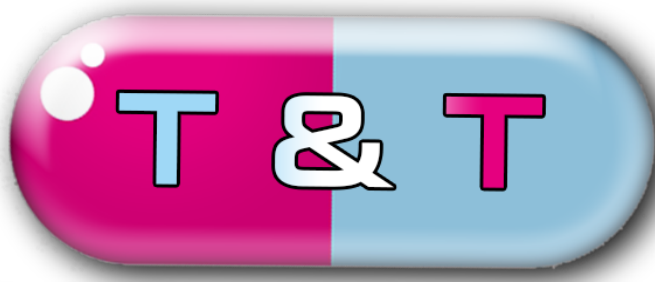


Final project

Treatment Tracer

MANUAL DE DESPLIEGUE



Esther Ponce García

D.A.W 2018/2021



ÍNDICE

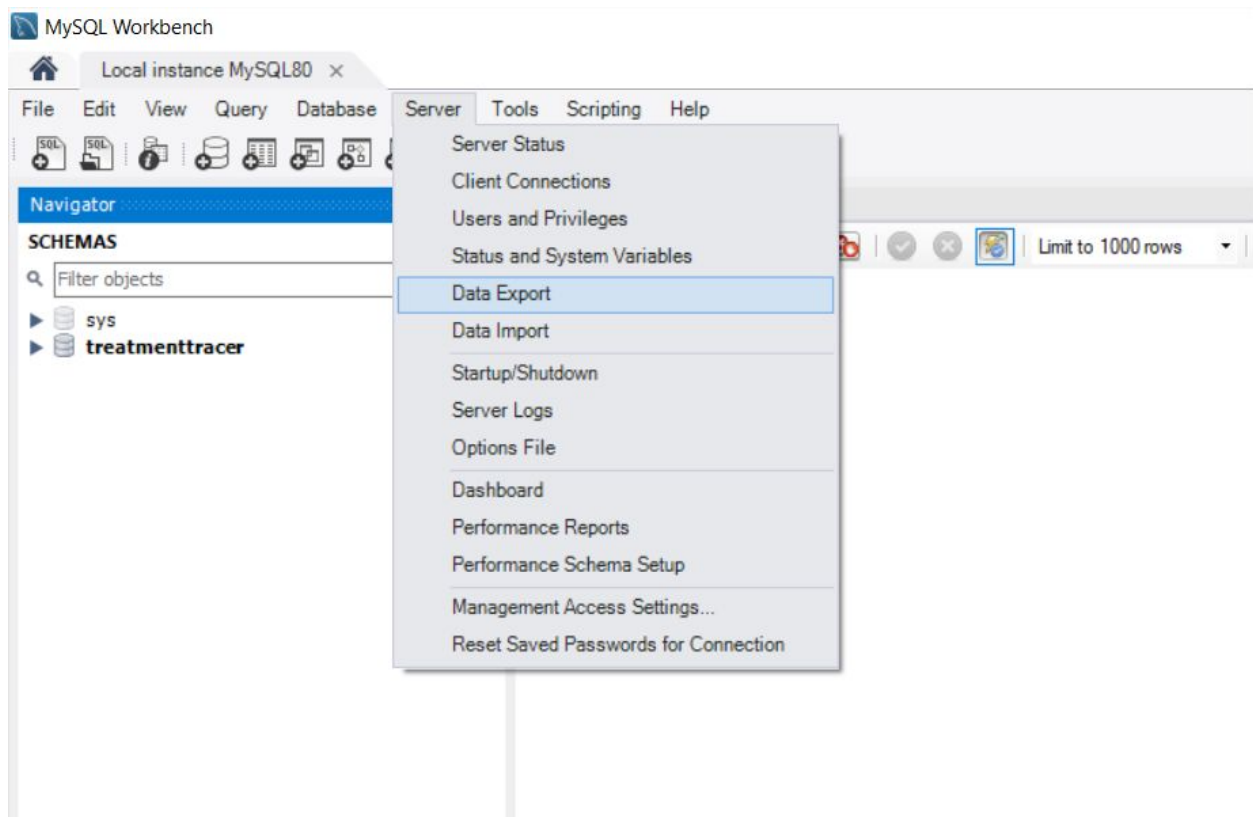
- Requerimientos previos al despliegue.
- Despliegue del sistema
 - Conexión con repositorios
 - Contenedores y servicios
 - Logs del sistema.
 - Primera puesta en servicio



REQUERIMIENTOS PREVIOS AL DESPLIEGUE

Para poder realizar el despliegue de manera satisfactoria, necesitaremos previamente:

- Disponer de una máquina virtual que tenga instalado Tomcat y MySQL.
- Disponer del archivo SQL para la generación de tablas QUARTZ, que serán usadas para la automatización de tareas, estas tablas no se generan de forma automática con hibernate por lo que necesitaremos disponer de ese script. Lo podemos obtener de la carpeta WAR del proyecto en gitHub.
- Disponer de la exportación de nuestra base de datos local del siguiente modo:
 - Abrimos MySQL y seleccionamos en el menú superior la server->data export



- Seleccionamos la base de datos que queremos exportar



Local instance MySQL80
Data Export Advanced Options...

Object Selection | Export Progress

Tables to Export

Exp...	Schema
<input type="checkbox"/>	sys
<input checked="" type="checkbox"/>	treatmenttracer

Refresh Dump Structure and Dat Select Views Select Tables Unselect All

Objects to Export

☐ Dump Stored Procedures and Functions ☐ Dump Events ☐ Dump Triggers

Export Options

☐ Export to Dump Project Folder C:\Users\Esther\Documents\dumps\Dump20201209 ...

Each table will be exported into a separate file. This allows a selective restore, but may be slower.

☒ Export to Self-Contained File C:\Users\Esther\Documents\dumps\Dump20201209.sql ...

All selected database objects will be exported into a single, self-contained file.

☐ Create Dump in a Single Transaction (self-contained file only) ☐ Include Create Schema

Press [Start Export] to start... Start Export

- Seleccionamos la ubicación del archivo que se va a generar.

Export Options

☐ Export to Dump Project Folder C:\Users\Esther\Documents\dumps\Dump20201209 ...

Each table will be exported into a separate file. This allows a selective restore, but may be slower.

☒ Export to Self-Contained File C:\Users\Esther\Documents\dumps\Dump20201209.sql ...

All selected database objects will be exported into a single, self-contained file.

☐ Create Dump in a Single Transaction (self-contained file only) ☐ Include Create Schema

Press [Start Export] to start... Start Export

- Marcamos el checkbox incluir schema y le damos al botón “Start export”

Export Options

☐ Export to Dump Project Folder C:\Users\Esther\Documents\dumps\Dump20201209 ...

Each table will be exported into a separate file. This allows a selective restore, but may be slower.

☒ Export to Self-Contained File C:\Users\Esther\Documents\dumps\Dump20201209.sql ...

All selected database objects will be exported into a single, self-contained file.

☐ Create Dump in a Single Transaction (self-contained file only) ☒ Include Create Schema

Press [Start Export] to start... Start Export



- Realizar algunos cambios en nuestro proyecto spring para que se pueda desplegar con Tomcat. En concreto necesitaremos añadir lo siguiente en la clase que lanza la aplicación, en nuestro caso esa clase es TreatmentTracerApplication

```
1 package org.iesalixar.eponceg;
2
3 import org.springframework.boot.SpringApplication;
4
5 @SpringBootApplication
6 public class TreatmentTracerApplication extends SpringBootServletInitializer {
7
8     public static void main(String[] args) {
9         SpringApplication.run(TreatmentTracerApplication.class, args);
10    }
11
12    @Override
13    protected SpringApplicationBuilder configure(SpringApplicationBuilder application) {
14        return application.sources(TreatmentTracerApplication.class);
15    }
16 }
```

Hemos añadido la herencia de la clase, y las líneas de código 15-18

- Revisar en nuestro proyecto todas las rutas están bien colocadas, y son accesibles.
- Debido a la configuración de MySQL en el servidor no podemos usar la contraseña root, para la base de datos por lo que tenemos que usar Temporal123. Este dato lo cambiamos en el application.properties del proyecto

```
TreatmentTracerApplication.java application.properties
1
2 # Configuración hibernate
3
4 spring.datasource.url=jdbc:mysql://localhost:3306/treatmenttracer?createDatabaseIfNotExist=true&serverTimezone=UTC&useLegacyDate
5 spring.datasource.username=root
6 spring.datasource.password=Temporal123
7 #
8
```



DESPLIEGUE DEL SISTEMA

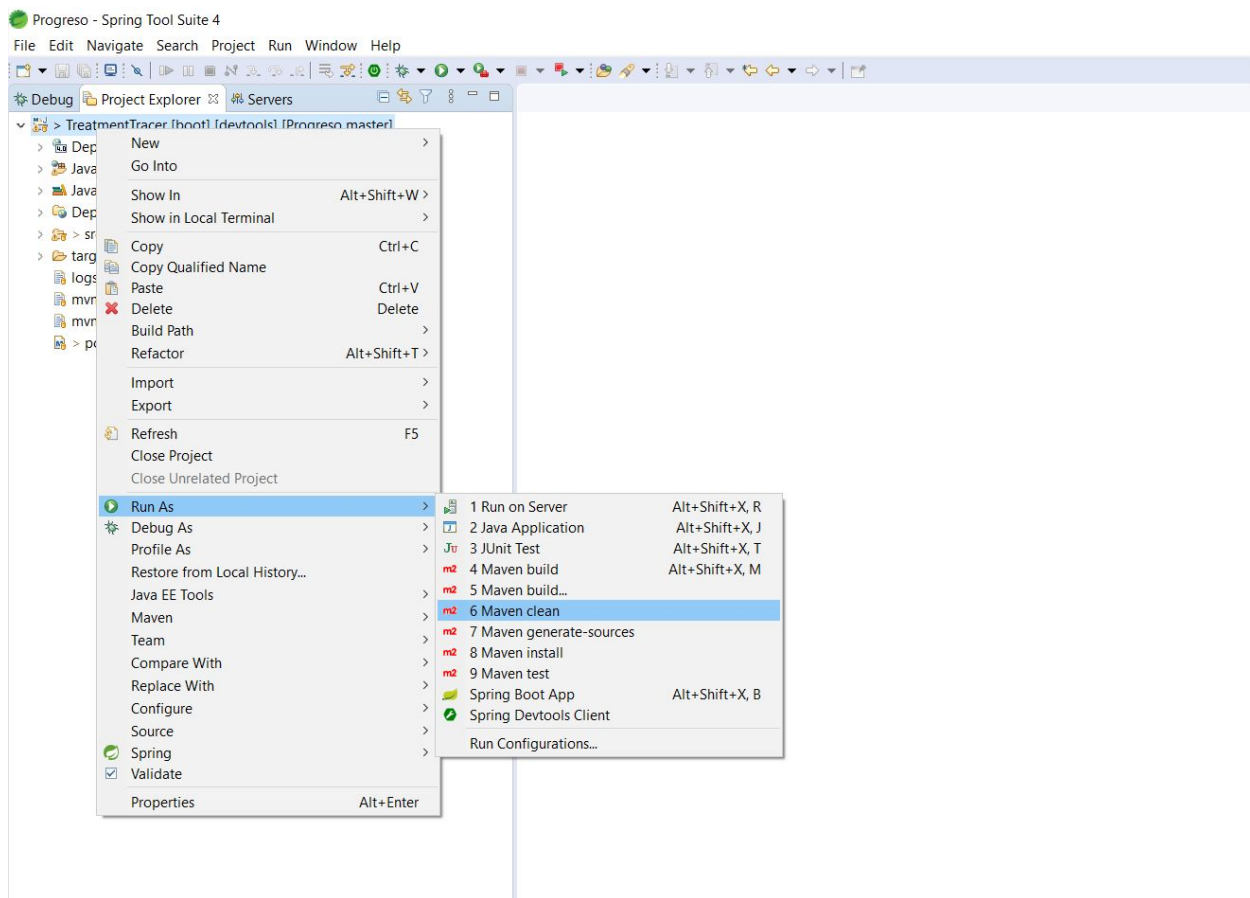
- Conexión con repositorios.

Una vez listos para comenzar con el despliegue se debe obtener el war del aplicativo, que será el que se emplee para desplegar el proyecto en el servidor.

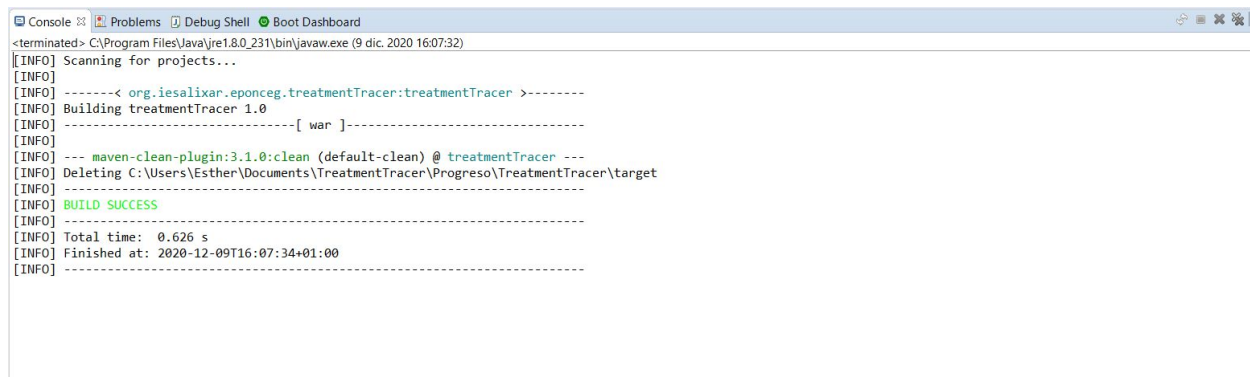
- En nuestro proyecto en local, se debe modificar el Pom.xml para darle un nombre y una versión al despliegue, en este caso hemos escogido darle el nombre TreatmentTracer y la versión 1.0.

```
TreatmentTracerApplication.java  application.properties  *TreatmentTracer/pom.xml  ⌕
1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
5   <modelVersion>4.0.0</modelVersion>
6   <parent>
7     <groupId>org.springframework.boot</groupId>
8     <artifactId>spring-boot-starter-parent</artifactId>
9     <version>2.2.4.RELEASE</version>
10    <relativePath /> <!-- lookup parent from repository -->
11  </parent>
12  <groupId>org.iesalixar.eponceg.treatmentTracer</groupId>
13  <artifactId>TreatmentTracer</artifactId>
14  <packaging>war</packaging>
15  <version>1.0</version>
16  <name>TreatmentTracer</name>
17  <description>Final Project</description>
18
```

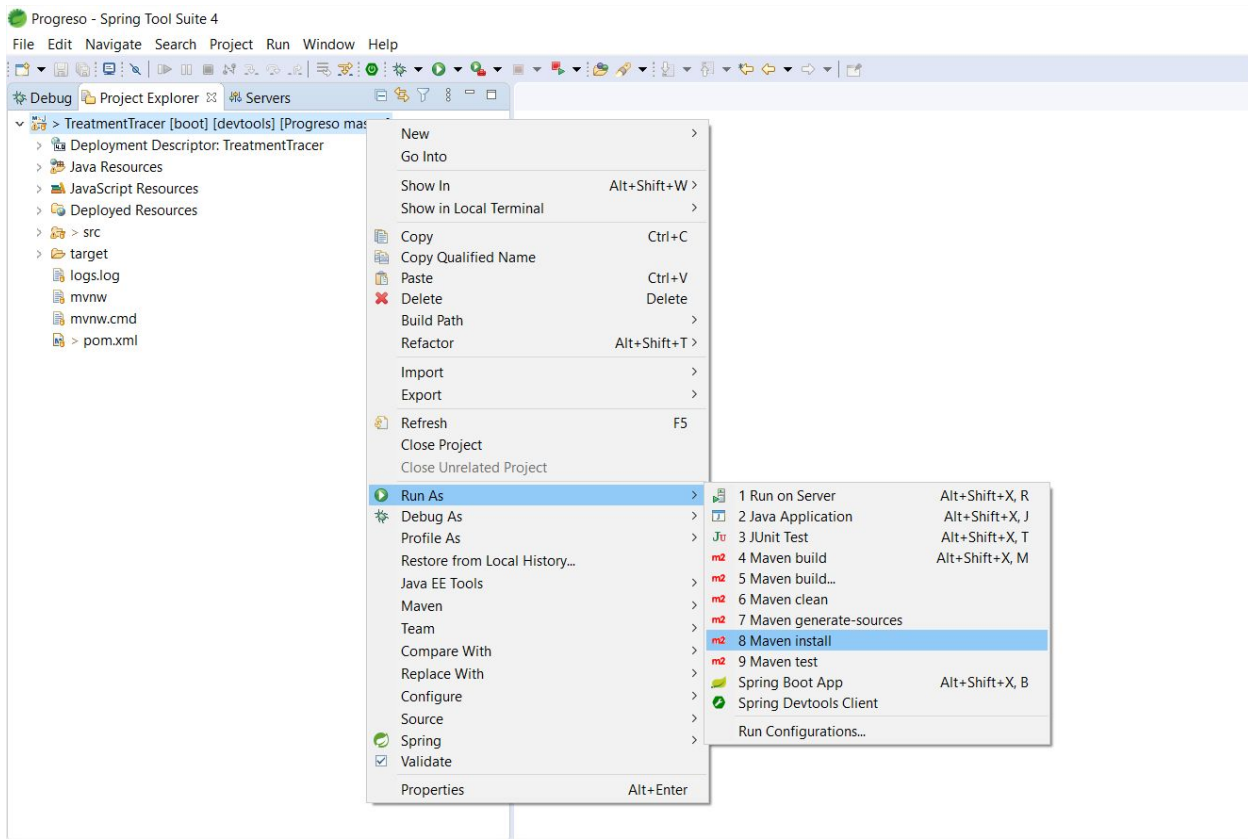
- Hacemos click derecho sobre el proyecto y le damos a run as “Maven clean” para limpiar el proyecto por si hubiera algún recurso generado con anterioridad.



- Cuando el limpiado sea correcto se verá el siguiente mensaje.



- Hacemos click derecho de nuevo sobre el proyecto y esta vez le damos a run as “Maven install” esto generará el archivo war, dentro de la carpeta target.





- Cuando está correcto se verá el siguiente mensaje y en la carpeta target estará el archivo war








entos > TreatmentTracer > Progreso > TreatmentTracer > target >

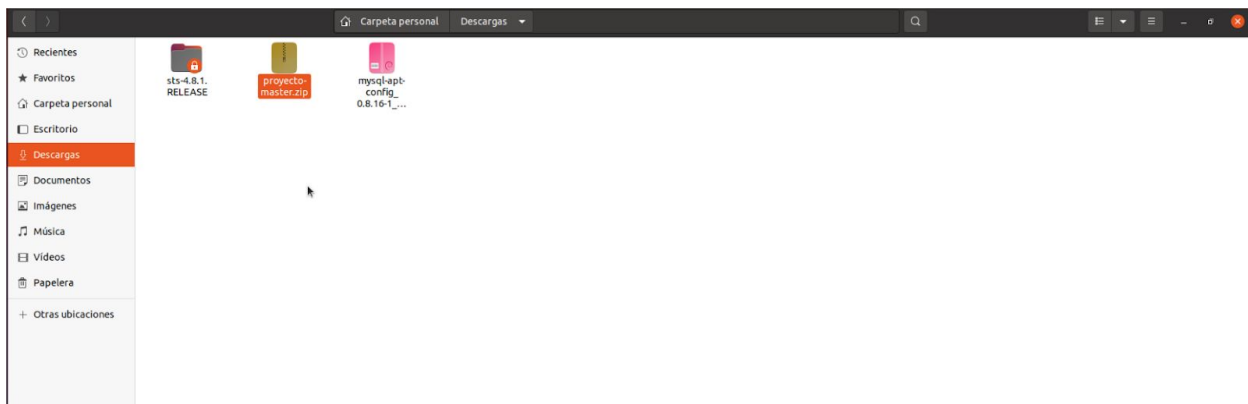
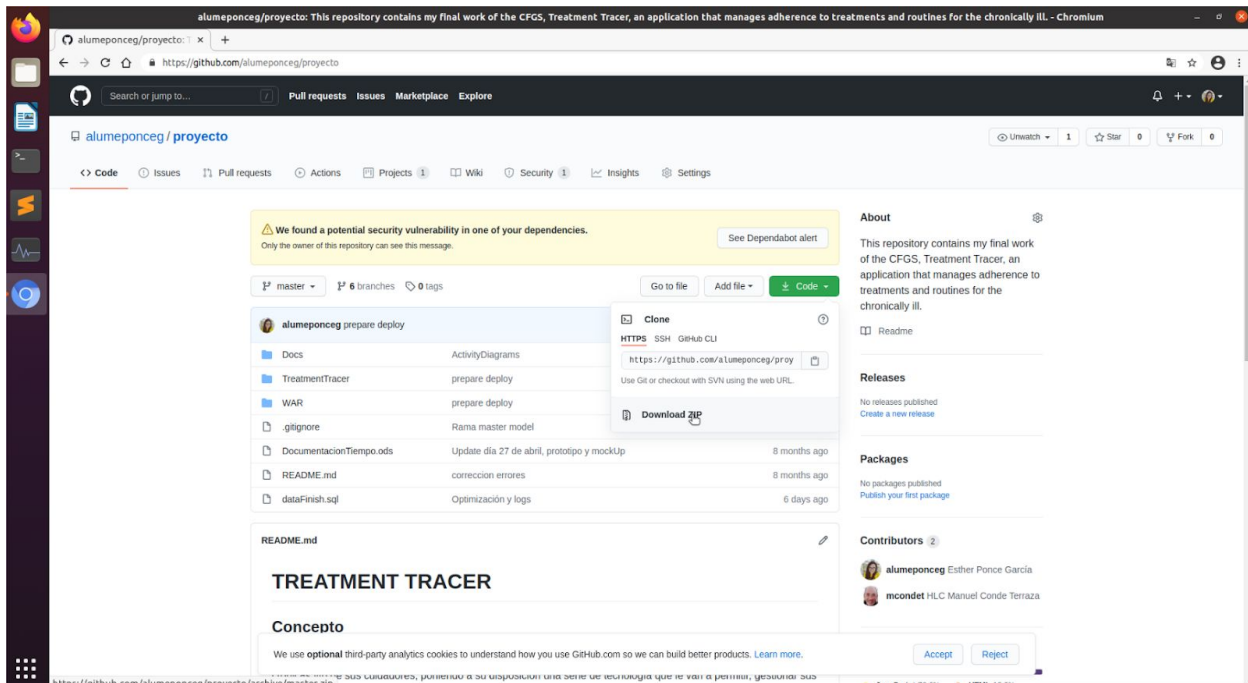
Nombre	Fecha de modificación	Tipo	Tamaño
classes	09/12/2020 16:10	Carpeta de archivos	
generated-sources	09/12/2020 16:10	Carpeta de archivos	
generated-test-sources	09/12/2020 16:10	Carpeta de archivos	
m2e-wtp	09/12/2020 16:10	Carpeta de archivos	
maven-archiver	09/12/2020 16:10	Carpeta de archivos	
maven-status	09/12/2020 16:10	Carpeta de archivos	
test-classes	09/12/2020 16:10	Carpeta de archivos	
treatmentTracer-1.0	09/12/2020 16:10	Carpeta de archivos	
 treatmentTracer-1.0.war	09/12/2020 16:10	Archivo WAR	48.158 KB
 treatmentTracer-1.0.war.original	09/12/2020 16:10	Archivo ORIGINAL	43.513 KB

- Para facilitar el acceso desde el servidor al archivo war, lo copiaremos y lo pegaremos dentro de la carpeta “Despliegue” del repositorio en GitHub, donde ya tendremos el sql de quartz y el sql exportado de la base de datos.

entos > TreatmentTracer > Progreso > WAR

Nombre	Fecha de modificación	Tipo	Tamaño
 dump.sql	09/12/2020 0:17	SQL Text File	36 KB
 treatmentTracer-1.0.war	09/12/2020 16:10	Archivo WAR	48.158 KB
 QUARTZ.sql	03/12/2020 17:00	SQL Text File	5 KB

- Nos conectamos al servidor donde vamos a hacer el despliegue, y descargamos el proyecto de gitHub, desde ahí solo tendremos que acceder a la carpeta WAR dónde se encuentran los archivos que vamos a necesitar para el despliegue



- Contenedores y servicios.

Cuando ya se tiene todo preparado y se dispone de los archivos necesarios en el servidor, se procede a cargar los datos en nuestros contenedores.

En primer lugar vamos con la persistencia de datos.

- Entramos en MySQL y ejecutamos el script que contiene la exportación de la base de datos.



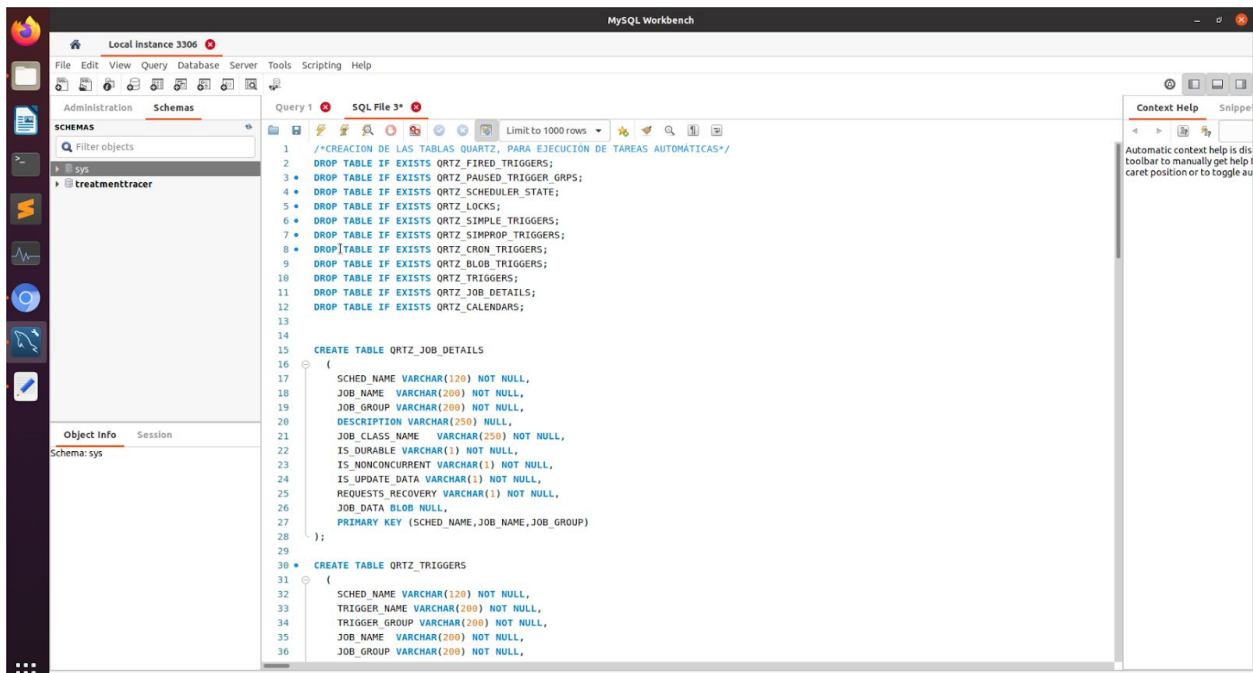
The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left shows the 'sys' schema. The main editor displays a SQL script for creating a database and a table. The script includes comments about the MySQL version and the character set. The table 'disease' is created with columns: 'id' (bigint(20) NOT NULL AUTO_INCREMENT), 'causes' (longtext), 'description' (longtext), 'image' (longtext), 'name' (varchar(45) DEFAULT NULL), 'symptom' (longtext), 'state' (bigint(20) DEFAULT NULL), and 'PRIMARY KEY' ('id').

```
1 • CREATE DATABASE IF NOT EXISTS `treatmenttracer` /*!40100 DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci */ /*!180016 DEFAULT EN
2 • USE `treatmenttracer`;
3 -- MySQL dump 10.13 Distrib 8.0.18, for Win64 (x86_64)
4 --
5 -- Host: localhost    Database: treatmenttracer
6 --
7 -- Server version: 8.0.18
8 --
9 /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
10 /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
11 /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
12 /*!50503 SET NAMES utf8 */;
13 /*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
14 /*!40103 SET TIME_ZONE='+00:00' */;
15 /*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
16 /*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
17 /*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
18 /*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
19
20 --
21 -- Table structure for table `disease`
22 --
23
24 • DROP TABLE IF EXISTS `disease`;
25 /*!40101 SET @saved_cs_client = @@character_set_client */;
26 /*!50503 SET character_set_client = utf8mb4 */;
27 • CREATE TABLE `disease` (
28   `id` bigint(20) NOT NULL AUTO_INCREMENT,
29   `causes` longtext,
30   `description` longtext,
31   `image` longtext,
32   `name` varchar(45) DEFAULT NULL,
33   `symptom` longtext,
34   `state` bigint(20) DEFAULT NULL,
35   PRIMARY KEY (`id`),
36   UNIQUE KEY `UK_am27lok2q3vo8p5cfc482b32u` (`name`),
37 
```

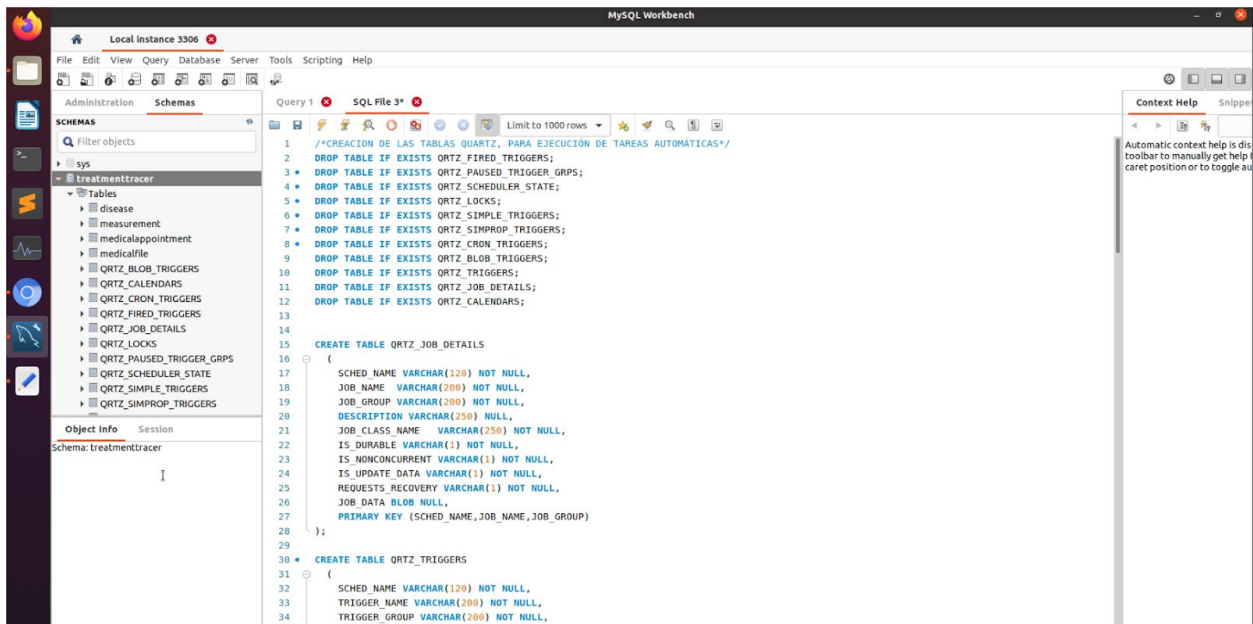
The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left shows the 'treatmenttracer' schema. The main editor displays a SQL script for creating a table and inserting data. The table 'medicalfile' is created with columns: 'id' (bigint(20) NOT NULL AUTO_INCREMENT), 'description' (varchar(255) DEFAULT NULL), 'file' (longtext NOT NULL), 'title' (varchar(255) NOT NULL), 'uploaddate' (datetime NOT NULL DEFAULT CURRENT_TIMESTAMP), and 'userid' (bigint(20) NOT NULL). The script also includes a comment about the MySQL version and the character set. The table 'role' is created with columns: 'id' (bigint(20) NOT NULL AUTO_INCREMENT) and 'description' (varchar(255) DEFAULT NULL).

```
110 --
111
112 • DROP TABLE IF EXISTS `medicalfile`;
113 /*!40101 SET @saved_cs_client = @@character_set_client */;
114 /*!50503 SET character_set_client = utf8mb4 */;
115 • CREATE TABLE `medicalfile` (
116   `id` bigint(20) NOT NULL AUTO_INCREMENT,
117   `description` varchar(255) DEFAULT NULL,
118   `file` longtext NOT NULL,
119   `title` varchar(255) NOT NULL,
120   `uploaddate` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
121   `userid` bigint(20) NOT NULL,
122   PRIMARY KEY (`id`),
123   KEY `FKa8b2lr8ijuyhckpcrch32ixd` (`userid`)
124 ) ENGINE=MyISAM AUTO_INCREMENT=15 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
125 /*!40101 SET character_set_client = @saved_cs_client */;
126
127 --
128 -- Dumping data for table `medicalfile`
129 --
130
131 • LOCK TABLES `medicalfile` WRITE;
132 /*!40000 ALTER TABLE `medicalfile` DISABLE KEYS */;
133 • INSERT INTO `medicalfile` VALUES (13,'Dieta del mes de noviembre','https://firebasestorage.googleapis.com/v0/b/treatmenttracer.appspot.c
134 /*!40000 ALTER TABLE `medicalfile` ENABLE KEYS */;
135 • UNLOCK TABLES;
136 --
137 -- Table structure for table `role`
138 --
139
140 • DROP TABLE IF EXISTS `role`;
141 /*!40101 SET @saved_cs_client = @@character_set_client */;
142 /*!50503 SET character_set_client = utf8mb4 */;
143 • CREATE TABLE `role` (
144   `id` bigint(20) NOT NULL AUTO_INCREMENT,
145   `description` varchar(255) DEFAULT NULL,
146 
```

- Cuando se complete, hacemos lo mismo con el script de las tablas QUARTZ.



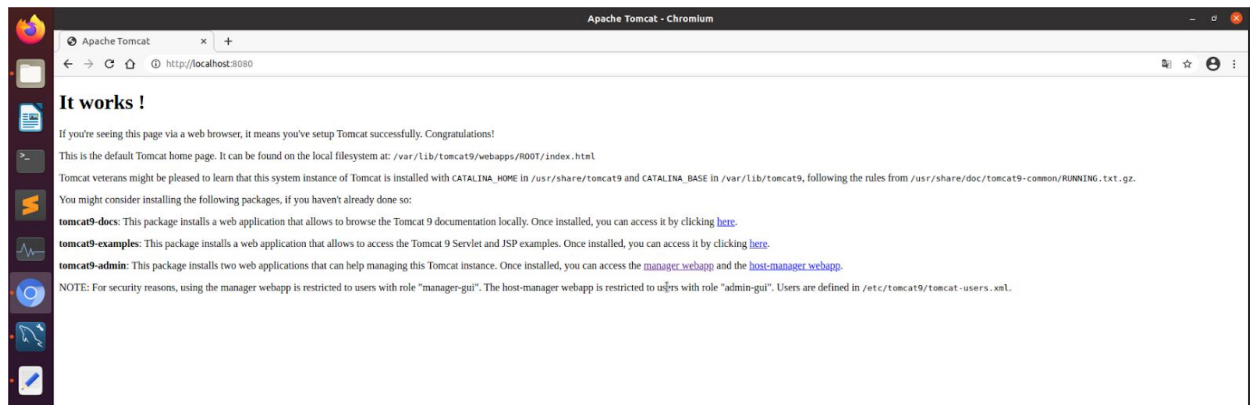
- Comprobamos que todo está correcto, que todas las tablas se han generado correctamente y que contienen los datos pertinentes.



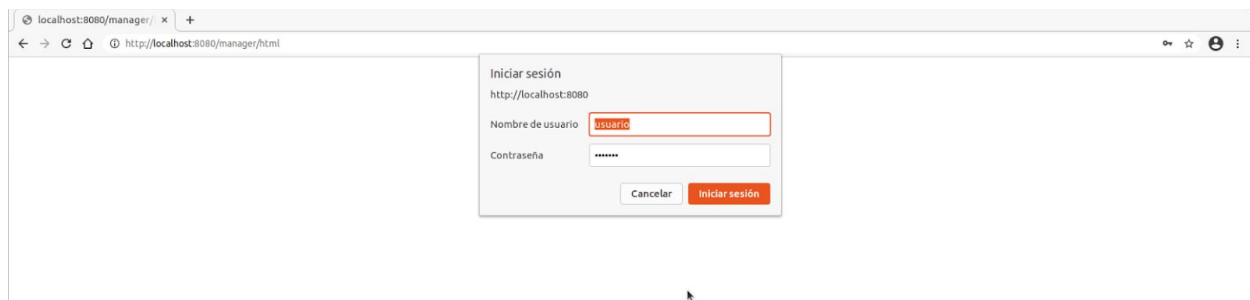


A continuación vamos a desplegar el archivo war en tomcat, para poder ver la aplicación en funcionamiento

- Entrar en localhost:8080 para acceder a la página principal de tomcat.





- Hacer click en manager app para entrar en la web de gestión. Introducir nuestro usuario y contraseña.



- Buscar el área dónde se sube el archivo war para desplegarlo, seleccionar nuestro archivo war, que se encuentra dentro del proyecto que hemos descargado de gitHub, en la carpeta WAR.



/manager x +
← → ↻ ⚙ http://localhost:8080/manager/html ☆ ⓘ

Gestor de Aplicaciones Web de Tomcat

Mensaje: OK

Gestor

[Listar Aplicaciones](#) [Ayuda HTML de Gestor](#) [Ayuda de Gestor](#) [Estado de Servidor](#)

Aplicaciones

Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar > 30 minutos
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar > 30 minutos
/manager	Ninguno especificado	Tomcat Manager Application	true	2	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar > 30 minutos

Desplegar

Desplegar directorio o archivo WAR localizado en servidor

Trayectoria de Contexto (opcional):
Version (for parallel deployment):
URL de archivo de Configuración XML:
URL de WAR o Directorio:
[Desplegar](#)

Archivo WAR a desplegar

Seleccione archivo WAR a cargar [Seleccionar archivo](#) Ningún archivo seleccionado
[Desplegar](#) Ningún archivo seleccionado

Configuration

Re-read TLS configuration files

TLS host name (optional)

- Hacer click en el botón desplegar y esperar a que tomcat conecte con la base de datos y cree la estructura de nuestro proyecto.

Archivo WAR a desplegar

Seleccione archivo WAR a cargar [Seleccionar archivo](#) treatmentTracer-1.0.war
[Desplegar](#)

- Comprobar que todo es correcto, si todo es correcto el despliegue debe aparecer en la lista superior de la manager-app y debe de mostrar el texto true en la columna de “En ejecución” eso significa que si hacemos click en el enlace a nuestro proyecto ya deberíamos vernos del mismo modo que en el local.



Apache Tomcat Manager

Gestor de Aplicaciones Web de Tomcat

Mensaje: OK

Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/manager	Ninguno especificado	Tomcat Manager Application	true	2	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/treatmentTracer-1.0	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos

- Logs del sistema.

Para una mejor trazabilidad de los errores en nuestro aplicativo o simplemente para obtener una información orientativa del uso del aplicativo, es necesario disponer de un sistema de logs.

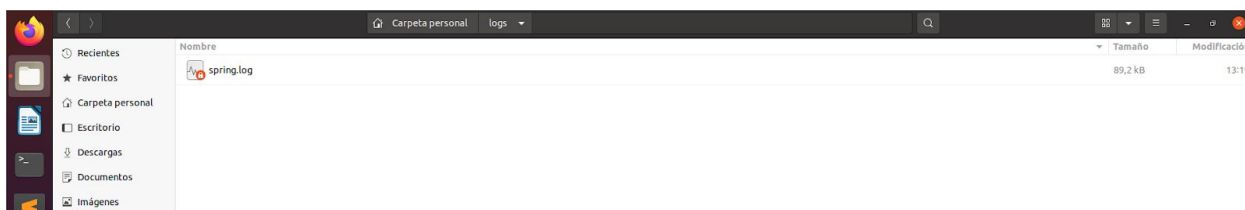
Para Treatment Tracer se ha optado por usar la librería Logback, cuya implementación con Spring es muy sencilla.

Debemos añadir en el application properties del proyecto la ruta donde queremos que se almacenen los logs en la propiedad *logging.file.path* en este caso hemos optado por la ruta */home/usuario/logs*

```
1 2 # Configuración hibernate
3
4 spring.datasource.url=jdbc:mysql://localhost:3306/treatmenttracer?createDatabaseIfNotExist=true&serverTimezone=UTC&useLegacyDatetimeCode=false
5 spring.datasource.username=root
6 spring.datasource.password=Temporal123
7 #
8 spring.datasource.testWhileIdle= true
9 spring.datasource.validationQuery= SELECT 1
10 spring.jpa.show-sql= true
11 spring.jpa.hibernate.ddl-auto= update
12 spring.devtools.livereload.enabled= true
13 spring.jpa.hibernate.naming-strategy= org.hibernate.cfg.ImprovedNamingStrategy
14 spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.MySQL5Dialect
15 spring.jpa.properties.hibernate.format_sql=true
16
17 spring.jpa.properties.hibernate.generate_statistics=true
18
19
20 spring.quartz.job-store-type=jdbc
21 spring.quartz.properties.org.quartz.jobStore.isClustered=true
22 spring.quartz.properties.org.quartz.scheduler.instanceId=AUTO
23
24 spring.messages.basename = messages, config.i18n.messages
25 spring.messages.fallback-to-system-locale = false
26
27 logging.file.path = /home/usuario/logs
28 #debug = false
29 #logging.level.root = WARN
```




Con ello cuando arranquemos nuestra aplicación y a medida que vayamos haciendo uso de ella, veremos que en la ruta que hemos especificado se muestra un archivo spring.log



Si accedemos al archivo se puede comprobar la traza que va dejando el uso de nuestra aplicación al igual que aquellos logs que hemos preparado específicamente.

En la línea 1292 se observa el mensaje “Se ha redireccionado al usuario a la página de inicio de sesión”.





- Primera puesta en servicio.

Tras comprobar que el despliegue ha sido correcto, accedemos al enlace que se muestra en la lista, y si todo el mapeo de rutas que se comentaba al principio como requisito previo, toda la conexión a la base de datos, y los cambios realizados en la clase Application han sido correctos, se mostrará la welcome page con el mismo formato y diseño que en el desarrollo en local. Basta con introducir los datos de inicio de sesión para comprobar la navegación entre páginas y que todos los recursos están disponibles.

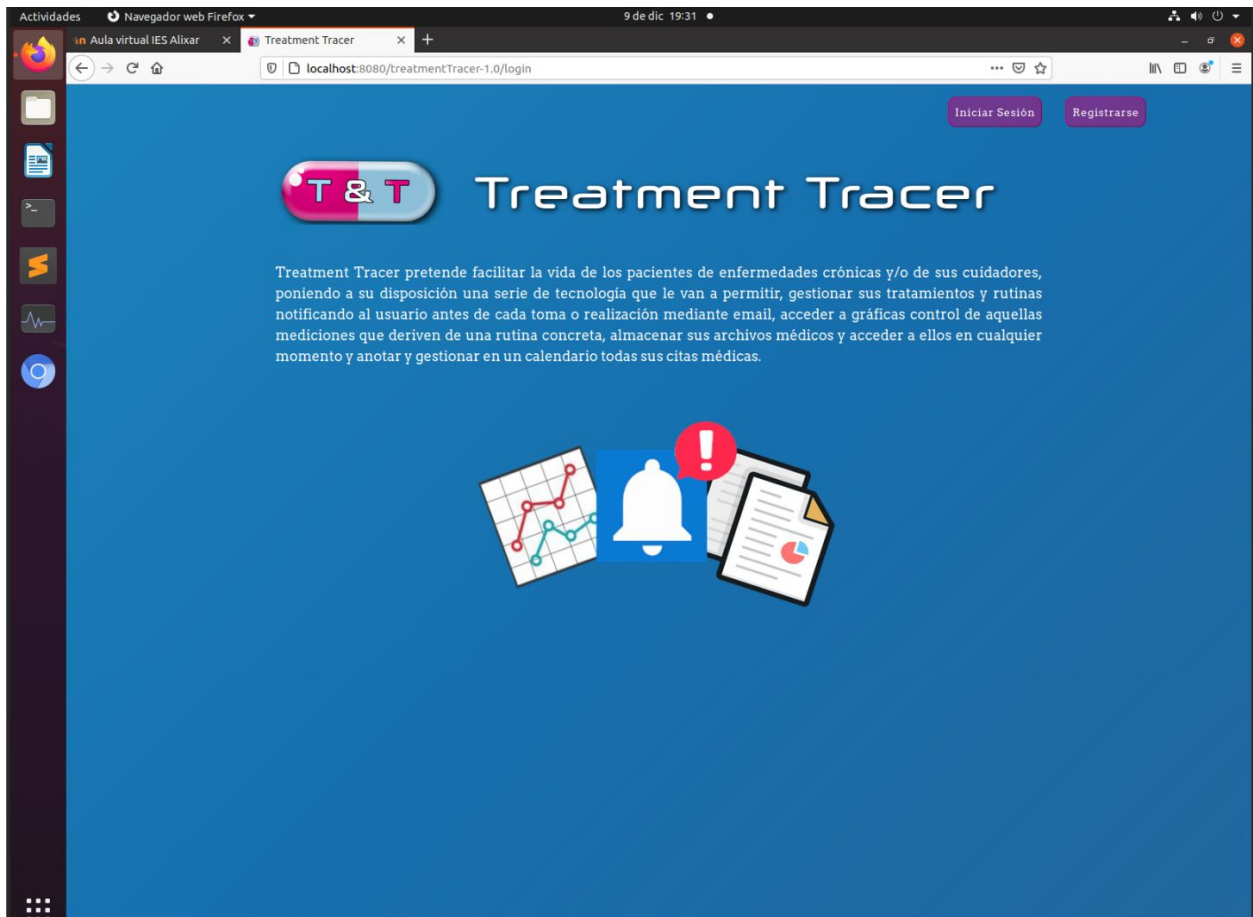
A continuación se muestra la lista de usuarios y contraseñas empleados para la prueba del servicio:

ROL	EMAIL	CONTRASEÑA
Paciente	Esther_virgo97@hotmail.com	Temporal123
Cuidador	alum.bluisrochag@iesalixar.org	Severa2011
Administrador	admin@admin.com	1234

En las siguientes imágenes se pueden ver algunas de las pestañas en funcionamiento.



Página de bienvenida:





Página principal del paciente

The screenshot shows a web browser window with the address bar displaying `localhost:8080/treatmentTracer-1.0/user/home`. The application is titled "Treatment Tracer" and has a navigation bar with links: "Tratamientos", "Rutinas", "Citas Médicas", "Archivos Médicos", and a "Cerrar Sesión" button. The main content area is titled "Inicio" and features a sidebar on the left with "Acciones Rápidas" (Quick Actions) including "Añadir Tratamiento" and "Añadir Rutina". The main content area displays information for "Diabetes". At the top, there is a form to "Seleccionar enfermedad" (Select disease) with a dropdown menu showing "EPOC" and a "Vincular enfermedad" (Link disease) button. Below this, the "Diabetes" section includes a "Descripción" (Description) and a "Causas" (Causes) section. The "Descripción" section contains a paragraph about diabetes and a small image of a person using a glucose meter. The "Causas" section lists several causes. The "Síntomas" (Symptoms) section lists several symptoms. At the bottom, there is a section titled "Eliminar enfermedad" (Remove disease) with a "Borrar" (Delete) button.

Actividades Navegador web Firefox 9 de dic 19:32

Aula virtual IES Alíxar X Bienvenido

localhost:8080/treatmentTracer-1.0/user/home

Treatment Tracer

Tratamientos Rutinas Citas Médicas Archivos Médicos Cerrar Sesión

Inicio

Acciones Rápidas

Añadir Tratamiento

Añadir Rutina

Seleccionar enfermedad Ver datos EPOC Vincular enfermedad

Diabetes

Descripción

La diabetes es una enfermedad crónica que se origina porque el páncreas no sintetiza la cantidad de insulina que el cuerpo humano necesita, la elabora de una calidad inferior o no es capaz de utilizarla con eficacia. La insulina es una hormona producida por el páncreas. Su principal función es el mantenimiento de los valores adecuados de glucosa en sangre. En las personas con diabetes hay un exceso de glucosa en sangre (hiperglucemia), ya que no se distribuye de la forma adecuada.

Causas

- No se conoce la causa exacta de la diabetes, entre otras cosas porque hay muchos tipos diferentes. De hecho, el momento de aparición de la enfermedad, las causas y los síntomas que presentan los pacientes dependen del tipo de diabetes.

Síntomas

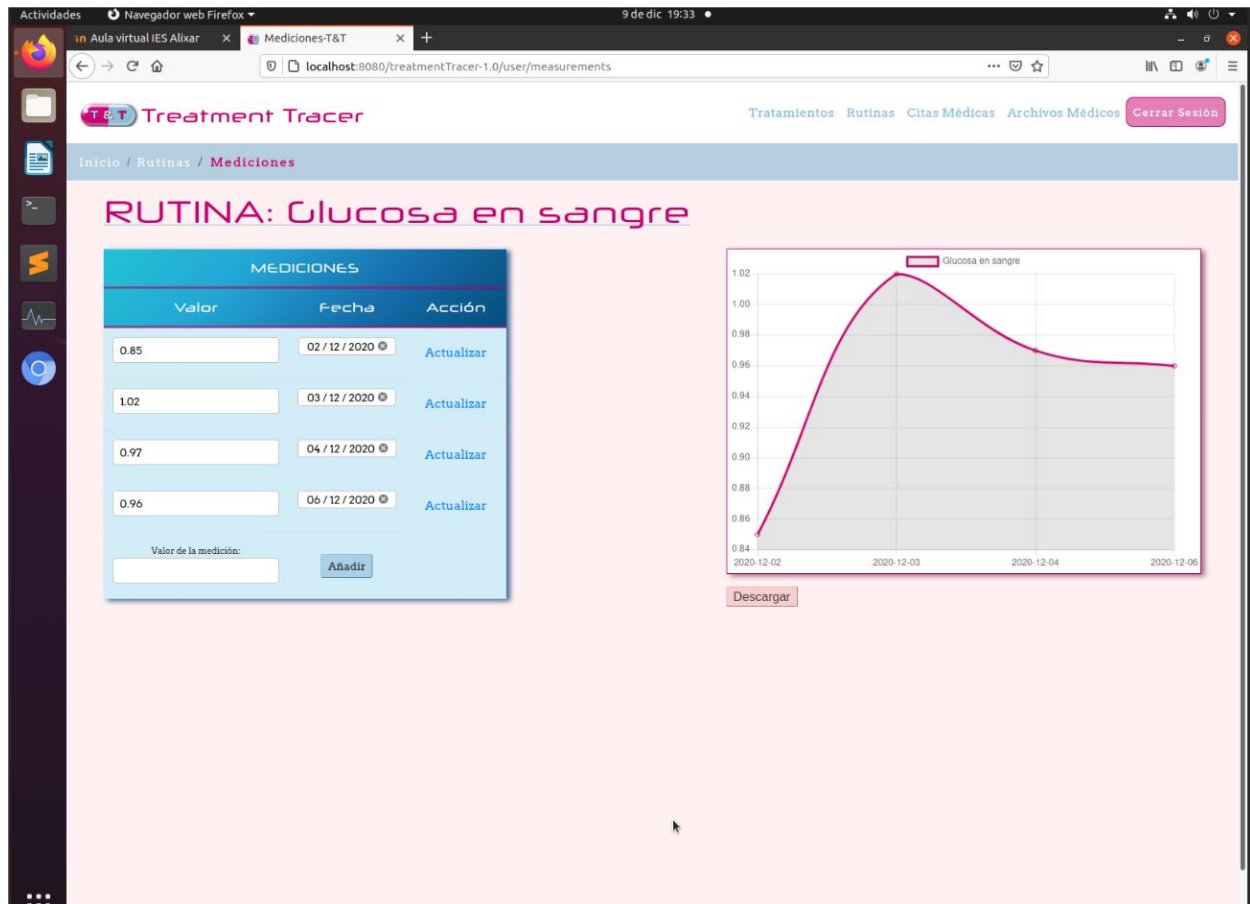
- Aumento de la sed
- ganas frecuentes de orinar
- hambre extrema
- pérdida de peso inexplicable
- presencia de cetonas en la orina (las cetonas son un subproducto de la degradación muscular y de grasas que se produce cuando no hay insulina suficiente disponible)
- fatiga
- irritabilidad
- visión borrosa.

Eliminar enfermedad

Borrar



Página de mediciones





Página de citas médicas

Citas en los próximos 30 días

- 2020-12-11: Revisión peso
- 2020-12-16: Analítica
- 2020-12-10: cita ejemplo 1
- 2020-12-09: otro ejemplo

lun.	mar.	mié.	jue.	vie.	sáb.	dom.
30	1	2	3	4	5	6
7	8	9 otro ejemplo	10 cita ejemplo 1	11 Revisión peso	12	13
14	15	16 Analítica	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

Anotar nueva cita

Fecha de la cita: dd/mm/aaaa

Título:

Especialidad:

Anotaciones:

Añadir