



EventoPass

Table of Contents

1. Autors	1
2. Objectiu de l'Aplicació	2
3. Disseny de Base de Dades	3
4. Taules de la Base de Dades EventoPass	4
4.1. Venues	4
4.2. Users	4
4.3. Categories	4
4.4. Events	5
4.5. Sessions	5
4.6. Ticket_Types	6
4.7. Purchases	6
4.8. Tickets	6
4.9. Roles	7
4.10. Event_Images	7
4.11. Comentaris	8
5. Taules de la Base de Dades de la API Imatges	9
5.1. Images	9
5.2. Optimized_Images	9
6. Sketching i Mockup d'Interfícies	11
7. Preparació del projecte en Local:	12
7.1. Preparació del projecte	12
7.2. Preparació de l'API d'Imatges	12
7.3. Preparació Cordova	13
7.4. Preparació BD i servidors API i EventoPass	13
7.5. Altres	14
8. Manual d'Instal·lació/Distribució	15
8.1. Diagrama de xarxa:	15
8.2. Instal·lació del apache2:	15
8.3. Preparació del servidor:	15
8.4. Creació del docker:	16
8.5. Aixecament de les bases de dades:	17
8.6. Configuració de servidor web:	17
8.6.1. EventoPass	17
8.6.2. API	18
9. Documentació de Permisos	19
10. Instal·lació del DomPDF	20
11. Descripció General de l'API	21
11.1. Informació General	21

11.2. Servidors	21
11.3. Rutes de l'API	21
11.3.1. Operacions sobre Imatges <i>/images</i>	21
GET	21
POST	22
11.3.2. Operacions sobre una Imatge Específica <i>/images/{id}</i>	22
GET	22
DELETE	22
POST	22
11.3.3. Operacions sobre Imatges Optimitzades	23
GET	23
12. Autenticació	24
12.1. Token d'Aplicació	24
13. Codis d'Estat d'Error Comuns	25
14. API de Validació de Tickets	26
14.1. Rutes de l'API	26
14.2. Validació de Sessió <i>/login</i>	26
14.2.1. POST	26
14.3. Validació de Ticket <i>/tickets/validate/{sessionId}/{hash}</i>	26
14.3.1. GET	26
15. Middleware Utilitzat a l'API de Validacion de Tickets	28
16. Web-grafia i Altres Recursos de Consulta	29
16.1. R18: Passarel·la de pagament	29
16.2. R28: Càrrega de recursos optimitzada	29
16.3. R29: Metadades i compartició a xarxes socials	29
16.4. R30: Microservei d'optimització d'imatges	29
16.5. R31: Proves de l'API amb Postman	30
16.6. R35: Creació d'aplicació per a validació d'entrades	30
17. Presentació del Projecte	31
18. Conclusions	32

Chapter 1. Autors

- [Alfred Perez](#)
- [Ferran Carmona](#)
- [Guiu Puigantell](#)

Chapter 2. Objectiu de l'Aplicació

El projecte té com a objectiu principal el desenvolupament d'un portal web innovador destinat a la venda d'entrades per a esdeveniments variats. Aquesta plataforma es concep com un eina clau per als promotors d'esdeveniments, oferint-los la capacitat de gestionar la venda d'entrades i la publicació dels seus esdeveniments de manera eficient i centralitzada.

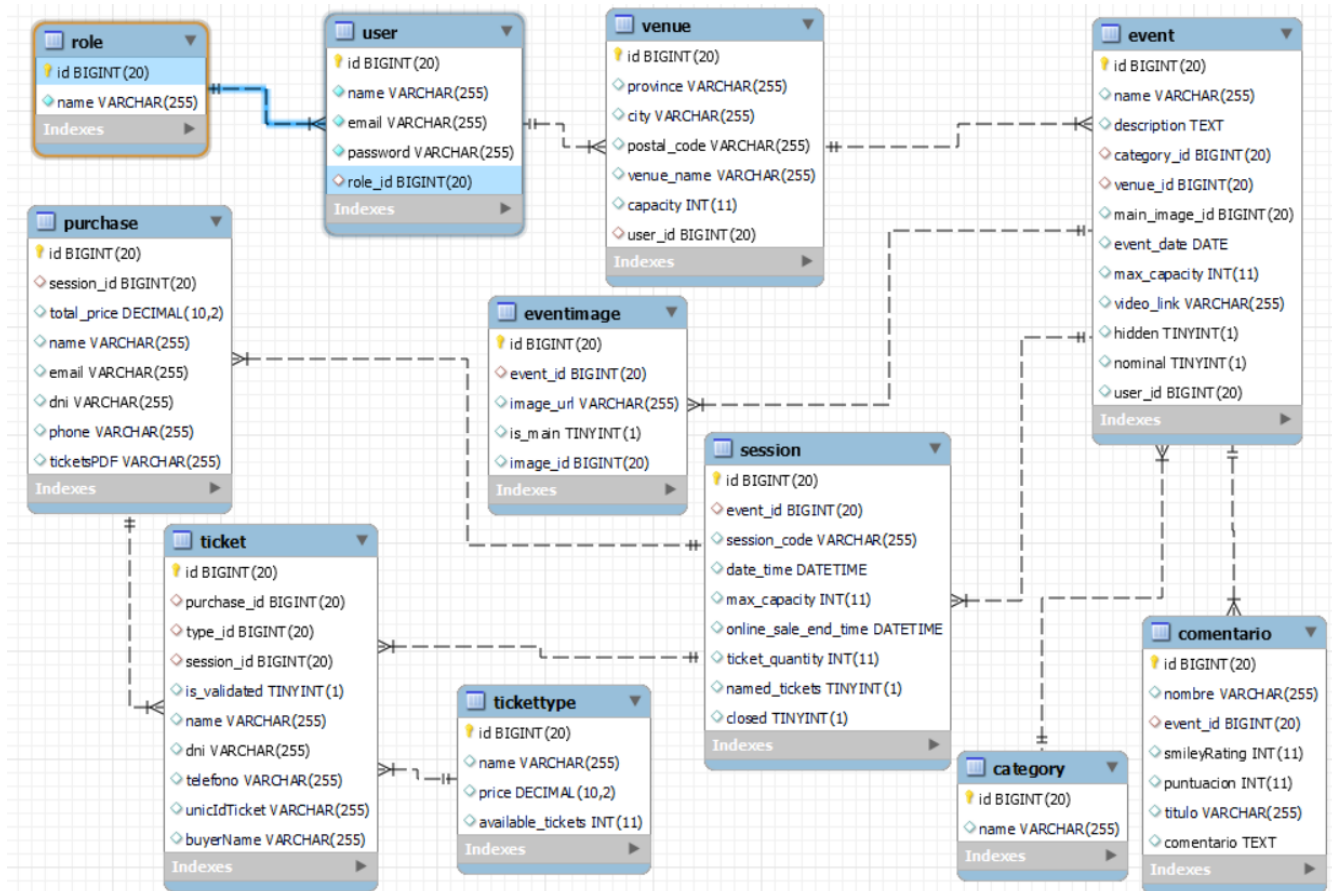
La plataforma es diferenciarà per les següents característiques:

1. **Venda d'Entrades:** Facilitarà la compra d'entrades de manera segura i fiable, satisfent les necessitats tant dels promotors com dels assistents.
2. **Gestió d'Esdeveniments pels Promotors:** Permetrà als promotors publicar i gestionar els seus esdeveniments, creant una experiència personalitzada i adaptada a les seves necessitats.
3. **Model de Negoci:** Implementarà un model de negoci basat en un percentatge sobre les vendes de cada entrada, assegurant un benefici mutu entre la plataforma i els promotors.
4. **Validació d'Entrades:** A través d'una aplicació web independent, es proporcionarà una solució per a la validació eficient d'entrades en l'entrada dels esdeveniments, garantint una gestió segura i ràpida.

Aquesta plataforma no només respondrà a una necessitat de mercat actual, sinó que també aportarà valor afegit als seus usuaris, facilitant una experiència d'usuari millorada i eficient tant per als promotors com per als assistents dels esdeveniments.

Chapter 3. Disseny de Base de Dades

El disseny de la base de dades per al portal web es basa en diverses taules interconnectades que garanteixen una gestió eficient de les dades dels esdeveniments. Aquestes taules principals són **categories**, **events**, **venues**, **sessions**, **purchases**, **users**, **ticket_types** i **tickets**. A continuació, es detallen aquestes taules i les seves relacions.



Chapter 4. Taules de la Base de Dades

EventoPass

4.1. Venues

Camp	Descripció	Tipus de Dada
<code>id</code>	Identificador únic per a cada venue (recinte)	<code>bigIncrements</code>
<code>province</code>	Província on es troba el venue	<code>string(255)</code>
<code>city</code>	Ciutat on es troba el venue	<code>string(255)</code>
<code>postal_code</code>	Codi postal del venue	<code>string(255)</code>
<code>venue_name</code>	Nom del venue	<code>string(255)</code>
<code>capacity</code>	Capacitat màxima del venue	<code>integer</code>
<code>user_id</code>	Clau forana que referència a l'usuari propietari del venue	<code>unsignedBigInteger</code> , clau forana referència <code>id</code> en <code>users</code>
<code>created_at</code> i <code>updated_at</code>	Camps automàtics per a registrar les dates de creació i última actualització	<code>timestamps</code>

4.2. Users

Camp	Descripció	Tipus de Dada
<code>id</code>	Identificador únic per a cada usuari	<code>id()</code>
<code>name</code>	Nom de l'usuari	<code>string</code>
<code>email</code>	Correu electrònic de l'usuari, ha de ser únic	<code>string</code> , únic
<code>email_verified_at</code>	Data i hora de la verificació del correu electrònic, pot ser nul·la	<code>timestamp</code> , nullable
<code>password</code>	Contrasenya de l'usuari	<code>string</code>
<code>rememberToken</code>	Token per a recordar la sessió de l'usuari	<code>rememberToken()</code>
<code>created_at</code> i <code>updated_at</code>	Camps automàtics per a registrar les dates de creació i última actualització de l'usuari	<code>timestamps</code>

4.3. Categories

Camp	Descripció	Tipus de Dada
<code>id</code>	Identificador únic per a cada categoria	<code>bigIncrements</code>
<code>name</code>	Nom de la categoria	<code>string(255)</code>

Camp	Descripció	Tipus de Dada
created_at i updated_at	Camps automàtics per a registrar les dates de creació i última actualització del registre	timestamps

4.4. Events

Camp	Descripció	Tipus de Dada
id	Identificador únic per a cada esdeveniment	bigIncrements
name	Nom de l'esdeveniment	string(255)
description	Descripció de l'esdeveniment	text
main_image_id	ID de la imatge principal de l'esdeveniment, pot ser null, vincula amb la BD de la API d'imatges	unsignedBigInteger, nullable
category_id	Clau forana que referència a la categoria de l'esdeveniment	unsignedBigInteger
venue_id	Clau forana que referència al recinte on es realitza l'esdeveniment	unsignedBigInteger
user_id	Clau forana que referència a l'usuari creador de l'esdeveniment, pot ser null	unsignedBigInteger, nullable
event_date	Data i hora de l'esdeveniment, pot ser null	timestamp, nullable
video_link	Enllaç al vídeo relacionat amb l'esdeveniment, pot ser null	string(255), nullable
hidden	Indica si l'esdeveniment està ocult o no	boolean, default false
nominal	Indica si l'esdeveniment és nominal o no	boolean, default false
created_at i updated_at	Camps automàtics per a registrar les dates de creació i última actualització del registre	timestamps

4.5. Sessions

Camp	Descripció	Tipus de Dada
id	Identificador únic per a cada sessió	bigIncrements
event_id	Clau forana que referència a l'esdeveniment associat	unsignedBigInteger
session_code	Codi únic de la sessió, pot ser null	string, únic, nullable
date_time	Data i hora de la sessió	timestamp
max_capacity	Capacitat màxima de la sessió, pot ser null	integer, nullable
online_sale_end_time	Temps final per la venda online de tiquets, pot ser null	timestamp, nullable
ticket_quantity	Quantitat de tiquets disponibles, pot ser null	integer, nullable

Camp	Descripció	Tipus de Dada
<code>named_tickets</code>	Indica si els tiquets són nominals	<code>boolean</code> , default false
<code>closed</code>	Indica si la sessió està tancada	<code>boolean</code> , default false
<code>created_at</code> i <code>updated_at</code>	Camps automàtics per a registrar les dates de creació i última actualització del registre	<code>timestamps</code>

4.6. Ticket_Types

Camp	Descripció	Tipus de Dada
<code>id</code>	Identificador únic per a cada tipus de tiquet	<code>bigIncrements</code>
<code>name</code>	Nom del tipus de tiquet	<code>string(255)</code>
<code>price</code>	Preu del tiquet	<code>decimal(8,2)</code>
<code>available_tickets</code>	Tiquets disponibles per aquest tipus, pot ser null	<code>integer</code> , nullable
<code>created_at</code> i <code>updated_at</code>	Camps automàtics per a registrar les dates de creació i última actualització del registre	<code>timestamps</code>

4.7. Purchases

Camp	Descripció	Tipus de Dada
<code>id</code>	Identificador únic per a cada compra	<code>bigIncrements</code>
<code>session_id</code>	Clau forana que referència a la sessió associada a la compra	<code>unsignedBigInteger</code>
<code>name</code>	Nom de la persona que realitza la compra	<code>string</code>
<code>dni</code>	Document Nacional d'Identitat de la persona que realitza la compra	<code>string</code>
<code>phone</code>	Telèfon de contacte de la persona que realitza la compra	<code>integer</code>
<code>email</code>	Correu electrònic de la persona que realitza la compra	<code>string</code>
<code>total_price</code>	Preu total de la compra	<code>decimal(8,2)</code>
<code>ticketsPDF</code>	Enllaç al PDF dels tiquets de la compra, pot ser null	<code>string</code> , nullable
<code>created_at</code> i <code>updated_at</code>	Camps automàtics per a registrar les dates de creació i última actualització del registre	<code>timestamps</code>

4.8. Tickets

Camp	Descripció	Tipus de Dada
id	Identificador únic per a cada tiquet	bigIncrements
is_validated	Indica si el tiquet ha estat validat	boolean, default false
purchase_id	Clau forana que referència a la compra associada, pot ser null	unsignedBigInteger, nullable
type_id	Clau forana que referència al tipus de tiquet	unsignedBigInteger
session_id	Clau forana que referència a la sessió associada	unsignedBigInteger
name	Nom de la persona titular del tiquet, pot ser null	string(255), nullable
dni	Document Nacional d'Identitat de la persona titular del tiquet, pot ser null	string(255), nullable
telefono	Telèfon de contacte de la persona titular del tiquet, pot ser null	string(255), nullable
uniqIdTicket	Identificador únic del tiquet, pot ser null	string(255), nullable
buyerName	Nom de la persona que ha fet la compra, pot ser null	string(255), nullable
created_at i updated_at	Camps automàtics per a registrar les dates de creació i última actualització del registre	timestamps

4.9. Roles

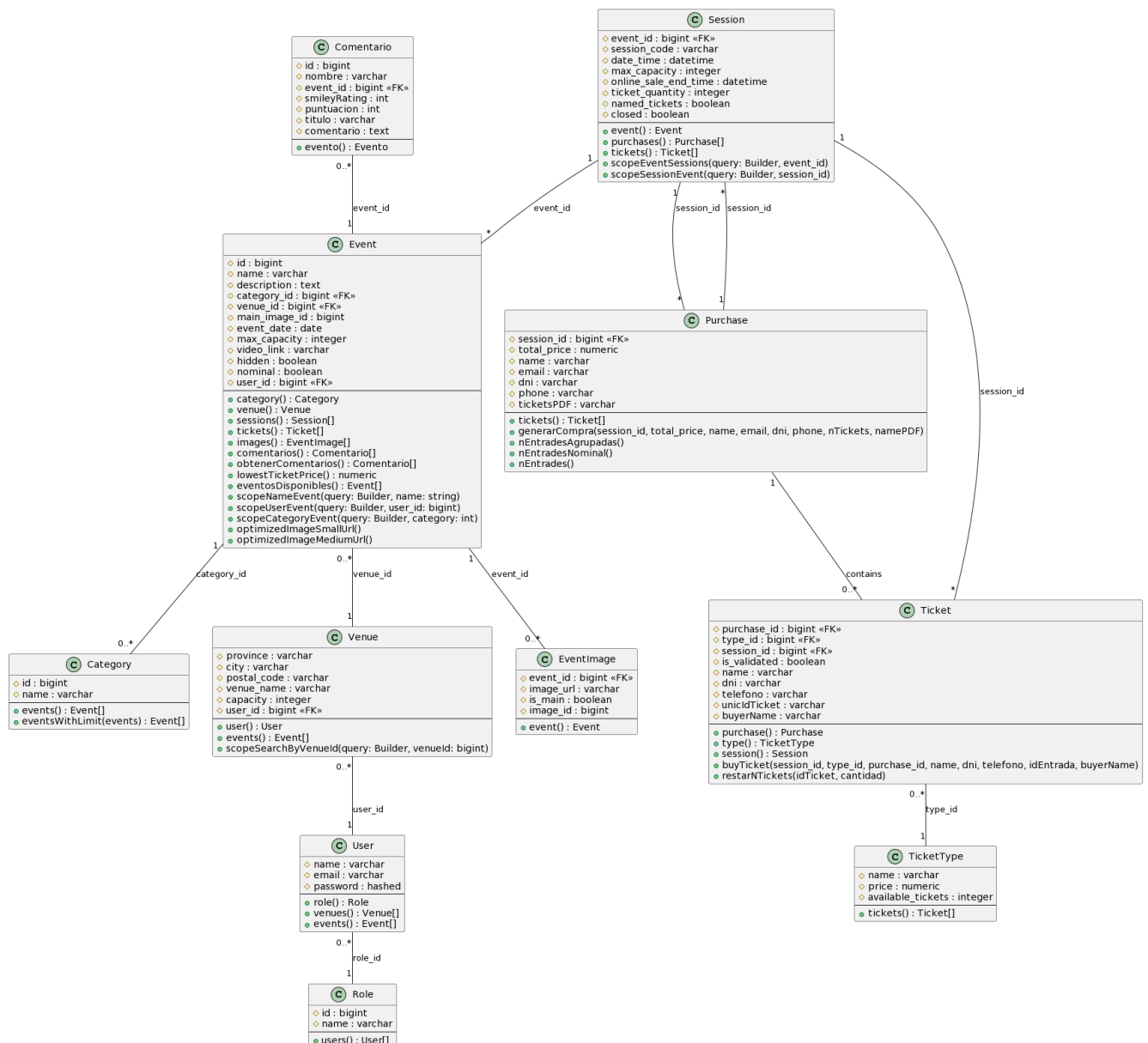
Camp	Descripció	Tipus de Dada
id	Identificador únic per a cada rol	id
name	Nom del rol, ha de ser únic	string, únic
created_at i updated_at	Camps automàtics per a registrar les dates de creació i última actualització del registre	timestamps

4.10. Event_Images

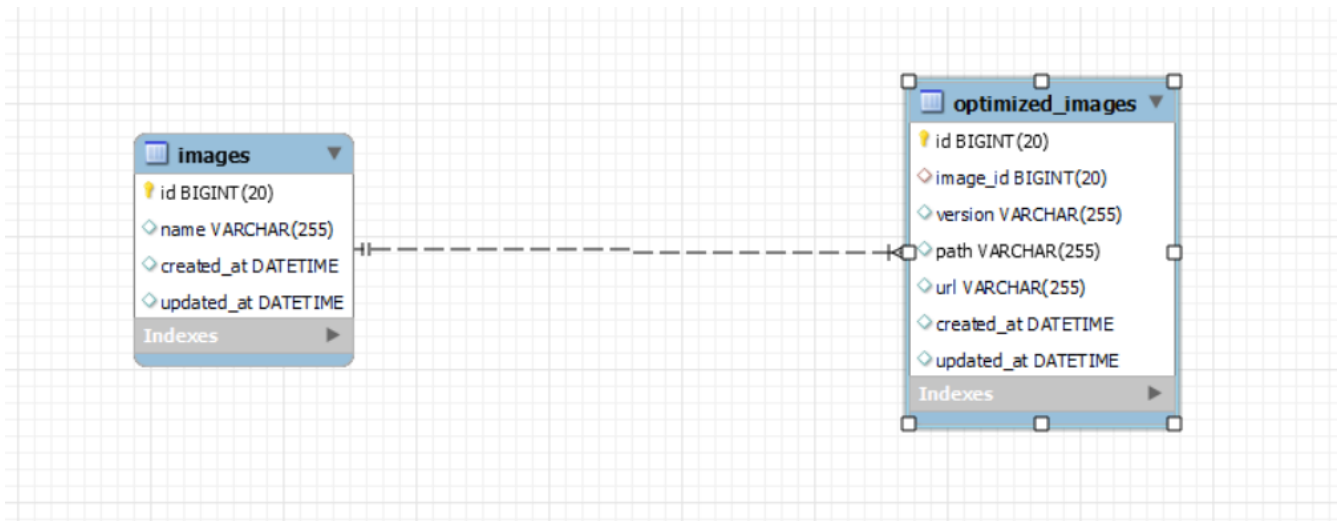
Camp	Descripció	Tipus de Dada
id	Identificador únic per a cada imatge d'esdeveniment	bigIncrements
event_id	Clau forana que referència a l'esdeveniment associat	unsignedBigInteger
image_id	Identificador de la imatge	string
is_main	Indica si la imatge és la principal de l'esdeveniment	boolean, default false
created_at i updated_at	Camps automàtics per a registrar les dates de creació i última actualització del registre	timestamps

4.11. Comentaris

Camp	Descripció	Tipus de Dada
id	Identificador únic per a cada comentari	bigIncrements
event_id	Clau forana que referència a l'esdeveniment associat	unsignedBigInteger
nombre	Nom de la persona que fa el comentari	string
smileyRating	Valoració amb emoticones, pot ser null	integer, nullable
puntuacion	Puntuació numèrica del comentari	integer
titulo	Títol del comentari	string
comentario	Text del comentari	text
created_at i updated_at	Camps automàtics per a registrar les dates de creació i última actualització del registre	timestamps



Chapter 5. Taules de la Base de Dades de la API Imatges

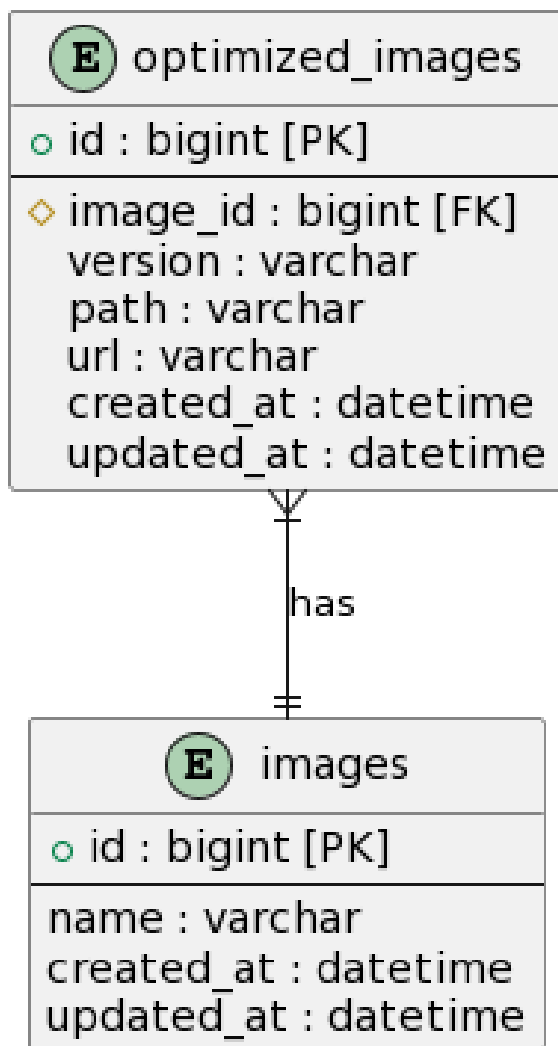


5.1. Images

Camp	Descripció	Tipus de Dada
id	Identificador únic per a cada imatge	bigIncrements
name	Nom de la imatge	string
created_at i updated_at	Camps automàtics per a registrar les dates de creació i última actualització del registre	timestamps

5.2. Optimized_Images

Camp	Descripció	Tipus de Dada
id	Identificador únic per a cada imatge optimitzada	bigIncrements
image_id	Clau forana que referència a la imatge original	unsignedBigInteger
version	Versió de la imatge optimitzada	string
path	Camí on es guarda la imatge optimitzada	string
url	URL on es pot accedir a la imatge optimitzada	string
created_at i updated_at	Camps automàtics per a registrar les dates de creació i última actualització del registre	timestamps



Chapter 6. Sketching i Mockup d'Interfícies

<https://www.figma.com/file/FqrK3TnRAHJla14AuuZgMQ/Grupo4-Puigantell-Carmona-Perez-team-library?type=design&node-id=0-1&mode=design&t=g8761nKUwxHuE5S8-0>

Chapter 7. Preparació del projecte en Local:

Primer de tot fem un clon del nostre repositori de git al terminal del VSC → git clone <https://git.copernic.cat/gpuigantell/gr04-puigantell-carmona-perez.git>

7.1. Preparació del projecte

Accedim al projecte d'Eventopass → cd Eventopass

Ara instal·lem les dependències del composer → composer install

Copiem el fitxer .env.example amb el nom .env → cp .env.example .env

I creem la clau d'Artisan → php artisan key:generate

A continuació, busquem dins de l'arxiu .env del projecte Eventopass la línia que conté → API_PATH=tu/ruta/de/api/aquí

I canviem tu/ruta/de/api/aquí per la teva ruta on estigui la carpeta API, per exemple → C:\Users\Alfred\Desktop\gr04-puigantell-carmona-perez\API

També en l'arxiu .env del Directori Eventopass canviem totes les següents línies per les nostres dades de MAILTRAP i de POSTGRESQL:

- DB_CONNECTION=pgsql
- DB_HOST=127.0.0.1
- DB_PORT=5432
- DB_DATABASE=Project2
- DB_USERNAME=Usuari
- DB_PASSWORD=Contrasenya
- MAIL_MAILER=smtp
- MAIL_HOST=mailtrap
- MAIL_PORT=1025
- MAIL_USERNAME=Usuari MAILTRAP
- MAIL_PASSWORD=Contrasenya MAILTRAP
- MAIL_ENCRYPTION=tls
- MAIL_FROM_ADDRESS="support@eventopass.com"
- MAIL_FROM_NAME="Eventopass"
- MAIL_FROM_NAME="\${APP_NAME}"

7.2. Preparació de l'API d'Imatges

Accedim al projecte d'Eventopass → cd API

Ara instal·lem les dependències del composer → composer install

Copiem el fitxer .env.example amb el nom .env → cp .env.example .env

I creem la clau d'Artisan → php artisan key:generate

A continuació, busquem dins de l'arxiu .env del projecte API, canviem totes les següents línies per les nostres dades de POSTGRESQL per a l'API:

- DB_CONNECTION=pgsql
- DB_HOST=127.0.0.1
- DB_PORT=5432
- DB_DATABASE=Image_Management_API
- DB_USERNAME=Usuari
- DB_PASSWORD=Contrasenya

7.3. Preparació Cordova

Instal·lar Cordova a la teva màquina I NODE.JS, seguir Documentació Oficial

Camp	Descripció
Instal·lació i creació APK	https://cordova.apache.org/ https://cordova.apache.org/docs/en/12.x/guide/platforms/android/index.html
Instal·lació de JDK de Java, SDK d'Android i Gradle	https://cordova.apache.org/docs/en/12.x/guide/platforms/android/index.html

Accedir al CMD de Node.js al buscador → Node.js command prompt

Accedir al directori → Mòbil del projecte

introduir la comanda → cordova build android

esperar a que acabi i es generarà un arxiu .apk i se't mostrarà un enllaç a aquest arxiu, per exemple
→ C:\Users\Alfred\Desktop\gr04-puigantell-carmona-perez\Mòbil\platforms\android\app\build\outputs\apk\debug\app-debug.apk

En aquest enllaç treure el nom de l'arxiu, per exemple → C:\Users\Alfred\Desktop\gr04-puigantell-carmona-perez\Mòbil\platforms\android\app\build\outputs\apk\debug

I accedir a aquest directori

Una vegada dins enviem l'arxiu .apk al nostre mòbil per Whatsapp Web per exemple, instal·lar l'apk al telèfon mòbil

7.4. Preparació BD i servidors API i EventoPass

Obrir una nova Terminal Powershell (l'objectiu és tenir 2 terminals PowerShell al VSC obertes)

En una escrivim la comanda → cd API

Una cop dins d'API introduïm aquesta comanda → php artisan serve --port=8080

després anem a l'altra Terminal PowerShell i introduïm la comanda → cd EventoPass

Una cop dins del Directori EventoPass introduïm la comanda → php artisan api:migrate:refresh

esperem a que acabi i introduïm la comanda → php artisan serve

Accedim des del buscador → <http://127.0.0.1:8000>

Assegurar-nos que a la màquina on hem seguit els passos per obrir els servidors estigui a la mateixa xarxa que el mòbil

Una vegada hem fet això, abans de construir l'apk de l'aplicació amb la comanda → cordova build android, accedir al CMD de windows i introduir la comanda → ipconfig per trobar la IP de la nostra màquina

Assegurar-nos que dins del directori Mòbil accedim a l'arxiu → Mòbil\www\js\index.js i canviem → <https://192.168.43.191:443> per → <https://LA> TEVA IP QUE HAGIS ACONSEGUIT AMB IPCONFIG:443

A continuació seguir amb els passos explicats més amunt per procedir a la creació de l'apk i a la instal·lació al teu telèfon mòbil

Si trobes problemes de permisos realitzant aquestes accions sense sudo, verifica els permisos del directori on estàs treballant. Pot ser que necessitis ajustar els permisos del directori (amb comandes com chown o chmod) per evitar l'ús de sudo per aquestes operacions.

7.5. Altres

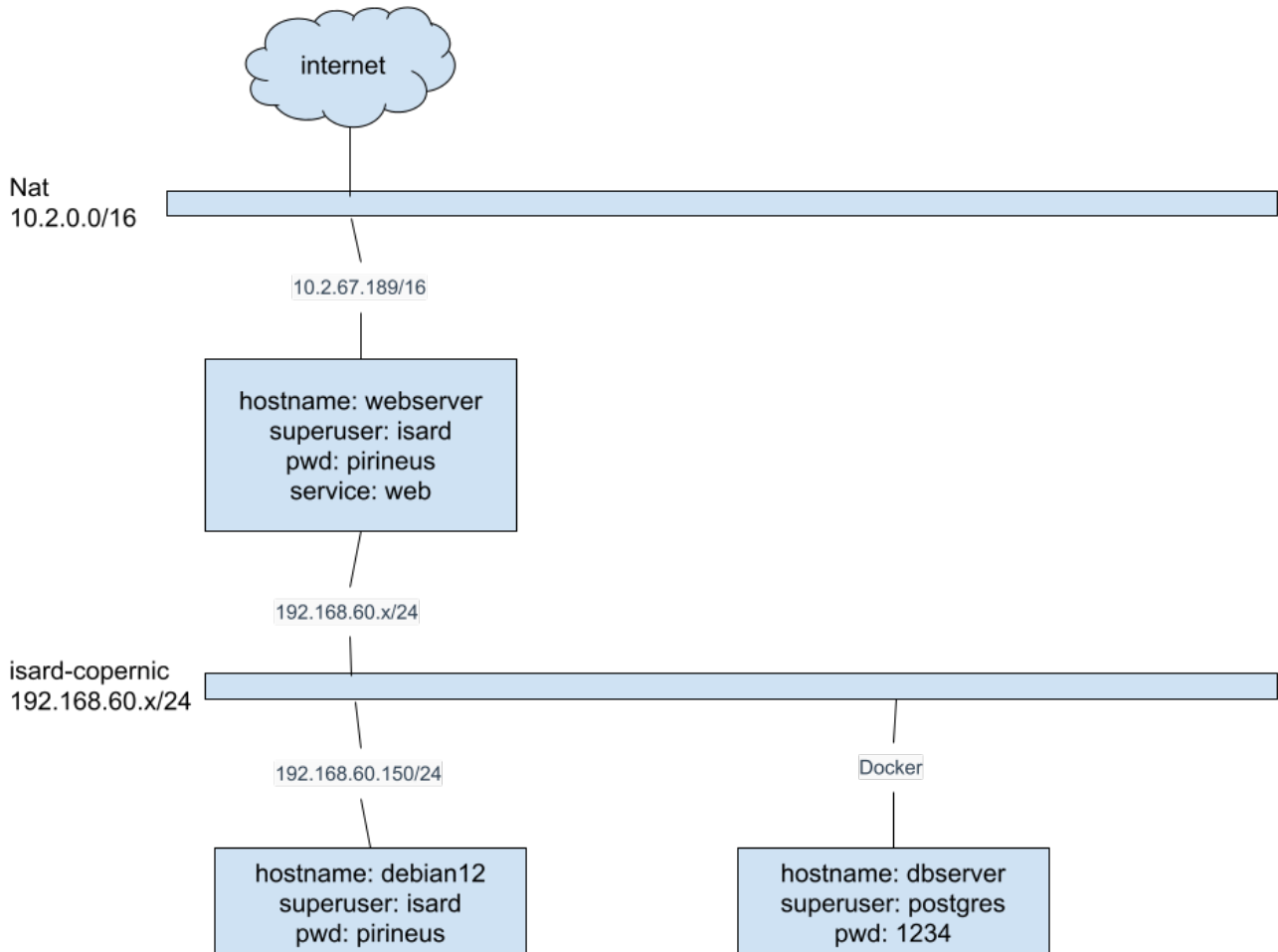
Per a poder fer compres a la pasarela de pagamen s'han d'utilitzar aquestes tarjetes a dia d'avui, si en el moment de fer proves aquestes tarjetes no funcionen es perquè el REDSYS canvia les tarjetes de prova, accedir al següent enllaç → <https://pagosonline.redsys.es/entornosPruebas.html> i provar les tarjetes de l'opció → Datos de prueba para DCC:

Descripción	Tarjeta	Numeración	Caducidad	CVV
Autenticación Frictionless - Tarjeta Master con DCC y EMV3DS	Mastercard	5424180805648190	12/34	123
Autenticación Challenge - Tarjeta Visa con DCC y EMV3DS	Visa	4117731234567891	12/34	123

Chapter 8. Manual d'Instal·lació/Distribució

Pas a pas detallat de com instal·lar i distribuir l'aplicació.

8.1. Diagrama de xarxa:



8.2. Instal·lació del apache2:

Fem un update → `sudo apt update`

Instal·lem l'apache2 → `sudo apt install apache2`

I el podem engegar → `sudo systemctl enable apache2`

8.3. Preparació del servidor:

Primer de tot fem un clone el nostre repositori de git en la ruta `/var/www/html` → `sudo git clone https://git.copernic.cat/gpuigantell/gr04-puigantell-carmona-perez.git`

Dins de `cd /var/www/html/gr04-puigantell-carmona-perez/Eventopass` y dins de `cd /var/www/html/gr04-puigantell-carmona-perez/Api` realitzem aqueste 4 comandes.

Ara instal·lem les dependències del composer → `sudo composer install`

Copiem el fitxer `.env.example` amb el nom `.env` → `sudo cp .env.example .env`

I creem la key del Artisan → `sudo php artisan key:generate`

I fem el link simbòlic del public i el storage → `sudo php artisan storage:link`

8.4. Creació del docker:

Instal·lem les dependències del Docker → `sudo apt install apt-transport-https ca-certificates curl gnupg lsb-release`

Afegim la clau GPG oficial del Docker → `curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg`

Configurem el repositori de Docker → `echo "deb [signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/debian $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null`

Fem un update → `sudo apt update`

Instal·lem el Docker CE → `sudo apt install docker-ce docker-ce-cli containerd.io`

Instal·lem el docker compose → `sudo curl -L "https://github.com/docker/compose/releases/download/VERSION/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose`

I li donem permisos → `sudo chmod +x /usr/local/bin/docker-compose`

Ara hem de crear el fitxer `docker-compose.yml`, així que anem a la ubicació del EventoPass, que hauria de ser → `cd /var/www/html/gr04-puigantell-carmona-perez/EventoPass`

Dins d'aquesta carpeta creem el fitxer → `sudo nano docker-compose.yml`

```
version: '3'

services:
  mi_postgres:
    image: postgres:latest
    container_name: mi_postgres
    environment:
      POSTGRES_USER: postgres
      POSTGRES_PASSWORD: 1234
      POSTGRES_DB: basededatos
    ports:
      - "5555:5432"
```

```
postgres_api:
  image: postgres:latest
  container_name: postgres_api
  environment:
    POSTGRES_USER: postgres
    POSTGRES_PASSWORD: 1234
    POSTGRES_DB: apibd
  ports:
    - "5554:5432"
```

Ara hem d'editar el nostre fitxer .env

Editarem el de la carpeta EventoPass → sudo nano /var/www/html/gr04-puigantell-carmona-perez/EventoPass/.env Les línies que hem de revisar són: url=, URL_API=, API_PATH= i les credencials del mail.

8.5. Aixecament de les bases de dades:

Crearem un servei perquè les bases de dades del Docker s'aixequin sempre a l'engegar el servidor:

Primer creem el fitxer → sudo nano /etc/systemd/system/eventoPass.service

```
[Unit]
Description=Llevantar Docker Compose

[Service]
WorkingDirectory=/var/www/html/gr04-Puigantell-Carmona-Perez/EventoPass
ExecStart=/usr/local/bin/docker-compose -f /var/www/html/gr04-Puigantell-Carmona-Perez/EventoPass/docker-compose.yml up -d
```

Un cop creat executem → "sudo systemctl daemon-reload" després → "sudo systemctl enable eventoPass.service" i → sudo systemctl start eventoPass.service

Reiniciem el servidor per activar i que s'executi el servei.

8.6. Configuració de servidor web:

8.6.1. EventoPass

Creem un fitxer de configuració del apache2 → /etc/apache2/sites-available/eventoPass.conf

Configurem el fitxer amb les nostres rutes i ips:

```
<VirtualHost *:80>
  ServerName 192.168.60.130
  DocumentRoot /var/www/html/gr04-Puigantell-Carmona-Perez/EventoPass/public
```

```
<Directory /var/www/html/gr04-Puigantell-Carmona-Perez/EventoPass>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>

</VirtualHost>
```

Habilitem el nostre virtual host → `sudo a2ensite eventoPass.conf` Reiniciem el servei apache2 → `sudo systemctl restart apache2`

8.6.2. API

Creem un fitxer de configuració del apache2 → `/etc/apache2/sites-available/api.conf`

Configurem el fitxer amb les nostres rutes i ips:

```
<VirtualHost *:8080>
    ServerName 192.168.60.130
    DocumentRoot /var/www/html/gr04-Puigantell-Carmona-Perez/API/public

    <Directory /var/www/html/gr04-Puigantell-Carmona-Perez/API>
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

</VirtualHost>
```

Habilitem el nostre virtual host → `sudo a2ensite eventoPass.conf` Reiniciem el servei apache2 → `sudo systemctl restart apache2`

I habilitem perquè apache pugui escoltar en el port 8080:

`sudo nano /etc/apache2/ports.conf`

Aquí afegim → `Listen 8080`

Chapter 9. Documentació de Permisos

Descripció detallada dels permisos necessaris per a components, programes, contenidors, etc.

Hem de donar permisos a la carpeta de logs, framework i public amb aquestes tres comandes →
`sudo chown -R www-data:www-data /var/www/html/gr04-Puigantell-Carmona-Perez/EventoPass/storage/framework` , `sudo chown -R www-data:www-data /var/www/html/gr04-Puigantell-Carmona-Perez/EventoPass/storage/logs` y `sudo chown -R www-data:www-data /var/www/html/gr04-Puigantell-Carmona-Perez/EventoPass/public`

tambe hem de fer le mateix amb l'API → `sudo chown -R www-data:www-data /var/www/html/gr04-Puigantell-Carmona-Perez/API/storage/framework` , `sudo chown -R www-data:www-data /var/www/html/gr04-Puigantell-Carmona-Perez/API/storage/logs` y `sudo chown -R www-data:www-data /var/www/html/gr04-Puigantell-Carmona-Perez/API/public`

Per acabar anem la ruta de l'API → `cd /var/www/html/gr04-Puigantell-Carmona-Perez/API` y dins executem → `sudo php artisan migrate:refresh`

Fem el mateix en EventoPass → `cd /var/www/html/gr04-Puigantell-Carmona-Perez/EventoPass` y dins executem → `sudo php artisan migrate:refresh`

I ja tindriem el nostre servidor operatiu.

Chapter 10. Instal·lació del DomPDF

Primer instal·lem el paquet DomPdf amb el composer -> composer require barryvdh/laravel-dompdf

En l'arxiu config/app.php a l'apartat de 'providers' fem la següent línia -> Barryvdh\DomPDF\ServiceProvider::class

I publiquem el paquet per a poder modificar-lo -> php artisan vendor:publish --provider="Barryvdh\DomPDF\ServiceProvider"

Chapter 11. Descripció General de l'API

L'API de Gestió d'Imatges permet la càrrega, optimització i gestió d'imatges. Aquesta API és accessible a través de la base URL <http://localhost:8080/api/V1> i està dissenyada per oferir una forma eficient de manejar imatges en aplicacions web i mòbils.

11.1. Informació General

- **Títol:** Image Management API
- **Descripció:** API per a la càrrega, optimització, i gestió d'imatges.
- **Versió:** 1.0.0

11.2. Servidors

- URL: <http://localhost:8080/api/V1>
- Descripció: URL base de l'API

11.3. Rutes de l'API

11.3.1. Operacions sobre Imatges **/images**

GET

Llista totes les imatges.

- Resum: Retorna una llista de totes les imatges disponibles.
- Respostes:
 - **200**: Llista d'imatges obtinguda amb èxit.

Esquema:

```
{
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "id": {
        "type": "integer",
        "example": 1
      },
      "name": {
        "type": "string",
        "example": "image.jpg"
      }
    }
  }
}
```

```
}
```

POST

Carrega una nova imatge i crea les seves versions optimitzades.

- Resum: Permet la càrrega d'una nova imatge.
- RequestBody: Requirit, **multipart/form-data**.
- Respostes:
 - **201**: Imatge carregada i optimitzada correctament.
 - **400**: Sol·licitud invàlida.
 - **415**: Tipus de mitjà no suportat.
 - **500**: Error intern del servidor.

11.3.2. Operacions sobre una Imatge Específica **/images/{id}**

GET

Obté una imatge específica pel seu ID.

- Resum: Retorna una imatge específica.
- Paràmetres: **id** - El ID de la imatge a obtenir.
- Respostes:
 - **200**: Imatge obtinguda amb èxit.
 - **404**: Imatge no trobada.

DELETE

Elimina una imatge específica i les seves versions optimitzades.

- Resum: Elimina la imatge especificada pel ID.
- Paràmetres: **id** - El ID de la imatge a eliminar.
- Respostes:
 - **204**: Imatge eliminada amb èxit.
 - **404**: Imatge no trobada.

POST

Actualitza una imatge específica i les seves versions optimitzades.

- Resum: Actualitza la imatge especificada pel ID.
- RequestBody: Requirit, **multipart/form-data**.
- Respostes:
 - **200**: Imatge actualitzada amb èxit.
 - **400**: Sol·licitud invàlida.

- **404**: Imatge no trobada.
- **415**: Tipus de mitjà no suportat.
- **500**: Error intern del servidor.

11.3.3. Operacions sobre Imatges Optimitzades

`/optimized-images/{image_id}/{version}`

GET

Obté una versió optimitzada específica d'una imatge.

- Resum: Retorna una versió optimitzada de la imatge.
- Paràmetres:
 - **image_id** - El ID de la imatge.
 - **version** - La versió de la imatge optimitzada a obtenir (small, medium, large).
- Respostes:
 - **200**: Versió optimitzada de la imatge obtinguda amb èxit.
 - **404**: Imatge o versió optimitzada no trobada.
 - **500**: Error intern del servidor.

Chapter 12. Autenticació

12.1. Token d'Aplicació

La verificació del token de l'aplicació és requerida per a realitzar operacions d'escriptura.

- Headers requerits per a les operacions POST, DELETE, i POST sobre `/images/{id}`:
 - `Authorization: Bearer {app_token}`

Chapter 13. Codis d'Estat d'Error Comuns

- **400** - Sol·licitud invàlida. Pot ser degut a dades d'entrada incorrectes.
- **404** - Recurs no trobat. L'ID especificat no correspon a cap recurs existent.
- **415** - Tipus de mitjà no suportat. L'arxiu carregat no és una imatge o supera la mida màxima permesa.
- **500** - Error intern del servidor. Error inesperat en el servidor.

Chapter 14. API de Validació de Tickets

La API de Validació de Tickets permet la verificació de codis de sessió i la validació de tickets dins de sessions específiques. Aquesta API és part del sistema de gestió de tickets i proporciona una interfície per a la validació segura i eficient dels tickets.

14.1. Rutes de l'API

14.2. Validació de Sessió **/login**

14.2.1. POST

Login amb codi de sessió.

- Resum: Maneja el procés de login verificant el codi de sessió. Permet el login si la sessió amb el codi proporcionat està tancada.
- Paràmetres de la sol·licitud: **session_code** - Requerit, codi de la sessió a verificar.
- Headers requerits: No aplica.
- Respostes:
 - **200**: Login exitós.
 - **401**: Codi de sessió invàlid o la sessió no està tancada per a nous logins.

Esquema de la resposta:

```
{
  "success": true,
  "message": "Login exitoso.",
  "session_code": "CODI_DE_SESSIÓ"
}
```

14.3. Validació de Ticket **/tickets/validate/{sessionId}/{hash}**

14.3.1. GET

Obté informació i valida un ticket.

- Resum: Obtiene la información de un ticket basado en el ID de sesión y un hash único. Valida si el ticket aún no ha sido validado y, de ser así, lo marca como validado.
- Paràmetres de la sol·licitud:
 - **sessionId** - Identificador de la sessió.
 - **hash** - Hash únic del ticket.
- Headers requerits: **Session-Code** - Codi de sessió per a la verificació de la sessió actual.

- Respostes:
 - 200: Ticket validat correctament.
 - 403: La sessió no coincideix o no es va trobar.
 - 404: Ticket no trobat.
 - 409: El ticket ja ha estat validat.

Esquema de la resposta:

```
{
  "success": true,
  "message": "Ticket validado correctamente.",
  "ticket_info": {
    "name": "NOM_DEL_PROPRIETARI",
    "dni": "DNI_DEL_PROPRIETARI",
    "phone": "TELÈFON_DEL_PROPRIETARI"
  }
}
```

Chapter 15. Middleware Utilitzat a l'API de Validacion de Tickets

- `verifySessionCode`: Verifica que el codi de sessió proporcionat en la capçalera de la petició sigui vàlid. Utilitzat per a totes les operacions que requereixen verificació de sessió.

Chapter 16. Web-grafia i Altres Recursos de Consulta

Durant el desenvolupament, s'han utilitzat diversos recursos i eines. A continuació, es detallen alguns dels més rellevants:

16.1. R18: Passarel·la de pagament

- **RedSys Sandbox - Versió REST:** Una plataforma per a la implementació de pagaments amb targeta de crèdit.
 - URL: <https://pagosonline.redsys.es/conexion-rest.html>

16.2. R28: Càrrega de recursos optimitzada

- ** i l'atribut loading:** Permet carregar imatges de manera diferida millorant el rendiment de la pàgina.
 - Guia MDN: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/img#attr-loading>
- **<picture>:** Utilitzat per a especificar múltiples fonts d'imatge o per a oferir versions d'una imatge per a diferents dimensions de pantalla.
 - Guia MDN: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/picture>
- **<source> i el srcset:** Permet definir diverses fonts d'imatge que el navegador pot triar entre elles basant-se en les dimensions de la pantalla i la densitat de píxels.
 - Guia MDN: https://developer.mozilla.org/en-US/docs/Learn/HTML/Multimedia_and_embedding/Responsive_images

16.3. R29: Metadades i compartició a xarxes socials

- **Metadata en HTML:** Important per a un bon reconeixement per part dels cercadors i per a la compartició en xarxes socials.
 - Article MDN: https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/The_head_metadata_in_HTML

16.4. R30: Microservei d'optimització d'imatges

- **Docker:** Permet crear una API RESTful en un contenidor aïllant la responsabilitat del sistema.
 - Web oficial: <https://www.docker.com/>
- **OpenAPI:** Eina per a la documentació d'APIs que facilita la descripció dels endpoints, mètodes i paràmetres.
 - Web oficial: <https://www.openapis.org/>

16.5. R31: Proves de l'API amb Postman

- **Postman:** Eina per a la prova i documentació d'APIs que permet enviar peticions HTTP i revisar les respostes.
 - Web oficial: <https://www.postman.com/>

16.6. R35: Creació d'aplicació per a validació d'entrades

- **Cordova:** Framework per al desenvolupament d'aplicacions mòbils utilitzant HTML, CSS i JavaScript.
 - Web oficial: <https://cordova.apache.org/>

Chapter 17. Presentació del Projecte

El projecte EventoPass serà presentat mitjançant una demostració en directe on es mostraran les funcionalitats clau de la plataforma, incloent la gestió d'esdeveniments, la compra i validació d'entrades, i l'ús de l'API de gestió d'imatges.

Chapter 18. Conclusions

En aquest projecte esperàvem aprendre a com fer una web utilitzant Laravel i Postgres majoritàriament. El qual hem après, i també com crear una api independent del projecte i fer-la servir en la nostra web.

Ens ha servit per aprendre més PHP, HTML, JS, CSS, SASS i el desplegament d'una web en un servidor amb Debian12 i Apache2.