PROGRAMMING ENTERPRISE COMPONENTS

Coursework

IGNACIO GONZÁLEZ

15/3/2019

The University of Greenwich

Contenido

Introduction

This report´s aim is to describe the development of the required application. The project was divided into a Web Application and a Desktop Application. Each part follows different requirements which were described below.

Web Application Section

The student was asked to develop:

· Log in
· Add/edit/delete drivers
· Add/edit/delete qualifications
· Add/edit/delete training types
· Schedule sessions for a training type
· Assign drivers to a training session and record the outcome (e.g. training completed successful)
· Search for a driver and see his/her profile including qualifications and training
· See a list of drivers who have qualifications/training expiring within the next 30 days

Currently there are two qualifications which are relevant to the taxi drivers, which are their driving license and geographical tests (there are 5 different geographical tests: Central London, North London, South London, East London, West London).

Training types which the company is currently offering include advanced driving course, driving at night, cyclist awareness, reduce fuel consumption.

It must be possible for an admin user to view a list of drivers and for a particular driver show his/her qualifications and training. It should also be possible to get a list of drivers who have qualifications/training expiring within the next 30 days.

Desktop Application Section

For the intermediate functionality, you should extend the basic system by including additional features.
Create a Log component which can receive the following four inputs:
1. Start of day log
2. End of day log
3. Start of journey log
4. End of journey log

It should calculate journey durations (when receiving an end of journey log) and total hours worked in a day (when receiving an end of day log). It should report an error if it receives a log in an unexpected sequence (e.g. an end of journey log when no start of journey log was received).
The component should be exposed through a Web Service (SOAP or REST), so that it can be called from the driver app. The prototype of the driver app should be created as a Java desktop application and it must be possible to submit the four types of log listed above. A driver must also be able to log into the application on start-up, so the application knows whose driver's logs are being submitted.

The driver app should not connect directly to the database or the Log component, but all communication must be via the Web Service (remember that the company is planning to replace the desktop application with a mobile app at a later stage).

There is another section in which the student was asked to develop advanced functionalities, from this section, this project fulfils incident management through the Web Application. Session Beans and Message Driven Beans were not tackled.

Assumptions

- At the beginning of the database creation, trainings expire after the first year. For new trainings and qualifications, this feature can be modelled.
- Due to difficulties to measure time, we assume journeys will not last more than an hour.
- Each training lasts a few hours, that is why different levels of the same course were thought for this project. It could be possible to have more than one training in one particular day.
- It is always necessary to have an 'admin' user in the database with the purpose of accessing to the Web Application because it was designed for people with 'admin' level of authorization.

The project

Section 1

In this section ER-diagrams (ERD) will be explained. Starting with the first ERD developed during the first stages of the project. It was thought to represent the main modules of the model that would work behind scenes (**Figure 1**).

Figure 2 depicts the ERD in its final form. It is incomplete because of the magnitude of the project; main components were represented although. A controller class was added within the model, Web Services, Hibernate and Web Interface were implemented as well. The desktop application was designed with the purpose of requesting the service provided by the Web Service module in the Web Application. It also has a controller class (**Controller2.java**) which holds different static variables necessary for the correct performance. In case the reader wants to access to the real images, they will be attached to the main folder of the report due to their size.

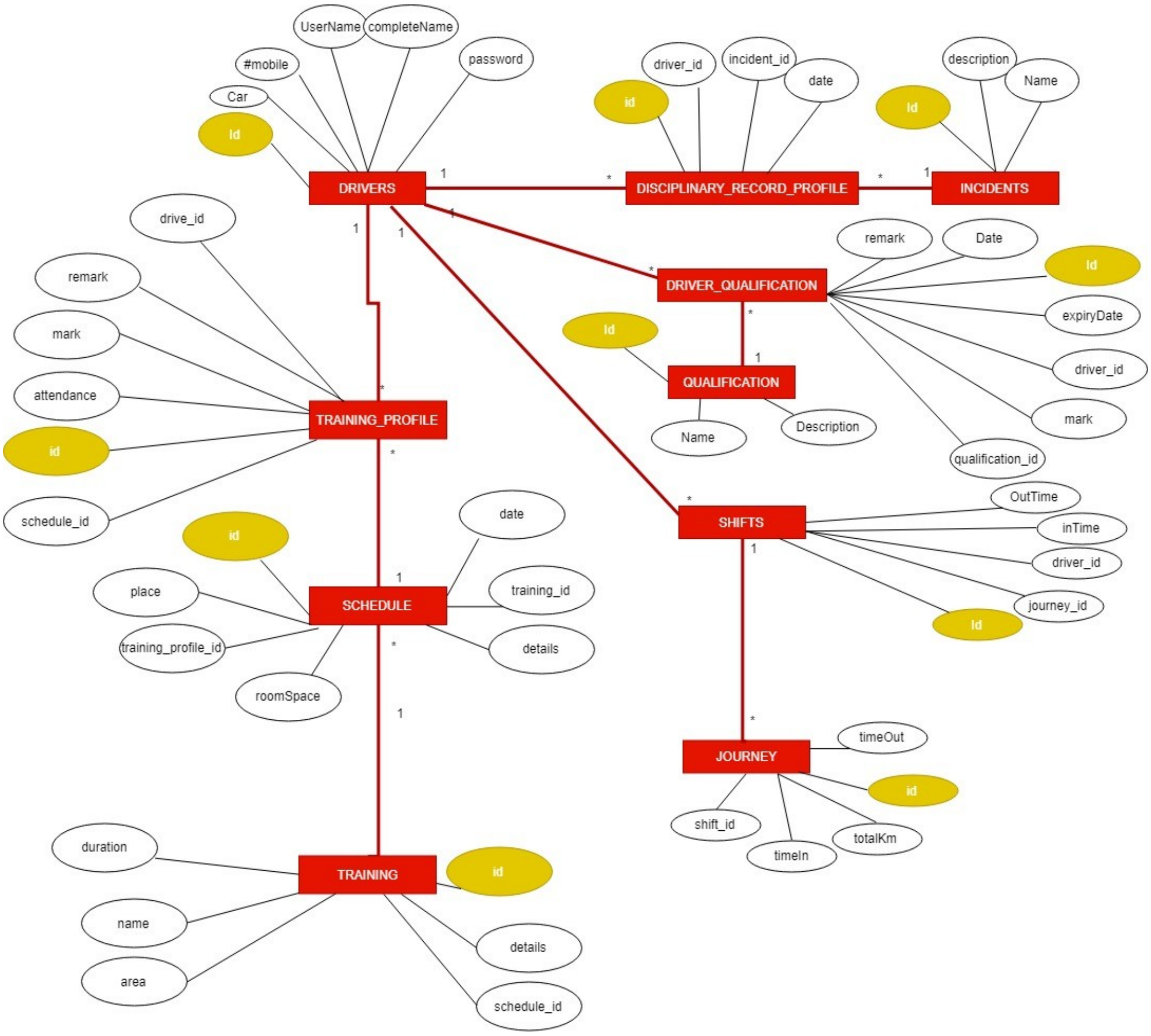


Figure 1 ERD during the first stages

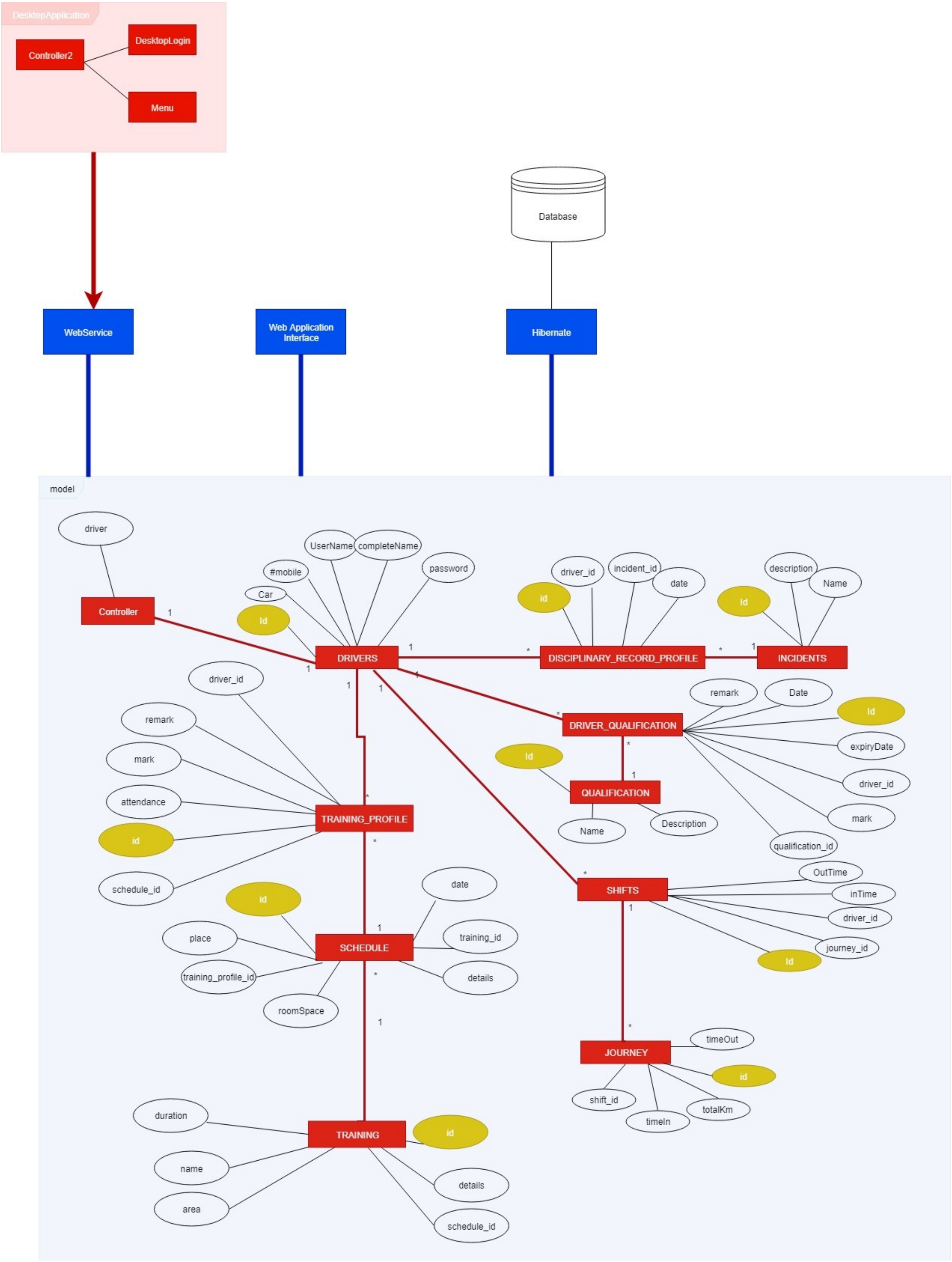


Figure 2 Final ERD

Section 2

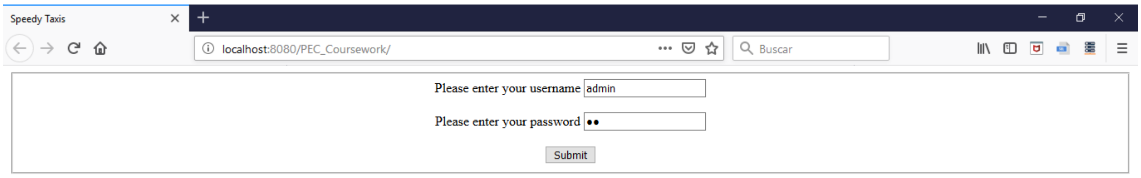
In this section, the required functionalities are explained using screenshots from the application.

Web Application

This application was thought to provide different services:

- To the administrator: Only administrators are allowed to use this application. They can perform different task like managing incidents, drivers, qualifications, and trainings. They also can set up dates for trainings and evaluate them. The vast majority of the fields that are required to be filled by the user are validated, in case the user clicks by accident in 'submit' without filling all of them, the system will give them an error. Functionalities will be explained specifically as we go over this section.
- To external applications: Like the desktop application. They can ask for services to the main application with the purpose of managing shifts/journeys. They record relevant information about each shift and its journeys and retrieve these data to the database through the web service.
- Feedback is provided either if something goes wrong or not. The applications will display an error message when the users are trying to perform an incorrect action regarding the application´s workflow.

Figure 3 shows the first screen of the application, the 'Log In'. It is required to log in with the username "admin", otherwise it will give us error.



Please enter your username admin

Please enter your password ••

Figure 3 Log In

After the administrator accesses, an initial menu is displayed showing different functionalities as shown in **Figure 4**.

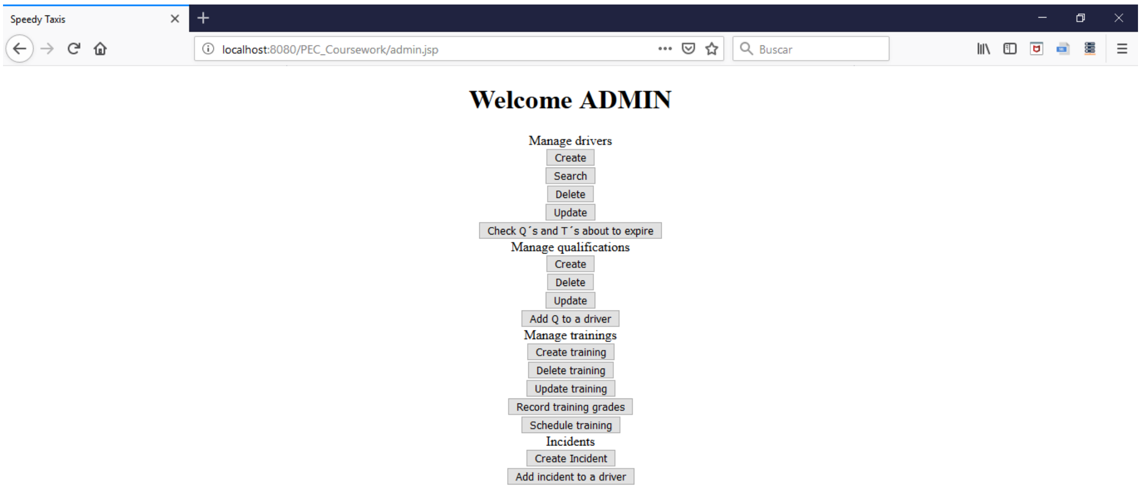


Figure 4 Initial menu

Figures 5, 6, 7 , 8, and 9 show different functionalities regarding the drivers. It is possible to create, delete, update , and search drivers. The results´ format is shown in the **Figure 7**, it displays the driver´s details, qualifications, trainings and incidents.

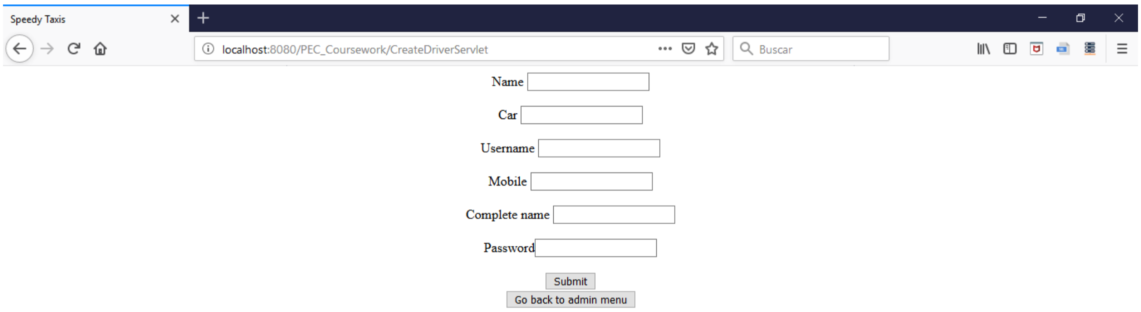


Figure 5 Add driver

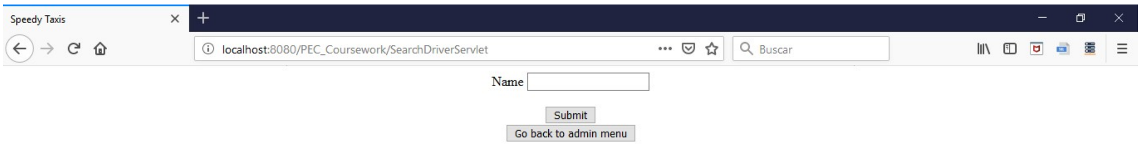
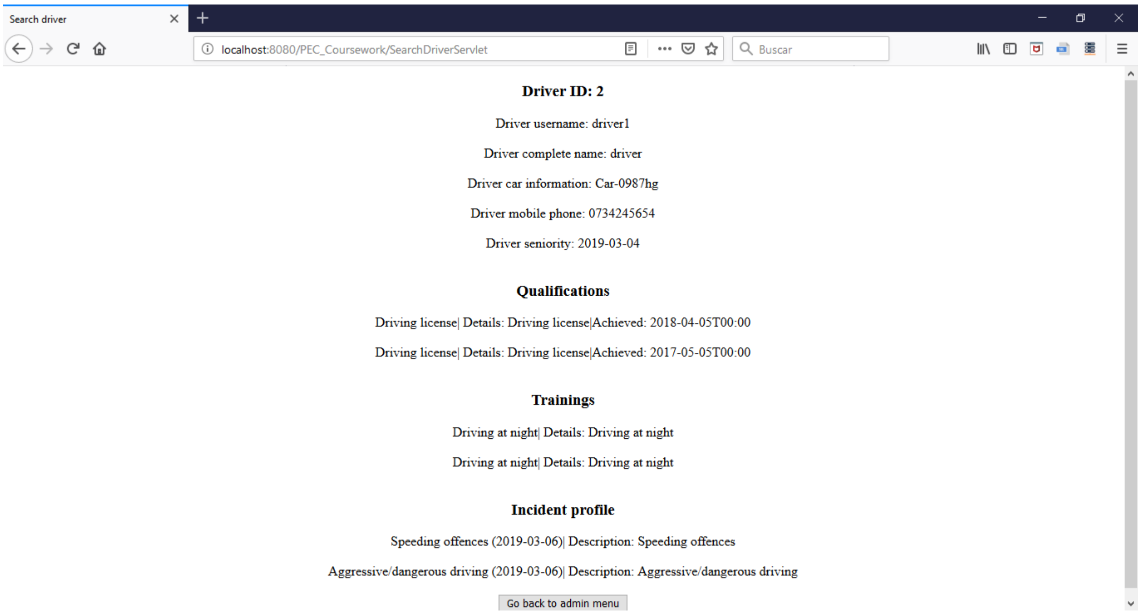


Figure 6 Search driver



Driver ID: 2

Driver username: driver1

Driver complete name: driver

Driver car information: Car-0987hg

Driver mobile phone: 0734245654

Driver seniority: 2019-03-04

Qualifications

Driving license| Details: Driving license|Achieved: 2018-04-05T00:00

Driving license| Details: Driving license|Achieved: 2017-05-05T00:00

Trainings

Driving at night| Details: Driving at night

Driving at night| Details: Driving at night

Incident profile

Speeding offences (2019-03-06)| Description: Speeding offences

Aggressive/dangerous driving (2019-03-06)| Description: Aggressive/dangerous driving

Figure 7 Search driver

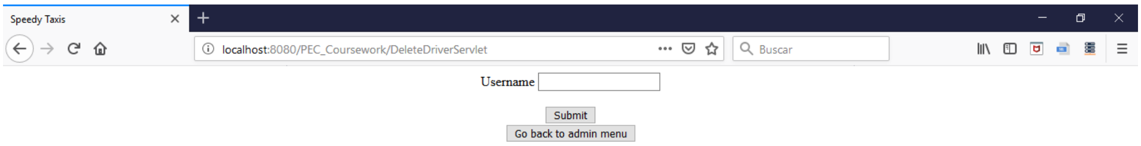
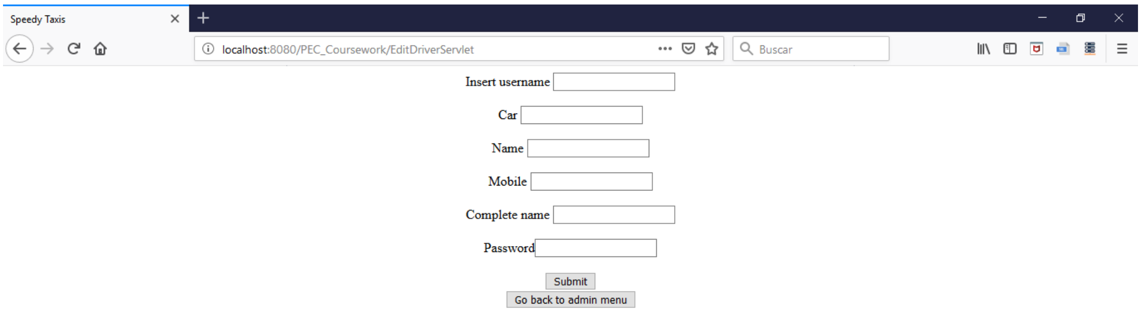


Figure 8 Delete driver



localhost:8080/PEC_Coursework/EditDriverServlet

Insert username

Car

Name

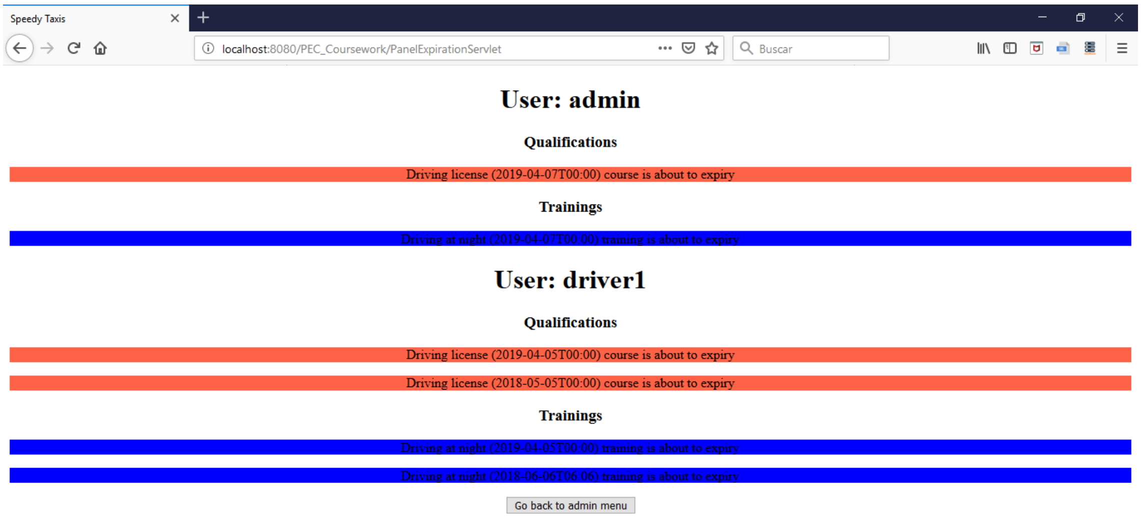
Mobile

Complete name

Password

Figure 9 Update driver

With the purpose of allowing the administrator to know about the courses/trainings that are about to expire, they can access to the next panel which shows the names of the drivers who have courses/trainings about to expiry (they are highlighted one month before expiring).



localhost:8080/PEC_Coursework/PanelExpirationServlet

User: admin

Qualifications

Driving license (2019-04-07T00:00) course is about to expiry

Trainings

Driving at night (2019-04-07T00:00) training is about to expiry

User: driver1

Qualifications

Driving license (2019-04-05T00:00) course is about to expiry

Driving license (2018-05-05T00:00) course is about to expiry

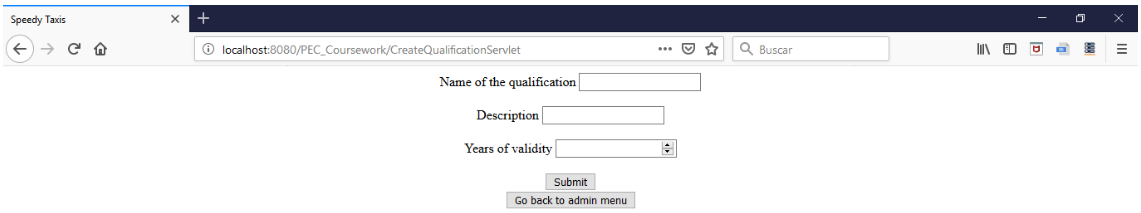
Trainings

Driving at night (2019-04-05T00:00) training is about to expiry

Driving at night (2018-06-06T06:06) training is about to expiry

Figure 10 Expiration courses panel

In **Figures 11, 12, and 13** functionalities for adding, deleting and updating qualification are shown.



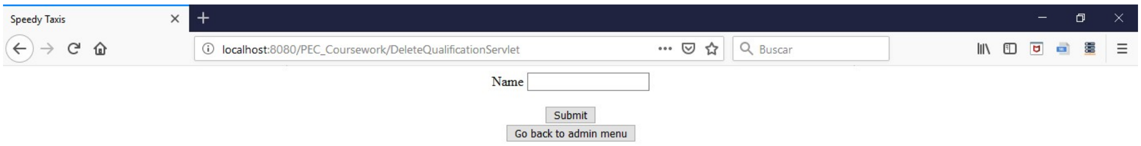
localhost:8080/PEC_Coursework/CreateQualificationServlet

Name of the qualification

Description

Years of validity

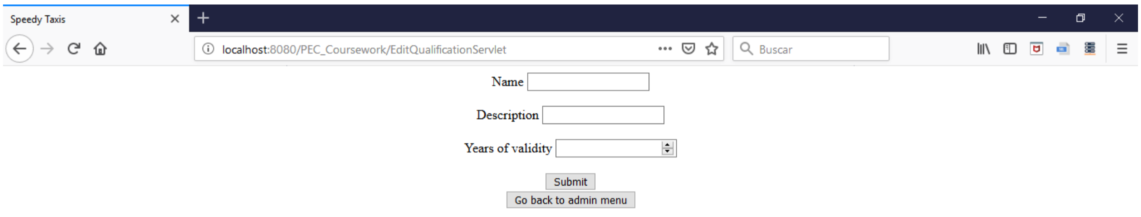
Figure 11 Create qualification



localhost:8080/PEC_Coursework/DeleteQualificationServlet

Name

Figure 12 Delete qualification



Name

Description

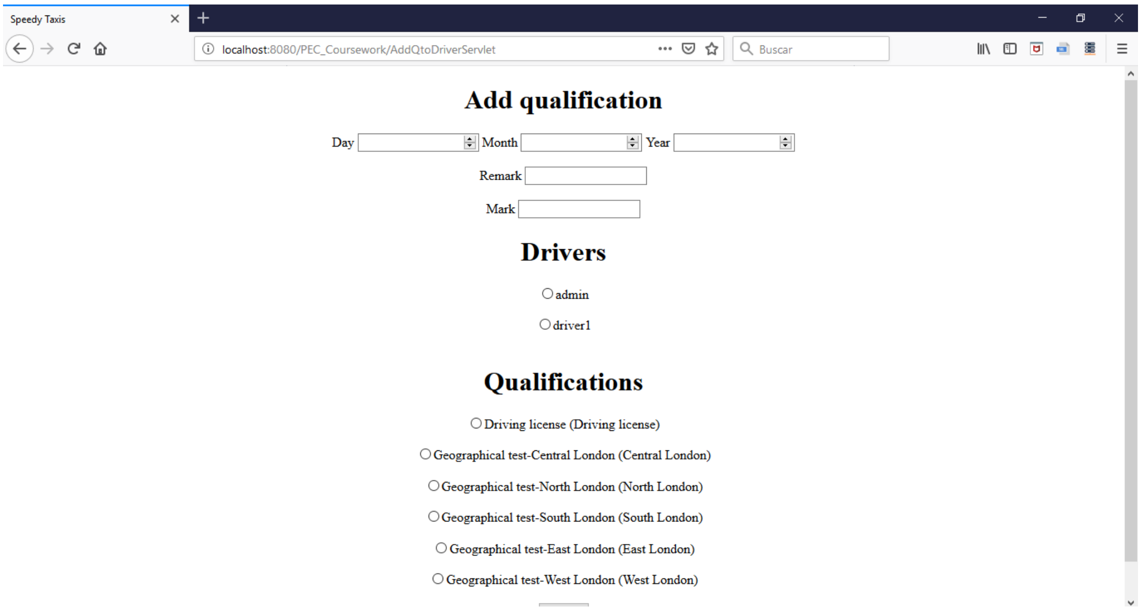
Years of validity

Submit

Go back to admin menu

Figure 13 Update qualification

The functionality shown in the **Figure 14** deals with assigning qualification to a particular driver. In this case, we can choose between driving license or different geographical tests.



Add qualification

Day Month Year

Remark

Mark

Drivers

○ admin

○ driver1

Qualifications

○ Driving license (Driving license)

○ Geographical test-Central London (Central London)

○ Geographical test-North London (North London)

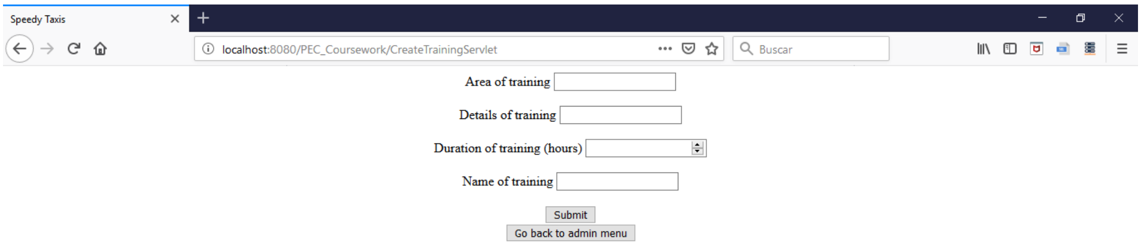
○ Geographical test-South London (South London)

○ Geographical test-East London (East London)

○ Geographical test-West London (West London)

Figure 14 Add qualification to a driver

Figures 15,16, and 17 show the functionalities related to adding, deleting and updating trainings.



Area of training

Details of training

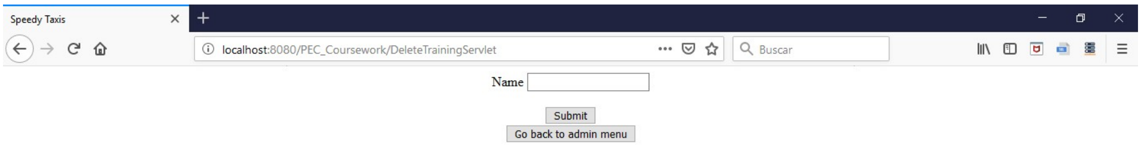
Duration of training (hours)

Name of training

Submit

Go back to admin menu

Figure 15 Create training

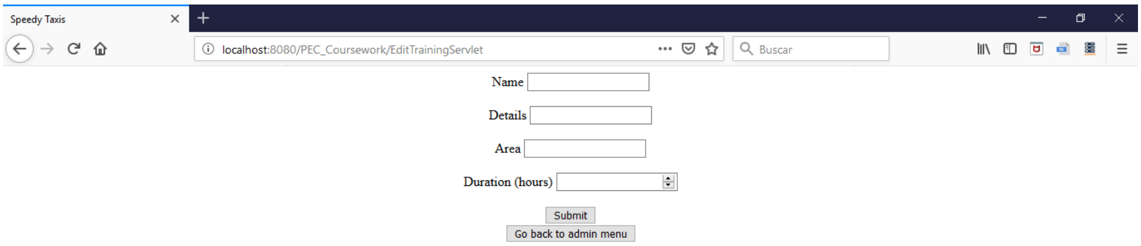


Name

Submit

Go back to admin menu

Figure 16 Delete training



Name

Details

Area

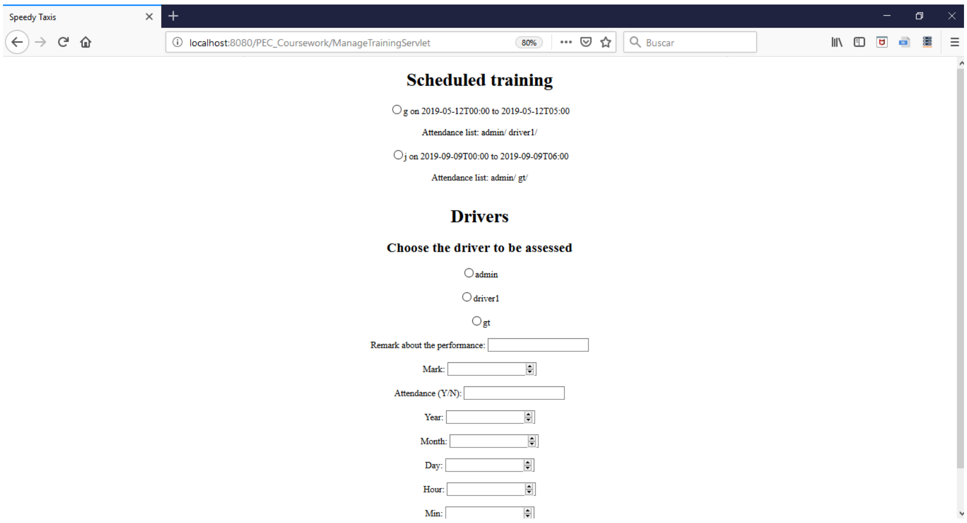
Duration (hours)

Submit

Go back to admin menu

Figure 17 Update training

The application allows us to schedule trainings as shown in **Figure 19** and to record the marks/ grades that a driver obtained in a particular training (shown in **Figure 18**). It is necessary to schedule a particular training before assessing any driver for that training, otherwise it will not appear. <u>When assessing drivers, each course will show the drivers that are expected to attend</u>.



Scheduled training

○g on 2019-05-12T00:00 to 2019-05-12T05:00

Attendance list: admin/ driver1/

○j on 2019-09-09T00:00 to 2019-09-09T06:00

Attendance list: admin/ gt/

Drivers

Choose the driver to be assessed

○admin

○driver1

○gt

Remark about the performance:

Mark:

Attendance (Y/N):

Year:

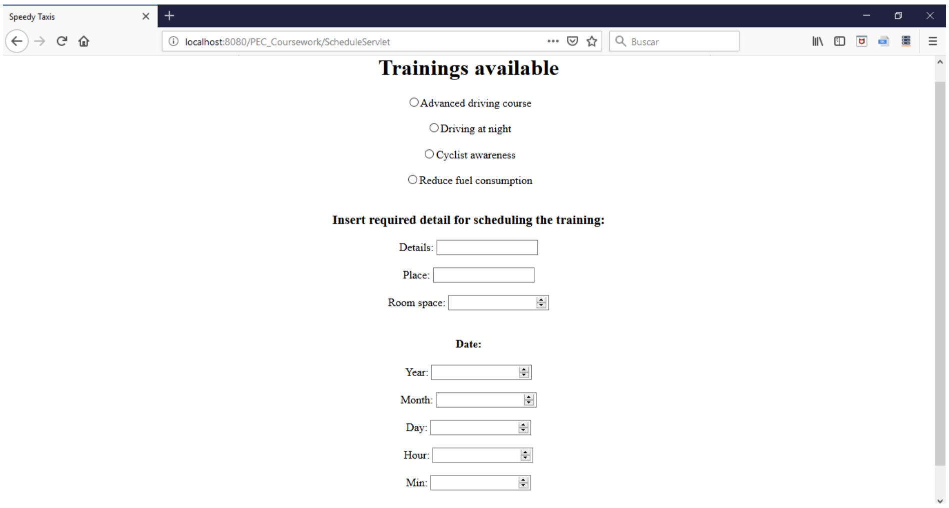
Month:

Day:

Hour:

Min:

Figure 18 Record training grades



Trainings available

○Advanced driving course

○Driving at night

○Cyclist awareness

○Reduce fuel consumption

Insert required detail for scheduling the training:

Details:

Place:

Room space:

Date:

Year:

Month:

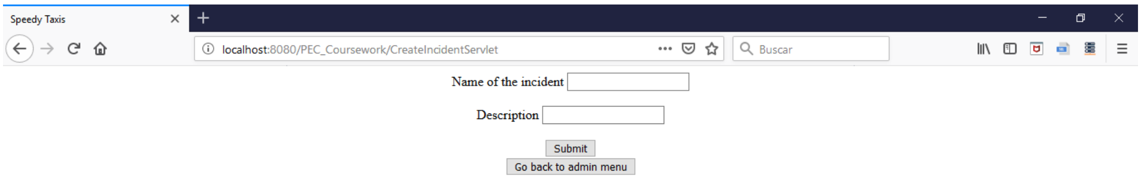
Day:

Hour:

Min:

Figure 19 Schedule training

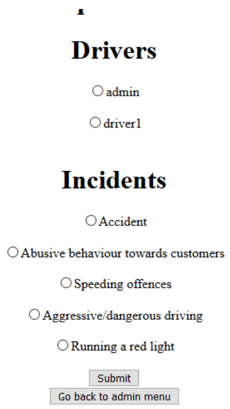
Finally, the administrator will be able to create and assign incidents to an incident profile that is part of the drivers´ general profile as shown in **Figure 20 and 21**.



Name of the incident

Description

Figure 20 Create incident



Drivers

○ admin

○ driver1

Incidents

○ Accident

○ Abusive behaviour towards customers

○ Speeding offences

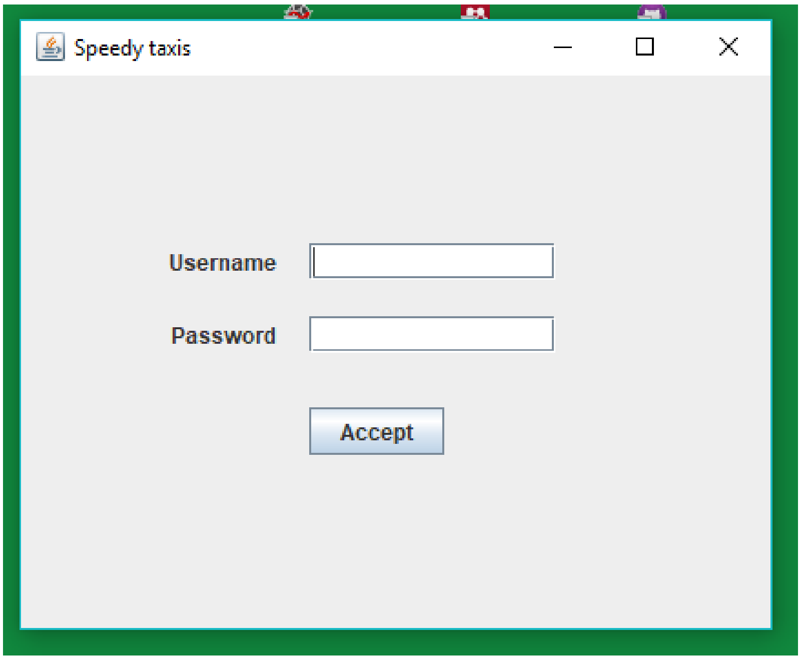
○ Aggressive/dangerous driving

○ Running a red light

Figure 21 Add incident to a driver

Desktop Application

Figure 22 shows the Log In screen appearance. In case wrong credentials are provided, it would launch an error dialog screen asking the user to try again.



Username

Password

Figure 22 Log In

Once the driver accesses to the application, it will show the ‘driving menu’ (**Figure 23**) in which the driver can manage shifts and journeys. The order of the sequences cannot be broken, this means:

In first place, the driver has to clock in by clicking 'Start shift'. Once it is done, they will have the possibility of either finishing the shift right after starting (because any hypothetical emergency) or starting a new journey. In case a new journey has started, it is needed to finish the journey before finishing the shift (allowing the system to record the data). In case, the driver clicks over the wrong button making the sequence to be broken, an error message will appear. The information related to the places the taxi is going to or coming from can be typed any time before finishing the journey.

All the information will be dumped into the database, in which the system will record relevant information about the shifts and journeys like the duration of the journeys and places. The duration of the shift will be the sum of the duration of all the journeys of this particular shift. In the 'driving menu' the number of minutes the driver has worked is shown in a label near to the 'finish journey' button as well as the state of the driver (clocked out/clocked in) and their availability in during the working hours (on duty/off duty).

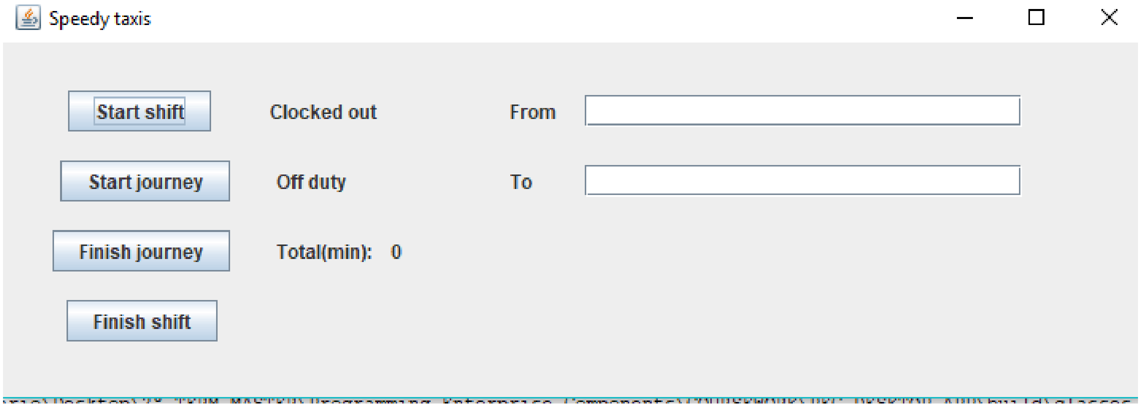
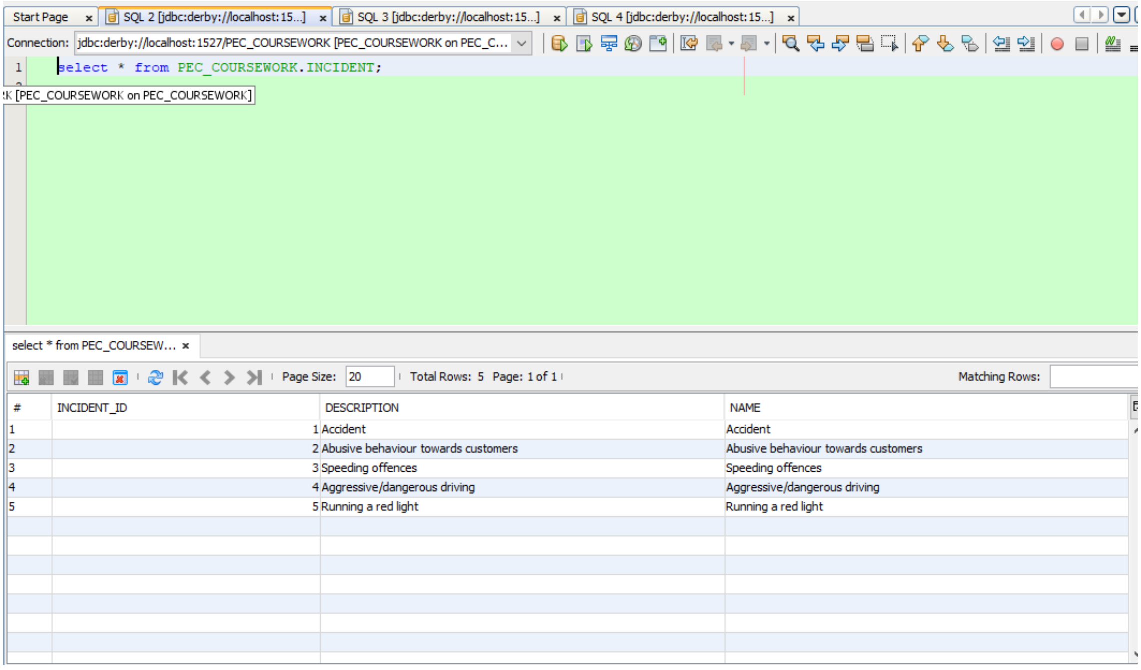


Figure 23 Driver menu

Initial records of the database

Required records to start working with the system.


Start Page
SQL 2 [jdbc:derby://localhost:15...]
SQL 3 [jdbc:derby://localhost:15...]
SQL 4 [jdbc:derby://localhost:15...]
select * from PEC_COURSEW...
Matching Rows:

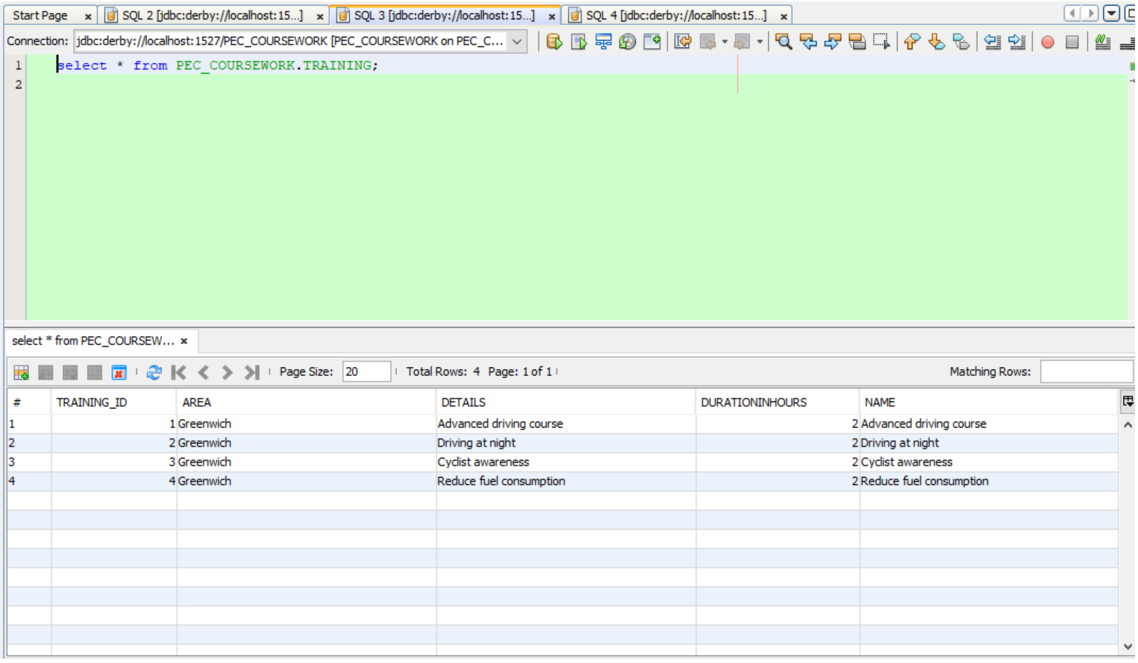
Connection: jdbc:derby://localhost:1527/PEC_COURSEWORK [PEC_COURSEWORK on PEC_C...

```
select * from PEC_COURSEWORK.INCIDENT;
```

Page Size: 20 Total Rows: 5 Page: 1 of 1

#	INCIDENT_ID	DESCRIPTION	NAME
1	1	Accident	Accident
2	2	Abusive behaviour towards customers	Abusive behaviour towards customers
3	3	Speeding offences	Speeding offences
4	4	Aggressive/dangerous driving	Aggressive/dangerous driving
5	5	Running a red light	Running a red light

Figure 24 Initial Incident records


Start Page
SQL 2 [jdbc:derby://localhost:15...]
SQL 3 [jdbc:derby://localhost:15...]
SQL 4 [jdbc:derby://localhost:15...]
select * from PEC_COURSEW...
Matching Rows:

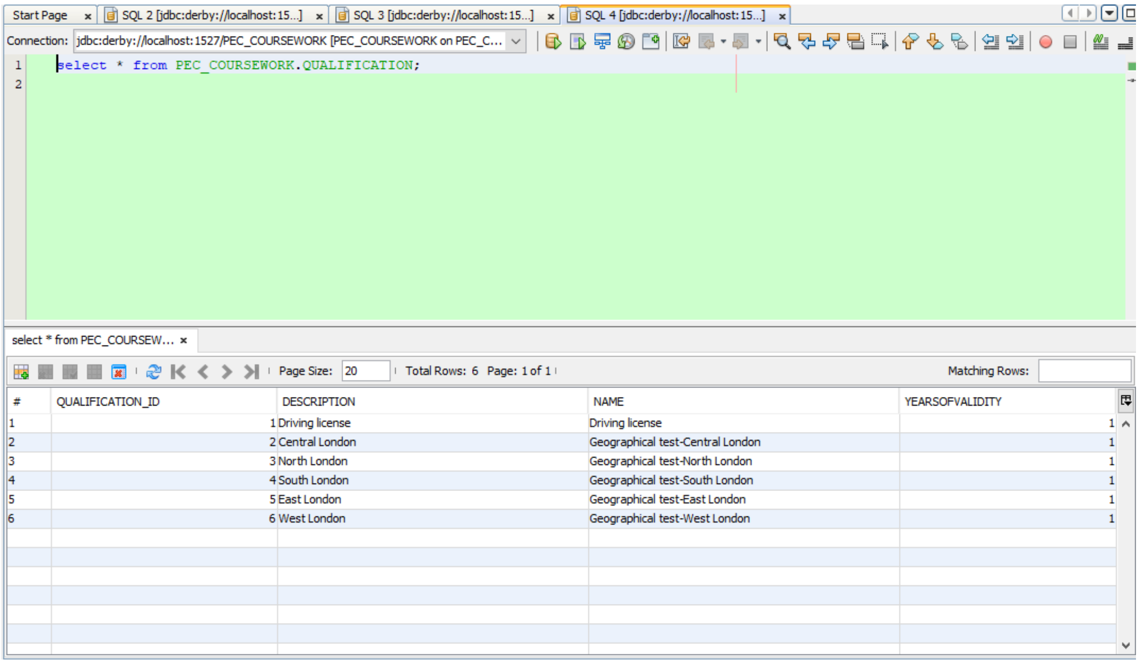
Connection: jdbc:derby://localhost:1527/PEC_COURSEWORK [PEC_COURSEWORK on PEC_C...

```
select * from PEC_COURSEWORK.TRAINING;
```

Page Size: 20 Total Rows: 4 Page: 1 of 1

#	TRAINING_ID	AREA	DETAILS	DURATIONINHOURS	NAME
1	1	Greenwich	Advanced driving course	2	Advanced driving course
2	2	Greenwich	Driving at night	2	Driving at night
3	3	Greenwich	Cyclist awareness	2	Cyclist awareness
4	4	Greenwich	Reduce fuel consumption	2	Reduce fuel consumption

Figure 25 Initial Training records


Start Page
SQL 2 [jdbc:derby://localhost:15...]
SQL 3 [jdbc:derby://localhost:15...]
SQL 4 [jdbc:derby://localhost:15...]
select * from PEC_COURSEW...

Connection: jdbc:derby://localhost:1527/PEC_COURSEWORK [PEC_COURSEWORK on PEC_C...

```
select * from PEC_COURSEWORK.QUALIFICATION;
```

Page Size: 20 Total Rows: 6 Page: 1 of 1 Matching Rows:

#	QUALIFICATION_ID	DESCRIPTION	NAME	YEARSOFVALIDITY
1	1	Driving license	Driving license	1
2	2	Central London	Geographical test-Central London	1
3	3	North London	Geographical test-North London	1
4	4	South London	Geographical test-South London	1
5	5	East London	Geographical test-East London	1
6	6	West London	Geographical test-West London	1

Figure 26 Initial Qualification records

Section 3

In this section, an evaluation will be conducted with the purpose of critically analyzing the different parts of the project.

Regarding the first part of the project (Basic Functionalities): It was difficult at the beginning stage since it was the first time the student was working with these tools. The fact that the student already knew how to program in object-oriented languages was really useful as he knew how to start researching solutions. All the requirements were fulfilled using NetBeans and NetBeans Data Persistence (Database). The Web Application was developed using servlets, thus the project has different packages that deal either with the web application and the logic classes. MVC (Singleton was implemented inside this class as well) pattern was used so as to have a class which controls all the relevant data for the web application. Regarding frameworks, 'Hibernate' was the selected tool in this case. It is easy to be used and it allows changes easily.

From the second part, all the requirements were achieved. The application is user-friendly although very simple regarding the user interface. However, for this hypothetical case, the company asked us to develop a prototype that would be substituted by the real mobile application. Therefore, it focuses on covering the critical functionalities.

It is necessary to mention that from the Advanced Functionalities, the possibility of adding and assigning incident through the web page was achieved. Therefore, each driver has a profile in which incidents are described.

Future work

- It could be interesting to make the measurement of journeys´ times more accurate. Eliminating the second assumption.

- When assessing drivers during trainings, the system only allows the admin to assess one driver. In normal systems, all the drivers scheduled for these trainings would appear and the admin would go through all of them, assessing their performance.
- The GUI (Graphical User Interface) focus on the functionality more than on the user experience. Improving it could be another task for future work.
- Finish all the advanced functionalities of the coursework.

Section 4

In this section, a little overview of the different tools/technologies regarding Java EE that the student learnt during lectures will be developed.

Why do enterprises use Java EE?

Java EE is a programming platform designed for enterprises with the purpose of developing Java-based projects which must fulfil particular standards and requirements. Java EE provides different kinds of API specifications like JDBC, RMI, e-mail, XML, etc. And it has components like servlets, portlets, JavaBeans, etc. All of these features allow developers to create portable and scalable enterprise applications. The vast majority of these applications work with patterns and frameworks that are pieces of code that you 'insert' in your code with the purpose of achieving more functionalities.

Java™
ENTERPRISE EDITION
DEVELOPER PRODUCTIVITY
Java EE 7
HTML
MEETING ENTERPRISE DEMANDS

Figure 27 Areas approached by Java EE

But, '*why do enterprises use Java EE?*'

These organizations need highly scalable and efficient applications for service delivering and assets utilization together with high levels of security. It enables developers that want to create distributed system to benefit from patterns, frameworks, emerging usages, etc (Oracle Corporation, 2013).

Distributed Systems

This concept describes a collection of autonomous computers connected by a network that treats all the users as a unique entity. Within this field we have to pay attention to features like resource sharing processes, openness, concurrency, the previously mentioned scalability, error tolerance, etc. Also, there are different approaches regarding the architecture.

- Architecture based on components: Focusing on the component development. A component is a piece of code that shows a well-defined interface and can work together with other components.

- Architecture based on services (SOA): Supports service delivering so as to help the business requirements. It allows the creation of high-scalable system which facilitates the interaction between systems.
- Architecture based on layers: Hierarchical distribution of layers and each one has a role in order to provide an effective solution (top-down and bottom-up architecture).

Even though Java EE has lots of functionalities regarding enterprise concerns, this is not the only technology of this type. *Alternatives* could be found in .NET, PHP or C-C++ (Serra Manchado & Franco Puntes, 2013).

PHP

It allows the creation of systems that can be executed in any server from a web page viewer program and provide answers depending on the data that the user is typing. PHP is easy, quick and user-friendly. Java Beans are slower than PHP, but Java is more portable and compatible.

C-C++

This technology was designed during 1970 focusing on improving the expensive hardware available during these years. Nowadays, due to portability and scalability provided by Java, C-C++ do not provide the required standards for big projects like the ones developed in J2EE.

.NET

Microsoft project that focuses on developing a platform in which software based on network transparency and quick application development is the key. It provides a range of solutions for different languages and it allows the operability with different environments (platform-independent) through MSIL (Microsoft Intermediate Language). It gives us Web and XML service support as well as COM (Component Object Model) support.

Some advantages

- Multiple language support while J2EE only works with Java
- More user-friendly tools than the homologous ones in J2EE

Drawbacks

- Strict security model against viruses
- Compatibility

'Technologies we already saw in this project'

Regarding to this project, technologies like servlets, servers and Web Services were used.

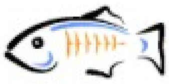


Figure 28 GlassFish logo was based on a real fish that has a transparent body

In case of the server, GlassFish-4.1 was used. It is an open-source Java EE application server that provides us the required Java EE specifications to deploy Java applications. It supports the latest versions of Java EE and it has commercial support available. It could be interesting to bring up the project developed by Costin-Marius and Sorin-Aurel (2012) as an example of an application based on GassFish. They developed a web application for industrial software monitoring using GlassFish together with a mobile application based on Adobe PhoneGap. **Figure 29** shows the system architecture.

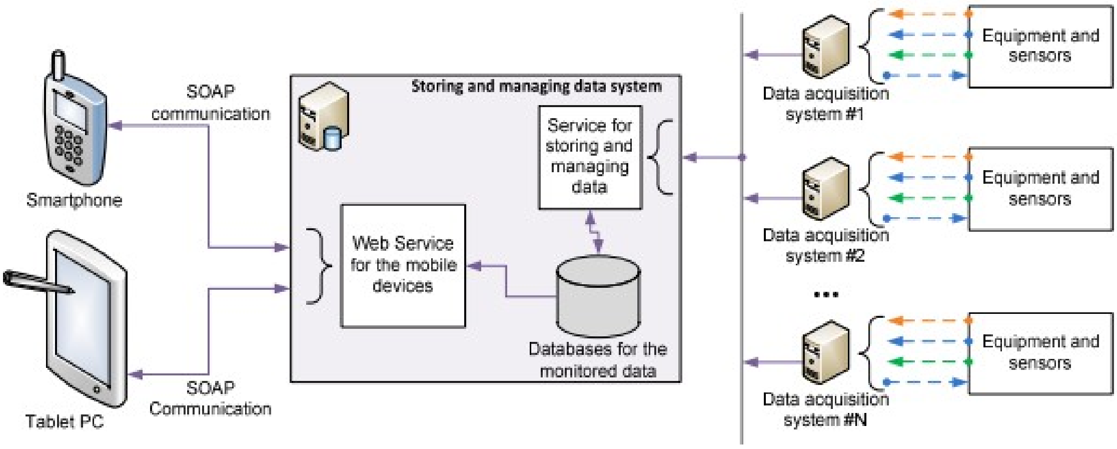

SOAP communication
Smartphone
Tablet PC
SOAP Communication
Storing and managing data system
Service for storing and managing data
Web Service for the mobile devices
Databases for the monitored data
Data acquisition system #1
Data acquisition system #2
...
Data acquisition system #N
Equipment and sensors
Equipment and sensors
Equipment and sensors

Figure 29 Proposed system architecture

GlassFish is composed of different modules, saving memory and space due to the fact that we can just download the modules we need (Serra Manchado & Franco Puntes, 2013).

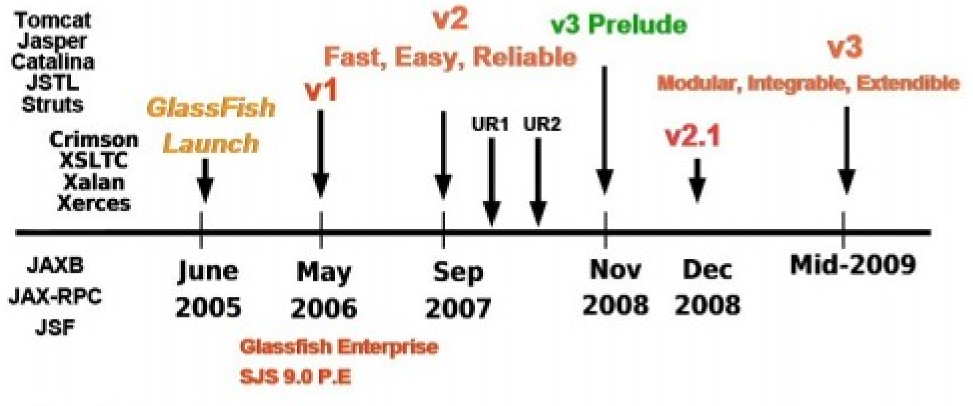


Figure 30 GlassFish evolution

The web application communicates with the desktop application through SOAP service procedures. SOAP is an XML-based protocol for messaging. It works on an existing transport protocol such as HTTP. It dictates which users will receive messages and how they must process them. For the communication between services and clients it uses WSDL files, documents that provides the web service´s interface and point of contact (Curbera et al., 2002).

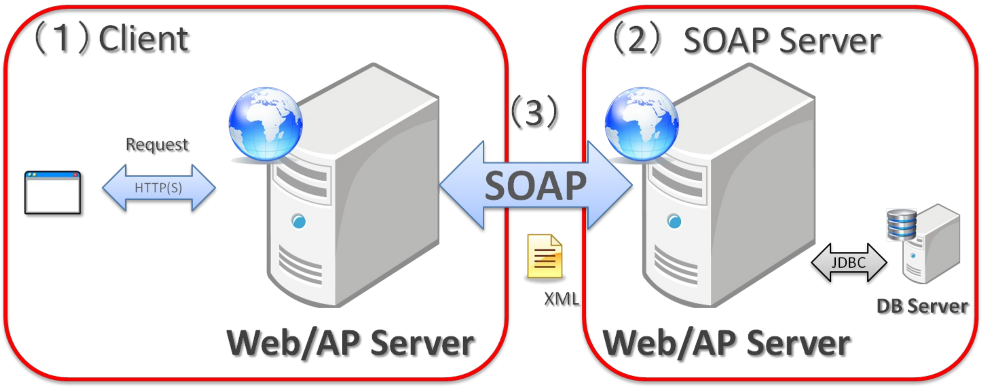


Figure 31 SOAP communication architecture (Terasoluna.org)

References

Oracle Corporation (2013), Introduction to Java Platform, Enterprise Edition 7. Available at: https://www.oracle.com/technetwork/java/javaee/javaee7-whitepaper-1956203.pdf [Accessed: 15/03/2019]

Serra Manchado, D., & Franco Puntes, D. (2013). Estudio del servidor de aplicaciones Glassfish y de las aplicaciones J2EE. Available at: https://ddd.uab.cat/pub/trerecpro/2013/hdl_2072_206748/SerraManchadoDavidR-ETISa2009-10.pdf [Accessed: 14/03/2019]

Grigorescu, C. M., Moraru, S. A., & Grama, C. (2012, October). Industrial software monitoring system extension for mobile devices based on GlassFish and PhoneGap. In 2012 International Conference and Exposition on Electrical and Power Engineering (pp. 756-761). IEEE. Available at: https://ieeexplore.ieee.org/abstract/document/6463839 [Accessed: 14/03/2019]

Curbera, F., Duftler, M., Khalaf, R., Nagy, W., Mukhi, N., & Weerawarana, S. (2002). Unraveling the Web services web: an introduction to SOAP, WSDL, and UDDI. IEEE Internet computing, 6(2), 86-93. Available at: https://ieeexplore.ieee.org/abstract/document/991449 [Accessed: 14/03/2019]

Terasoluna.org, TERASOLUNA Server Framework for Java (5.x) Development Guideline [Online] Available at: https://terasolunaorg.github.io/guideline/5.2.0.RELEASE/en/index.html [Accessed: 15/03/2019]