Vilniaus Gedimino technikos universitetas

Programming Techniques Course Project

Author:
Paolo Wattebled
(ID: 20224582)

Lecturer:
prof. dr. Jūratė Janutėnaitė-Bogdanienė

Spring Semester 2023

Contents

1 Field Analysis

1.1 Objectives

The project aims to develop a Java application for a driver company. The application will have a user interface and a database with a Web API. The user interface will allow the users to interact with the application and perform various tasks such as adding drivers, trucks, managing trips or creating forms to interact with each other. The database will store all the information about the drivers, trucks, trips, cargo and forms. And finally, the Web API will allow the user interface to communicate with the database and perform CRUD operations.

1.2 Scope

The user interface will be developed using JavaFX and will be designed to be user-friendly.

The database will be designed using SQL and Hibernate. The database will store all the data related to the drivers, trucks, trips, cargo and forms. The design of the database will be based on the requirements of the application and will be optimized for efficient storage and retrieval of data.

The Web API will be developed using REST API thanks to Spring Boot. The Web API will allow the user interface to communicate with the database and perform CRUD operations. This will enable the users to access the data on their mobile devices thanks to a potential Android application or other web-based applications.

1.3 System User

In this application, there will be two types of users; managers and drivers. The managers will be able to add, modify and delete cargo, trucks, trips and forms. The drivers will be able to view their assigned trips and available ones, as well as create forms. Moreover, a manager could be an administrator, giving him the ability to add, modify and delete users.

Otherwise, this application, as said before, will interact with a database and a Web API, offering the possibility to access the data on mobile devices and not just on the desktop application.

2 System Architecture

2.1 Data structures

To get a view of the data structures, here's the class diagram of the application:

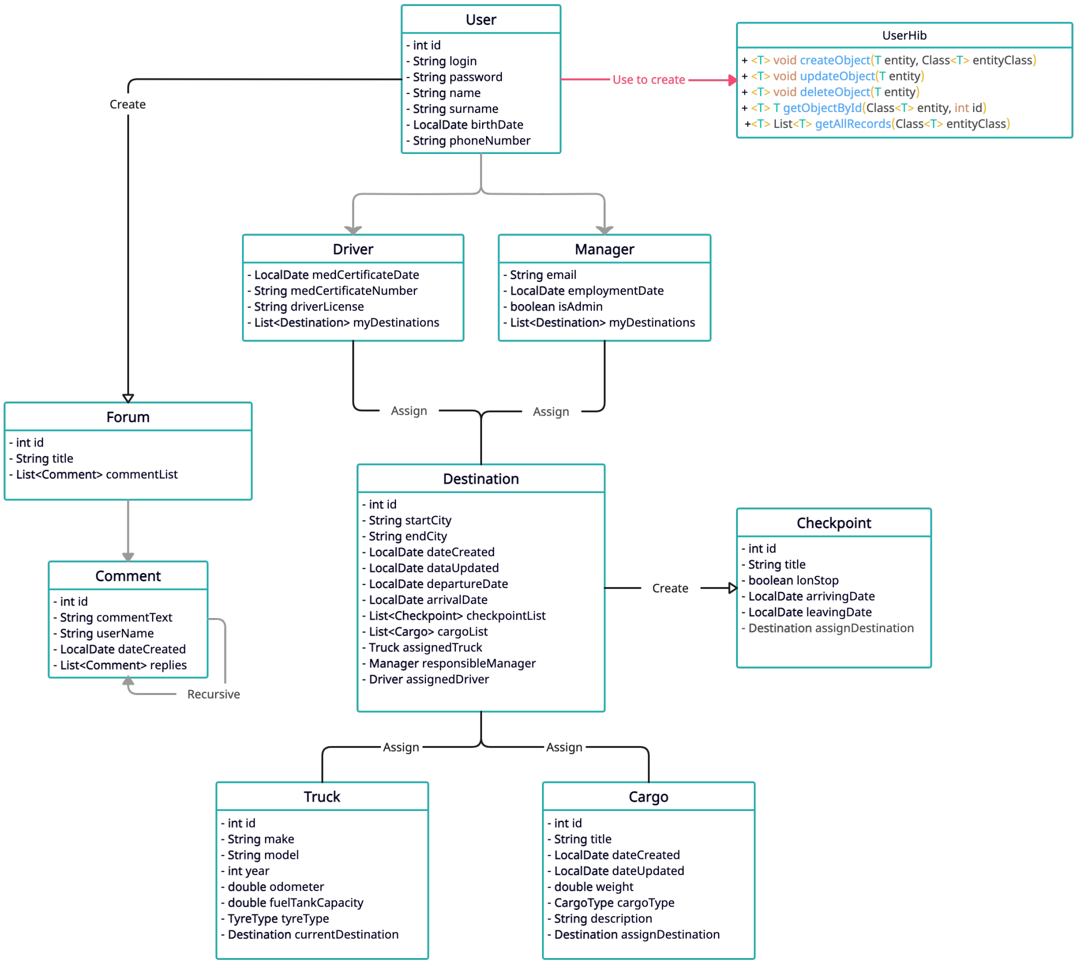


Figure 1: Class Diagram

As we can see, the application will be composed of 5 main classes: User, Truck, Destination, Cargo and Form. Each of these classes will have its attributes and methods. Plus, each class will be linked together. For example, the destination will have an assigned manager, driver and truck. To get a better view of the relationships between the classes, here's the database diagram:

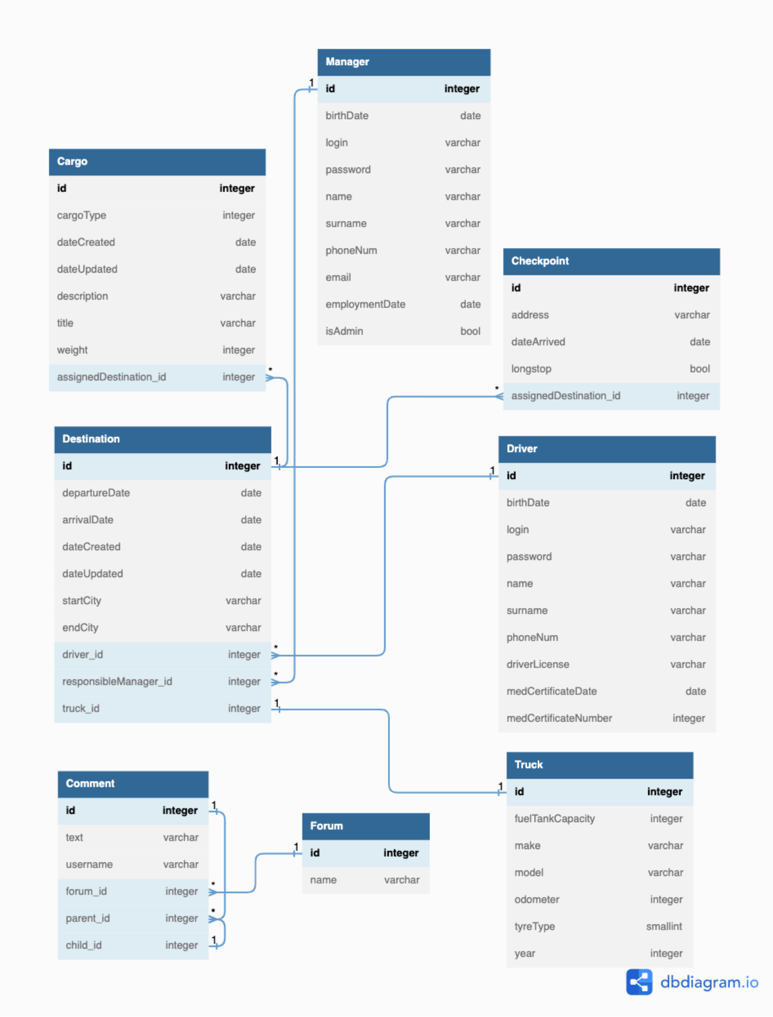


Figure 2: Database Diagram

2.2 Additional libraries

Java is a popular programming language with an extensive library of resources available for developers. One such library, Project Lombok, can help reduce the amount of boilerplate code. For instance, instead of writing lengthy getter and setter for each attribute, developers can use the annotations *@Getter* and *@Setter* to generate the code automatically. This not only makes the code more concise but also easier to read and maintain.

Another useful tool that I utilized for creating the user interface is *SceneBuilder*. With this visual design tool, developers can create the user interface and see how it looks without having to run the application. Furthermore, it generates FXML code, eliminating the need for manual coding and making it easier to develop complex user interfaces. Overall, these tools help streamline the development process, saving time and improving the quality of the code.

2.3 Main functions of the system

One of the main functions, if not the most important, is the ability to create, modify and delete a destination. Plus, this application got a login system, allowing the users to access the application with their credentials. Finally, it's possible to create a form, where it will be possible for the user to add comments and respond to the others. Now, let's go deeper into the main functions of the system.

2.3.1 Create, modify and delete a destination

When a manager is logged in, he will have access to a destination management panel where he can input different fields. He will have to enter the start and end location, the start and end date, the assigned truck, the assigned driver and the assigned manager. Another option that is available to the manager is to create a checkpoint for the destination. This will allow the driver to take a break (small or long) if he needs to.

To continue with the destination, when it is a driver that is logged in, he will have access to his assigned destination and will be able to add or delete a checkpoint.

Creating a destination means creating a cargo to be transported. That's why when a manager wants to create a cargo, he can assign a cargo to a specific destination.

2.3.2 Login and register system

The login system is a very important part of the application. Indeed, it allows the users to access the application with their credentials. The login system is composed of a login page and a register page. The login page will allow the users to enter their credentials and access the application. The register page will allow the users to create an account choosing if they wanna be a manager or a driver. Plus, he will have to choose a username and a password.

Now if a user wants to log in and enters the wrong credentials (password or login), he won't be able to access the application and will be stuck on the login page.

2.3.3 Form and comments

This application will have a form system. Indeed, the users will be able to create a form and add comments to it. The form will be composed of a title and a comment section. The comments in the form of a tree meaning that a comment can have a child comment. This will allow the users to respond to a specific comment.

Is worth saying that normal users (ie drivers and managers) won't be able to delete a form or a comment. Only the administrator will be able to do that.

3 Developed system and its functionalities

3.1 Login and register system

When a user launches the application, they will be directed to the login page where they can enter their login credentials. If they do not have an account yet, they can create one by clicking on the register button.

The registration process will prompt the user to choose their role (either as a manager or a driver) and input their necessary details. After successfully creating their account, they will be redirected back to the login page where they can now enter their credentials and access the application.

This process aims to provide a smooth and convenient experience for all users, whether they are new to the application or returning users. With a streamlined login and registration process, users can quickly and easily access the application's features.

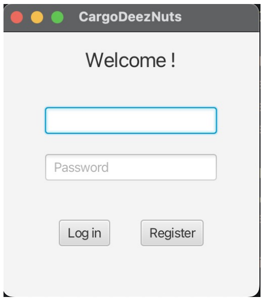


Figure 3: Login Page

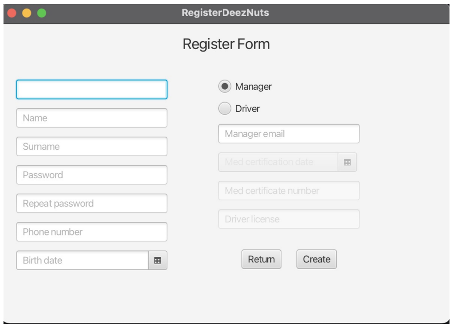


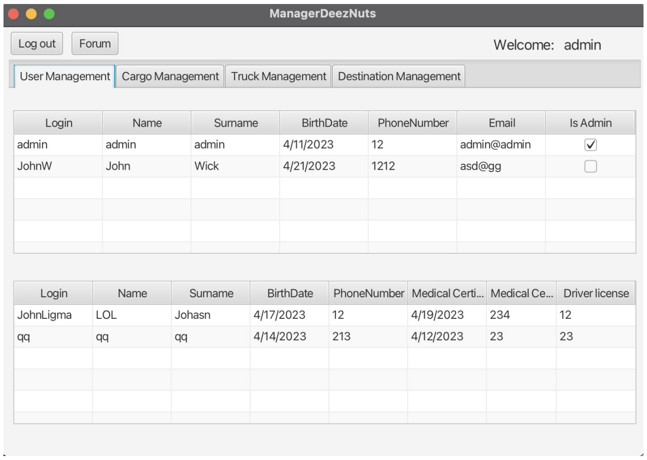
Figure 4: Register Page

3.2 Manager interface

When a manager logs in, they will be directed to the manager interface where they can access different tabs: User Management, Cargo Management, Truck Management and Destination Management.

3.2.1 User Management

The User Management tab allows the manager to view all users and their details. They can also add, modify and delete users only if the manager is an administrator.



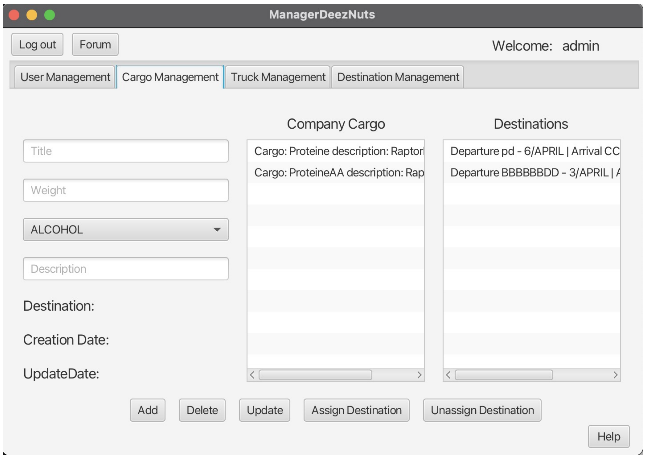
Login	Name	Surname	BirthDate	PhoneNumber	Email	Is Admin
admin	admin	admin	4/11/2023	12	admin@admin	☑
JohnW	John	Wick	4/21/2023	1212	asd@gg	□

Login	Name	Surname	BirthDate	PhoneNumber	Medical Certi...	Medical Ce...	Driver license
JohnLigma	LOL	Johasn	4/17/2023	12	4/19/2023	234	12
qq	qq	qq	4/14/2023	213	4/12/2023	23	23

Figure 5: Manager User Management

3.2.2 Cargo Management

The Cargo Management tab allows the manager to view all cargo and their details. They can also add, modify and delete them. A cargo has a title, a description, a weight, a type and an assigned destination. All this information can be modified by the manager and must be entered to create a cargo.



ALCOHOL

Cargo: Proteine description: Raptor
Cargo: ProteineAA description: Rap

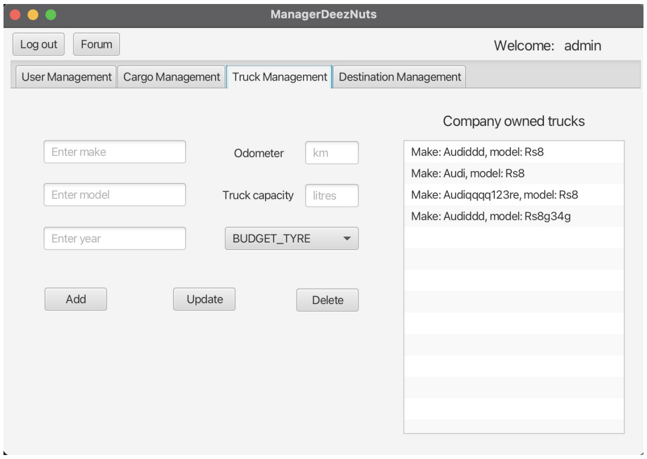
Departure pd - 6/APRIL | Arrival CC
Departure BBBBBBDD - 3/APRIL | A

Figure 6: Manager Cargo Management

3.2.3 Truck Management

This tab is quite similar to the Cargo Management tab. Indeed, the manager will be able to register a truck in the application by entering its information: the make, the model, the year, the odometer, the capacity and the tire type.

Plus, if while creating the truck, the manager made a mistake, he still can modify the truck by clicking on the truck and modifying the information and then clicking on the "Update" button.



BUDGET_TYRE

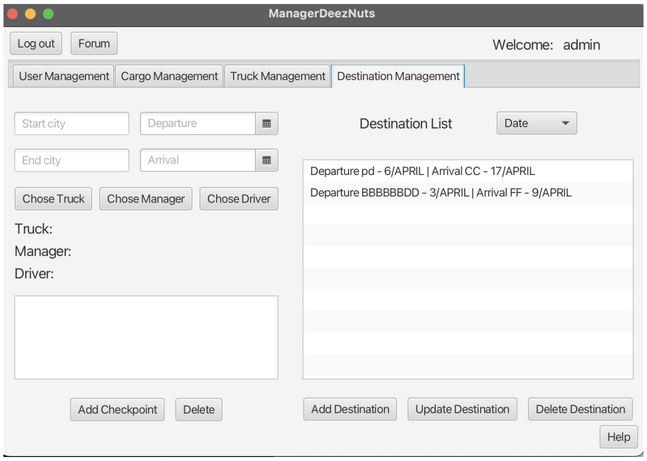
Make: Audiddd, model: Rs8
Make: Audi, model: Rs8
Make: Audiqqqq123re, model: Rs8
Make: Audiddd, model: Rs8g34g

Figure 7: Manager Truck Management

3.2.4 Destination Management

In this tab, there are different fields to fill in order to create a destination: The start and end location, the start and end date, the assigned truck, the assigned driver and the assigned manager. If while creating a destination, the manager doesn't want to directly assign a driver, he can leave the field empty and assign a driver later, leaving the destination free for any driver to take it.

There is another important field which is the checkpoint. Indeed, the manager can add a checkpoint to the destination. This will allow the driver to take a break (small or long) if he needs to. To do so, the manager will have to click on the "Add Checkpoint" button and then enter the checkpoint information: the location and the date of the checkpoint.



Welcome: admin

Truck:
Manager:
Driver:

Departure pd - 6/APRIL | Arrival CC - 17/APRIL
Departure BBBBBBDD - 3/APRIL | Arrival FF - 9/APRIL

Figure 8: Manager Destination Management

It goes without saying that it is possible to modify or delete a destination.

3.3 Driver interface

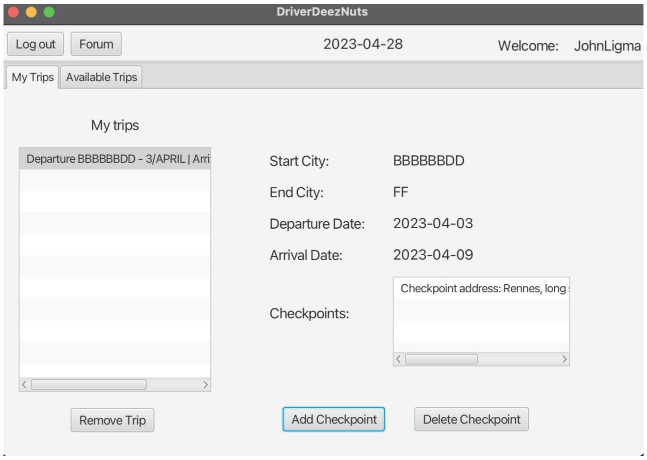
When a driver logs in, they will be directed to the driver interface where they can access 2 tabs: Assigned trip and available trips.

3.3.1 Assigned trip

In this tab, the driver will be able to see the destination that has been assigned to him. He will be able to add or delete a checkpoint if he needs to.

Plus, he will have all the information such as the start and end location, the start and end date and the assigned truck.

Finally, if the driver doesn't want to take the destination, he can click on the "Cancel Trip" button and the destination will be available for any driver to take it. Being lazy is not an option but sometimes, it's better to stay home and watch Netflix.



2023-04-28 Welcome: JohnLigma

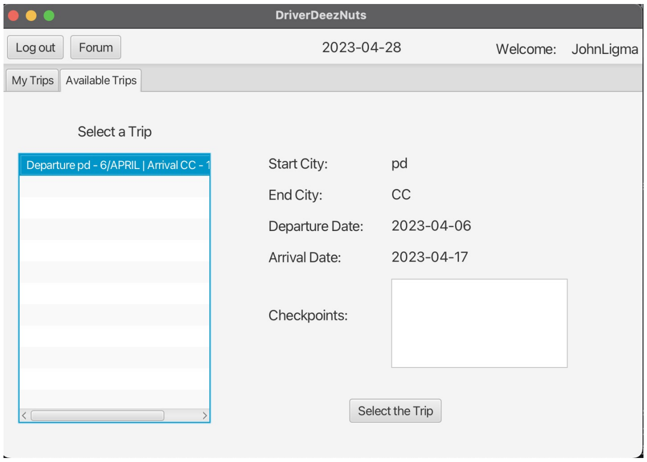
Departure BBBBBBDD - 3/APRIL | Arri

Start City: BBBBBBDD
End City: FF
Departure Date: 2023-04-03
Arrival Date: 2023-04-09
Checkpoints: Checkpoint address: Rennes, long

Figure 9: Driver Assigned Trip

3.3.2 Available trips

In this tab, the driver will be able to see all the available destinations. He will be able to take a destination by clicking on the "Select the Trip" button. Then if he goes back to the "My Trip" tab, he will see that the destination has been assigned to him.



2023-04-28 Welcome: JohnLigma

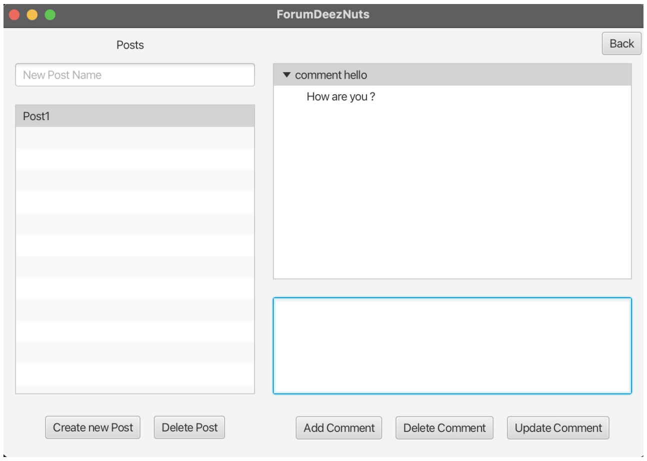
Departure pd - 6/APRIL | Arrival CC - 1

Start City: pd
End City: CC
Departure Date: 2023-04-06
Arrival Date: 2023-04-17
Checkpoints:

Figure 10: Driver Available Trips

3.4 Form

In this application, the users will be able to create a form and add comments to it. The form will be composed of a title and a comment section. The comments will be in the form of a tree meaning that a comment can have a child comment. This will allow the users to respond to a specific comment.



Post1

comment hello

How are you ?

Figure 11: Form

Is worth saying that normal users (ie drivers and managers) won't be able to delete a form

4 Summary of the project

This project provided a great opportunity for me to learn new technologies. While I had experience in programming languages such as *C, C#, Python, and OCaml* from my home school, I had not worked with Java before. Thus, it was very interesting to create a project in Java and use not only Java itself, but also JavaFX, Hibernate, and Spring Boot.

As is often the case when learning new things, I faced some challenges along the way. For instance, I struggled to understand how to use Hibernate and ensure that all CRUD operations were working properly. However, after conducting extensive research and numerous attempts, I eventually gained a clear understanding of Hibernate and successfully created the application.

On the other hand, working with JavaFX was relatively easy, largely due to the convenience of Scene Builder. This tool made it simple to create a GUI and link it to the code.

Using Spring Boot presented some initial difficulties, particularly in understanding why it was necessary and how to use it. With the help of lectures and research, however, I eventually grasped the concept and successfully created the API. Although I considered linking the API to my application, I ultimately opted to stick with the Hibernate controller due to the difficulty of doing so.

Overall, I am very pleased with the outcome of this project. I was able to create a functional and user-friendly application, while also gaining valuable experience with new technologies that will prove useful in my future work.

Thank you for reading this report and I hope you enjoyed it.