

**Relatório do Trabalho Prático**

**Desenvolvimento de um sistema distribuído**

**1ª Fase - Planeamento**

Arquitetura e Integração de Sistemas

**Docente:** Bruno Lima

**Trabalho realizado por:**

14829 José Gomes

15708 Pedro Carvalho

15709 José Carreira

Ano letivo 2023/2024

Mestrado em Engenharia Informática

Barcelos, novembro 2023

Índice

[1. Introdução 3](#_Toc150683177)

[2. Tema escolhido 4](#_Toc150683178)

[2.1. Problema Identificado 4](#_Toc150683179)

[2.2. Análise do problema 4](#_Toc150683180)

[2.3. Abordagem/Estratégia 5](#_Toc150683181)

[3. Processo de negócio 6](#_Toc150683182)

[3.1. BPMN 6](#_Toc150683183)

[4. Requisitos 7](#_Toc150683184)

[4.1. Requisitos Funcionais 7](#_Toc150683185)

[4.2. Requisitos não funcionais 8](#_Toc150683186)

[5. Arquitetura do Sistema 9](#_Toc150683187)

[6. Funcionalidades a explorar 10](#_Toc150683188)

[7. Pontos a salientar 10](#_Toc150683189)

[8. Conclusão 11](#_Toc150683190)

# Introdução

Este relatório é apresentado no âmbito do trabalho prático da unidade curricular de Arquitetura e Integração de Sistemas, do Mestrado de Engenharia Informática. Esta UC tem como objetivo dar a conhecer os fundamentos de arquiteturas de sistemas, baseadas em *microservices.*

Pretende-se efetuar o desenvolvimento de um sistema distribuído que permita gerir e agilizar um determinado processo de negócio, utilizando uma arquitetura de sistema baseada em *microservices.* Permitindo independência entre os diferentes *microservices,* tornando o sistema escalável, com flexibilidade na utilização de tecnologias diferentes e agilidade no processo de desenvolvimento.

# Tema escolhido

A aplicação a ser desenvolvida tem como objetivo agilizar a ida a eventos desportivos. Disponibilizando informação útil que torne uma ida a este tipo de eventos mais confortável para o utilizador da aplicação.

## Problema Identificado

Os adeptos de futebol normalmente gostam de assistir a jogos, sejam eles da equipa que apoiam ou outro qualquer jogo.

O problema identificado foi a complexidade na preparação de uma ida a um evento desportivo. Pela indisponibilidade demonstrada, desconhecimento de possibilidades e falta de tempo que as pessoas têm atualmente, de forma a preparar atempadamente os detalhes associados ao evento:

* Onde estacionar;
* Onde almoçar/jantar;
* Como chegar lá.

Por estes motivos, uma experiência que deveria ser agradável, com momentos de alegria, entusiasmo e diversão, por vezes acaba por ser motivo de stress e desagradável.

## Análise do problema

Pretende-se desenvolver uma aplicação que ajude a resolver este problema, fornecendo informação ao utilizador sobre diferentes jogos, locais de estacionamento, estado da meteorologia, restaurantes/bares, direções até ao estádio, de forma rápida e assertiva.

Um utilizador pode verificar eventos desportivos do seu clube ou outros eventos que existam perto dele, dentro de um determinado período temporal e, para cada evento, obter um plano com informação relevante (direções, meteorologia, tempo de viagem, restaurantes/bares perto do recinto desportivo e locais de estacionamento).

## Abordagem/Estratégia

A estratégia assumida passa pela criação de um *microservice* para cada funcionalidade que se pretende disponibilizar na aplicação:

* Jogos
* Direções
* Lazer
* Tempo
* Estacionamento

Cada *microservice* tem a sua própria arquitetura e é independente dos restantes, podendo comunicar com *API’s* externas e/ou base de dados. Também será utilizado o *microservice* de autenticação (que funcionará com alojamento na *Cloud*) que será desenvolvido para o trabalho da unidade curricular Sistemas de Computação na *Cloud*, esse *microservice* será desenvolvido tendo em consideração a articulação com o restante sistema, garantindo a segurança e gestão de acessos.

# Processo de negócio

## BPMN

# Requisitos

## Requisitos Funcionais

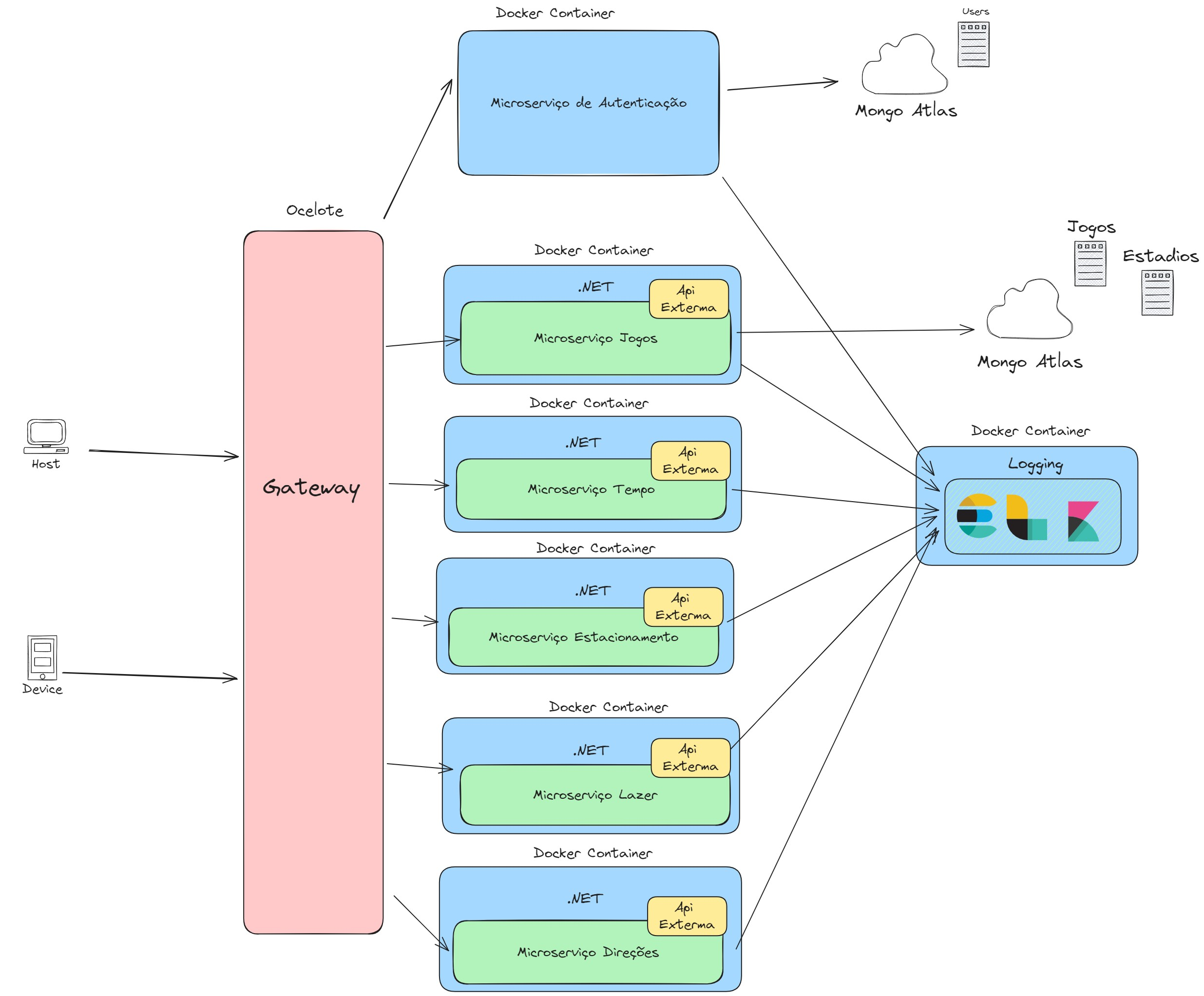
|  |  |  |  |
| --- | --- | --- | --- |
| **Requisitos Funcionais** | | | |
| **ID** | **Requisito** | **Descrição** | **Prioridade** |
| RF01.1 | Gestão de Utilizadores | O sistema tem de permitir, ao perfil Administrador, registar novos utilizadores. Campos obrigatórios:   * Login; * Password; * Nome; * Email; * Contacto; * Data Nascimento; * Tipo de Perfil; | MUST |
| RF01.2 | O sistema deverá considerar os seguintes Tipos de Perfil:   * Utilizador; * Administrador; | MUST |
| RF01.3 | O sistema poderá permitir inativar utilizadores. | SHOULD |
| RF01.4 | O sistema poderá permitir atualizar a informação pessoal ao próprio utilizador:   * Nome; * Email; * Contacto; * Data Nascimento;   Alterações possíveis enquanto o utilizador tem a sessão ativa. | COULD |
| RF01.5 | O login deverá ser possível através de:   * Registo de utilizador, apresentando os campos:   + Login;   + Password; | MUST |
| RF01.6 | O sistema poderá disponibilizar a opção de recuperação de password por Email. | COULD |
| RF01.7 | O sistema poderá controlar o nº de tentativas de passwords erradas, 3 erros no mesmo dia e fica com o login bloqueado durante 1h. | SHOULD |
| RF01.8 | O sistema poderá permitir a funcionalidade de alterar password, onde deverá ser registado:   * Password Antiga; * Password Nova; * Confirmar Password; | SHOULD |
| RF01.9 | O sistema deverá ter "pré-registado" um Administrador, cuja login seja gerida pelo proprietário. | MUST |
| RF01.10 | O sistema poderá permitir associar/retirar funcionalidades do perfil. | COULD |

## Requisitos não funcionais

Não sei se faz sentido

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requisitos Não Funcionais** | | | | |
| **ID** | **Requisito** | **Descrição** | **Prioridade** |
| RNF01 | Desempenho | O sistema deverá apresentar bom desempenho: respostas rápidas após clique e a carregar conteúdos. | MUST |
| RNF02 | Usabilidade | O sistema deverá apresentar uma interface simples e intuitivo, facilitando a sua utilização. | MUST |
| RNF03 | Eficiência | O sistema deve ter capacidade para processar até 1000 utilizadores simultaneamente. | SHOULD |
| RNF04 | Interoperabilidade | O sistema deverá comunicar com API’s, Cloud e MongoDB Atlas. | MUST |
| RNF05 | Confiabilidade | O sistema deverá correr sem problemas 99% do tempo, o serviço online é fundamental. | SHOULD |
| RNF06 | Idiomas | O sistema deverá suportar o idioma: Português. | MUST |
| RNF07 | Segurança e Privacidade | O sistema terá de garantir a segurança e privacidade dos dados dos utilizadores, pesquisas e ações efetuadas. | MUST |
| RNF08 | Compatibilidade | O sistema deverá ser capaz de correr nos browsers Chrome, IE e Mozilla, e dispositivos móveis. | SHOULD |
| RFN09 |  | O sistema deverá suportar a comunicação com *cloud* para consumir o *microservice* de autenticação no sistema. | SHOULD |
| RFN10 | Integração | O sistema deverá suportar a integração da API do Google Maps e Jogo | SHOULD |

# Arquitetura do Sistema



# Funcionalidades a explorar

Explicar de forma breve para cada microservice:

* Validação que existe na classe negócio para a ligação a cada microservice e como se valida a autenticação (se validar)
* Ligação às API’s em cada microservice
* Qual a BD a usar

# Pontos a salientar

* Forma como a classe de negócio comunica com a gateway e esta encaminha o pedido para o *microservice* correspondente?
* Articulação com o *microservice* autenticação, para validar a chamada dos restantes *microservices*
* As várias tecnologias utilizadas (de forma muito breve) em cada *microservice*
* Gestão de tokens
* MongoBD e ELK stack

# Conclusão

A realização da primeira fase deste trabalho foi bastante interessante, perceber a melhor forma de definir a arquitetura do sistema levou-nos a explorar algumas possibilidades e que há vários caminhos possíveis. É um desafio muito interessante escolher a estrutura, de forma correta e seguindo os princípios fundamentais aprendidos em aula (arquitetura modular e escalável), que melhor se enquadra com o modelo de negócio que se pretende operacionalizar.

Esta fase do trabalho também possibilitou a utilização de ferramentas e tecnologias diferentes (exemplo: *Docker*), sendo um desafio que acaba por aumentar o nosso conhecimento, permite-nos organizar melhor o trabalho e a ser mais assertivos na resolução de problemas e ajudam a agilizar o processo de desenvolvimento.