

## **Chapter 3**

### **System Analysis**

---

### 3.1 Introduction

#### 3.1.1 Purpose

Obesity is the most prevalent, fatal, chronic, relapsing disorder of the 21st century. Obesity is a leading cause of world's mortality, morbidity, disability, healthcare utilization and healthcare costs. It is likely that the increase in obesity will strain our healthcare system with millions of additional cases of diabetes, heart disease and disability. Significantly, excess adiposity or obesity causes insulin secretion, which can cause insulin resistance that leads to type-2 diabetes.

Regular physical activity will help you to avoid these things. Physical activity is essential to prevent and reduce risks of many diseases and improve physical and mental health. It can even help you live longer—research from the American Journal of Preventative Medicine indicates that regular exercise can add up to five years to your life.

Since regularity in physical activity is very important, the development of the fitness application that can run in your mobile device is mandatory.

#### 3.1.2 Scope

This document provides details of the following:

1. Functional overview of the 'Stay Fit' application.
2. Broad functionalities within 'Stay Fit' application.
3. External and internal interfaces of the modules of 'Stay Fit'.

#### 3.1.3 How to Use this Document

All users of this document should carefully read it, and ensure the following:

- All requirements have been listed and detailed
- Functionality described in this document is clear and unambiguous.

#### 3.1.4 List Of Abbreviations

Table 3.1: List Of Abbreviation

Abbreviation/ Acronym	Description
BLE	Bluetooth Low Energy
API	Application Program Interface
SDK	Software Development Kit
GPS	Global Positioning System
GL ES	Graphics Library Embedded System
HRM	Heart Rate Monitor
FPS	Frame per second
iOS	An operating system developed by Apple Inc. for mobile devices such as iPhone and iPad

## 3.2 Requirement Specifications.

A Software requirements specification (SRS), a requirements specification for a software system, is a complete description of the behaviour of a system to be developed and may include a set of use cases that describe interactions the users will have with the software. In addition it also contains non-functional requirements. Non-functional requirements impose constraints on the design or implementation.

### 3.2.1 Functional Requirements

#### R1 User's Information

##### R1.1 Save User Profile

**Description:** Get user's general information, which is required in calculating of expended energy.

**Input:** Name  
DOB  
Gender  
Weight  
Height

**Processing:** The information will be used to calculate the calories burned during the particular workout session.

##### R1.2 Edit User Profile

**Description:** Edit user's general information, which is required in calculating of expended energy.

**Input:** New Values of following :  
Name  
DOB  
Gender  
Weight  
Height

**Processing:** The information will be saved and the new information will be used in processing.

### R1.3 Reset All Data

**Description:** The all data that is stored about the user is deleted and the application starts from the state when it was first installed. This way the application handles an event of change of user.

**Input:** User commands to reset the application.

**Processing:** The data that was saved for a particular user is deleted and now application is ready to be used by another user

## R2 Workout Process

### R2.1 Set Workout Goals

**Description:** This module allows user to set his/her workout goals.

**Input:** Distance (meters) or  
Duration (minutes) or  
Step Count (number of steps) or  
Energy Expended (kcal)

**Output:** Application will notify user on completion of any goal. Application will also show percentage of Goal achieved where user can frequently check.

**Processing:** The information about goal provided by user will be used to track the workout process and remind the user whenever necessary.

### R2.2 Workout Reminder

#### R2.2.1 Add New Workout Reminder

**Description:** User will be able to add a reminder for workout.

**Input:** Time of reminder  
Repeat sequence  
Keep Remind me up to date  
Alarm Configuration

**Output:** Application will remind user at the time of Workout.

**Processing:** The application reminders which are set by the user within the application are synchronized with the ios application of reminders. So the alarms can be triggered at appropriate time by the ios system itself.

**R2.2.2 Remove Workout Reminder**

**Description:** User will be able to delete an existing reminder.

**Input:** User selects a reminder to be deleted from a list.

**Output:** Application will delete a reminder.

**Processing:** The application removes the reminder from the default reminder tab of the application.

**R2.2.4 Edit Workout Reminder**

**Description:** User will be able to edit an existing reminder.

**Input:** New Attributes of the reminder.

**Output:** Application will modify a reminder.

**Processing:** The application saves new parameters of the reminder from the default reminder tab of the application.

**R2.2.5 Disable Workout Reminder**

**Description:** User will be able remove an alarm from reminder.

**Input:** Reminder to disable.

**Output:** Application will remove the alarm and synchronize.

**R2.2.6 Enable Workout Reminder**

**Description:** User will be able add an alarm to disabled reminder.

**Input:** Reminder to enable.

**Output:** Application will add an alarm and synchronize.

### R2.3 Start New Workout

**Description:** User will be able to start new workout session either predefined or custom.

**Input:** The option of start workout is selected by user.

**Output:** User will be able to view his/her workout attributes which will contain total workout duration, distance, step count, Path on Map, Avg. Speed, Energy Expended & percentage of goal achieved.

**Processing:** On starting a new workout, application will start collecting workout data like, Heart Rate, Speed (Min, Max & Avg.), Energy Expended (Calories burned), Duration, Distance, Step Count, Path on Map.

### R2.4 Workout Report Generation

**Description:** User will be able to generate a report on the end of the workout session and also will be able to get the history of workout session he/she had previously undergone to.

**Input:** Duration during which the user had taken workout.

**Output:** Graph / report for workout duration, distance, step count, energy expended, heart rate for a day / week / month.

**Processing:** The information collected during a workout session is used to calculate the attributes of report.

## R3 HR Monitor Connection

**Description:** The user will be able to connect BLE peripheral defined according to HR profile of Bluetooth 4.0.

### R3.1 Discover BLE HR peripheral devices

**Input:** User will select the option to discover peripheral devices which provide the HR monitoring services.

**Processing:** Application will discover nearby HR Monitor devices.

**Output:** List of BLE devices with HR profile.

### **R3.2 Connect to BLE device**

**Input:** User will select the device to which the connection has to be made over Bluetooth

**Processing:** Application will connect to the device selected by user.

### **R3.2 Disconnect BLE device**

**Input:** User will select the command to disconnect from currently connected device.

**Processing:** Application will disconnect BLE device and user will now be able to connect to other devices.

## **R4 Continuous Hear Rate Monitoring**

### **R4.1 Record HR continuously**

**Description:** User can opt for continuous Heart Rate monitoring through this application.

**Input:** The HR of user measured in BPS, given by BLE device.

**Processing:** If user opts for this option, application will track user's heart rate for whole day. (one record for each second, total of  $24 \times 60 \times 60$  HR records per day).

### **R4.2 Plot HR record on Graph**

**Input:** The duration for which user wants to see the HR measurement which are previously recorded.

**Output:** The graph in which the HR records are plotted.

**Processing:** A graph is plotted in which the time is on X axis and the HR is on Y axis. The graph is scaled in best resolution according to the duration for which the user wants to see the HR measurements.

## R5 Voice Assistance

**Description:** Application will periodically provide workout information through voice guidance.

**Processing:** On starting a new workout, application will start collecting workout data like, Heart Rate, Speed (Min, Max & Avg.), Energy Expended (Calories burned), Duration, Distance, Step Count, Path on Map and time to time the application will give feedback to user in form of audio.

## R6 Music Player

**Description:** User will able to play music from device's music library during workout.

**Processing:** On starting a new workout, application will start Playing music which will be directly played from default music application of the device.

## R7 Social Network Sharing

**Description:** User will able share the details of his/her workout to facebook and twitter account.

**Preconditions:** The user must have a account configured with the settings app of the device.

**Processing:** The attributes of the workout will be shared on the preconfigured account of the user.



### **3.2.2 Non Functional Requirements**

#### **R1 Usability**

##### **R1.1 Ease of Use**

- This application is very user friendly so any non technical user can also use this application very easily.
- The graphs, which show HR measurements in application, will be interactive. The user will be able to change the scale of graph to see more or less finer data.

##### **R1.2 Required User Ability**

The user should know the basic handling of iOS devices

#### **R2 Reliability**

##### **R2.1 Error Handling**

1. The application should not crash
2. The application should not have memory leak
3. If any error occurs during runtime, it should notify user about it

##### **R2.2 Ease of Recovery**

If any crashing or error occurs in application, it should restore the all information and configuration data about workout.

#### **R3 Performance**

##### **R3.1 Response Time**

- The application will be able to record the HR of user continuously (1 record per each second).
- The application will be able to save the location (co-ordinates) which is visited by user during workout session so that it can show it on map when requested by user.

##### **R3.2 Resource Usage**

The memory requirement of application should be as minimum as possible. Also, the memory occupied by application should not affect the performance of other applications on that mobile.

**R4 Supportability****R4.1 Ease of Installation**

The installation procedure of application should be made easy so any novice user can install the application to his mobile.

**R5 Accuracy**

- The location tracked for the user will be taken from GPS hardware of device, configured for best possible accuracy.
- The expended energy measured, will be the best approximation of the original kcal burned by user during workout.
- The current HR of user depends upon the BLE device, which sends data. Thus the application is not responsible if the data sent by the device is not correct.

### 3.3 Use Case Diagram

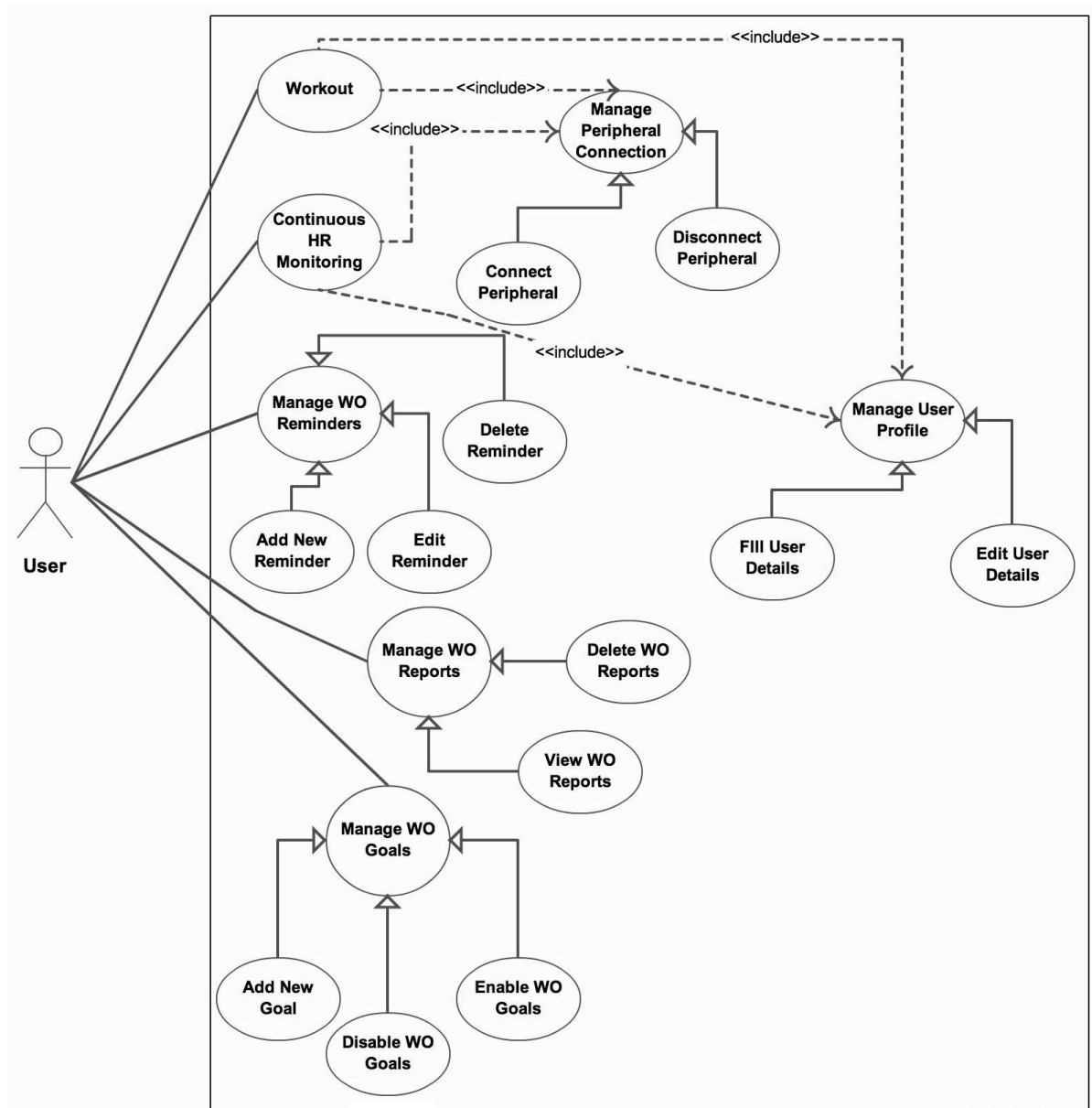


Figure 3.1: Use Case Diagram

### 3.4 Class Diagram

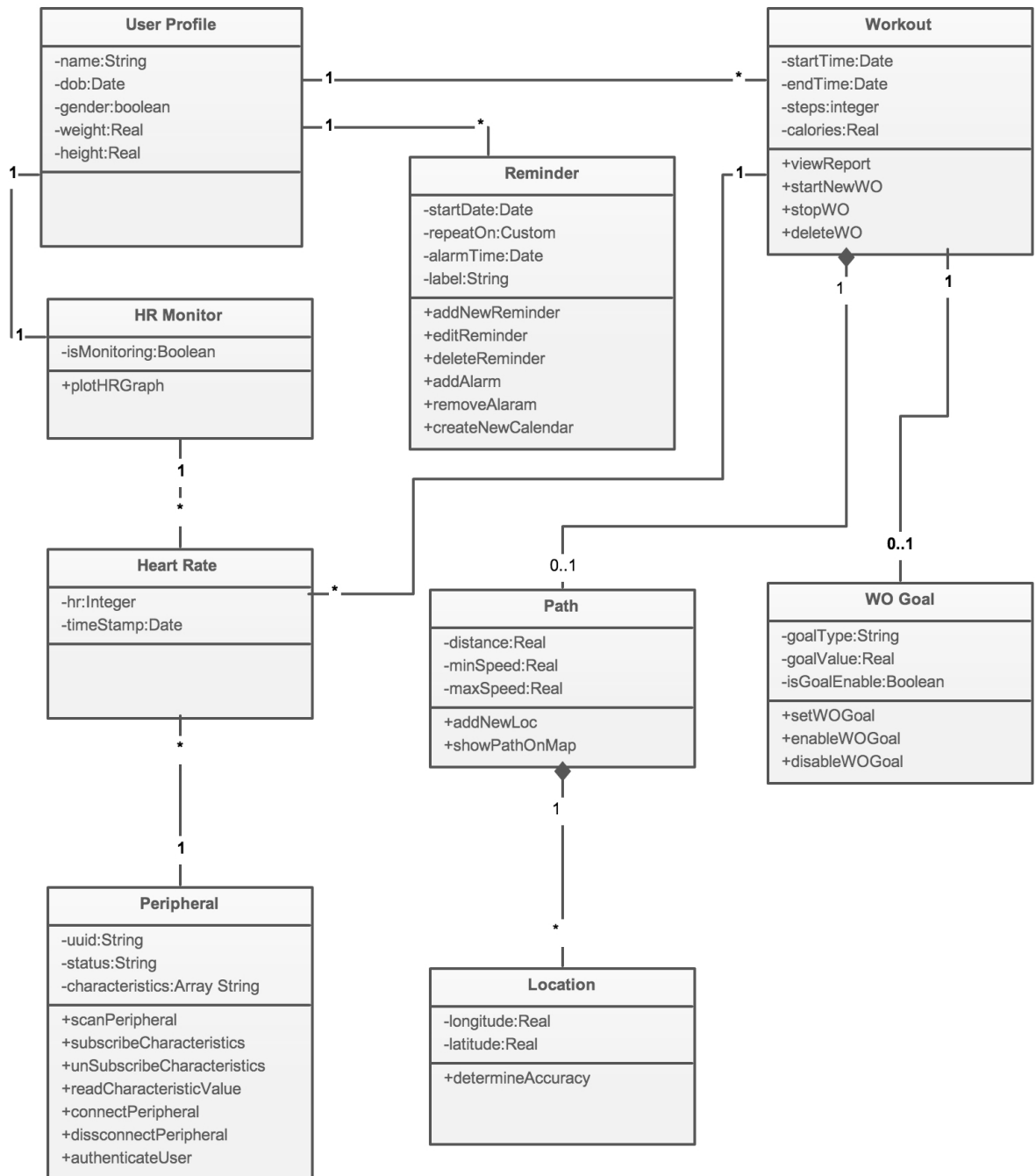


Figure 3.2: Class diagram

### 3.5 Sequence Diagrams

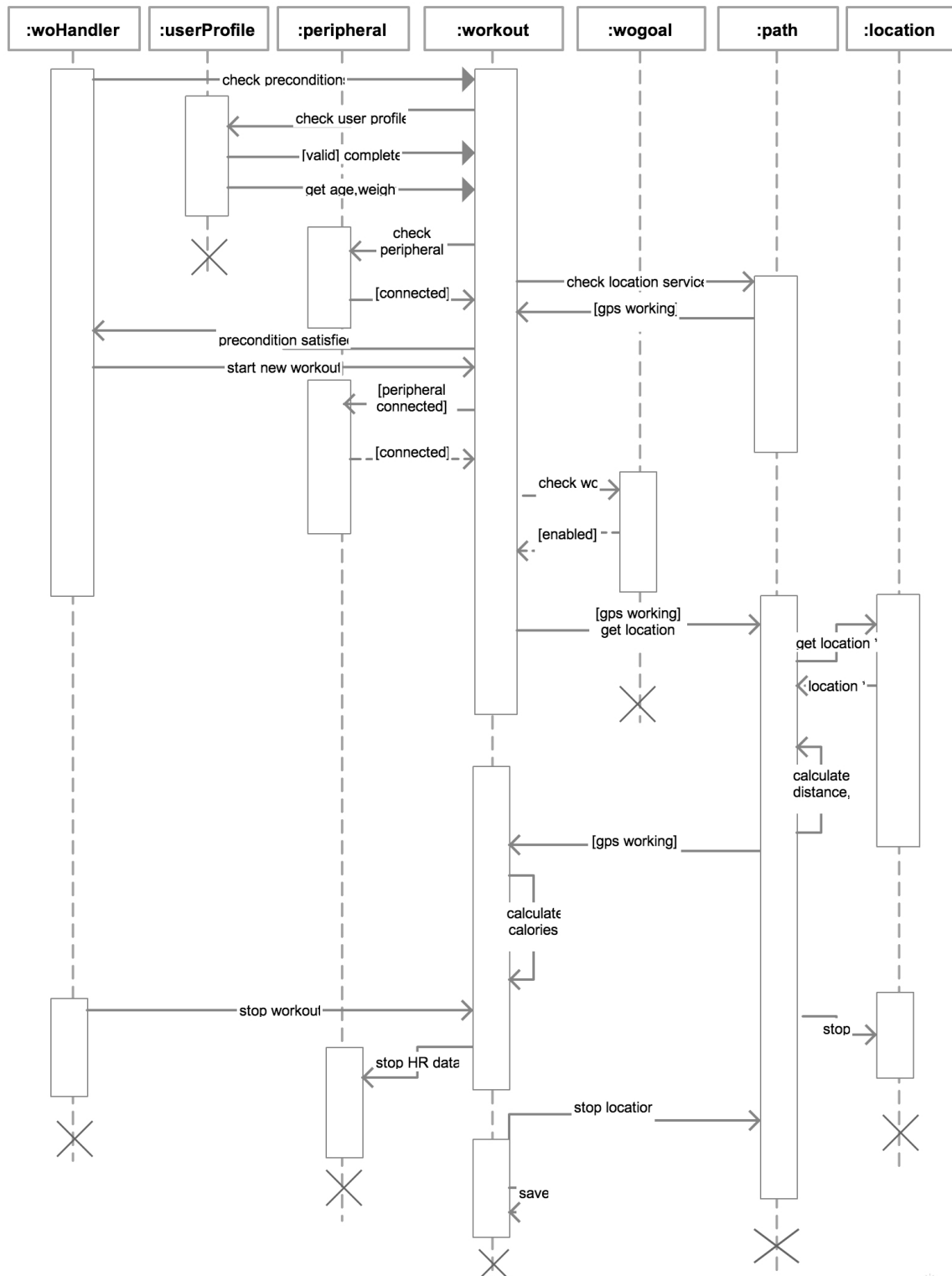


Figure 3.3: Sequence diagram for Start Workout

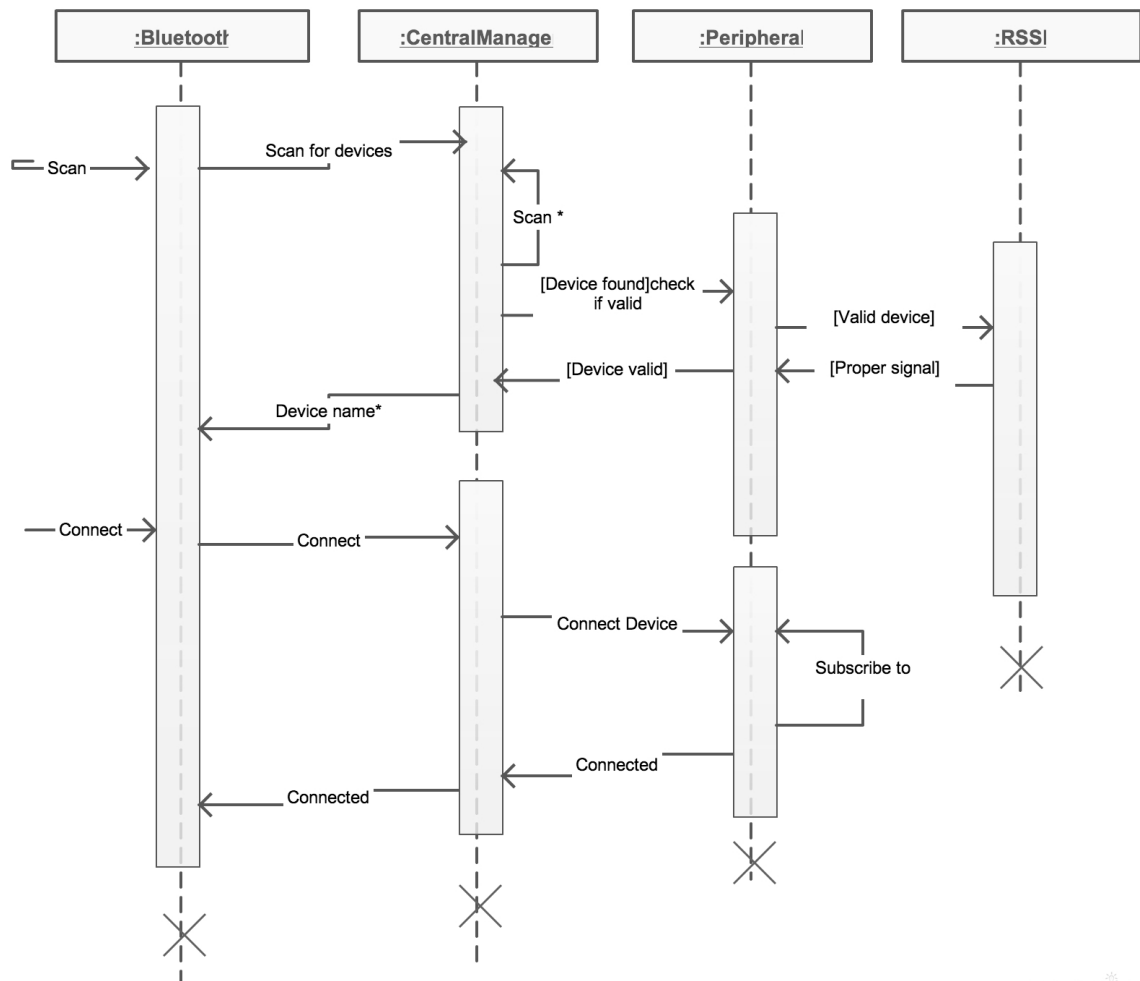


Figure 3.4: Sequence diagram for Connect BLE Device

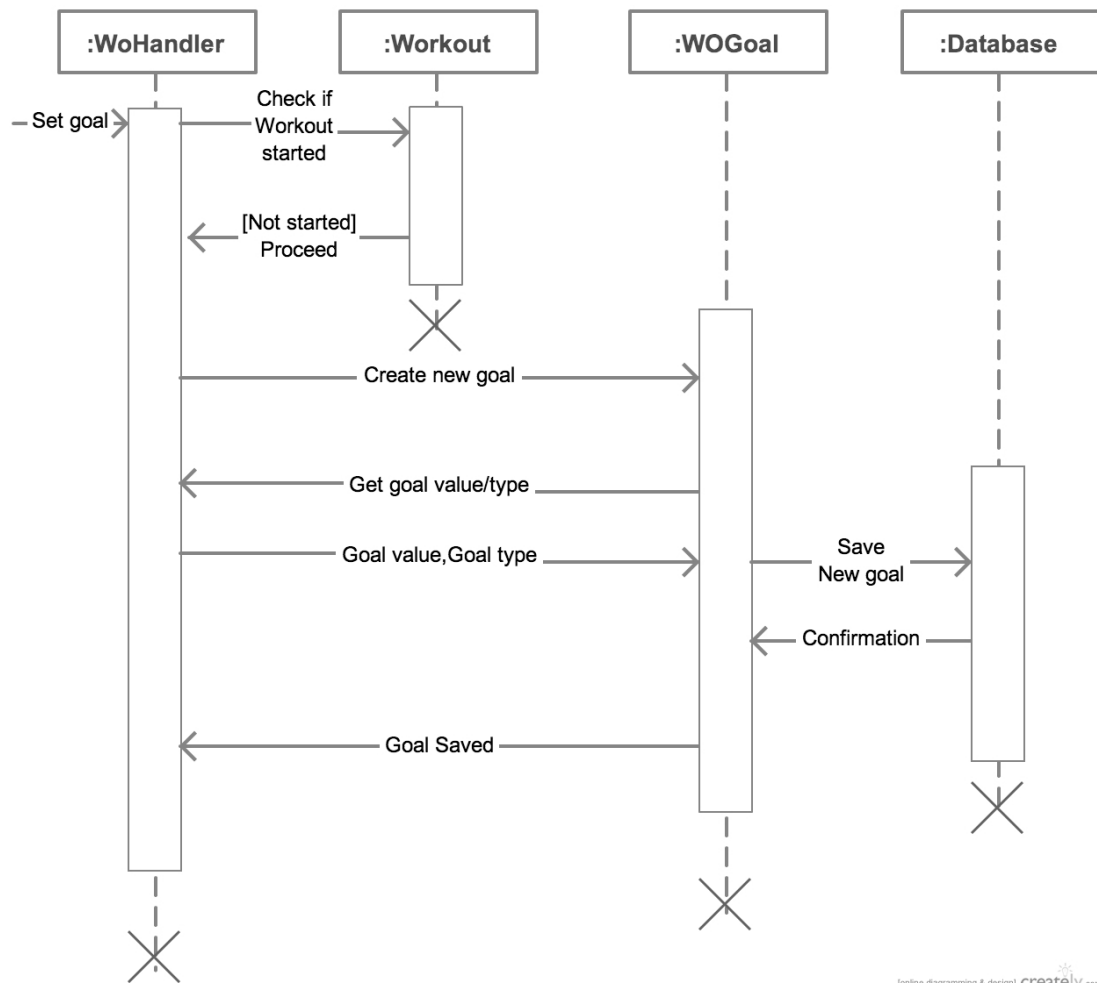


Figure 3.5: Sequence diagram for save workout goal

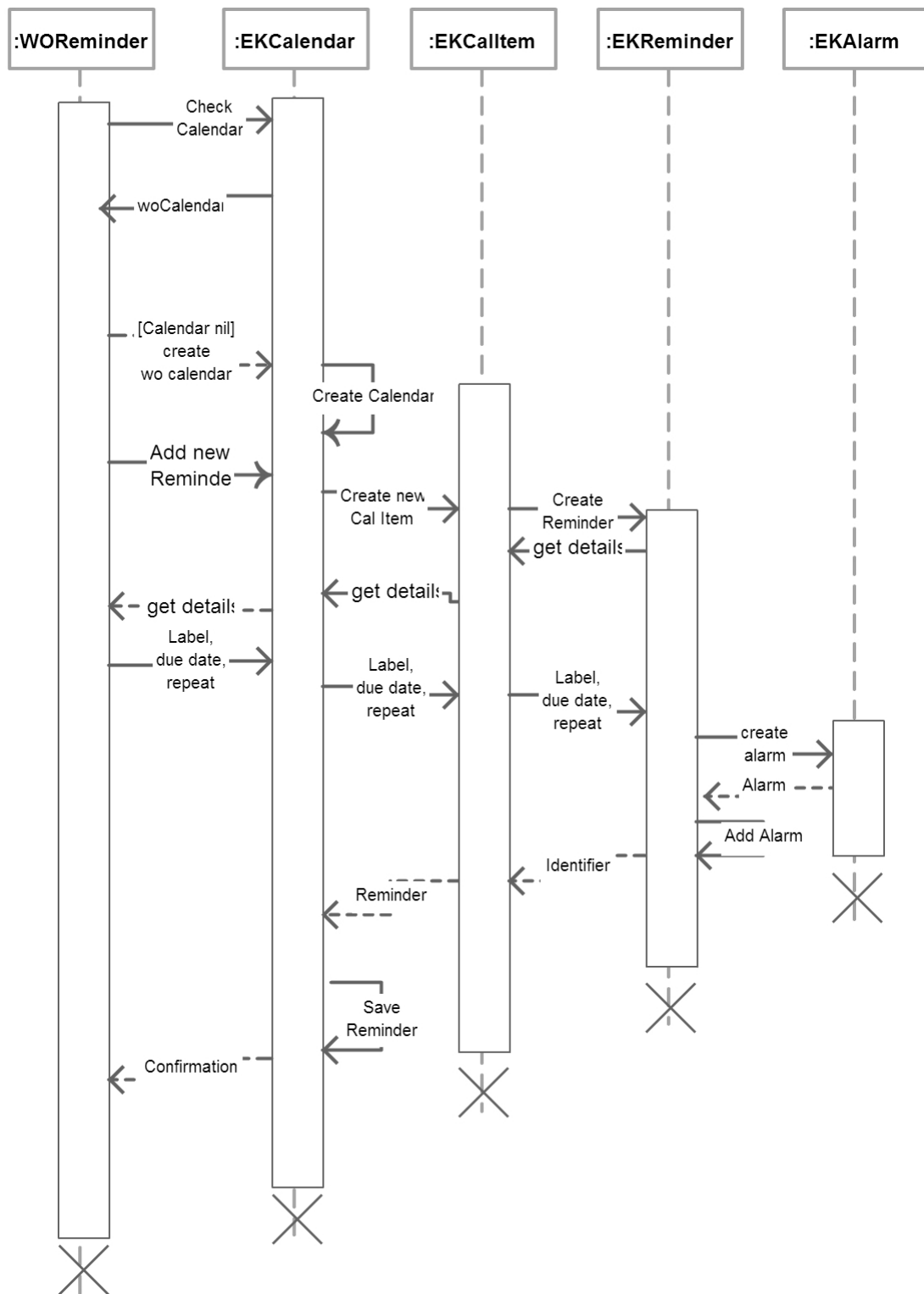


Figure 3.6: Sequence diagram for Adding new Reminder



### 3.6 State Diagrams

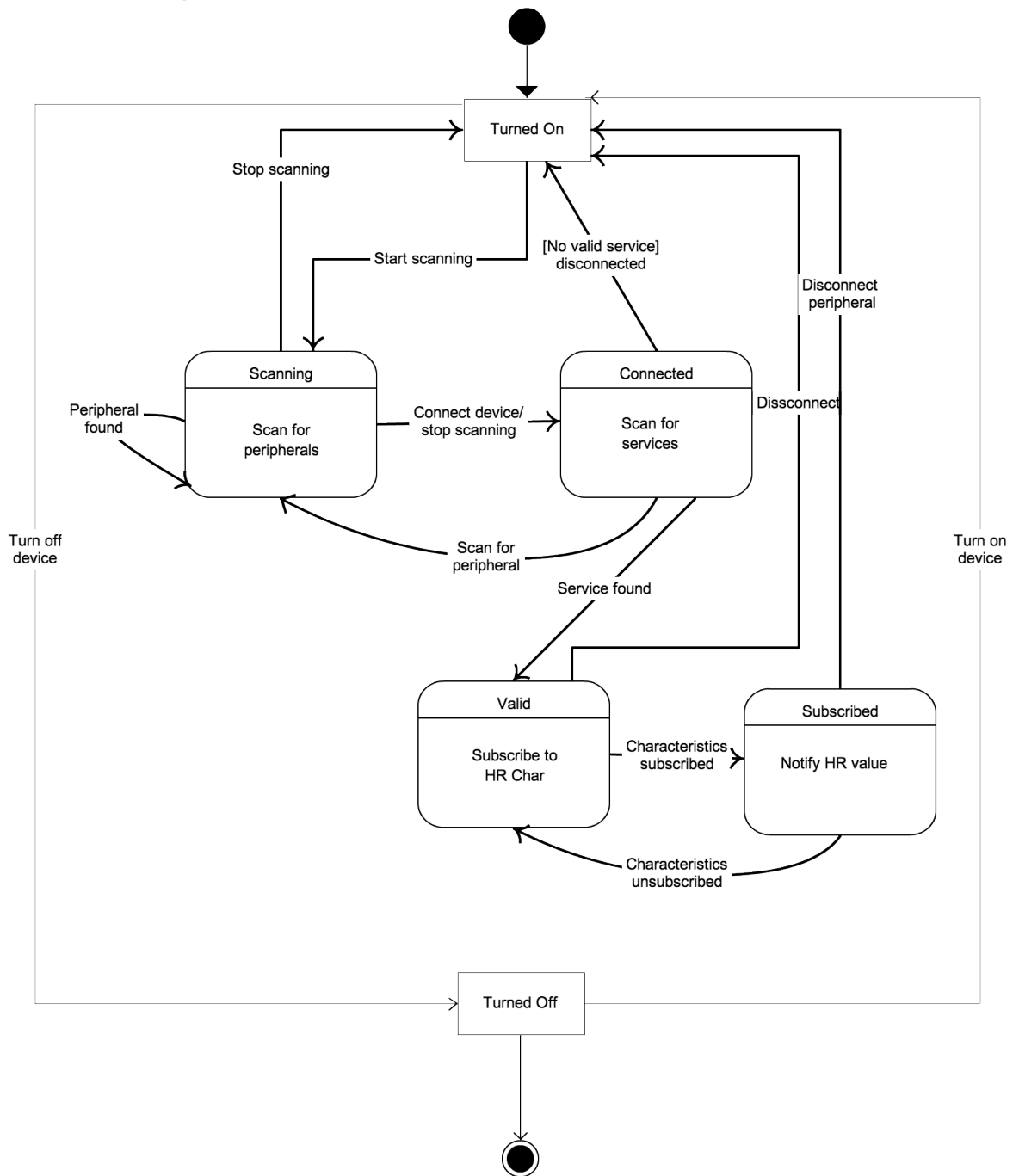


Figure 3.7: State diagram for Peripheral Object

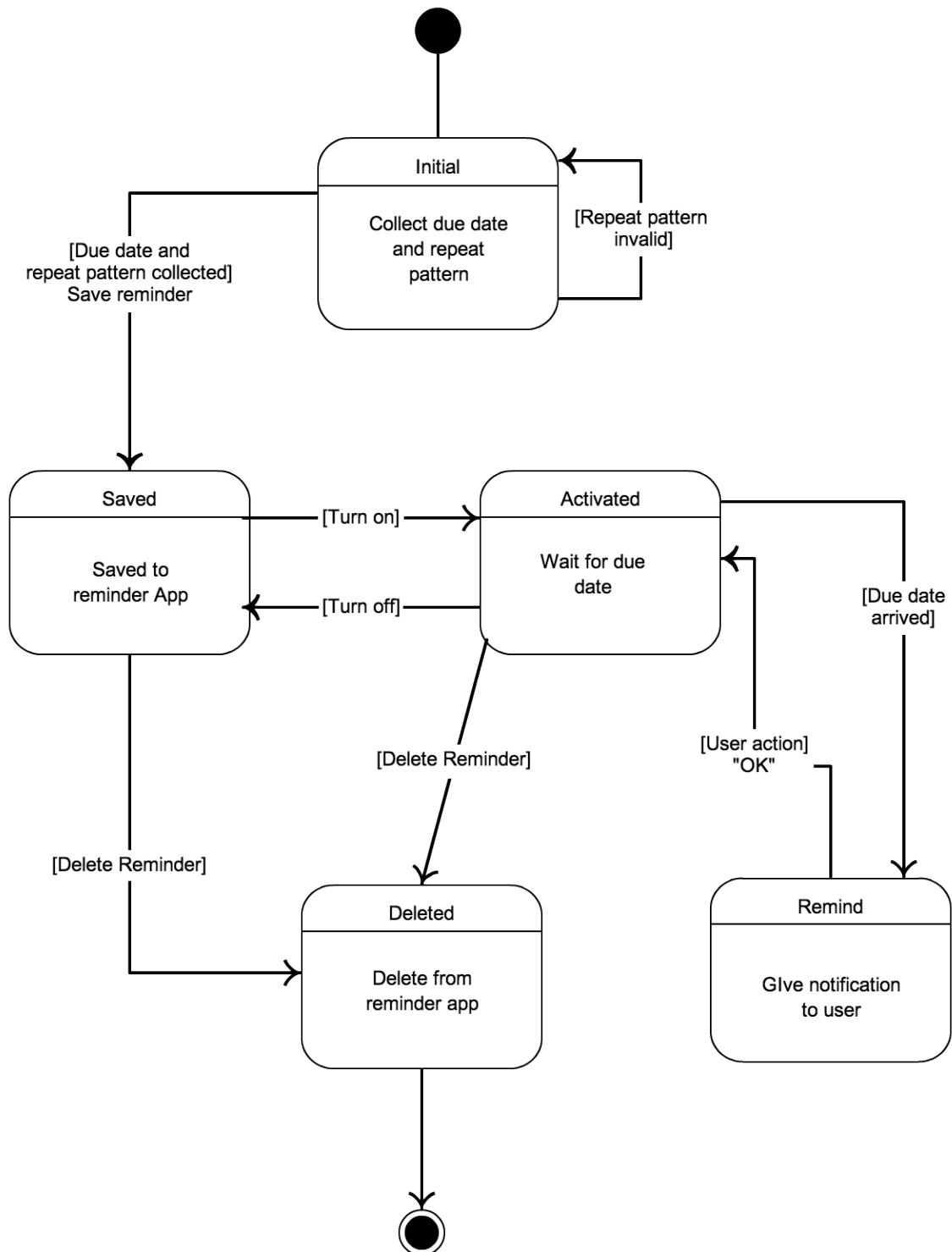


Figure 3.8: State diagram for Reminder Object

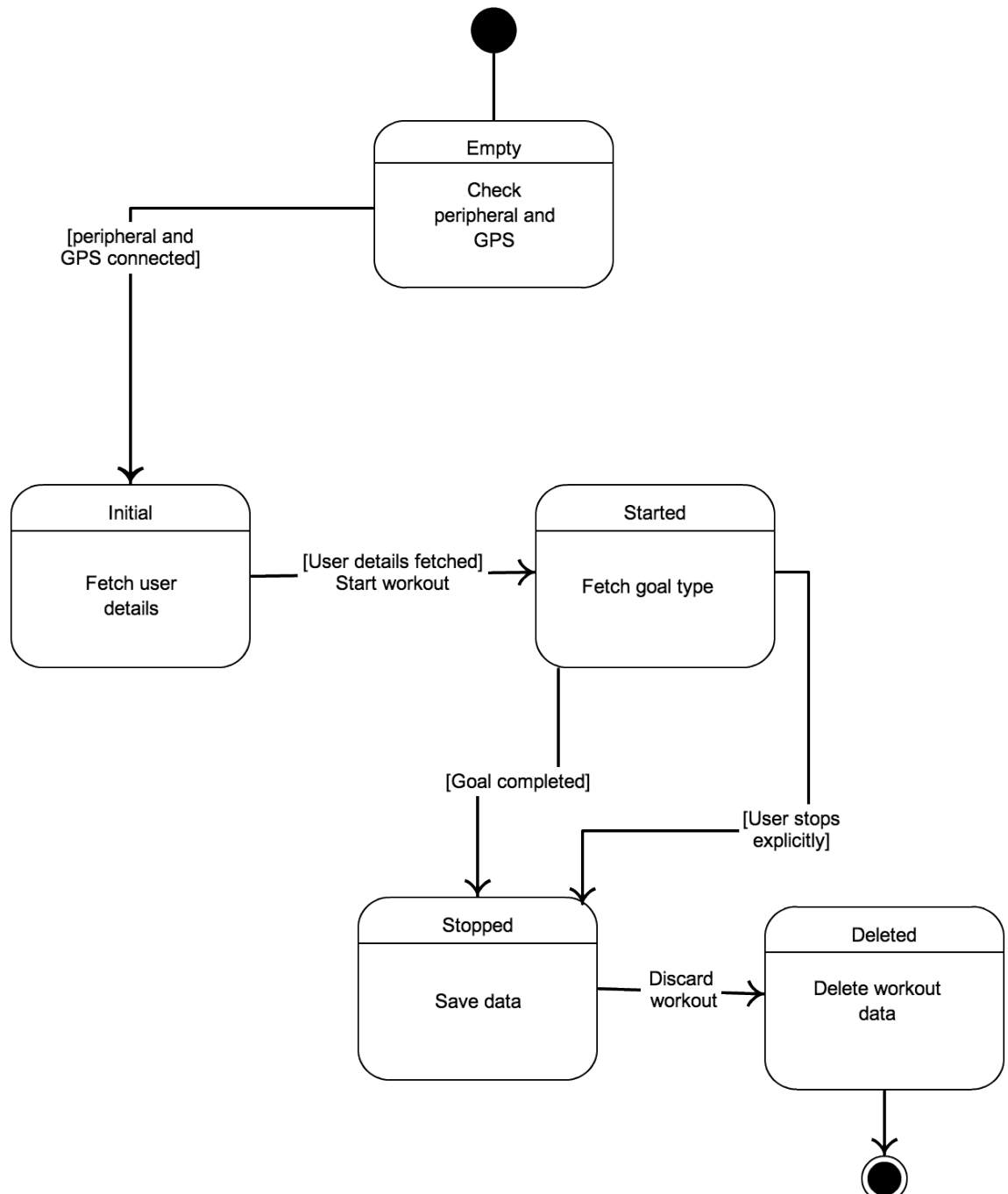


Figure 3.9: State diagram for Workout Object

### 3.7 Activity Diagram

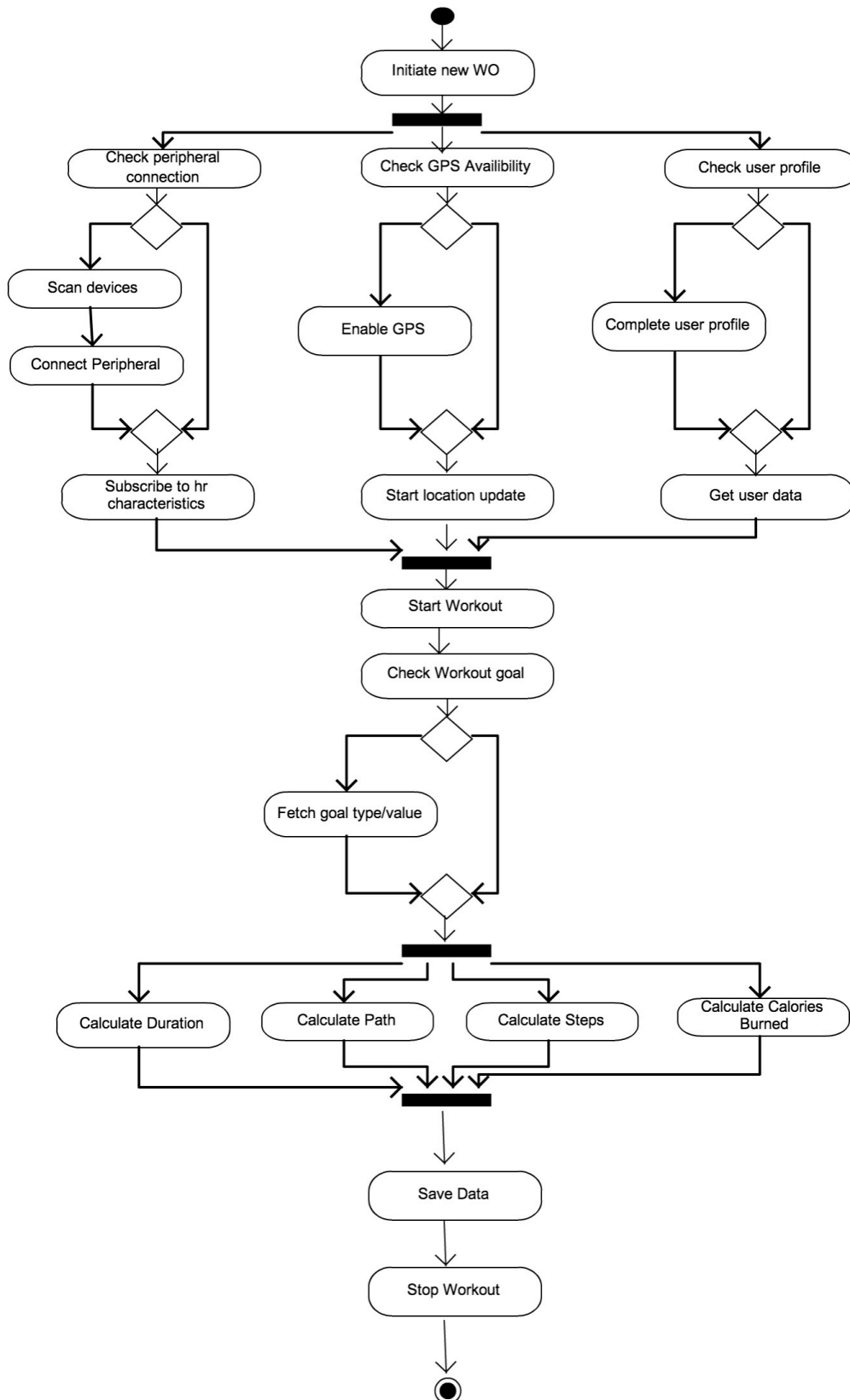


Figure 3.10: Activity diagram for Start Workout