Schedule Optimizer
Use Case Specification

Submitted to:

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

Submitted by:
Cavan, Antonio D.
Ramos, John Matthew G.
See, Engelberg Jeremy T.

In partial fulfillment of academic requirements
for the course
CS 191 Software Engineering I
of the
1st Semester, AY 2019-2020

Unique Reference:

The documents are stored in the https://github.com/DarkLuminosity/Schedule-Optimizer/tree/master/02-Requirements%20Engineering/Project%20Deliverables referenced with ScheduleOptimizer-Use Case 1-Indicate Preference.pdf

Document Purpose:

The purpose of this document is to identify and present the different scenarios that may occur within the use case presented.

Target Audience:

The target audience would be University of the Philippines-Diliman undergraduate students.

Revision Control:

Revision Date	*Person Responsible*	*Version Number*	*Modification*
09/19/19	Engelberg See	1.0	Prepare Initial Document.
09/21/19	Antonio Cavan	2.0	Added purpose and Use case description
09/22/19	Antonio Cavan	3.0	Added flowchart

***Use-Case Name*:** Indicate Preference

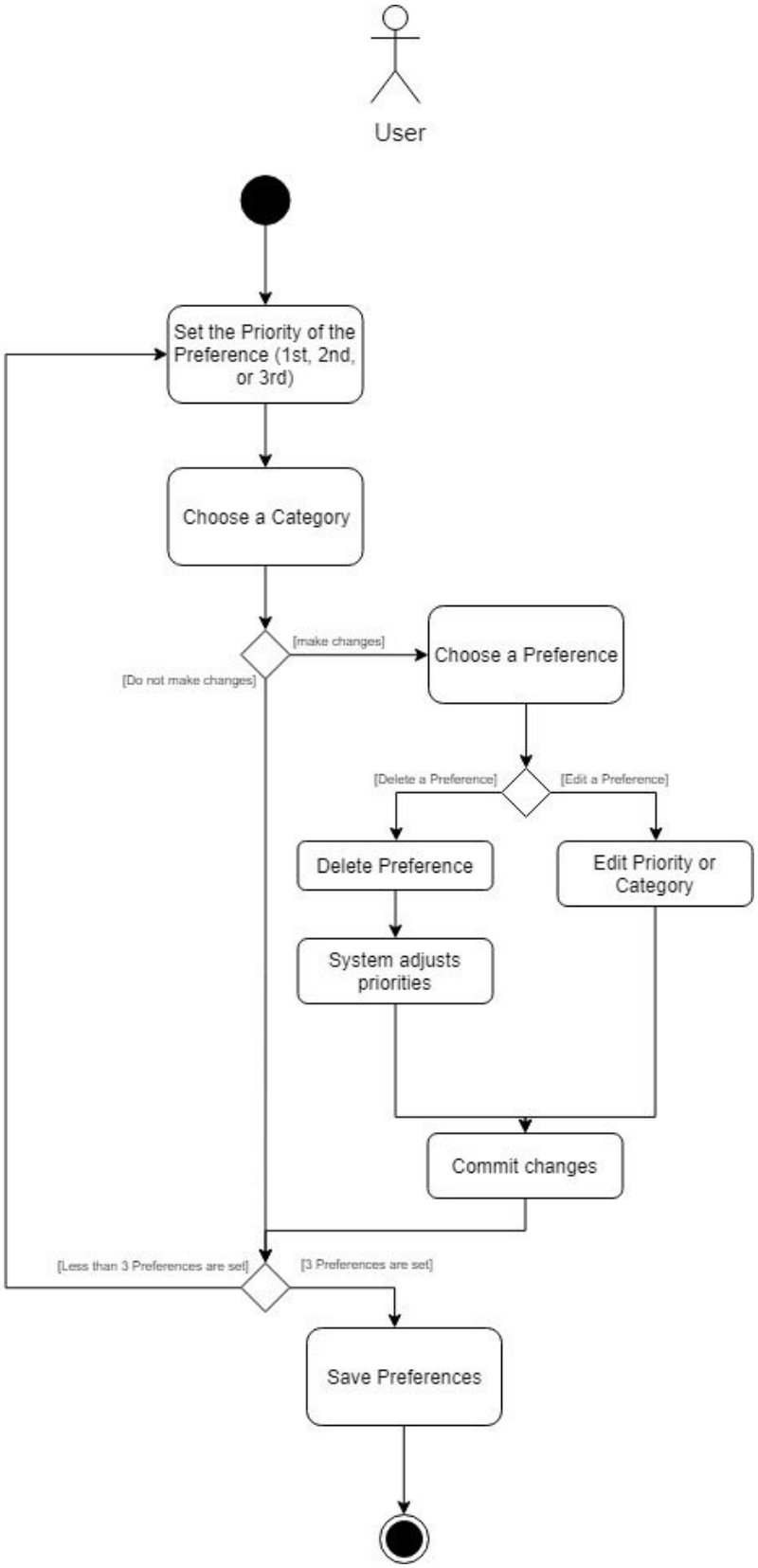
Description: This function of the Schedule Optimizer is to set the necessary variables that would factor in the scheduling algorithm. Indicate Preference is a prerequisite in creating an optimized schedule, i.e the software will not be able to process a request if the preferences are left blank. The User may indicate three preferences ranked from first, second and third. These ranking reflects the priority of the preference and will factor in the scheduling algorithm. While in the Indicate Preference Case, the user may add, edit or delete preferences.

Preconditions: To proceed into this Use Case, the User must have finalized the subjects to be scheduled.

Flow of Events:

Scenario Name	*Description*
Scenario 1 (Basic Flow) User indicates three preferences with respective priorities.	1. User sets the priority (1st, 2nd, or 3rd) of the preference. 2. User chooses from a list of preferences. 3. If User has successfully chosen three Preferences, confirm changes. If not, repeat step 1.
Scenario 2 (Alternative Flow) User wishes to edit a Preference	1. User sets the priority (1st, 2nd, or 3rd) of the preference. 2. User chooses from a list of preferences. 3. User chooses a preference to edit. 4. User chooses to edit the priority or the category. System saves the changes. 5. If User has successfully chosen three Preferences, confirm changes. If not, repeat step 1.
Scenario 2 (Alternative Flow) User wishes to delete a Preference	1. User sets the priority (1st, 2nd, or 3rd) of the preference. 2. User chooses from a list of preferences. 3. User chooses a preference to delete. 4. System makes the appropriate priority adjustment based on the deletion of a category. System saves the changes. 5. If User has successfully chosen three Preferences, confirm changes. If not, repeat step 1.

Activity Diagram of the Flow of Events:



Other Diagram: NONE

Postcondition: NONE

Relationships: NONE

Special Requirements:
NONE