Assignment #3

June 16, 2018

This assignment is the last assignment of SENG 300 (Spring 2018) and counts for 5% of your final grade. This assignment includes all of the deliverables you submitted for Assignment #1 and #2 plus **TWO** new deliverables: **Deliverable #8** and **Deliverable #9**. The document you submit for this assignment can be used for your professional portfolio to showcase your work and to demonstrate your software engineering skills.

A. Requirements and Use Case Modelling

For this portion of the assignment, your deliverable will consist of three things:

1. A software process model that you used for developing your product
2. A set of functional and non-functional requirements of your system
3. Use cases and a use case diagram **OR** user stories and a story map

need to adjust this a bit

Deliverable #1:
(1-1): For this portion of the assignment you are to choose one of the following software process models:

- Opportunistic
- Waterfall
- Spiral

- Concurrent
- Scrum

Justify why you chose this methodology for developing your system over the others in 1 to 3 paragraphs. *A successful submission will include a thoughtful comparison of the benefits of the selected method over the other methods.* Details about planning and implementing this process model are **NOT** required for this assignment. If you received full marks for this deliverable in Assignment 1 and your process model has remained the same, you can use your original submission. Otherwise, please update your original deliverable.

(1-2): Submit five well-defined functional and non-functional requirements (5 FRs and 5 NFRs). *A successful submission will include specific and understandable descriptions of each requirement.* If you received full marks for this deliverable in Assignment 1 and your requirements have remained the same, you can use your original submission. Otherwise, please update your original deliverable.

Deliverable #2:
(2-1): To represent your system's requirements, you need to choose **one** of the following paths:

- Use cases and a use case diagram
- User stories and a story map

Justify why you chose one method of defining what your system has to do over the other in 1 to 2 paragraphs. *A successful submission will include a thoughtful comparison of the benefits of the selected method over the other method.*

(2-2): Next, you are to provide three well-defined *use cases* or *user stories*. *A successful submission will include specific and understandable descriptions of what the user will be able to do following the format discussed in class for the chosen method.*

(2-3): The use case diagram or story map describing Deliverable 2-2. *A successful submission will be syntactically-correct and legible.*
If you received full marks for this deliverable in Assignment 1 and if your requirement has not been changed, you can use your original submission. Otherwise, please update your original deliverable.

B. Low - Fidelity Prototyping

Deliverable #3
For this portion of the assignment, your deliverable will consist of three things:

1. Sketches of your system's interface
2. A storyboard of your system's behaviour

3. A Wizard of Oz prototype of your system

(3-1): Sketches
Create 5 overview sketches showing individual snapshots of your system's interface. Each sketch should represent a different idea for the interface. Choose one of the overview sketches and create 5 more detailed sketches elaborating on it. Explain why you chose the overview sketch that you did (*Hint*: this should be a considered design decision). If you submitted all these sketches in Assignment 1, you can use your original submission. Otherwise, please update your original deliverable. *A successful submission will include neat and readable hand-drawn sketches that are appropriately labeled and explained.*

(3-2): Storyboard
Create a storyboard showing a user performing a task with your system. If you submitted a complete storyboard in Assignment 1, you can use your original submission. Otherwise, please update your original deliverable. *A successful submission will include a sequence of neat and readable sketches that are annotated, show user behaviour as well as the system's response.*

(3-3): Wizard of Oz
Create a video showing a user interacting with your system. Although your functionality will not be complete at the point, the system responses can be simulated using another person. If you submitted a video in Assignment 1, you can use your original submission. Otherwise, please update your original deliverable. *A successful submission will include a short brief video showing one of the use cases of the system.*

need to adjust this a bit

C. Planning and Effort Estimation

Deliverable #4
For this portion of the assignment, your deliverable will consist of two things:

1. An estimation of effort using the Poker Effort Estimation method
2. Comparison of estimates

(4-1): Poker Effort Estimation: Perform a planning poker session with your teammates. Break down one of your functional requirements into a set of tasks, and provide a set of cards to each team member. Have each team member pick a card for each task, then reveal your cards and compare estimates. For each task, each team member must provide estimate in person-hours. Estimation in terms of ratings in any scale (e.g. 1 to 10) will not be accepted. Repeat this process with each task and tabulate the estimated effort for all tasks in terms of person-hours. For three of your tasks, describe the process how overall consensus was reached. In addition, describe the justification of the estimates given by each team member.

(4-2): For all the identified tasks in deliverable 4-1, compare the estimated effort using Poker Estimation approach with actual effort in terms of person-hours. You specifically need to comment on the accuracy of the estimates and the reason for discrepancy among the estimated and actual effort.

D. Software Testing

Deliverable #5

For this deliverable you are to implement one of your functional requirements using a test-driven development approach. This requirement should be the same as the requirement you selected for *Assignment#1* and it should be relatively complex and substantial. Your deliverable will contain:

1. Initial Tests
2. Tests with Partial Implementation
3. Demo of the Full Implementation

(5-1): Initial Tests
Before you start to work on your implementation, write at least five test cases that will be used to test your functional requirement. These tests should all fail initially. Submit screenshots showing your tests and the results obtained when running your tests. These screenshots should be accompanied by a brief explanation. If you received full marks for these tests in Assignment 2, you can use your original submission. Otherwise, please update your original deliverable.

(5-2): Tests with Partial Implementation
Begin to implement your functional requirement. Write enough code so that the test cases you created now pass. Stub out methods when necessary. Submit screenshots showing the state of your implementation and the results obtained when running your tests. These screenshots should be accompanied by a brief explanation. If you received full marks for these tests in Assignment 2, you can use your original submission. Otherwise, please update your original deliverable.

(5-3): Tests with Full Implementation
Complete your functional requirement. Submit screenshots showing the state of your implementation and a short video to demo the implemented requirement. These screenshots should be accompanied by a brief explanation. Please upload the video to Youtube and put its link in your document. If you received full marks this section in Assignment 2, you can use your original submission. Otherwise, please update your original deliverable.

E. Structural and Behavioural Modelling

Deliverable #6

For this portion of the assignment, your deliverable will consist of:

1. A class diagram
2. A detailed explanation of your diagram

Create one class diagram using standard tools (e.g. Microsoft Visio, StarUML) following the Unified Modelling Language (UML) notations. Please note that no hand written diagrams will be accepted. Try to model only the essential features of your system, and abstract away unnecessary details without oversimplifying. In addition to clarifying your diagram, your explanations should highlight any assumptions that you are making. If you received full marks for this deliverable in Assignment 2, you can use your original submission. Otherwise, please update your original deliverable (Your marks for Assignment #2 will be posted on D2L on Monday, June 19^{th}).

Deliverable #7

For this portion of the assignment, your deliverable will consist of:

1. Two sequence diagrams that describe success scenario of two important use cases of your system
2. Detailed explanations of these sequence diagrams

A successful submission will include sequence diagrams that have been created using standard tools (e.g. Microsoft Visio, StarUML) following the Unified Modelling Language (UML) notations. Please note that no hand written diagrams will be accepted. Your detailed explanation should include a discussion of why you chose the two use cases that you described with the sequence diagrams. If you received full marks for this deliverable in Assignment 2, you can use your original submission. Otherwise, please update your original deliverable.

F. Software Design Pattern

Deliverable #8

For this deliverable you have to pick one design pattern and apply it in your project. Your deliverable will contain:

1. Class diagram of the selected design pattern (using standard tools such as Microsoft Visio, StarUML) in terms of your own implementation
2. Source code of the files that implemented that pattern (as a zip file)

3. Describe (in natural language) the design pattern in terms of your own implementation. You have to clearly mention which files were used in implementing the pattern and the role of each class in the implemented pattern. In addition, you have to justify why you selected this pattern and how this pattern is going to help your project in terms of ease of maintenance (or ease of adaptation with future anticipated changes) of your project, reducing coupling and increasing coherence among classes.

You can select one pattern from the following patterns that were covered in your lectures and tutorials:

1. Facade
2. Strategy
3. Adapter
4. Singleton
5. Observer

G. Implementation

Deliverable #9
For this deliverable you will create a short video (at least 30 seconds long) to demo the functional requirements that you implemented in this course (and possibly the high fidelity prototype of the other features of your App [optional]). Please upload the video to Youtube and put its link in your document. Also create a powerpoint presentation with two slides, with a link to the video included in one of the slides (**A template for these two slides and the details of the presentation has been posted on D2L. All of the teams need to use this template for making their presentation**). This presentation will be shown during the celebration session on June 19^{th}, from 3:00 to 4:00 PM, EEEL 161.

H. Collaboration

Deliverable #10: Details of your collaboration (e.g. meeting, email, stand-up meetings, etc) with your customer. *A successful submission will include the details of your collaboration, such as:*

- Screenshot(s) of your email(s)
- A photo of one of your meetings as well as a brief description about the topics discussed and decisions made during this meeting *(please mention "Who were involved in this meeting?")*

How this collaboration helped you to do this assignment! And, what do you see as the greatest challenge in terms of communicating with your customers?

How to submit?

This assignment is to be submitted in **TWO** phases. Please submit your deliverable #9 (a presentation file, which includes a link to a short video to demo the functional requirements that you implemented) by 11:59 **AM** on June 18^{th} via D2L. This portion of the assignment counts for 20% of assignment #3 grade.
For the second phase, submit the remaining deliverables via D2L as a ***.zip*** file by 11:59 PM on June 23^{rd}. Make sure to put the names of all group members on the first page of your assignment. Only submit **ONE** copy per group!