POLITÉCNICO DE LEIRIA | ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO

Departamento de Engenharia Informática
Morro do Lena - Alto do Vieiro
2411 - 901 Leiria - Portugal

ENGENHARIA INFORMÁTICA

Tópicos Avançados de Engenharia de Software

3º ano - 1º semestre

2022/2023

Project Assignment

FasTuga Driver

The **FasTuga** restaurant intends to further increase its business by providing extra value to its customers. Besides the current platform development to support its employees and customers (Appendix A - DAD Project Assignment), the restaurant has planned to offer a delivery service allowing customers to enjoy their meals at home, or at any desired place.

The project assignment is to develop the **FasTuga Driver** application dedicated to drivers that will deliver the orders to customers. This companion application will allow drivers to engage in the business, and to pickup and deliver the orders, while earning money.

The application should interact with a centralized support system that will hold details about the orders, customers and drivers. For development purposes, the Firebase[1] platform or any other platform that provides an effective support to the application's functional needs can be used (mandatory to read Appendix A - DAD Project Assignment).

When starting the **FasTuga Driver** application, the driver will have access to a dashboard where she can see the list of available orders and a dedicated area of the orders that are assigned to her. She will also be able to check the balance and have access to all of the major features of the app.

precisa autenticação

The driver will be able to self-assign herself to a specific order. From the list of available orders in the dashboard, the driver might choose an order to deliver it to the customer. Once the restaurant notifies the driver that the order is ready for pickup, the driver can claim the order and initiate the delivery process. Once the driver reaches her destination and completes the handover of the order to the client, she finishes the process. If, by any chance, the driver has any difficulties with the delivery, she should inform, through the application, that she was unable to deliver the order.

The driver should be able to select an order from a list based on the criteria of the nearest or farthest location - the starting point is fixed and is always the restaurant. When delivering

[1] https://firebase.google.com/

the order, the driver should be able to get the directions to the destination[2] to optimize total time of delivery.

Drivers earn fixed fees: 2€ per routes up until 3 km; 3€ per routes up until 10 km; and, 4€ for routes of greater length. Once the delivery process is finished with success, a driver of the **FasTuga Driver** will be able to see her updated account balance.

The **FasTuga Driver** app should also provide a statistics feature. For instance, allowing the driver to check the number of deliveries conducted, the average and total time of the deliveries, the total amount earned, the number of distinct customers served, … (among others).

The **FasTuga Driver** user has to register to use the application. Upon conducting a valid login in the application, all the features will be available and, at any time, the user can logout from the application or even opt out from the application.

The **FasTuga Driver** should be simple and easy to use.

Please note that:

- *any issue or question regarding the context presented should be clarified with the professors;*
- *any extra features that your team believes will increase the value of the app should be proposed and agreed upon with the professors. Only after the successful deployment of the above mentioned features will any extra features be considered;*
- *it is absolutely mandatory an accurate reading of the appendix - another project assignment, even if you are not engaged in the other course.*

Work Completion and Delivery

The work must be: developed following the software development process defined and addressed in laboratory classes; developed as a native mobile application OR web application (designed with a “mobile-first design” approach). The technological selection is defined by each team and must include all the features requested regardless of the technological solution adopted (100% working software).

This course is based on the execution of a software engineering process, so the work described here will be assessed according to two main perspectives: (1) the development process and (2) the final product.

One of the major elements in the software development process that will be used involves teamwork and therefore the project will be carried out by a team of 4 students from the same laboratory class, thus allowing that the team engagement and the practice of pair-programming is ensured throughout the process.

[2] https://www.openstreetmap.org/

The choice of features to deliver in each sprint is the entire responsibility of the team. At the beginning of each development cycle, the team will plan and commit to what they think they will be able to demonstrate to the customer in the demo.

At the deadline for the delivery of the project, the team will have the final opportunity, if it has not already done it, to see the functionalities accepted, or not, by the customer.

It will also be on the delivery date that the software development process and the adequacy of the team's work to it, will be assessed.

The development process of the application implemented in this project should be the object of a written report presented from the perspective of the software engineering used in its development. The report must be scientifically sound. Consider using references from the theoretical classes or other reliable references from the bibliography. It will be a description of the features of the application, of the entire process that was followed in the development, which architecture was designed, and the tests that were carried out.

The report must follow the structure of a scientific paper and should not exceed 6 pages (template available in moodle). It must include the sections: System Description, Development Methodology, Architecture and, Validation and Verification (Tests), in addition to the standard sections already available in the template (introduction, references,…). Optionally, you may include a section on Software Design Patterns, if applicable. A PDF document is expected and all the PDFs submitted will be, later, made available for everyone enrolled in the course.

Assessment

Grade = (60%*Process + 20%*Product + 20%*Report) * Public Presentation Session

[60%] Process

- [30%] Process planning and execution (commitment of features to be delivered versus features delivered and accepted)

- [30%] Adequate project control (adaptation to the development process covered in lab classes - including, among others, the proper use of the: version control and management system: Git, and the planning support system: JIRA)

[20%] Product

- [20%] Implemented features and corresponding automated tests

[20%] Written Report

- [20%] methodology + [20%] architecture + [20%] tests + [20%] mandatory sections + [20%] system description