

## **Meet Lisbon**

Afonso Goulart (20211056), Maria Martins (20211010) e Mário Nascimento (20210387)

Universidade Europeia, IADE – curso de Engenharia Informática

Trabalho realizado no âmbito da unidade curricular de Projeto de Desenvolvimento Móvel

sob a supervisão do professor Pedro Rosa

Lisboa, 06 de outubro de 2022

Link para o repositório GitHub: <https://github.com/orgs/Meet-Lisbon/repositories>

## **Introdução**

De modo a criar uma aplicação divertida e útil pensámos numa aplicação que pudesse vir a ajudar os turistas a conhecer a nossa cidade, Lisboa. A nossa aplicação tem então como público-alvo os turistas, podendo, no entanto, servir também para portugueses à procura de conhecer melhor o seu próprio país. Explicando um pouco mais profundamente a essência da nossa aplicação, Meet Lisbon irá ser utilizada para podermos espalhar um pouco da nossa história com quem não a conhece. Através da aplicação, abrindo a câmara, o utilizador poderá fotografar um ponto turístico e o mesmo será reconhecido utilizando técnicas de inteligência artificial, como por exemplo Template Matching, mostrando depois vários factos históricos e alguns factos interessantes. Outro uso possível da aplicação será a pesquisa de pontos turísticos, podendo adicioná-los a uma *wishlist*.

## **Objetivos**

Com a criação da nossa aplicação definimos vários objetivos, passando estes por proporcionar aos turistas uma viagem onde possam aprender a história e cultura do nosso país, queremos possibilitar uma viagem tranquila e facilitada uma vez que podem utilizar a nossa aplicação para procurar vários pontos turísticos e conhecer a sua história em apenas uma ferramenta. Um dos nossos objetivos é também facilitar o fluxo de informação podendo utilizar a nossa aplicação em vez de ter de ocupar memória, tempo e recursos no dispositivo móvel com o que se calhar teriam de ser várias aplicações diferentes.

## **Pesquisa**

Para podermos construir uma aplicação o mais fidedigna possível fizemos várias pesquisas sobre a área do turismo. Entre estas pesquisas verificámos o crescimento do turismo em Portugal para podermos ter a certeza que esta aplicação seria de facto utilizada e não apenas mais uma aplicação sem grande uso na App Store. No gráfico da Fig. 1

verifica-se um padrão de crescimento ao longo dos anos, tendo descido naturalmente em 2020 e esperando atingir os valores mais altos até hoje em 2023.

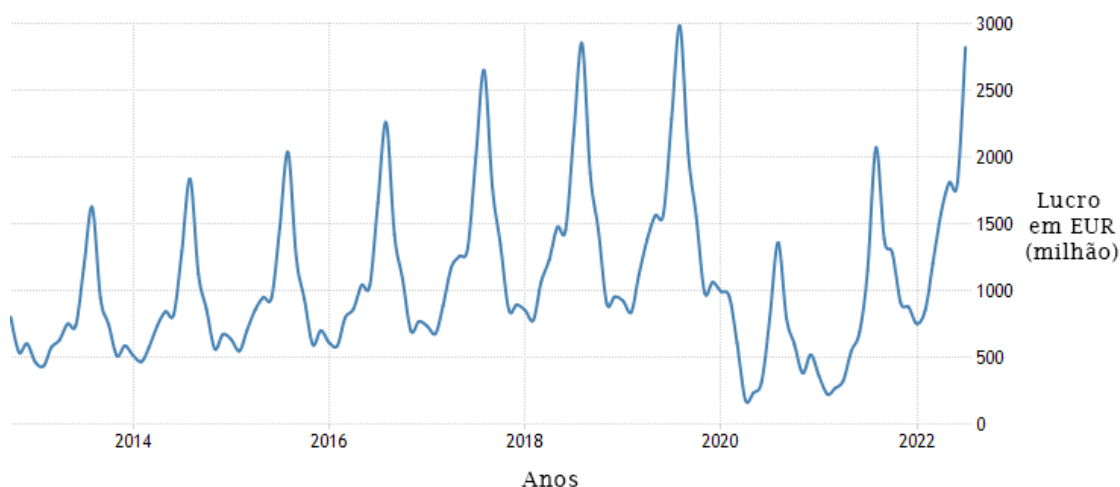


Fig. 1 - Lucro de turismo por ano em milhares de euros

### Benchmarking de mercado

De modo a podermos diferenciar-nos de muitos outros, fizemos pesquisas sobre as aplicações já existentes neste mercado para perceber quais os aspetos que têm de ser melhorados nesta área applicativa. Encontrámos várias aplicações e vamos falar um pouco das suas características: **Lisboa Cool** - encontra informação detalhada sobre os eventos do momento e os locais mais “cool” da capital portuguesa; **Citymapper** - nesta aplicação não só encontra informação sobre todos os transportes disponíveis em Lisboa, como também lhe são fornecidos os melhores trajetos para o seu destino; **Fever** - centenas de sugestões à espera das mais diversas áreas, como cinema e teatro, restaurantes, festas ou programas de bem-estar; **Lisbon Street Art** - dá a conhecer (quase) toda a street art da capital e arredores; **Lisbon Travel Guide** - pode planear percursos, de acordo com os mapas que são disponibilizados;

Como podemos ver existem várias aplicações para diferentes usos, mas nenhuma que consiga reconhecer um ponto turístico a partir da câmara e contar um pouco sobre a sua história, e é por essa razão que achamos o nosso projeto útil e interessante.

### **Guiões de teste**

Guião 1 (objecto *core* do projeto) - Um turista está a andar pelas ruas de Lisboa e depara-se com o Padrão dos Descobrimentos, ganha curiosidade pelo monumento e abre a aplicação que tem no telemóvel: “Meet Lisbon”. Assim que abre a aplicação, carrega no ícone da câmara, tira uma fotografia ao monumento conseguindo assim ler sobre este, perceber a sua história e até ler alguns *fun facts*.

Guião 2 (utilização diferente do *core* do projeto) - Um português que vive no Porto sabe que irá visitar Lisboa nas férias e instala a aplicação “Meet Lisbon” de modo a poder ter um auxílio de viagem. Num dia está a preparar a viagem e vai utilizar a aplicação para poder ver quais os monumentos e pontos turísticos que tem interesse. O senhor abre a aplicação e vai ao ícone do mapa para poder ver quais os monumentos disponíveis no leque da aplicação e ver o quão perto uns dos outros estes se situam para ver o tempo que demora de viagem de uns para os outros e planear assim o seu dia.

Guião 3 (utilização diferente do *core* do projeto) - Durante a sua viagem a Lisboa, um turista abre a aplicação “Meet Lisbon” de modo a ver quais os sítios que já visitou e quais os sítios que falta visitar, para isso carrega no ícone com um coração para poder ver a *wishlist* que fez em casa com todos os pontos de interesse.

### **Descrição genérica da solução a implementar**

Como descrevemos na introdução, a nossa aplicação passa por ajudar os turistas a conhecer melhor a capital portuguesa. Este projeto inclui vários aspetos como o uso da câmara fotográfica de modo a poder haver um reconhecimento do ponto turístico por parte do software, irá incluir localização geográfica pois teremos um mapa virtual com os todos os pontos turísticos para os utilizadores poderem adicionar à *wishlist* e criar uma rota de viagem, entre outros aspetos.

### **Enquadramento nas diversas Unidades Curriculares**

Sendo este um projeto multidisciplinar, arranámos maneira de incluir todas as cadeiras na idealização da nossa aplicação:

- Bases de Dados - Utilizaremos bases de dados para armazenamento, busca e manipulação de informação sobre os utilizadores, as suas preferências, monumentos e outros dados pertinentes que requerem permanência;
- Programação Orientada a Objetos - Vamos utilizar uma backend RESTful API feita numa linguagem maioritariamente orientada a objetos (java), com o objetivo de servir de centro de controlo. Este servidor servirá como intermediário entre a aplicação móvel (frontend) e a base de dados e será também a aplicação que vai receber os pedidos e enviar as respostas adequadas;
- Programação de Dispositivos Móveis - Será feita uma aplicação móvel que funcionará como a frontend do projeto. O utilizador usará esta app para interagir com os recursos disponibilizados pela backend, base de dados e outros serviços. A nossa aplicação terá um GUI através do qual os utilizadores poderão fazer esta interação.
- Competências Comunicacionais - Iremos utilizar as competências aprendidas em aula de modo a melhorarmos as nossas apresentações orais de entrega a entrega;
- Matemática Discreta - Para manipulação e reconhecimento de imagens usaremos algoritmos e estratégias que se enquadram na cadeira de matemática discreta, como por exemplo Algoritmo de Dijkstra para computação de caminhos, TM\_SQDIFF ou TM\_CCoeff para template matching, entre outros.

## Requisitos Técnicos para desenvolvimento do projeto

Para o desenvolvimento deste projeto, será necessária a aprendizagem e utilização de várias linguagens de programação e outras plataformas: **Linguagem Java** -> Para desenvolvimento em Spring Boot da backend API e da aplicação Android; **Linguagem SQL** -> Para uso e configuração da base de dados; Programação móvel; Orquestração e coordenação de containers para criação do cluster; Automatização de testes e integração/distribuição contínuas; Funcionamento e programação de algoritmos para análise de imagens -> trabalhar com a biblioteca OpenCV.

## Arquitetura da Solução

A nossa aplicação, embora pequena, requer uma coordenação entre vários equipamentos e tecnologias, assim como uma infraestrutura sólida e de fácil manutenção.

Os diversos componentes a integrar são:

- \* Base de dados
- \* Backend API
- \* Aplicação Android

A base de dados e backend API têm de garantir uma disponibilidade alta, assim como uma latência baixa, visto que a interação com o utilizador é baseada na sua localização e movimentação em tempo real. Por outro lado, no início poderá ser mais vantajoso dar prioridade à simplicidade e organização dado que as necessidades iniciais serão baixas. Para tal, optamos por uma infraestrutura de single cluster com load balancing. A orquestração das diversas componentes da backend api e base de dados será feita pelo Kubernetes, como demonstrado na Fig. 2.

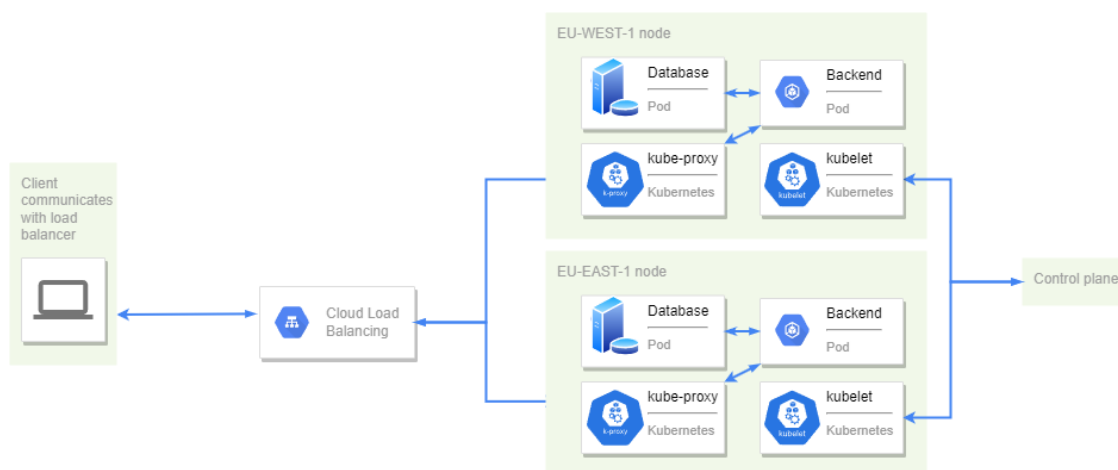


Fig. 2 - Diagrama de rede

A base de dados e a backend api correm em containers individuais dentro de pods. Estes pods correm indefinidamente em worker nodes. Estes comunicam com a rede externa através do kube-proxy. A manutenção, monitorização e orquestração são organizadas pelo control plane. A kubelet, presente em todos os pods, tem como objetivo comunicar com o control plane. O plano é utilizar Google Kubernetes Engine (GKE) para gerir o cluster.

### Armazenamento permanente

Utilizaremos uma base de dados, PostgreSQL. Este servidor não terá comunicação com o exterior, interagindo apenas com a backend. Desta forma aumenta-se a segurança da nossa aplicação, reduzindo os vetores de ataque expostos.

### Backend API

O servidor backend será feito utilizando Spring Boot numa arquitetura REST. Esta servirá para receber requests da frontend, processá-los e devolver uma resposta. A comunicação com a base de dados e eventualmente com outros recursos internos (ou, possivelmente, externos) será feita pela backend quando necessária.

### Load balancer

Os utilizadores (na verdade, a frontend) não comunica diretamente com os worker nodes do nosso cluster. Para garantir uma boa disponibilidade e rapidez geral, o load

balancer é responsável por encaminhar os pedidos para os nodes que fazem mais sentido, tendo em conta dados como a proximidade ao utilizador e a carga atual sobre os servidores.

### Aplicação Android

Esta aplicação terá a função de frontend. O seu objetivo será possibilitar uma interação com os utilizadores através da sua interface gráfica e a comunicação com a backend, intermediando o utilizador e os recursos internos.

### Integração e distribuição

Para limitar o número de erros humanos é vantajoso automatizar a maior parte dos espaços, especialmente aqueles mais extensos e que se repetem. Planeamos então utilizar Github Actions e, possivelmente, algumas plataformas como Jenkins para facilitar estas tarefas. Assim, a ideia geral da pipeline para integração e distribuição contínuas está representada na Fig. 3.

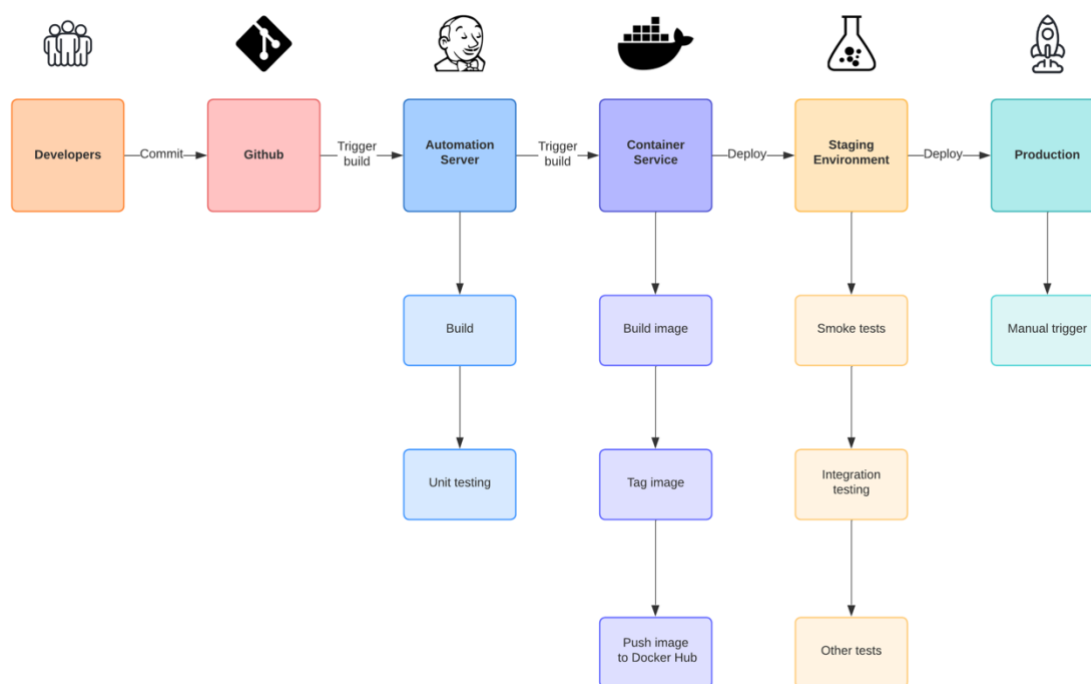


Fig. 3 - CI/CD pipeline





## Diagrama de Classes

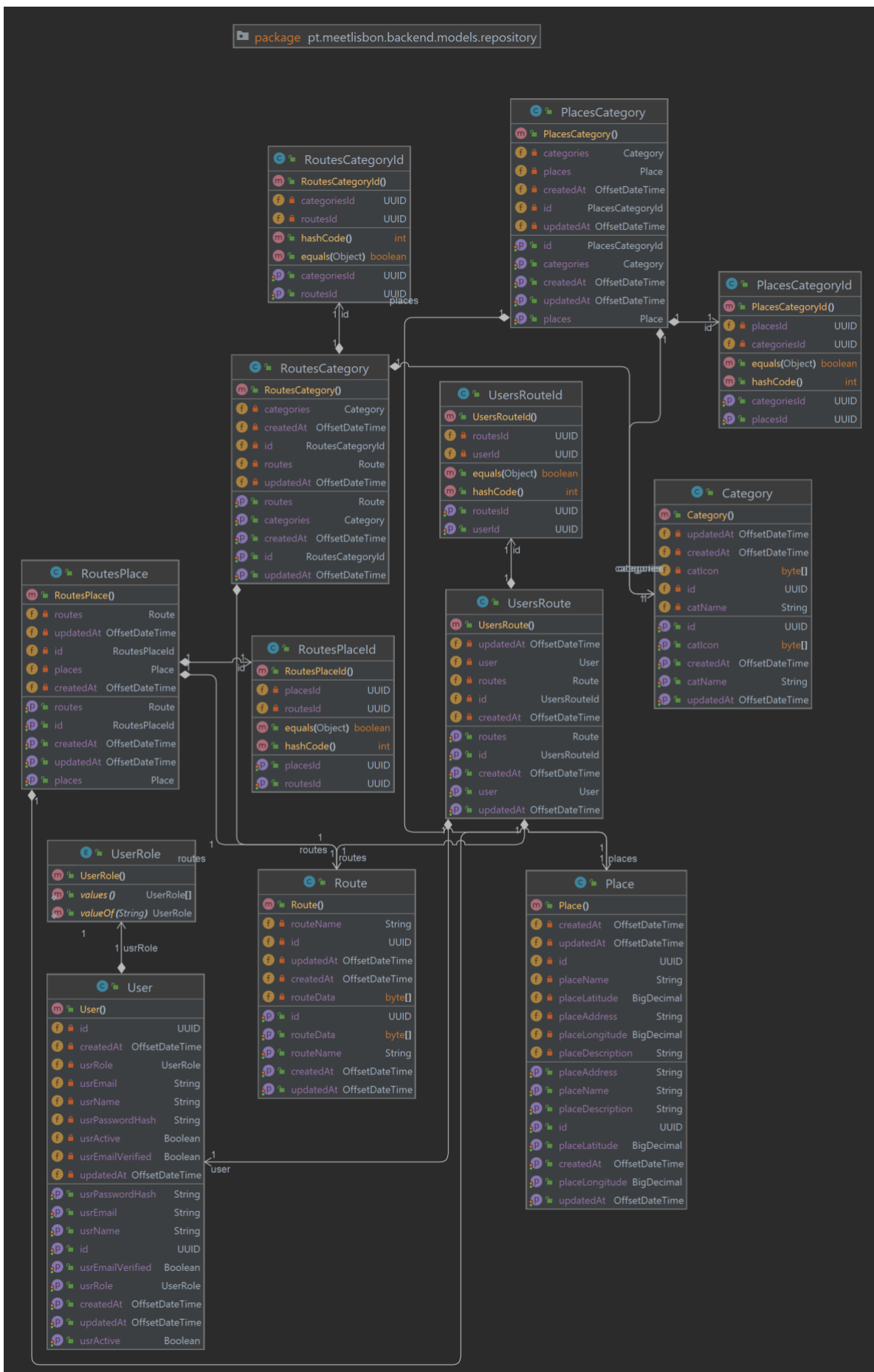


Fig. 6 – Diagrama de Classes

## Documentação REST

Ver documentação REST neste link: <https://github.com/Meet->

[Lisbon/documents/blob/main/G01-documentac%C3%A7%C3%A3o-REST.pdf](https://github.com/Meet-Lisbon/documents/blob/main/G01-documentac%C3%A7%C3%A3o-REST.pdf)

## Dicionário de Dados

| Table name        | isKey | Col Name           | Data type                   | isNull | Description                                    |
|-------------------|-------|--------------------|-----------------------------|--------|--|
| categories        | PK    | cat_id             | uuid                        | yes    | The category's uuid                            |
| categories        |       | cat_name           | character varying           | yes    | The category's name                            |
| categories        |       | cat_icon           | bytea                       | yes    | The category's icon                            |
| categories        |       | created_at         | timestamp with time zone(6) | yes    | Creation date                                  |
| categories        |       | updated_at         | timestamp with time zone(6) | yes    | Last updated date                              |
| places            | PK    | place_id           | uuid                        | yes    | The place's uuid                               |
| places            | UK    | place_name         | character varying           | yes    | The place's name                               |
| places            |       | place_latitude     | numeric(6,5)                | yes    | The place's latitude                           |
| places            |       | place_longitude    | numeric(6,5)                | yes    | The place's longitude                          |
| places            |       | place_address      | character varying           | no     | The place's address                            |
| places            |       | place_description  | character varying           | no     | The place's description                        |
| places            |       | created_at         | timestamp with time zone(6) | yes    | Creation date                                  |
| places            |       | updated_at         | timestamp with time zone(6) | yes    | Last updated date                              |
| places_categories | FK,PK | places_id          | uuid                        | yes    | The place's uuid                               |
| places_categories | FK,PK | categories_id      | uuid                        | yes    | The category's uuid                            |
| places_categories |       | created_at         | timestamp with time zone(6) | yes    | Creation date                                  |
| places_categories |       | updated_at         | timestamp with time zone(6) | yes    | Last updated date                              |
| routes            | PK    | route_id           | uuid                        | yes    | The route's uuid                               |
| routes            |       | route_data         | bytea                       | yes    | The route's data                               |
| routes            |       | route_name         | character varying           | yes    | The route's name                               |
| routes            |       | created_at         | timestamp with time zone(6) | yes    | Creation date                                  |
| routes            |       | updated_at         | timestamp with time zone(6) | yes    | Last updated date                              |
| routes_categories | FK,PK | routes_id          | uuid                        | yes    | The route's uuid                               |
| routes_categories | FK,PK | categories_id      | uuid                        | yes    | The category's uuid                            |
| routes_categories |       | created_at         | timestamp with time zone(6) | yes    | Creation date                                  |
| routes_categories |       | updated_at         | timestamp with time zone(6) | yes    | Last updated date                              |
| routes_places     | FK,PK | routes_id          | uuid                        | yes    | The route's uuid                               |
| routes_places     | FK,PK | places_id          | uuid                        | yes    | The place's uuid                               |
| routes_places     |       | created_at         | timestamp with time zone(6) | yes    | Creation date                                  |
| routes_places     |       | updated_at         | timestamp with time zone(6) | yes    | Last updated date                              |
| users             | PK    | usr_id             | uuid                        | yes    | The user's uuid                                |
| users             | UK    | usr_name           | character varying           | yes    | The user's username                            |
| users             | UK    | usr_email          | character varying           | yes    | The user's email                               |
| users             |       | usr_password_hash  | character varying           | yes    | The user's password                            |
| users             |       | usr_email_verified | boolean                     | yes    | If user's email is verified (false by default) |
| users             |       | usr_role           | Enum (user, admin)          | yes    | The user's role (user or admin)                |
| users             |       | usr_active         | boolean                     | yes    | If user is active (true by default)            |
| users             |       | created_at         | timestamp with time zone(6) | yes    | Creation date                                  |
| users             |       | updated_at         | timestamp with time zone(6) | yes    | Last updated date                              |
| users_routes      | FK,PK | user_id            | uuid                        | yes    | The user's uuid                                |
| users_routes      | FK,PK | routes_id          | uuid                        | yes    | The route's uuid                               |
| users_routes      |       | created_at         | timestamp with time zone(6) | yes    | Creation date                                  |
| users_routes      |       | updated_at         | timestamp with time zone(6) | yes    | Last updated date                              |

Fig. 7 – Dicionário de Dados

## Modelo Entidade-Relação

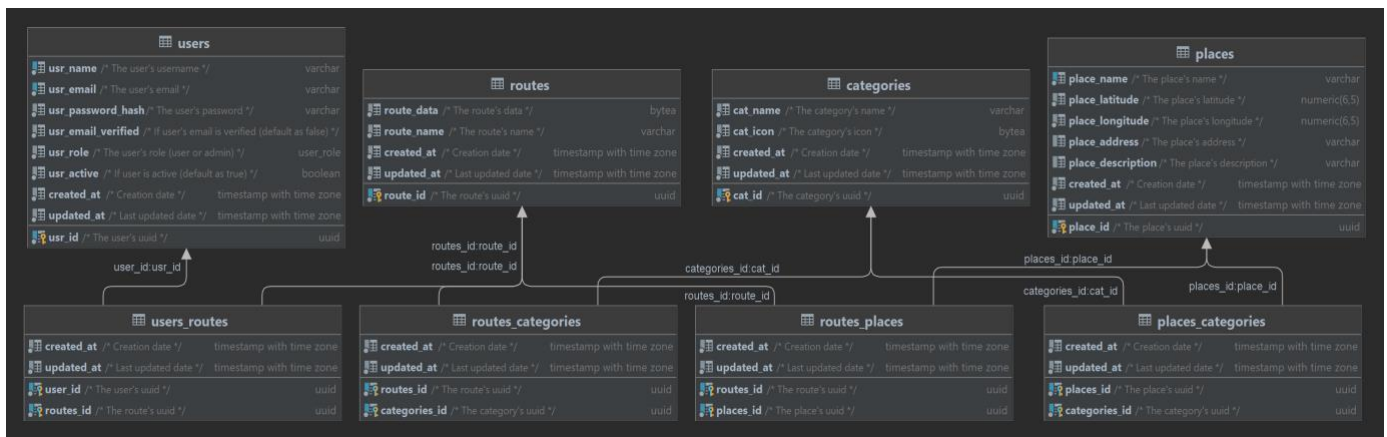


Fig. 8 – Modelo Entidade-Relação

## Referências Bibliográficas

Costa, J. S. L. (n.d.). A Procura Turística em Portugal: Evolução Recente. 85.

O que fazer em Lisboa: As apps que o vão ajudar. (2019, April 17). Moving to Portugal.

<https://www.movingtoportugal.pt/lazer/o-que-fazer-em-lisboa/>

Portugal Tourism Revenues—September 2022 Data—1996-2021 Historical. (n.d.).

Retrieved October 7, 2022, from <https://tradingeconomics.com/portugal/tourism-revenues>

Portugal's Travel & Tourism could surpass pre-pandemic levels in 2023. (n.d.). 3.

Rodrigues, N. (2019, March 7). As melhores apps para descobrir Lisboa. Lisboa Secreta.

<https://lisboasecreta.co/as-melhores-apps-para-descobrir-lisboa/>

T: +351 210 312 700, F: +351 210 312 899, [Atl@visitlisboa.com](mailto:Atl@visitlisboa.com), & [Www.visitlisboa.com](http://Www.visitlisboa.com).

(n.d.). Lisboa OFFICIAL Site. Turismo de Lisboa. Retrieved October 7, 2022, from

<https://www.visitlisboa.com/pt-pt>

## License

GNU General Public License v3.0