, bugs ou melhorias, para que essas sigam uma visão definida para o produto ou projeto.
- **Development Team - Kairo Matheus Sales Barbosa e Luis Fernando Batista Lima**
 - O Development Team é a equipe de profissionais responsável por transformar o Product Backlog em um produto funcional. São eles que desenvolvem as versões incrementais do produto “Pronto” que são entregues ao final de cada Sprint.