

# The Student's Guide of Capstone Project Document for the FPT University Students

(Up dated under Decision No. 157, dated 09/04/2013)

--- \* ---

## Introduction words

**Capstone project's description:** Develop a significant software system or information system or embedded system designed for specific control functions within a larger system and embedded as part of a complete device often including hardware and mechanical parts (from now on commonly referred as "system"), employing knowledge gained from courses throughout the product including defining requirements, design, implementation, and quality assurance. Students must follow a suitable process model, pay attention to quality issues, and manage the project themselves, following all appropriate project management techniques. Success of the project is determined mostly by whether students have adequately delivered project documents on time and solved their customer's problem.

### *Sample deliverables:*

Students should deliver one or several iterations of a system, along with all artifacts appropriate to the process model they are using. These would likely include a project plan (perhaps updated regularly and containing cost estimations, risk analysis, division of the work into tasks, etc.), requirements (including use cases), architectural and design documents (including algorithms), test plans, source code, and an installable system.

### *Additional teaching considerations:*

- It is anticipated that this course will not have formal lectures, although students are expected to attend progress presentations by other groups.
- It is suggested that students be required to have a 'customer' for whom they are developing their system. This could be a company, a professor, or several people selected as representing people in the potential market. The objective of the project would be to solve the customer's problem, and the customer would therefore assist the instructor in evaluating the work.
- It is strongly suggested that students work in groups of at least four, and preferably five or six, on their capstone project. Strategies must be developed to handle situations where the contribution of team members is unequal.
- The System Development Life Cycle (SDLC) is a sequence of activities carried out by the developers (analysts, designers, users and others.)

**The entire System Development Life Cycle (SDLC)** can be divided into 7 phases:

1. Preliminary Investigation
2. Analysis
3. Design
4. Development
5. Testing
6. Implementation
7. Maintenance

This document focuses only on 4 phases from phase 2 Analysis to phase 5 Testing.

**Analysis:** A detailed document at the end of this phase is the Requirements Specification.

**Design:** A document produced at the end of this phase is called the Design Description.

**Development:** Actual implementing of the program is executed in this phase.

**Testing:** After the programs are tested individually, the system as a whole needs to be tested.

**Object Oriented Analysis (OOA)** is phase of the project during which a precise and concise model of the problem in terms of real world objects and concepts as understood by the user is developed. The analysis phase will identify entities. For example, in the case of a car dealership, the entities include Customers, Salesmen, Orders, and Cars. The analysis will also identify the relevant characteristics, interactions and relations between these entities.

**Object Oriented Design (OOD)** is phase in which products are organized as co-operative collections of objects. OOD looks at refinement of OOA and focuses on optimizing the solution provided while ensuring complete coverage of requirements. Several diagrams are created at the end of OOD phase. These diagrams can be classified into static and dynamic. The static diagrams deal with objects template and objects, where as the dynamic diagrams deal with interaction between the objects and their precise behavior. The algorithms should be showed to present the implementation in next phases

**Unified Modeling Language (UML):**

The language used to represent a model is called modeling language. Unified Modeling Language (UML) is used for modeling software in the Object Oriented Paradigm. A Model is a method to represent objects of a system.

**The IT Capstone Project Document is composed of 6 parts called Reports:**

- Report No.1: Introduction
- Report No.2: System Project Management Plan (SPMP)
- Report No.3: System Requirements Specifications (SRS)
- Report No.4: System Design Description (SDD)
- Report No.5: System Implementation & Test (SIT)
- Report No.6: System User's Manual (SUM)

**Structure and main content of this document**

**The FPT University Student's Guide to the Capstone Project Document includes two parts:**

**Part I:** Regulations of FPT University Academic Department.

**Part II:** Capstone Project Document template that includes 6 report document templates.

## **PART I: Capstone Project regulation**

FPT University students must follow the regulations presented here during the time doing the Capstone project. (In this document CP stands for Capstone project).

### **A. Common regulations**

- All students of FPT University have to do a CP in order to get a bachelor certificate.
- CP will be carried out in group. Each group includes 5 students (may be adjusted  $\pm 1$ ).
- The CP is worth 10 credits.
- The time allotted for doing and presenting the final CP is the entire semester.
- All CPs have to be presented to a defense committee assembled by the chancellor of FPT University advisor.
- Students must be present at all meetings with their supervisor according to a previously arranged schedule.
- English is official language for all documents.

### **B. Condition to apply for CP**

- Successfully complete on-the-job training (OJT).
- Have knowledge and skills about:
  - i. Software architecture and design
  - ii. IT project management
- Have knowledge and skills about at least 3 elective subjects from the same specialization area (IS, ES...) selected by student

### **C. Reports of CP**

The Capstone Project Document composes mainly following of 6 Reports:

- Report No.1: Introduction
- Report No.2: Project Management Plan (PMP)
- Report No.3: System Requirements Specifications (SRS)
- Report No.4: System Design Description (SDD)
- Report No.5: System Implementation & Test (SIT)
- Report No.6: System User's Manual (SUM)

### **D. Milestone table of CP implementation**

- At the beginning of week 2 submit: Report No.1 Introduction (duration: 1 week)
- At the beginning of week 3 submit: Report No.2 Project Management Plan (PMP) (duration: 1 week)
- At the end of week 4 submit: Report No.3 System Requirements Specifications (SRS) (duration: 2 weeks)
- At the end of week 6 submit: Report No.4 System Design Description (SDD) (duration: 2 weeks)
- At the end of week 10 submit: Report No.5 System Implementation & Test (STD) (duration: 4 weeks)
- At the beginning of week 12: show: The products (development (coding or/and hardware or/and components, algorithms) duration: 5 - 7 weeks) (continue to complete Report No.1-4; weekly show code to the Supervisor)

- At the beginning of week 14: Report No.6 System User's Manual (duration: 3 week) (continuing to implement the system and finish the system testing)
- In weeks 13+14 demonstrate the complete program, submit the final documentation, composed of Reports No. 1-6, and submit the products (including code or/and hardware, document, etc...))
- At the beginning of week 15 is final Capstone Project defense ceremony.

### **E. Evaluation of the Capstone Project**

- Based on the Group Mark (GM), each group's reports will be evaluated by its supervisor at the end of each stage. This evaluation is called continuous assessment.
- A Personal Mark in the range from 0%GM to + 120%GM will also be given for all Reports or all continuous assessment.
- Final continuous (on-going) assessment is collected from 6 reports with ratios as below:

Report	
a.Report No.1: Introduction	(5%)
b.Report No.2: Project Management Plan (PMP)	(10%)
c.Report No.3: System Requirements Specifications (SRS)	(15%)
d.Report No.4: System Design Description (SDD)	(25%)
e.Report No.5: System Implementation & Test (STD)	(40%)
f. Report No.6: System User's Manual (SUM)	(5%)

- A defense mark is given by each member of committee for each student after the final project defense. This personal mark is the average of the total marks given by each member of the committee in the "capstone project evaluation form".
- On-going assessment = 5% report 1 (intro) + 10% report 2 (plan) + 15% report 3 (SRS) + 25% report 4 (Design) + 40% report 5 (Implementation & Testing) + 5% report 6 (UM)
- Project Defense = 40% Design + 40% Product + 20% Presentation
- Final Project Result = On-going assessment (6 Reports) x 50% + Defense Mark (by Committee) x 50%
- All marks are converted to the scale of 0-10, precise to one place after the decimal. (Example: 6.7 marks)
- Completion criteria
  - a. On-going Criteria: Yes (after 6 weeks: 4 reports  $\geq 5$ ) and
  - b. On-going assessment  $\geq 5/10$  and
  - c. Project Defense  $\geq 5/10$

## **PART II: Capstone Project Document template**

### **A. Table of contents for the capstone project document**

Students must include table of contents with their results report. The following is a template, provided for reference.

- Front page (see next section)
- Table of contents
- Index [Optional]
- Report No1: Introduction
- Report No.2: Project Management Plan (PMP)
- Report No.3: System Requirements Specification (SRS)
- Report No.4: System Design Description (SDD)
- Report No.5: System Implementation & Test (SIT)
- Report No.6: System User's Manual (SUM)
- Appendix:

. Glossary [Optional]

*<Place all definitions or abbreviation used in this document >*

. References [Optional]

*<Place all referenced materials used in this document >*

. Others [Optional]

### **B. Front page template**

Next page is the template of front page of CapStone project document. Groups must follow this template. In this template: Replace any text in <> with your text. When you are done, there should be no <> or text surrounded by <> in this front page. Remember, the purpose of this page acts as a cover; it will not be graded.

# FPT UNIVERSITY

## Capstone Project Document

---

[Project name]

<Group Name>	
<b>Group Members</b>	<Member name><RollNo><Student code > <Member name><RollNo>< Student code > <Member name><RollNo>< Student code > <Member name><RollNo>< Student code > <Member name><RollNo>< Student code >
<b>Supervisor</b>	
<b>Ext Supervisor</b>	
<b>Capstone Project code</b>	

- Hanoi, <month>/<year> -

*<The Table of Contents goes here>*

## **C. Capstone Project Document Template**

### **Report No.1: Introduction**

This part is about the group of developers, the initial idea of the group's Capstone project, overview of similar existing solutions, overview of existing methods, business outline, limitations of the existing system. From that point, you will advance your knowledge/skill levels in the topic you have chosen, and benefits of expected system.

*<Introduction to this document, how it is organized, and what it contains>*

### **Report No.2: Project Management Plan (PMP)**

#### **2.1. Problem Definition**

2.1.1. Name of this CapStone Project

2.1.2. Problem Abstract

*<Write down the problem abstract. This can be paraphrased from the Customer's Requirement>*

2.1.3. Project Overview

2.1.3.1. The Current System

*<Describe the current system/situation. This can be paraphrased from the Customer's Requirements>*

2.1.3.2. The Proposed System

*<Describe the system under development. This can be paraphrased from the Customer's Requirements>*

2.1.3.3. Boundaries of the System

*<List the scope/boundaries of the system under development. This can be paraphrased from the Customer's Requirements>*

2.1.3.4. Development Environment

*<Describe the environment for system under development. Include software and hardware requirements>*

#### **2.2. Project organization**

2.2.1. System Process Model

2.2.2. Roles and Responsibilities

2.2.3. Tools and Techniques

#### **2.3. Project management plan**

2.3.1. Tasks:

2.3.1.n Task-n:

- Description
- Deliverables
- Resources Needed
- Dependencies and Constraints
- Risks

2.3.2. Task Sheet: Assignments and Timetable

2.3.3. All Meeting Minutes

*<Put all your-team's meeting minutes here as an appendix >*

#### **2.4. Convention Rules**

#### **2.5. Other material (if any)**



## Report No.3: System Requirements Specifications (SRS)

### 3.1. User Requirement Specification

*<Summarize the customer requirements in a compact form>*

### 3.2. System Requirement Specification (Specific Requirements)

*<Summarize the system requirements in a compact form>*

#### 3.2.1. External Interface Requirements

##### 3.2.1.1. User Interfaces

##### 3.2.1.2. Hardware Interfaces (if any)

##### 3.2.1.3. System Interfaces (if any)

##### 3.2.1.4. Communications Protocol

#### 3.2.2. Functional Requirements

##### 3.2.2.n. Functional Requirement n

##### 3.2.2.n.1. Use Case-n Diagram (otherwise use another model)

##### 3.2.2.n.2 .Use Case-n Specification

*<Write down all non-trivial use cases. This should reflect what you get when your team does the system analysis. Use the template to write a detailed specification for use cases>*

#### USE CASE-n SPECIFICATION

Use-case No.	<UC001>	Use-case Version	<1.0>
Use-case Name	<Name>		
Author	<Members>		
Date	Dd/mm/yyyy	Priority	<High/Normal/Low>

**Actor:**

*<Lit all actors>*

**Summary:**

*<Briefly describe the used case >*

**Goal:**

*<Briefly describe the goal of used case >*

**Triggers**

*<What does lead in using this case?>*

**Preconditions:**

*<List the required pre-conditions for using this case>*

**Post Conditions:**

*<List the required post-conditions for using this case>*

**Main Success Scenario:**

*<List the main steps for using this case to reach the goal successfully >*

**Alternative Scenario:**

*<List other steps for using this case to reach the goal in some alternative conditions >*

**Exceptions:**

*<List exceptions of this use case >*

**Relationships:**

*<List the relationships that use case relates to>*

**Business Rules:**

*<Any concern about the business>*

**3.2.3. Non-Functional Requirements**

3.2.3.1 Reliability

3.2.3.2 Availability

3.2.3.3 Security

3.2.3.4 Maintainability

3.2.3.5 Portability

3.2.3.6 Performance

**3.3. Entity Relationship Diagram or Data Structures (if any)**

*<Provide the ERD Diagram for the system here. If your team uses a file or in-memory storage facility instead of a database, replace this section by 'Data Structures'. Note, use only ERD or Data Structures>*

**3.4. Other material (if any)**

## **Report No.4: System Design Description (SDD)**

### **4.1. Design Overview**

### **4.2. System Architectural Design**

- 4.2.1 Choice of System Architecture
- 4.2.2 Discussion of Alternative Designs
- 4.2.3 Description of System Interface

### **4.3. Component Diagram**

### **4.4. Detailed Description of Components**

#### **4.4.1. CRC Cards (Class-Responsibility-Collaborators) (if any)**

##### **4.4.1.n. Component-n**

##### **4.4.1.n.1. Class Diagram**

##### **4.4.2.n.2. Class Diagram Explanation**

*<Provide a brief explanation of the class diagram above. You do not need to explain "obvious" parts of your class diagram>*

##### **4.4.3.n.3. Algorithms of important methods in each class, specified in pseudo code or by Flow-Chart**

### **4.5. Sequence Diagram**

### **4.6. User Interface Design/ Hardware Interface Design (if any)**

#### **4.6.1 Description of the User Interface**

##### **4.12.1.1 Screen Images/ Hardware Description**

##### **4.12.1.2 Objects and Actions/ Component**

### **4.7. Database Design or Data Structures or algorithms**

*<Provide the detailed database design for the system here. If your team uses a file or in-memory storage facility instead of database, remove this section; use the 'Data Structures' section. >*

### **4.8. Other material (if any)**

## **Report No.5: System Implementation & Test (SIT)**

### **5.1. Introduction**

5.1.1 System Overview

5.1.2 Test Approach

### **5.2. Database Relationship Diagrams/Hardware Charts (if any)**

### **5.3. Screen shots/ Performance measures/...**

### **5.4. Test Plan**

5.4.1 Features to be tested

5.4.2 Features not to be tested

5.4.3 Testing Tools and Environment

### **5.5. Test Cases**

5.5.*n* Case-*n*

5.5.*n*.1 Purpose

5.5.*n*.2 Inputs

5.5.*n*.3 Expected Outputs & Pass/Fail criteria

5.5.*n*.4 Test Procedure

### **5.6. Checklists**

5.6.1. Checklist of Validation

*<Put the checklist here. Describe how it is used and the resulted checklist>*

5.6.2. Submission Checklist

*<Put the checklist here. Describe how it is used and the resulted checklist>*

### **5.7. Other material (if any) (including appendix A)**

APPENDIX A. TEST LOGS

A.*n* Log for test *n*

A.*n*.1 Test Results

A.*n*.2 Incident Report

## **Report No.6: System User's Manual**

### **6.1. Installation Guide**

### **6.2. User's Guide**

### **6.3. Other [Optional]**