

Documentação do Modelo de Processo de Software e Implementação Ágil

Referências:

IEEE Std. 830 – 1993. IEEE Recommended Practice for Software Requirements Specifications.

Filho, W.P.P. Engenharia de Software: Fundamentos, Métodos e Padrões. LTC: Rio de Janeiro, 2001.

Integrantes:

Arthur Souza Lima (RA: 12723112788)

Aurea Reis (RA: 1272313156)

Alfredo Ruas Neto (RA: 1272326487)

Daniel Santos Lopes (RA: 12723130024)

Enrique Silva dos Reis (RA: 12724149018)

Elvis Oliveira dos Reis (RA: 1272318921)

Edioelson Júnior A. B. Teixeira (RA: 1272318423)

Gabriel Silva Magalhães (RA: 1272313274)

Gabriel Moreira de Oliveira (RA: 1272311981)

Hanspeter Dietiker (RA: 1272313332)

Natan Oliveira da Silva (RA: 12723211400)

Introdução

➤ Objetivo

- O projeto consiste na criação de um site para feedback de restaurantes, com o propósito de fornecer aos usuários uma intuitiva plataforma de estabelecimentos culinários. Com isto em mente, o modelo de processo de software escolhido para o projeto foi o de prototipação junto da metodologia de desenvolvimento ágil Kanban para mapear as etapas do projeto e permitir uma adaptação mais rápida para atender as necessidades do usuário.

Modelo de Processo de Software de Prototipação

➤ Justificativa

- O modelo de Prototipação foi escolhido porque ele permite a visualização e testagem do software antes da entrega final.

➤ Fases:

- Configuração de Ambiente e Coleta para Levantamento de Requisitos;
- Modelagem do Projeto;
- Desenvolvimento;
- Implementação, Entrega e Feedback;

Plano Detalhado Para Cada Fase

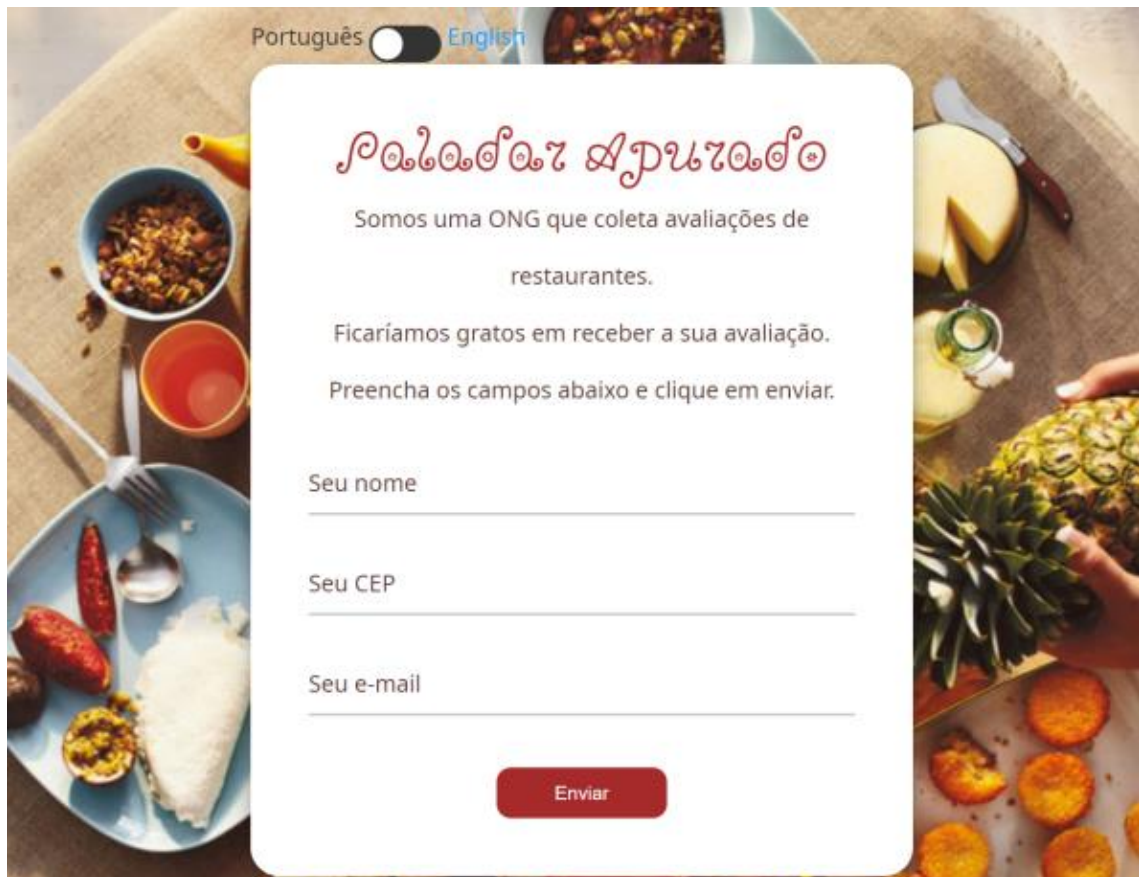
➤ Configuração de Ambiente e Coleta para Levantamento de Requisitos

- Esta fase consiste em configurar todo o ambiente de desenvolvimento no GitHub e Git Actions além de levantar os requisitos funcionais e não funcionais do projeto e realizar toda a documentação necessária com o intuito de agilizar o desenvolvimento e evitar problemas de lógica.

➤ Modelagem do Projeto

- Esta fase consiste nas elaborações de modelagens de software com a criação de diagramas de classe e de casos de uso e na criação antecipada de um protótipo do projeto.

➤ Protótipo Inicial do Projeto



Português ☒ English

Salvo e Apuro

Somos uma ONG que coleta avaliações de restaurantes.

Ficariamos gratos em receber a sua avaliação.

Preencha os campos abaixo e clique em enviar.

Seu nome

Seu CEP

Seu e-mail

➤ Desenvolvimento

- Fase com maior duração pois é dedicada para o desenvolvimento geral da aplicação tanto do Front end com as interfaces criadas com HTML, CSS e JavaScript quanto do Back end com toda a lógica desenvolvida em Java e a conexão do banco de dados PostgreSQL.
- Nesta também deve ser realizada paralelo ao desenvolvimento os testes do software necessários para otimização e minimização de bugs do código.

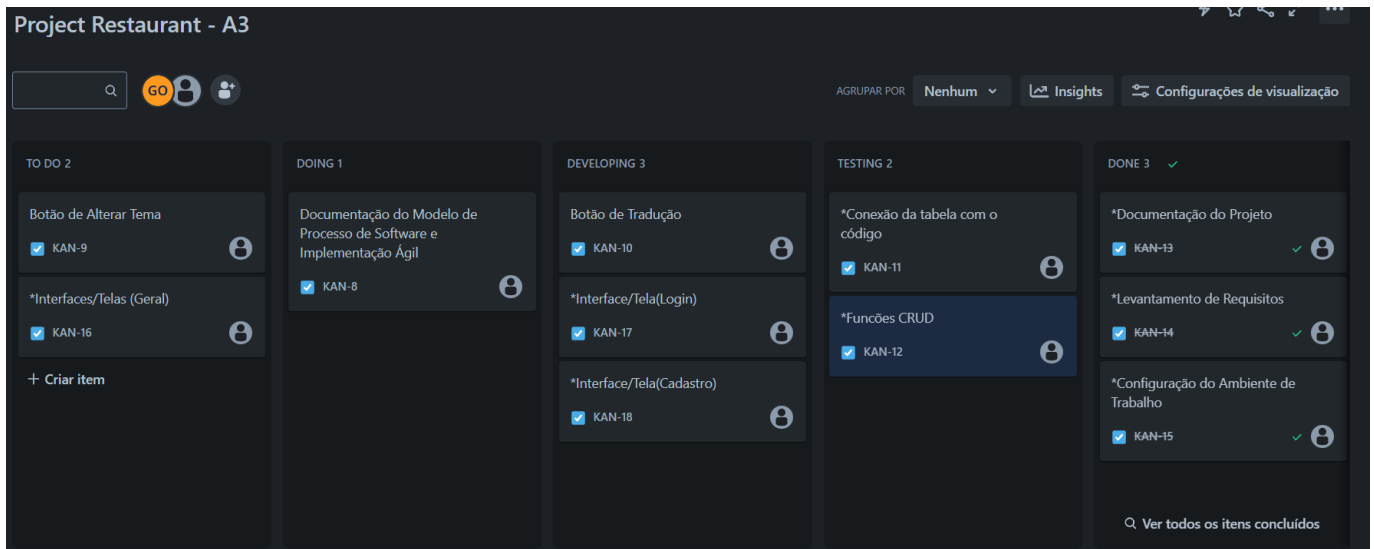
➤ Implementação, Entrega e Feedback

- Fase final que tem o intuito de Colocar o projeto no ar e obter o feedback necessário para futuras melhorias do projeto.
- Plano detalhado para cada fase do modelo de processo, destacando as atividades a serem realizadas e os marcos a serem alcançados.

Metodologia Ágil Kanban

➤ Justificativa

- A metodologia Kanban foi selecionada para um melhor gerenciamento do fluxo de trabalho, utilizando 5 colunas: To-do, Doing, Developing, Testing e Done. Cada uma para representar uma fase do processo de desenvolvimento.



➤ TO DO

- Tarefas que ainda irão ser realizadas como futuras atualizações ou funções principais que ainda não foram iniciadas.

➤ DOING

- Tarefas que estão sendo realizadas no momento e que não são necessariamente de desenvolvimento de código.

➤ DEVELOPING

- Tarefas de desenvolvimento de código que estão sendo realizadas no momento.

➤ TESTING

- Códigos já desenvolvidos que estão sendo testados para otimização e redução de bugs.

➤ DONE

- Tarefas já realizadas por completo.

➤ Backlog

- Backlog não foi criado como uma lista específica de prioridades mas implementado nas colunas do Kanban com um asterisco (*) para na mesma tela de atividades poder ser visualizado as que possuem maior prioridade.

➤ Sprints

- Foram criadas 4 sprints para o projeto, todas com um intervalo de 3 a 4 semanas:
 - 1º Sprint (Entrega 10/04/24) - Configuração do ambiente de desenvolvimento e documentação do escopo do projeto.
 - 2º Sprint (Entrega 24/04/24) - Documentação do Modelo de Processo de Software e Implementação Ágil.
 - 3º Sprint (Entrega 22/05/24) - Protótipos e código-fonte do front-end.
 - 4º Sprint (Entrega 13 à 19/06/24) - Código-fonte da lógica de negócio.

➤ Rituais Ágeis

- Reuniões realizadas presenciais nos dias de aula para organizar as funções do projeto (parte teórica) e semanalmente de forma online para desenvolver todas as funcionalidades, requisitos e correções (parte prática).

Configuração das Ferramentas de Gestão de Projetos

➤ Ferramenta de Gestão

- A principal ferramenta de gestão do projeto foi o Jira, configurado para ser possível a visualização de colunas seguindo a metodologia Kanban, com o fim de ter um fluxo de trabalho mais otimizado, além de poder ter conexão com o GitHub para fácil acesso ao repositório do programa.

Informações de Suporte

Documentação Spring Boot: <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>

Repositório do Projeto: <https://github.com/hanspeterdietiker/A3-Project-Unifacs>

Link do Gestor do Projeto:
<https://hanspeter.atlassian.net/jira/software/projects/KAN/boards/1>