

**PORTAL D’OCUPACIÓ**

**JOB SENIOR**

Bernat Guerola Trull

Projecte fi de cicle 2DAM (DESENVOLUPAMENT D’ APLICACIONS MULTIPLATAFORMA)

24 de setembre de 2020



Datos de portada: Título del Proyecto (tamaño letra 36), nombre del autor, curso, año, ciclo formativo, ), nombre del centro y del tutor individual (tamaño letra 20). Logotipos del centro y de la familia profesional.

Información contraportada: resumen del contenido (en cuadro de texto, tamaño de letra 16).

Primera página: título del proyecto y nombre del autor.

Formato de página: márgenes no superiores a 3 cm, tipo de letra (Arial, Times New Roman, o equivalente, tamaño 11 ó 12), interlineado sencillo. Encabezado: título proyecto. Pie de página: nombre alumno, número de página.

Imágenes insertadas: tamaño adecuado a la información que contenga y a su nitidez. Debe llevar pie de foto, con orden numérico e indicación del contenido. Todas estarán incluidas en una carpeta del CD. Indicación del Copyright.

Tamaño del contenido del documento: un mínimo de 20 páginas y/o 5000 palabras. Un máximo que no exceda del 50% adicional.

Segunda página: índice (incluyendo anexos), con apartados y subapartados con formato numérico. Paginación de los apartados. El apartado introducción incluirá el resumen del contenido.

Contingut

[1. PRESENTACIÓ – JUSTIFICACIÓ 4](#_Toc52623934)

[2. CONTINGUT 5](#_Toc52623935)

[3. FERRAMENTES 6](#_Toc52623936)

[3.1. PARTE SERVIDOR 6](#_Toc52623937)

[3.2. PARTE CLIENTE WEB 7](#_Toc52623938)

[3.3. TECNOLOGÍAS ANDROID 8](#_Toc52623939)

[3.4. TECNOLOGÍAS COMUNES 9](#_Toc52623940)

[4. DESENVOLUPAMENT 11](#_Toc52623941)

[4.1. CONTENIDOR DOCKER 11](#_Toc52623942)

[4.2. DISENYANT MYSQL 13](#_Toc52623943)

[4.3. INICIALITZANT SPRING 23](#_Toc52623944)

[4.4. CONFIGURANT HIBERNATE 23](#_Toc52623945)

[4.5. DISENYANT ANDROID 23](#_Toc52623946)

[4.6. ACCEDINT A LES DADES 23](#_Toc52623947)

[5. RECURSOS 24](#_Toc52623948)

[6. AUTOAVALUACIÓ I CONCLUSIONS 25](#_Toc52623949)

[7. INDICE DE IMÁGENES Y CONTENIDOS USB 26](#_Toc52623950)

[8. BIBLIOGRAFIA 27](#_Toc52623951)

[9. ANNEXOS 28](#_Toc52623952)

# PRESENTACIÓ – JUSTIFICACIÓ

fonamentació teòrica.

Contextualització, destacant les aportacions i peculiaritats que el treball ofereix, i la possibilitat de comercialització/materialització real de l'obra o treball.

# CONTINGUT

# FERRAMENTES

## PARTE SERVIDOR



**Spring** es un framework para el desarrollo de aplicaciones y contenedor de inversión de control, de código abierto para la plataforma Java. Con él se creara el programa servidor.



MYSQL es un sistema de administración de bases de datos relacionales. Una base de datos relacional almacena los datos en tablas separadas en lugar de poner todos los datos en un solo lugar. MySQL WORKBENCH será el programa utilizado para gestionar las conexiones con la base de datos, insertar, actualizar datos en tablas



**Hibernate** es una herramienta de mapeo objeto-relacional (ORM) para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) o anotaciones en los beans de las entidades que permiten establecer estas relaciones. Se aplicará en Spring Suite.

## PARTE CLIENTE WEB



La parte cliente web se realiza con **HTML5** es un lenguaje de marcado que nos permite indicar la estructura de nuestro documento mediante etiquetas. **CSS** «Hojas de estilo en cascada», es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado.



**Bootstrap** es una biblioteca multiplataforma o conjunto de herramientas de código abierto para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño. Solo se ocupa del desarrollo front-end.



**Thymeleaf** es un motor de plantillas de XML/XHTML/HTML5 .Se integra con Spring MVC, por lo que se puede utilizar para reemplazar completamente a los archivos JSP. El objetivo principal de Thymeleaf es permitir la creación de plantillas de una manera elegante y un código bien formateado y hacer la comunicación con los datos.

## TECNOLOGÍAS ANDROID



**Android** es un sistema operativo móvil desarrollado por Google, basado en núcleo Linux y otros software de código abierto. **Android Studio** es el entorno de desarrollo integrado oficial para la plataforma Android. **Java** es el lenguaje de programación utilizado.

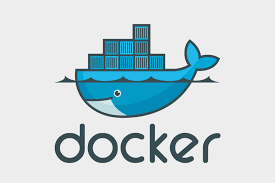


**SQLite** es un sistema de gestión de bases de datos relacional, contenida en una relativamente pequeña. La biblioteca SQLite se enlazara con el programa pasando a ser parte integral del mismo. Por este motivo se utilizará en la aplicación Android



RETROFIT es una librería para hacer llamadas red y obtener el resultado estructurado de una vez. Simplifica la generación de peticiones.

## TECNOLOGÍAS COMUNES



**Docker** es un proyecto de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de virtualización de aplicaciones en múltiples sistemas operativos. En él alojaremos la base de datos MySQL en la fase de diseño.



**GitHub** es una plataforma de desarrollo colaborativo) para alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de ordenador. En el se alojará el proyecto entero en dos partes: programa servidor y el programa Android.

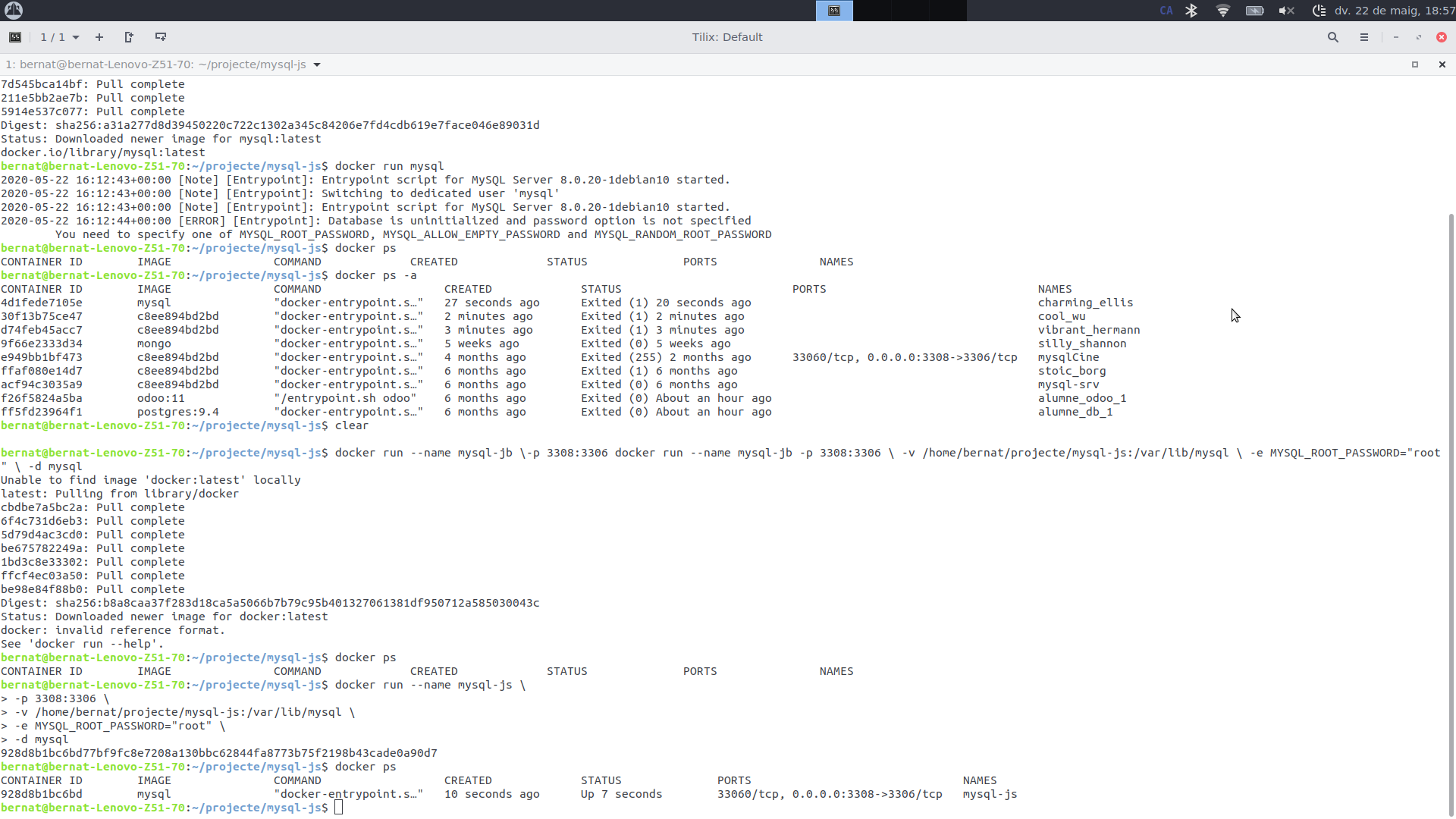
# DESENVOLUPAMENT

Descripció del procés i de les fases de realització,

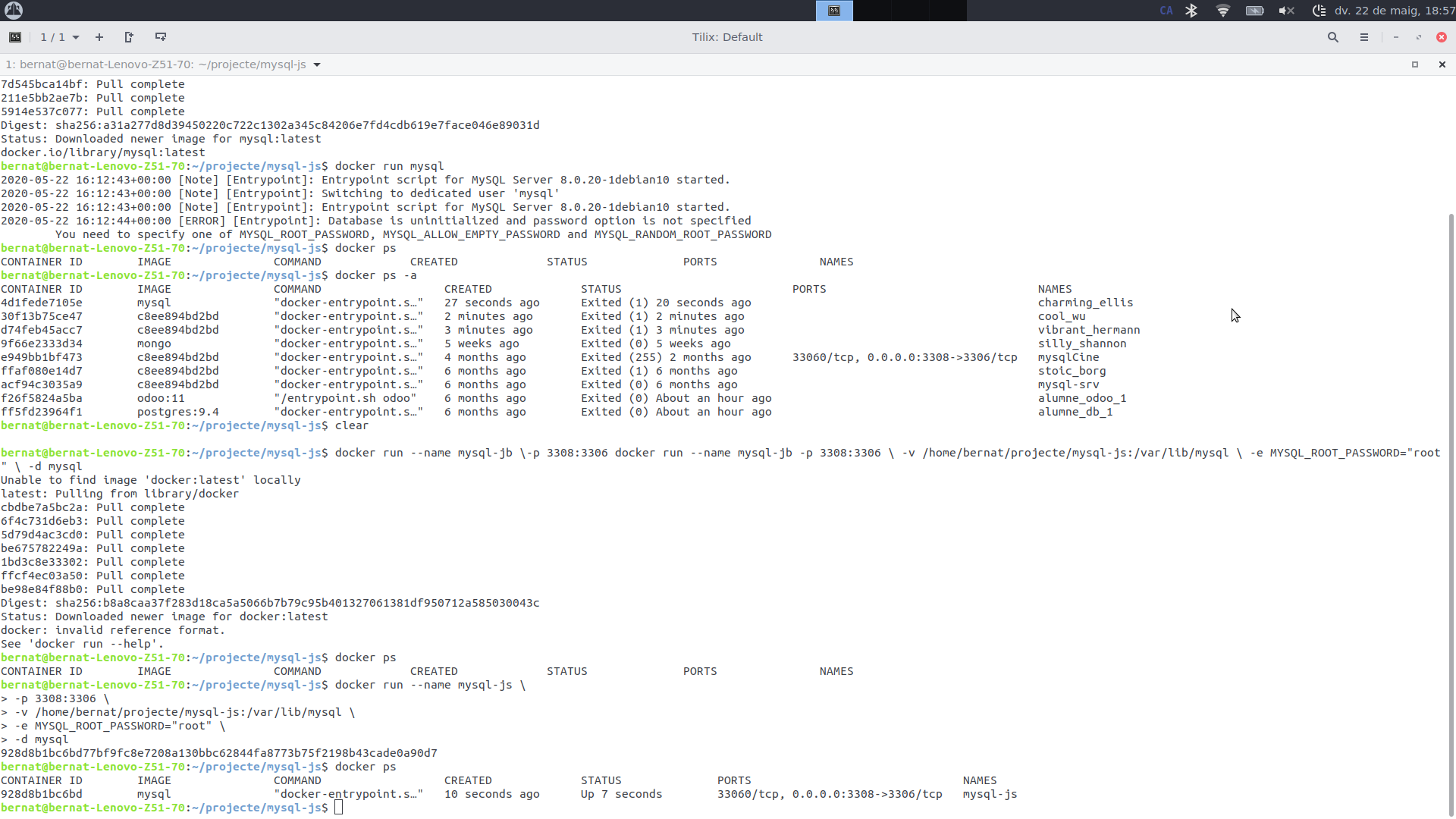
descripció del resultat final, tenint en compte els aspectes funcionals, artístics, tècnics i econòmics.

## CONTENIDOR DOCKER

L’ordre docker ps serveix per veure els contenidors que s’estan executant en un moment donat. Si fem en un terminal:



Amb l’ordre docker ps -a podem veure els contenidor que tenim disponibles



Anem a treballar amb la imatge oficial del SGBD MySQL (https://hub.docker.com/\_/mysql). Per talde descarregar-nos la imatge més recent del SGBD, farem:

1 $ docker pull mysql

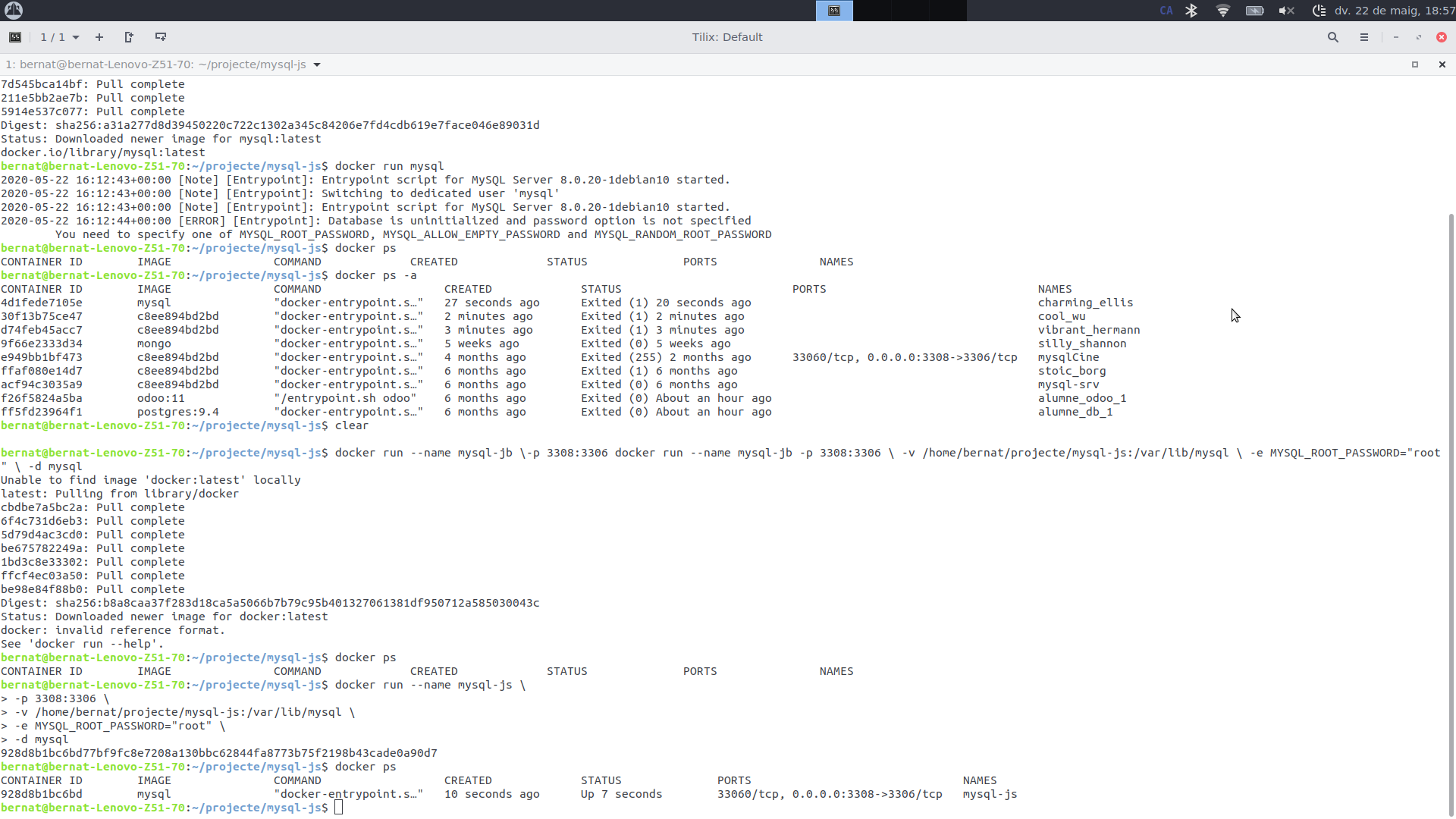
Una vegada descarregada la imatge, podem llançar-la amb un simple docker run mysql, però no tindríem el servei disponible des del nostre equip, i les seues dades desapareixeríen quan esborràrem el contenidor.

Per tal d’ferir persistència, haurem d’enllaçar la carpeta on guarda MySQL tota la informació amb una carpeta del nostre equip, coneguda en terminologia de Docker com a *volum*.

Per a això, crearem una carpeta, per exemple a /srv/mysql-data:

1 $ sudo /var/lib/mysql

I ara llançarem un contenidor amb la següent ordre (les \ són per indicar canvi de línia en *bash*, però no s’han de posar si ho escrivim tot a una línia):



Les opcions que hem utilitzat han estat:

• --name mysql-js: Li donem un nom (*mysql-js*) al contenidor, per a quan hajam d’aturar-lo o eliminar-lo referir-nos a ell de forma més senzilla.

• -p 3308:3306: Aci fem el que es coneix com *exposició de ports*, és a dir, *exposem* els ports pels que treballa el contenidor per defecte a través del ports del nostre equip. En aquest cas, el port per defecte de MySQL (3306) del contenidor, estarà visible en la nostra màquina a través del port 3308.

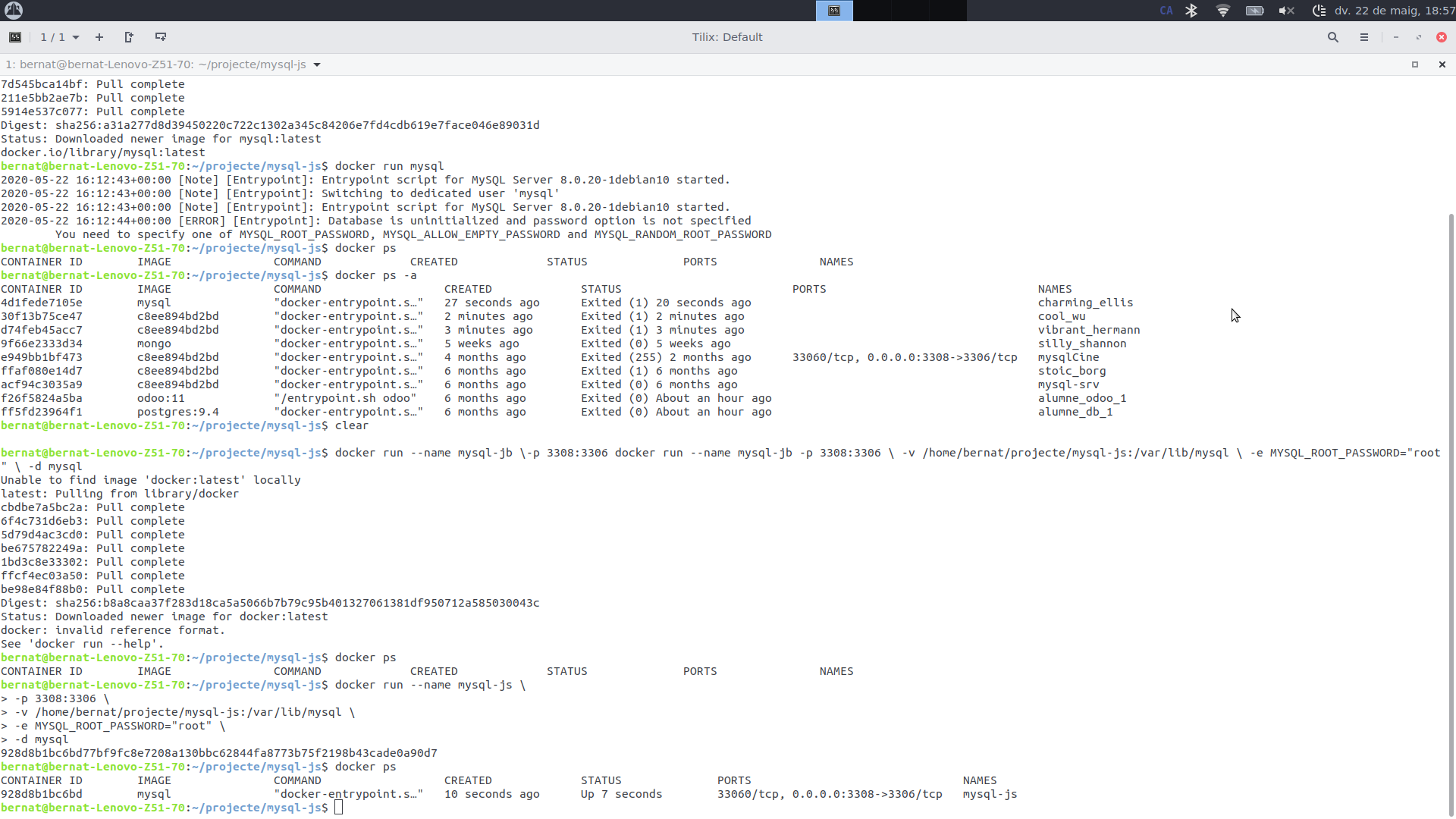
• -v /home/bernat/projecte/mysql-js:/var/lib/mysql: Enllacem el volum que acabem de crear (la carpeta /srv /mysql), amb la carpeta /var/lib/mysql, que és on el servidor de MySQL emmagatzema les dades.

• -e MYSQL\_ROOT\_PASSWORD="root": Amb -e establim variables d’entorn. En aquest cas, estem establint el valor de la variable MYSQL\_ROOT\_PASSWORD (és a dir, la contrassenya de *root*) com a root.

• -d: Indica que anem a llançar el contenidor en background, sense que ens mostre tots els missatges de log per pantalla.

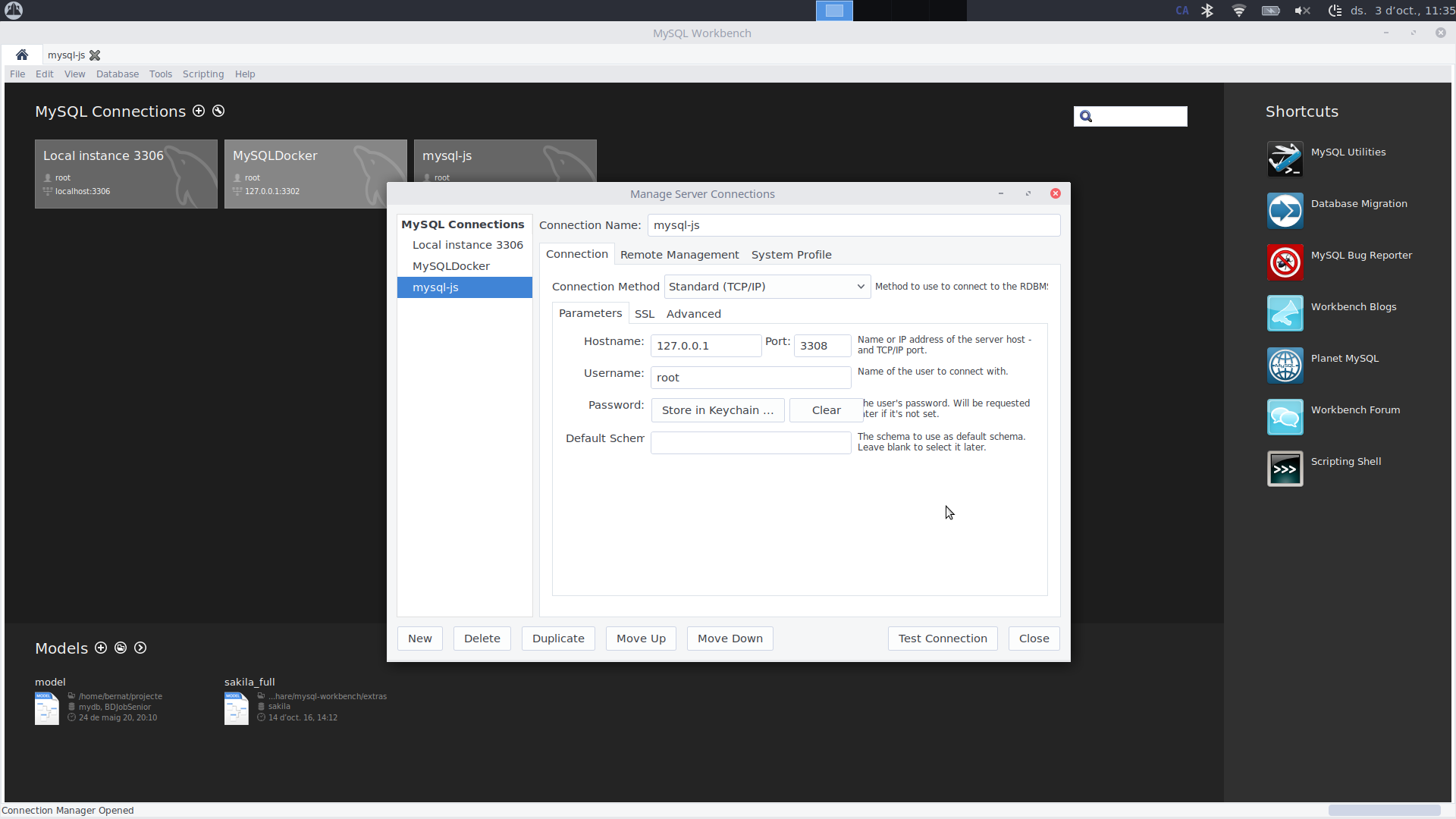
I finalment, hem indicat el nom de la imatge mysql a llançar.

Amb açò, si fem un docker ps, obtindrem:

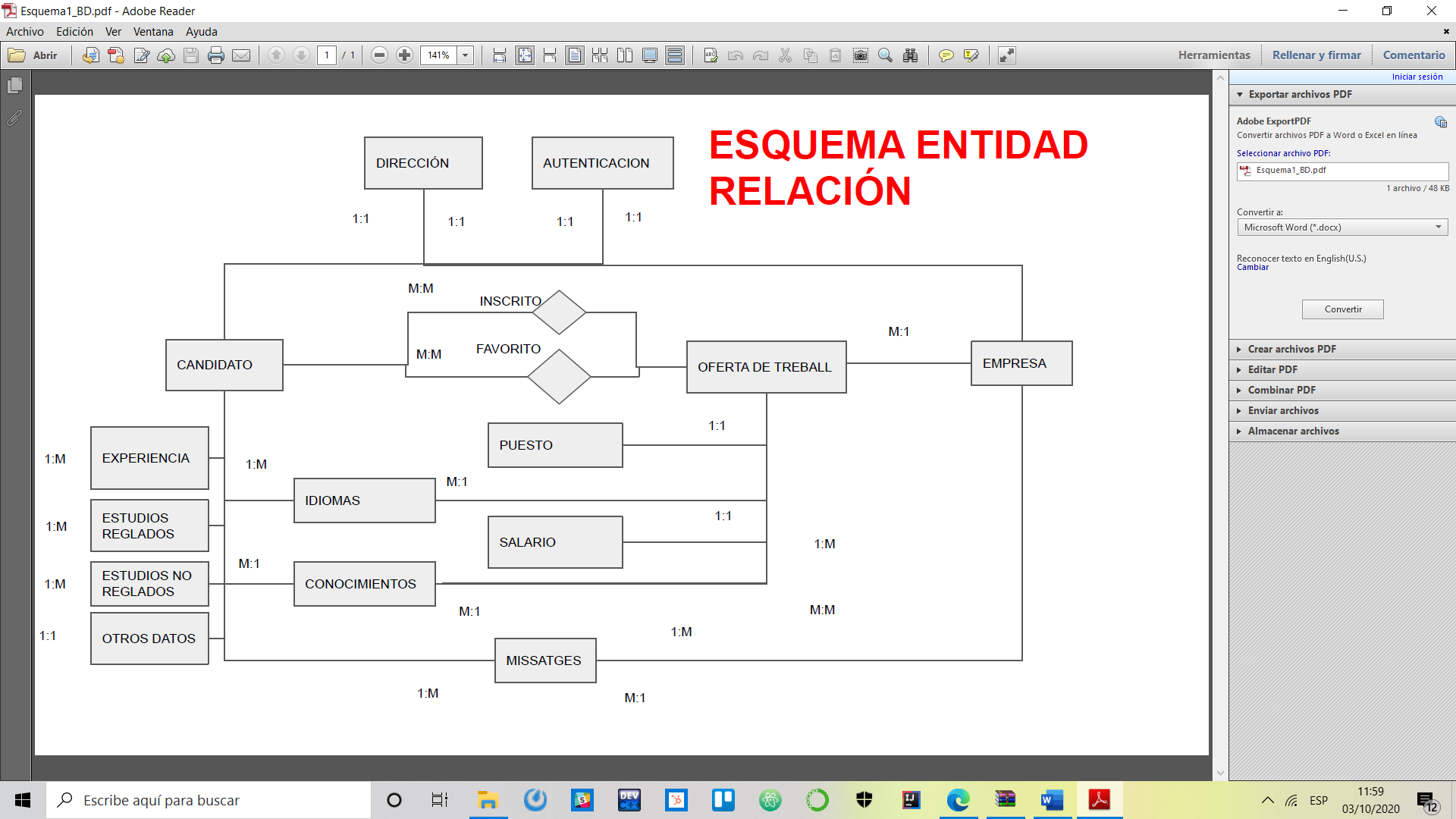


## DISENYANT MYSQL

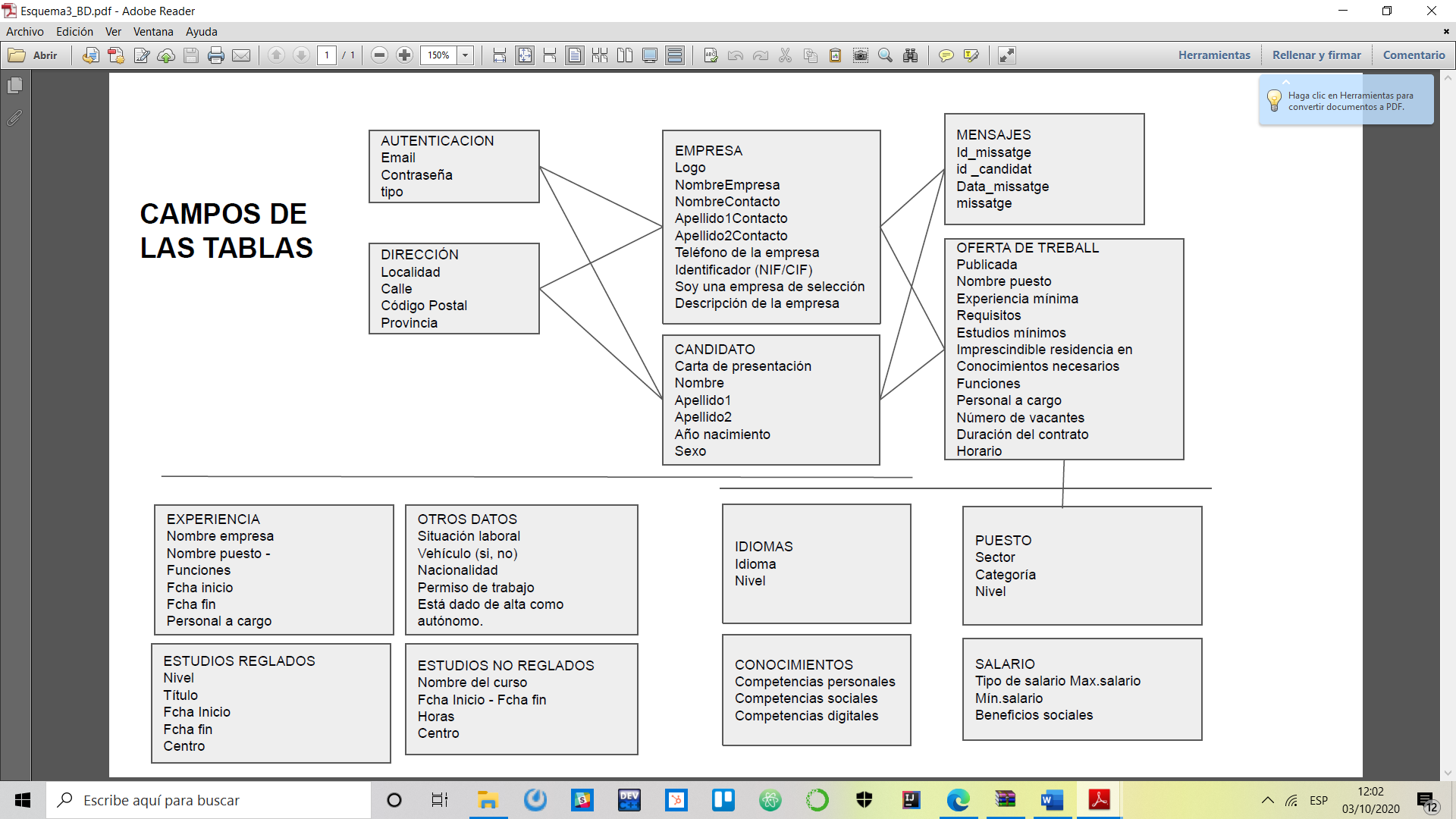
Primer configurem la base de dades, posant el mateix nom del port que en Docker.



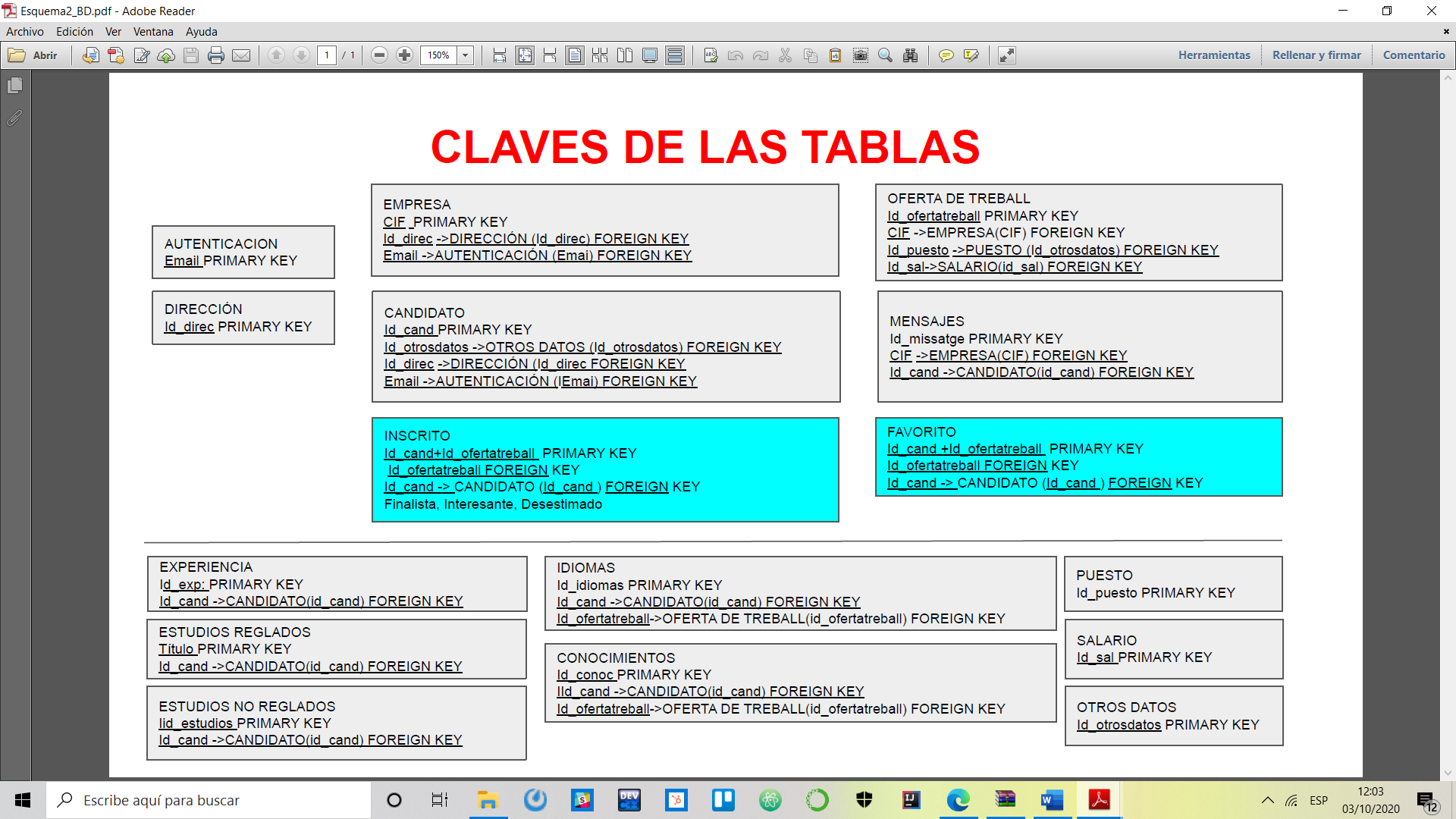
Abans de programar en WORKBENCH disenye les taules segons el model relacional.



También especifico las columnas de valores que ha de contener cada tabla



Un pas important es determinar les relacions entre les taules que ho farem mitjançant les claus alienes.



**SCRIPT**

El script executat en el workbench es el següent:

USE BDJobSenior;

drop table if exists CANDIDATO;

create table CANDIDATO (

id\_cand int auto\_increment PRIMARY KEY,

carta\_presentación text(1700),

nombre char(30) not null,

apellido1 char(50) not null,

apellido2 char(50),

año\_nacimiento date,

sexo enum('hombre', 'mujer')

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists EMPRESA;

create table EMPRESA (

CIF char (9) not null PRIMARY KEY,

logo varchar (255),

nombreEmp char(50) not null,

nombreCont char(30) not null,

apellido1Cont char(50) not null,

apellido2Cont char(50),

telfEmpre char(15),

empresaSelec set ('si', 'no'),

descripEmp text (1700)

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists AUTENTICACION;

create table AUTENTICACION(

email char (30) not null primary key,

contrasena char(40),

metodo char(5),

tipo enum ('empresa', 'candidato')

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists DIRECCION;

create table DIRECCION(

id\_direc int auto\_increment PRIMARY KEY,

localidad char (30),

calle char (30),

provincia char (30),

código\_postal char (5)

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists OFERTA\_TRABAJO;

create table OFERTA\_TRABAJO(

id\_ofertatreball int auto\_increment primary KEY,

publicada date,

nomPuesto char(50),

experiencia\_mín varchar (200),

requisitos text (1200),

estudios\_mínimos text(1200),

residencia\_en char(50),

Conoc\_necesarios varchar (200),

funciones text (1700),

personal\_a\_cargo int,

vacantes int,

duración\_contrato varchar (200),

horario varchar (200)

) ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists EXPERIENCIA;

create table EXPERIENCIA(

id\_exp int auto\_increment primary KEY,

nomEmpr\_exp char(50) not null,

nomPuesto char(50),

funciones text (1700),

CIF CHAR (9),

id\_sal int,

fchaInicio date,

fchaFin date,

personal\_a\_cargo int

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists OTROS\_DATOS;

create table OTROS\_DATOS(

id\_otrosdatos int auto\_increment primary KEY,

situacion\_laboral enum ('Estoy trabajando', 'No tengo trabajo', 'buscando una mejora'),

vehículo enum ('si', 'no'),

nacionalidad enum ('Española', 'Europea', 'Extranjera'),

permiso\_trabajo enum ('si', 'no'),

autonomo enum ('si', 'no')

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists ESTUDIOS\_REGLADOS;

create table ESTUDIOS\_REGLADOS(

id\_estreglados int auto\_increment primary KEY,

nivel enum ('Estudios Universitarios', 'FPII/CFGS', 'Bachillerato/FP1/CFGM', 'ESO', 'Certificado de escolaridad/básica', 'Sin estudios'),

título varchar(50),

fchaInicio date,

fchaFin date,

Centro char(70)

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists ESTUDIOS\_NO\_REGLADOS;

create table ESTUDIOS\_NO\_REGLADOS(

id\_estnoreglados int auto\_increment primary KEY,

curso char(50),

fchaInicio date,

fchaFin date,

Centro char(70)

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists PUESTO;

create table PUESTO(

id\_puesto int auto\_increment PRIMARY KEY,sector enum ('1.Agraria', '2.Marítimo-Pesquera', '3.Industrias Alimentarias', '4.Química', '5. Imagen Personal', '6. Sanidad', '7. Seguridad y Medio Ambiente', '8. Fabricación Mecánica', '9. Instalación y Mantenimiento','10. Electricidad y Electrónica', '11. Energía y Agua','12. Transporte y Mantenimiento de Vehículos', '13. Industrias Extractivas', '14. Edificación y Obra Civil','15. Vidrio y Cerámica','16. Madera, Mueble y Corcho', '17. Textil, Confección y Piel','18. Artes Gráficas','19. Imagen y Sonido', '20. Informática y Comunicaciones', '21. Administración y Gestión', '22. Comercio y Marketing','23. Servicios Socioculturales y a la Comunidad', '24. Hostelería y Turismo', '25. Actividades Físicas y Deportivas','26. Artes y Artesanías'),

Categoría enum('Administrativo', 'Administrativo de empresa', 'Atención al cliente', 'Calidad, producción e I+D', 'Comercial y ventas', 'Compras, logística y almacén', 'Diseño y artes gráficas', 'Educación y formación', 'Finanzas y bancos', 'Informática y telecomunicaciones', 'Ingenieros y técnicos', 'mantenimiento electromecánico y general', 'Legal', 'Marketing y Comunicación', 'Profesiones, artes y oficios', 'Recursos Humanos', 'PRL y Salud', 'Farmacia y laboratorio', 'Turismo y restauración', 'Venta al detalle'),

Nivel enum ( '1 Ingenieros y Licenciados','2 Ingenieros Técnicos, Peritos y Ayudantes Titulados', '3 Jefes Administrativos y de Taller','4 Ayudantes no Titulados','5 Oficiales Administrativos', '6 Subalternos','7 Auxiliares Administrativos','8 Oficiales de primera y segunda','9 Oficiales de tercera y Especialistas', '10 Peones','11 Trabajadores < 18 años')

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists CONOCIMIENTOS;

create table CONOCIMIENTOS(

id\_conoc int auto\_increment primary KEY,

compt\_pers set('Trabajo en equipo', 'Resolución de conflictos', 'Toma de decisiones', 'Pensamiento positivo', 'Adaptación al cambio', 'Comunicación eficaz', 'Proactividad', 'Liderazgo', 'Creatividad', 'Tolerancia a la presión', 'Orientación a resultados'),

compt\_soc set('Escucha activa', 'Asertividad', 'Empatía', 'Capacidad de negociación', 'Tolerancia', 'Honestidad', 'Receptividad/apertura de mente', 'Control emocional', 'Extroversión', 'Saber expresarse'),

compt\_dig enum('Dominio de programas informáticos profesionales y redes sociales',

'Tramites on-line de nivel avanzado', 'Destreza en programas informáticos, APPs', 'Acceso a información de interés, envio email', 'Manejo básico de ordenador')

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists IDIOMAS;

create table IDIOMAS(

Id\_idiomas int auto\_increment primary KEY,

Nivel enum ('A1', 'A2', 'B1','B2', 'C1', 'C2')

) ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists SALARIO;

create table SALARIO(

id\_sal int auto\_increment primary KEY,

tipo\_salario enum ('Bruto año', 'Bruto mes', 'Bruto día'),

max\_salario int,

min\_salario int,

beneficios\_sociales varchar(500)

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

create table FAVORITO(

id\_cand int,

id\_ofertatreball int,

PRIMARY KEY (id\_cand, id\_ofertatreball),

FOREIGN KEY (id\_cand) references CANDIDATO (id\_cand),

FOREIGN KEY (id\_ofertatreball) references OFERTA\_TRABAJO (id\_ofertatreball)

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

drop table if exists INSCRITO;

create table INSCRITO(

id\_cand int,

id\_ofertatreball int,

estado enum ('Finalista', 'Reservado', 'Descartado'),

PRIMARY KEY (id\_cand, id\_ofertatreball),

FOREIGN KEY (id\_cand) references CANDIDATO (id\_cand),

FOREIGN KEY (id\_ofertatreball) references OFERTA\_TRABAJO (id\_ofertatreball)

)ENGINE = InnoDB DEFAULT CHARSET=utf8mb4;

alter table CANDIDATO

ADD id\_otrosdatos int,

ADD id\_direc int,

ADD email char (30),

ADD FOREIGN KEY (id\_otrosdatos) references OTROS\_DATOS(id\_otrosdatos) ON DELETE CASCADE ON UPDATE CASCADE,

ADD FOREIGN KEY (id\_direc) references DIRECCION(id\_direc) ON DELETE CASCADE ON UPDATE CASCADE,

ADD FOREIGN KEY (email) references AUTENTICACION(email) ON DELETE CASCADE ON UPDATE CASCADE;

alter TABLE EMPRESA

ADD id\_direc int,

ADD email char (30),

ADD FOREIGN KEY (id\_direc) references DIRECCION(id\_direc) ON DELETE CASCADE ON UPDATE CASCADE,

ADD FOREIGN KEY (email) references AUTENTICACION(email) ON DELETE CASCADE ON UPDATE CASCADE;

alter table OFERTA\_TRABAJO

ADD CIF char (9),

ADD id\_puesto int,

ADD id\_sal int,

ADD FOREIGN KEY (CIF) references EMPRESA (CIF) ON DELETE no action ON update no action,

ADD FOREIGN KEY (id\_puesto) references PUESTO (id\_puesto) ON DELETE CASCADE ON UPDATE CASCADE,

ADD FOREIGN KEY (id\_sal) references SALARIO (id\_sal) ON DELETE CASCADE ON UPDATE CASCADE;

alter table EXPERIENCIA

ADD id\_cand int,

ADD FOREIGN KEY (id\_cand) references CANDIDATO(id\_cand) ON DELETE CASCADE ON UPDATE CASCADE;

alter table ESTUDIOS\_REGLADOS

ADD id\_cand int,

ADD FOREIGN KEY (id\_cand) references CANDIDATO(id\_cand) ON DELETE CASCADE ON UPDATE CASCADE;

alter table CONOCIMIENTOS

ADD id\_cand int,

ADD id\_ofertatreball int,

ADD FOREIGN KEY (id\_cand) references CANDIDATO(id\_cand) ON DELETE CASCADE ON UPDATE CASCADE,

ADD FOREIGN KEY (id\_ofertatreball) references OFERTA\_TRABAJO(id\_ofertatreball) ON DELETE CASCADE ON UPDATE CASCADE;

alter table IDIOMAS

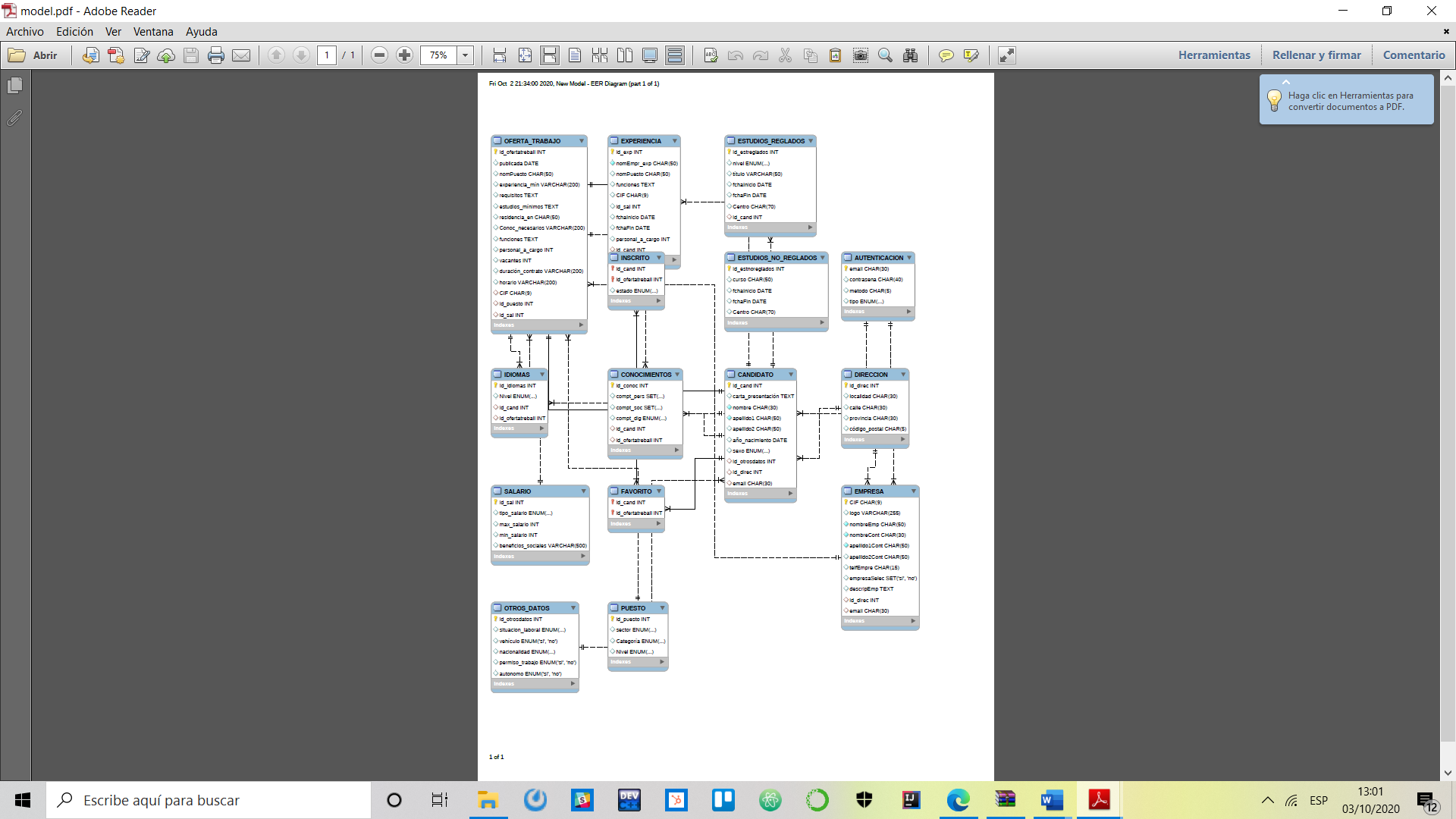
ADD id\_cand int,

ADD id\_ofertatreball int,

ADD FOREIGN KEY (id\_cand) references CANDIDATO(id\_cand) ON DELETE CASCADE ON UPDATE CASCADE,

ADD FOREIGN KEY (id\_ofertatreball) references OFERTA\_TRABAJO(id\_ofertatreball) ON DELETE CASCADE ON UPDATE CASCADE;

El disseny relacional que treu WORKBENCH es el següent:



## INICIALITZANT SPRING

## CONFIGURANT HIBERNATE

## DISENYANT ANDROID

## ACCEDINT A LES DADES

# RECURSOS

Documentació, recerca i anàlisi.

# AUTOAVALUACIÓ I CONCLUSIONS

# INDICE DE IMÁGENES Y CONTENIDOS USB

Penúltimas páginas: índice de imágenes, explicación de contenidos del CD/DVD. Carpetas contenidas, lista de ficheros que tiene cada una, y relación de ellos con los contenidos. Explicación de uso si procede. Desglose del software utilizado.

# BIBLIOGRAFIA

Bibliografia i webgrafia normalitzada.

# ANNEXOS

Annexos amb el testimoni gràfic de les diverses etapes de l'execució (si es considera necessari).

Última página: bibliografía (nombre, autor, editorial, año), enlaces consultados (direcciones web).