Sked-it

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:

Dela Sierra, Joshua Joseph Riki V
Garcia, Patric Charles M.
Granda, Justin Tristan Gabriel R..

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/jvdelasierra/sked-it/tree/master/02-Requirements%20Engineering/Project%20Deliverables referenced with “sked-it-3.0-Link_Schedule.pdf" .

Document Purpose:

To be able to specify the requirements and other details of Link Schedule , Use Case 3.0, of the sked-it application.

Target Audience:

The target audience of this document are programmers and designers of the sked-it. Also, anyone who is interested in the application and also acquainted to most, if not all, jargons in this document.

Revision Control:

Revision Date	*Person Responsible*	*Version Number*	*Modification*
09/21/19	Justin Tristan Gabriel R. Granda	1.0	Preparation of initial document
09/21/19	Justin Tristan Gabriel R. Granda	2.0	Link Schedule Use Case Flow of Events and Activity Diagrams

***Use-Case Name*:** 3.0 Link Schedule

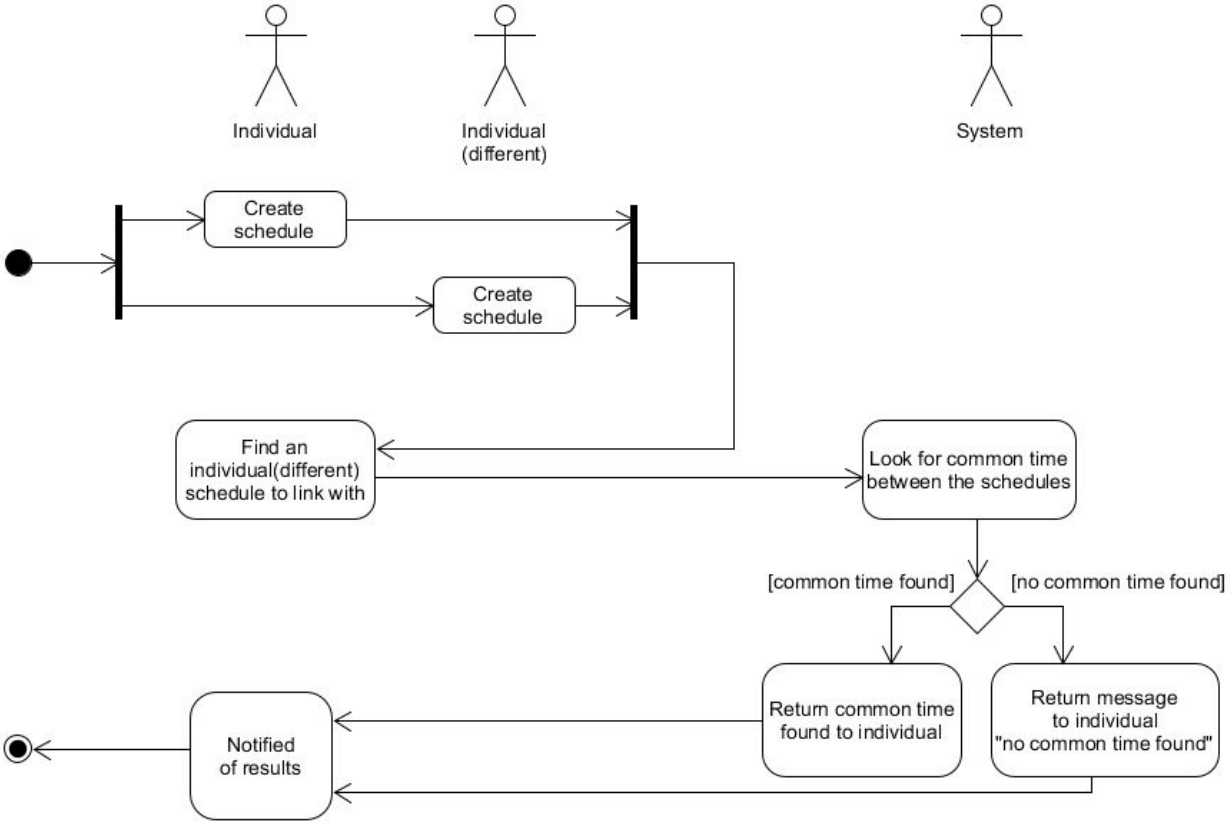
Description: This is the system's main functionality. Given some schedules, it returns the common free time found between them. This can be done between individual users or within a group.

Preconditions: Users need to have created schedules in order to use this.

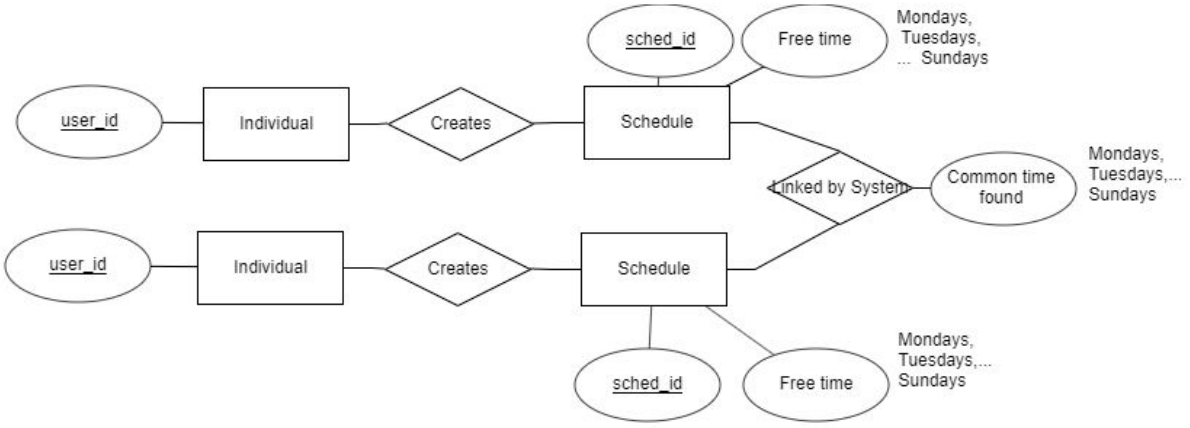
Flow of Events:

Scenario Name	*Description*
Scenario 1 (Basic Flow) Individual user wants to link his schedule with someone else's.	1. Individual user must have already created a schedule. 2. Individual user finds a schedule already created by a different user to link with. 3. System receives the schedules and looks for the common time. 4. Common time is found, relay this to the user. 5. Individual user is notified of the results.
Scenario 2 Users in a group want to link their schedules in order to plan for a certain event/activity.	1. Group chooses members whose schedules are to be linked. 2. System receives the schedules and looks for the common time. 3. Common time is found, relay this to the group. 4. Group is notified of the results.
Scenario 3 Individual user links his schedule with someone else, but no common time is found between them.	1. Individual user must have already created a schedule. 2. Individual user finds a schedule already created by a different user to link with. 3. System receives the schedules and looks for the common time. 4. No common time was found, relay error message to the user. 5. Individual user receives message stating no common time was found.

Activity Diagram of the Flow of Events:



Other Diagram:



Postcondition: NONE.

Relationships: NONE.

Special Requirements:

NONE.