**Name:** Matthew Rasler **Date:** 11/17/2011

**(Type answers and include question)**

**CS 360 Midterm I: Software Architecture and OO Detailed Design (Fall 2011)**

1. **[10 pts] We have covered the architectural hierarchical decomposition process from your initial vision document. Describe:**
	1. **Application architecture**
		1. Is the planning and design activity at the project scope level of detail, resulting in reusable content that aids in assuring scalable, reliable, and manageable applications.
	2. **Information architecture**
		1. Organization, labeling, and navigation schemes within an information system to facilitate task completion and intuitive access to content.
	3. **Systems architecture**
		1. Is the conceptual model that defines the structure and behavior of a system which comprises system components, external properties, and the relationships between them.
	4. **UML architecture**
		1. We created UML architectures to express a class view of the system as well as a component view of the system utilizing the UML standards to express them in a highly conceptual way.
2. **[10 pts] In RUP, we are leveraging the axiomatic design process in the inception and elaboration phase. Describe the requirements engineering and elicitation process in more detail from initial gathering of VOC to CN to FR in axiomatic design, including risk concepts.**
	1. **Design matrix results in which UML diagram type?**
		1. Class UML
	2. **DSM results in which UML diagram type?**
		1. Component UML
	3. **What is the V-Model**
		1. Describes the decomposition process from customer attributes through Functional Requirements resulting in the design matrix which can then be used at the module level to build back up the object-oriented model to the end software product.
	4. **QFD**
		1. Quality Function Deployment : assists in translating often subjective customer desires into objective design characteristics.
	5. **FMEA**
		1. Failure mode and effects analysis: is a methodology for analyzing potential reliability problems early in the development cycle, allowing engineers to design out failures and produce more reliable products at the design level.
3. **[5 pts] We have covered how to establish class architecture first as a foundation to detailed design with UML. When using Axiomatic Design process to develop classes, how do the following map to OO Design object elements [hint: see paper handout on “Object Oriented Design with Axiomatic Design” :**
	1. **FR**
		1. Object or Behavior
	2. **DP**
		1. Data or input
	3. **FR/DP design matrix intersection**
		1. Method
4. **[5 pts] Describe the purpose of these tools/software used in your project:**
	1. **Acclaro DFSS**
		1. To facilitate the mapping of functional requirements to end software level objects, classes, and methods by creating design structure matrices and design matrices that will not only provide information about design issues during the design phase, but also allow for the changes of functional requirements through the design phase with only marginal rework. Also, the tool aids in the demarcation of risk area by doing FMEA, and aids in QFD by its House of Quality tool integration.
	2. **Visio**
		1. Visio has aided in the rapid development of UML diagrams, Application Architectures, and Flow Charts.
	3. **Basecamp**
		1. To coordinate between group members and allow for cross communication that is recorded for reference sake and aids in the goal of centralized discussion. It also aids in keeping all project files in one location so the most recent is always available. It also enables the project manager and program manager to place milestones and check progress on them.
	4. **MS Project**
		1. Used as a tool to control and record project schedules and a view of those schedules (Gantt charting), aids in communication of project information, helps organize work and group members to assist in assuring timely resolution of projects and subprojects.
	5. **MS Powerpoint**
		1. Used to communicate progress by highly visual means, aids in the quick development of progress reports.
5. **[5 pts] Describe the standards applied in this project**
	1. **IEEE-830 SRS**
		1. Software Requirements Specification: a description of the behavior and intended purpose of the system being developed, what it will do, and how it will perform. It also defines how an application will interact with system hardware and software.
	2. **IEEE-1058 PMP**
		1. A document that expresses the priority of project components and deliverables; and assists in the centralized management of those components.
	3. **IEEE-1016 SDD**
		1. A representation of a design view of the project to be used for communicating design information to its stakeholders.
6. **[5 pts] In OO design, describe the concept [with symbol]:**
	1. **Aggregation**
		1. Similar to composition, but it does not imply ownership in the way composition does. The aggregate object is not destroyed when the owning object is destroyed.
		2. 
	2. **Composition**
		1. Allows for the combining of simple objects or data types into more complex ones.
		2. 
	3. **Polymorphism**
		1. Ability to create a variable, object, or function that has more than one form.
		2. 
	4. **Inheritance**
		1. Is a way to reuse code of existing objects, establish a subtype from an existing object or both.
		2. 
	5. **Blackbox**
		1. An object that can be viewed solely in terms of its input, output, and transfer characteristics without any knowledge of its internal workings.
7. **[5 pts] In OO design, describe the concept [with symbol]:**
	1. **Public operations**
		1. Any object can refer to or call the operation.
	2. **Private operations**
		1. Only the current object will have access to the operation.
	3. **Inclusions**
		1. Allows for handling of values of any type having a subtype relation with the expected type**.**
	4. **Extensions**
		1. Same as inheritance
8. **[5 pts] You were given a handout on software project management by Fairley. Describe how the following concepts are used in your project.**
	1. **Architecture Decomposition View (ADV)**
		1. Was used by the project manager (myself) to develop the WBS.
	2. **Work Breakdown Structure (WBS)**
		1. Used to allocate responsibilities and timelines to individuals working on subprojects associated with the project as a whole.
9. **[5 pts] In Object oriented (OO) design, describe the concept [with symbol]:**
	1. **Association**
		1. A bidirectional (distinct to aggregation) relationship between two objects where one object uses another. Aggregation and Composition can both be said to be more restrictive subsets of Association.
		2. 
		3. 
	2. **Generalization**
		1. Reduces complexity by replacing multiple entities which perform similar functions with a single construct.
	3. **Dependency**
		1. When an object requires the use of another object to operate. The Dependency Inversion Principle in OOD prioritizes the use of dependency on abstract objects over concrete objects.
	4. **Realization**
		1. A rough breakdown of the processes, actors, and data required for the system; a general outline of what is required by the system; in class form.
	5. **Annotation**
		1. A well defined comment, typically one that follows a documentation style for comment parsers, i.e. Javadoc.
		2. Symbol: text in box with folded corner
	6. **Interface**
10. **[5 pts] Define each performance attribute:**
	1. **Efficiency**
		1. Software should not make wasteful use of system resources such as memory and processor cycles.
	2. **Flexibility**
		1. How easily change can be made to a system.
	3. **Integrity**
		1. The level of insurance that the systems program and data are not damaged
	4. **Security**
		1. The ability of the system to resist attack.
	5. **Maintainability**
		1. Software should be written in such a way that it may evolve to meet the changing needs of customers.
	6. **Portability**
		1. Measures how many target systems apply to the system.
	7. **Reliability**
		1. What is the probability of the component failing or producing an incorrect output.
	8. **Usability**
		1. Software must be usable, without undue effort by the type of user for whom it is designed, should have appropriate user interface and documentation.
11. **[5 pts] Describe how you have used your concept map and basecamp tool to organize your work as a team using RUP as a guide.**
	1. Our CMAP is demarcated into the RUP phases: Inception Phase, Elaboration Phase, Construction Phase and Transition Phases. This allows for the visual understanding of which material belongs to which phase and this in turn allows for an understanding of how the material should be interpreted. It has predominantly acted as a communication tool and storage mechanism that is available to all group members and Program Manager. Objects represent specific tasks, and lines represent associations between tasks. The flow of the CMAP works from top down indicating that a node higher up associated to a node (object) below it is a pre-requisite. This has been useful in the describing and allocation of tasks. We have not utilized Basecamp other than to initiate it as per request, and to learn of it for possible future uses. We have not utilized Basecamp because we had already established the functionality that the software offered through other means, and I (the project manager) thought it unwise to allocate time and energy in utilizing this product at a loss to efficiency. The CMAP tool has been amazingly useful in its ability to act as a virtual filing cabinet, check list, and progress demonstrator.
12. **[5 pts] You are working on SWEBOK reviews KA-1 through KA-11. Which three areas apply to your role in the project and why?**
	1. **Area 1**
		1. KA1: As a member of the group in developing this project, I am a Process Actor in that I have a legitimate interest in profiting from developing the software and as such had to weigh my requirements against those of the customer in defining requirements of the system.
	2. **Area 2**
		1. KA2: As the secondary Application Architect and the author of the third version of the Application Architecture, I have had to consider issues related to this KA, such as distribution of components, interaction of components, and data persistence.
	3. **Area 3**
		1. KA3: Through the use of Axiomatic Design, we as a group have utilized a process that assists in minimizing complexity of the construction of the system. This process also allows for the anticipation of change in the construction phase by allowing for minimal effort to change requirements by utilizing the Aclarro tool.
13. **[10 pts] Other than class and component diagrams that all teams are required to use, select 3 UML behavior diagram types that your project could use and why?** 
	1. We could benefit from a State Machine Diagram; there is an expected outcome at each activity, and designing this view could aid in determining communication issues and help with verification.
	2. We have utilized a Communication Diagram to assist in verifying certain assumptions of our component diagram.
14. **[10 pts] We reviewed UML and related concepts for automation in software systems engineering.**
	1. **What is the relationship between UML and SysML?**
		1. SysML is an extension of a subset of the UML using UML’s profile mechanism; is defined as a dialect of UML 2.x, and reuses some of its notations and semantics.
	2. **What is executable UML?**
		1. It produces a comprehensible high-level independent abstraction of the organization of the software, is machine and language independent, derive models that have formal semantics so they can be tested, are executable, and can be translated directly into code.
15. **[10 pts] Describe how this course has helped you organize your team**
	1. **Management**
		1. This course has helped me communicate more effectively by being able to immediately observe the results of my communication style, method, and frequency. It has helped me in delegate more effectively by being responsible for the outcome of projects that are too large for only one man.
	2. **Architecture**
		1. Through the various application architectures and the two UML Diagrams, we have been forced as a team to consider the project at a level of depth that was previously neglected by me. This process has provided insight into potential pitfalls and error situations and has aided in the general understanding of the workings of the end result. The Axiomatic Design Process has shown a different side to the project as well by forcing all the Customer’s Needs into the System as they are the foundation for the Functional Requirements and the architectures created from them.
	3. **Detail design**
		1. Doing Requirement Specification helped discern the customer’s needs more clearly; the application architectures, vision document, and SRS helped in assuring the whole group was on the same page in what those customer needs were and how we planned on tackling them. Breaking down aspects of the system with Axiomatic Design, and subsequent Application Architectures helped in solidifying these concepts. I believe our system is small scale enough that we could have tackled it from ground up with little in the way of design, but iteratively going through the design processes and breakdown of components has helped us decipher a more efficient construction game plan, and helped us as a group have cohesion in our direction and purpose in even the smallest of aspects, which will enable us to interact more fluently in the construction phase.
	4. **Documentation**
		1. Through the IEEE standards, the given templates, and the mandate to actually complete the documentation, our documentation is more systematic, professional, and complete than it would be otherwise.

**Bonus: [10 pts] List up to 10 aspects of this course you enjoyed/learned from the most?**

* + - 1. I enjoyed the role of the project manager in that it took me out of my comfort zone and forced me to adapt to get stuff accomplished. I have never been shy of working hard, but relying on and trusting others puts me outside my comfort zone (kind-of a control freak), so it was hard initially to responsibly delegate. But the vastness of paper work shortly forced me to adapt or sink. And my group members are great. This is probably what I would consider the most valuable aspect of what I learned from this course.
			2. I enjoyed working with my group and getting to know them as people and team-mates; it was good to brain storm and figure this out with them and to be acquainted with such intelligent and capable figures.
			3. I learned a lot about the Rational Unified Process, which is great considering Raytheon wants to see that from new software engineering hires.
			4. I learned a lot software this semester, some very useful, some that looks great on a resume, and some that I will probably uninstall as soon as the semester ends. But knowing Aclarro, Visio, Project, CMAP tools and BaseCamp is powerful whether I use it again or not; they have expanded my efficiency level and communication capability.
			5. I learned a lot about what Software Engineering entails from standardized documents, in class, and SWEBOK; this is helpful in determining where I want to go with my career.
			6. I enjoyed learning about the embedded hardware, the software that powers it, and the communication standards that control it for our project. I have always been interested in this, but haven’t had the opportunity to actually play around with anything until this project.