

GetUP

Use Case Specifications

Submitted to:

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

Submitted by:

Jian Chan
Gab Datiles
Hans Santos

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



This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).

The documents are stored in the <https://github.com/gab51299/GetUP/tree/master/02-Requirements%20Engineering/Project%20Deliverables> referenced with GetUP-1-Maintain Workouts.pdf.

Document Purpose:

This document is intended to elaborate on the use case specifications for the GetUP project. Specifically, this document will be focusing on the “Maintain Workouts” use-case.

Target Audience:

UP Diliman community (students and faculty), particularly the class of CS 191 for the 1st semester of AY ‘19-’20.

Revision Control:

<i>Revision Date</i>	<i>Person Responsible</i>	<i>Version Number</i>	<i>Modification</i>
09/22/19	Jian Chan	1.0	Initial Document
09/23/19	Jian Chan	1.1	Document purpose, Target audience, Description, Use-case name, and Preconditions
09/23/19	Hans Santos	2.0	Add Flow of Events and Activity Diagram
09/23/19	Jian Chan	2.1	Add ER Diagram and list relationships

Use-Case Name: 2.0 Maintain Workouts

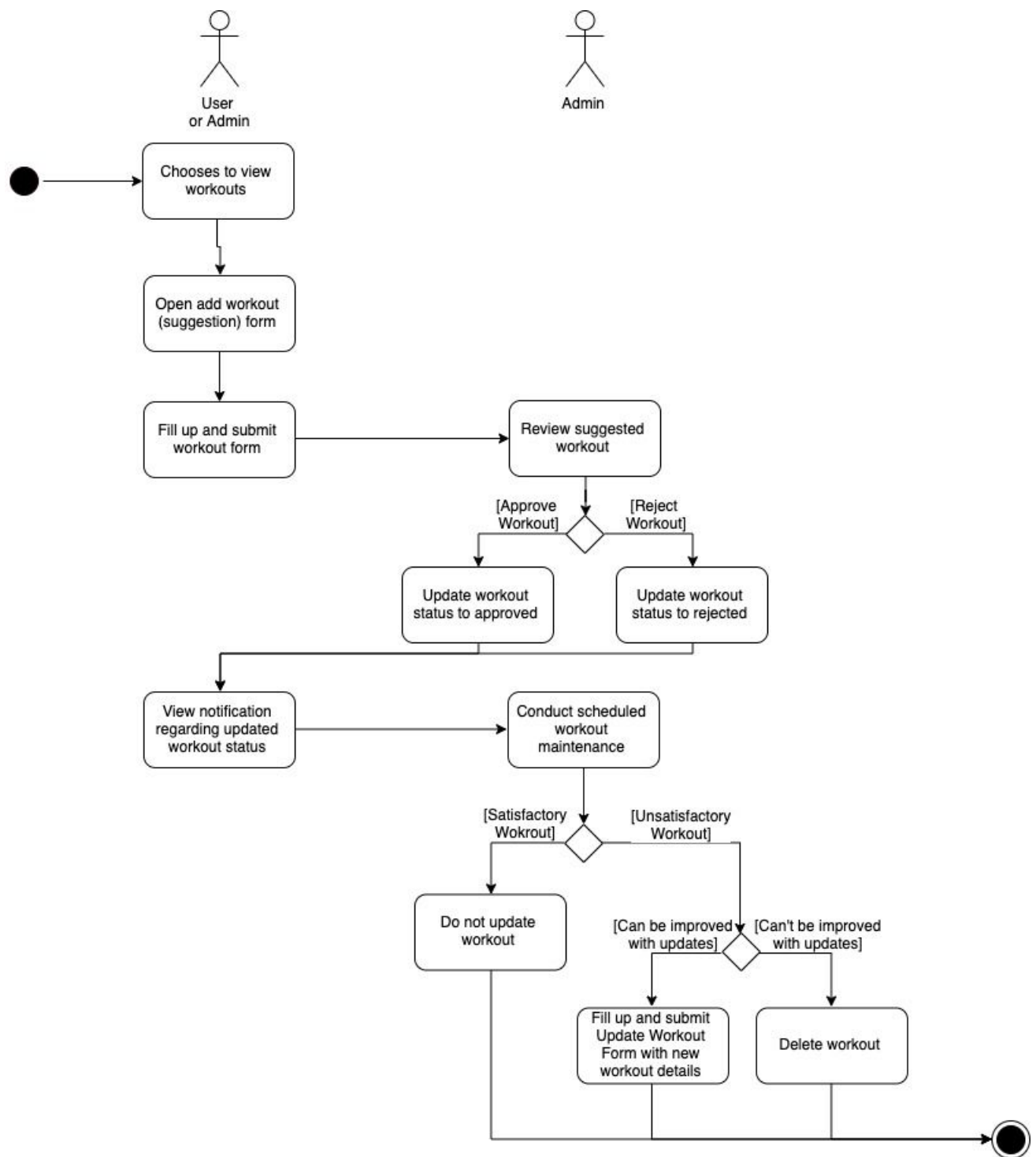
Description: The users and administrators can add workouts, that have default status of pending. Only administrators can approve these workouts by updating their status to approved. Administrators may also delete or update existing workouts, when necessary.

Preconditions: User or administrator is logged in

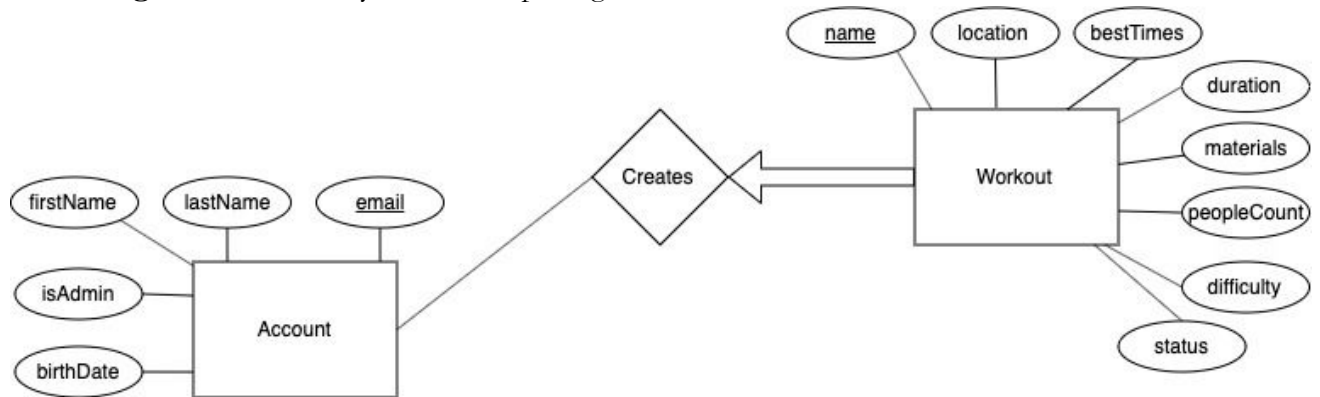
Flow of Events:

<i>Scenario Name</i>	<i>Description</i>
Scenario 1 (Basic Flow) User / Admin adds a workout, which is approved by an Admin. The workout is later on checked by an Admin and is not updated.	<ol style="list-style-type: none">1. User/Admin opens the application and is viewing the list of workouts.2. User/Admin opens the add workout form.3. User/Admin submits the completed form and confirms this submission.4. If workout is approved, Admin updates its status to approved.5. User/Admin who submitted the workout is notified of the change in status.6. During workout maintenance, if workout is deemed satisfactory, Admin leaves it as is.
Scenario 2 User / Admin adds a workout, which is approved by an Admin. The workout is later on checked by an Admin and is deemed unsatisfactory, thus updated.	<ol style="list-style-type: none">1. User/Admin opens the application and is viewing the list of workouts.2. User/Admin opens the add workout form.3. User/Admin submits the completed form and confirms this submission.4. If workout is approved, Admin updates its status to approved.5. User/Admin who submitted the workout is notified of the change in status.6. During workout maintenance, if workout is deemed unsatisfactory, Admin evaluates if workout can be improved.7. If workout can still be improved and is still usable, Admin updates the workout accordingly.
Scenario 3 User / Admin adds a workout, which is approved by an Admin. The workout is later on checked by an Admin and is deemed unsatisfactory, thus deleted.	<ol style="list-style-type: none">1. User/Admin opens the application and is viewing the list of workouts.2. User/Admin opens the add workout form.3. User/Admin submits the completed form and confirms this submission.4. If workout is approved, Admin updates its status to approved.5. User/Admin who submitted the workout is notified of the change in status.6. During workout maintenance, if workout is deemed unsatisfactory, Admin evaluates if workout can be improved.7. If workout can no longer be improved and is no longer usable, Admin deletes the workout.
Scenario 4 User / Admin adds a workout, which is rejected by an Admin.	<ol style="list-style-type: none">1. User/Admin opens the application and is viewing the list of workouts.2. User/Admin opens the add workout form.3. User/Admin submits the completed form and confirms this submission.4. If workout is approved, Admin updates its status to rejected and will be deleted.5. User/Admin who submitted the workout is notified of the rejected workout.

Activity Diagram of the Flow of Events:



Other Diagram: Entity-Relationship Diagram



Postcondition: NONE

Relationships: Creates - Relationship between workout and account, each workout must be created by a single account

Special Requirements: NONE