



UNIVERSIDADE DE FORTALEZA

Curso: Ciência da Computação

Turma: T197-12

Disciplina: Desenvolvimento de Plataformas Móveis

Aluno(s): Davi Klein, Pedro Lucca, Josué Castro

Turno: Manhã

Data:09/09/24

Período Letivo: Terceiro Semestre

TRABALHO

DUOPROG

FORTALEZA

2024

SUMÁRIO

<i>VISÃO GERAL</i>	3
Introdução	3
Objetivo	3
Justificativa	3
Benchmark	3
<i>METODOLOGIA PARA O DESENVOLVIMENTO DE SOFTWARE</i>	4
Descrição da Metodologia	4
Processo da Metodologia	4
<i>ARTEFATOS DO PRODUTO</i>	5
Atores	5
Requisitos Funcionais	5
Requisitos Não-funcionais	5
Protótipo de Baixa Fidelidade	6
Diagrama de Caso de Uso	7

VISÃO GERAL

Introdução

O trabalho refere-se a um aplicativo voltado ao ensino de linguagens de programação de forma simplificada, prática, intuitiva e eficiente, contando com funcionalidades que entregam entretenimento e conhecimento.

Objetivo

O objetivo do desenvolvimento desse trabalho é ensinar, educar e popularizar a prática e conhecimento da programação ao mundo de hoje, que cada vez mais vem evoluindo tecnologicamente, assim, fazendo com que a necessidade do conhecimento básico para manuseio de uma máquina cresça cada vez mais.

Justificativa

O desenvolvimento deste trabalho pode ser justificado a partir do ponto em que, o mundo hoje evolui, no geral, muito mais rápido do que antes, as informações são trocadas e adquiridas de forma muito mais ágil, porém, a área que mais se expandiu exponencialmente nesses últimos tempos foi justamente a área da tecnologia, que tanto se prejudica pela falta de profissionais qualificados. É a partir desta problemática que surge a ideia de trazer conhecimento ao máximo de público possível para que, assim, possamos evoluir e educar a sociedade tecnologicamente para o futuro.

Benchmark

	Duolingo	SoloLearn	Codecademy
UX/UI	X	X	X
Progreção baseada em níveis	X	X	X
Feedback em tempo real	X	X	X
Suporte a várias linguagens de programação		X	X

Diversidade de exercícios e métodos de ensino	X	X	X
Acesso offline e multiplataforma			

METODOLOGIA PARA O DESENVOLVIMENTO DE SOFTWARE

Descrição da Metodologia

Kanban

A metodologia para realização do projeto escolhida pelo grupo será Kanban, que se trata, basicamente, de uma forma de abordagem simples, eficiente e direta para a administração de demandas. Pode ser definido pelos seguintes itens:

- Sistema visual: um processo, definido em um quadro com colunas de separação, que permite dividir o trabalho em segmentos ou pelo seu status. Ele fixa cada item em um cartão e coloca em uma coluna apropriada para indicar onde ele está em todo o fluxo de trabalho;
- Cartões: descrevem o trabalho real que transita por este processo;
- Limitação do trabalho em andamento: que permite atribuir os limites de quantos itens podem estar em andamento em cada segmento ou estado do fluxo de trabalho

Processo da Metodologia

O **Kanban** é uma metodologia ágil que visa melhorar o fluxo de trabalho e a eficiência da equipe por meio de visualização e controle de tarefas. O processo inclui:

1. **Visualizar o fluxo de trabalho:** Usar um quadro Kanban com colunas representando as etapas do processo.
 2. **Limitar o Trabalho em Progresso (WIP):** Definir limites para o número de tarefas em andamento, evitando sobrecarga.
 3. **Gerenciar o fluxo:** Garantir que as tarefas fluam eficientemente pelo processo e identificar gargalos.
 4. **Tornar políticas explícitas:** Definir regras claras para o fluxo de trabalho.
 5. **Melhorar continuamente:** Revisar o processo periodicamente para otimizar o desempenho.
 6. **Obter feedback rápido:** Realizar reuniões curtas para ajustar o andamento das tarefas. O objetivo é aumentar a eficiência e a entrega contínua.
- Para a realização das entregas serão feitas reuniões semanais entre os membros da equipe através da plataforma Discord para discutir necessidades, o que fazer e o que incluir.



ARTEFATOS DO PRODUTO

Atores

Os atores presente no nosso projeto serão os próprios 3 desenvolvedores no aplicativo (administradores) e contaremos com um dos administradores para a função de moderador do aplicativo, enquanto o terceiro ator seria o próprio usuário

Requisitos Funcionais

Cod	Requisito	Prioridade
RF01	Capacidade de cadastro do usuário	1
RF02	Capacidade de login do usuário	
RF03	O sistema deve notificar o usuário sobre novos eventos ou atualizações através de push notifications.	2
RF04	O sistema deve possuir uma barra de pesquisa que retorne resultados relevantes baseados em palavras-chave.	2
RF05	O sistema deve contar com uma base de feedback	2
RF06	O sistema deve contar com um sistema de recompensas com base no progresso do usuário	3
RF07	O aplicativo possui um banco de questões ligadas a um banco de dados	1
RF08	O aplicativo possui questões divididas por níveis (fácil, médio e difícil), que poderá ser definida pelo usuário de acordo com sua preferência.	2

RF09	O aplicativo possui 3 módulos de questões sendo divididos em questões de Java, C e Python.	2
RF10	O usuário poderá escolher qual modulo deseja realizar	2
RF11	O aplicativo deve contar com um sistema de notificações	2
RF12	O aplicativo deve contar com módulos de revisão ao fim dos níveis de dificuldade dos exercícios	3

RF13	O sistema deve contar com a permissão de acesso a lições e exercícios de modo offline com sincronização automática quando o usuário estiver online novamente	2
RF14	O sistema deve contar com um algoritmo de recomendação de outros tipos de linguagem baseadas nos gostos do perfil do usuário	3
RF15	O sistema contará com um teste de nível para saber onde o usuário se “encaixa” na jornada de aprendizado	3
RF16	O sistema terá a capacidade de funcionar em segundo plano	3
RF17	O sistema contará com a capacidade de cadastro de administradores	1
RF18	O sistema contará com a capacidade de login de administradores	1
RF19	O sistema contará com uma área dedicada à administração do aplicativo	1
RF20	O sistema contará com integração à internet	3
RF21	Os administradores do sistema poderão excluir contas de usuário	2
RF22	Os administradores do sistema poderão adicionar perguntas ao aplicativo	2
RF23	Os administradores do sistema poderão remover perguntas do aplicativo	2
RF24	Os administradores do sistema poderão alterar perguntas no aplicativo	2

Requisitos Não-funcionais

Cod	Requisito	Prioridade
RNF01	O sistema deve ser intuitivo e fácil de usar, com uma curva de aprendizado mínima para novos usuários	1
RNF02	O sistema deve ser acessível para usuários com deficiência, conforme as diretrizes WCAG (Web Content Accessibility Guidelines)	2
RNF03	O sistema deve permitir a adição de novos módulos ou funcionalidades sem necessidade de reescrever a arquitetura existente	3
RNF04	O sistema deve suportar diferentes resoluções sem perda de qualidade e/ou desempenho	3

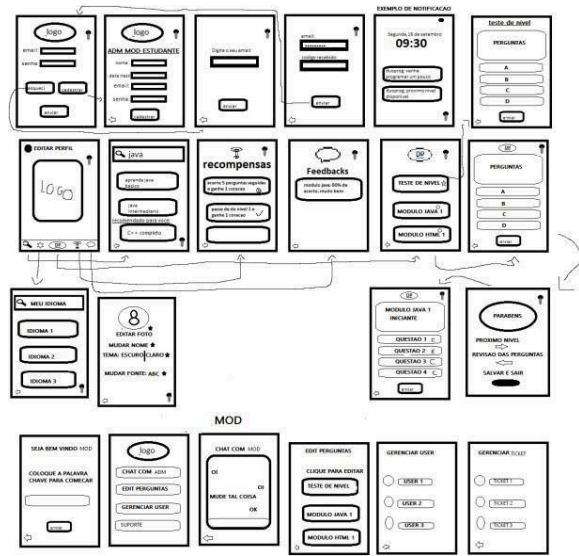
RNF05	O sistema deve ser compatível com 1 ou mais tipo de plataforma	3
RNF06	O código deve ser facilitado para possíveis manutenções ou futuras atualizações	3
RNF07	O sistema deve ser regulamentado de acordo com a proteção de dados de usuários	2
RNF08	O aplicativo deve oferecer suporte para idiomas	3
RNF09	O sistema deve incluir um sistema de salvamento de progresso do usuário	2
RNF10	O código fonte do aplicativo deverá ser de fácil acesso ao administrador	3
RNF11	O sistema deve ser otimizado para não consumir mais do que 5% de bateria por hora de uso	3
RNF12	O aplicativo deve funcionar sem queda de desempenho em dispositivos com, no mínimo, 2 gb de ram	3

RNF13	O código fonte do aplicativo deve ser de fácil manuseio para futuras atualizações	2
RNF14	O aplicativo contará com a possibilidade de algumas personalizações para melhorar a UX	3
RNF15	O acesso à área administrativa deve ser protegida por autenticação	1
RNF16	O sistema deve ter alta disponibilidade, garantindo que os administradores possam acessar as ferramentas de gerenciamento facilmente	2
RNF17	A interface de administração deve ser intuitiva e fácil de usar	2
RNF18	O sistema deve ser modular para permitir a fácil atualização ou substituição elementos	2
RNF19	O sistema deve garantir a proteção de dados de usuários, administradores	2
RNF20	O sistema deve ser capaz de passar por atualizações	2

Classificação para Prioridade dos requisitos:

- 1 - (Deve ter): São os requisitos essenciais e de alta prioridade, considerados fundamentais para o sucesso do sistema. São os requisitos que devem ser implementados obrigatoriamente, pois sem eles o sistema não atenderia às necessidades básicas dos usuários finais ou não cumpriria os objetivos do projeto.
- 2 - (Deveria ter): São os requisitos importantes, mas não tão críticos quanto os de prioridade 1. Esses requisitos devem ser implementados se possível, mas podem ser adiados caso haja restrições de tempo ou recursos.
- 3 - (Poderia ter): São os requisitos desejáveis, mas não essenciais. São requisitos que podem trazer benefícios adicionais ao sistema, mas sua ausência não afetaria significativamente a funcionalidade principal.

Protótipo de Baixa Fidelidade



ADMS

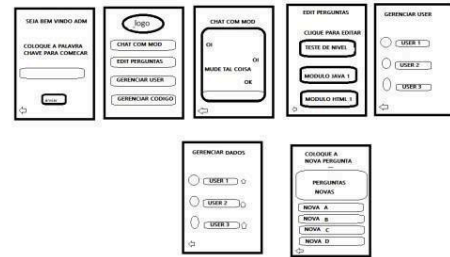


Diagrama de Caso de Uso

