



Tribute - Rede social de Voluntariado

Licenciatura em Engenharia Informática e de Computadores

Autores: Guilherme Allen
Leonardo Martins

Orientador: Nuno Leite

Instituto Superior de Engenharia de Lisboa – 22 Setembro
2020

Índice

- Introdução
 - Motivação
 - Solução
- Arquitetura
 - API
 - Aplicação móvel
 - Aplicação *web*
 - Implantação
- Demonstração
- Trabalho futuro

Introdução

O voluntariado é importante porque promove a valorização do meio social e a solidariedade.



Fonte: <https://www.unl.pt/>

A participação neste tipo de ações é também uma mais valia para o voluntário devido às *soft skills* adquiridas.

Introdução - Motivação



As ações de voluntariado são tipicamente divulgadas através de redes sociais ou *websites*.

Introdução - Solução

Desenvolvimento de uma rede social de voluntariado, onde é possível consultar informações relacionadas com este tipo de ações e interagir com outros utilizadores.



Arquitetura - Geral

- **API:** desenvolvida em Typescript e executada sobre Node.js;
- **Aplicação móvel:** desenvolvida em Kotlin para o sistema operativo Android;
- **Aplicação web:** desenvolvida utilizando React para clientes *browser*.



Arquitetura geral do projeto.

API - Conceptualização e Funcionalidades

○ Entidades:

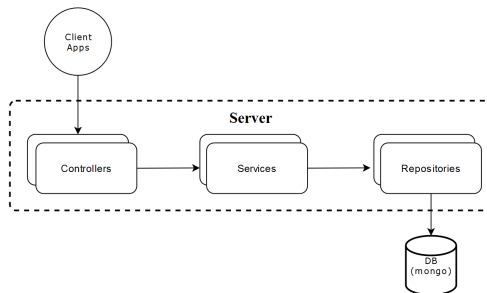
- **Voluntário**: indivíduo participante da ação;
- **Organização**: entidade que organiza ação;
- **Evento**: ação de voluntariado;
- **Post**: meio de interação entre utilizadores.

○ Funcionalidades:

- adicionar e alterar entidades;
- assegurar interação entre utilizadores;
- permitir que organizações possam verificar e obter contatos de voluntários interessados.

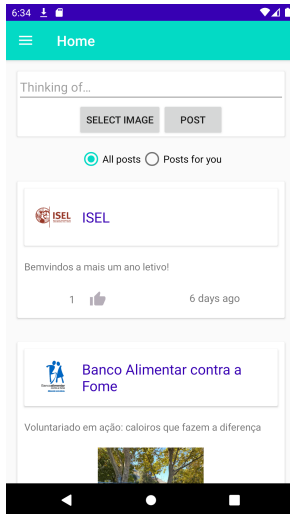
API - Arquitetura

- **Controladores:** definem *endpoints* e lidam com pedidos HTTPS;
- **Serviços:** implementam a lógica de negócio;
- **Repositórios:** interagem com a base de dados.



Arquitetura da API.

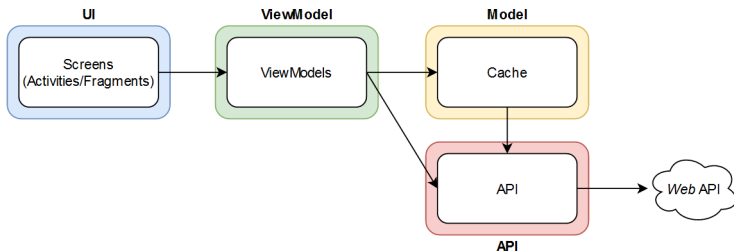
Aplicação Móvel



Ecrã inicial da aplicação.

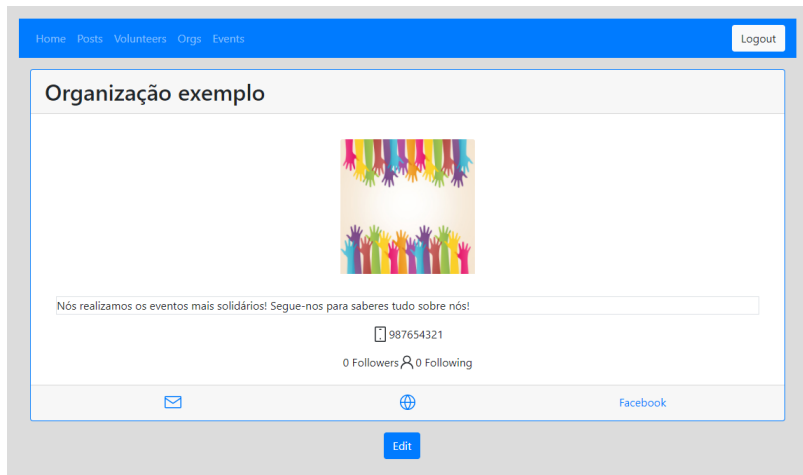
Aplicação Móvel - Arquitetura

- **UI**: define interfaces de utilizador e trata operações de *input*;
- **Model**: define DTO's (*Data Transfer Objects*) e implementação de Cache;
- **API**: funciona como *proxy* da *web API*.



Arquitetura da aplicação móvel.

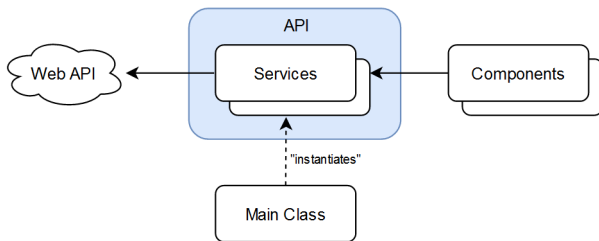
Aplicação web



Dashboard da aplicação.

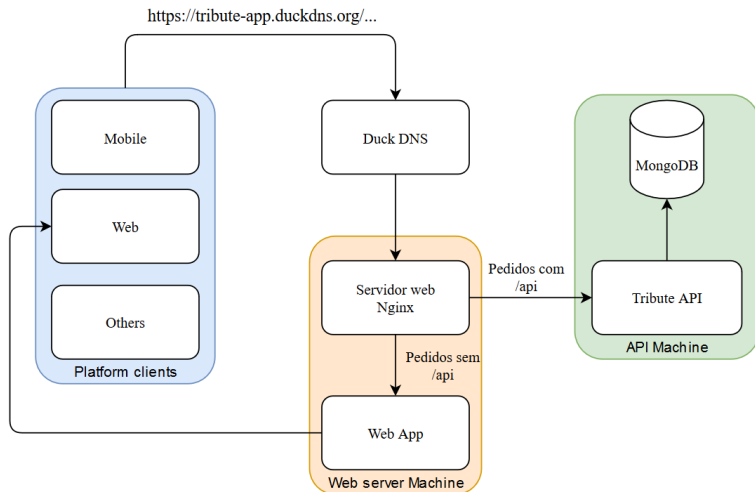
Aplicação *web* - Arquitetura

- **Classe Principal:** instancia serviços da API e define roteamento da aplicação;
- **Componentes:** definem a interface de utilizador, incluindo o tratamento de operações de *input*;
- **API:** contém implementação de serviços utilizados para aceder à *web API*.



Arquitetura da aplicação *web*.

Arquitetura - Implantação



Implantação do projeto.

Demonstração

Trabalho Futuro

- adição de novas funcionalidades, como por exemplo, comentários em *posts*, voluntários participantes em eventos e *curriculum vitae* de voluntariado;
- integração de registo e autenticação usando contas de outras plataformas (através de OAuth 2.0).