**CS 460 Exam 1 (in-class)**

**Name: Alek Bouillon Date: 4/9/12**

**Total: 100 pts**

**Part I: Software Engineering [Subtotal 80 pts]**

**(1) SWEBOK (Only Chapter 1: Intro to the Guide) [35 pts]**

**(2) PMBOK (Only Chapter 1: Intro to the Guide) [18 pts]**

**(3) RUP (Only pages 1-15) [12 pts]**

**(4) Team Roles/Tools/Standards [15 pts]**

**Part II: Essay [Subtotal 20 pts]**

**Provide an overview of your Project contribution:**

**(1)  Completed up to Spring Break (1/2 page + diagram) [10 pts]**

**(2)  Planned after Spring Break (1/2 page + diagram) [10 pts]**

**(3) Bonus [10 pts]**

**Part I: Software Engineering (SWEBOK, PMBOK, RUP, and Team Roles /Tool)**

**(1a) [T/F] [10 pts] SWEBOK Essentials**

|  |  |  |
| --- | --- | --- |
| **#** | **T/F** | **Statement**  |
| 1. | F | There is no difference between software engineering and programming code. |
| 2. | T | SWEBOK promotes a consistent view of software engineering worldwide and characterize the contents of the software engineering discipline. |
| 3. | T | At least 500 reviewers from 42 countries contributed to SWEBOK. |
| 4 | T | www.swebok.org is where more information can be found on SWEBOK. |
| 5. | T | Computer science is another discipline related to Software Engineering. |
| 6. | F | SWEBOK knowledge areas are hierarchically decomposed but not defined. |
| 7. | T | Each SWEBOK KA description also includes a matrix relating the reference material to the listed topics. |
| 8. | T | A requirement is defined as a property that must be exhibited in order to solve some real-world problem. |
| 9. | T | According to the IEEE definition [IEEE610.12-90], design is both “the process of defining the architecture, components, interfaces, and other characteristics of a system or component” and “the result of [that] process.” |
| 10. | T | The 2004 Guide is simply the current edition of a guide which will continue evolving to meet the needs of the software engineering community. |

**(1b) [3 pts] Define Software Engineering according to SWEBOK and describe how it relates to coding.**

According to SWEBOK, Software Engineering is “(1) the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software;” or applying engineering principles to software and studying the approaches. This definition relates to coding in that coding is the application of an engineering principle and it is systematic, disciplined, and quantifiable.

**(1c) [22 pts] SWEBOK Knowledge Areas**

|  |  |
| --- | --- |
| **Identify official Knowledge Area** | **Brief Definition** |
| 1.SoftwareRequirements | This knowledge area encompasses the elicitation, analysis, specification, and validation of any and all software requirements. |
| 2.SoftwareDesign | Defining the architecture, components, interface, and other characteristics of a system. |
| 3.SoftwareConstruction | This is the detailed creation of working, meaningful software using coding, verification, unit testing, integration testing, and debugging. |
| 4.SoftwareTesting | The evaluation and improvement of software quality by identifying any defects and problems. |
| 5.SoftwareMaintenance | Providing support for produced software to patch any defects in the systems that were not discovered during testing. |
| 6.SoftwareConfigurationManagement | SCM supports a process that supports the life cycle of a system such that project management, development and maintenance activities, assurance activities, the customer, and users benefit. |
| 7.SoftwareEngineeringManagement | The application of management activities to ensure that the development and maintenance of software is systematic, disciplined, and quantifiable. |
| 8.SoftwareEngineeringProcess | The process of applying engineering principles to the development of software. |
| 9.SoftwareEngineering Tools and Methods | Tools and methods that are designed to assist in the development and maintenance of software. |
| 10.SoftwareQuality | Testing the software to ensure that it meets all functional requirements and is free of major defects. |
| 11.RelatedDisciplines | All fields that share a common boundary with Software engineering. |

**(2) [18 pts] PMBOK Knowledge Areas**

|  |  |
| --- | --- |
| **Identify official Knowledge Area** | **Brief Definition** |
| 1.ProjectIntegrationManagement | The processes and activities that integrate the elements of project management are identified, defined, combined, unified, and coordinated. |
| 2.Project ScopeManagement | The process of determining that the project encompasses all the work required, and only the required work needed to complete it. |
| 3.Project TimeManagement | The process of ensuring timely completion of the project. |
| 4.Project CostManagement | The process of planning, estimating, budgeting, and controlling costs to ensure that the project is completed within the budget. |
| 5.Project QualityManagement | The process of ensuring that the project meets quality standards and will satisfy the objectives it was intended to satisfy. |
| 6.Project HumanResourcesManagement | The process of organizing and managing the development team. |
| 7.ProjectCommunicationsManagement | The process of generating, collecting, disseminating, and storing of project information. |
| 8.Project RiskManagement | The process of determining risks associated with the project and controlling them. |
| 9.ProjectProcurementManagement | The process of purchasing or acquiring the products, services or results, and contract management processes. |

**(3) [12 pts] RUP Phases**

|  |  |  |
| --- | --- | --- |
| **Identify Phase** | **Identify at least 1 project deliverable**  | **Briefly describe project deliverable** |
| 1.Inception | Project Plan | The overall plan for execution of the project. |
| 2.Elaboration | UML diagrams | An overview of the classes the project will consist of. |
| 3.Construction | System Code | Tested code that should be bug free and meet the requirements, Beta tested. |
| 4.Transition | Product | The finished product is delivered to the customer |

**(4) [15 pts]** **List your team members, state their roles, and match the tool/standard they use.**

|  |  |  |
| --- | --- | --- |
| **Team Member Name** | **At least 1 role**  | **At least 1 tool/standard used** |
| 1. Marat Kurbanov | Project Manager | MS Visio |
| 2. Trent Forkert | Database Developer | Postgres |
| 3. Yeisol Woo | Web Developer | XHTML |
| 4. Conner Becker | Asst. Project Manager | Acclaro |
| 5. Alek Bouillon | Web Developer | Java Servlet Technology, jsp |

**II. Project Essay**

**Provide an overview of your Project contribution:**

**(1)  Completed up to Spring Break (1/2 page + diagram)**

**(2)  Planned after Spring Break (1/2 page + diagram)**

**< next page>**

1. **[10 pts] Completed up to Spring Break (1/2 page + diagram)**

My assigned tasks during the time up to Spring Break were to create a Use Case Specification for View/Print Bingo, extract course information from the iCal file generated by Sugar, and display the extracted information on a web page.

I completed the Use Case Specification for View/Print Bingo one week behind due to a misunderstanding. I did not ask questions to ensure that I understood my assignment as well as I should have. The specification was merged with Marat’s specification for CRUD Bingo upon completion.

Extracting the course information from the iCal fell behind schedule and took longer than initially planned. Before Spring Break I had managed to convert the iCal file to a text file which provided vital information for parsing and extracting the necessary information. I had completed the extraction of the necessary information by the middle of Spring Break. This was accomplished by creating a dummy program in which an iCal file was scanned and read for specified tags and printed to a text file.



Figure : The revision history from the Use Case Specification View/Print Bingo.

1. **[10 pts] Planned after Spring Break (1/2 page + diagram)**

After Spring Break, my plans are to create and populate a Schedule object from the data extracted from the iCal file, display the contents of the object on a jsp, and merge the working code with the rest of the project.

This will be accomplished by modifying the code written prior to Spring Break to work as a web servlet. The servlet will accept a given iCal, which will be stored on the database with a unique identifier for each student, and parse the file for the tags specified. Once the tags are found, it will extract the information and store it in temporary variables that will be sent to the Schedule class for composition. Each part of the required information is then assembled into one String and added to an ArrayList. The ArrayList will be made accessible to the jsp using the Java Sessions API. This will be made possible by adding the Schedule class as an attribute of the session. Once the Schedule is visible to the jsp, I plan to use JSTL tags to loop through the Schedule to display each of the elements. Once the schedule is displayed, I plan to refine the appearance so that it better matches the standard layout for the pages of the group.



Figure : A snippet of the code to be modified.

**Bonus [10 pts total]**

**B-1 [5 pts] List up to 5 topics you learned this semester from class.**

**B-2 [5 pts] List up to 5 favorite topics from class.**

**B-3 [10 pts] Identify and describe your implementation duties in CS 460.**