**Professor: Mohamed Khan ( Instructors : Kindly submit any suggestions for edits to me prior to any change to this document – consistency is key to success of the project across sections)**

# COMP 246 OO Software ENGINEERING 2016

# Project & Evaluation GUIDELINES

**Software Requirements:**

You can use any tools capable of creating required documents. Some suggestions are:

* Visual Paradigm
  + - Community version ( VP-UML)

(<http://www.visual-paradigm.com/product/vpuml/vpumluserguide.jsp>)

**Term Project**

Done in phases and iterations and is based on the THREE principles documented in the IBM white paper for modern software development –

Process-driven and UML based.

These are

1. Iterative and incremental development
2. Use-Case Oriented
3. Architecture-Centric -- diagram to code

## Part A –Requirements Analysis– Domain Problem Statement & Model (the Business case)

The process begins with a high-level problem statement which enables the development of different activity diagrams to model the problem domain operations of the organization. The activity diagrams show the requirements with respect to the scope and workflow of the system.

Gathering information is the key to success and getting the correct requirements. – Interviews (users & domain experts), questionnaires, documentation, identifying standards and terminologies in the problem domain.

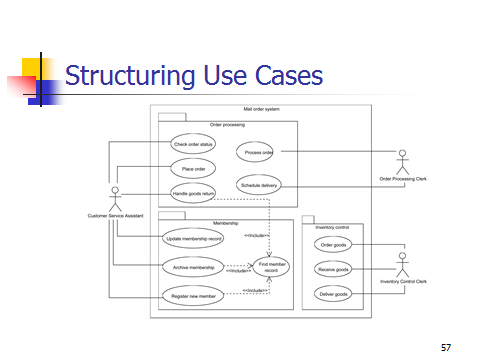
Part A of the term project is the Domain Business Model which includes

1. A problem statement at the workflow level of the system domain.
2. UML Activity diagrams illustrating the system scope and business workflows of the systems domain
3. Use Case List must contain only the goal Use Cases
4. Use Case Diagrams - May contain <<include>> & <<extend>> and generalization where appropriate. Must be fully labeled. A package approach is used – See diagram below. Follow guidelines given in text and/or Visual Paradigm for system boundary, connecting lines – bold & broken lines as well as directional arrows.
5. Use Case Description/Users Stories – provides key additional information on each goal Use Case contained in each of the packages identified. Use template – use Agile or UP templates)
6. Domain Class
7. Gantt Chart to be updated at each deliverable point

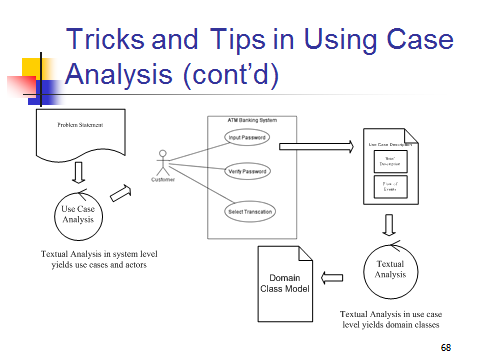
# Marks Breakdown

For Part A (10 of 30– Due Week 5)

|  |  |  |
| --- | --- | --- |
| Business Modeling – domain Problem | | |
| Subject | Mark |
| Problem statement ( Problem Needs and Subsystems + Vision Statement + System capabilities + Business benefits) – Chapter 1—See slides 20 & 22 | 0.5 |
| Workflows textual description + Activity Diagrams of workflows -- Chapter 2 See Slides 36 & 37 --Use Visual Paradigm tutorial to draw Workflows – every sub system needs a workflow diagram. | 2 |
| Use Cases ( Users + Users’ goals/Use Cases – 1 chart) + ( Use case + User stories – 1 chart) + 1 chart for each subsystem identified + Simple Use Case diagram for each subsystem - Chapter 3- ( Slides 10, 25-28, 31-33 -- etc See Visual Paradigm tutorial to draw Use cases | 4 |
| Domain Class diagram – Chapter 4 - See Visual Paradigm tutorial to draw domain Class Diagram - | 2.5 |
| Technology tools for Software Development ( List) ( Incomplete) | 0.5 |
| Project Plan ( Part A)- see tutorial in Additional material folder on ecentennial ( Incomplete) | 0.5 |
| Presentation - deductions |  |
| **Total** | **10** |



The software systems proposed will take on different levels of abstractions in a series of iterations with each iteration addressing the users requirements of each module within the package (subsystem) and that will finally be consolidated in a final project design specifications. Each iteration will reflect an n-tiered development – for example = the Presentation tier, Business Logic tier and the Data tier



## Part B – Software Design Specifications

Part B of the term project is high level design and designing user experience. It includes

1. Detail class diagram and Package diagrams (Complete)
2. Sequence Diagrams ( 3)
3. State Diagrams ( 3)
4. Project Plan Update (Incomplete)

In addition to the above you are also required to attach the revised copies of the previous document (corrected version as indicated from instructor feedback) of Parts A and B. The objective is to present a complete document incorporating all functional and analysis and design details for each iteration of your project.

MARKS Breakdown

For Part B (10 points) – due in Week 9

|  |  |
| --- | --- |
| Analysis & Design Specifications | |
| Subject | Mark |
| Detail Design class diagram - Package diagrams (Complete)  ( Annotate with CRC cards) Chapter 4, 12 & 13 | 4.5 |
| Sequence Diagrams ( From 3 use cases/methods) – Chapter 12 & 13 | 2.5 |
| State diagrams ( 3 separate Object states) – Chapter 4 | 2.5 |
| Project Plan – from Part A – include Part B | .5 |
| Presentation - deductions |  |
| **Total** | **10** |

In addition to the above you are also required to attach the revised copies of the previous document (corrected version as indicated from instructor feedback) of Parts A and B. The objective is to present a complete document incorporating all functional and analysis and design details for each iteration of your project.

## Part C – Design Specifications Cont’t

Part C of the term project is detailed design and realization for the top 2 use. It includes

1. High level architecture design (component diagram) (Multi-layer, MVC, etc.)
2. Mock-up UI
3. ERD model with database schema
4. Skeletal or stub code generated from Class Diagram using Visual Paradigm or VS-studio
5. Component & Deployment diagrams
6. Technologies for the Software Implemenation

In addition to the above you are also required to attach the revised copies of the previous document (corrected version as indicated from instructor feedback) of Parts A and B. The objective is to present a complete document incorporating all functional and analysis and design details for each iteration of your project.

MARKS Breakdown

For Part C (10 points) – due in Week 13

|  |  |
| --- | --- |
| Analysis & Design Specifications | |
| Subject | Mark |
| High level architecture design (Multi-layer, MVC class diagram etc. – from Part B) | 1.5 |
| Start with Chapter 8 charts – Slides 25 and 26 ( use cases – See Part A ). Mock-up UI ( use a wire-framing or VS-studio to do this) - include the functional sub-system chart from Chapter 8.-- 4-5 UI views – Include two report/form designs. | 3 |
| ERD Model & Database Schema( Visual Paradigm will generate from your domain class diagram show without details + Table descriptions using charts as in Chapter 9 -include field type and sizes for each field) – You may also use Visio to draw your ERD. ( Derive from your revised Domain class diagrams) | 2.5 |
| Component & Deployment – See Additional Material folder for illustrations | 1 |
| Skeletal or stub code generated from Class Diagram using Visual Paradigm or VS-studio -- Do not write the program logic | 1 |
| Final Technology tools for Software Implementation – complete the list that you began in Part A. | .5 |
| Project Plan -- this is to be completed as per the entire project. | 0.5 |
|  |  |
|  |  |
| Presentation - deductions |  |
| **Total** | **10** |