Schoodle

Software Development Plan (Small Project)

Version 0.1

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 21/10/2019 | 0.1 | Initialize project base | nhgia-apcs1 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[1. Introduction 3](#_Toc307271011)

[1.1 Purpose 3](#_Toc307271012)

[1.2 Scope 3](#_Toc307271013)

[1.3 Overview 3](#_Toc307271014)

[2. Project Overview 3](#_Toc307271015)

[2.1 Project Purpose, Scope, and Objectives 3](#_Toc307271016)

[2.2 Assumptions and Constraints 3](#_Toc307271017)

[2.3 Project Deliverables 3](#_Toc307271018)

[3. Project Organization 3](#_Toc307271019)

[3.1 Organizational Structure 3](#_Toc307271020)

[3.2 Roles and Responsibilities 3](#_Toc307271021)

[4. Management Process 3](#_Toc307271022)

[4.1 Project Estimates 3](#_Toc307271023)

[4.2 Project Plan 3](#_Toc307271024)

[4.2.1 Phase Plan 3](#_Toc307271025)

[4.2.2 Iteration Objectives 3](#_Toc307271026)

[4.2.3 Releases 3](#_Toc307271027)

[4.2.4 Project Schedule 3](#_Toc307271028)

[4.2.5 Project Resourcing 3](#_Toc307271029)

[4.3 Project Monitoring and Control 3](#_Toc307271030)

[4.3.1 Requirements Management 3](#_Toc307271031)

[4.3.2 Reporting and Measurement 3](#_Toc307271032)

[4.3.3 Risk Management 3](#_Toc307271033)

[4.3.4 Configuration Management 3](#_Toc307271034)

Software Development Plan (Small Project)

# 

# Introduction

This documentation is provided full-detailed about the new, fresh and convenient way of student management system called Schoodle.

Our targeted customer is a small-unit of school (e.g. academy, institute, …). Which can be used by Admin, Academic sections (Staff), Teachers, Students, Parents.

Schoodle means the Moodle for school.

## Purpose

**The Software Development Plan aims to keep a good connection between students, staffs, and teachers/lecturer by using Schoodle – A Student Management Software. This plan contains all detail and information to create and deploy this Student Management Software.**

**The plan describes in a particular way approaching the development of Schoodle. All the research are collected and carefully selected for making this plan.**

The following people use the *Software Development Plan*:

* The **project manager**: The PJ must take responsibility of the project
  + Assign tasks to the team member
  + Push the growth of the project to the next level.
  + Control resources
  + Tracking and backtracking tasks.
* **Project team members:** 
  + Follow the schedule
  + Understand things which other members are doing it.

## Scope

This Software Development Plan establishes the schedule for software implementation, qualification, and deployment of Schoodle. The Schoodle is being developed under direction of the Course TA.

## Overview

This *Software Development Plan* contains the following information:

Project Overview — The Schoodle Project involves the development and deployment for the Student Management Software, written for mobile devices.

Project Organization — Agile Model

# Project Overview

## Project Purpose, Scope, and Objectives

The Schoodle Project wants to produce an easy way for students, staffs, and lecturers to receive and process the information from the school effectively. From that, time saved, life standard increases, good results are made.

Our team create an mobile application to serve our purpose. A complete Student Management System including:

* Courses Enrollment
* Schedule Management
* Post/View/Update/Delete GPA
* Materials Sharing
* Making Notifications
* Financial Report

## Assumptions and Constraints

The schedules, estimation and costs are made base on the assumptions list below. Team member and Lecturer/TAs of CS300 have the right to re-estimate the schedule and the cost if there are mistakes in the assumptions

* Baselined Inputs
  + Schedule A
  + Related Work B
* Methodology
  + The project will be done by using tools and frameworks prescribed by the team leader.
  + Deliverables prescribed by the methodology are defined in section (?).
* Performance
  + Traditional style of measuring performance, where users take an action and wait for a response. Measuring the time it take to complete the action
  + The performance measurement will be mentioned in section (?)
* Dataset
  + All the data is produced manually by team members
  + High practicality
* Budget
  + 2 months time
  + 3,600,000 VND/each person
* Staff and Equipment
  + Research to adapt project requirement (learning new framework, tools)
  + Have all machine with 3 OS: Windows, Linux, Macintosh

## Project Deliverables

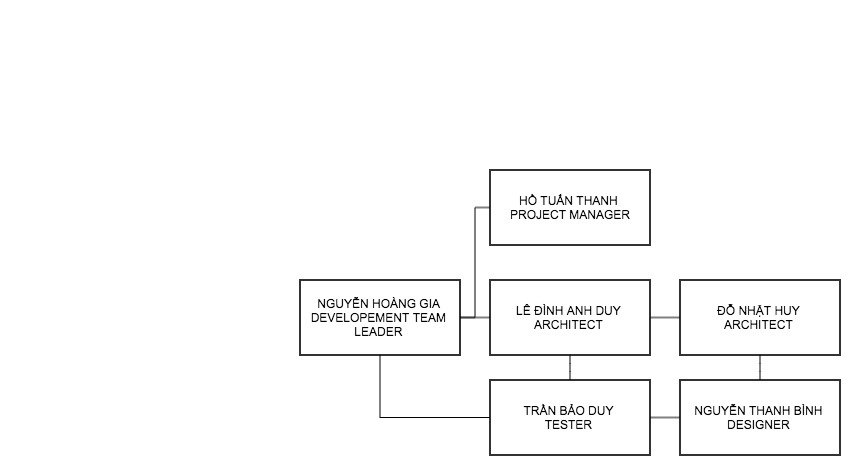
[A list of the artifacts to be created during the project, including target delivery dates. The text below is provided as an example.]

The following deliverables will be produced during the project:

* Student Management System
* Student Management Mobile Application
* Data model
* Design model
* Database Design
* Project Vision Documentation
* Test Package

# Project Organization

## Organizational Structure



## Roles and Responsibilities

[Identify the project organizational units that will be responsible for each of the disciplines, workflow details, and supporting processes. The text below is provided as an example.]

The following table shows the roles represented in the project diagram above and each of their main

responsibilities:

|  |  |
| --- | --- |
| Project Manager | The Project Manager allocates resources, shapes priorities, coordinates interactions with the customers and users, and generally tries to keep the project team focused on the right goal. The Project Manager also establishes a set of practices that ensure the integrity and quality of project artifacts. |
| Team Leader |  |
| Architect | The Architect leads and coordinates technical activities and artifacts throughout the project. The Architect establishes the overall structure for each architectural view: the decomposition of the view, the grouping of elements, and the interfaces between these major groupings. Thus, in contrast with the other workers, the Architect's view is one of breadth, as opposed to depth. |
| Tester | The designer defines the responsibilities, operations, attributes, and relationships of one or several classes and determines how they should be adjusted to the implementation environment. In addition, the designer may have responsibility for one or more design packages or design subsystems, including any classes owned by the packages or subsystems. |
| Designer | The creative designer leads and coordinates the prototyping and design of the Web interface, by capturing requirements on the Web interface, including usability requirements, building Web page prototypes, involving other stakeholders of the Web interface, such as end-users, in usability reviews and use testing sessions, and reviewing and providing the appropriate feedback on the final implementation of the Web interface (as created by other developers, i.e. designers and implementers). |

# Management Process

## Project Estimates

[Provide the estimated cost and schedule for the project, as well as the basis for those estimates, and the points and circumstances in the project when re-estimation will occur.]

This project will take 6 weeks. Initial estimates of each s

## Project Plan

[This section contains the schedule and resources for the project.]

### Phase Plan

[Include the following:

 Work Breakdown Structure (WBS) — optional for small projects

 a timeline or Gantt chart showing the allocation of time to the project phases or iterations

 identify major milestones with their achievement criteria

Define any important release points and demos.]

### Iteration Objectives

[List the objectives to be accomplished for each of the iterations.]

### Releases

[A brief description of each software release and whether it’s demo, beta, and so on.]

### Project Schedule

[Diagrams or tables showing target dates for completion of iterations and phases, release points, demos, and other milestones.]

### Project Resourcing

 [Identify the numbers and type of staff required here, including any special skills or experience, scheduled by project phase or iteration.

 Describe how you will approach finding and acquiring the staff needed for the project.

 List any special training project team members will require, with target dates for when this training should be completed.]

## Project Monitoring and Control

 [The following is a checklist of items to consider:

* Requirements Management : Specify the information and control mechanisms which will be collected and used for measuring, reporting, and controlling changes to the product requirements.
* Reporting and Measurement: Describe internal and external reports to be generated, and the frequency and distribution of publication. Specify which metrics should be collected and why.
* Risk Management: Describe the approach that will be used to identify, analyze, prioritize, monitor and mitigate risks. Include a list of risks and their current status.
* Project Close-out: Describe the activities for the orderly completion of the project, including staff reassignment, archiving of project materials, post-mortem debriefings and reports, and so forth.
* Configuration Management: Describe the process by which problems and changes are submitted, reviewed, and dispositioned. Describe how project or product artifacts are to be named, marked, and numbered, including hardware, system software, Commercial-Off-The-Shelf (COTS), plans, models, components, test software, results and data, executables, and so on. Describe retention policies, and the back-up, disaster, and recovery plans. Also describe how the media is to be retained—online, offline, media type, and format.

The text that follows is provided as an example.]

### Requirements Management

The requirements for this system are captured in the Vision document. Requested changes to requirements are captured in Change Requests, and are approved as part of the Configuration Management process.

### Reporting and Measurement

Updated cost and schedule estimates, and metrics summary reports, will be generated at the end of each iteration.

The Minimal Set of Metrics, as described in the RUP [Guidelines: Metrics](..\..\..\process\modguide\md_metri.htm), will be gathered on a weekly basis. These include:

Earned value for completed tasks. This is used to re-estimate the schedule and budget for the remainder of the project, and/or to identify need for scope changes.

Total defects open and closed – shown as a trend graph. This is used to help estimate the effort remaining to correct defects.

Acceptance test cases passing – shown as a trend graph. This is used to demonstrate progress to stakeholders.

In addition, overall costs will be monitored against the project budget.

### Risk Management

Risks will be identified in Inception Phase using the steps identified in the RUP for Small Projects activity “Identify and Assess Risks”. Project risk is evaluated at least once per iteration and documented in this table. The risks of the greatest magnitude are listed first in the table.

|  |  |  |
| --- | --- | --- |
| **Risk Ranking (High, Medium, Low)** | **Risk Description and Impact** | **Mitigation Strategy and/or Contingency Plan** |
|  |  |  |

### Configuration Management

Appropriate tools will be selected which provide a database of Change Requests and a controlled versioned repository of project artifacts.

All source code, test scripts, and data files are included in baselines. Documentation related to the source code is also included in the baseline, such as design documentation. All customer deliverable artifacts are included in the final baseline of the iteration, including executables.