**CIS 4327 – Information Systems Senior Project I**

**Project Deliverable 3**

**Due Date:** December 3, 2015

**Relevant Disciplines:** Project Management, Requirements Management, Design and Analysis, and Construction

**Artifacts to be delivered:**

1. Activity Diagrams – One for each team member
2. Analysis Model – One for each team member
3. Sequence Diagrams – One for each team member
4. Preliminary Database Design
5. Sprint Backlog and Product Backlog in JIRA
6. Third Iteration Product Release
7. Update artifacts delivered in earlier project deliverables
8. Activity Diagrams

An activity diagram in the use-case model illustrates the flow of events of a use case. The flow of events of a use case describes what needs to be done by the system to provide value to an actor. It consists of a sequence of activities that together produce something for the actor. The flow of events consists of a basic flow and all alternative flows.

Each member of the team would develop an **activity diagram for a specific use case specification** (or system context). The choice of use case specification (or system context) of a team member **should be different from Project Deliverable 2**, i.e., you would be developing activity diagram for a use case specification developed by one of your team mate.

For each activity diagram, add a note containing your name – “Prepared by Your Name”. Use Microsoft Visio to develop Activity Diagrams.

Guidelines for preparing Activity Diagrams can be found at:

<http://sce.uhcl.edu/helm/rationalunifiedprocess/process/modguide/md_bactd.htm>

1. Analysis Model

The analysis model contains analysis classes (boundary, control and entity) and relationships among them. Analysis model represents an early conceptual design model of the system.

Analysis classes capture the major “clumps of responsibility” in the system. These classes represent the major abstractions that the system must handle. Analysis classes represent an early conceptual model for ‘things in the system which have responsibilities and behavior’.

Each member of the team would develop an **analysis model for a specific use case specification** (or system context). The choice of use case specification of a team member **should be different from Project Deliverable 2 as well as from one chosen for activity diagram**.

For each Analysis Model, add a note containing your name – “Prepared by Your Name”. Use Microsoft Visio to develop Analysis Model.

Guidelines for identifying classes:

<http://sce.uhcl.edu/helm/rationalunifiedprocess/process/modguide/md_acls2.htm>

Guidelines for representing associations between classes:

<http://sce.uhcl.edu/helm/rationalunifiedprocess/process/modguide/md_assoc.htm>

Guidelines for representing aggregation relationship between classes:

<http://sce.uhcl.edu/helm/rationalunifiedprocess/process/modguide/md_aggrg.htm>

Guidelines for representing generalization relationship between classes:

<http://sce.uhcl.edu/helm/rationalunifiedprocess/process/modguide/md_gener.htm>

1. Sequence Diagrams

A sequence diagram describes a pattern of interaction among objects, arranged in a chronological order; it shows the objects participating in the interaction by their "lifelines" and the messages that they send to each other. In most cases, we use a sequence diagram to illustrate how objects interact to perform the behavior of all or part of a use case. You can have objects and actor instances in sequence diagrams, together with messages describing how they interact. The diagram describes what takes place in the participating objects, in terms of activations, and how the objects communicate by sending messages to one another.

Each member of the team would develop a **sequence diagram for a specific use case specification** (or system context). Develop a sequence diagram for a successful use case scenario. The choice of use case specification of a team member **should be different from Project Deliverable 2 as well as from one chosen for activity diagram and analysis model** (ignore the choice of use case constraint for sequence diagram, if your team has less than or equal to three members).

For each Sequence Diagram, add a note containing your name – “Prepared by Your Name”. Use Microsoft Visio to develop Sequence Diagram.

Guidelines for preparing Sequence Diagrams can be found at:

<http://sce.uhcl.edu/helm/rationalunifiedprocess/process/modguide/md_seqdm.htm>

1. Preliminary Database Design

Database are core to the Web application systems. It is critical to design a preliminary database as a part of the analysis process. Preliminary database is designed based on our current understanding of the system. As our understanding changes, so will our product design and our solution. We will revisit database design to finalize it to fit with updated view of the system design.

**Create an entity relationship diagram** depicting the information that is created, stored, and used by your application system. The basic steps in building an entity relationship diagram are: identifying the entities, add the appropriate attributes to each entity, and then draw relationships among entities to show how they are associated with one another. The entities should represent the major categories of information that you need to store in your system. If you begin your data model using a use case, look at the major inputs to the use case, the major outputs, and the information used for the use case steps. Essentially, different information along with inputs and outputs managed by the system are good starting points to brainstorm for entities. Attributes and relationship between entities can be identified from the information that describes each entity, typically found in the requirements specification documents and other documents gathered during requirements phase. Preliminary database design does not need to apply normalization techniques, but you are welcome to do so.

Submit a Visio file containing preliminary database design as an entity relationship diagram developed using Crow’s Foot notation.

1. Sprint Backlog

A brief weekly meeting (weekly scrum) should be held by the team to inspect progress made on Sprint Backlog items. Report verbally as a team and individually to instructor on progress made on the project.

For this deliverable, your team is expected to provide the sprint backlog for November month iteration and sprint backlog plan for December month iteration.

You will be documenting Sprint Backlog using JIRA system. Please note that each student is expected to add and maintain progress made on relevant Sprint Backlog items in the JIRA system.

1. Third Iteration Product Release

As part of this deliverable, each team member is expected to **develop working prototype** to demonstrate something concrete and executable for a particular Use Case that is a core to the software. Prototype might be a small application created to test the performance of a key software component, or it may be a way of clarifying requirements, or to see if the developer understands a particular behavioral or technical requirement.

Each team member is expected to schedule an appointment with the professor in the second week of December (Dec 7, 8, and 9) to demonstrate the third iteration product release.

Please submit other relevant artifacts via blackboard. Remember to include updated use case specifications, project vision, and software development plan artifacts.