Cairo University
Faculty of Computers and Artificial Intelligence

Advanced Software Engineering

Project Description

2021/2022

Version 1.0

Project Team

Staff: Dr SohaMakady s.makady@fci-cu.edu.eg

TAs: Eng Hassan Mourad h.morad@fci-cu.edu.eg
Eng Mohamed Samir m.samir@fci-cu.edu.eg
Eng. Esraa Abbas e.salem@fci-cu.edu.eg
Eng. Ahmed Samir a.samir@fci-cu.edu.eg
Eng. Ahmad Hassan h.mahmoud@fci-cu.edu.eg

Project Description – 2021/2022

Introduction

- In this project you will design and implement a non-trivial software system. You will practice the concepts you learned during the course.
- Project will be based on agile practices with at least 3 phases
- In each phase we will focus on designing and implementing some requirements.
- In each phase you are required to deliver the following **through blackboard as one zip file**, named with your student IDs as follows: **CS352_Sprint1_ID1_ID2_ID3_ID4**. Following such naming is a MUST. The deliverables are:
    - Proposed class diagram
    - sequence diagrams for the **most complex** scenarios. The submitted sequence diagrams should be **2 x the size of the team**, where each team member would be responsible **for submitting two sequence diagrams**.
    - Check the SDS document with the project description
    - Sprint document
        - Meeting minutes for the sprint starting meeting.
        - Meeting minutes for the sprint standup meetings.
        - Meeting minutes for the spring retrospective meeting.
        - Trello board screenshot
    - Git repository for the developed source code.
    - Zipped copy of the source code
- Check the Sprint document with the project description
- Your project customer (whom you can check requirements with) and coach is your TA.
- For more information about the different sprint terms mentioned above, and the overall agile software development process, please refer to this link https://www.mountaingoatsoftware.com/agile/scrum/resources/overview

Project Logistics

1 Students from the same lab/TA will be divided into groups; each group consists of 3-4 members.
2 If a team is formed out of more than 4 members, their submission will be rejected, and they would get a zero for the sprint.
3 Your team will register their names with the TA and **you CANNOT change teams** after registration.
4 Academic honesty is assumed. All work submitted must be original and written by your team (Not copied from students, the net, outside sources). Plagiarism will be penalized.

- Soon, you will be our colleague and we will be proud of you.
- Professional conduct and practice is essential in your career.

Project Phases:

Phase	Deliverables	Deadline	Mark
Sprint 1 submission on blackboard	**Design and implement Sprint 1 user stories (mentioned below)** Submit all the required deliverables (mentioned above)	Nov. 23, 2021 Late submission is not allowed	

Project overview

In the current days the transportation technologies are growing rapidly. Therefore, in this project we are going to develop an application that helps users to communicate with car drivers to transport users to any area.

In this project you are required to develop the requested functions with respect to the SOLID principles.

Your design also should adhere the OOP concepts. So the basic unit in your class should be the "class".

You should think about an efficient design that will be suitable if the requirements are extended. Also you should think about a portable design to be used in any other user cases.

You are free to choose any programming language that you want. However, The design concepts in the labs will be explained with Java.

The current functions will be exposed as a normal member functions part of a class. In the next sprint we are going to expose these functions as web services.

Sprint 1 requirements

1 The user should be able to register to the system. The user should provide username, mobile number, email (optional), and password. If the user is going to register to the system as a driver so the driving license and national id should be provided. The user should be able to login into the system once the registration is completed. If the user registers as a driver, so the user should be able to login into the system once the admin user verify the registration

2 The admin user should be able to verify driver registration. So the admin should be able to list all pending driver registrations and verify any pending driver registration.

3 The user should be able to request a ride given a source and a destination. For a simplicity user can enter the source area's name and the destination area's name.

4 The driver should be able to add some areas to get notification when any ride is requested and one of these areas is added as the source area. These areas will be called as "favorite areas".

5 The driver should be able to list all rides with source area within one of the driver's favorite areas. The driver should be able to suggest a price to this ride and notify the user with this price. Each price suggestions is called an "offer".

6 The admin should be able to suspend any driver/user account. By suspending any account the corresponding user shouldn't be able to login into the application.

7 The user should be able to rate any driver. The user should provide a star rating to the driver from 1 to 5 (1 worst, 5 Best)

8 The driver should be able to list user ratings. All user ratings should be visible to the driver.

9 The user should be able to check the average rating for the driver. The driver's information should include the average user rating.

10 Data from the above requirements should be persisted (i.e., saved in some location and retrieved as needed). There are many "ways" to ***persist*** these, you can persist them in database or you can persist them in memory (arrays, lists ,,, etc). Now you are required to develop at least one of these options or both of them, however if you developed one of these options, or the two of them, or switched between the two, this should NOT affect the rest of the application, nor the way the rest of the application is interacting with this specific part.

Evaluation Criteria

1. Properly working functionality as per the sprint requirements.
2. Quality of project configuration (i.e. task management, version control, SDS documentation)
3. Consistency between the various submitted system models.
4. Consistency between the submitted system models, and the working product.

Policy Regarding Plagiarism:

Students have collective ownership and responsibility of their project. Any violation of academic honesty will have severe consequences and punishment for ALL team members.

١ تشجع الكلية على مناقشة الأفكار و تبادل المعلومات و مناقشات الطلاب حيث يعتبر هذا جوهريا لعملية تعليمية سليمة
٢ ساعد زملاءك على قدر ما تستطيع و حل لهم مشاكلهم فى الكود و لكن تبادل الحلول غير مقبول و يعتبر غشا.
٣ أى حل يتشابه مع أى حل آخر بدرجة تقطع بأنهما منقولان من نفس المصدر سيعتبر أن صاحبيهما قد قاما بالغش.
٤ قد توجد على النت برامج مشابهة لما نكتبه هنا أى نسخ من على النت يعتبر غشا يحاسب عليه صاحبه.
٥ إذا لم تكن متأكدا أن فعلا ما يعد غشا فلتسأل المعيد أو أستاذ المادة.
٦ فى حالة ثبوت الغش سيأخذ الطالب سالب درجة المسألة ، و فى حالة تكرار الغش سيرسب الطالب فى المقرر.