

INSTITUTO FEDERAL DO CEARÁ – IFCE *Campus* TIANGUÁ
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO - 5º SEMESTRE - 2024.2

JOÃO PEDRO ARAÚJO ARRUDA

JOÃO RAYAN GOMES FERNANDES

LEYSSA EMANUELLA ALVES DE SOUSA

MARIA EULÁIA DOS SANTOS ARAÚJO

ENGENHARIA DE SOFTWARE

REQUISITOS DE SOFTWARE

Tianguá - CE

Dezembro de 2024

JOÃO PEDRO ARAÚJO ARRUDA

JOÃO RAYAN GOMES FERNANDES

LEYSSA EMANUELLA ALVES DE SOUSA

MARIA EULÁIA DOS SANTOS ARAÚJO

ENGENHARIA DE SOFTWARE

REQUISITOS DE SOFTWARE

Atividade proposta no contexto do curso de Ciência da Computação no Instituto Federal de Educação, Ciência e Tecnologia do Ceará (IFCE) - Campus Tianguá, referente à disciplina de Engenharia de Software.

Prof: Cynthia Pinheiro Santiago.

Tianguá -CE

Dezembro de 2024

Histórico de Versões

Data	Versão	Descrição	Autor	Revisor	Aprovado por
20/12/2024	1.0	Versão inicial	João Pedro João Rayan Leyssa Emanuella Maria Euláia	João Pedro João Rayan Leyssa Emanuella Maria Euláia	João Pedro João Rayan Leyssa Emanuella Maria Euláia
20/12/2024	1.1	Realizando ajustes no texto	João Pedro João Rayan Leyssa Emanuella Maria Euláia	João Pedro João Rayan Leyssa Emanuella Maria Euláia	João Pedro João Rayan Leyssa Emanuella Maria Euláia

1. Objetivo

Esse glossário tem como objetivo facilitar o entendimento dos principais termos técnicos utilizados no **Sistema de Gerenciamento de Estacionamento**..

2. Termos Utilizados

2.1. Caso de Uso (Use Case):

Descrição de uma interação entre o sistema e os usuários, com o objetivo de atingir um objetivo específico, como salvar dados ou exibir um resumo.

2.2. Arquivo JSON (JavaScript Object Notation):

Formato de troca de dados amplamente utilizado para armazenar e transportar informações em sistemas baseados em texto. No contexto do sistema de estacionamento, os dados das vagas ocupadas e informações de veículos são armazenados e carregados neste formato

2.3. Arquivo TXT (Text File):

Formato de arquivo simples que contém dados em texto legível. No sistema de estacionamento, o arquivo TXT é utilizado para salvar informações de forma legível e não estruturada.

2.4. Fluxo Principal (Main Flow):

Sequência de etapas que o sistema segue quando o usuário realiza uma ação esperada, sem interrupções ou erros.

2.5. Fluxo Alternativo (Alternative Flow):

Caminho alternativo dentro do caso de uso que ocorre quando há uma condição diferente da esperada, como um erro ou uma escolha diferente.

2.6. Fluxo de Exceção (Exception Flow):

Fluxo que lida com situações de erro inesperado ou falhas, permitindo que o sistema tome ações como exibir mensagens de erro ou realizar tentativas de recuperação.

2.7. Pré-condições (Pre-conditions):

Condições que devem ser atendidas antes da execução de um caso de uso. Por exemplo: "O arquivo JSON deve existir e estar em formato válido" para carregar os dados.

2.8. Pós-condições (Post-conditions):

Condições que devem ser verdadeiras após a execução de um caso de uso. Exemplo: "Os dados do estacionamento são salvos corretamente em formatos JSON e TXT."

2.9. Requisitos Não Funcionais (Non-Functional Requirements):

Critérios que descrevem como o sistema deve operar, como desempenho, segurança, usabilidade e compatibilidade, em vez de funcionalidades específicas.

2.10. Critérios de Aceitação (Acceptance Criteria):

Definem os requisitos mínimos para que um caso de uso seja considerado bem-sucedido, como a exibição de uma mensagem de sucesso após salvar os dados.

2.11. Interface Visual (User Interface):

A interface com a qual o usuário interage diretamente, incluindo botões, campos de entrada, tabelas e mensagens de erro ou sucesso.

2.12. Fluxo de Dados (Data Flow):

O caminho que os dados percorrem dentro do sistema, desde sua entrada até a manipulação ou armazenamento, como os dados de um arquivo JSON sendo processados pelo sistema.

2.13. Ponto de Extensão (Extension Point):

Referência a funcionalidades adicionais que podem ser integradas no futuro ao sistema, como suporte para outros formatos de arquivo ou novos tipos de busca.

2.14. Auditoria (Audit):

Processo de monitoramento e registro de ações e eventos dentro do sistema, como acessos e erros, para posterior análise e melhoria contínua.

2.15. Log de Erro (Error Log):

Registro detalhado de falhas ou exceções que ocorrem no sistema, utilizado para diagnóstico e correção de problemas.

2.16. Permissão Insuficiente (Insufficient Permission):

Situação onde o usuário não tem as permissões necessárias para realizar uma ação no sistema, como excluir dados ou acessar informações restritas.

2.17 Reconhecimento de Veículo (Vehicle Recognition):

Processo de identificar um veículo no estacionamento, geralmente utilizando dados como a placa, modelo e horário de entrada.

2.18. Estrutura JSON (JSON Structure):

A maneira como os dados são organizados dentro de um arquivo JSON. A estrutura deve ser compatível com o sistema para garantir que os dados sejam carregados e manipulados corretamente.

2.19. Banco de Dados (Database):

Sistema que organiza e armazena dados de forma estruturada, como informações de vagas, veículos e transações dentro do estacionamento.

2.20. Vaga Ocupada (Occupied Slot):

Uma vaga de estacionamento que está em uso por um veículo, com informações relacionadas à hora de entrada, número da vaga, entre outras.

2.21. Veículo (Vehicle):

Objeto que ocupa uma vaga de estacionamento, geralmente identificado por atributos como placa, modelo, hora de entrada e, em alguns casos, hora de saída.

2.22. Modal de Confirmação (Confirmation Modal):

Caixa de diálogo que aparece na interface para que o usuário confirme uma ação importante, como a exclusão de todos os dados, garantindo que o usuário deseja continuar.

2.23. Busca por Placa (License Plate Search):

Funcionalidade que permite ao usuário localizar um veículo no estacionamento, fornecendo o número da placa, retornando informações sobre a vaga ocupada e o tempo de permanência.

2.24. Tempo de Permanência (Time Spent):

A duração do tempo em que um veículo permaneceu estacionado no estacionamento, geralmente calculado a partir da hora de entrada até a hora de saída.

2.25. Sistema de Estacionamento (Parking System):

Software que gerencia as vagas, veículos e transações dentro de um estacionamento. Inclui funcionalidades como carregar, salvar, excluir e buscar dados, além de exibir informações de ocupação e status.