

Mobile Application for fitness

Runner



November 8, 2015

Computer science and EnGineering

Khulna University

**Team Members**

Name Student ID

Shahrima Islam 130206

Rahima Akter 130229

Rafflesia Khan 130333

Table of Contents

[1. Introduction 3](#_Toc434713487)

[1.1. Methodology 3](#_Toc434713488)

[1.1.1. Different Phase of RUP 3](#_Toc434713489)

[1.2. Purpose 4](#_Toc434713490)

[1.3. Project Scope 4](#_Toc434713491)

[2. Project Overview 4](#_Toc434713492)

[2.1. Proposed System 4](#_Toc434713493)

[2.2 Our Plan 5](#_Toc434713494)

[2.3 Application Architecture 5](#_Toc434713495)

[2.4 Sensor Service Architecture to Computer Number of Steps taken 5](#_Toc434713496)

[2.5 Software Interface 6](#_Toc434713497)

[2.6 Hardware Interface 6](#_Toc434713498)

[2.7 Communication Interface 6](#_Toc434713499)

[2. 8 Constraints 6](#_Toc434713500)

[3. Use case Scenarios 6](#_Toc434713501)

[3.1. Use Case: Create/Edit Profile 7](#_Toc434713504)

[3.1.1. Use Case Diagram 8](#_Toc434713505)

[3.1.1. User Interface Diagram 8](#_Toc434713509)

[3.2. Use Case: Runner 9](#_Toc434713510)

[Use case ID No : Runner0002 9](#_Toc434713511)

[Actors : User 9](#_Toc434713512)

[Pre-Condition : User must have application installed in smart phone 9](#_Toc434713513)

[Post-Condition on success : User can interact with device with augmented reality 9](#_Toc434713514)

[3.2.1. Use Case Diagram 10](#_Toc434713515)

[3.1.2. User Interface Diagram 11](#_Toc434713516)

[3.3. Use Case: Overview 12](#_Toc434713517)

[Use case ID No : Runner0003 12](#_Toc434713518)

[Actors : User 12](#_Toc434713519)

[Pre-Condition : User must have application installed in smart phone 12](#_Toc434713520)

[Post-Condition on success : User can view his/her status of fitness data on a graph 12](#_Toc434713521)

[3.3.1. Use Case Diagram 13](#_Toc434713522)

[3.1.3. User Interface Diagram 14](#_Toc434713523)

[3.4. Use Case: Activity Recognition 14](#_Toc434713524)

[Use case ID No : Runner0004 14](#_Toc434713525)

[Actors : User 14](#_Toc434713526)

[Pre-Condition : User must have application installed in smart phone 14](#_Toc434713527)

[Post-Condition on success : User can view his/her current activity 14](#_Toc434713528)

[3.4.1. Use Case Diagram 15](#_Toc434713529)

[3.1.4. User Interface Diagram 15](#_Toc434713530)

[3.5. Use Case: About 16](#_Toc434713531)

[Use case ID No : Runner0005 16](#_Toc434713532)

[Actors : User 16](#_Toc434713533)

[Pre-Condition : User must have application installed in smart phone 16](#_Toc434713534)

[Post-Condition on success : User can view information related to application 16](#_Toc434713535)

[3.5.1. Use Case Diagram 16](#_Toc434713536)

[3.1.5. User Interface Diagram 17](#_Toc434713537)

[4. Dependencies, conflicts and schedule 17](#_Toc434713538)

[4.1 Dependencies 17](#_Toc434713539)

[4.2 Conflicts: 17](#_Toc434713540)

[4.3 Open issues 17](#_Toc434713541)

# Introduction

## Methodology

The Rational Unified Process brings together elements from all of the generic process models, supports iteration and illustrates good practice in specification and design. The RUP is normally described from three perspectives: A dynamic perspective that shows the phases of the model over time. A static perspective that shows the process activities that are enacted. A practice perspective that suggests good practices to be used during the process.]

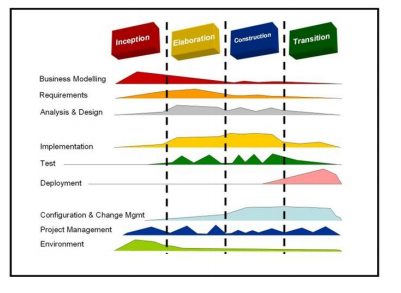


Figure 1: Phase of RUP

### Different Phase of RUP

The different phases in RUP are

#### Inception

The goal of the inception phase is to establish a business case for the system. Identifying all external

entities that will interact with the system and defining these interaction. This information is used to

assess the contribution of system to business.

#### Elaboration

The goals of the elaboration phase are to develop an understanding of the problem domain, establish

an architectural framework, develop project plan and identify key project risks.

#### Construction

This phase is concerned with system design, programming and testing. Parts of the system are developed in parallel and integrated during this phase.

#### Transition

This is the final phase of RUP and is concerned with moving the system from the development community to the user community and making it work in real environment.

## Purpose

**Runner**  is a mobile application which provides real time **Augmented Reality (AR)** feature in order to detect user’s (in this case user of mobile phone) real-time time movement and based on the change of his/her movement it is capable of taking different decisions. For example, if user jumps it can detect jumps and on an animated screen it can show his/her avatar to jump as well.

In recent days, mobile user are more interested about more connectivity with their device through Augmented Reality and enable Internet of Things. In this application, different movement for this user either on Bicycle, in vehicle, running, walking, still or jumping have been detected using AR and based on the change of movement it has been reflected on different transition in the application. Besides AR another popular feature has been used in this application which is **Fitness.** User can detect how many steps he has been taking while walking for a definite duration which is also being measured. At the same time (s)he will also be able to measure how much calorie has been burned in that definite time duration.

In recent days, application for fitness using Augmented Reality is one of the most popular application. This application will be very popular among user and it helps people to maintain a neat health record and lead to a fit life.

## Project Scope

1. The basic user is anyone with a smart phone and this application installed. As this is an **Anroid** application .therefore, the minimum API must be 4.0 or more.
2. User must create his profile in application. This will help to keep record of his continuous fitness record. Such as his calorie burn, weight and steps taken.
3. User can store his record in database and look for later.
4. User can take actions and AR implementation of this application will help device to make appropriate action.
5. User can view his/her status as graph.
6. User will have an animated interaction and can have his/her own avatar to interact with the application.
7. **Gamification** has been introduced in order to make the interaction more user friendly and interesting.

# Project Overview

The **Runner** Application is mainly implemented for smart phone with Android. This application will assist user to organize and take care of his/her fitness activities using an animated interface.

## Proposed System

* Database Integration
* Augmented Reality based Interaction
* Sensor Service Integration
* Fitness Service implementation
* Activity Recognition Service access
* Google Cloud Service Integration

## 2.2 Our Plan

* User can create his/her own profile to store his weight height and other information related to his/her fitness
* User can detect his/her current activity
* User can interact with an animated interface to access augmented reality in order to reflect his/her current activity
* User can count his steps
* User can computer how much calories has been burned over definite time duration
* User can see data on graphical presentation

## 2.3 Application Architecture

Figure 2 Shows the **Runner** Application Architecture.

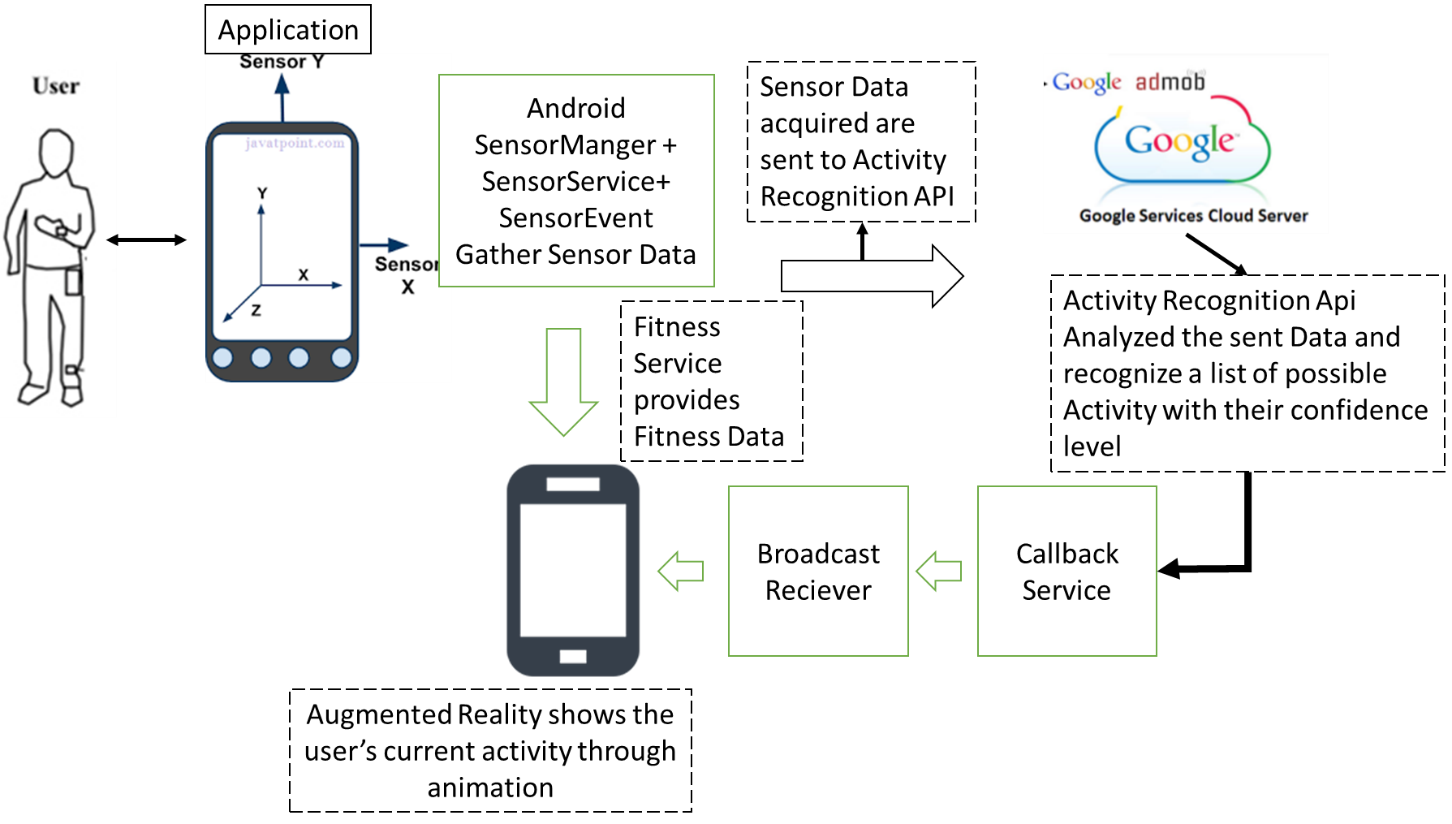


Figure 2: Application Architecture

### 2.4 Sensor Service Architecture to Computer Number of Steps taken

In this Application, Android Sensor has been used in order to detect any shake action taken by user and in order to detect how many steps have been taken by the during definite time. Fig. [3] describes the flow of SensorEvent and responses in order to sense any sensor movement.

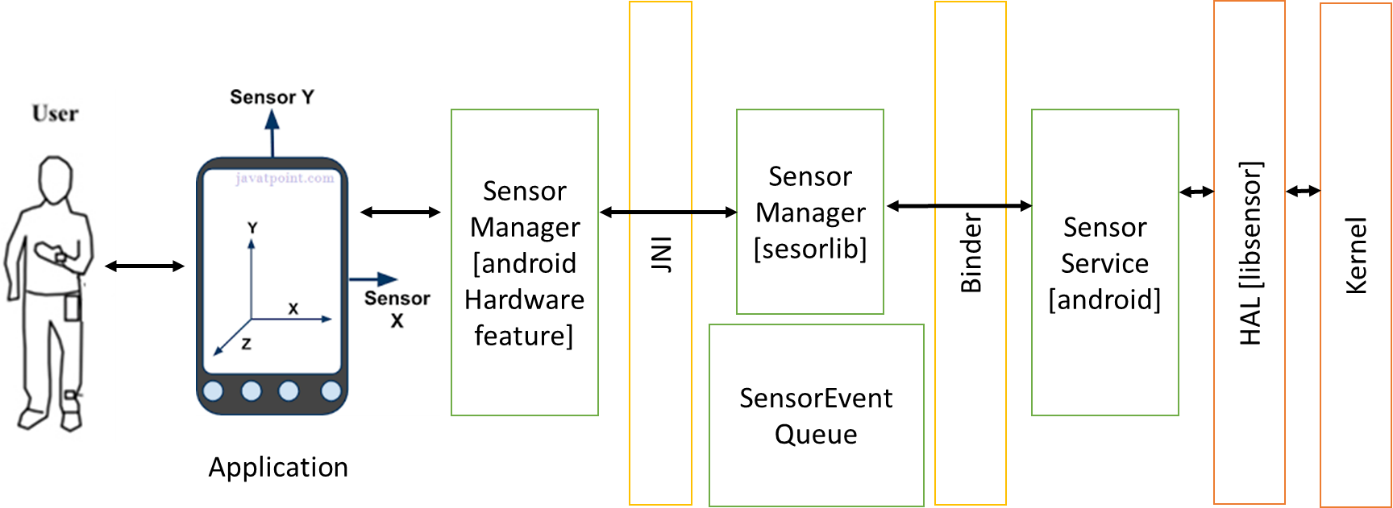


Figure 3: Architecture of Android Sensor Service to detect Step Taken & Shake

## 2.5 Software Interface

1. Java
2. Android 6.0
3. Google Play Service
4. Google Fitness Service
5. Google Cloud Service
6. Android Sensor
7. Android Graph Service
8. Android Animation
9. Spreadsheet Animation
10. GitHub
11. Eclipse
12. Microsoft Visio
13. Microsoft Word
14. Augmented Reality
15. Sqllight Database

## 2.6 Hardware Interface

1. Android SDK 23.0
2. Smart Phone with Sensor Hardware feature
3. Google Play Service

## 2.7 Communication Interface

1. Client (customer) on Internet will be using HTTP/HTTPS protocol.
2. Client (system user) on Internet will be using HTTP/HTTPS protocol.

## 2. 8 Constraints

1. GUI is only in English.
2. Single User Implementation.
3. Limited to HTTP/HTTPS.
4. This system is working for cloud server

# 3. Use case Scenarios

User can have following use cases



## Use Case: Create/Edit Profile

Use case ID No : Runner0001

Actors : User

Pre-Condition : User must have application installed in smart phone

Post-Condition on success : User will be saved in database

Main Flow:

1. User launches the application.
2. User will be given a set of Menu
3. User Select Profile from the menu options.
4. Application will populate the create profile console with input fields for user.
   1. If there is no data saved for the profile this window will be presented with blank input fields
   2. If the profile is already saved this window will open in edit mode with all saved data
5. User input all required input.
6. Use can Change profile picture
7. User select Submit button.
8. All data are saved in the sqllight database in a table named **Profile**
9. Application launches window of Main Menu.

Alternative Flow:

1. User selects “Cancel” Button.
2. User will return back to Main Menu. Without saving profile

Extension: NA

**Actors point:** User : Complex, Hand-device: Smart Phone

**Use case point:** Average

## Use Case Diagram



Figure 4: Use case Diagram for Profile Create



## User Interface Diagram

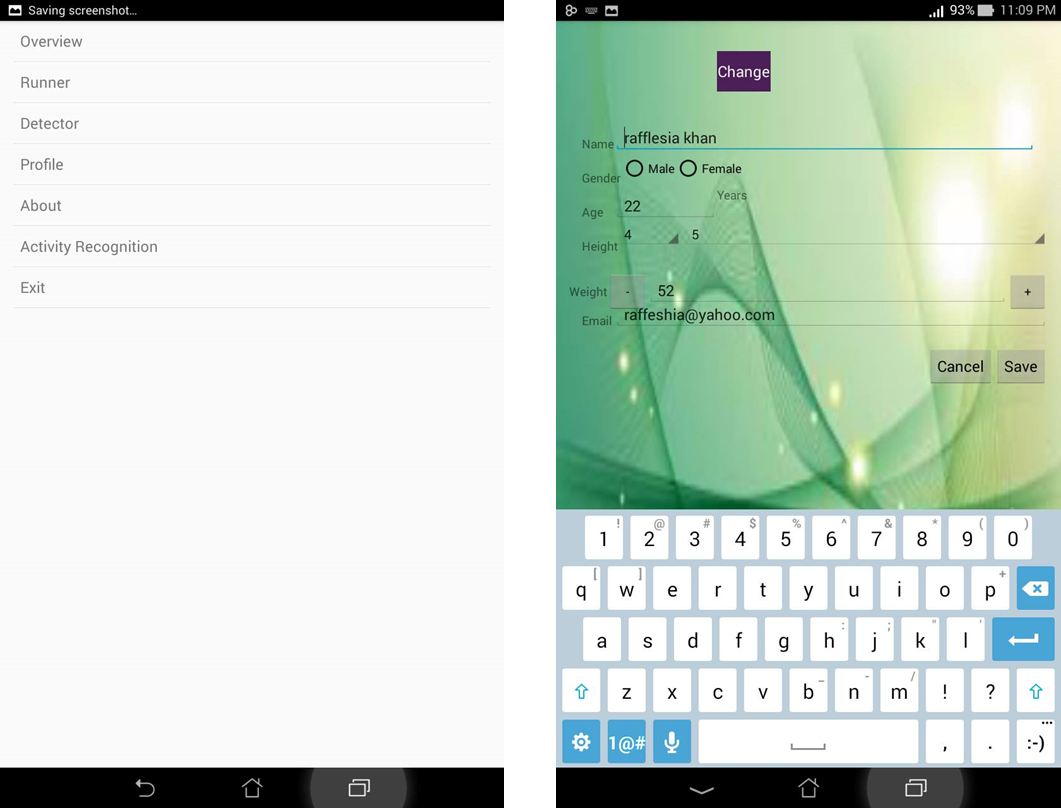


Figure 5: Screen Shot for Use Case of Create/Edit Profile

## Use Case: Runner

### Use case ID No : Runner0002

### Actors : User

### Pre-Condition : User must have application installed in smart phone

### Post-Condition on success : User can interact with device with augmented reality

Main Flow:

1. User launches the application.
2. User will be given a set of Menu
3. User Selects **Runner** from the menu options.
4. Application will populate the **Runner** console.
5. User selects time duration startTime and End time in order to set a goal that how long he wants to run/walk/drive/cycling [This version has implemented animated version for only walking and running.]
6. User Selects Start Activity Button
7. An animated interface launce to show user’s current interaction
   1. If user jump the animated avatar of user also imitate his/her activity and jump.
   2. In backend, sensor service detects users sensor event data and activity recognition service detects user’s activity
8. Step counter detect how many steps have been taken
9. Augmented reality portrays his/her activity on screen
10. User select “Stop” button
    1. Time, distance and burned calorie are computed and saved in database for this activity session
11. Time, distance and burned calorie are Shown to the user
12. User selects Back Button
13. Application launches window of Main Menu.

Alternative Flow:

1. User selects “Start” Button in step 6.
2. An animated interface launce to show user’s current interaction
   1. In backend only sensor service detects users sensor event data and an local service detects corresponding user activity
   2. If user jump the animated avatar of user also imitate his/her activity and jump.
3. User select “Pause” button
4. Session is paused until it has been resumed by the user.

Extension: NA

**Actors point:** User : Complex, Hand-device: Smart Phone

**Use case point:** Average

## Use Case Diagram



Figure 6: Use case Diagram for Runner

## User Interface Diagram

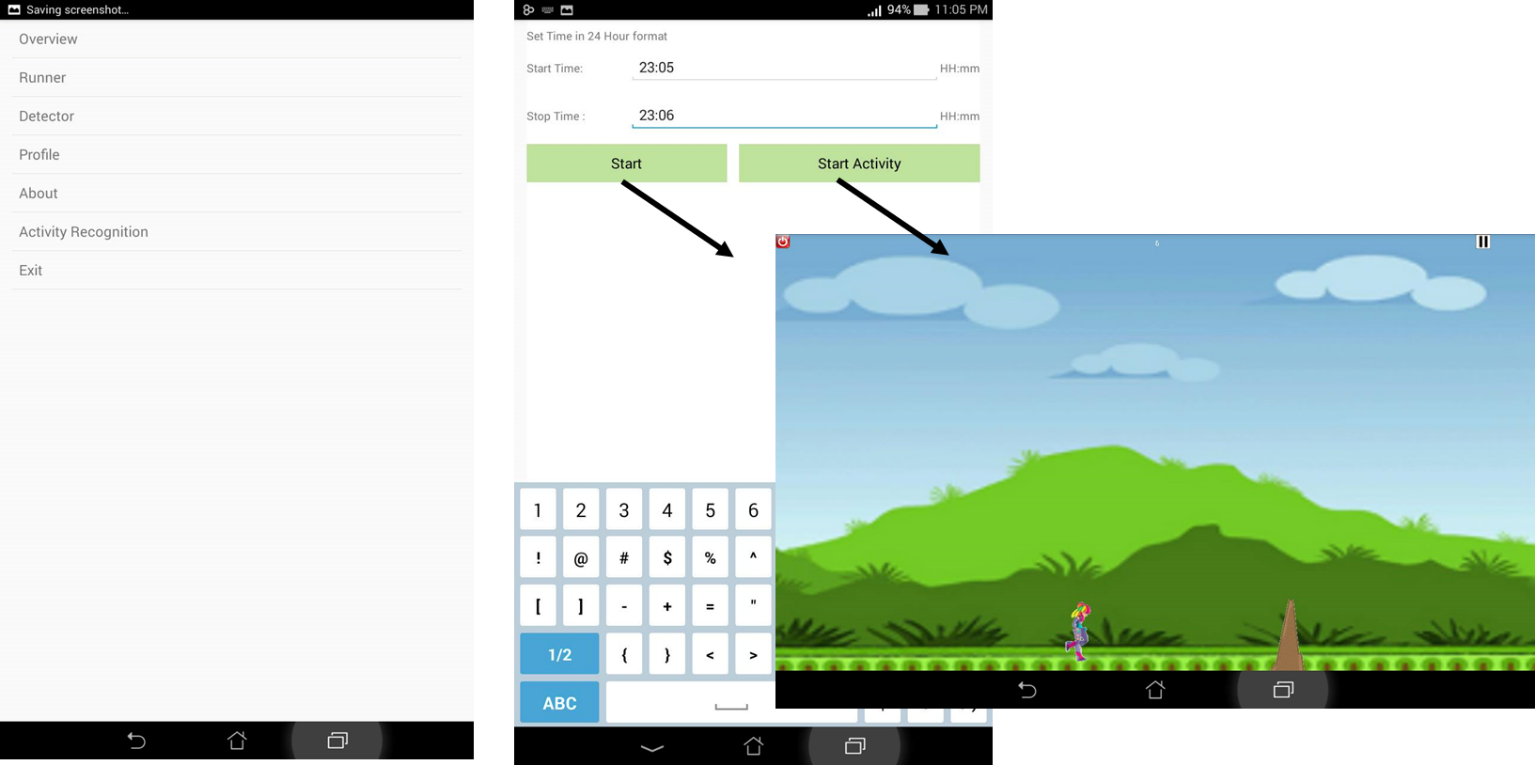


Figure 7: Screen Shot for Runner Start with or Without Activity Recognition API

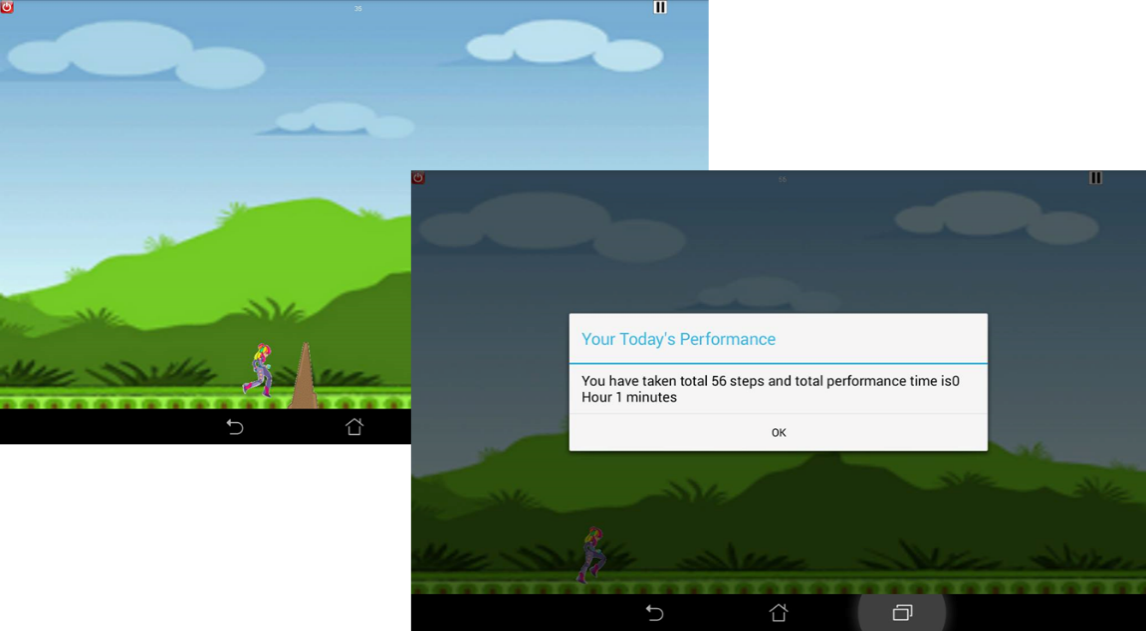


Figure 8: On Stop of Activity The application Shows current Status

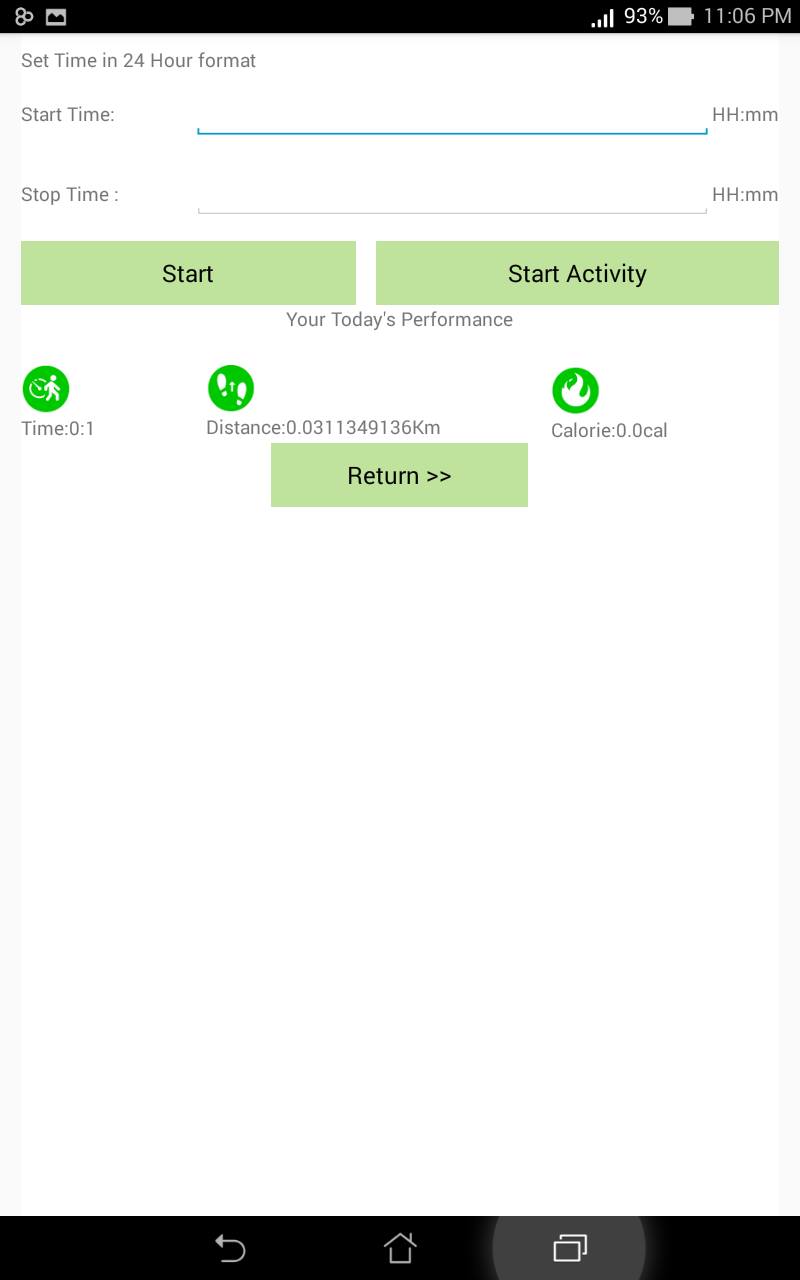


Figure 9: Status for the Activity Session showing Step, distance and burned calorie on this run

## Use Case: Overview

### Use case ID No : Runner0003

### Actors : User

### Pre-Condition : User must have application installed in smart phone

### Post-Condition on success : User can view his/her status of fitness data on a graph

Main Flow:

1. User launches the application.
2. User will be given a set of Menu
3. User Selects Overview from the menu options.
4. Application will populate the overview console.
5. User selects date duration from and To
6. User Selects Overview Button
   1. If there is no data saved for the profile this window will be presented with no graph
   2. If there are data for running session and user profile this window will preview a graphical representation of user’s running time, distance and calorie burned for all the session in selected duration.
7. User back Button
8. Application launches window of Main Menu.

Extension: NA

**Actors point:** User : Complex, Hand-device: Smart Phone

**Use case point:** Average

## Use Case Diagram



Figure 10: Use case Diagram for Overview

## User Interface Diagram

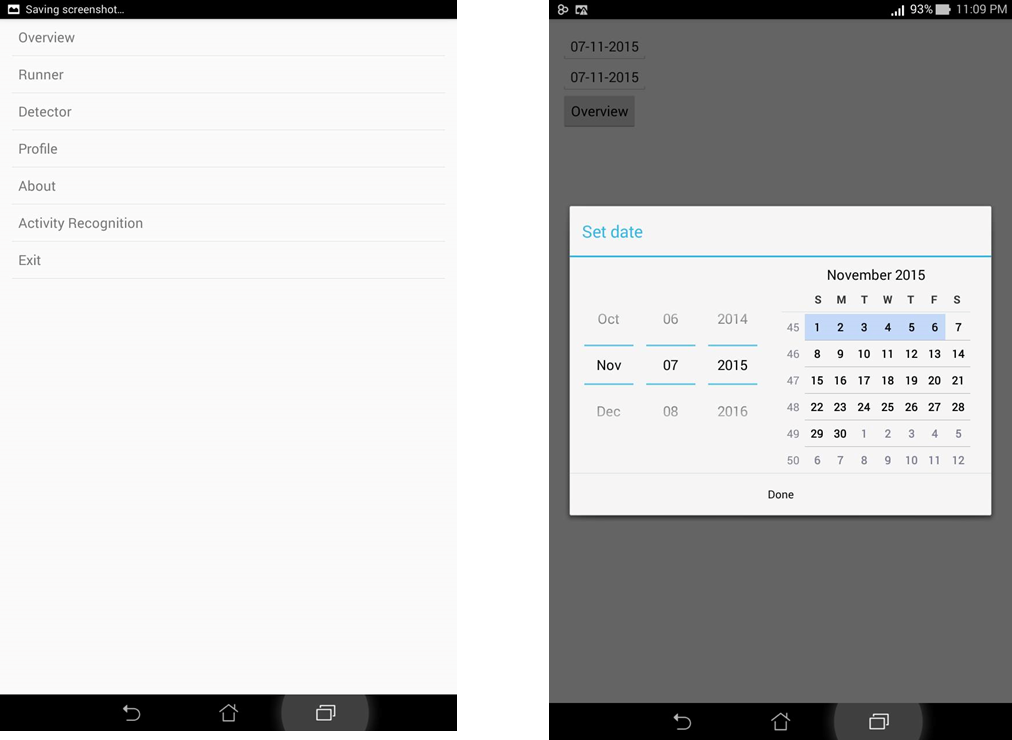


Figure 11: Set Time Duration for Status

## Use Case: Activity Recognition

### Use case ID No : Runner0004

### Actors : User

### Pre-Condition : User must have application installed in smart phone

### Post-Condition on success : User can view his/her current activity

Main Flow:

1. User launches the application.
2. User will be given a set of Menu
3. User Selects **Activity Recognition** from the menu options.
4. Application will populate the **Activity Recognition** console.
5. User can view a list of all possible activity such as still, on-foot, walking, running, On a bicycle, In a vehicle, Tilting or even unknown activity along with their corresponding confidence level.
6. User back Button
7. Application launches window of Main Menu.

Extension: NA

**Actors point:** User : Complex, Hand-device: Smart Phone

**Use case point:** Average

## Use Case Diagram



Figure 12: Use case Diagram for About

## User Interface Diagram

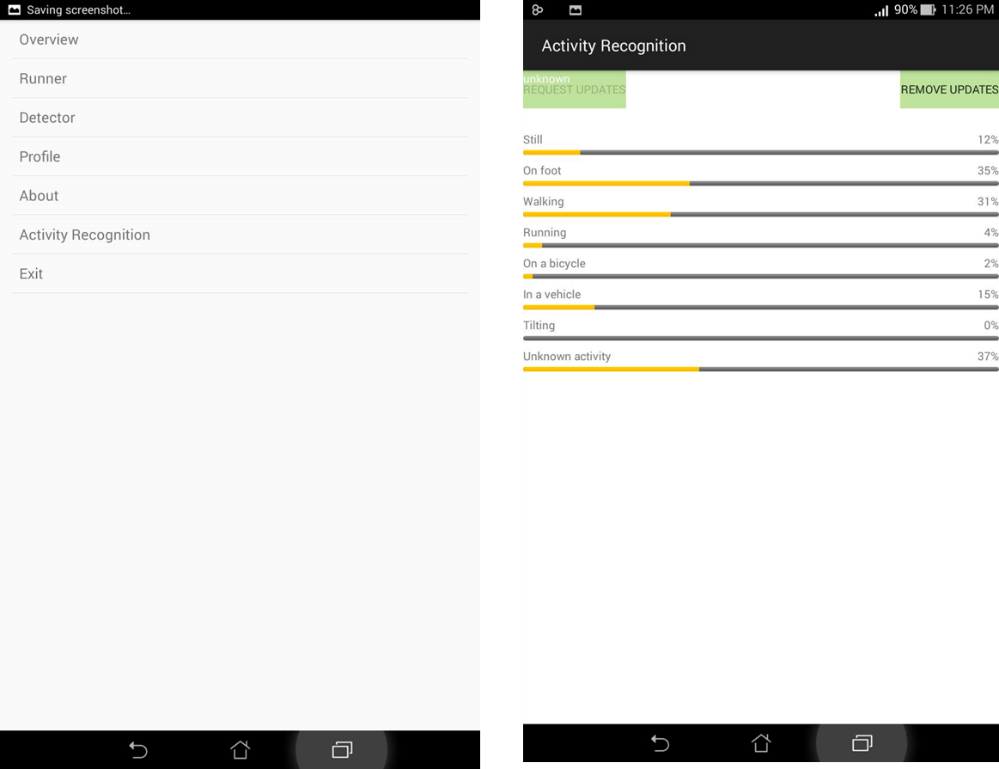


Figure 13: List of Activity with corresponding confidence level

## Use Case: About

### Use case ID No : Runner0005

### Actors : User

### Pre-Condition : User must have application installed in smart phone

### Post-Condition on success : User can view information related to application

Main Flow:

1. User launches the application.
2. User will be given a set of Menu
3. User Selects **About** from the menu options.
4. Application will populate the **About** console.
5. User selects can view all related information related developer, android version application version and so on.
6. User back Button
7. Application launches window of Main Menu.

Extension: NA

**Actors point:** User : Complex, Hand-device: Smart Phone

**Use case point:** Average

## Use Case Diagram



Figure 14: Use case Diagram for About

## User Interface Diagram

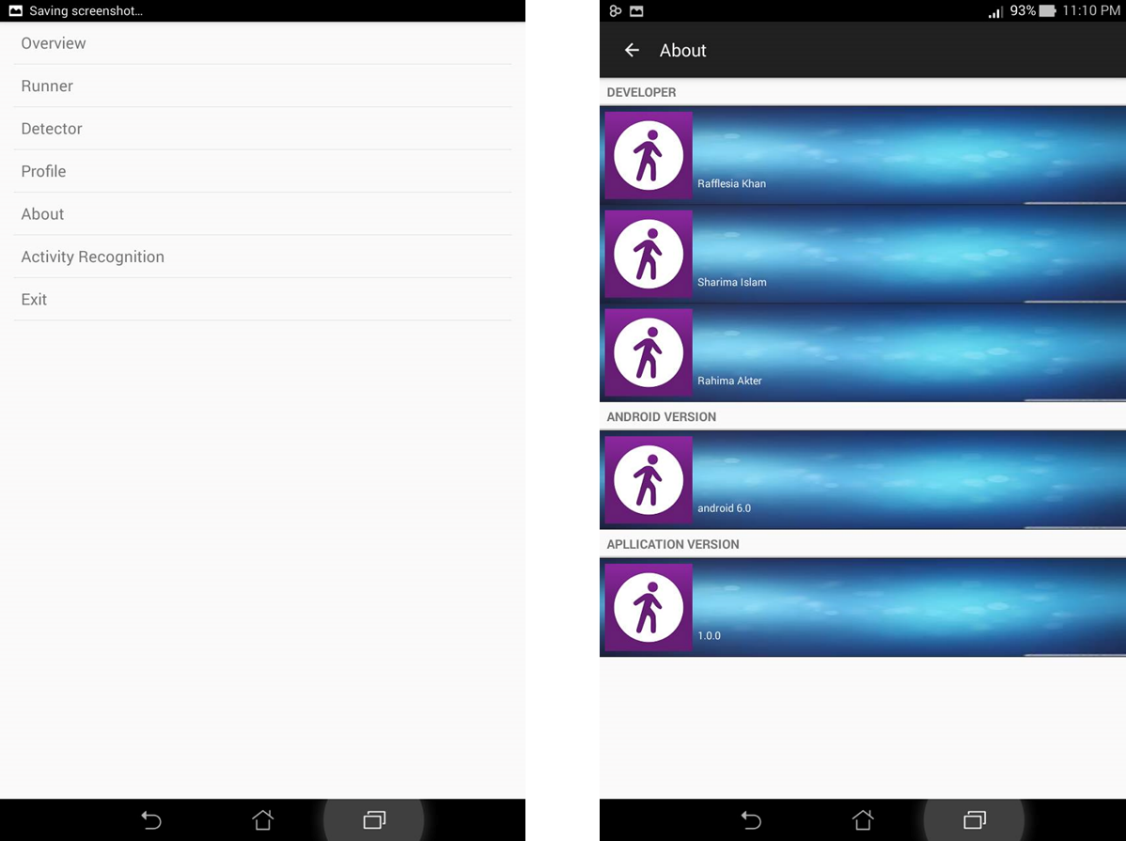


Figure 15: About Screen of Application

# 4. Dependencies, conflicts and schedule

## 4.1 Dependencies

No actual dependencies for this use case other than system login**.** Other pages should be available for successful navigation.

## 4.2 Conflicts:

There is no possibility of conflicts of any tasks with the tasks of other use cases.

## 4.3 Open issues

There is no possibility of conflicts of any tasks with the tasks of other use cases.