**Name: Ekaterina Schwartz Date: November 21, 2011**

***(Type answers and include question/only brief answers required/1-3 sentences each subtopic)***

**CS 560 Midterm I: Software Engineering(Fall 2011)**

1. **[10 pts] We have covered the architectural hierarchical decomposition process from your initial vision document. Describe:**
   1. **Application architecture**

The application architecture includes various component artifacts and how they work together. It involves defining the functional interaction between applications, databases, and middleware systems.

AMAM specific: The application architecture includes presentation layer, application layer, and data layer, each specifying components and interactions within. Security spans across all three layers.

* 1. **Information architecture**

Information architecture is defined by the Information Architecture Institute as:

* The structural design of shared information environments.
* The art and science of organizing and labeling web sites, intranets, online communities, and software to support findability and usability.
* An emerging community of practice focused on bringing principles of design and architecture to the digital landscape.

AMAM specific: The information architecture of AMAM is represented with FR-DP design matrix and Design Structure Matrix developed following the three layers of the application architecture.

* 1. **Systems architecture**

Systems architecture is the conceptual model that defines the structure, behavior, and views of a system. It is used to convey the informational content of the elements comprising a system, the relationships among those elements, and the rules governing those relationships.

AMAM specific: System Architecture is created with the Acclaro tool based on FRs entered.

* 1. **UML architecture**

UML is a modeling language used to describe system architectures. It offers a standard way to visualize a system's architectural blueprints. It is used for modeling the structure, behavior and interactions of software, hardware and business systems, standardized by the Object Management Group (OMG).

AMAM specific: class diagrams, component diagram, use case diagrams, and sequence diagrams are implemented.

1. **[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.**

The initial gathering of VOC for the AMAM project consisted of project member meetings for brainstorming, which led to the determination of demands that the application is to satisfy. Such were reflected in various sections of AMAM’s Vision document, along with a number of customer needs. Following, the application architecture was created to reflect the product overview and demands as dictated by the Vision document. Functional requirements were created based on the application architecture and Vision document, and were listed by implementing the Acclaro tool. Design parameters were mapped to each of the functional requirements. Risks associated with the project application were also recorded with the Acclaro tool, under FMEA analysis.

* 1. **Design matrix results in which UML diagram type?**

The design matrix results in the UML class diagram.

* 1. **DSM results in which UML diagram type?**

DSM results in the UML component diagram.

* 1. **What is the V-Model**

The V-model for software represents a software development process. The first step in the V-model is to design the software, build the software hierarchy that follows the top-down approach of axiomatic design, and then generate the full design matrix table to define modules. The final step is to build the object-oriented model with a bottom-up approach. The top-down approach starts with customer needs and includes: listing of FRs, mapping such to DPs, and Decomposition. The Full design matrix includes the definition of modules. The bottom-up approach includes: identifying classes, establishing interfaces, and code with system architecture and concludes with the software product.

* 1. **QFD**

Quality Function Deployment is a systematic approach to design based on a close awareness of customer desires, coupled with integration of corporate functional groups. The goal of QFD is to translate subjective quality criteria into objective criteria that can be quantified and measured and can then be used to design and manufacture the product. The 3 main goals in implementing QFD are to prioritize spoken and unspoken customer wants and needs, translate these needs into technical characteristics and specifications, build and deliver a quality product or service by focusing everybody toward customer satisfaction.

* 1. **FMEA**

Failure modes and effects analysis (FMEA) is a procedure for analysis of potential failure modes within a system. A successfully completed FMEA helps to identify potential failure modes, enabling the design team to eliminate those failures with the minimum of effort and resource expenditure, thereby reducing development time and costs. Failure modes are any errors or defects in a process, design, or item, especially those that affect the customer, and can be potential or actual. Effects analysis refers to studying the consequences of those failures.

1. **[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**

Object=FR

FR’s of the axiomatic design are the classes in the UML class diagram.

* 1. **DP**

Attributes data structure =DP

DP’s can be mapped to the attributes of the UML class.

* 1. **FR/DP design matrix intersection**

Method FRi = Aij DPj

The FR/DP intersections are the methods of the class.

1. **[5 pts] Describe the purpose of these tools/software used in your project:**
   1. **Acclaro DFSS**

The Acclaro DFSS is used in the AMAM project for the completion of the FR-DP decomposition, FR-DP Design Matrix, DP-DP Design Structure Matrix, Failure Mode and Effects Analysis (FMAE), and Quality Function Deployment (QFD) House of Quality.

* 1. **Visio**

Visio is used in the AMAM project for the completion of the Application Architecture, UML Use-case diagrams, UML Activity Diagrams, UML Component Diagrams, and UML Sequence Diagrams.

* 1. **Basecamp**

Basecamp is a project management tool that provides project team members with tools for communication, file sharing, due-date assignments, and others. Due to the small size of the AMAM team, basecamp is not utilized for the completion of the AMAM project.

* 1. **MS Project**

Microsoft Project is used for the completion and update of the project’s Gantt chart.

* 1. **MS Powerpoint**

Microsoft Powerpoint is used in the AMAM project for the creation of multiple progress presentations and the final AMAM project presentation.

1. **[5 pts] Describe the standards applied in this project**
   1. **IEEE-830 SRS**

The IEEE-830 standard describes recommended approaches for the specification of software requirements. It lists the references made to other standards, provides definitions of specific terms used, provides background information for writing a good SRS, discusses each of the essential parts of an SRS. It also provides two annexes, one which provides alternate format templates, and one which provides guidelines for compliance with IEEE/EIA 12207.1-1997.

* 1. **IEEE-1058 PMP**

The IEEE-1058 standard describes the format and contents of software project management plans and identifies the elements that should appear in all software project management plans. It defines the scope and purpose of the standard, provides references to IEEE standards that should be followed when applying the 1058 standard, and provides an overview and detailed specification of the standard and the required and optional components that should or may be included in the project management plan based on the standard.

* 1. **IEEE-1016 SDD**

The IEEE-1016 standard describes software designs and establishes the information content and organization of a software design description (SDD), a representation of a software design to be used for recording and communicating design information to key design stakeholders.

The IEEE-1016 standard does not prescribe specific methodologies for design, configuration management, or quality assurance and does not require the use of any particular design languages, but establishes requirements on the selection of design languages for use in an SDD.

1. **[5 pts] In OO design, describe the concept [with symbol]:**
   1. **Aggregation**

Aggregation is the relationship between two objects that is described as a ‘has a’ and a ‘whole/part’ relationship.

Following is the symbol used for aggregation:



* 1. **Composition**

Composition is a form of aggregation that involves a strong relationship between an aggregate object and its component objects. With composition, the component exists as a component only within the composite object.

Following is the symbol used in UML for composition:



* 1. **Polymorphism**

Polymorphism occurs when an operation has the same name in different classes and performs different operations. In object-orientation each class "knows" how that operation is supposed to take place.

* 1. **Inheritance**

Classes can inherit attributes and behavior from pre-existing classes referred to as superclasses, parent classes, or ancestor classes. The new classes are known as derived classes, subclasses or child classes. The parent class is more general than the child class. A child class can be the parent of another child class.

Example: Appliance is a parent class with washing machine, dryer, and refrigerator each being a child class. Appliance is more general than the child classes.

Following is the symbol used in UML for inheritance:



* 1. **Blackbox**

A ‘black box’ view is an external view of a model’s component’s interfaces. It does not reveal what is inside a block.

Following is an example of blackbox view of the tutorial database interface of the AMAM project:



1. **[5 pts] In OO design, describe the concept [with symbol]:**
   1. **Public operations**

With public operations usability extends to other classes.

Following is the symbol used in UML to indicate public operations: +

* 1. **Private operations**

With private operations, only the original class can use the particular operation or attribute.

Following is the symbol used in UML to indicate private operations: -

* 1. **Inclusions**

Including a use case (Inclusion) is a technique of reusing a use case. An include dependency is a generalization relationship denoting the inclusion of the behavior described by another use case.

Following is the symbol used in UML to denote inclusion:



* 1. **Extensions**

Extension refers to the creating of a new use case by adding steps to an existing use case. The new use case is said to extend the original one because it adds new steps to the sequence in the original use case. The original use case is called the base use case.

Extension can only take place at specific designated points within the base case's sequence, referred to as extension points.

Following is the symbol used in UML to denote extension:



1. **[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)**

The architecture decomposition view (ADV) specifies the hierarchical ‘is-part-of’ relationship among software components. The ADV concept is used in our project by implementing the 4 layers of architecture: Application, Information, Systems, and UML. Acclaro’s FR-DP Decomposition and Systems Architecture (Module Diagram) of the AMAM project depict an ‘is-part-of’ relationship among software modules. The application architecture depicts a high view of the ‘is-part-of’ relationship among components.

* 1. **Work Breakdown Structure (WBS)**

The relationships among activities and tasks in a Work Breakdown Structure are ‘is-part-of’ relationships. WBS is a tool for planning, estimating, measuring, and controlling a software project. AMAM’s Gantt chart is used to plot task and artifacts to be completed, the timelines within they are to occur, and the assigned resources. Tasks are based on and completed concurrently based on the architecture decomposition view.

1. **[5 pts] In Object oriented (OO) design, describe the concept [with symbol]:** 
   1. **Association**

Association is a relationship between objects in which one object instance causes another to perform an action on its behalf.

Following is the symbol used in UML to denote an association:



* 1. **Generalization**

The generalization relationship works for use cases as well as classes. Classes can inherit from one another, and so can use cases. In use case inheritance, the child use case inherits behavior and meaning from the parent and adds its own behavior. You can apply the child wherever you apply the parent.

Following is the symbol used in UML to denote generalization:



* 1. **Dependency**

Dependency occurs when one class uses another. The most common usage of a dependency is to show that the signature of one class's operation uses another class.

It indicates that one class depends on another because it uses it at some point of time. One class depends on another if the latter is a parameter variable or local variable of a method of the former.

Following is the symbol used in UML to denote dependency:



* 1. **Realization**

The relationship between a class and its interface is called realization. A class is said to realize its interface.

Following is the symbol used in UML to denote realization:



* 1. **Annotation**

The note is the UML's annotation element. Notes enable one to attach constraints, comments, requirements, and explanatory graphics to your models.

Following is the symbol used in UML to denote an annotation:



* 1. **Interface**

An interface is a class that just has operations and has no attributes. Objects present an interface so that other objects or people can call its operations.

Following are the symbols used in UML to denote an interface:



1. **[5 pts] Define each performance attribute:**
   1. **Efficiency**

The software should not make wasteful use of system resources such as memory and process cycles, etc. Efficiency includes responsiveness, processing time, memory utilization, etc.

* 1. **Flexibility**

Flexibility describes the ease with which the software can be used in various situations and environments.

* 1. **Integrity**

Integrity is the ability of a system to withstand attacks to its security.

To measure integrity one must define:

* Threat: the probability that an attack will occur within a given time.
* Security: the probability that the attack of a specific type will be repelled.
  1. **Security**

Security is the process of ensuring confidentiality, integrity, and availability of a system, its programs, hardware devices, and data.

* 1. **Maintainability**

Maintainability refers to the ability to easily understand, enhance, and correct software. Criteria for maintainability include consistency, simplicity, conciseness, self-descriptiveness, and modularity. MTTC (mean time to change) is the time it takes to analyze, design, and implement a change.

Maintainable programs have a lower MTTC.

* 1. **Portability**

Portability refers to the capability of software to be transferred from one environment to another. Criteria for portability are: self-descriptiveness, modularity, machine independence, and software system independence.

* 1. **Reliability**

Reliability is the capability of software to maintain its level of performance under stated conditions for a stated period of time. It is also defined as the probability of failure-free operation. Criteria for reliability include: accuracy, error tolerance, consistency, and simplicity.

* 1. **Usability**

Usability is an attempt to define user friendliness, the ease with which a system can be used. It can be measured in terms of: physical and intellectual skill required to learn the system, time required to become moderately efficient in the use of the system, the net increase in productivity over the system it replaces, a subjective assessment of users' attitudes towards the system.

1. **[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.**

The AMAM’s project concept map displays the first two phases of the Rational Unified Process: Inception and Elaboration, along with project details such as project name, course, etc. The Inception and Elaboration phases of AMAM’s Cmap further provide links to Inception and Elaboration Cmaps respectively where documents pertaining to each phase are linked. With the implementation of Cmap, the work completed pertaining to each phase can be clearly identified and obtained.

Basecamp was not utilized in the development of the AMAM project.

1. **[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**

**Software Requirements** KA applies to the establishing of functional and non-functional software requirements, requirements process, requirements elicitation, requirements analysis, and requirements specification of the AMAM project.

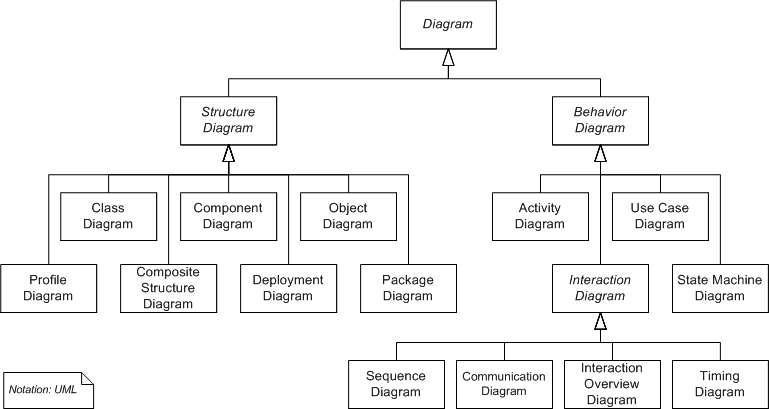
* 1. **Area 2**

**Software Design** KA applies to software decomposition and organization, and the interfaces between those components of the AMAM project. This KA applies to the SDD and its structural and behavioral aspects of the AMAM project.

* 1. **Area 3**

**Software Engineering Management** KA applies to the AMAM project in the determination of requirements, establishing of process for review and revision, and determining the effort and schedule, as incorporated in the Project Management Plan of the AMAM project.

1. **[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?**



**Use case Diagrams:**

A use case is a description of a system's behavior from a user's standpoint. We selected to implement use cases in order to depict application requirements as envisioned by each project member.

**Activity Diagrams:**

An activity diagram details the sequence of steps, processes, decision points, and branches. It provides a view of what happens during a process or operation. It is an integral part of system analysis, and therefore was selected to be incorporated in the AMAM project.

**Sequence Diagrams:**

A sequence diagram shows how an object interacts with other objects, while including the dimension of time. It shows the interactions among objects that take place in a specified sequence, and the sequence takes time to go from beginning to end. We selected to incorporate the sequence diagram in our project, as it provides the necessary detail of interaction of the objects depicted in the class diagram.

1. **[5 pts] We reviewed UML and related concepts for automation in software systems engineering.**
   1. **What is the relationship between UML and SysML?**

The Unified Modeling Language (UML) is a graphical language for modeling the structure, behavior and interactions of software, hardware and business systems, standardized by the Object Management Group (OMG).

The Systems Modeling Language (SysML) is a general-purpose modeling language for systems engineering applications that supports the specification, analysis, design, and verification and validation of a broad range of complex systems. These systems may include hardware, software, information, processes, personnel, and facilities.

SysML is a UML Profile that represents a subset of UML 2 with extensions.

* 1. **What is executable UML?**

xUML is an executable subset of standard UML that can be used to define, in an operational style, the structural and behavioral semantics of systems.

A behavioral model is executable if it is complete enough that the specified behavior can be enacted or simulated by an automated execution tool.

1. **[5 pts] Describe how this course has helped you organize your team**
   1. **Management**

The MS Project software provided to us for the completion of the Gantt chart was used by our team for project management. Following the Rational Unified Process has helped our team management by providing us with an overview of the project’s phases and expected deliverables.

* 1. **Architecture**

The Visio software provided to us was used for the creation of the application architecture. The Visio tool was also used for the creation of the project’s UML architecture. The reference to Teach Yourself UML in 24 Hours was helpful in completing and selecting the UML architecture of the project. The Acclaro tool provided to us was used for our teams’ information and system architectures.

* 1. **Detail design**

The Design Matrix and Design Structure Matrix implemented in Acclaro were used as a reference in our project’s detail design for determining classes, methods, attributes, and interfaces. Visio software and UML book were used for the completion of UML diagrams. Document templates helped ensure the necessary degree of project detail.

* 1. **Documentation**

The IEEE standards and document templates provided helped guide our team through the completion of necessary documentation.

1. **[5 pts] Cyberphysical systems: Describe** 
   1. **What is it?**

Cyberphysical systems deeply integrate communication, computation, and control into physical systems. Its characteristics are:

* Cyber capability in physical components
* Networked at multi and extreme scales
* Complex in many scales
* Dynamically reorganizing/reconfiguring
* High degrees of automation
* Operation must be dependable and/or certified

Cyberphysical systems are not traditional embedded real time systems or today’s sensor nets. CPS exploit pervasive, networked computation, sensing and control.

* 1. **Why is it important?**

One reason why CPS is important is that it can be used to enable cyber advanced discovery and innovation to enhance understanding and management of complex systems. It can be used in diverse fields such as health and medicine and automotive, among others.

1. **[5 pts] Software Engineering: Identify a topic of interest in**
   1. **Past**

Software Engineering first appears as a term in the literature in the 60s. In the 70s, IBM Federal Systems takes an initiative to train all programmers in structured programming and leaders in structured design. IBM forms Software Engineering Institute, an educational organization at the corporate level. Software engineering books start to appear. In the 80s, curriculum workshops are held and a MSE Model curriculum is published. In the 90s, CSEE becomes CSEET, an IEEE Conference.

* 1. **Present**

Experts are engaged to help mentor present and future software engineering educators. Academy for Software Engineering Education and Training is added to the CSEET conference (Conference on Software Engineering Education and Training). CSEET goes international.

* 1. **Future**

The future of Software Engineering outlines the development of new curriculum, such as Integrated Software and Systems Engineering. The future of Software Engineering also holds globalization – not just international programs but programs   
that reflect the global nature of software engineering. Another trend that will be a part of the future of Software Engineering is the use of electronic publishing – journals, books, use of the Web for research, and Webcasts.

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

* I learned from my project collaboration with Monica Gloudemans.
* Making available to us the software needed (Visio, MS Project) for the completion of certain project deliverables provided flexibility to our project progress.