DesapegAuto

Guia de Análise e Projeto

Versão <1.0>

Índice Analítico

Conteúdo

[Introdução](#_Toc321330753) 3

[Mapeamento](#_Toc321330754) 3

[Classe de fronteira](#_Toc321330755) 3

Classes de Controle 3

[Classes do tipo Entity](#_Toc321330756) 4

[Classe do tipo EntityCollection](#_Toc321330757) 5

Documento de Arquitetura de Software

# Introdução

Este documento tem como objetivo descrever a visão de implementação do software DesapegAuto e mostrar como foi feito o mapeamento dos elementos de análise (atores, casos de uso e classes) para a estrutura do projeto. Ele serve como um guia para entender a arquitetura e o design do sistema, detalhando os componentes principais e suas interações.

# Mapeamento

## Classe de fronteira

Diagrama

O conteúdo gerado por IA pode estar incorreto.As classes de fronteira (Boundary) representam as interfaces com as quais os atores interagem, como telas do sistema. Para cada classe de fronteira da análise, é criada uma classe Controller no projeto. A classe Controller é armazenada no pacote Controller e é responsável por receber as requisições do usuário e delegar a lógica de negócio.

Figura 1 – Mapeamento das classes de fronteira

**Classes de controle**

As classes de controle (Control) da análise contêm a lógica de negócio e a coordenação das operações. Para cada classe de controle, é criada uma classe Service no projeto. A classe Service é armazenada no pacote Service e encapsula as regras de negócio do sistema.

Diagrama

O conteúdo gerado por IA pode estar incorreto.

Figura 2 – Mapeamento das classes do tipo Controle

## 

**Classes de controle e do tipo Entity**

As classes de entidade (Entity) representam os objetos de negócio do sistema. Para cada classe do tipo Entity encontrada, é criada uma classe de mesmo nome no projeto, armazenada no pacote Core. Esta classe encapsula os dados e comportamentos fundamentais do domínio.

Diagrama, Esquemático

O conteúdo gerado por IA pode estar incorreto.

Figura 3 – Mapeamento das classes do tipo Entity

**Classe do tipo EntityCollection**

EntityCollection, que na fase de análise representam o conjunto de dados persistidos de uma entidade, são substituídas por uma única classe de contexto do projeto. Esta classe de contexto centraliza o acesso e a manipulação dos dados, atuando como a principal ponte de comunicação com o banco de dados.

No projeto, essa classe de contexto é implementada dentro do pacote Core. A Figura 4 demonstra como as responsabilidades de uma classe de persistência individual, como PersistenciaUsuario, são absorvidas pela classe de contexto unificada, que gerencia operações como salvar (SaveChanges), deletar (Delete) e obter (Get) os dados.

Tela de celular com texto preto sobre fundo branco

O conteúdo gerado por IA pode estar incorreto.

Figura 4 – Mapeamento das classes do tipo EntityCollection