**Elaboration Phase Specification – Deliverables**

|  |  |
| --- | --- |
| **Item** | **Description** |
| System Requirements  **Done (S will put on Drive)** | Provide a complete (and updated) set of system requirements, organized as a hierarchy. Include functional and non-functional requirements. State each requirement using the “shall” format described in Arlow and Neustadt. Provide an introductory narrative. |
| Use Case Diagram  **Cameron** | Create a use case diagram for the use cases. Use the guidelines in Arlow and Neustadt in formatting the diagram.    Provide a narrative explaining how to interpret the use case diagram; i.e., what you would want the client to conclude about the system model when viewing the diagram. |
| Trace Matrix  **Savanah** | Provide an updated trace matrix that maps use cases to system requirements. Provide a narrative that explains how use cases may be traced to top-level system requirements. |
| Use Cases  **Done (S will put on Drive)** | Provide a complete (and updated) set of detail, essential use case descriptions. Follow the actor/system response format shown in Arlow and Neustadt. |
| Sequence Diagrams  **Done (S will put on Drive)** | Create a sequence diagram for the “happy day” scenario of each use case. Follow the format shown Arlow and Neustadt. List the use case main flow along the left side of the sequence diagram. Place each sequence diagram immediately behind its corresponding use case when organizing the deliverable. |
| Class Diagram(s) | From the collection of sequence diagrams, create a class diagram(s) for all problem domain classes. Include data attributes, operations, and relationships among classes. Use the guidelines in the SAD text in formatting the class diagram.    Explain the class diagram in terms the client would understand. Also, provide a narrative explaining how the diagram was derived using either: (1) CRC cards; (2) verb-noun analysis; or (3) prototype analysis. Include the CRC cards if they were used in the analysis. |

|  |  |
| --- | --- |
| Database Design and Data Definitions | Create the data storage tables for a relational database management system deployment. Design the database in 3NF. Explain design decisions needed to achieve 3NF. Provide an Entity-Relationship Diagram (ERD) of the database layout. Provide an explanation of how the ERD was created from the class diagram.    Also, provide data descriptions (data types and sizes) for the attributes of each table; i.e., a database designer would have adequate information to create the database tables and fields. |
| User Interface Navigation Diagram and Screen Layouts  **Ryan** | Create a Window Navigation Diagram depicting the screens needed to realize the use cases. Use the guidelines in the SAD text in formatting the diagram. Provide a narrative explanation.    Provide screen layouts for input screens/forms (data capture) and output screens/reports (data presentation). Layouts may be borrowed from the HTML prototype (recommended) or presented in storyboard form. Explain how the screen layouts realize the use cases. |
| **Physical Architecture Design**  **Joseph** | Create a deployment diagram of the system architecture, presuming server hosting and support is provided by a third party. Allocate software components to hardware; i.e., follow the examples shown in the SAD text. Provide an explanation of the architecture decisions in terms that the client would understand. |
| **Design Procedures for Security Concerns and Non-functional Requirements**  **Zac** | List and justify procedures needed to address the non-functional requirements with special emphasis on security requirements. Explain how each requirement may be addressed during detailed design. |
| Gantt Chart  **Tyler** | Provide Gantt chart(s) showing project tasks for all iterations from the third iteration through the present one. The Gantt chart(s) should identify:  · Task ID number and name (dates from Syllabus Schedule)  · Task start date and duration  · Task responsibility (individual team member level)  · Task dependencies |
| Elaboration Phase Prototype  **Done (S will put on Drive)** | Create a high-level prototype (HTML or similar screen mockups) that represents the data needs and process flows of the use cases. The prototype may not connect to databases or use anything other than the minimum of programming. |