

REQUISITS BÀSICS PER L'EXECUCIÓ DE L'USUARI

La nostra pàgina web el que fa és un registre d'incidències, en el qual tenim els **Usuaris**, els **Tècnics** i l'**Administrador**. Cadascun té unes funcions diferents.

L'Usuari pot **crear una incidència** amb el seu nom, el nom del departament i la seva descripció de la incidència. També pot **veure les incidències**, encara que no estiguin assignades a un tècnic.

El Tècnic pot **veure el llistat de les incidències** ordenades per prioritat, de més alta a més baixa, en el qual només apareixen les incidències que estan assignades a algun tècnic. També pot **Esborrar una incidència**, **afegir Accions** sobre el que ha fet en cada moment per resoldre la incidència, indicant la descripció del que ha fet, el temps que ha durat i l'estat d'actuació (si està fet, en procés o no fet). També pot **veure les Actuacions** que ha anat realitzant, posant també el dia actual en què ha fet cada acció.

L'Administrador pot fer el mateix que el Tècnic, però l'única diferència és que té una taula separada en la qual pot **Assignar els Tècnics a les incidències** i establir també la prioritat de cada incidència (alta, mitjana o baixa).

GUIA PAS A PAS DE DOCKER

El primer pas consisteix en crear el fitxer **docker-compose.yml**, el qual s'encarregarà de definir i desplegar tres contenidors: un servidor **Apache amb PHP**, una base de dades **MySQL** i una interfície d'administració mitjançant **Adminer**.

L'objectiu d'aquesta infraestructura és implementar una aplicació web per al registre i gestió d'incidències.

Per començar, es defineixen els serveis dins de l'apartat **services**: del fitxer **docker-compose.yml**, on cada servei representa un contenidor.

El primer servei, anomenat **web**, correspon al contenidor que allotja **Apache i PHP**. Aquest es construeix mitjançant la directiva **build**, la qual inclou el **context** —que indica la ubicació del directori arrel del projecte— i el nom del fitxer **Dockerfile** que s'utilitzarà per generar la imatge del contenidor.

A continuació, es defineixen els **ports** a través de l'apartat **ports**, on el valor a l'esquerra indica el port del sistema host que s'utilitzarà per accedir al servei mitjançant el navegador, i el valor a la dreta especifica el port exposat pel servidor **Apache** dins del contenidor.

L'apartat **volumes** permet mapar el directori **/src** del projecte local al directori per defecte que el servidor **Apache** utilitza per servir els fitxers web, assegurant així que la pàgina web s'executi correctament amb el contingut desitjat.

Finalment, es fa servir la directiva **depends_on** per establir la dependència del servei web respecte al servei **db**, que és el contenidor de **MySQL**, garantint que aquest últim estigui disponible abans d'iniciar el servidor web.

```
services:
  web:
    #La imatge php:apache no conté els drivers de mysql, hem de fer una imatge a mida
    #image: php:apache
    build:
      context: ./images
      dockerfile: Dockerfile_php
    ports:
      - "8080:80"
    volumes:
      - ./src:/var/www/html
    depends_on:
      - db
```

Una vegada fet això, el següent pas serà crear el contenidor de **MySQL**, anomenat **db**.

Es defineix la directiva **image:**, on es especifica la imatge que es vol descarregar. En aquest cas, utilitzarem **mysql:9.3**.

A continuació, es troba l'apartat **environment:**, on es defineixen els paràmetres principals per configurar la base de dades (no es definirà el nom de la base de dades, ja que es crearà mitjançant l'**script** que es troba més endavant, a l'apartat de **volumes**). El primer valor que es defineix és **MYSQL_ROOT_PASSWORD**, el qual no modificarem ja que no el farem servir. Després es defineix **MYSQL_USER**, que serà l'usuari amb el qual accedirem a la base de dades. Finalment, es defineix **MYSQL_PASSWORD**, que és la contrasenya de l'usuari creat anteriorment. Dins de l'apartat **environment** també es troba **LANG**, que no es modifica.

Per últim, en l'apartat **volumes** de **db**, es defineixen dues línies. La primera, **./db_data:/var/lib/mysql**, on la ruta de la dreta és la ruta per defecte dins del contenidor, i la de l'esquerra correspon a la ruta en el nostre projecte, la qual s'utilitza per emmagatzemar totes les dades de la base de dades. La segona línia, **./db_init:/docker-entrypoint-initdb.d**, fa referència a la ruta on es troba l'script per crear la base de dades.

```
db:
# Tota la informació d'aquesta imatge està https://dockerhub.com/ /mysql
image: mysql:9.3
environment:
  MYSQL_ROOT_PASSWORD: passwordDeRootQueNoShaDeFerServirMai
  # És millor no crear la BBDD aquí, ja que no pots controlar la codificació
  # de caràcters i aleshores donarà problemes amb accents i caràcters especials
  # La BBDD es crearà a l'inici del contenidor amb els script d'inicialització
  # MYSQL_DATABASE: persones
  MYSQL_USER: usuari
  MYSQL_PASSWORD: paraula_de_pas
  LANG: C.UTF-8
# El mysql no s'exposa a l'exterior
# L'aplicació web hi accedirà per la xarxa interna de docker
# ports:
#   - "3306:3306"

# La carpeta de mysql ha d'estar al .gitignore per no pujar-la al repositori
volumes:
  - ./db_data:/var/lib/mysql
  - ./db_init:/docker-entrypoint-initdb.d
```

Una vegada fet el **MYSQL**, passarem al **Adminer**.

Per començar, definirem el contenidor **adminer**, com es va fer anteriorment. Un cop fet això, afegirem l'apartat **image** amb el paquet adminer. Seguidament, definirem l'apartat **ports**:, on per accedir-hi de manera local haurem d'utilitzar el port que assignem, que serà el port de l'esquerra, ja que el de la dreta és el port per defecte que utilitza **adminer**. Per últim, afegirem el paràmetre **depends_on**: i definirem el contenidor de **MYSQL**, que es **db**.

```
# Aquesta imatge és un client web per gestionar el mysql via web
# està disponible a http://localhost:8081
# i les credencials són les del mysql (per que "simplement" es connecta al mysql)
adminer:
  image: adminer
  ports:
    - "8081:8080"
  depends_on:
    - db
```

Després d'això, el fitxer **docker-compose.yml** hauria de crear l'estructura de directoris, on tindrem la carpeta **Apache/PHP**, que en el nostre cas és **./images**. A continuació, tenim les dues carpetes per a **MYSQL**. La primera és la carpeta on es guarden les dades de la base de dades, que ha d'estar buida; en el nostre cas és **./db_data**. La segona carpeta conté l'**script** per crear la base de dades, que en el nostre cas és **./db_init**.

A continuació, tenim el contenidor d'**adminer**, però aquí no cal crear cap carpeta, ja que només s'utilitzarà per veure i modificar les taules de la base de dades. En el nostre cas, utilitzarem el **port :8081**. Finalment, per accedir a la nostra pàgina web de manera local, utilitzarem el **port :8080**, que es el que hem definit en el contenidor.

COM CONECTAR L'APLICACIÓ A LA BD

Per connectar l'aplicació amb la base de dades hem de crear un arxiu PHP que permet la connexió.

El primer que hem de posar són les variables de:

Servername: en el qual posem la IP/domini on es troba la base de dades.

Username: el qual posem l'usuari que hem creat a Hestia quan van crear la base de dades.

Password: la contrasenya que vam posar al crear la base de dades a Hestia.

dbname: el nom que hem donat a la base de dades quan la vam crear a Hestia.

Després d'això, crearem una variable de conn la qual es connectarà al MySQL utilitzant les credencials que hem posat a les variables anteriors.

Un cop feta la connexió, posarem un if per si no funciona la connexió, que ens mostri un error.

```
connection.php X
src > connection.php
1  <?php
2
3  $servername = "daw.inspedralbes.cat";
4  $username = "a24ikematgar_incidencia";
5  $password = ":8VBky{Uq9D^W-xk";
6  $dbname = "a24ikematgar_incidencia";
7
8
9  $conn = new mysqli($servername, $username, $password, $dbname);
10
11
12  if ($conn->connect_error) {
13      die("Error de conexió: " . $conn->connect_error);
14  }
15  ?>
16
```

Per acabar, haurem de posar en tots els fitxers PHP que necessitin registrar/borrar/consultar una taula o dades de la base de dades, haurem de posar a la primera línia del PHP el següent:

```
require_once 'connection.php';
```

PROCEDIMENT SENZILL PER A LES CÒPIES DE SEURETAT

Fer còpies de seguretat dels arxius i la base de dades d'una pàgina web de registre i resolució d'incidències per garantir la seguretat i possibilitar la restauració en cas de pèrdua de dades.

Pas 1: Còpia dels arxius

Connecta't al servidor mitjançant FTP (amb FileZilla) o SSH.

Descarrega els arxius del directori principal de la pàgina web, que inclou els arxius PHP, CSS, JavaScript i les imatges relacionades amb les incidències.

Exemple: Arxius com index.php, incidencies.php, reGISTRE.php, etc.

Pas 2: Còpia de la base de dades

Accedeix a la base de dades on es guarden les incidències (com MySQL o PostgreSQL) mitjançant phpMyAdmin o accedint directament a MySQL.

Exporta la base de dades en format SQL, que contindrà taules com:

Usuaris: Per emmagatzemar informació dels usuaris (nom, correu, etc.).

Incidències: Per emmagatzemar les incidències registrades pels usuaris.

Tècnics: Per emmagatzemar la informació dels tècnics assignats a cada incidència.

Historial d'actuacions: Per guardar les accions realitzades per cada tècnic.

Exemple: Descarregar un fitxer incidencies_backup_2025-05-14.sql.

Pas 3: Emmagatzematge

Desa tant els arxius de la web com la base de dades en un lloc segur:

Arxius: Disc dur extern, servidor de còpies de seguretat o núvol.

Base de dades: Emmagatzema el fitxer SQL en un lloc segur.

Organitza les còpies amb noms clars i amb la data, com per exemple:

backup-web-incidencies-2025-05-14/

backup-db-incidencies-2025-05-14.sql

Pas 4: Verificació

Verifica els arxius: Assegura't que tots els arxius de la web s'hagin descarregat correctament.

Verifica la base de dades: Intenta restaurar el fitxer de la base de dades en un entorn de prova per assegurar-te que és vàlid i complet.

Freqüència de les còpies de seguretat

Realitza còpies setmanals o diàries, depenent de la quantitat d'incidències noves i de les actualitzacions de la web.

Exemple: Si els tècnics actualitzen les incidències diàriament, és recomanable fer còpies diàries.

RESOLUCIO DE PROBLEMAS COMUNS

PROBLEMAS MAS INTERNOS

PREGUNTAS FRECUENTES

AL REGISTRAR UNA INCIDENCIA NO TIENE LA MISMA BASE DE DATOS
CONNECTION.PHP

NO ENTRA CON LOS PUERTOS AL LOCALHOST