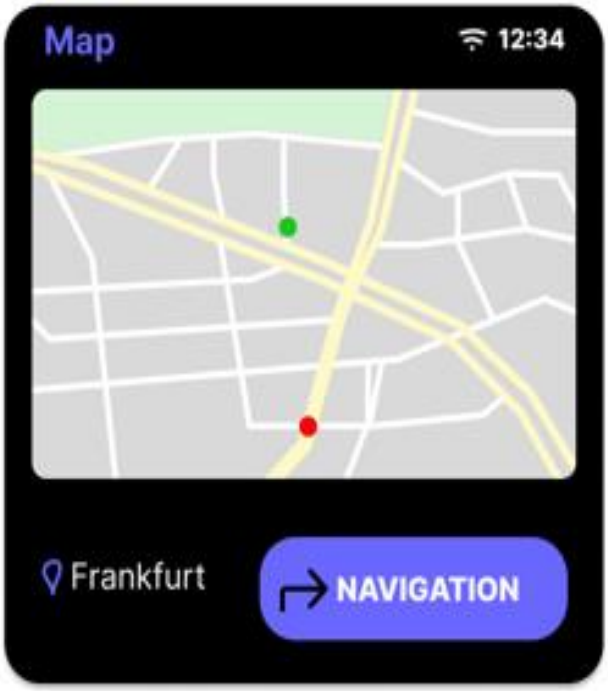
Karthik Bhat

6.1 Motion and Positioning Data – GPS

6.1.1 Introduction to GPS

The Global Positioning System, commonly known as GPS, is a satellite-based navigation system that provides precise positioning and timing information anywhere on Earth. This technology relies on a constellation of orbiting satellites that transmit signals to GPS receivers. And I have used this GPS Technology in our Smartwatch, which helps the User in many ways.This Smartwatch which is equipped with GPS, allows the User to track their precise location in real-time, for example running, hiking, or cycling, where the User can monitor their routes and distance 95escribes. It also allows the User to navigate themselves to a certain Location by giving directions. In the event of an emergency, the Smartwatch can provide accurate location information to emergency services. This feature enhances personal safety.



How does GPS work ??

GPS satellites circle the Earth twice a day in a precise orbit. There are 24 Satellites orbiting the Earth. Each satellite transmits a unique signal that allow GPS devices to decode and compute the precise location of the satellite. GPS receivers use this information to calculate a user's exact location. The GPS receiver measures the distance to each satellite by the amount of time it takes to receive a transmitted signal. With distance measurements from a few more satellites, the receiver can determine a User's position and display it to measure User’s running route, find a way home.

What is GPS Lock ??

To calculate your 2D position (latitude and longitude) and track movement, a GPS receiver must be locked onto the signal of at least three satellites. With four or more satellites in view, the receiver can determine your 3D position (latitude, longitude and altitude) and compute accurate position and time. This is called GPS Lock.

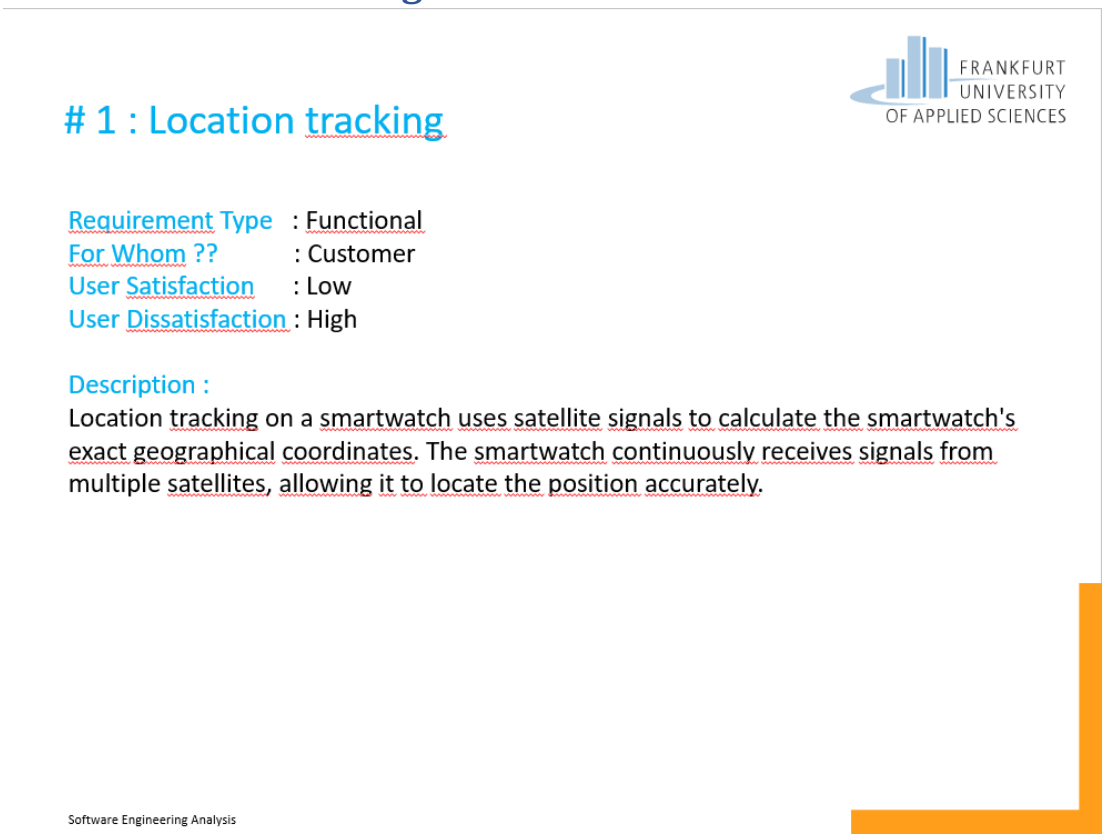
6.1.2 Use Case for Tracking Motion and Positioning Data – GPS

Use Case Name	Tracking Motion and Positioning Data - GPS
ID	1
Description	GPS in a Smart watch provides the User with accurate location data, enables navigation, and improves the overall User experience
Trigger	User trying to find his/her Location
Actors	User, GPS System
Pre-condition	User should have GPS enabled Device which is powered ON and fully functional
Post-condition	GPS System monitors and provides accurate Location data
Basic Flow	• User activates GPS application on the Smart Watch • The Watch establishes a connection with GPS Satellite • User searches for his/her Location or Destination • User starts the navigation to the Location.
Exceptional Flow	1) User experiences Signal loss: • Watch loses connection with the GPS Satellite because of Infra-structure like Tunnels, Skyscraper Building etc • Watch informs the User that the Location my not be accurate • It tries to reconnect with Satellite
Alternate Flow	2) System goes Offline Mode: • If the device is in Offline mode, The GPS system notifies the user that it may not have access to real-time map data. • The system informs the user about limited functionality due to offline mode.

6.1.3 Requirement Table

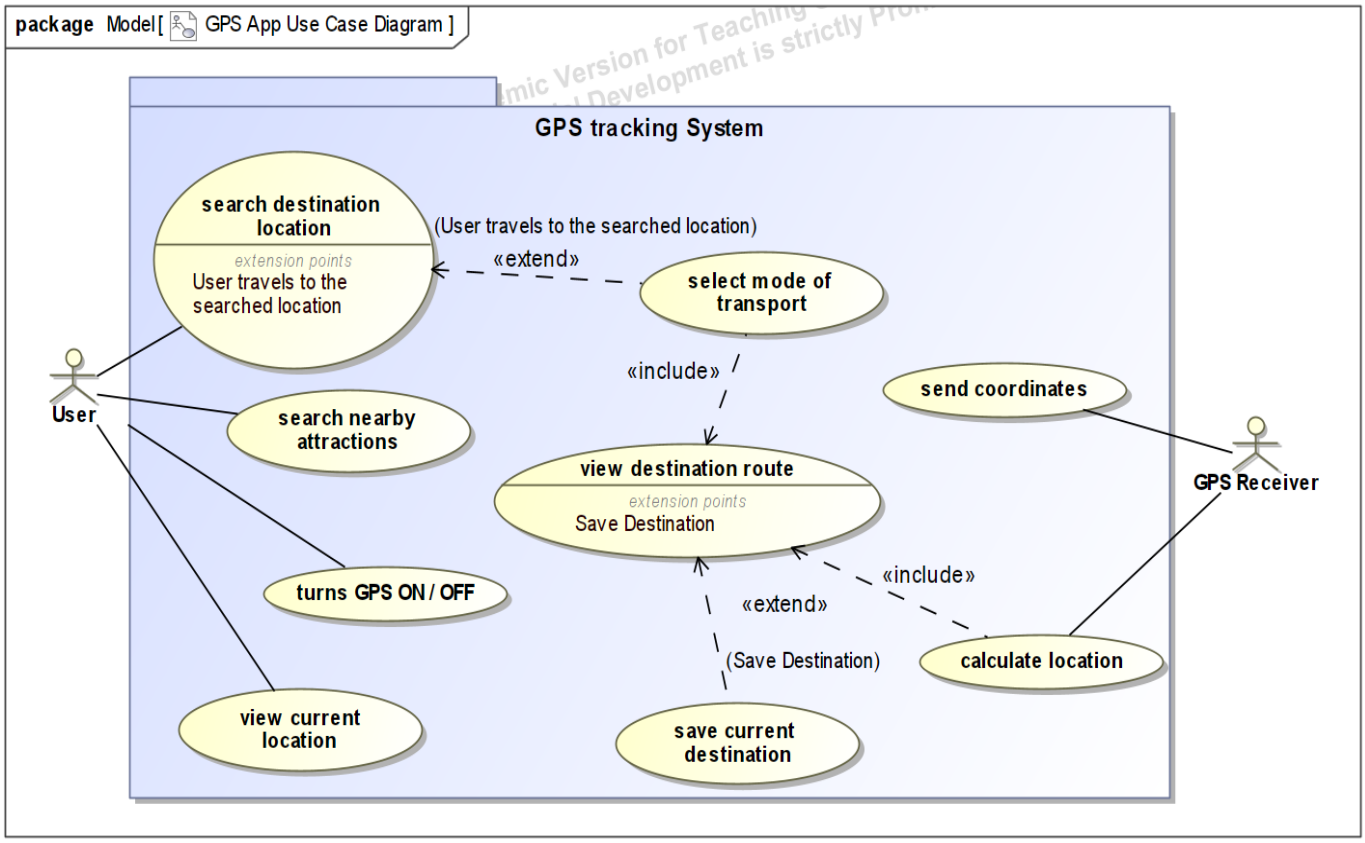
ID	Requirement	Requirement type	Business Value
1	Location Tracking	Functional	High
2	Map Diaplay	Functional	High
3	Navigational Assistance	Functional	Mid
4	Accracy and Precision	Non Functional	Low
5	Fast Respone Time	Non Functional	Low
6	Power Effcient	Non Functional	Mid
7	Weather Information	Non Functional	Mid
8	Location History	Non Fuctional	Mid
9	Offline Support	Non Functional	High
10	Location Sharing	Non Functional	Mid

6.1.3 Location Tracking



Out of all the Requirements from the Requirements Table, I selected Location Tracking and designed all UML Diagrams based on Location Tracking needs, because its one of the most functional and important Requirement.

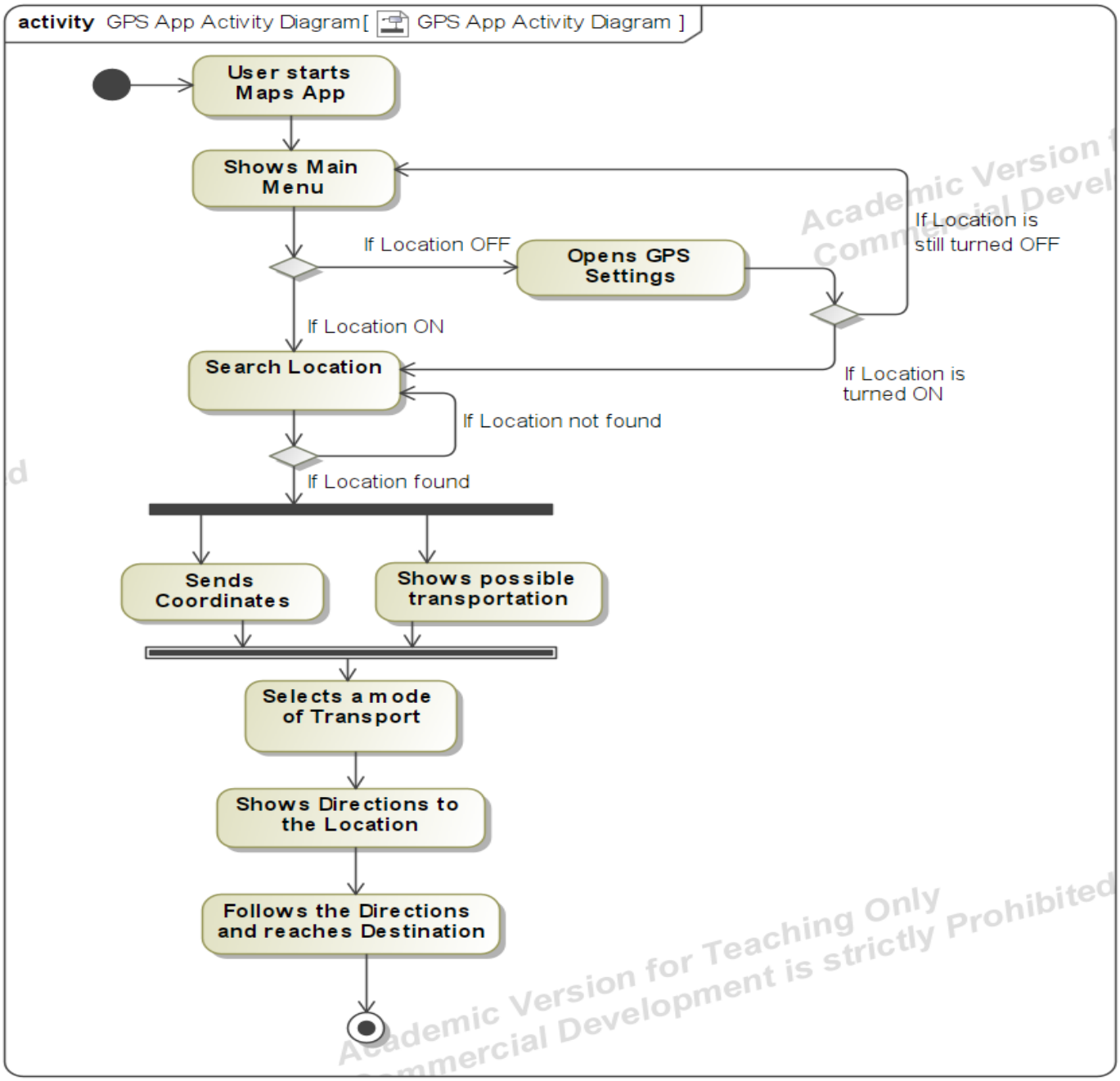
6.1.4 Use Case Diagram for Location tracking



This Use Case Diagram describes how the User would interact with the GPS System. The User can turn ON / OFF the GPS in his/her Device, view his/her current Location, search desination and also nearby Attractions etc. For example the User needs to travel to a particular Location. The User then can Search the Location. Afterwhich the User gets an option of choosing the mode of Transport, with which the User is going to reach the Location. After selecting the appropriate mode of transportation The User gets the suggested Route to this Location. After the User reaches the Location, the User simply can exit the Map. On the other hand we have to the GPS Receiver which Calculates location and sends coordinates.

Use case diagrams provide a representation of the system's functionalities. It is very helpful for stakeholders, Product Owners, developers, to understand the requirements of the GPS system on the smartwatch. Business value is often realized through the development of features. Use case diagrams act as a roadmap for implementing these features. These Features and other information are valuable for marketing the product in the market.

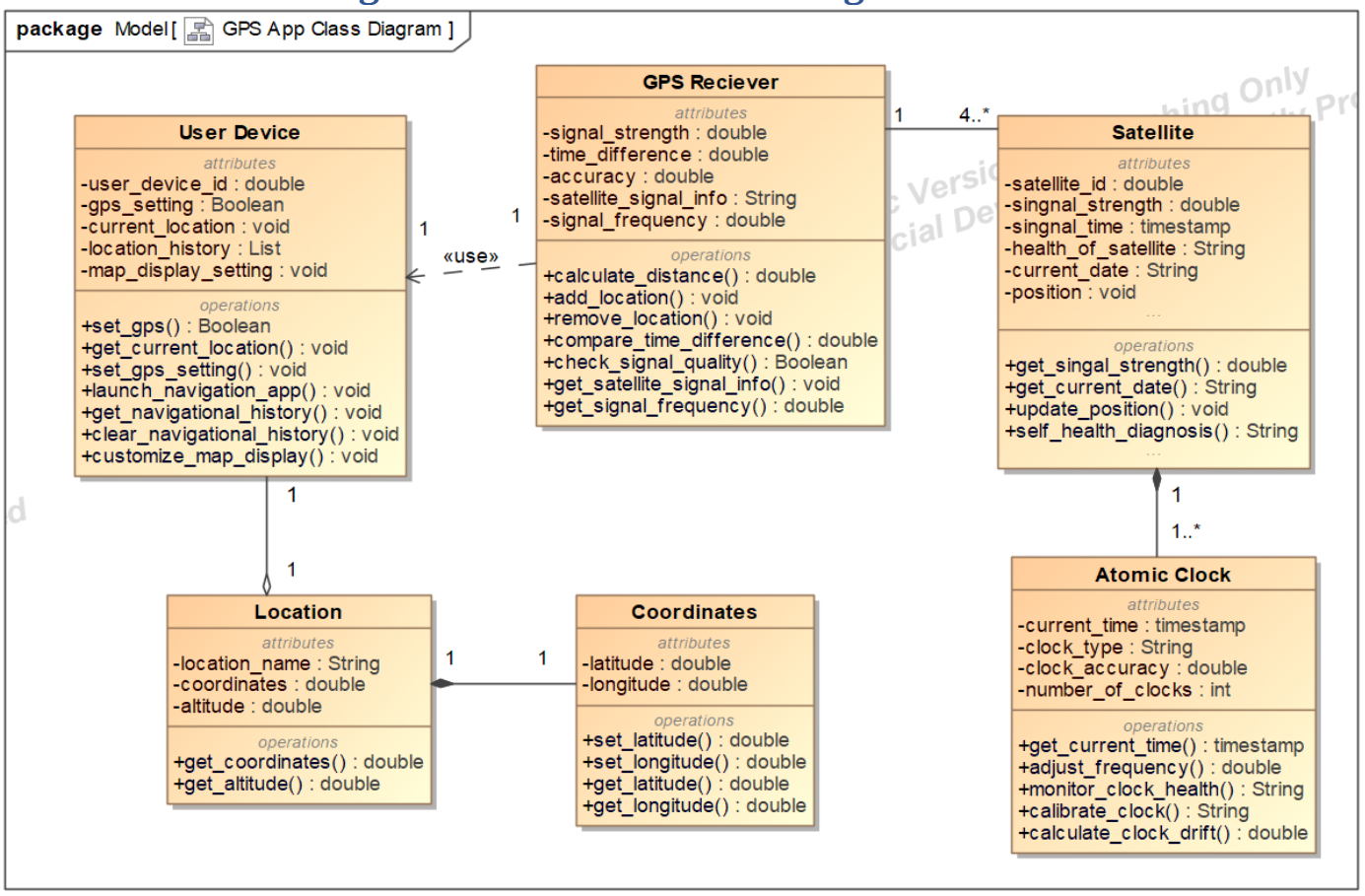
6.1.5 Activity Diagram for Location tracking



Based on the earlier Use Case Diagram, I designed a similar Activity Diagram. The activity diagram begins with the start state, when the User starts the Maps App. The Activity Diagram ends, When the User follows the given Directions and reaches the Destination.

This activity diagram provides a overview of the key steps and interactions involved with GPS System in the smartwatch. It helps in visualizing the flow of activities and events from user interaction to the presentation of location-related information.

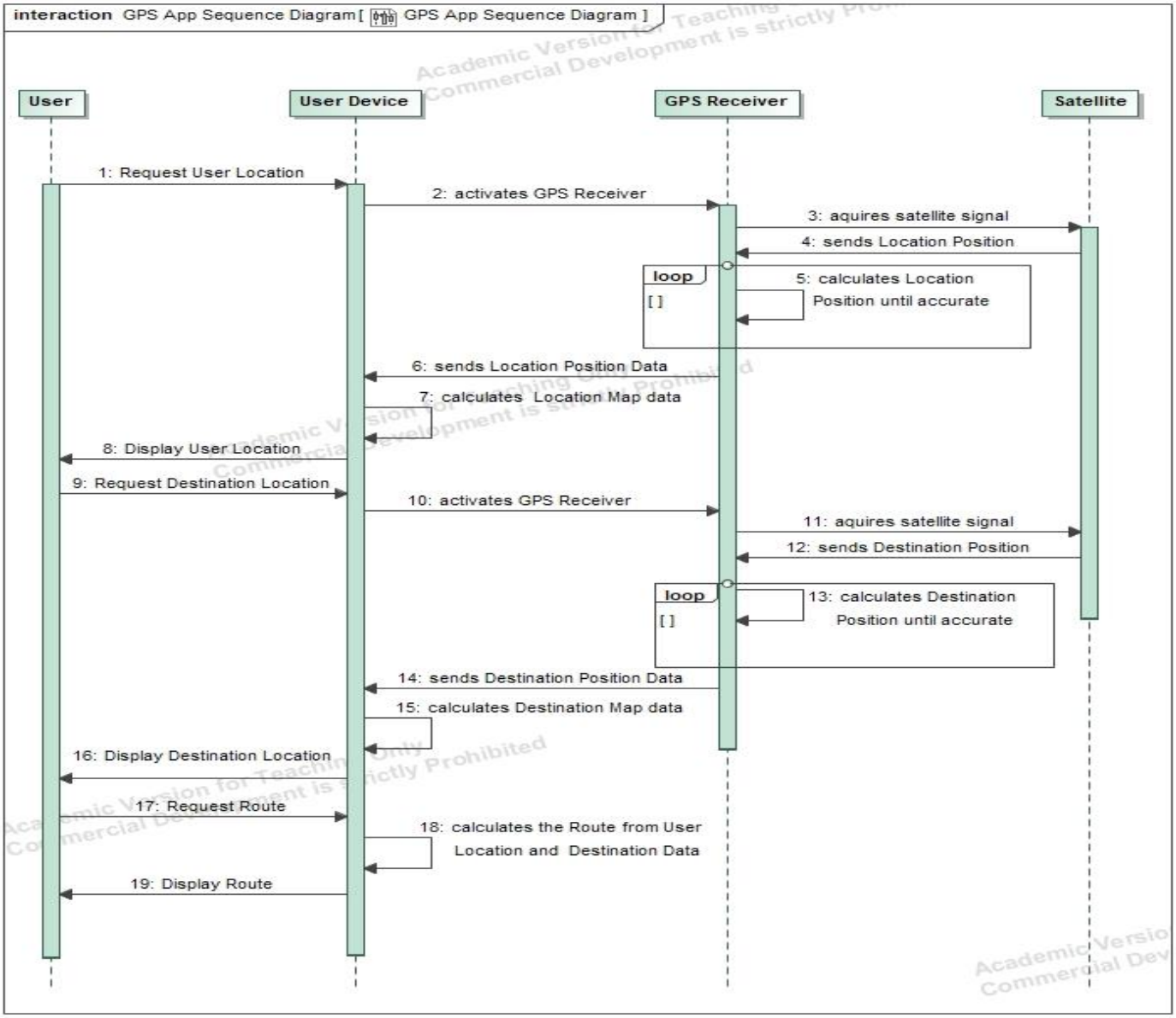
6.1.6 Class Diagram for Location Tracking



This class diagram for GPS in a smartwatch represents the relationships and structures of key classes involved in the GPS functionality. In this Class Diagram, there are 3 main Classes (User Device, GPS Receiver, Satellite)

- The Location Class is connected to Coordinate Class with the composition relationship with the Cardinality 1 , because every Location has one and only unique Coordinates and without Location Class Coordinate Class cannot exist.
- Every User Device has one Location, and instead of writing Location as a method of User Device Class, I connected User Device Class with Location Class with Inheritance relationship.
- The User Device Class is then connected to GPS Receiver Class, and GPS Receiver Class is further connected to Satellite Class with Association relationship.
- Inoder for the GPS System to be accurate, the GPS System has to be connected to atleast 4 or more Satellite, thy why the Cardinality for Satellite between GPS Receiver Class and Satellite Class is 4..*
- The Satellite Class is further connected to Atomic Clock Class, wherein every Satellite has atleast 1 Atomic Clock.

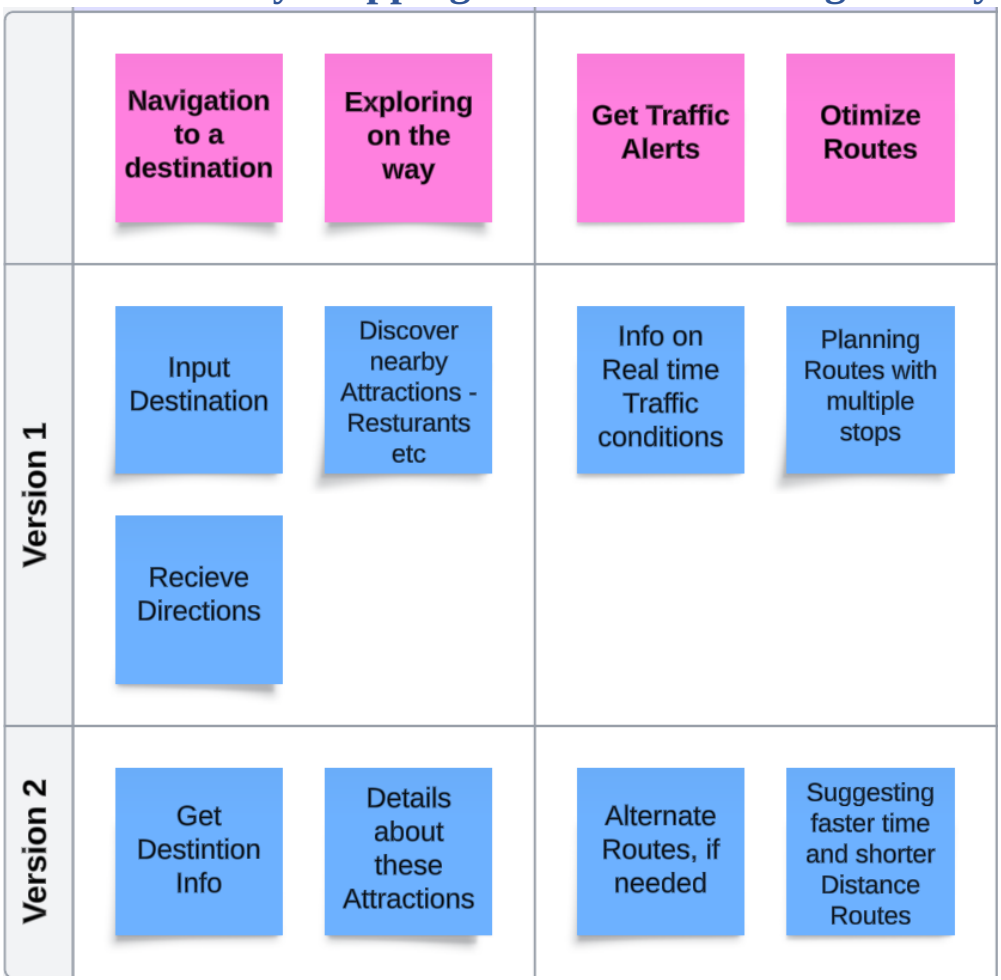
6.1.7 Sequence Diagram for Location Tracking



This Sequence Diagram for GPS in a smartwatch shows the interactions in chronological order of events between various actors involved in GPS functionality. This is a Sequence Diagram based on the earlier Class Diagram, wherein we can see what exactly happens when the User initiates and requests User and Destination Location and its Route. The Sequence Diagram has 3 Classes (User Device, GPS Receiver, Satellite) from previous Class Diagram and not all Classes, because the other Classes cannot be integrated in a Sequence Diagram.

6.2 Advanced Navigation System

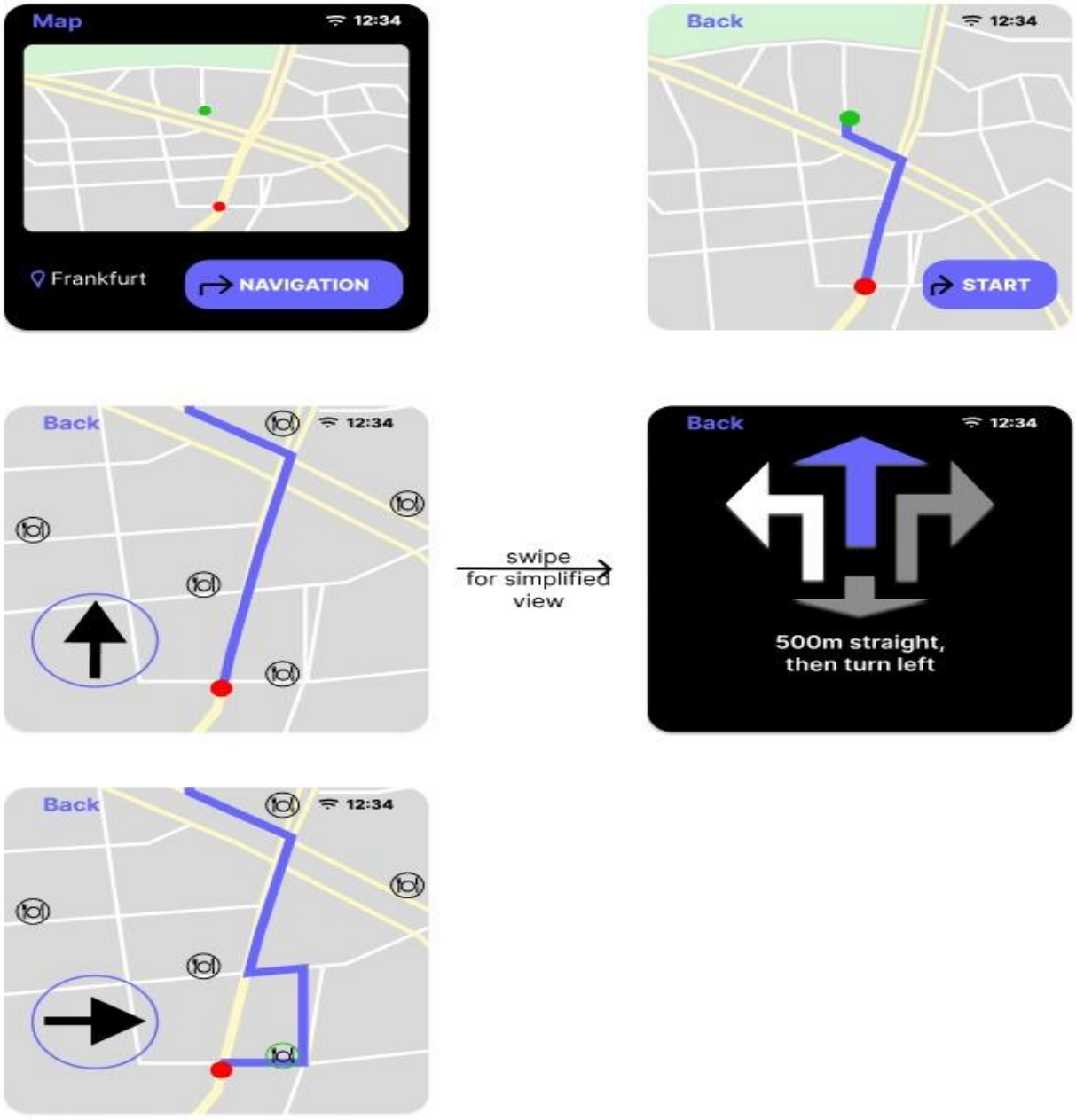
6.2.1 User Story Mapping for Advanced Navigation System



This User Story Map provides a visual representation of the User's journey and helps prioritize things based on User needs and preferences.

The new Feature of this Smartwatch is that, it gives you different options of nearby Attractions on your way to the Location. For example Lets say, You are travelling with your Family to a different City for a Vacation. You Start your Journey with the help of the Smartwatch giving you Directions to your Destination. On the way, the Smartwatch shows different nearby Attractions, one of which is Gas Station, and Suddenly you remember that you have low Fuel in your Vehicle and then you can fill your Gas Tank. Then on the way, the Smartwatch informs you that there is a Traffic Problem on the way, and it suggests you a better Route to your Destination, helping you save a lot of time and fuel. Then the Smartwatch gives you Recommendations and Info of nearby Restaurants on the way, where you can get good Food and etc. Most of the Smartwatches does not have this Feature, which makes this Smartwatch special and has a Unique Selling Point in the Smartwatch Industry.

6.2.2 UI Prototype for Advanced Navigation System

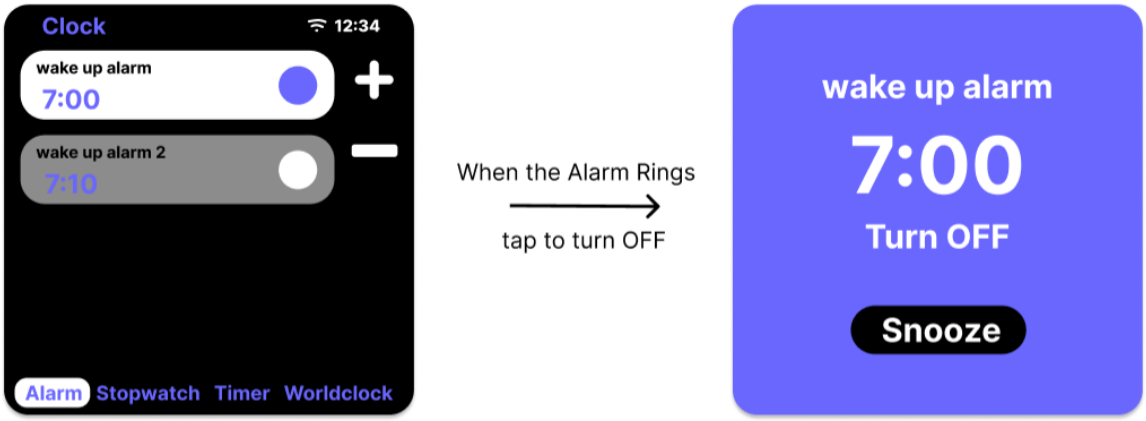


Based on the earlier User Story Mapping for the Advanced Navigation System, some UI Prototype were made. Just to get an Idea how the Advanced Navigation could look like. This is something new that I was working on and as its non Functional Requirement, not many Users might use this Feature, thats why no UML Diagrams were created, just UI Prototypes.

6.3 Clock App – (Timer, Stopwatch, Alarm)

6.3.1 UI Prototype for the Clock App

6.3.1.1 Alarm

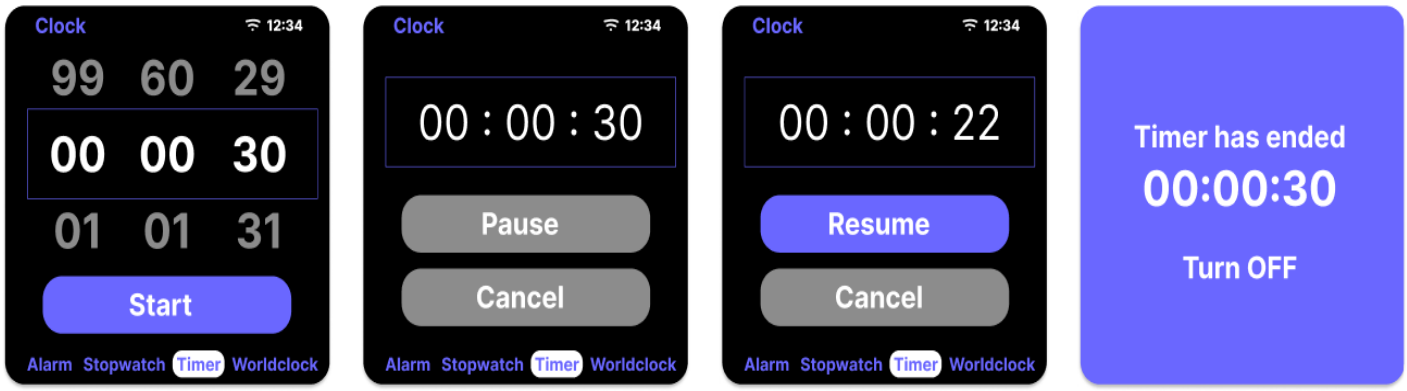


The main screen of the Clock App displays a list of existing alarms. Each alarm could be having further details such as the set time and alarm tone. Users can add a new alarm by tapping a + button, and delete with – button. Additional settings such as snooze duration, vibration for the alarm are also available. A save button confirms the alarm settings and adds the new alarm to the list. When the Alarm rings the User can tap the Turn OFF button.

6.3.1.2 Stopwatch

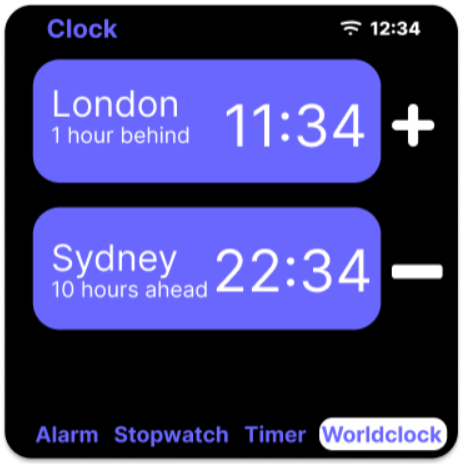
The stopwatch feature is accessible from the Clock App's main screen. The stopwatch has a interface that displays elapsed time and updates in real-time as the stopwatch runs. The User can start the Stopwatch just by clicking the Start button, there are also buttons for lap , restart , stop. A lap button allows users to record lap times during timing. To reset the time on Stopwatch, the User can click restart button.

6.3.1..3 Timer



The timer feature is also accessible from the Clock App's main screen. Users can set the desired countdown duration using a sliders for hours, minutes, and seconds. Start, pause, resume and cancel buttons are provided for controlling the timer. The remaining time is displayed and updating as the timer counts down. Users receive audible and visual notifications when the timer reaches zero, also by vibration if enabled. User can stop the Timer with the Turn OFF Button.

6.3.1.4 World Clock



The main screen of the world clock displays a list of cities, time zones, and time delay. Users can add a new city to the world clock by tapping a + button, and remove with - button. An option is provided to automatically detect and add the user's current location to the world clock.
* World Clock not featured in Use Case Diagram, because 3 Use Case Diagrams for Alarm, Stopwatch, Timer are already made.

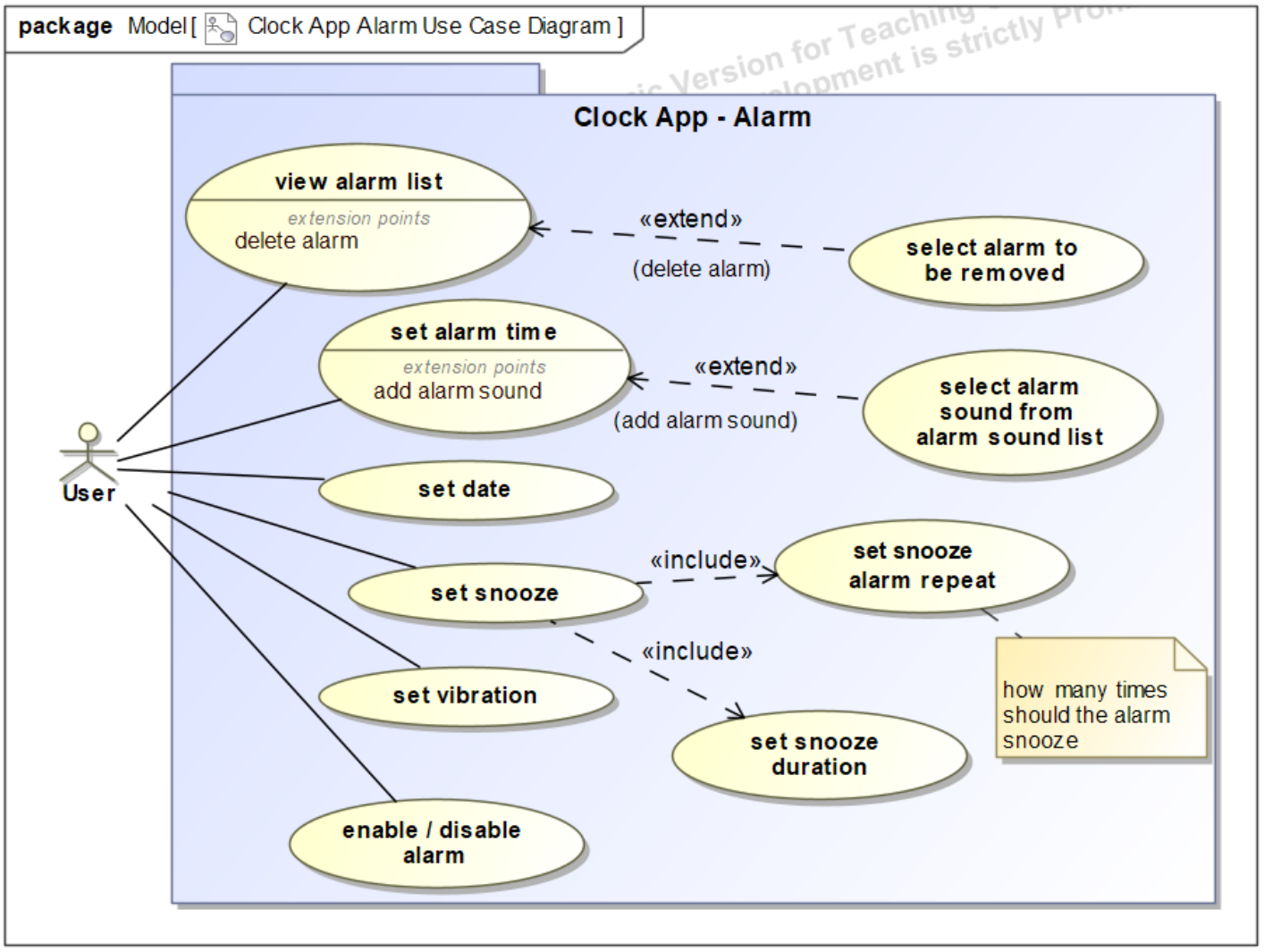
6.3.2 Use Case for the Tracking Time

Use Case Name	Tracking Time with Timer, Alarm, Stopwatch, World Clock
ID	2
Description	The Clock App has alot of features like Timer, Stopwatch, Alarm etc. It is a valuable tool for various time-related activities such as waking up, cooking, exercising, and more
Trigger	Timer - user sets a specific duration and then starts the countdown. Alarm - user sets for a specific time in the future. When the current time matches the set alarm time, the alarm ring. Stopwatch – user wants to measure the elapsed time of an event.
Actors	User
Pre-condition	The Clock App is installed and accessible on the user's smartwatch
Post-condition	App display updates in real-time to show elapsed time, time remaining for Alarm to ring etc
Basic Flow	• User starts Clock App on smartwatch • User starts time tracking (Turns Stopwatch / Alarm / Timer ON) • System shows elapsed time on Stopwatch / existing Alarms / remaining time on Timer • User can stop or lap times in Stopwatch / edit or delete Alarms / pause or cancel time in Timer
Alternate Flow	1. User cancels/stops time tracking • The Clock App resets time and goes to Main menu • The System also gives the User opportunity to track time again • User can start time tracking or close the Clock App

6.3.3 Requirement Table

ID	Requirement	Requirement type	Business Value
1	Start / Stop time tracking	Functional	High
2	Recieve Notification	Functional	Mid
3	Sharing / Exporting Time date	Non Functional	Mid
4	Manual Time Entry	Non Functional	Low
5	Integration with Calendar	Non Functional	Mid
6	Compatibility to other Devices	Non Functional	High
7	Bed Time Mode	Non Functional	Mid
8	Event Countdown	Non Functional	Low
9	Multiple Time Zone Display	Non Functional	Mid

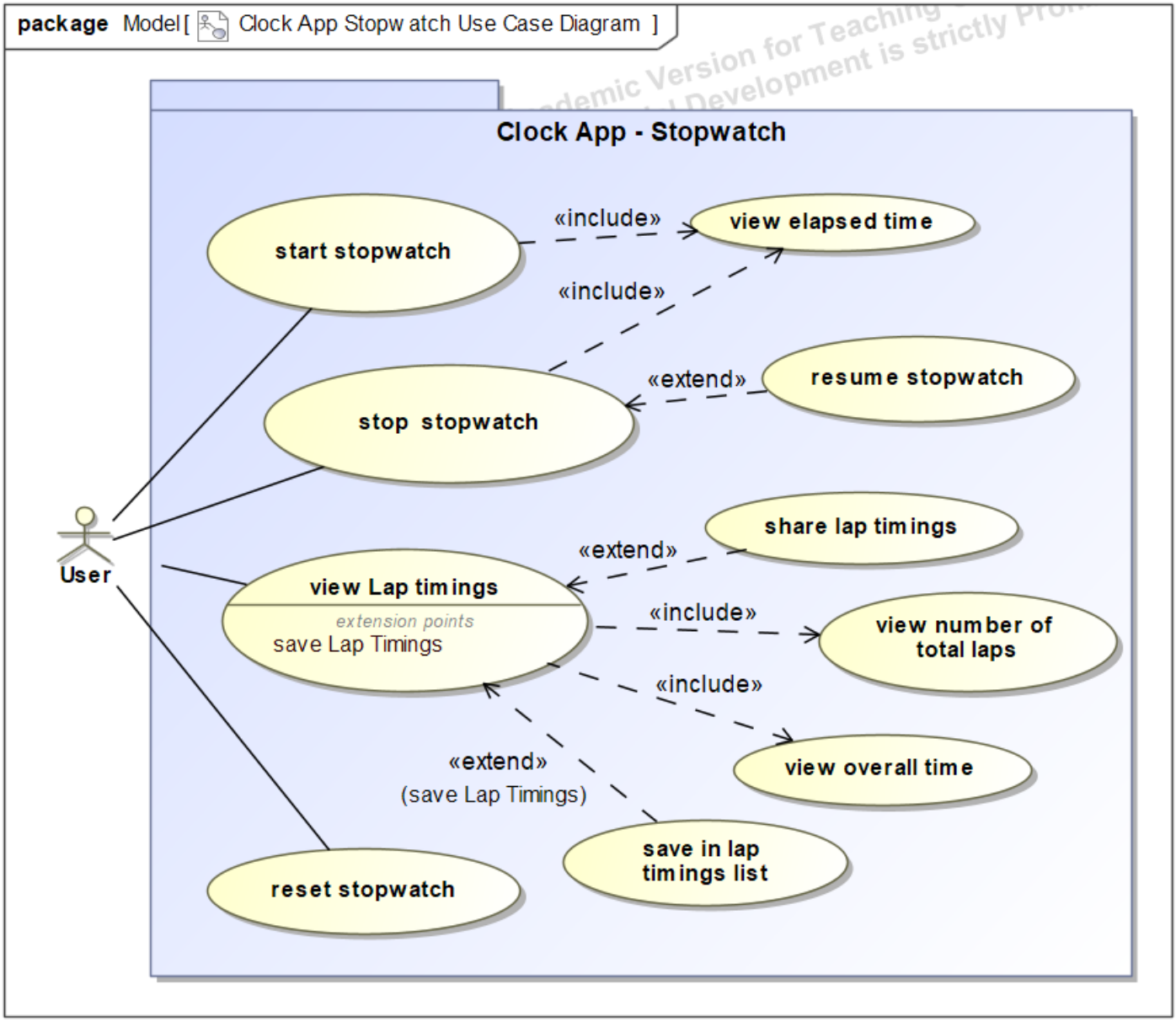
6.3.4 Use Case Diagram for Alarm



The User can do multiple things in our Alarm system in smartwatch like setting Alarm time, date, snooze and vibration also the User can view the Alarm list and enable or disable alarm and more. If the User decides to set Snooze on the Alarm, he/she has to also set the Duration of snooze and also set how many times it should snooze, that's why they have include relationship.

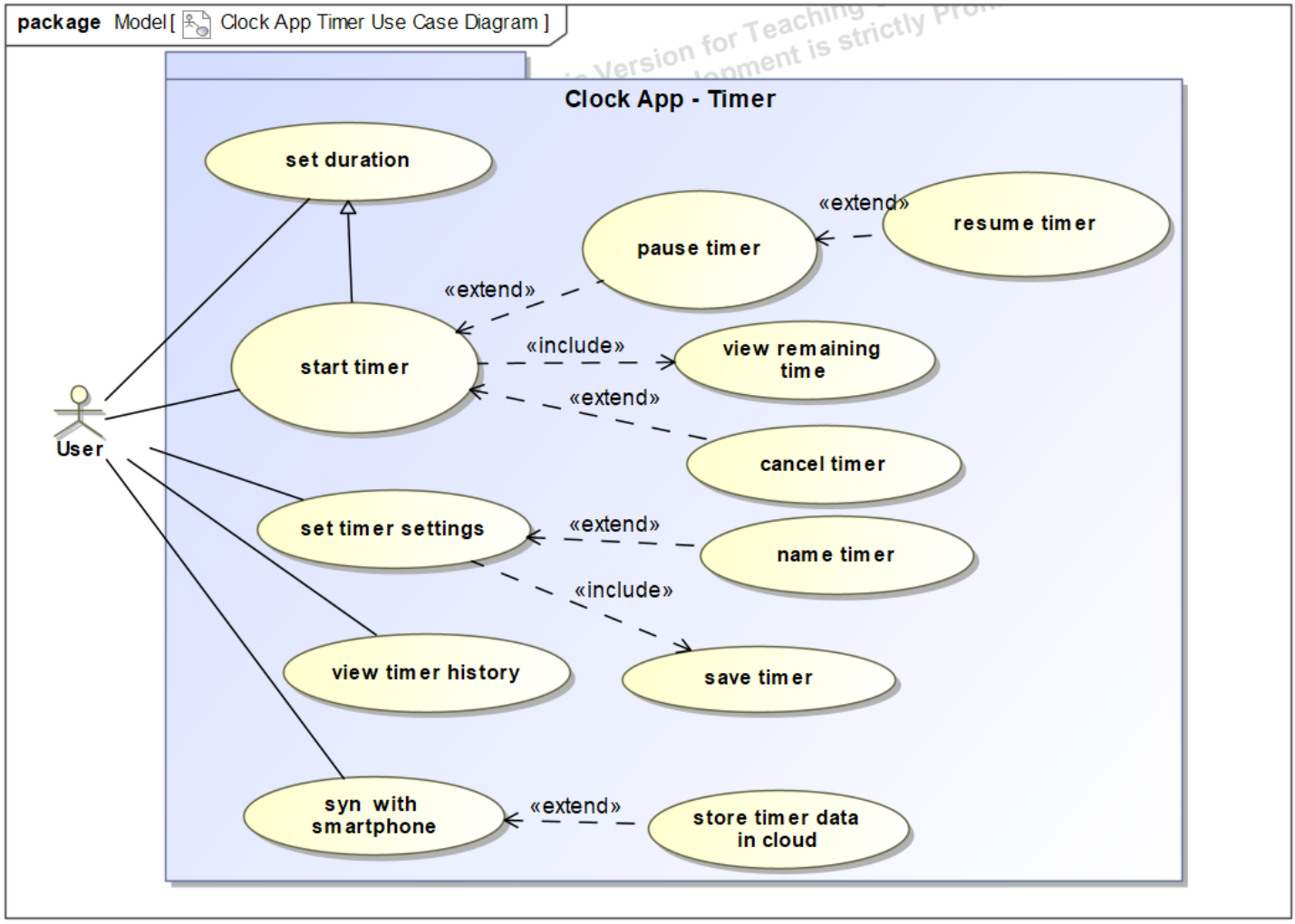
These are the main Use Cases that the User needs to use Alram.

6.3.5 Use Case Diagram for Stopwatch



The User can start, stop and reset the Stopwatch. T the Stopwatch shows overall time and number of total laps everytime the User views his/her Lap Timings, thats why i used include relationship. Also the User has the option to further share the lap timings and also save them in the lap timing list.

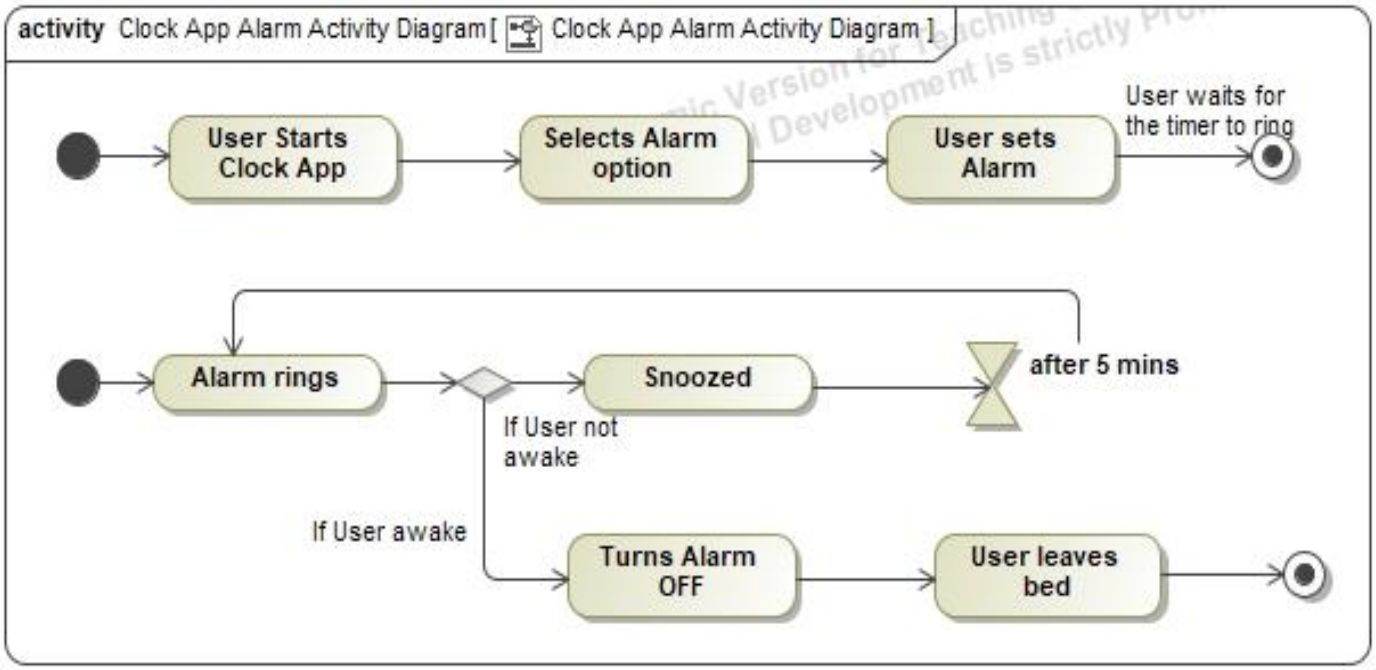
6.3.6 Use Case Diagram for Timer



The User can start timer and further pause and cancel the timer, the Timer always shows the remaining time to the User. Not only this, but the User also can set Timer duration, Timer Settings and more. A new feature that is pretty cool is that, the Timer allows the User the sync the timer Data with his/her Smartphone also helping in storing data in cloud.

6.3.7 Activity Diagram for Tracking Time

6.3.7.1 Alarm

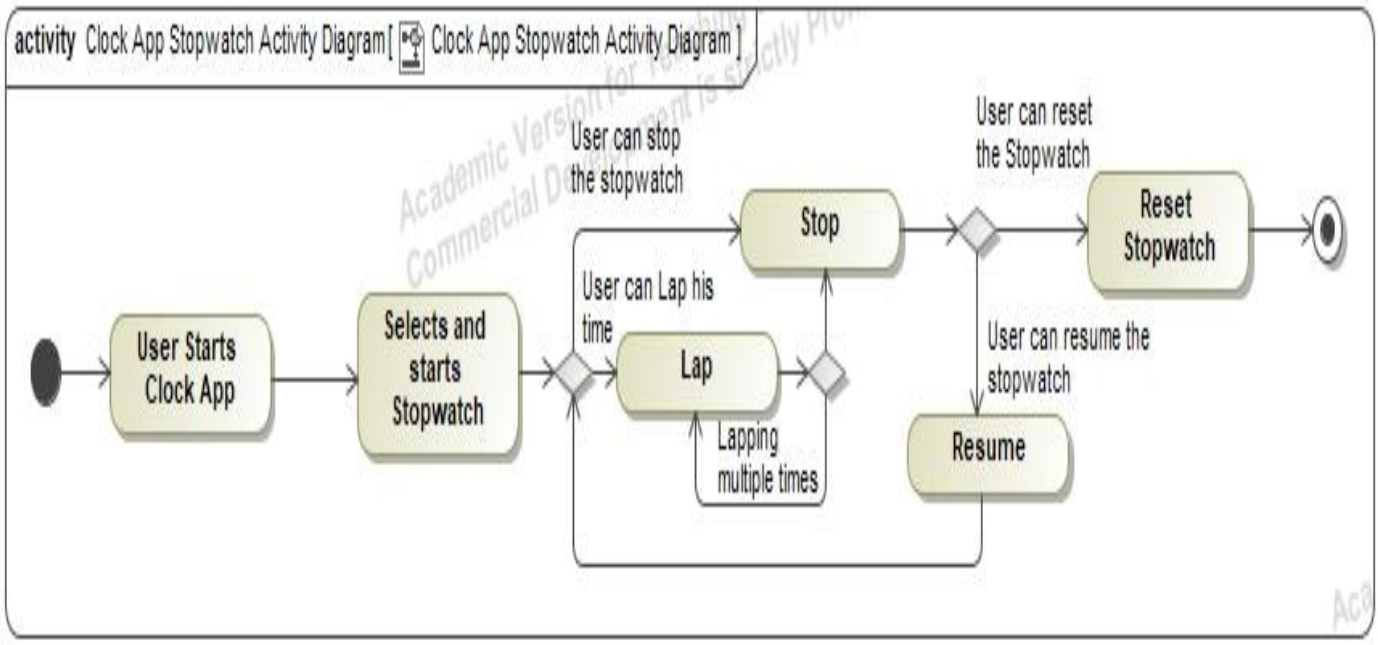


This activity diagram provides a visual representation of the steps involved in setting and managing alarms in the Clock App, guiding users through the process from configuration to alarm triggering and response.

In the first Activity Diagram, it basically shows how the User can set the Alarm and then waits for the Alarm to ring.

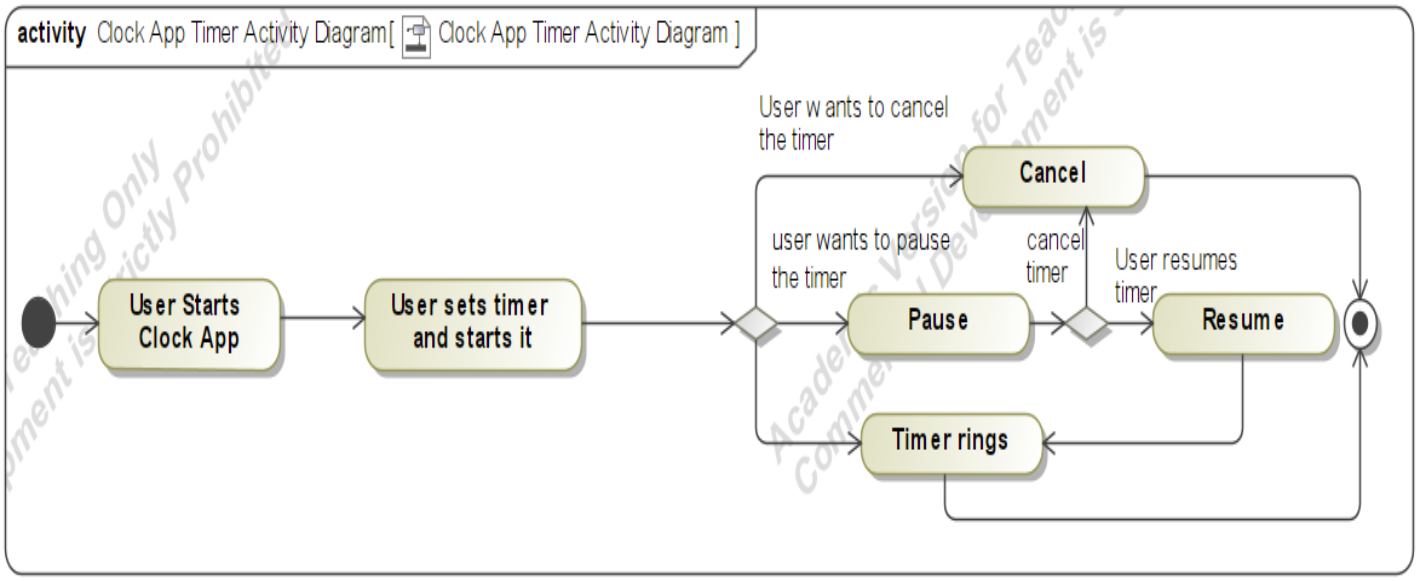
In the second Activity Diagram, it shows the Activity what happens after the Alarm rings. Lets say the User has set the snooze time to be 5 mins. After the Alarm rings, if the User is awake the User has the option to turn the Alarm OFF or if the User is still not awake User has a option to snooze it. As the User set the snooze time of 5 mins, after 5 mins the Alarm will ring again and this goes on in loop, The loop ends when the User turns OFF the Alarm.

6.3.7.2 Stopwatch



The above Activity Diagram is for Stopwatch. After the User starts the clock app he/she starts the Stopwatch. After starting the Stopwatch, the User can either stop the stopwatch or lap his/ her time. The User can then Lap the Stopwatch as many times as the User wishes and this could go on in loop. The loop ends when the User stops the Stopwatch further the User can also resume the Stopwatch. But at last the User has to Stop the Stopwatch, in order to get the timings. If the User wishes the start the Stopwatch, User has to just reset the Stopwatch.

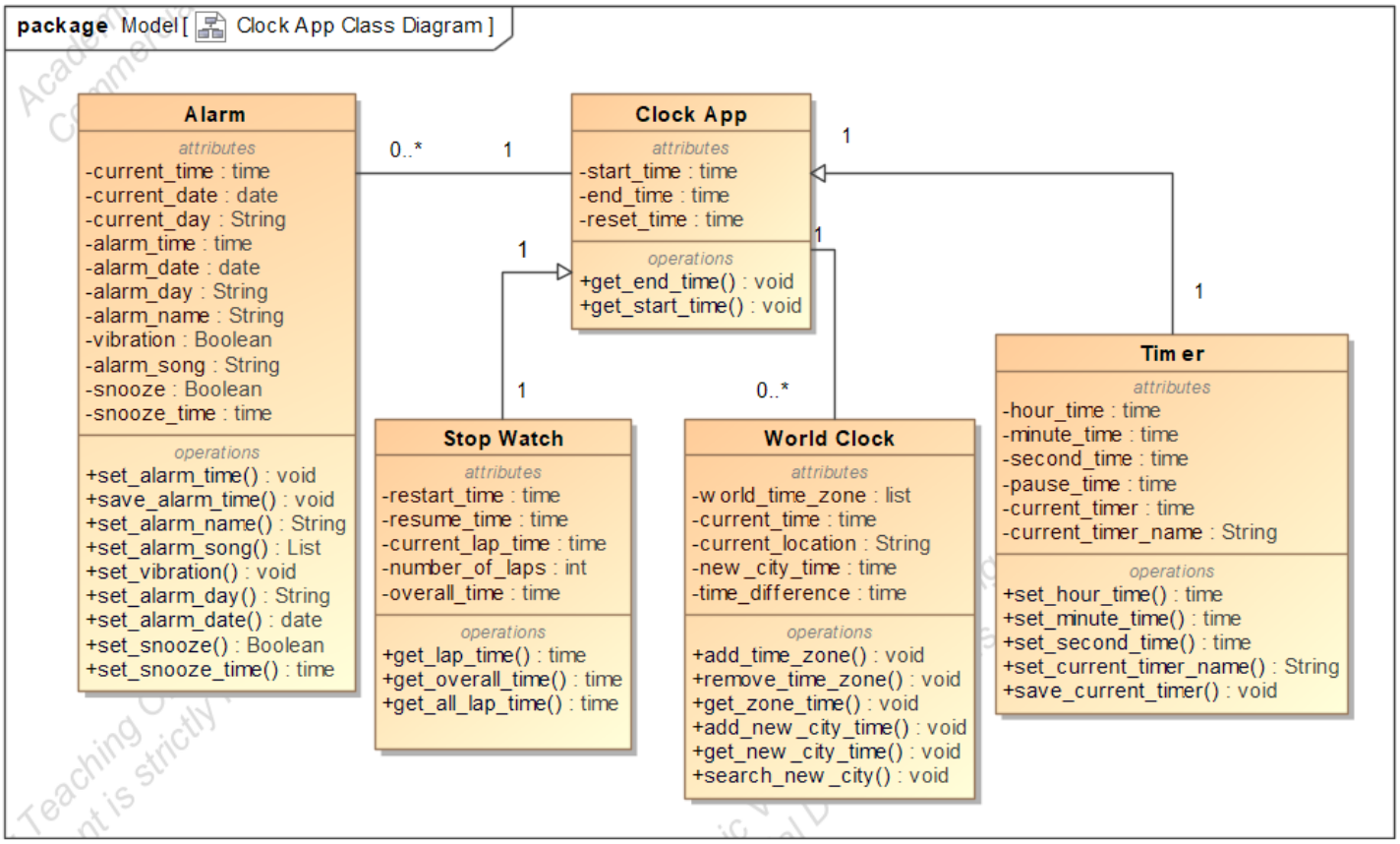
6.3.7.3 Timer



This is a Activity Diagram for Timer. After the User starts the Clock App , User then first sets the timer and then starts it., Then 3 things could happen.

1. The Timer rings and the User can again set a new Timer
2. The User decides to pause the Timer. The User then has 2 options, either to cancel the timer or resume it and let the Timer to ring.
3. The User also can directly cancel the Timer without resuming the Timer.

6.3.8 Class Diagram for Tracking Time



This is a Class Diagram for Tracking Time. There are 5 main Classes (Clock App, Alarm, Stop-watch, World Clock and Timer)

- All the classes are connected to Clock App Class, because the User could use all the Time tracking applications only via the Clock App.
- The Alarm Class and World Clock Class are connected to Clock App via association relation, wherein the User can have 0 or more alarms and World clock time zone that's why the cardinality of these classes is 0..*
- The Stopwatch Class and Timer Class are connected with Clock App with generalization (inheritance), where the Stopwatch inherits methods and attributes like – end_time and get_end_time , also same with the Timer Class, where it inherits attributes and methods like – start_time, end_time , get_end_time.

No sequence Diagram was made, because these Classes have no sequence of occurrence.