UPCC
Software Architecture

Submitted to:

Asst. Prof. Ma. Rowena C. Solamo
Faculty Member
Department of Computer Science
College of Engineering
University of the Philippines, Diliman

Submitted by:
Abaja, James Gabriel
Cruz, Rayven Ely
Lim, Ciana

In partial fulfillment of Academic Requirements
for the course
CS 192 Software Engineering II
of the
1st Semester, AY 2017-2018

Revision Control

History Revision:

Revision Date	***Person Responsible***	***Version Number***	***Modification***
11/24/17	Ciana Lim	1.0	Initial Document. Added the Purpose, Audience, Description of the document, the Software Architecture Diagram, and the User Interface Design Classes.
11/28/17	James Abaja	2.0	Added the Data Design Classes and their descriptions.
11/29/17	Rayven Ely Cruz	3.0	Added Business Logic Classes and their descriptions.
02/08/18	Ciana Lim	4.0	Edited the Audience and Description to fit the current system. Added activity_select_curriculum, SelectCurriculum, and UPCC classes. Modified the descriptions for Curriculum and Subject classes.
02/08/18	Rayven Ely Cruz	5.0	Updated InputSubjects controller and activity_input_subject UI.
02/09/18	James Abaja	5.1	Updated Software Architecture Diagram; changed the <curriculum> FileClass into a DBClass.
02/23/18	Ciana Lim	6.0	Updated Software Architecture Diagram and classes to reflect Sprint 2.
03/09/18	Ciana Lim	7.0	Updated Software Architecture Diagram and classes to reflect Sprint 3.
03/23/18	Ciana Lim	8.0	Updated Software Architecture Diagram and classes to reflect Sprint 4.

Purpose:

This document’s purpose is to show the software architecture of the UPCC system.

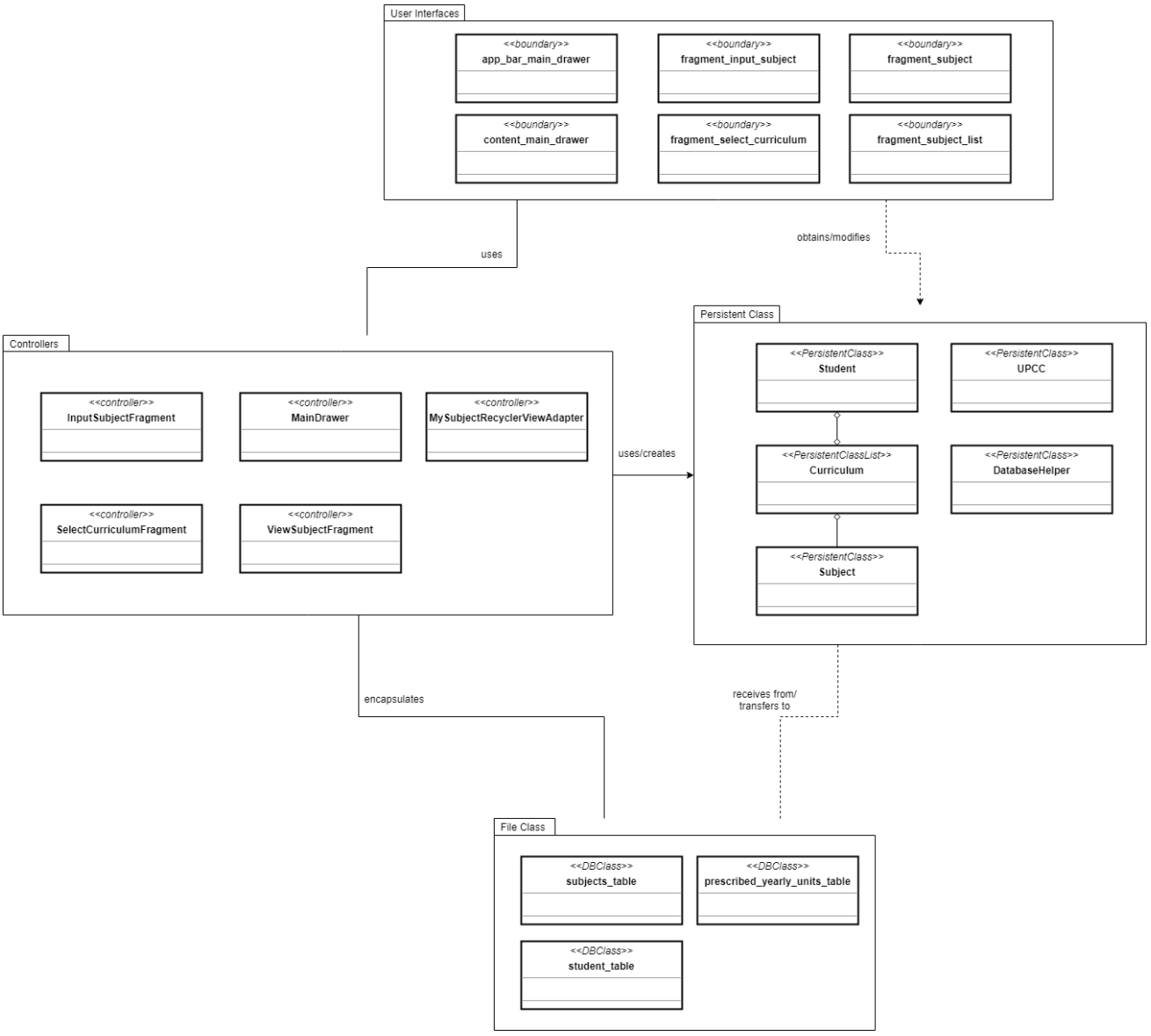
Audience:

The target audience of this document are the developers who are interested in continuing and extending the project, users who are interested in how the system works, and the professor handling the course.

System Name: UPCC

Description: The UPCC system allows UP Diliman students under the BS Computer Science (BS CS) program to select the curriculum they're currently following. They can also input the subjects they've already passed, and view the subjects they can take afterwards. The system also allows administrators to manage the curriculums by adding, editing, and deleting curriculums, and subjects.

Software Architecture Diagram:



User Interface Design Classes:

Screen Name	Description
app_bar_main_drawer	This interface represents the toolbar of the application. One can see the title of the screen, help menu, and the navigation button.
content_main_drawer	This interface represents the navigation drawer of the application. It can navigate to fragment_input_subject and fragment_select_curriculum.
fragment_input_subject	This interface allows the Student user to mark the subjects from the list. The marked subjects will then be sent to InputSubjectFragment which will update the list of subjects passed by the Student user.
fragment_select_curriculum	This interface allows the Student user to select a curriculum that is available from the database.
fragment_subject	This interface shows the layout of a subject.
fragment_subject_list	This interface shows the layout of the subject list.

Business Logic Classes:

Control	Description
InputSubjectFragment	This controller is in charge of marking and unmarking the subjects that the Student user input.
SelectCurriculumFragment	This controller is in charge of selecting the Student's chosen curriculum, and passing the selected curriculum's data to the InputSubjects controller.
MainDrawer	This controller is in charge of switching between fragments and handling the data being passed between fragments.
ViewSubjectFragment	This controller is in charge of showing the subjects that the Student can take.
MySubjectRecyclerViewAdapter	This controller is in charge of fixing the data of a subject so that it can be shown.

Data Design Classes:

Class Name	Description
Curriculum	The Curriculum class defines the structure of a curriculum to be used inside the system. It contains the curriculum name, and list of subjects under the curriculum, defined under the Subject class. It also has methods to get the curriculum's name, to add a subject to the curriculum, to print details about the curriculum and its subjects, to return the curriculum's subjects in an array of Subject objects, and to check if there are subjects listed under the curriculum.
Student	The Student class defines the data that are associated with the student user. It holds the curriculum being used, and its completed subjects.
Subject	The Subject class contains the information of a subject inside the Curriculum class such as the subject name, subject description, prerequisites and corequisites, etc. There are also methods to get the curriculum's name where the subject is listed, the subject's name and description, the number of units, and its rules.
UPCC	The UPCC class maps the column number of the subjects_table from the database to a variable (which is also the column's name). It provides modularity when a controller wants to access the specific columns of the database's subjects_table. It also maps column tables to the prescribed_yearly_units_table, and constants that mean the standing of a student.
DatabaseHelper	The DatabaseHelper class holds the database tables together.
subjects_table	The subjects_table Database Class holds all the subjects of all the curriculums that are available in the app.
student_table	The student table Database Class holds all the subjects that the student has already passed in his/her selected curriculum.
prescribed_yearly_units_table	The prescribed_yearly_units table Database Class holds the yearly units that each curriculum needs for a student to reach its desired standing.