# PROJECT DELIVERABLES - OBJECT-ORIENTED METHODOLOGY

## Cover Page

Title
Group Members Names
Course Number and Name
School
Instructor Name
Date

Table of Contents
List of Figures
List of Tables

## Executive Summary

A maximum two page overall description of the entire project

## I. Problem Statement

*Environment Description*
*Organization Chart*
*Business Problems Challenges, Opportunities or Issues*
*Proposal: Vision, Goals, Objectives to Be Achieved*
*System Development Request/Need for System*
*Assumptions/Constraints*
*Preliminary*

## II. Preliminary Investigation and Feasibility Analysis

*Project Scope and Constraints*
*Costs and Benefits*
*Preliminary Effort/Time Estimate*
*Make vs. Buy (Outsource)*
*Project Plan (Gantt charts, WBS, metrics, etc.)*
    Project milestones and deliverable products
*Procurement*
*RFPs*
    Criteria for Vendors Selection
*Risk Assessment*
*Size Assessment*
*Acceptance Plan*

## III. Preliminary Investigation and Feasibility Analysis

*Intended Development Process*
*Project Workbook Outline*
*Resource Plan*
*Schedule*
*Release Plan*
*Quality Assurance Plan*
*Risk Management Plan*
*Reuse Plan*
*Metrics*
*Project Dependencies*
*Issues*

## IV. Requirements

*Use Case Model*
*Business Case*
*Grouping based on:*
- Functional
  - Input
  - Processing
  - Output
  - Data/Storage
  - Control
- Non-Functional
  - Performance (Timing)
  - Security
  - Reliability
  - Economics (Cost)
  - Conversion
- Other
  - Anticipated Growth
- ID for each requirement
- Priority (High, Medium, Low)
- Traceability Matrix

*Feasibility*
- Technical
- Economic (Financial)
- Operational
- Schedule

*Facts-Finding Results*
- Interviews Summaries
- Surveys/Questionnaires
- Observations

## V. Analysis

*Analysis Guidelines*
*Major Workflows*
*Subject Areas*
*Analysis Object Model*
*Analysis Scenarios*
*Analysis Object Integration Diagrams*
*Analysis State Models*
*Analysis Class Descriptions*

## VI. User Interface Model

*User Interface Guidelines*
*Screen Flows*
*Screen Layouts*
*User Interface Prototype*

## VII. Design

*Design Guidelines*
*System Architecture*
*Application Programming Interfaces (APIs)*
*Target Environment*
*Subsystem Model*
*Design Object Model*
*Design Scenarios*
*Design Object Interaction Diagrams*
*Design State Models*
*Design Class Descriptions*
*Alternatives*
*Rejected Design Alternatives*
*Cost-Benefit Matrix*
*Network Diagrams*
*Pseudocode*
*Input Screens (Samples)*
*Output Screens/Reports (Samples)*
*Dialog Flow Diagram*

## VIII. Implementation of the Design/Coding

*Development Environment*
*Coding Guidelines*
*Physical Packaging Plan*
*Development Environment*
*Source Code*
- Prototype
- Working Model
- Final Version

*Physical Packaging*
*User Support Materials*

## IX. Testing

*Test Plan*
- Test Cases/Scenarios
- Test Procedures
- Test Scripts
- Test Reports

*Environment*
- Test Hardware
- Test Software
- Other Test Equipment

*Personnel Needed*
*Levels of Testing*
- Unit
- Subsystem
- System
- User Acceptance
- Regression

## X. Installation/Delivery Plan

*Schedule*
*User Training/User Manual*
*Implementation Method*
- Abrupt/Direct Cutover
- Phased
- Pilot
- Parallel

## XI. Operation, Support, Maintenance

*Post Implementation Review*
*Maintenance Issues/User Support*
*Enhancements Handling*
*Delivery Methods*
- Release
- Versioning
- Incremental Patches

*Systems Support Procedures*
*Operational Results Report*
- Problems Encountered
  - Identify those that were anticipated and the unexpected.
  - Describe how these problems were dealt with, both the anticipated and real-time resolutions.
  - What might you have done differently in earlier phases to prevent these problems, or reduce their impact?
- System Benefits
  - Describe the expected benefits?
  - Were the actual benefits different from the expected and, if so, how?
  - What might have been differently in earlier phases to assure system benefits?
- Customer Satisfaction
  - Describe the level of customer satisfaction and any tangible measures of satisfaction with the system.
- Other Observations

*Conclusions and Lessons Learned*

## XII. Parallel Activities - Across the Project Life-Cycle*

*Project Management*
*Configuration Management*
*Quality Assurance*
*(Independent) Verification and Validation*
*Documentation*
- System
- User

* To be produced at the beginning of the project and updated at every major milestone!

## XIII. Appendices

*Glossary*
*Sample Documents*
*References*
*Figures/Tables*
*Historical Work Products*

***Work Product Definition***

*Description*
*Purpose*
*Participants*
*Timing*
*Techniques*
*Strengths*
*Weaknesses*
*Notation*
*Traceability*
*Advice & Guidance*
*Verification*
*Examples*
*References*
*Importance*

***Work Product Structure***

*Work Product Specific Part*

*Identifier*
*Date*
*Author*
*Owner*
*Status*
*Issues*
*Metrics*
*Traceability*
*History*

**PROJECT MANAGEMNET TECHNQIUES**

A Depth-First Approach to Software Development
Iterative and Incremental Development
Selecting Object-Oriented Modeling Tools
Prototyping as a Risk Management Technique

**DEVELOPMENT TECHNIQUES**

Performing a Domain Analysis
Getting Started with Semantic Networks
Building a Draft Object Model using Transcribe and Coverage
Wrapping with Non-OO Systems
Object-Oriented Implementation in a Non-OO Programming Language
Scenario-Driven Development
Design Patters
Providing Object Persistence
Interfacing to Relational Data
Visual Programming
Program Determination

**REUSE TECHNIQUES**

Reuse in General
Using Assets
Making Project Parts Reusable
Creating Truly Reusable Assets