**Exam II**

**CS 460 Spring 2012**

**Questions 1-4**

**1. Make a 1-page overview table which lists**

**a. Your project roles/titles**

**b. Tools used**

**c. Tasks assigned**

**d. Tasks completed**

****

****

**===**

**2. Make a table of tasks completed and describe the following for each task in table format (see Gantt Chart for details):**

**a. Task name**

**b. Task description**

**c. Task start and end dates**

|  |  |  |
| --- | --- | --- |
| **Tasks Completed** | **Description** | **Start/End** |
| **View/Print Bingo Use Case Specification** | **A use case specification was created for the option of viewing a printing the bingo sheet for a student** | **Start:****1/25/12****End:****2/8/12** |
| **SWEBOK Reviews** | **Reviews for each of the 11 knowledge areas of the Software Engineering Body of Knowledge** | **Start:****11/30/11****End:****12/15/11** |
| **PMBOK Reviews** | **Reviews for each of the 9 knowledge areas in the Project Management Body of Knowledge** | **Start:****1/31/12****End:****4/10/12** |
| **Extract course info from iCal file** | **Created a servlet that reads and extracts specified information from a given iCal file** | **Start:****2/8/12****End:****4/9/12** |
| **Populate schedule object with extracted info** | **Created a Schedule class that processes the extracted information and places it into an ArrayList of course information and prepares it for display** | **Start:****2/8/12****End:****4/9/12** |
| **Display contents of schedule object on jsp** | **Created a jsp that displays the information in the ArrayList using jstl tags to iterate over the information** | **Start:****2/8/12****End:****4/9/12** |
| **Read and understand git workflow** | **Read the instructions and information on branching workflow on the git repository** | **Start:****4/14/12****End:****4/14/12** |
| **Create personal branch for feature development** | **Created a branch on the git repository for development of features** | **Start:****4/14/12****End:****4/14/12** |

**===**

**3. You completed 11 SWEBOK reviews and 9 PMBOK reviews by now that outline software engineering and project management body of knowledge. Identify at least three Knowledge Areas (KA) from SWEBOK or PMBOK that pertain to your team roles this semester.**

**a. Describe the KA with figure [1 page] (You may use your SWEBOK/PMBOK reviews here)**

**b. Describe your roles and their relationship to the area [1 page]**

**c. Provide diagrams/screenshots of any work output related to your area [1-3 pages] with captions.**

**d. Describe how you produced the work output in 3c [1-3 pages] (1 page per diagram)**

**SWEBOK KA 2**

The second SWEBOK Knowledge Area is Software Design. This area focuses on the key tasks involved in designing any software system. It contains the areas of Software Design Fundamentals, Key Issues in Software Design, Software Structure and Architecture, Software Design Quality Analysis and Evaluation, Software Design Notation, and Software Design Strategies and Methods.

The software design fundamental area contains all of the fundamentals of designing a piece of software. The key issues in software design area contain information about critical issues to keep in mind during software design. The software structure and architecture area outlines different styles for designing the software. The software design, quality analysis, and evaluation area defines attributes or quality software, analysis techniques, and a standard on how each attribute should stack up to form quality software. The software design notation area contains information about static and dynamic views of a software design. The software design strategies and methods outline methods and strategies that can be used to design software.

My roles as programmer, web developer, and software tester relate to this knowledge are because each requires an understanding of the overall software design in order to work effectively. A programmer must understand the overall design in order to write the code for the software to the set specifications. A web developer must understand the overall design in order to understand how the software will work in an online environment. A software tester must understand the overall design of the software in order to know how it should function and expedite the process of locating bugs and glitches. In addition to understanding the overall design of the software, each of my roles must understand how to design each part relating to the role.



Figure : Revision History for the View/Print Bingo Use Case Specification

I produced this document by downloading the template provided from a sponsor provided website, developing basic and alternative flows, and filling in the information using Microsoft Word.

**SWEBOK KA 3**

The third knowledge are in the SWEBOK is Software Construction. This area outlines fundamentals for software construction, managing construction, and practical considerations. The basic fundamentals of software construction give an outline of the most basic procedures that should be followed while constructing any program. Managing software construction gives us the models, planning, and measurement needed to effectively manage the construction of a program. The practical considerations are small tidbits that can be used to avoid common errors in construction or to improve efficiency.

This knowledge area realtes to my roles as programmer, web developer, and software tester because each role involves software construction in some form. A programmer must understand how to constuct software because that is a majority of the job. A programmer must have a basic understanding of the languages being used for the project and be flexible enough to learn new languages, concepts, or methods for software constuction. A web developer is no different from a programmer, the only difference is the platform. A software tester must understand software construction in order to be able to debug a program effectively to determine the location and cause of failures.



Figure : A screenshot of the iCal conversion servlet in Eclipse Indigo Java EE.

The output above was developed through a series of trial and error. The code above is from the servlet designed to parse information from the iCal file that will be generated by Sugar. It will later be modified to query a database for a student’s calendar file in order to parse the needed information for display purposes. If a student logs on after a schedule has been created, they will have the option to view that schedule; if they choose to view it the servlet will check the specified path for the file and read it using a buffered reader to trace the data in the file for each piece of information. All the information gathered from a specific event is then collected and added to a Schedule object that is displayed using jsp technology.

**SWEBOK KA 4**

The fourth area covered in the SWEBOK is software testing. This area focuses on rigorously testing software and its components to ensure functionality. This testing is done to verify that the software behaves as it should.

There are five topics in this knowledge area software testing fundamentals, test levels, test techniques, test related measures, and test process. Software testing fundamentals covers all of the basics of testing software from terminology to how testing relates to other activities. Test levels provide an overview of the different testing target and different testing objectives. Test techniques defines different techniques that can be used based on the intuition and experience of the tester; it also provides guidance on how to combine certain techniques. Test related measures provide a basis for the evaluation of software under the test and the evaluation of the tests performed. Test process defines some practical considerations for testing software and different test activities.

Software testing relates to each of my roles because each role involves testing in some manner. A programmer must test their code thoroughly before submitting it for addition to the overall project. A web developer must test each application in order to ensure that it works according to specification and operation on different web browsers. A software tester is always testing assigned software; the software tester ensures that the finished product is free of bugs and glitches.



Figure : The result of converting an iCal file to raw text.

The above output was created by construction a java program to read an iCal file. The program was designed to read a given iCal file and print each line of data to a text file. This file was then used in the development of the servlet used for display the course information contained in the iCal file. Each Line begins with a specific tag based on the data encoded on that line. Each line contains data specific to how the event will display, how often the event occurs, time and dates for start and end, time zone, and other pertinent information that contributes to the overall display.

**===**

**4. Attach:**

**All your individual progress reports for each week of the semester already submitted in table format to your project manager, who routinely submitted to the program manager every Wednesday all teamwork completed for the week.  Hence, for each week, simply attach your individual progress reports that provide information on your role activity for each week.  These weekly reports should be assembled into 1 individual table separated by rows of weeks, which include at least:**

**a. Tasks completed for each week.**

**b. Tasks planned for the following week.**

|  |  |  |
| --- | --- | --- |
| **Week** | **Tasks Completed** | **Tasks Planned** |
| 1/18/12-1/25/12 | N/A | Use Case Spec for View/Print Bingo SheetPMBOK Reviews KA 1 - KA 3 |
| 1/25/12-2/1/12 | PMBOK Reviews KA 1 – KA 3 | Use Case Spec for View/Print Bingo Sheet |
| 2/1/12-2/8/12 | Created Use Case Spec for View/Print Bingo Sheet | Refine Use Case SpecificationBegin conversion of Sugar output to Text for display |
| 2/8/12-2/15/12  | None | Convert Sugar output to text for display |
| 2/15/12-2/22/12 | Converted iCal file to raw text | Parse course information from calendar file for display based on text file |
| 2/22/12-2/29/12 | Information parsed from iCal file using information gathered from the text version of iCal file | Create and display a Schedule object containing the extracted information |
| 2/29/12-3/7/12 | None | Create and display a Schedule object containing the extracted information |
| 3/7/12-3/14/12 | None | Create and display a Schedule object containing the extracted information |
| 3/14/12-3/21/12 | None | Create and display a Schedule object containing the extracted information |
| 3/21/12-4/4/12 | Created a Schedule object containing the extracted information | Display schedule contents on jspRedo Exam IComplete PMBOK Reviews |
| 4/4/12-4/11/12 | Display schedule contents on jspRedo Exam IComplete PMBOK Reviews | Redirect error log thread produced by Sugar to JSP |