PHPM 631 Assignment 4: Software Requirements
Due date: Submit on E-Campus by 11:59pm Sunday 2/24 & 3/3

Submission. Submit on E-Campus. See Requirements Section Below for details

- Week 1 (2/24): Report Part 1
- Week 2 (3/3): Report Part 2

Late Assignments: Each student will be allowed one late assignment, due 7 days from the due date. NO other late assignments or make up will be accepted.

Guideline for assignment grading (4%)

- 70% (70 points): Bad (√ --) Did NOT follow all instructions
- 80% (80 points): Reasonable (√ -) Followed all instructions
- 90% (90 points): Good (√) Followed all instructions, and did good work
- 100% (100 points): Great (√ +) Followed all instructions and did great work

Objective

By the end of this assignment, you should be able to

- communicate effectively with engineers about your software needs

Team Project

You are encouraged to work in a team of 3 for this assignment. If you do, all of you will receive the same grade for this part of the assignment. Submit one assignment for the whole team. BUT at the very top, clearly label that you worked as a team and the name of the team members. You may choose to work alone, but you will be expected to do the same amount of work as a team.

Privacy note: I may share good submissions in class.

Required Reading for this Assignment

Turner, A. M., Reeder, B., & Ramey, J. Scenarios, personas and user stories: User-centered evidence-based design representations of communicable disease investigations. Journal of Biomedical Informatics, 46(4), 575-584. doi: 10.1016/j.jbi.2013.04.006

Assignment: Software Requirements

Requirements errors are the most expensive to fix during production.

In IT management, an important skill is to be able to communicate with engineers about your software needs. Your ability in figuring out your hospital's software needs and specifying it to the engineers will directly impact the success of the software projects.

In this assignment, you will pick a software need and communicate it to engineers by writing a form of software requirements used in the agile method, called user stories.

"In software development and product management, a **user story** is a description consisting of one or more sentences in the everyday or business language of the end user or user of a system that captures what a user does or needs to do as part of his or her job function. User stories are used with agile software development methodologies as the basis for defining the functions a business system must provide, and to facilitate requirements management. It captures the 'who', 'what' and 'why' of a requirement in a simple, concise way, often limited in detail by what can be hand-written on a small paper notecard." (Wikipedia)
http://en.wikipedia.org/wiki/User_story
http://guide.agilealliance.org/guide/user-stories.html
http://wiki.directproject.org/User_Stories

Recommended Action Plan

1. [Build a Team] Form a group of 3 to work together.
2. [Pick a Software Need] Select/define a potential software need in a health care provider setting. It can be an independent software or a customization of an existing software system, such as the EMR. For example, developing a customized component in the EMR or developing the EMR component to reduce ED users by implementing the high ED user program.
3. [Read the required article] See required reading section above
4. [Submit a Progress Report by next week] See required submission section below for details
5. [Write the Concept] See required submission section below for details.
6. [Write the Personas] See required submission section below for details.
7. [Write the Scenarios] See required submission section below for details.
8. [Write the User Stories] See required submission section below for details.

Required Submissions

Write a requirements report for a specific software need at a health care provider. Note that the required article has an example of persona, scenario, and user story. I provide this as a grounding starting point. You may use them as a guideline to write yours. There is a wide range of what is acceptable. You may follow the examples in the article as closely as you like. That is, your submission format could be very similar to examples in the required article or depart quite a bit.

1. Report Part 1: Submit a word document. It needs to include the following information:
   a. Team members
   b. A short informal description of the software need you will work on for the assignment. A short paragraph will suffice.
   c. Concept (close to final draft): This is the fully developed software need description
      The concept document is the layman's description of the project. It should capture the motivation of the project and act as an anchor point if the team or project starts drifting. The project team should be able to turn to this document any time during the development process to determine if the project has drifted off-course.
      - Format - a one-page summary: The single page summary may include pictures or screenshots if appropriate. Include an architecture statement that is clearly labeled.

2. Report Part 2: Submit a word document. It needs to include the following information
   a. Concept Final version. You need two versions of the document:
      - a one-page summary: ok to be identical to report part 1, if no updates are needed.
      - a tweet: The tweet must be under 140 characters. You should not be actually tweeting this (unless you want to). Just writing into a word document what you might tweet.

   b. Personas: at least 3 personas- one per team member. Please list team member who wrote each. Personas are fictitious characters created to represent the different user types within a targeted demographic that might use your software. The description should capture behavior patterns, goals, skills, attitudes, and environment, with a few fictional personal details to make the persona a realistic character. There should be at least one persona that can stand in for each type of user in your system. If there are radically different skills among that group, you will have more than one persona. For example, the one-time user and the repeat user would probably be separate personas.

   c. Scenarios: at least 3 scenarios-one per team member. Please list team member who wrote each. Scenarios are stories. They are stories about people and their activities. They can be a composite of activities. Scenarios highlight goals suggested by the appearance and behavior of the system, what people try to do with the system, what procedures are adopted, not adopted, carried out successfully or erroneously, and what interpretations people make of what happens to them. Scenarios include or presuppose a setting. Scenarios also include agents or actors. Each agent or actor typically has goals or objectives. These are changes that the agent wishes to achieve in the circumstances of the setting. Every scenario involves at least one agent and at least one goal. When more than one agent or goal is involved, they may be differentially prominent in the scenario. Often one goal is the defining goal of a scenario, the answer to the question “why did this story happen?” Similarly, one agent might be the principal actor, the answer to the question “who is this story about?” Scenarios have a plot; they include sequences of actions and events, things that actors do, things that happen to them, changes in the circumstances of the setting and so forth. Particular actions and events can facilitate, obstruct, or be irrelevant to given goals.

d. User stories: at least two per team member. Please list team member who wrote each.
“It captures the 'who', 'what' and 'why' of a requirement in a simple, concise way, often limited in detail by what can be hand-written on a small paper notecard” (Wikipedia)

User stories are 2-3 sentence descriptions of the tasks that different users will want to accomplish. They should be short enough to fit on an index card and should be written from the perspective of a user. The key aspect of the user story is that it should be the way that the client thinks of the problem.

There are two styles of user stories: a first-person style and third person. The difference comes at least partially from the fact that the client may or may not be a user. If a user, he/she may give the description in first person; if not, he/she is more likely to use third person. Following is a simple example in both styles:

First person example: As an employee submitting a personnel change form, I want to be able to save the changes that I make (in all the previous user stories) without submitting them so that I can confirm my changes with my spouse.

Third person example: Ernie Employee has just gotten married and needs to change the beneficiary, tax status, and payroll deductions that the company has for him. After completing the arduous process of going through each of the sections of the HR website, he is nervous that he may have not gotten everything the way that he and his wife agreed. He does not want to re-enter the information, but would like to confirm the changes and submit them tomorrow.