

Documentação: Arquitetura Lógica e Física

Autores: Pedro Vargas, Pedro Teixeira e Kawan Silva

1. Introdução

Este documento descreve as arquiteturas lógica e física da aplicação desenvolvida no projeto da disciplina Engenharia de Software 2 (ES2). A aplicação segue o padrão arquitetural MVC (Model-View-Controller) e utiliza tecnologias modernas como Cloud Computing e Infrastructure as Code (IaC).

2. Arquitetura Lógica

A arquitetura lógica da aplicação é programada na linguagem Python e é baseada no padrão MVC, que promove a separação de responsabilidades:

- **Model (Modelo):** Responsável pela lógica de negócios e acesso aos dados. No repositório, esta camada está representada pela pasta `model/`.
- **View (Visão):** Gerencia a interface com o usuário, apresentando os dados de forma compreensível. Os arquivos relacionados estão na pasta `view/`.
- **Controller (Controlador):** Atua como intermediário entre o Modelo e a Visão, processando as entradas do usuário e atualizando o modelo e a visão conforme necessário. Esta funcionalidade está implementada na pasta `controller/`.

3. Arquitetura Física

A arquitetura física descreve a infraestrutura onde a aplicação é implantada e executada:

- **Servidor de Aplicação:** Virtual Machine (Máquina Virtual) que hospeda a aplicação Python, executando o arquivo `main.py`.
- **Banco de Dados Relacional:** Utiliza o MySQL para armazenamento persistente dos dados de “cadastro”. A configuração do banco está na pasta `model/`.
- **Banco de Dados Não Relacional:** Implementa o Redis para armazenamento de dados da seção “chat”, pois o mesmo apresenta boa aplicabilidade nessa situação. Sua configuração também está na pasta `model/` e `redis/`.
- **Infraestrutura como Código (IaC):** Utiliza o Terraform para provisionamento e gerenciamento da infraestrutura na nuvem Oracle Cloud Infrastructure (OCI). Os scripts estão na pasta `oci-terraform/`.
- **Ferramentas de Desenvolvimento:** Inclui configurações do DBeaver para gerenciamento do banco de dados (`dbeaver/`) e do Uptime Kuma para monitoramento da aplicação (`UptimeKuma/`).

Sobre o iaC:

Este código Terraform é responsável por provisionar a infraestrutura necessária para a aplicação na nuvem Oracle Cloud Infrastructure (OCI).

Ele cria recursos como:

1. Instâncias de máquina virtual (VM) para hospedar a aplicação.
2. Redes e sub-redes para garantir a conectividade segura entre os recursos.
3. Configuração de segurança (firewall, grupos de segurança) para controlar o acesso à VM.
4. Provisionamento de volume de armazenamento para persistência de dados.

A infraestrutura é definida de forma declarativa, permitindo fácil replicação e escalabilidade em diferentes ambientes.

Além disso, o código pode ser reutilizado e versionado, facilitando a manutenção e a consistência entre ambientes de desenvolvimento, teste e produção.

4. Conclusão

A aplicação desenvolvida demonstra uma arquitetura bem estruturada, utilizando práticas modernas de desenvolvimento e implantação. A separação clara entre as camadas do MVC facilita a manutenção e escalabilidade do sistema.