Requirements and Architecture Class Activities

The objective is to reinforce lecture material through practice exercises. We will use your team project as the basis for in-class activities, and generally you will work as teams. Tasks will progress from requirements through architecture activities, building on each other in a progression. Activity learnings will precede what will be needed for project deliverables and should contribute content to project deliverables. You will be expected to submit in-class work activity artifacts to a myCourses dropbox as directed by the instructor for participation grading. Periodically individuals will be required to answer quiz questions about the activity to be submitted for grading as directed by the instructor. The instructor will provide more detailed instructions for each activity in class.

#1 Business Requirements

Prepare an interview plan and interview the stakeholders to determine business requirements and rules. Research competition. What are the vision, purpose of the proposed solution, the scope, and the system boundaries? Specify a few important business requirements.

#2 User Requirements

Do another round of stakeholder interviews to identify the primary user roles, their goals, and their characteristics. What are the major functional features? Write user stories corresponding to major functional features. Model relevant use cases that correspond to the user stories. Review with stakeholders. [As stakeholders represent user requirements to external interface designers.]

#3 Quality Attributes

Explore the following quality attributes for your system with the stakeholders: performance, security, availability, interoperability, and modifiability. Create relevant QA scenarios for some of the QA’s. Derive and specify a few representative quality attribute requirements. Review with stakeholders.

#4 System Requirements

Perform use case analysis. Based on what you have learned about the system so far, specify any additional relevant system requirements and constraints. Specify interfaces. Review with stakeholders.

#5 Functional Requirements

Replace user stories with well stated requirements statements. Review all of your requirements. Are they “well stated”? Are there ambiguities? Review with stakeholders. Also, as individuals, review and critique an existing SRS and be prepared to report to the class.
#6 Quality Attribute (QA) Scenarios

Prioritize the QA’s for your project. Represent them in a QA utility tree. Write descriptive scenarios for each using the textbook template. Are they traceable to the SRS, do you need to revise the QA requirements as defined in the SRS? Do you need to create new QA scenarios? Review with stakeholders.

#7 Architecture Design Patterns

Research and present an architecture design pattern. Each team will be assigned a pattern by the instructor.

#8a Architecture Design

Apply the attribute driven design method (ADD) to design a first order module structure for your project. What patterns are you applying? What is your rationale for choosing those patterns? Each member of the team should first work independently to create their own design view and rationale. Next the team should consolidate individual designs into one consensus team design.

#8b Refine and Extend Architecture Design

As a team, refine and extend your design to support more requirements by going down a level in module abstraction using ADD. Try using a function driven approach. Add another structural view.

#9 Design Tactics

Each member of the team should choose a high priority quality attribute scenario from your utility tree. Choose a design tactic(s) you think is appropriate to address the QA scenario you previously authored and/or create additional scenarios. What is your design decision rationale? What are possible tradeoffs with other QAs of your decisions? Next the team should discuss individual design tactics to reach consensus on refinements to the team’s design.

#10 Architecture Tradeoff Analysis Method (ATAM) Evaluation

Perform an ATAM evaluation of your software architecture design in its current state with your stakeholders. Present the current design. Update and utilize the QA utility tree. Document and report your results.

#11 API Design and Construction

Individually, specify the API for a project system component that has an important internal-to-the-system interface role, or represents an external interface at the system boundary. Create a
prototype of the interface based on the specification. Test the usability of the interface prototype by conducting a “cognitive walkthrough” with a member of your sponsor team. Provide usage guidance as necessary. You will submit the specification, the API prototype, notes from your walkthrough, and a reflection on what you learned from the activity.