Architectures for Software Systems

**Sales system using a loyalty card point system**

**Final Project Guidelines**

Project Description: 9 January 2012

Draft Report Reviews: 6 May 2012

Final Report Due: 13 May 2012

Final Presentations: 20 May 2012

**PROJECT OVERVIEW**

**Extension Requirements: With the same system requirements described in Assignment 1, Company A decided to choose the Web solution using ASP.NET MVC 3 framework, only Web browser, no local Database needed for any POS terminal. And authorized managers can display statistic reports from Internet.**

The purpose of this project is to give you an opportunity to develop the architecture design of a realistic software system using the concepts presented in class. You will work in the teams assigned at the beginning of the semester. For this project your team will propose and document an architectural design, and analyze the relative merits of that design.

A key objective of this assignment is practice key principles presented in the course and to foster deeper thinking about what motivates architectural design decisions, practice architectural reasoning, and architectural documentation. Prior to developing your architecture design, you should first identify the key architectural drivers for your system (high level functional requirements, constraints, and quality attributes). Make sure that you describe the quality attributes using clear quality attribute scenarios as presented in class. You should prioritize your scenarios – numerous examples were provided in the class.

Once the architectural drivers have been developed and prioritized, they should be used to design the architecture, reason about alternatives, and evaluate the fitness of the architecture.

You may use QAW, ATAM, and ACDM as references. However, it is not necessary that you bind yourself to these – or any – methodology. The point here is not to blindly follow a method but practice key principles presented in class for how to articulate architectural drivers and use them to guide your design and in evaluating your architecture designs. Once you have designed your candidate architecture based on the architectural drivers, you should evaluate your architecture to ensure that it meets the architectural drivers. Again, you may use ATAM or ACDM techniques as a reference to guide the review and evaluation of your design. You may do one single design as suggested by ATAM or iterative evaluations and refinements as ACDM suggests. Again, you don’t have to follow either of these methods verbatim – in fact, it would be nearly impossible to perform a complete ATAM evaluation to your architecture. Feel free

to tailor these methods and techniques to meet your project needs. To support your analysis, you should identify and describe the architecture drivers. It can be helpful to prioritize functionality and quality attributes (as described throughout the course). Make sure that quality attributes are quantified (again, various strategies were discussed in class). To evaluate the fitness of the architectures, analyze the structures and tactics you have selected and explain how the design satisfies the architectural drivers.

Functionality can be analyzed by stepping through use cases and tracing through the design. When evaluating quality attribute requirements, think in terms of the following to frame the analysis: *“Given the stimulus from the stated source, under the environmental conditions indicated, stimulating the artifact listed in the scenario, show how the system responds within the response measure indicated in the scenario.”*

The team should use the architectural artifacts to trace the stimulus and analyze how the architecture design responds. Capture the architectural decisions and reasoning that is reveled during this analysis.

Capture any issues you may have discovered and how you addressed them in the design.

**PROJECT DELIVERABLES**

**Draft Report Review**

The purpose of the preliminary report is an opportunity for your team to get direct feedback from the instructors and TAs regarding your progress on the final project. The preliminary report will not be graded – but you must do it or it will negatively impact your grade. The document does not need to be in a completed final format, but the general outline should be in place. At a minimum the preliminary report should include the problem statement, architectural drivers, preliminary utility tree, candidate styles/patterns, and key design decisions and rationale you may have already made. Once we review the preliminary report, we will set up interviews with the teams to provide direct feedback from the instructors and TAs.

**Final Report**

The final report is a typically 15-30 page document containing the description of the project and ATAM output including: project context, architectural drivers, architectural design representations and descriptions, rationale, analysis/comparison of the key design decisions and evaluation.

Each team should prepare an architectural design document (final report) that contains the following parts:

**1**. A brief description of the project context and business case for your system.

**2**. The architectural drivers: You may document key functionality any way you like. Include a prioritized list of quality attribute requirements. Describe business and technical constraints.

**3**. A description of your proposed architecture, using any architectural views and prose that you deem appropriate to communicate the design and support analysis. YOU MUST FOLLOW THE GUIDELINES PRESENTED IN THE COURSE WHEN REPRESENTING YOUR ARCHITECTURE. Poor documentation practices at this point in the course will be graded harshly.

**4.** An analysis of the architectural design based upon the architectural drivers as described earlier. Your analysis should expose key issues and how the architecture fares with respect to architectural drivers. Your analysis should include a discussion the various alternatives you considered and the important tradeoffs your team made. Alternatives may be small structures, larger systemic patterns, or even alternative architectures depending upon the nature of your project. However, your alternatives should be realistic, plausible alternatives, not obvious nonstarters.

**Final Presentations (In-class presentation: 30’ for presentation, 10’ for demo and 15’ for answering questions):**

The final presentations will be held during finals week – the schedule will be published at a later date. The final presentation should summarize the 4 elements of the paper described above.

**Marking:**

Architecture Drivers document: 30%

Architecture Document: 50%

Analysis of Architecture Document: 20%

Presentation: Penalty depend

Implement and demo at least the sales operation: Penalty depend