Teacher Guides Document

Information System Programming Project (ISP392) Topic
Application Development Project (SWP391) Topic

------- * -------

Table of Contents

I. Introduction

The project lifecycle is divided into 3 phases

- ***Initiation***: this phase includes one iteration in one week (6 contact slots)
    - Build team,
    - Prepare working environment
    - Get familiar with the working environment & tracking tools
    - Prepare/define overall requirement & design for the whole project
    - Develop POC (Proof Of Concept) which include writing detailed requirement, code design, & code-review-test-integrate, each member one screen/function.
- ***Construction***: This phase includes 3 development iterations; each iteration would last for two weeks (12 contact slots) with below main tasks to build the project assigned product.
    - Build the software package
    - Prepare the documentation as guided by the teacher,
    - Submit & demonstrate the working result to the teacher for evaluating.
- ***Closing***: complete the software package and related documents, the phase lasts for three weeks (18 contact slots), and would be completed with student teams' presentation as final evaluation.

II. Implementation Guides

The teachers are recommended to study the Student Guides Document also to imagine what and how the student to understand more about their work and can provide relevant guidance to the students during the topic delivering.

1. Initiation Phase

Introduce the topic syllabus & detailed evaluations scheme

Arrange the working teams & assign the project topic.

- Each team includes 4-6 students, select a team leader themselves
- The project working topic (assignment) is defined/verified by the teacher.
- The project topic can be taken from the project bank, defined by the teacher or registered by the student team. And the scope needs to be big enough for student practices (1800-3600 LOC, in which each software function equivalence 60, 120 or 240 LOC depending on its complexity of simple, medium or complex)

Assign & explain the POC requirements to the class

*(Refer the **Iteration 1_POC Requirements** document)*

Guide the teams on creating Git environment & logging issues to the GitLab.

Work with teams to adjust their team assignment, guide them to prepare iteration submit

Guide & request teams to adjust the software functional specification & software design to follow during the next Construction Phase

2. Construction Phase

Work with teams to adjust their team assignment for each iteration at the iteration begin.

Guide the team to prepare the functional requirement specification, code designs, follow the coding conventions, versioning controlling during their working progress.

To complete developing for a function/screen, below tasks need to be implemented

- Write the function requirement specification
- Prepare the code design document
- Code & unit test for each screen/function
- Integrate with other functions/screens

For the code designs, students may not have to prepare the code design for all the functions/screens. In an iteration, each student need can prepare only code design for one of his/her assigned functions/screens.

During the construction phase, the teacher continue guiding the teams to log & track requirements, tasks, defects, issues,.. as well as code version controlling in the GitLab.

3. Closing Phase

Guide the teams to complete/upgrade the project documents and the source codes for the whole project as well as prepare extra guide/demo materials (Project Tracking, Final Release Document, Team-Demo Video, and Presentation Slide).

Arrange presentation sessions for the team to introduce about their working results for the whole project in the slots 55-60.

III. Extra Guides & Appendixes

1. Training Discussions

The teacher can guide the student teams to discuss and share the individual information, updates, or initiatives via the constructive questions as listed below.

Of course, that the teacher can adjust and/or add more discussion items to fit with the class context and the students' background/selection.

In some slot, the teacher might need to provide the explanation for the submits (assignment) that student need to prepare for each iteration. The content of those assignments can be as described blue highlighted rows

Weekly Report	What are your working progress & next working plan? - What have you done for the project last week? - What will you do this week for the project? - Do you have any questions/issues/difficulties (requirements, technical, team,..) ?	7, 10, 13, 16, 19, 22, 25, 28
Daily Report	What are your working progress & next working plan? - What have you done for the project from last training session? - What will you do from today till the start of next training session? - Do you have any questions/issues/difficulties (requirements, technical, team,..) ?	Beginning slot in the training day
Git & GitLab	What are the Git commands that you would use to manage source codes in your project? For each command, list out the name, purpose and the syntax to call.	S2
IterX Plan	Your plan for the next iteration? - What are the functions/screens that you would be incharged? - Is there any unclear things that need to clarify or concerning you?	S3, 9, 15, 21

Iter1 Plan Submit	Teams to prepare and submit the plan for the iteration 1 (iter1) which include following items: 1. Project Tracking file: following the attached template, with the following sheets filled (as guided in the sheets) - Product: include all the functions/screens of the target application - Iter1: up-to-date information/status, 2. SRS Document: following the attached template, with the following sections filled - Record of Changes, - Section I – Overview 3. POC Source Codes & SQL File (to create database and demonstrating data)	S3
Iteration Tasks	What are the deadlines for the tasks to complete each screen/function that you are in-charged in the iter2? Note that there are below specific tasks to complete each of the screen/function - Write functional requirement specification - Prepare code design document - Code & Unit test - Integrate with other screens/functions	S4
Code Design	To implement your assigned screens/functions what are the code modules (classes/files, methods) you would need? - What are the code modules you would have? - What are the purposes of each module?	S5
Package Design	How would you organize your source codes into packages/folders? - What are the packages/folders you would have? - What is the role/description of each package/folder? - What are the naming convension you would have for the classes/files in each package/folder?	S6
Testing & Integrating	How do you do the testing & integrating you're assigned screens/functions? - How to you do the unit testing? - How do you integrate with other screens/functions? - What are common issues you have discovered after testing/integrating?	S8

IterX Submit	Teams to complete release package for the **IterX** which include following items: 1. Updated Project Tracking using the provided template with the following sheets filled (as guided in the sheets) - Project: update the information as guided - **IterX**: up-to-date information/status for the **IterX** - **IterY**: plan & up-to-date information/status for the **IterY** functions/screens 2. Updated SRS Document using the provided template with the following sections filled/updated: - Record of changes: update for the changes beneath - Section I - Overview: provide adjustments/updates if needed - Section II - Functional Requirements: add the functional requirements for the **IterX**'s features & screens/functions and adjustments for the other iterations if needed 3. Updated SDS Document using the provided template with the following sections filled - Record of changes: update for the changes beneath - Section I - Overview: provide adjustments/updates if needed - Section II - Code Designs: add the code designs for the **IterX**'s features & screens/functions and adjustments for the other iterations if needed 4. Source codes & SQL file (to create database structure & demonstrating data): a text file that contains link to the tagged codes (from the iteration's branch) in GitLab, provided as description for this assignment submit 5. Link to ONE demo video where each team member record his/her demonstration on the functions/screens he/she has completed (note to mention clearly who is demonstrating and keep total duration of the video less than or equal 20 minutes) 6. Issues Report: include all the issues (requirements, tasks, defects, issues, Q&A) of the iteration using the provided template. You can filter & export those from the GitLab + fill the column Functions/Screens are filled manually with the related function(s) or screen(s) for each issue.	S9, 15, 21
IterX Status	Which function/screen have you been working on in the last iteration? - Among those functions, which function were completed and ready for demonstration? - List out the pending issues with that function (integrating with others, use common code modules, defects/bugs,…)	S10, 16, 22

IterX Demo - Gy	What are the most 3 significant issues from Gy that you might have with your own works? Gy team members to list out the updates/changes (as results from the team's demo) that you would have to make on the demonstrated codes to have better results in the final iteration	S11-14, S17-20, S23-26
Project Retrospective	What have you achieved after the project? what are the experiences/issues/difficulties that you've faced in the project? What have you learnt and apply in your future projects/works?	S27
Final Submit	Teams to record a video to demo/introduce about the completed functionalities in each workflow, prepare & submit the final project package (in a zip file) which include: - Final Release Document: following the attached template - SRS, SDS Documents: include updates & specifications for all the functions/screens + records of changes (in the page 2) - Project Tracking: update the sheet Project as guided + update the sheet Iter4 to have up-to-date information/status for Iter4 - Issues Report: team's issues/tasks extracted from the GitLab export + add Related Screen/Function information - Presentation slide Some other notes: - The demo/introduction video's length must be under 30 minutes where: o A team representative to show brief introduction about the project topic and team's assignemnts for the detailed shows below o Each member (mentioned names clearly) shows the details of each workflow (or the features/functions related to each user classes) - The presentation slide need to include below contents (max 15 slides) o Project Members, Product Brief Introduction o Use case diagram for each Role/Actor o Application Design: code package diagram, database schema o Screen Flow(s) and UI design (used theme + sample screen shots for Admin and Web module) o Project Results: Done, Not done, Lessons Learnt	S27

Final Presentation	What are the 3 good things from the presenting team that you can learn & apply? List out 2 things that you think you can do better?	S28-30
Work Dependencies	What are predecessor(s) and successor(s) of your assigned screens/functions? What database tables each of your assigned screens/functions depends on (have query, insert, update or delete transactions to)?	
Implementing Progress	What are status of each function/screen that you are incharged in the Iter2? What are the tasks needed to complete those in the iter2? Any unclear things or emerging issues/conflicts?	

2. Student Evaluations

The teacher would track & evaluate the student works in the project using the templates as defined in the ***Template1_Student List*** file. To have team & LOC evaluation filled in this file, the teacher would use the other two templates for detailed evaluation

- Template2_Evaluation-Team: for team evaluations
- Template3_Evaluation-LOC: for team members' LOC evaluations