Slides 1-3

1. Title (Introduce group members, Aadil, Don, Dustin, John, and Ketra), we are ABoT, also known as Always Be on Time, and we developed the Super Hospital System.
2. Topic Description
   1. For our project, we chose to design software for use in major hospitals. We envisioned it as being a system that provided all the functionality hospitals need for keeping track of patient information, medications, hospital pharmacy inventory, paging doctors and nurses, adding and removing patients and staff from the hospital’s database, and so on.
   2. This excerpt from our problem statement shows the Nurse paragraph. This paragraph describes the functionality that will be available to Users with Nurse permission. These will include allowing the Nurse to view only patient data required for providing care to the patient (name, room number, medications, condition, diagnosis, etc.). It will alert the Nurse if the next scheduled medication is scheduled too soon in order to ensure a patient does not overdose on the medication. It will also alert the Nurse if a patient’s condition (given by hospital machines that monitor pulse and heart rate), and allow nurses to page any of the Doctors on duty.
3. WSD

Slides 4-5

1. Rationale
   1. There is currently no hospital software on the market that keeps a database of the various medicinal and recreational drugs that hospital staff often use and encounter. (reader note, everything looked up in a physical book)
   2. We felt that software with an extensive database of drugs and their uses would greatly enhance the care patients receive while making the hospital staff’s lives easier. (reader note, software tracks what patients are taking, if its compatible with new medications or recreational drugs already in their system, etc., and notifying hospital staff of conflicts or overdose)
   3. We also felt that any hospital software should also keep patient information at the staff’s fingertips, and provide that information in a clear and easily understood fashion. (reader note, uncluttered display, ease of retrieving and editing data)
2. Requirements Traceability Matrix
   1. Our RTM has 30 entries total. Each entry has its corresponding paragraph in problem statement, a description, a type (hardware or software), and its corresponding use case
   2. Paragraph 2 shall statements are general statements regarding access to the system.
   3. Paragraph 3 shall statements give the required functionality for Users with doctor permissions
   4. Paragraph 4 gives required functionality for Nurses
   5. Paragraph 5 gives required functionality for Admins
   6. Each of these different Users are shown different things by the software and access to information is restricted based on permission
   7. Doctors are the only ones who can view all patient data
   8. Nurses can only view patient data relevant to providing patient care. Can’t view things such as SSN and other sensitive information.
   9. Administrators can only view staff information, and are responsible for adding/removing staff from the system, changing/resetting staff passwords, and setting the staff schedules in the system.

Slides 6-8

1. Use Case 14
   1. This Use Case describes how a User is added to the database by an Administrator.
   2. This Use Case describes how a User is added to the database by an Administrator.
   3. In the administrator home screen, Admin\_View, the User can click the New User button.
   4. This will bring up a pop-up with the username, password, first name, last name, and permission fields.
   5. Once they are filled out, the Administrator can hit submit and a new User will be added to the database.
2. Use Case Diagram 14
   1. Shows how an Administrator can add a User to the database.
   2. The information entered in the fields is held in the pop-up until the User either hits Submit or Cancel.
   3. Upon hitting submit, the information in the fields is added to a new User in the database and the pop-up will be destroyed.
   4. Upon hitting cancel, the fields will be cleared and the pop-up will be destroyed.
3. Category Interaction Diagram
   1. Login information is sent to the database, which checks the permission of the User by finding the matching Username and Password, and then showing the homescreen which matches the corresponding permission.

Slides 9-10

1. Function Point Cost Analysis
   1. Used average level complexity, which gave us 318.
   2. Reason for the 2 external interfaces is because we originally wanted to have the software be able to read from hospital machines such as heart rate monitors.
   3. Complexity adjustment factor of 53.17
   4. The 274 in the formula was the new value after we did it again with a simple complexity.
   5. Total FP 323.32, and total 1142.86 dollars per FP assuming 7 FP per month and $8000 labor per month
   6. 1142.86 x 323.32 = $369,508.57
2. COCOMO
   1. See slide

Slides 11-13

1. Dictionary
   1. Quick look at some of the terms and jargon we used in our project
   2. Give quick overview of terms
2. Gantt Chart
   1. This snapshot of our Gantt Chart shows our first 3 documents, and how we allocated our time and personnel to complete them.
   2. Black bar denotes the time period from start to completion of that part of the project
3. Project Legacy

GOOD

* 1. Project has almost all of the functionality we originally wanted in the text based format, and the GUI looks exactly as shown in the horizontal prototype.

BAD

* 1. Communication problems plagued the development phase to the point where we essentially had two groups, each designing a different piece of software to do the same thing.

FOR FUTURE RELEASES

* 1. GUI will be integrated with the text-based version of the software to provide a user-friendly program.
  2. Staff will have their schedules encased in each instance of the User object and they will be viewable by other Users.
  3. Software will be capable of reading information from hospital equipment to allow nurses and doctors to view a patient’s pulse, breathing rate, and other vital medical information from their terminal.