CS6680 Advanced Software Engineering

Project 1: Requirements Analysis & Specification

"The hardest single part of building a software system is deciding precisely what to build. No other part of the conceptual work is as difficult as establishing the detailed technical requirements, including all the interfaces to people, to machines, and to other software systems. No part of the work so cripples the resulting system if done wrong. No other part is more difficult to rectify later." [Brooks, 1987]

1. Summary
   For this project, each team will choose one of the project topics listed in section 2 (but not limited to) and generate a document for "Requirements Analysis & Specification". Each team will also come up with its own Company Name and include it in the document.

2. Project Topics
   2.1. Ambulance Dispatch System
   Calling 911 and asking for the ambulance service would connect the caller to a dispatcher (also called dispatch controller) who feeds the information s/he receives from the caller into the system. The system would allocate and mobilize a suitable ambulance within 3 minutes, transmit details to the selected vehicle, and track and monitor actual performance and position. An exception message shall be generated if no free ambulance is available for at least 11 minutes. The system would show the location of each patient and the nearest three ambulances.

   2.2. Home Automation System
   People make the same mistake very often; they easily forget to turn off the light or stove when they are in a hurry to leave home. If we have home automation system, they can control the lights, stove, AC, garage door and etc. by using smart phones or computers outside of the house. To make this possible, we want to build home automation system with all software components. The system shall be capable of monitoring and controlling all (or most of) the electrical components at home. And if error (or abnormal condition) is detected from any devices (or parts), the system shall notify users about it.

   2.3. Restaurant Automation System
   Restaurant owner wants to make the ordering process automated. Once the customer gives an order at the counter, the employee enters the order on the screen (touch screen). Then the cook sees the order on the screen in the kitchen and starts making the food. Once the food is ready, the cook pushes the button and the server checks his/her own screen to know that the order is ready for which table. Finally, the server delivers the plates to that table.

2.4. Inventory System for Car Parts

Most of the time, it is hard to maintain correct inventories in automotive (parts) business without a support of computer system. To make it easy, the inventory system is to be built for automotive parts and the system shall maintain all status of the inventory. Inventory manager and general workers are potential users of the system. Once the parts arrive from the factory, the barcode of the box should be scanned so that the system can update the number of parts in the database. When the parts leave the storage, the inventory system should be also updated properly.

2.5. You can choose any topic that is not listed in this document.

3. The Deliverable

Your description should be elegant, comprehensive and of professional quality. Each team should submit one copy of the document to Troy Blackboard system.

3.1. Communication and Coordination

For each meeting, the team leader should send at least one e-mail message, before the meeting takes place, to all the team members and also cc it to the course instructor (Dr. Chang: hjchang@troy.edu). The message should describe the time and location of the meeting, as well as an agenda for the meeting. The team leader should also send one e-mail message, after the meeting, summarizing what has been discussed in the meeting.

3.2. Methodology for your team work

Describe how your team members were divided up, how many meetings were held, who were in the meetings (especially if outsiders were involved), and what were discussed in the meetings.

3.3. Issues

Discuss any issues your team have encountered and any alternatives your team have considered.

3.4. Requirements Specification

Present a requirements specification using your own favorite notation (e.g., English and/or UML) and using your own favorite standard (e.g., IEEE/ANSI 830-1993 (please see appendix below).

Appendix (The structure recommended in IEEE/ANSI 830-1998)

1. Introduction
   - 1.1 Purpose of the requirements document
   - 1.2 Scope of the product
   - 1.3 Definitions, acronyms and abbreviations
   - 1.4 References
   - 1.5 Overview of the remainder of the document

2. General description
   - 2.1 Product perspective
   - 2.2 Product functions

2.3 User characteristics
2.4 General Constraints
2.5 Assumptions and dependencies

3. Specific requirements

This covers functional, non-functional and interface requirements. These should document external interfaces, functionality, performance requirements, logical database requirements, design constraints, system attributes and quality characteristics.