Project 1.0: Let the games begin!

Due date: 09/30/2019
Presentation: 09/30/2019~10/02/2019

1 Introduction

In this project, you are to develop a software system to play a specific board game[1], henceforth referred to as the *X game*. Besides allowing users to play the *X game*, there are some other features that the system should provide.

This is an incremental project that you will develop using agile practices. There will be three sprints along the semester. While the rules of the *X game* are immutable, the rest of the requirements are always prone to change.

Read carefully this document before proceeding!

2 The *X game* system

The *X game* system is a platform that allows users to play the *X game* online against other players. The next transcript provides more information about it. If you have any doubt, ask for clarification.

> *"What I need is something like a platform that allows users to play the X game online. Anyone could register to this platform, for example by using an email, which would be unique for that user. To register, the person should also provide a nickname (also unique, maybe public???) and a password.*
>
> *What can a user do in the platform? Mmmm. She could create a new match (so she can play it). Since she can't play by herself, she should be able to invite another user to join the match. Perhaps she could send more than one invitation, and then it would be something like "first come, first served", so the first user accepting the invitation will be the one joining the match??? Is that possible?? I guess a user also needs to be able to reject an invitation, so it would be nice if the user who sent it receives a notification anyway.*
>
> *It would be cool if a user could be part of multiple games at the same time, though maybe she would want to quit from any game at any time? I guess a user would also want to be able to unregister.*
>
> *The platform also needs to record the history of matches played by a user. Info like players, start and end dates and times, and end results would be useful, you know, to know who won or lost or if there was a withdraw. I guess info about abandoned games should be also recorded. All this info would be part of the user profile, which can only be viewed by registered users.*
>
> *The gameplay, well—the X game has a set of rules[2], which need to be followed during a match. Besides that, of course a game can't start until enough players have joined and I'm guessing that after a match starts no other player should be able to join. Who starts the match? If I'm not wrong, that should be specified in the rules of the game. If I am wrong… the user who created the match would be the one making the first move. Mmmm, the system should be able to determine whose turn is it... according to the rules, right? Meaning, a player can only make moves when it's her turn... allowed moves, that is... the rules.*

[1] Find out which board game was assigned to your team in Section 8.
[2] Section 9 provides a starting list of resources for the *X game*. The rules of the game itself will not change.

What else? Oh right. The state of the matches should be saved in some way, so the user can play whenever she wants. My guess is that users won't be playing the whole time, so for example, a user would make a move whenever is her turn and log out, and after a while she would come back and check if the other player made a move and it's her turn again. Asynchronous matches, I think that describes it. The system needs to know when a game is over and should let know the players who won or lost. All according to the rules."

3 Development practices

Although it is difficult to follow a strict agile methodology to develop this project, the idea is to apply as many agile practices as possible.

You are to work as a scrum team, where the TA and the instructor will act as product owners (POs). The scrum master will be selected by each team and will act in this role for the entire term. You will carry out all the scrum ceremonies for each sprint, i.e., the sprint planning, the "daily" scrums, the sprint review, and the sprint retrospective. If you need the PO to be present during a ceremony, the scrum master must make the necessary arrangements.

You are also required to use a Kanban board to keep track of your progress in terms of user stories (or tasks).

Unit testing is another mandatory practice. Other kinds of testing and continuous integration are optional but strongly encouraged.

4 Requirements engineering, software design and implementation

As a development team, you are to flesh out the requirements listed above as user stories. Remember that user stories include: a title, a priority, an estimate and acceptance criteria. Moreover, you are to break user stories into tasks.

You are to use CRC cards and class diagrams to sketch the design of your software system.

As soon as you have defined the sprint backlog and a design sketch, you can start addressing the sprint tasks. You are required to keep a traceability matrix between use story tasks and classes.

5 Development environment setup

There are only a few restrictions in the tools and technologies used to develop the project.

- The project must be stored in a GitHub repository. Name the repository after your team's name as follows: `cs414-f18-001-teamname`. The repository should be used to store and track changes to the source code and any other document generated during the development of the project (i.e., development artifacts, presentation slides, etc.).
- The project must be written in Java.
- Extra points will be given for using continuous integration and testing techniques other than unit testing.

6 Deliverables

There are seven deliverables for this assignment:

1. User stories and tasks. They should be uploaded to the GitHub repository. Find a good way of doing so, e.g., in a PDF or MD file.

2. **Kanban board**. A weekly screenshot of the Kanban board should be provided. Find a good way of doing so, e.g., in a PDF or MD file.
3. **Design artifacts**. This document includes the CRC cards and class diagrams. Find a good way of doing so, e.g., in a PDF or MD file.
4. **Source code**. The code should include the JUnit test cases. Well-documented code will be rewarded.
5. **Development manual**. The document should describe how to set up the development environment to work on the project, how to run the system as a developer, and how to run the tests (put yourself in the place of a newcomer—what are the necessary steps for her to start working on the project?).
6. **Traceability link matrix**. This is a spreadsheet document showing the traceability links between user story tasks and the code. While each row represents a task, each column represents an implemented class in the system. A mark in the cell intersecting the task *UC-1A* and the class *MyClassB* indicates that the class *MyClassB* is directly involved in the implementation of the user the task *UC-1A*, as shown next:

	MyClassA	MyClassB	…	MyClassZ
US-1A	X	X		
US-2B		X		X
…				
US-nY	X			X

7. **Output of scrum ceremonies**. Make sure you take notes during the sprint review and the sprint retrospective. Find a good way of sharing your notes, e.g., in a PDF or MD file.
8. **Presentation**. The progress on the project is to be presented during class, on **09/30/2019**. In addition to your progress in the project, you are to present a brief description of the *X game*, as well as any process/product decision you have made. You are also to present the results of your scrum ceremonies.
9. **Peer evaluation** (individual). Each team member is required to fill out the peer evaluation and submit it through Canvas (***do not store it in the team's repository***).

The grades for this assignment will apply to the deliverables uploaded to the GitHub repository before class (i.e., 9:00am).

7 Notes

- Delivery dates associated with deliverables will be verified in the repository. Late work policies apply. Once the repository has been created, add the instructor[3] and TA[4] as collaborators.
- Instead of making assumptions about the requirements, talk with the product owner.
- Grading criteria:
  - Presentation: 30%
  - Rest of deliverables: 70%
- Points will be deducted if:
  - The submission requirements are not met.
  - The instructor and TA are not added to the GitHub repository.
  - You are late with the submission.
- You will not receive credit for this assignment if:

[3] GitHub account: lmorenoc
[4] GitHub account: salonich

- You do not submit the deliverables.
- You do not present during class.

8 Board game assignment

Team	Board game
970 Dream Team	Hnefatafl
Blueberries!	Rollerball
Byte Mechanics	Congo
Hackers Anonymous	Banqi
Java the Hutt	Congo
NN	Hnefatafl
Runtime Terror	Rollerball
Spaghetti-Coders	Jungle
Sparkans	Banqi
Team.name()	Jungle

9 Board game resources

9.1 Hnefatafl

- Hnefatafl – The strategic board game, Sten Helmfrid
  Available: http://hem.bredband.net/b512479/#Chap9
- Rules of the game Hnefatafl, Dragonheel's Lair
  Available: http://www.dragonheelslair.com/en/ruleshnef.php
- Fetlar Hnefatafl, Cyningstan
  Available: http://tafl.cyningstan.com/page/88/fetlar-hnefatafl

9.2 Banqi

- Banqi, Wikipedia
  Available: https://en.wikipedia.org/wiki/Banqi
- Banqi, Woody Thrower
  Available: https://woodpress.org/banqi/

9.3 Jungle

- Jungle, Wikipedia
  Available: https://en.wikipedia.org/wiki/Jungle_(board_game)
- Shou Dou Qi – The battle of the animals, Chess variants
  Available: http://www.chessvariants.com/other.dir/animal.html
- How to play Dow Shou Qi "The jungle game," Ancient Chess
  Available: http://ancientchess.com/page/play-doushouqi.htm

- Dow Shou Qi ("Game of fighting animals), Jonathan K. Vis
  Available: http://liacs.leidenuniv.nl/~visjk/doushouqi/about.html

9.4 Rollerball

- Rollerball, Wikipedia
  Available: https://en.wikipedia.org/wiki/Jungle_(board_game)
- Rollerball, A world of chess
  Available: http://history.chess.free.fr/rollerball.htm
- Rollerball, Chess variants
  Available: http://www.chessvariants.com/40.dir/rollerball/index.html

9.5 Congo

- Congo, Wikipedia
  Available: https://en.wikipedia.org/wiki/Congo_(chess_variant)
- Congo, MindSports
  Available: http://www.mindsports.nl/index.php/side-dishes/interesting-games?start=2
- Congo, Chess variants
  Available: https://www.chessvariants.com/ms.dir/congo.html