
System Combined Requirements and Design Document (SCRD)

for

System for Intelligent eBikes (SIE)

Version 1.0 approved

3 December 2019

Table of Contents

Introduction	4
Purpose	4
Document Conventions	4
Intended Audience and Reading Suggestions	4
System Scope	4
References	5
Overall Description	5
System Perspective	5
System Functions	5
User Classes and Characteristics	6
Boundary Class: WebsiteMainGUI, MobileAppMainGUI, BikeMainBUI	6
Control Class: SystemManager	6
Entity Class: User	6
Operating Environment	6
Design and Implementation Constraints	6
Assumptions and Dependencies	6
External Interface Requirements	7
User Interfaces	7
WebsiteMainGUI	7
MobileAppMainGUI	7
BikeMainGUI	8
WebsiteRideHistoryGUI	8
MobileAppRideHistoryGUI	9
WebsiteStatisticsGUI	9
MobileAppStatisticsGUI	10
FindMyeBikeGUI	10
WebsiteConnectGUI	11
WebsiteEditConnectGUI	11
MobileAppConnectGUI	12
MobileAppEditConnectGUI	12
MobileAppPostGUI	13
Hardware Interfaces	14
Software Interfaces	14
Communications Interfaces	14

Requirements	15
Functional Requirements	15
Functional Statements	15
Use Case Diagrams	17
Use Case Specifications	21
Nonfunctional Requirements	41
Performance Requirements	41
Safety Requirements	41
Security Requirements	41
Software Quality Attributes	41
Analysis	41
Activity Diagrams	42
Registration Subsystem	42
User Statistics Subsystem	43
eBike Integrity Subsystem	44
Safety Features Subsystem	45
Security Subsystem	45
Biking Control Subsystem	47
Class Model Analysis Level	48
Object Model Analysis Level	49
Use Case Realizations Analysis Level:	50
Design	54
Classes Design Level	54
6.2 Use Case Realizations at Design Level	55
State Machines	59
eBike Integrity Subsystem State Machines	59
6.2.2 Safety Features Subsystem State Machines	60
Implementation	61
System for Intelligent eBikes Deployment Model	61
System for Intelligent eBikes Component Model	62

1. Introduction

1.1 Purpose

The System for Intelligent eBikes provides a premium product for eBike Users. The system allows Users to register their eBike, track their eBike usage statistics, and connect to fellow eBike Users. The eBike also has safety features that will keep the rider's eBike in a well working condition, help them while on the road, and help prevent theft. This system includes User interfaces on a website, a mobile app, and the eBike itself. This SCRD is inclusive of all of the aforementioned subsystems.

1.2 Document Conventions

The following standards and typographical conventions will be followed in this SCRD:

- All titles and subsystems will be bolded.
- Classes will be italicized.
- W - Website
- MA - Mobile App

1.3 Intended Audience and Reading Suggestions

This document is intended for individuals participating in the design and deployment of the SIE. It is not intended for users. It is recommended that developers, project managers, marketing staff and testers begin on Section 1.4 System Scope in order to understand the depth of the project. Marketing Staff will find the most use of Section 2.2 System Functions, 2.3 User Classes and Characteristics, and 2.4 Operating Environment. In these sections, the Marketing Team will understand what they are marketing and who they are marketing to. It is recommended that project managers, developers and testers read the entire document in depth.

1.4 System Scope

The System for Intelligent eBikes Software will consists of five subsystems: Registration, User Statistics, eBike Integrity, Safety Features, Security, Biking Control, eBike Connect. The Registration Subsystem will allow Users to register, edit, and delete their eBike account on a website or mobile app. The User Statistic Subsystem will track the User's eBike data, aggregate the data, and display the User's statistics on the website, mobile app, and eBike interface. The Bike Integrity subsystem includes the sensor management and notification system that keeps the User alert on their bike integrity. This includes tire pressure, battery life, and motor service. The eBike's Security Subsystem monitors the eBike's lock, alerting the User if their is any misuse, and keeps a real time location tracker of the eBike. The Biking Control Subsystem allows the User to monitor the amount of assistance they would like. The eBike Connect allows the User to make, edit, and delete their Connect account. Once a Connect account is created, the User can create posts and

challenges. These subsystems will work together to create make the eBike not only a product, but a modern and complete experience.

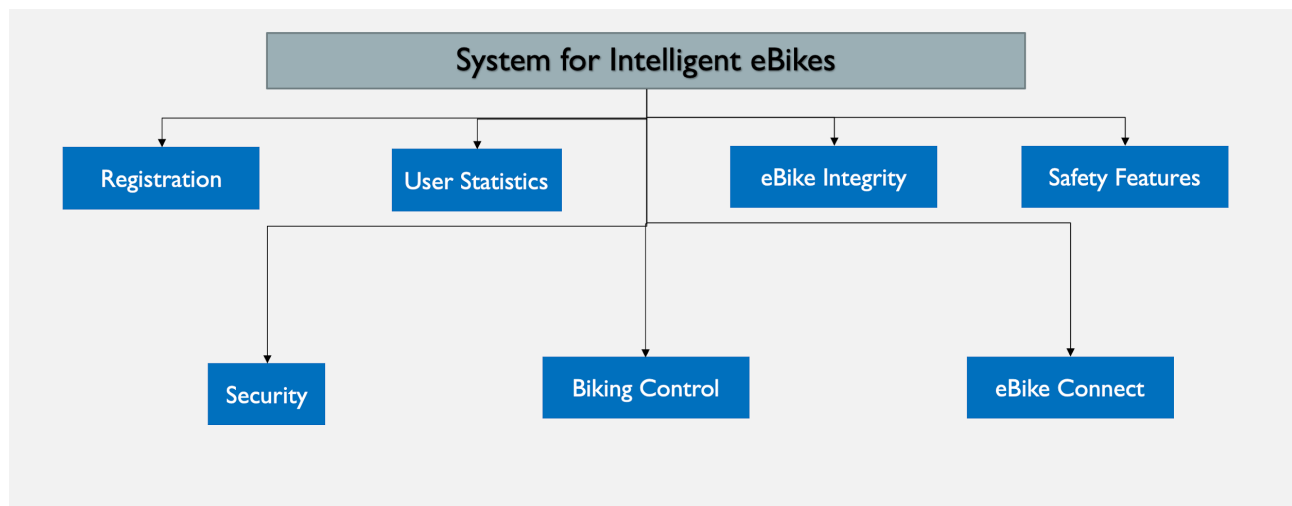
1.5 References

Arlow, Jim, and Ila Neustadt. UML and the Unified Process Practical Object-Oriented Analysis and Design. TPB, 2005.

2. Overall Description

2.1 System Perspective

The SIE System is a new, self-contained system that is focused on the electric bicycle technology. Many pieces of the subsystem requirements have been used in other industries like automotive and social media. This is the first system to combine this mix of technologies in the bicycle industry. The SIE System consists of seven subsystems (Registration, User Statistics, eBike Integrity, Safety Features, Security, Biking Control, eBike Connect) shown in the diagram below.



2.2 System Functions

The major functions the system must perform or must let the User perform are as follows:

- The system must allow the user to register and manage their eBike account.
- The system must track, aggregate, and display user data/statistics.
- The system must keep the user updated on the integrity of the eBike.
- The system must assist in keeping the user safe.
- The system must assist in theft prevention.
- The system must allow the user to connect in a Social Media manner to other eBike users.

2.3 User Classes and Characteristics

2.3.1 *Boundary Class: WebsiteMainGUI, MobileAppMainGUI, BikeMainBUI*

These classes allows users to interact with boundary classes. They provide a main interface for our system and allow users to register,check their ride history, and access the Connect feature.

2.3.2 *Control Class: SystemManager*

This class will allow us to provide interaction between the GUIs and users. It will also allow us to add, delete, and manage other entities.

2.3.3 *Entity Class: User*

This class is how we will interact with users. This will allow us to add, delete, and modify users.

2.4 Operating Environment

The SIE System will be web based and accessible by any computer or laptop and smartphone compatible on both ios and Droid operating systems. The system must also integrate with the eBike.

2.5 Design and Implementation Constraints

- The User must allow GPS Location tracking.
- The User must keep the mobile app up to date.
- The software is currently only available in English.
- The company may not share any user's personal account information.
- The software will store only 5 years worth of user statistics.
- The user's information will be stored on a secure server and can only be accessed by a systems administrator.

2.6 Assumptions and Dependencies

- The User will agree to the company's terms and conditions.
- The User will allow GPS Location tracking.
- The User will keep the mobile app up to date.

3. External Interface Requirements

3.1 User Interfaces

3.1.1 WebsiteMainGUI



3.1.2 MobileAppMainGUI



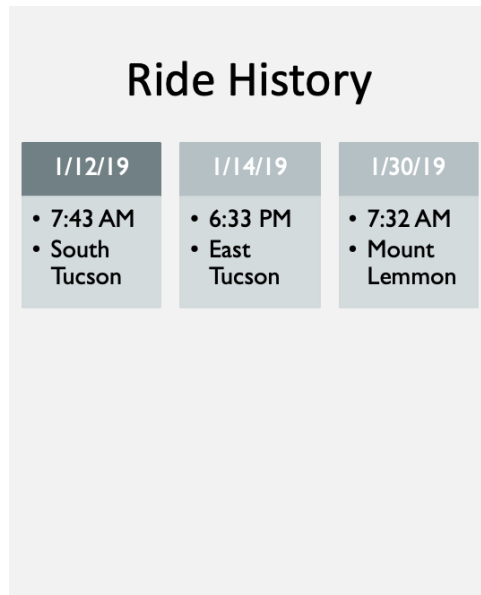
3.1.3 BikeMainGUI

MPG	00.00
RPM	00.00
Watts	00.00
Calories	00.00
Elevation	00.00
Duration	00.00
Assistance Level	None
	Partial
	Full

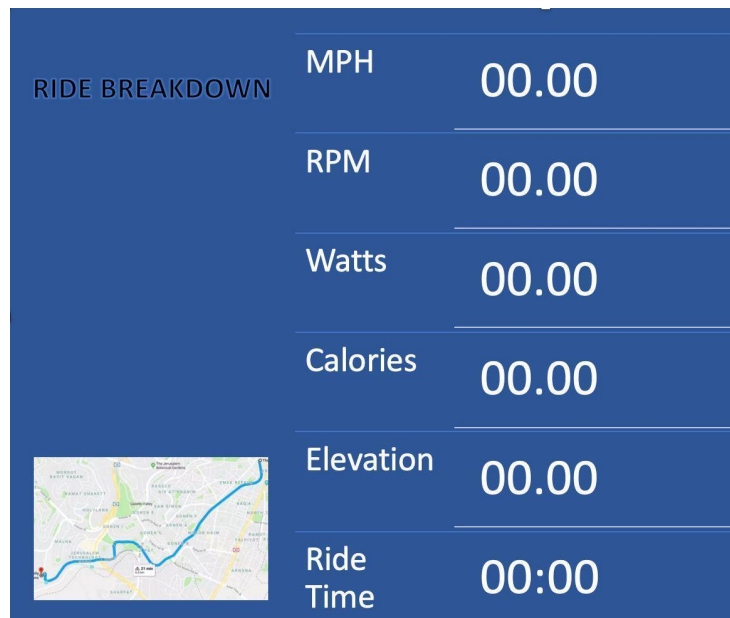
3.1.4 WebsiteRideHistoryGUI

Ride History		
1/12/19	1/14/19	1/30/19
<ul style="list-style-type: none">• 7:43 AM• South Tucson	<ul style="list-style-type: none">• 6:33 PM• East Tucson	<ul style="list-style-type: none">• 7:32 AM• Mount Lemmon

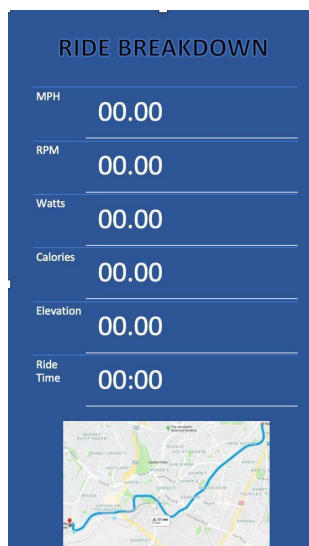
3.1.5 MobileAppRideHistoryGUI



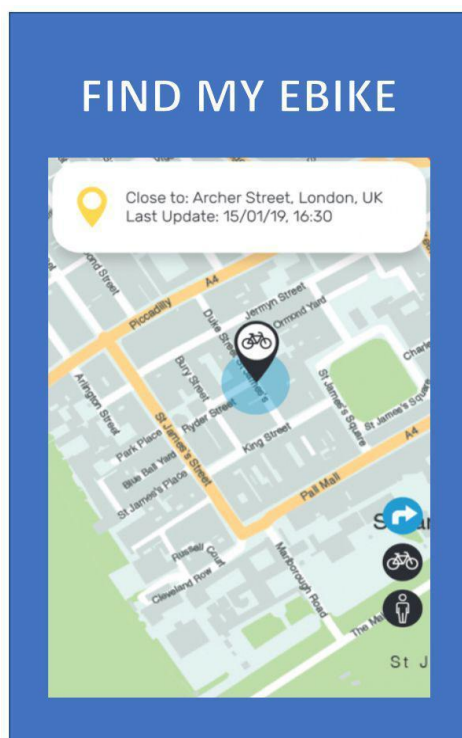
3.1.6 WebsiteStatisticsGUI



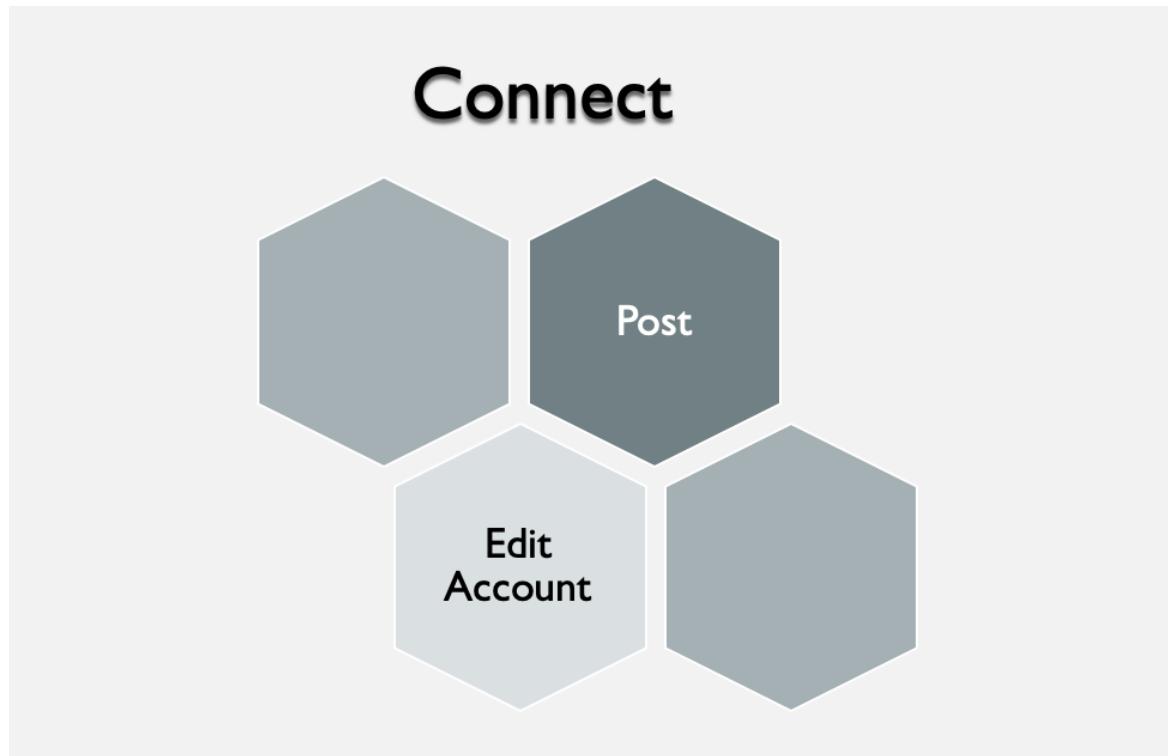
3.1.7 MobileAppStatisticsGUI



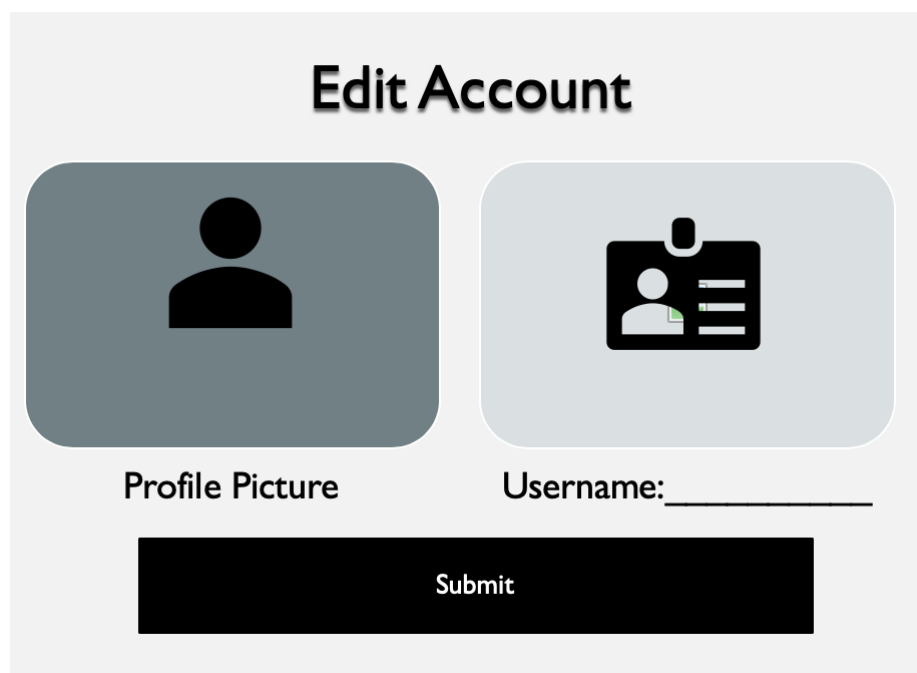
3.1.8 FindMyBikeGUI



3.1.9 WebsiteConnectGUI

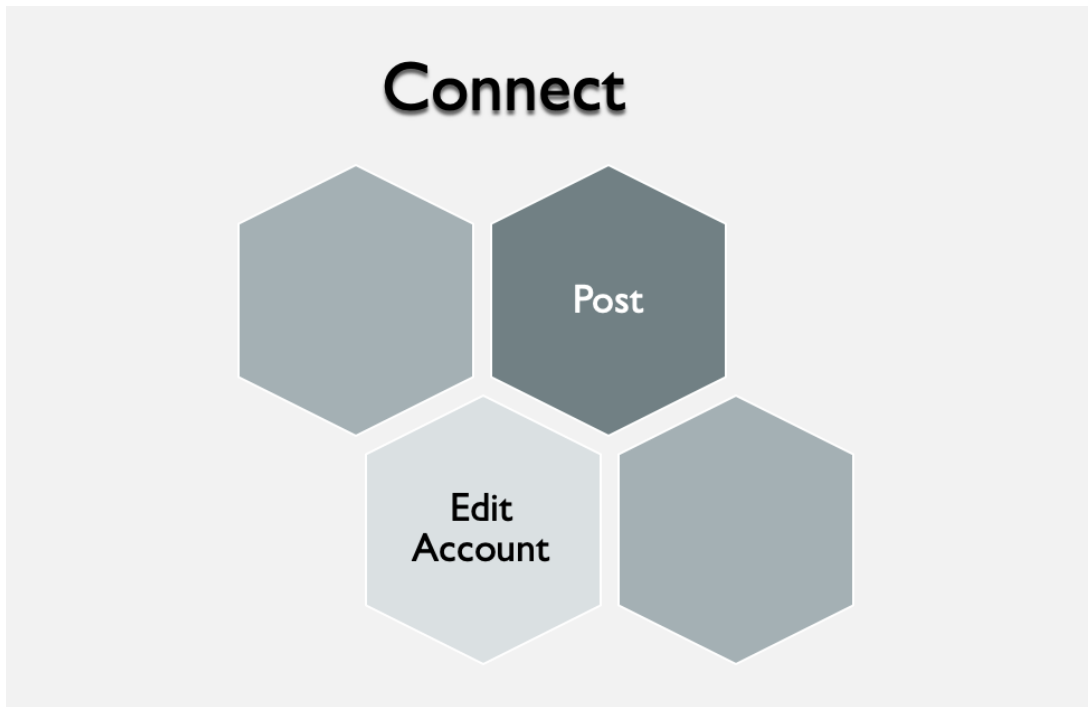


3.1.10 WebsiteEditConnectGUI

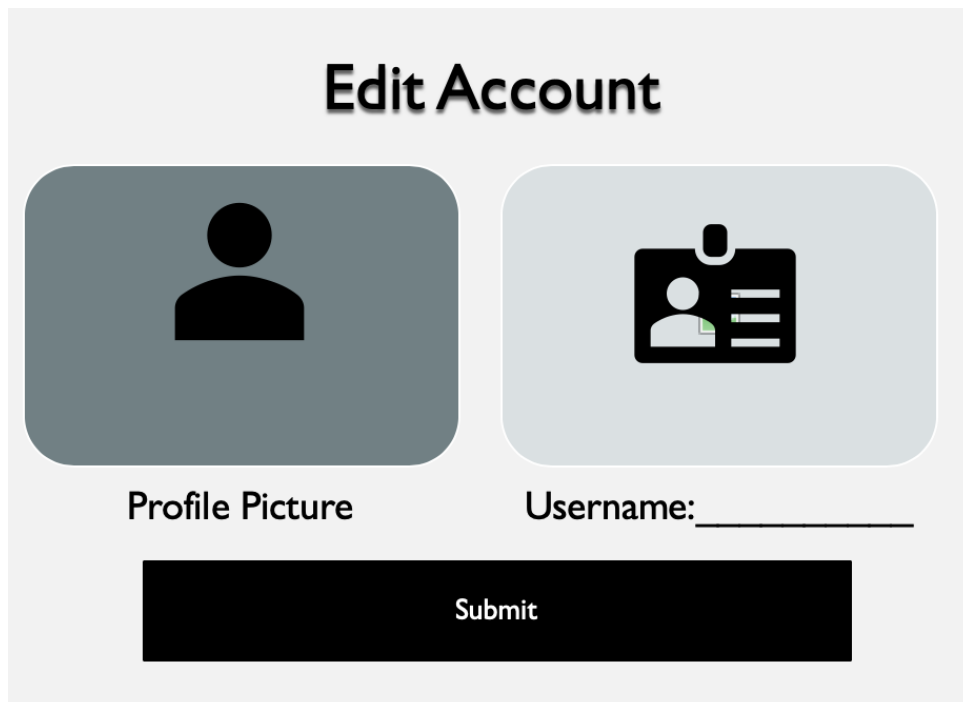


The image shows a GUI titled "Edit Account" in a large, bold, black font at the top center. Below the title, there are two rounded rectangular buttons. The left button is dark gray and contains a black silhouette of a person's head and shoulders. Below this button is the text "Profile Picture". The right button is light gray and contains a black icon of a person's head and shoulders next to a document with lines, representing a username or ID card. Below this button is the text "Username:" followed by a horizontal line for input. At the bottom of the interface is a large black rectangular button with the word "Submit" in white text. The entire interface is set against a light gray background.

3.1.11 MobileAppConnectGUI



3.1.12 MobileAppEditConnectGUI



3.1.13 MobileAppPostGUI

The image displays a mobile application interface for submitting a post. It features a dark gray header with the text "Submit Post" in white. Below the header is a form with four input fields, each with a white label: "Picture", "Text", "eBike Stats", and "Challenges". The form is set against a light gray background.

Submit Post			
Picture	Text	eBike Stats	Challenges

3.2 Hardware Interfaces

- Mobile Devices
- Bike Screen
- Computers
- GPS
- Object Detection Sensor
- TirePressureSensor
- MotorSensor
- BatterySensor
- Light Sensor
- Lock Sensor

3.3 Software Interfaces

- The web based system will be created using Java.
- The Mobile App will be created using Java.
- The Bike Screen will be created using Java.
- Visitor data will be saved in a server and will use C++ for the data management.

3.4 Communications Interfaces

- eMail
 - Password Authentication
 - FTP
- Web Browsers
 - Password Authentication
 - FTP
- Mobile App
 - Password Authentication
 - FTP
- Mobile App Notifications
 - Synchronization mechanism
 - FTP
- Bike Screen Notifications
 - Synchronization mechanism
 - FTP
- eBike Connect Account
 - Password Authentication
 - FTP

4. Requirements

The Requirements section contains artifacts created during the requirements capture phase including Function and Nonfunctional requirements and descriptions and thoughts on those artifacts.

4.1 Functional Requirements

4.1.1 Functional Statements

4.1.1.1 Registration Subsystem

4.1.1.1.1 The system shall allow the User to register their eBike through the website.

4.1.1.1.2 The system shall allow the User to register their eBike through the mobile app.

4.1.1.1.3 The system shall allow the User to edit profile information on the website.

4.1.1.1.4 The system shall allow the User to edit profile information on the mobile app.

4.1.1.1.5 The system shall allow the User to delete their profile on the website.

4.1.1.1.6 The system shall allow the User to delete their profile on the mobile app.

4.1.1.2 User Statistics Subsystem

4.1.1.2.1 The system shall track usage statistics for the eBike.

4.1.1.2.2 The system shall aggregate all eBike usage information in the cloud.

4.1.1.2.3 The system shall display usage statistics on the eBike.

4.1.1.2.4 The system shall display usage statistics on the website.

4.1.1.2.5 The system shall display usage statistics on the mobile app.

4.1.1.3 eBike Integrity Subsystem

4.1.1.3.1 The system shall alert the User when the tire is below minimum psi on the eBike and mobile app.

4.1.1.3.2 The system shall notify the User when the motor needs service on the eBike screen and mobile app.

4.1.1.3.3 The system shall warn the User when the battery reaches 20% on the eBike interface and mobile app.

4.1.1.3.4 The system shall notify the User when the battery reaches full charge on the eBike User interface and mobile app.

4.1.1.4 Safety Features Subsystem

4.1.1.4.1 The system shall provide a warning for any close obstacles to the eBike.

4.1.1.4.2 The system shall automatically turn on/off eBike headlights in low lighting/ well lit conditions.

4.1.1.4.3 The system shall have an emergency auto braking feature.

4.1.1.5 Security Subsystem

4.1.1.5.1 The system shall notify the User when the eBike detects the lock being hampered with.

4.1.1.5.2 The system shall keep a real time location of the eBike.

4.1.1.6 Biking Control Subsystem

4.1.1.6.1 The system shall allow the User to toggle between no motor assistance, partial motor assistance, and complete motor assistance.

4.1.1.7 eBike Connect Subsystem

4.1.1.7.1 The system shall allow the User to create a Connect Account on the website.

4.1.1.7.2 The system shall allow the User to create a Connect Account on the mobile app.

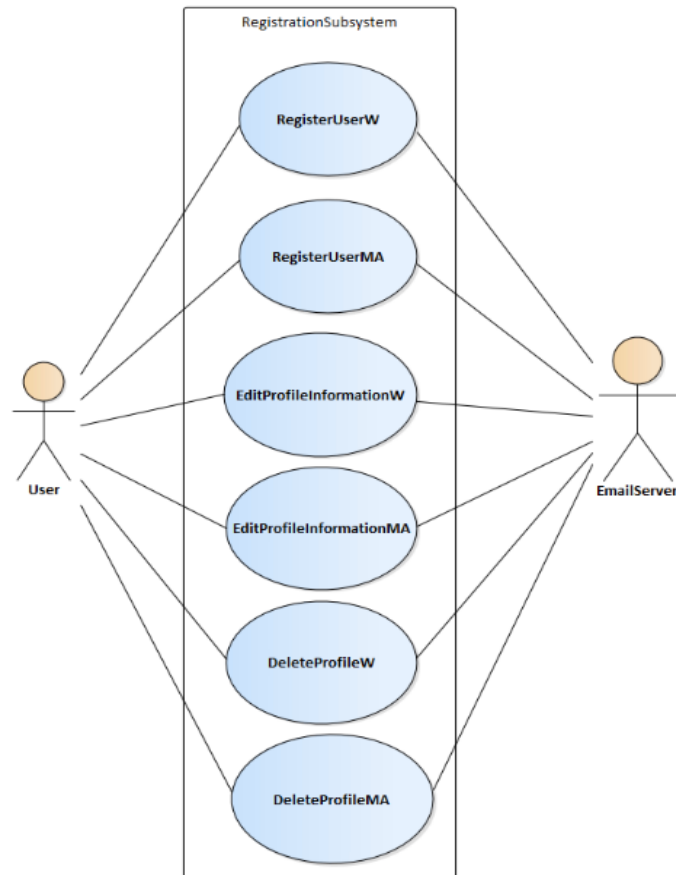
4.1.1.7.3 The system shall allow the User to edit their Connect Account on the website.

4.1.1.7.4 The system shall allow the User to edit their Connect Account on the mobile app.

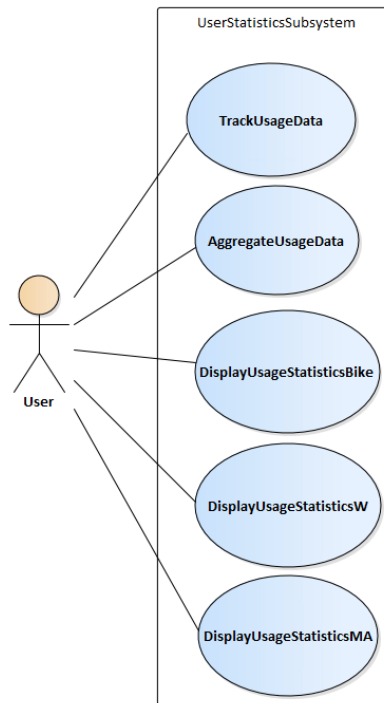
4.1.1.7.5 The system shall allow the User to create a post.

4.1.2 Use Case Diagrams

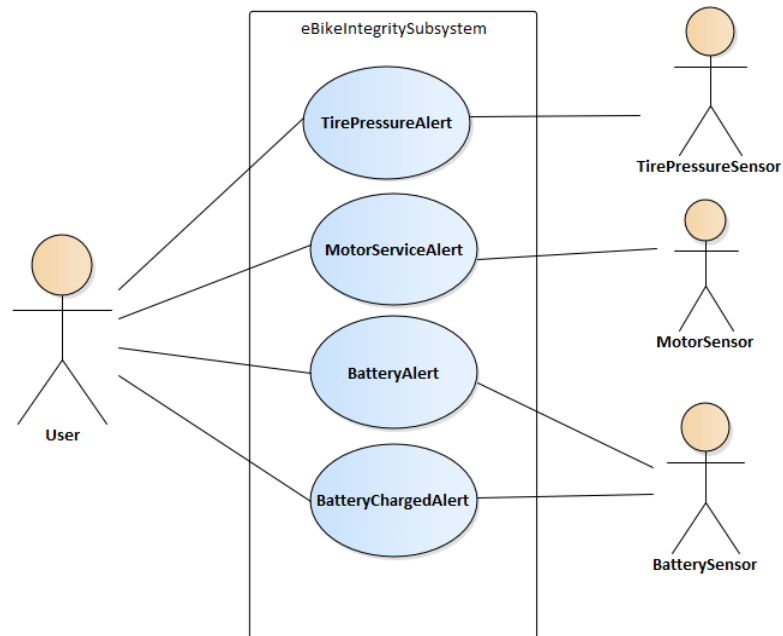
4.1.2.1 Registration Subsystem Use Case Diagram



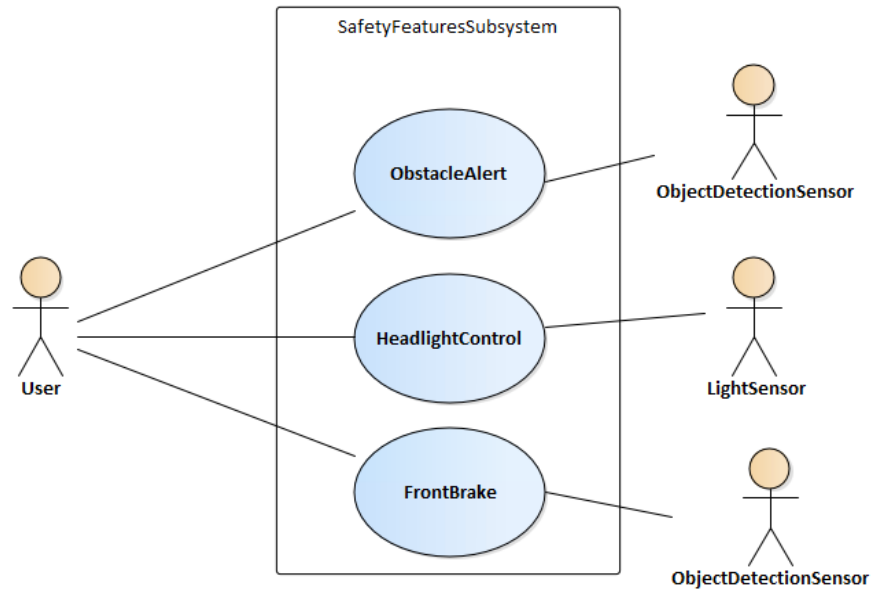
4.1.2.2 User Statistics Subsystem Use Case Diagram



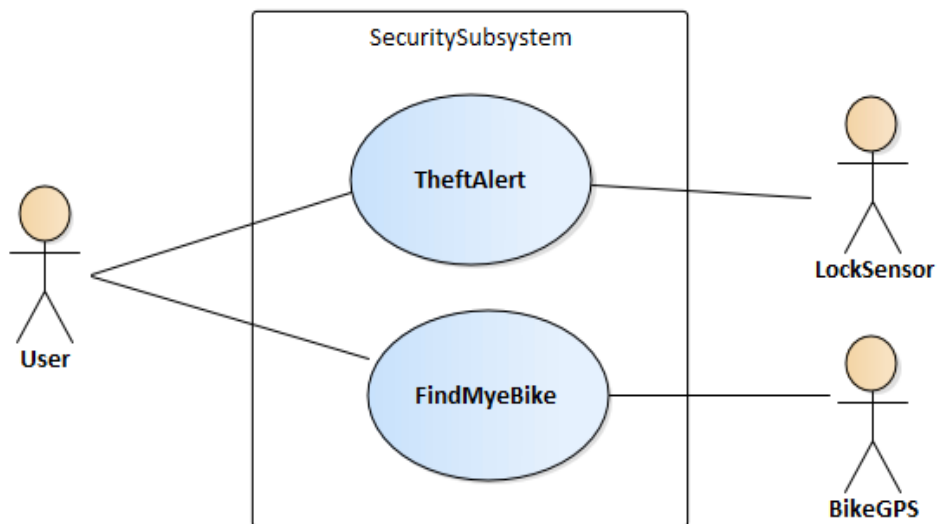
4.1.2.3 eBike Integrity Subsystem Use Case Diagram



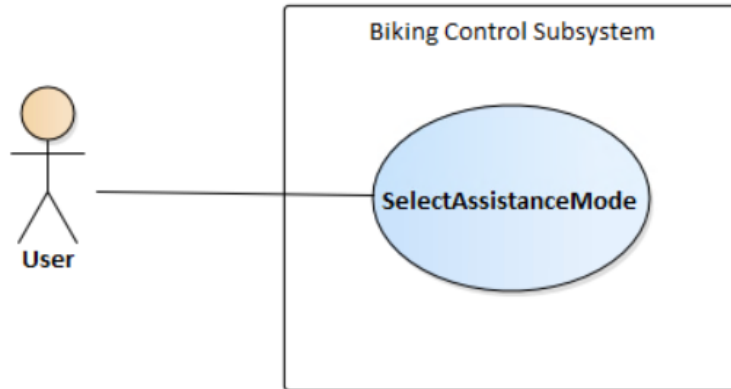
4.1.2.4 Safety Features Subsystem Use Case Diagram



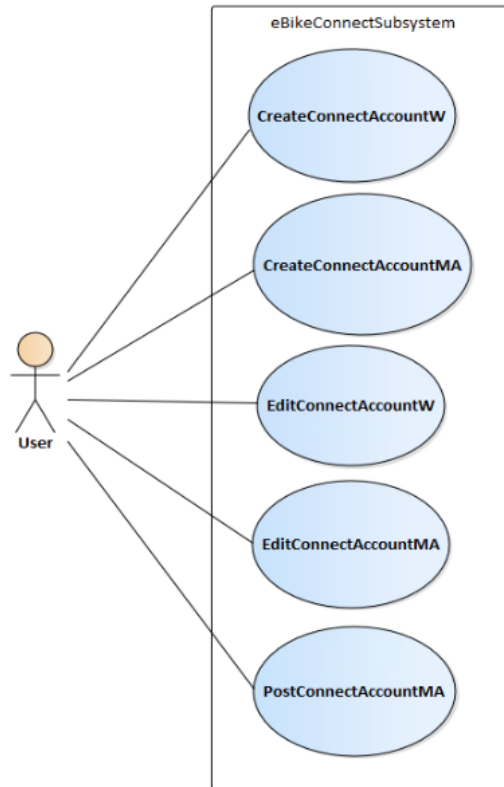
4.1.2.5 Security Subsystem Use Case Diagram



4.1.2.6 Biking Control Subsystem Use Case Diagram



4.1.2.7 eBike Connect Subsystem Use Case Diagram



4.1.3 Use Case Specifications

4.1.3.1 Registration Subsystem Use Case Specifications

Use Case: RegisterUserW
ID: 4.1.1.1.1
Brief Description: New Users register into the system on the website.
Primary Actor: User
Secondary Actors: EmailServer
Precondition: User owns eBike.
<p>Main Flow:</p> <ul style="list-style-type: none"> a. Use case begins when the User selects “Register” from the WebsiteMainGUI b. The system displays: Name, email, phone number, address, date of birth, eBike model, purchased from, purchase date, farm serial number, planned primary use, male/female/prefer not to say, password c. The User enters all the information requested and clicks “Submit” d. The system displays confirmation when registered. e. The system sends confirmation email. f. The User confirms email. g. The use case ends.
PostCondition: User has a profile.
Alternative Flow: None

Use Case: RegisterUserMA
ID: 4.1.1.1.2
Brief Description: New Users can register into the system on the mobile app.
Primary Actor: User
Secondary Actors: EmailServer
Precondition: User owns eBike and has Mobile App downloaded.
<p>Main Flow:</p> <ul style="list-style-type: none"> a. Use case begins when the User selects “Register” from the MobileAppMainGUI. b. The system displays: Name, email, phone number, address, date of birth, eBike model, purchased from, purchase date, fram serial number, planned primary use, male/female/prefer not

to say, password. c. The User enters all the information requested and clicks “Submit” d. The system displays confirmation when registered. e. The system sends confirmation email. f. The User confirms email. g. The use case ends.
PostCondition: User has a profile.
Alternative Flow: None

Use Case: EditProfileInformationW
ID: 4.1.1.1.3
Brief Description: The User edits their profile information on the website.
Primary Actor: User
Secondary Actors: EmailServer
Precondition: User is logged into their profile.
Main Flow: a. Use case begins when the User selects “Edit Information” from the WebsiteMainGUI b. The system displays: Name, email, phone number, address, date of birth, eBike model, purchased from, purchase date, frame serial number, planned primary use, male/female/prefer not to say, password. c. The User enters the information displayed and clicks “Submit” d. The system displays confirmation when finished. e. The system sends confirmation email. f. The use case ends.
PostCondition: The User’s profile is updated.
Alternative Flow: None

Use Case: EditProfileInformationMA
ID: 4.1.1.1.4
Brief Description: The User edits their profile information on the mobile app.
Primary Actor: User

Secondary Actors: EmailServer
Precondition: User is logged into their profile.
<p>Main Flow:</p> <ol style="list-style-type: none"> Use case begins when the User selects “Edit Information” from the MobileAppMainGUI. The system displays: Name, email, phone number, address, date of birth, eBike model, purchased from, purchase date, frame serial number, planned primary use, male/female/prefer not to say, password. The User enters the information displayed and clicks “Submit” The system displays confirmation when finished. The system sends confirmation email. The use case ends.
PostCondition: The User’s profile is updated.
Alternative Flow: None

Use Case: DeleteProfileW
ID: 4.1.1.1.5
Brief Description: The User deletes their profile on the website.
Primary Actor: User
Secondary Actors: EmailServer
Precondition: The User is logged into their profile
<p>Main Flow:</p> <ol style="list-style-type: none"> Use case begins when the User selects “Delete Account” from the WebsiteMainGUI. The system asks for confirmation. The User selects “Confirm” The system sends confirmation email. The use case ends.
PostCondition: The User no longer has a profile.
Alternative Flow: None

Use Case: DeleteProfileMA
ID: 4.1.1.1.6

Brief Description: The User deletes their profile on the mobile app.
Primary Actor: User
Secondary Actors: EmailServer
Precondition: The User is logged into their profile
Main Flow: a. Use case begins when the User selects “Delete Account” from the MobileAppMainGUI. b. The system asks for confirmation. c. The User selects “Confirm” d. The system sends confirmation email. e. The use case ends.
PostCondition: The User no longer has a profile.
Alternative Flow: None

4.1.3.2 User Statistics Subsystem Use Case Specifications

Use Case: TrackUsageData
ID: 4.1.1.2.1
Brief Description: The system tracks usage data.
Primary Actor: User
Secondary Actors: None
Precondition: The User has a profile
Main Flow: a. Use case begins when the User begins pedaling their eBike. b. The system begins to track ride route, distance, duration, rotations per minute, power output, elevation gained, elevation lost, calories burned, miles per hour. c. The system sends data to the cloud. d. The use case ends.
PostCondition: User's usage statistics is aggregated in the cloud.
Alternative Flow: None

Use Case: AggregateUsageData
ID: 4.1.1.2.2
Brief Description: The system aggregates usage statistics.
Primary Actor: User
Secondary Actors: None
Precondition: The User has rode the eBike.
Main Flow: a. Use case begins when the system sends statistics to the cloud. b. The system aggregates the data in the cloud. c. The use case ends.
PostCondition: User's usage statistics is aggregated in the cloud.
Alternative Flow: None

Use Case: DisplayUsageStatisticsBike
ID: 4.1.1.2.3
Brief Description: The system displays the usage statistics on the eBike.
Primary Actor: User
Secondary Actors: None
Precondition: The system has aggregated the User's Data
Main Flow: a. Use case begins when the User begins pedaling their eBike. b. The system displays the Users ride route, distance, duration, rotations per minute, power output, elevation gained, elevation lost, calories burned, miles per hour on the BikeMainGUI . c. The use case ends.
PostCondition: None
Alternative Flow: None

Use Case: DisplayUsageStatisticsW
ID: 4.1.1.2.4
Brief Description: The system displays the usage statistics on the website.
Primary Actor: User
Secondary Actors: None
Precondition: The system has aggregated the User's Data
Main Flow: a. Use case begins when the User signs in the website. c. The User selects "Ride History" from WebsiteMainGUI. d. The User selects a ride from the WebsiteRideHistoryGUI. e. The system displays the Users ride route, distance, duration, rotations per minute, power output, elevation, calories burned, miles per hour WebsiteStatisticsGUI. f. The use case ends
PostCondition: None
Alternative Flow: None

Use Case: DisplayUsageStatisticsMA
ID: 4.1.1.2.5
Brief Description: The system displays the usage statistics on the Mobile App.
Primary Actor: User
Secondary Actors: None
Precondition: The system has aggregated the User's Data.
Main Flow: a. Use case begins when the User completes a ride on their eBike. b. The User opens the mobile app. c. The User selects "Ride History" from MobileAppMainGUI. d. The User selects a ride from the MobileAppRideHistoryGUI. e. The system displays the Users ride route, distance, duration, rotations per minute, power output, elevation gained, elevation lost, calories burned, miles per hour MobileAppStatisticsGUI. f. The use case ends.
PostCondition: None
Alternative Flow: None

4.1.3.3 eBike Integrity Subsystem Use Case Specifications

Use Case: TirePressureAlert
ID: 4.1.1.3.1
Brief Description: The system will detect and alert the User of low tire pressure.
Primary Actor: User
Secondary Actors: TirePressureSensor
Precondition: None.
Main Flow: a. Use case begins when the eBike's tire reaches a below minimum PSI. b. The Tire Pressure Sensor will detect the below minimum PSI. c. The Tire Pressure Sensor will send a notification to the User's profile on the MobileAppMainGUI and BikeMainGUI.

d. User will dismiss notification. e. The use case ends.
PostCondition: None.
Alternative Flow: None.

Use Case: MotorServiceAlert
ID: 4.1.1.3.2
Brief Description: The system will detect and alert the User when the motor needs service.
Primary Actor: User
Secondary Actors: MotorSensor
Precondition: None.
Main Flow: a. Use case begins when the motor is not working correctly. b. The Motor Sensor will detect the error. c. The Motor Sensor will send a notification to the User's profile on the MobileAppMainGUI and BikeMainGUI. d. User will dismiss notification. e. The use case ends.
PostCondition: None.
Alternative Flow: None.

Use Case: BatteryAlert
ID: 4.1.1.3.3
Brief Description: The system will alert the User when the eBike battery is low.
Primary Actor: User
Secondary Actors: BatterySensor
Precondition: None.

Main Flow:

- a. Use case begins when the battery reaches 20% .
- b. The Battery Sensor will detect the low battery.
- c. The Battery Sensor will send a notification to the User's profile on the MobileAppMainGUI and BikeMainGUI.
- d. User will dismiss notification.
- e. The use case ends.

PostCondition: None.

Alternative Flow: None.

Use Case: BatteryChargedAlert
ID: 4.1.1.3.4
Brief Description: The system will alert the User when the eBike battery is charged.
Primary Actor: User
Secondary Actors: BatterySensor
Precondition: None.
Main Flow: <ul style="list-style-type: none"> a. Use case begins when the battery reaches 100% charge . b. The Battery Sensor will detect the charged battery. c. The Battery Sensor will send a notification to the User's profile on the MobileAppMainGUI and BikeMainGUI. d. User will dismiss notification. e. The use case ends.
PostCondition: None.
Alternative Flow: None.

4.1.3.4 Safety Features Subsystem Use Case Specifications

Use Case: ObstacleAlert
ID: 4.1.1.4.1
Brief Description: The system will detect and alert the User of obstacles.
Primary Actor: User
Secondary Actors: ObjectDetectionSensor
Precondition: None
Main Flow: <ul style="list-style-type: none"> a. Use case begins when the User comes within 4 feet of an external obstacle. b. The Object Detection Sensor will detect the obstacle. c. The User be alerted on the eBike of the obstacle. d. The use case ends.
PostCondition: None

Alternative Flow: None

Use Case: HeadlightControl
ID: 4.1.1.4.2
Brief Description: The system will detect low lighting and turn on the eBike's headlight or well lit lighting and turn the eBike's headlight off.
Primary Actor: User
Secondary Actors: LightSensor
Precondition: None
Main Flow: a. Use case begins when the User is riding. b. The Light Sensor will detect the light condition. c. The eBike headlights will turn on in low light conditions or off in well lit conditions. d. The use case ends.
PostCondition: None
Alternative Flow: None

Use Case: FrontBrake
ID: 4.1.1.4.3
Brief Description: The system will have a front brake that slows down before completing a full stop when an object is too close to the front.
Primary Actor: User
Secondary Actors: ObjectDetectionSensor
Precondition: An object is getting too close to the eBike from front of eBike.
Main Flow: a. Use case begins when the User is riding. b. The ObjectDetectionSensor detects obstacle and signals the eBike brake. c. The eBike comes to a complete stop. d. The use case ends.
PostCondition: The eBike stops.
Alternative Flow: None

4.1.3.5 Security Subsystem Use Case Specifications

Use Case: TheftAlert
ID: 4.1.1.5.1
Brief Description: The system will detect lock misuse and alert the User. If the eBike is being stolen the User can lock the wheels into place, preventing it from moving any further.
Primary Actor: LockSensor
Secondary Actors: User
Precondition: None
Main Flow: <ul style="list-style-type: none"> a. Use case begins when the lock of the eBike is being misused. b. The eBike wheels will automatically lock. c. The User receives an alert on the MobileAppMainGUI. d. The User will select “Keep Locked” or “Unlock” e. The use case ends.
PostCondition: None.
Alternative Flow: None

Use Case: FindMyeBike
ID: 4.1.1.5.2
Brief Description: The system will display the location of the eBike on the mobile
Primary Actor: User
Secondary Actors: BikeGPS
Precondition: User is on the MobileAppMainGUI
Main Flow: <ul style="list-style-type: none"> a. Use case begins when the User selects the “FindMyeBike” on the MobileAppMainGUI. b. The BikeGPS will detect the location of the bike. c. The system will display the current location of the eBike through the FindMyBikeGUI d. The use case ends.
PostCondition: None.
Alternative Flow: None

4.1.3.6 Biking Control Subsystem Use Case Specifications

Use Case: SelectAssistanceMode
ID: 4.1.1.6.1
Brief Description: The system will allow the User to select how much motor assistance they want.
Primary Actor: User
Secondary Actors: None.
Precondition: The battery contains a charge.
Main Flow: a. Use case begins when the User selects “No Assistance”, “Partial Assistance”, “Full Assistance” on the BikeMainGUI. b. The motor adjusts the amount of assistance. c. The use case ends.
PostCondition: None.
Alternative Flow: None

4.1.3.7 eBike Connect Subsystem Use Case Specifications

Use Case: CreateConnectAccountW
ID: 4.1.1.7.1
Brief Description: User creates a Connect Account on the Website.
Primary Actor: User
Secondary Actors: None.
Precondition: User is registered and logged onto their account on the website.
Main Flow: a. Use case begins when the User selects “Connect” from the WebsiteMainGUI b. The system displays: UserName and Profile Picture on WebsiteEditConnectGUI. c. The User enters the information requested and clicks “Submit” d. The use case ends.

PostCondition: User has a Connect Account.
Alternative Flow: None

Use Case: CreateConnectAccountMA
ID: 4.1.1.7.2
Brief Description: User creates a Connect Account on the Mobile App..
Primary Actor: User
Secondary Actors: None.
Precondition: User is registered and logged onto their account on the mobile app.
Main Flow: a. Use case begins when the User selects “Connect” from the MobileAppMainGUI b. The system displays: UserName and Profile Picture on MobileAppEditConnectGUI. c. The User enters the information requested and clicks “Submit” d. The use case ends.
PostCondition: User has a Connect Account.
Alternative Flow: None

Use Case: EditConnectAccountW
ID: 4.1.1.7.3
Brief Description: The User edits their Connect Account on the website.
Primary Actor: User
Secondary Actors: None.
Precondition: User is registered and logged onto their account on the website.
Main Flow: a. Use case begins when the User selects “Edit Account” from the WebsiteConnectGUI. b. The system displays Username and Profile Picture c. The User enters the information displayed and clicks “Submit” on the WebsitEditConnectGUI. d. The use case ends.
PostCondition: The User’s Create Account is updated.
Alternative Flow: None

Use Case: EditConnectAccountMA
ID: 4.1.1.7.4
Brief Description: The User edits their Connect Account on the Mobile App.
Primary Actor: User
Secondary Actors: None.
Precondition: User is logged into their profile and selects “Connect.”
Main Flow: a. Use case begins when the User selects “Edit Account” from the MobileAppConnectGUI. b. The system displays Username and Profile Picture. c. The User enters the information displayed and clicks “Submit” d. The use case ends.
PostCondition: The User’s Create Account is updated.
Alternative Flow: None

Use Case: PostConnectAccountMA
ID: 4.1.1.7.5
Brief Description: The User post on their Connect Account from their mobile app.
Primary Actor: User
Secondary Actors: None.
Precondition: User is logged into their profile and selects “Connect.”
Main Flow: a. Use case begins when the User selects “Post” from the MobileAppConnectGUI. b. The system allows the user to post text, pictures, usage statistics, and challenges. c. The User enters the information displayed and clicks “Submit” d. The use case ends.
PostCondition: The User’s content is posted for other eBike users to see.
Alternative Flow: None

4.2 Nonfunctional Requirements

4.2.1 Performance Requirements

- 4.2.1.1 The system shall use GPS location for tracking.
- 4.2.1.2 The system shall update the Usage statistics in less than 10 seconds.
- 4.2.1.3 The web based system shall be coded in Java.
- 4.2.1.4 The Mobile App shall be coded in Java.
- 4.2.1.5 The Bike Screen shall be coded in Java.
- 4.2.1.6 User data shall be saved in an online server.
- 4.2.1.7 User data shall use C++ for the data management.

4.2.2 Safety Requirements

- 4.2.2.1 The system shall not send text notifications while the user is biking.
- 4.2.2.2 The system shall alert the user in a non-visual manner while the user is biking.

4.2.3 Security Requirements

- 4.2.3.1 The system shall have duo-authentication for administrative access to user data.
- 4.2.3.2 The system shall keep user location safeguarded

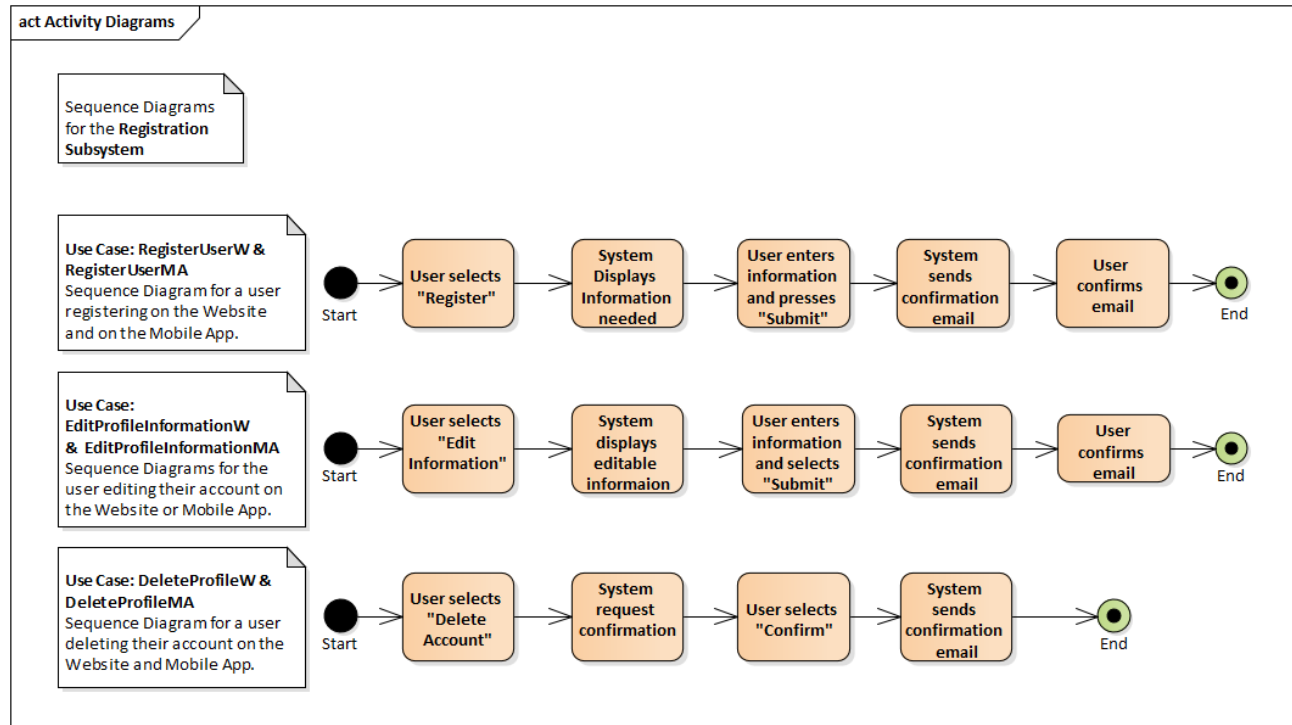
4.2.4 Software Quality Attributes

- 4.2.4.1 The web based system shall be coded in Java.
- 4.2.4.2 The Mobile App shall be coded in Java.
- 4.2.4.3 The Bike Screen shall be coded in Java.
- 4.2.4.4 User data shall be saved in an online server.
- 4.2.4.5 User data shall use C++ for the data management.

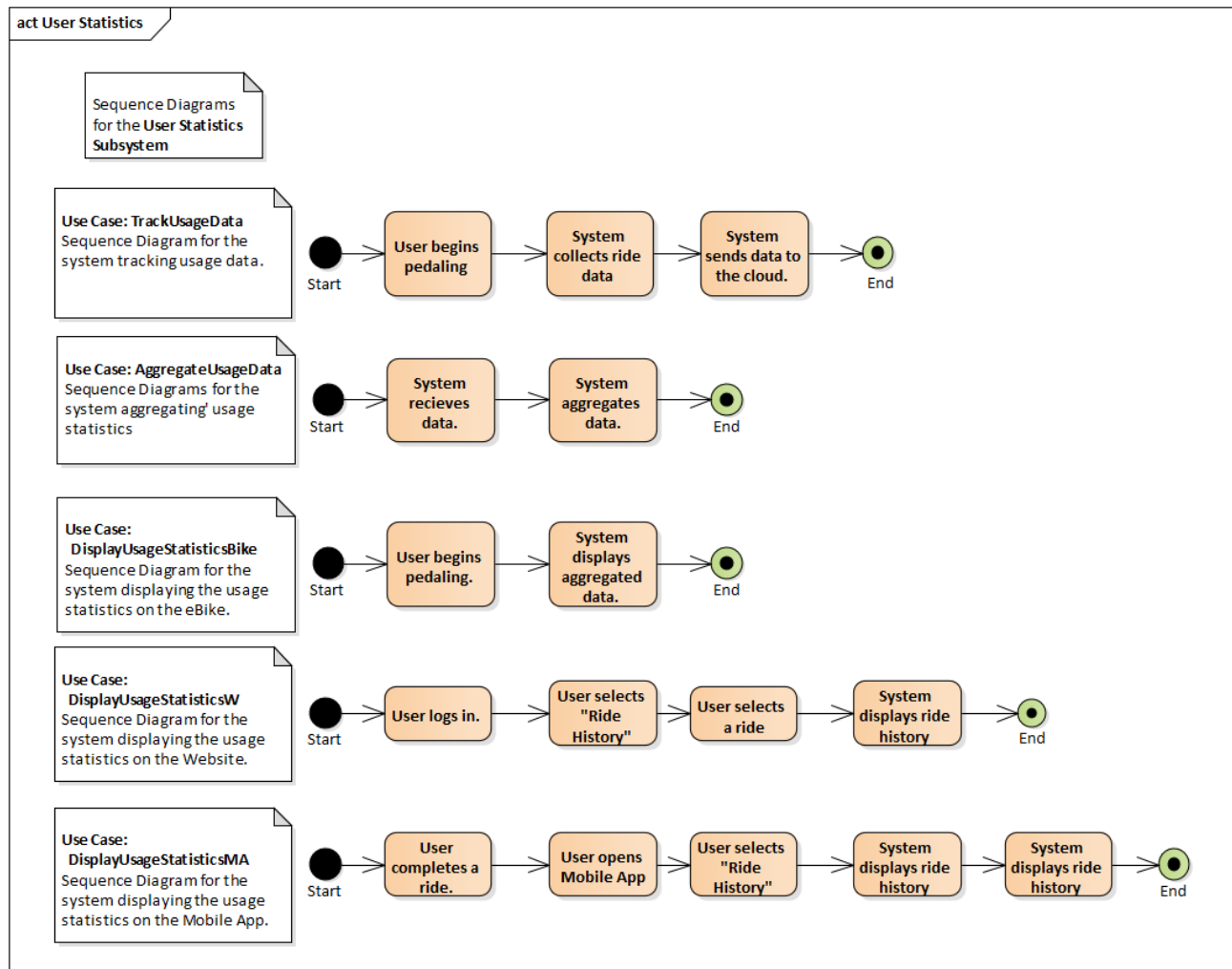
5. Analysis

5.1 Activity Diagrams

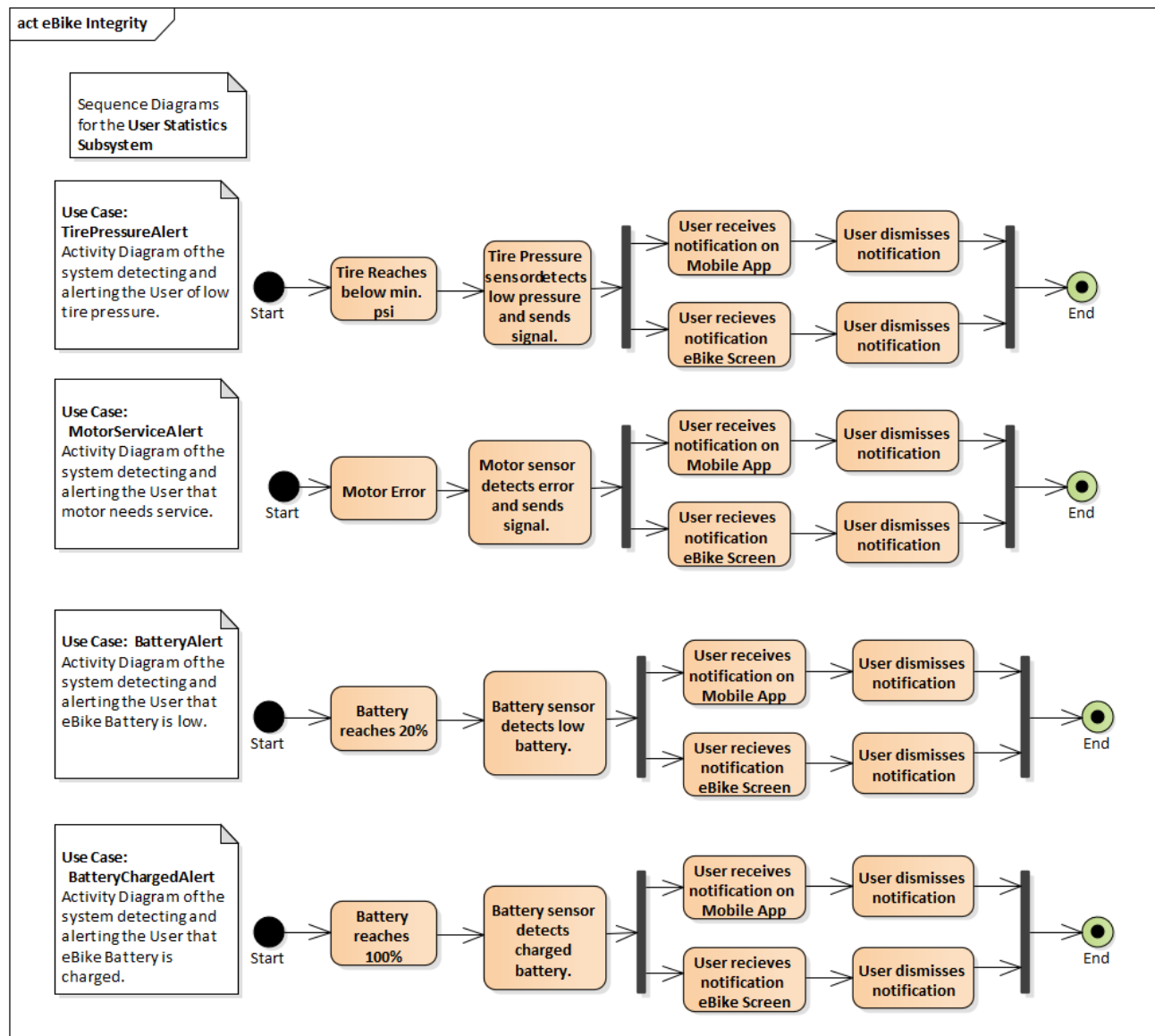
5.1.1 Registration Subsystem



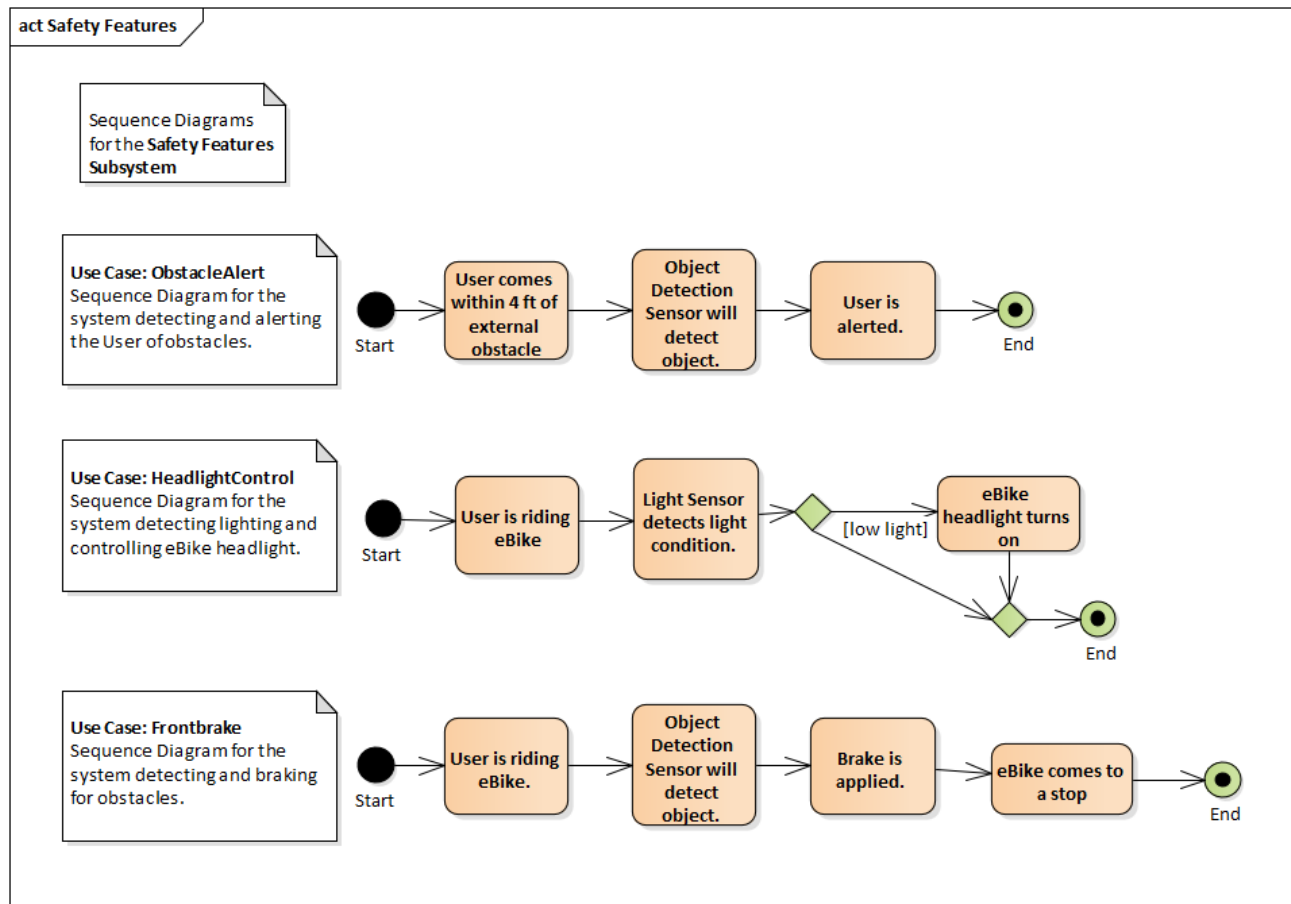
5.1.2 User Statistics Subsystem



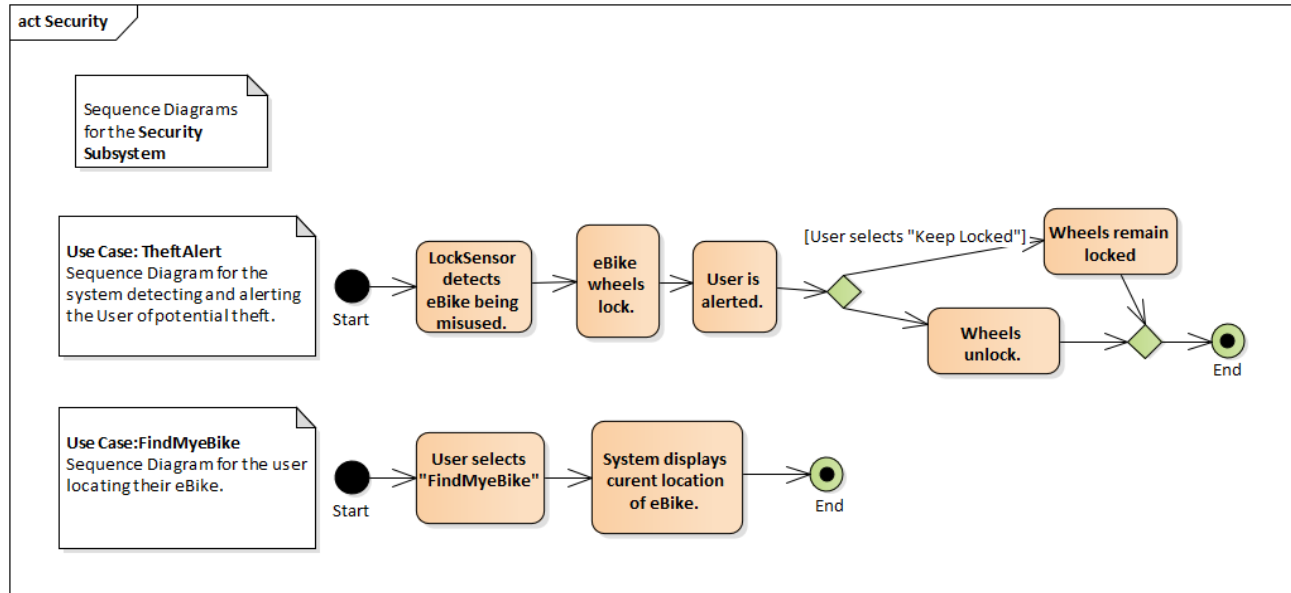
5.1.3 eBike Integrity Subsystem



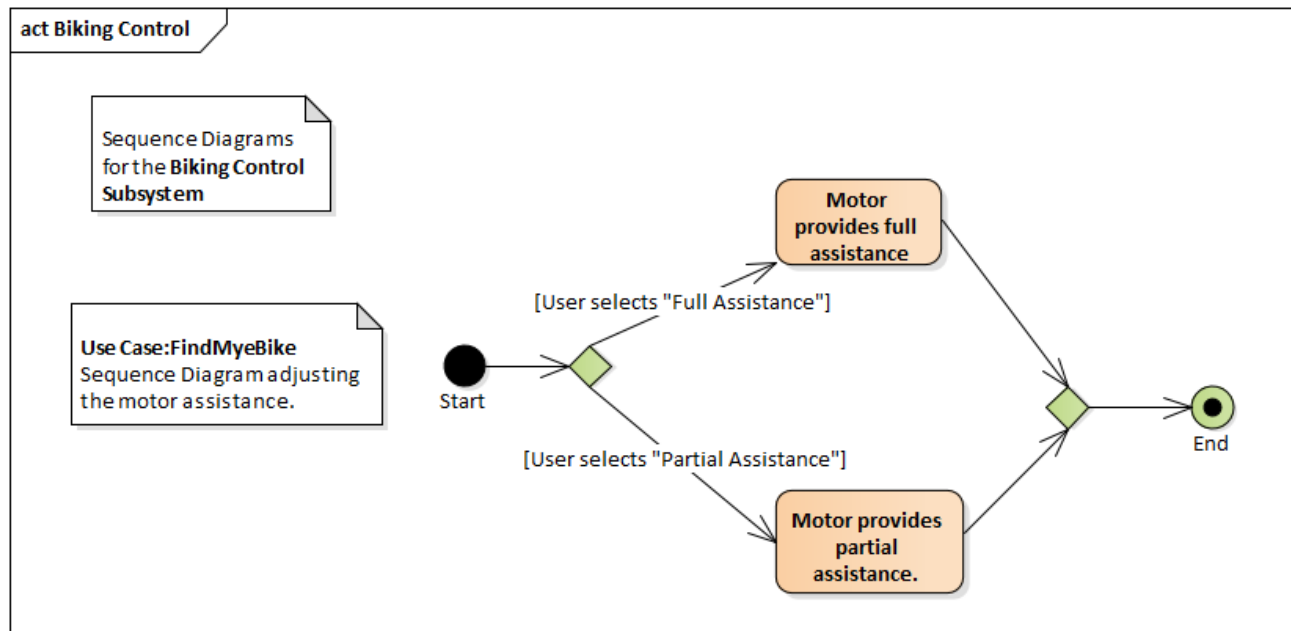
5.1.4 Safety Features Subsystem

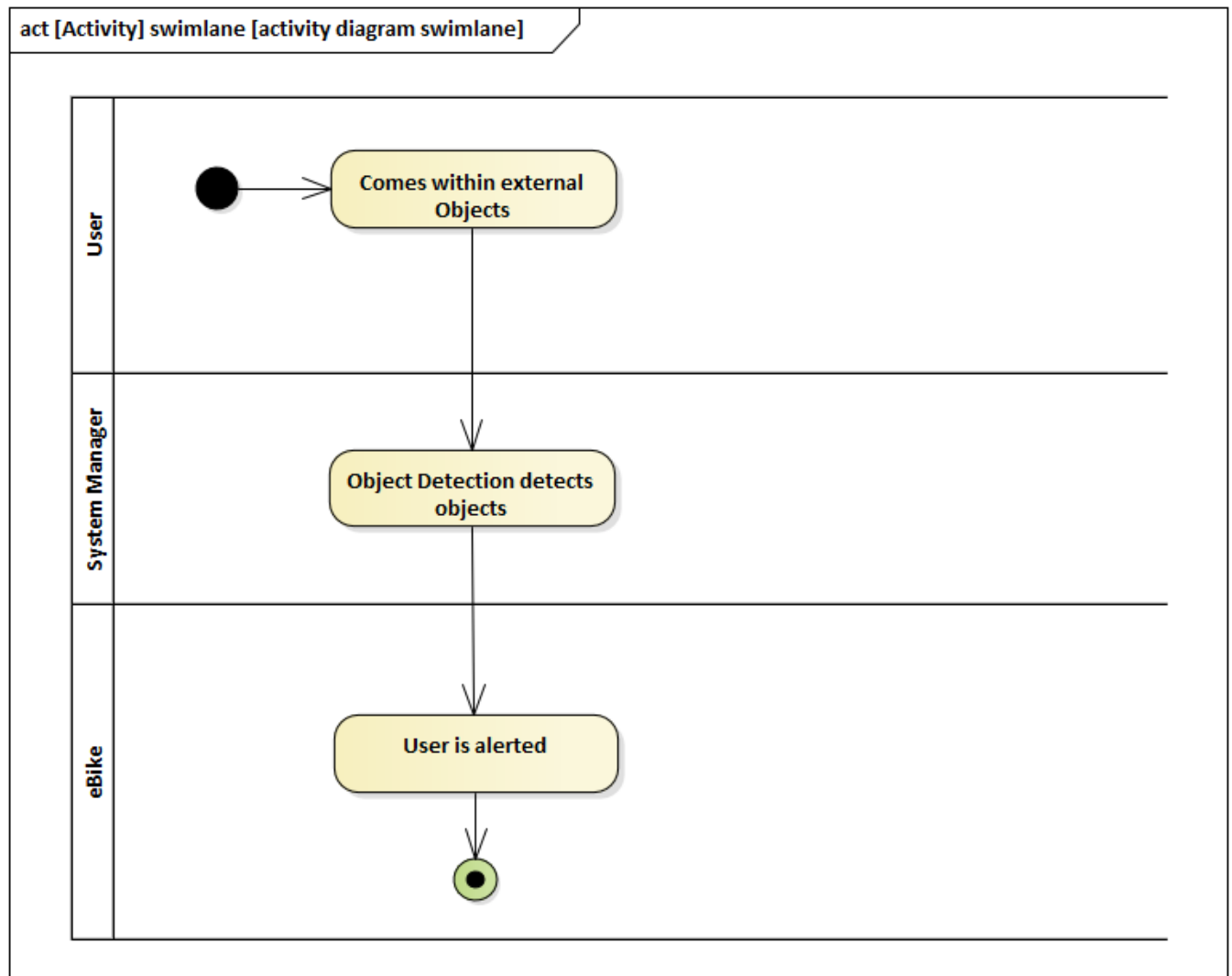


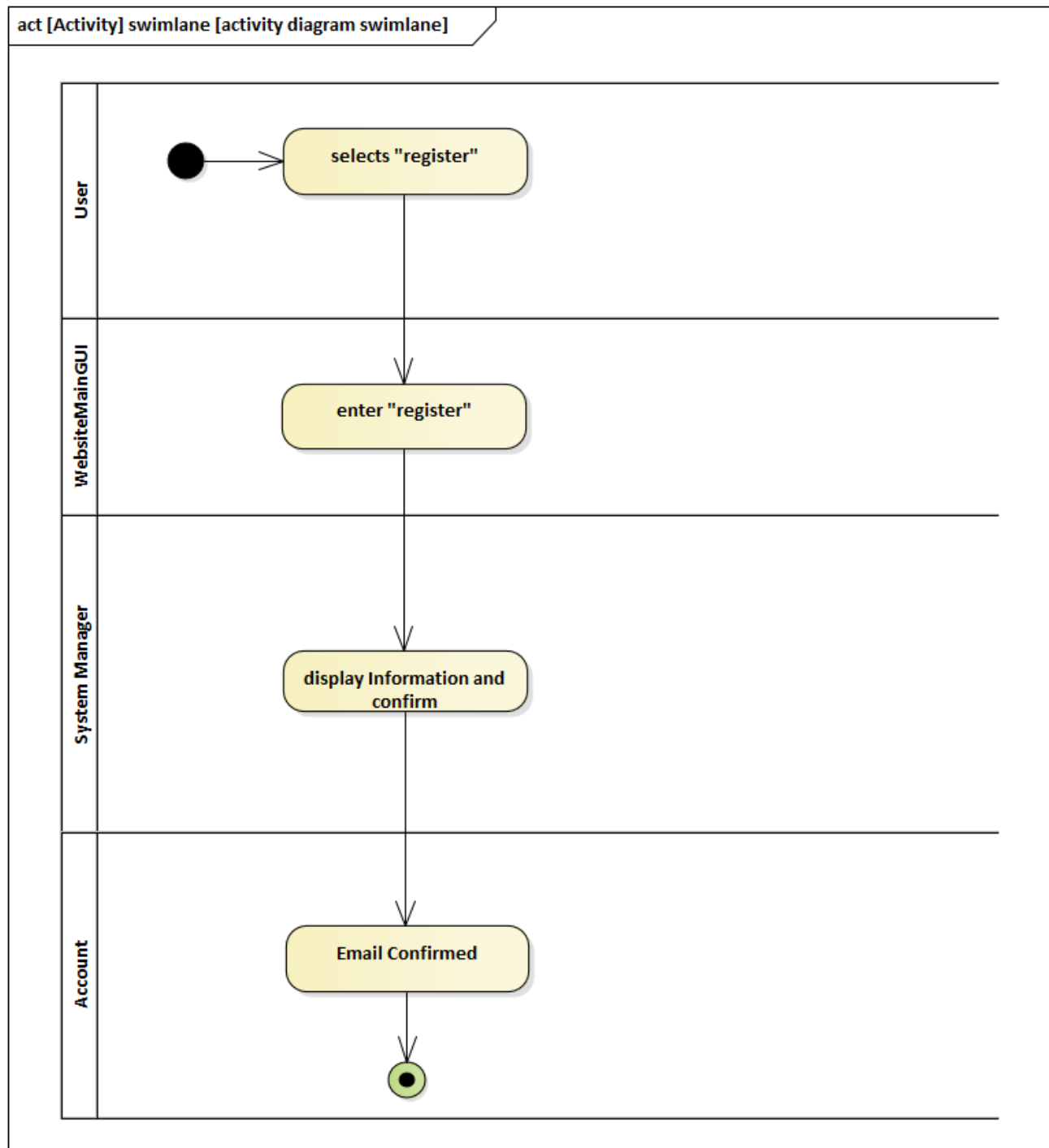
5.1.5 Security Subsystem



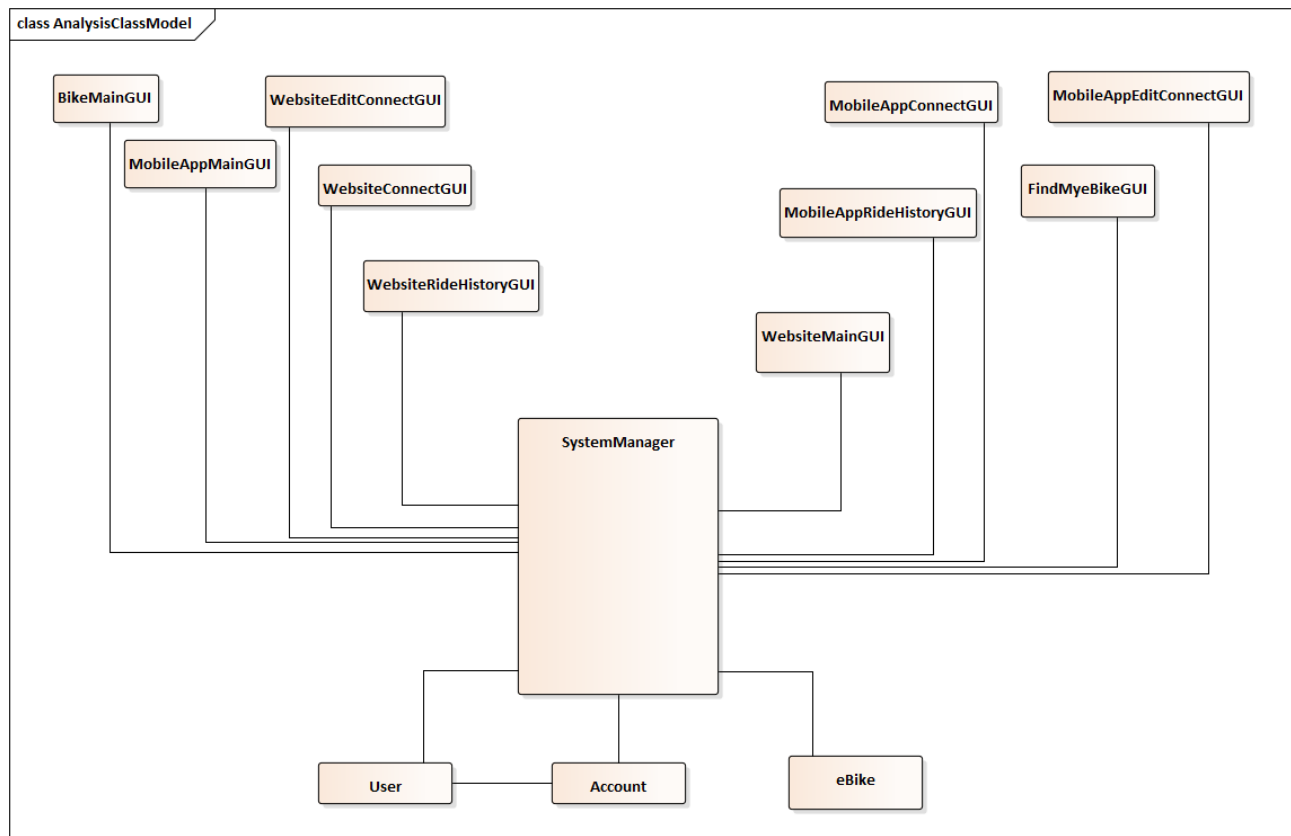
5.1.6 Biking Control Subsystem



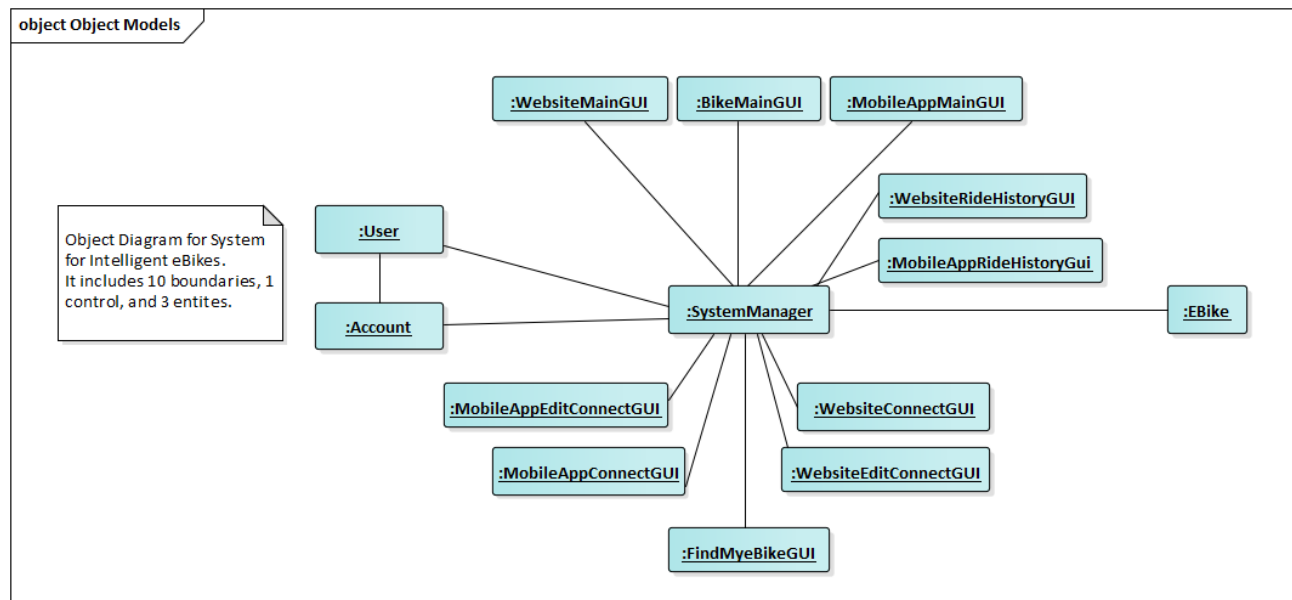




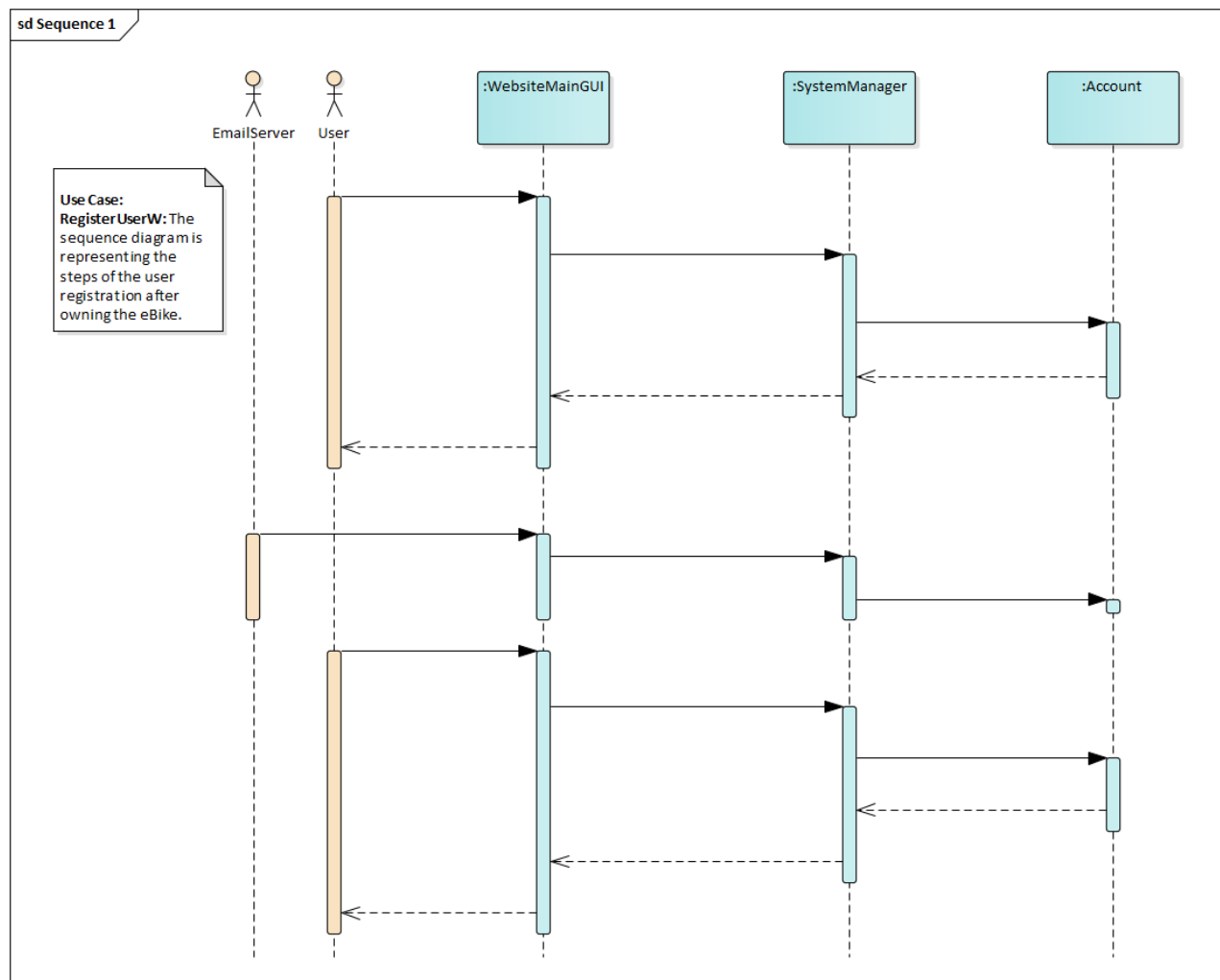
5.2 Class Model Analysis Level

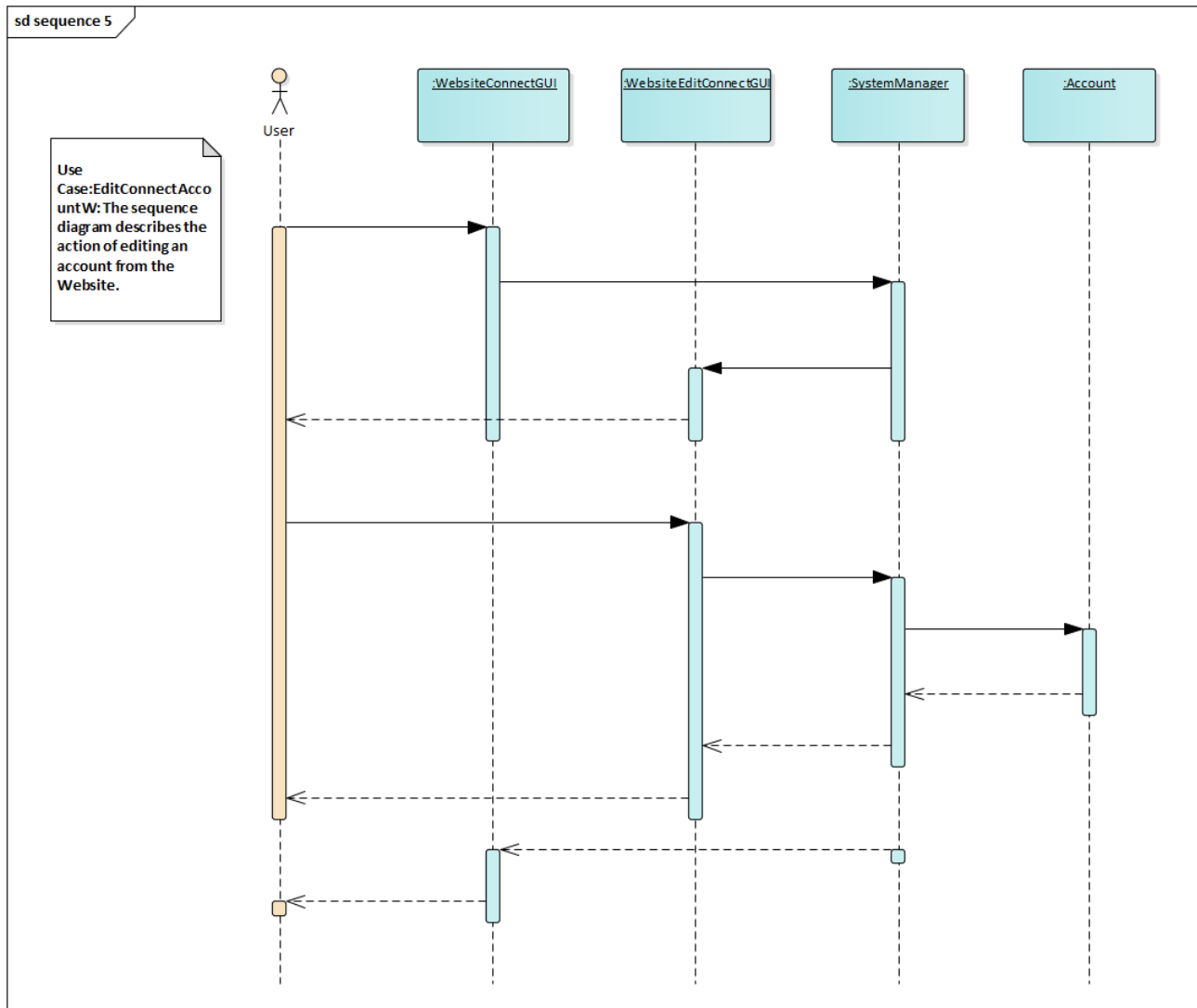


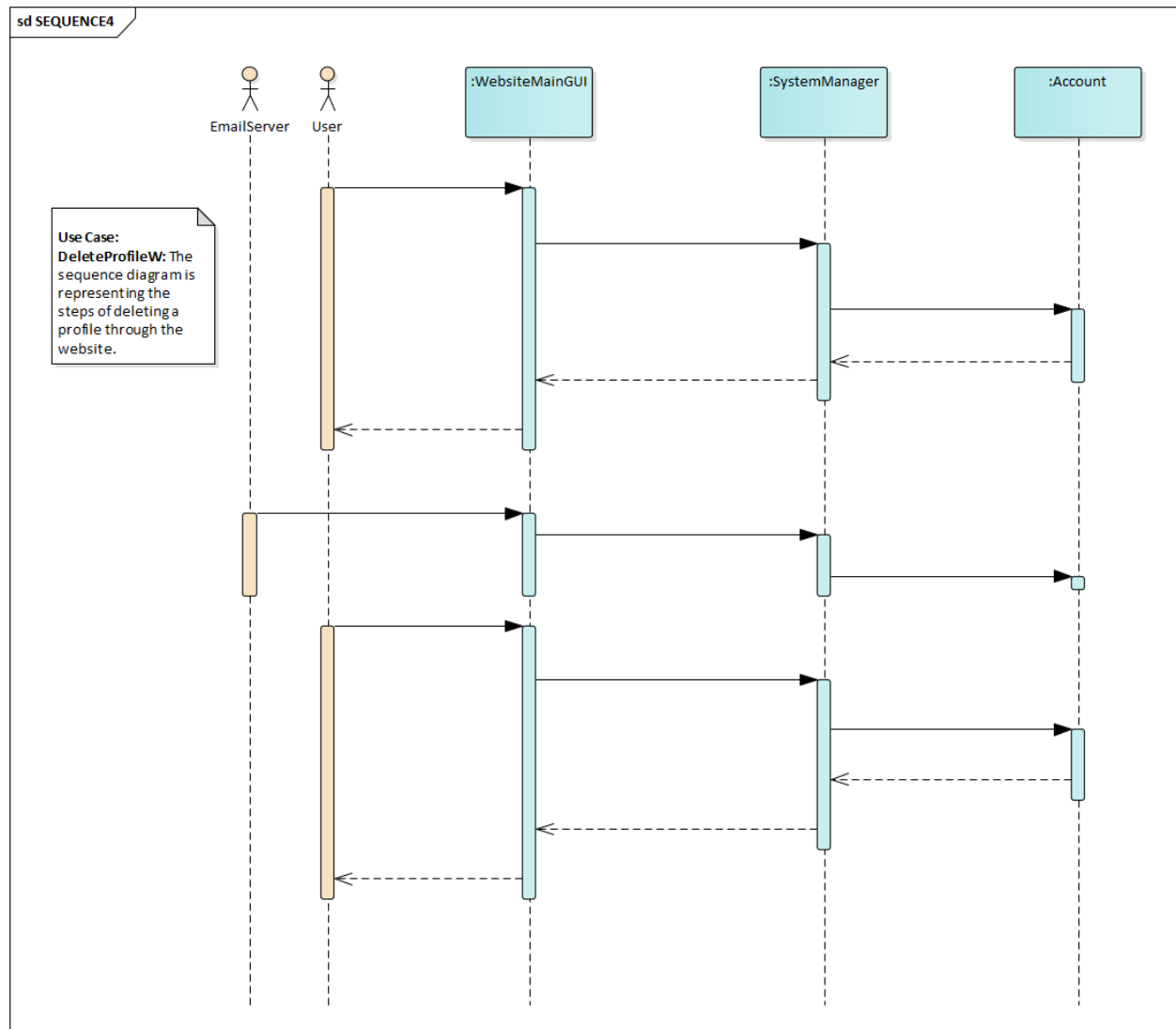
5.3 Object Model Analysis Level

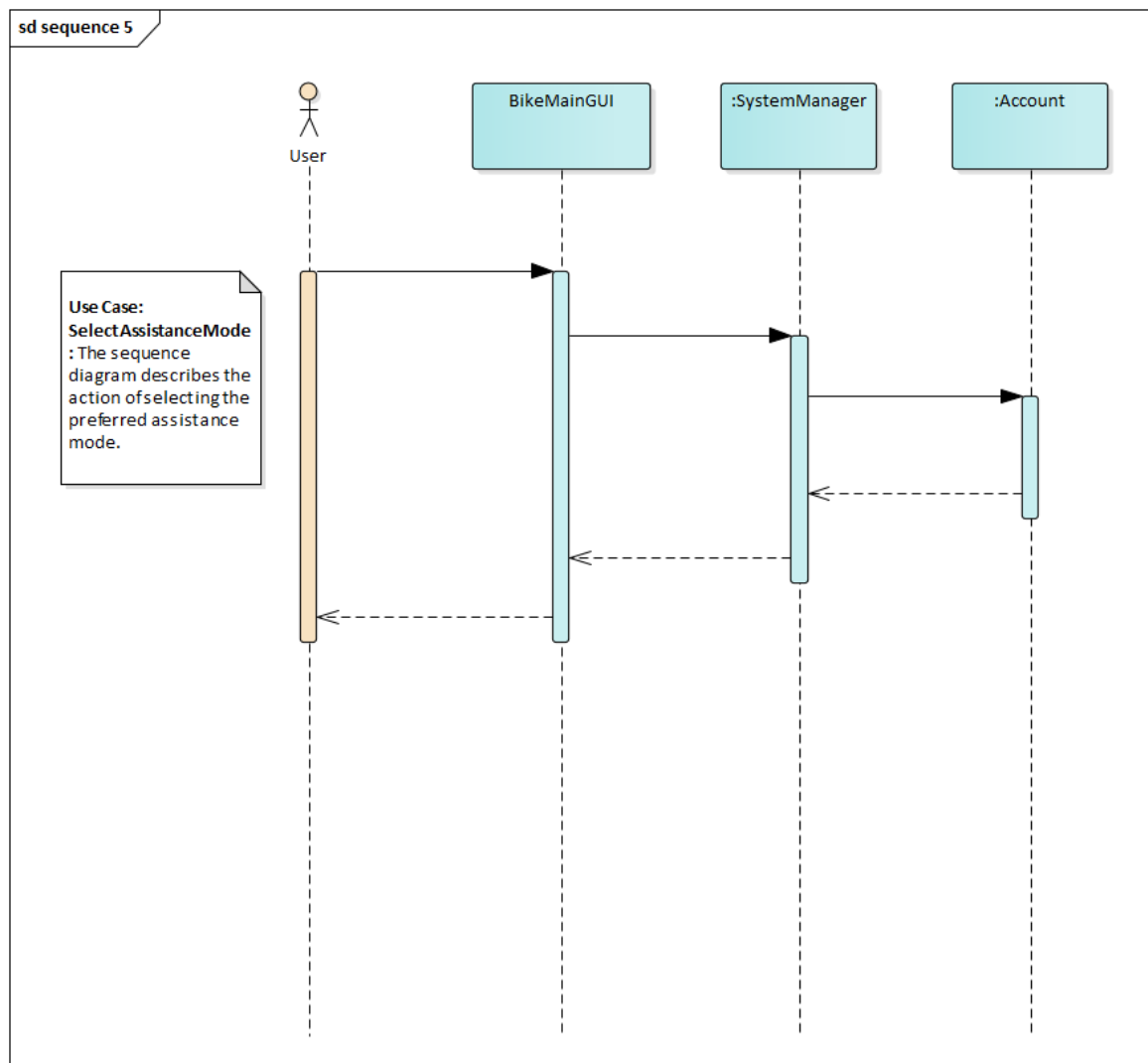


5.4 Use Case Realizations Analysis Level:





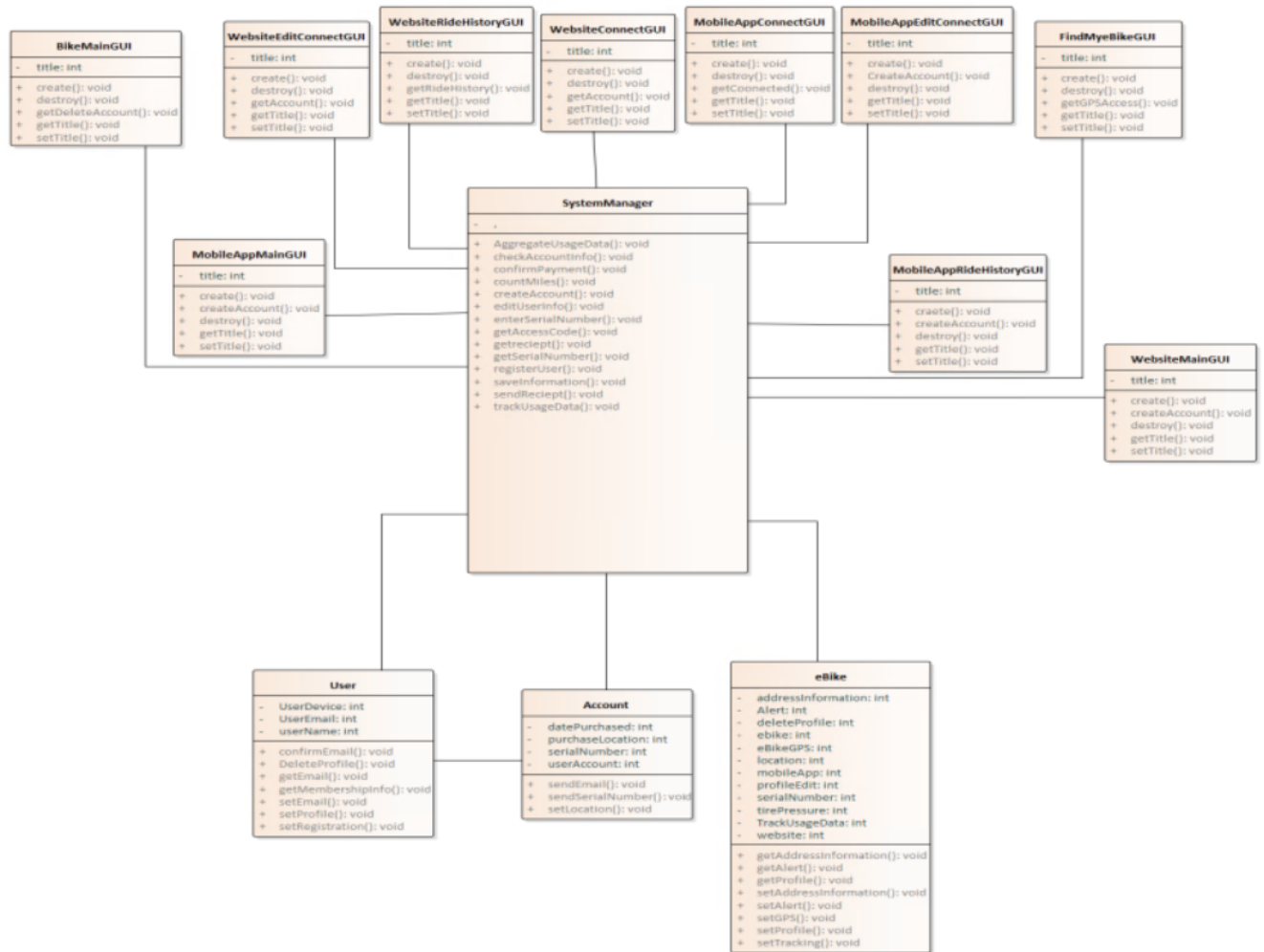




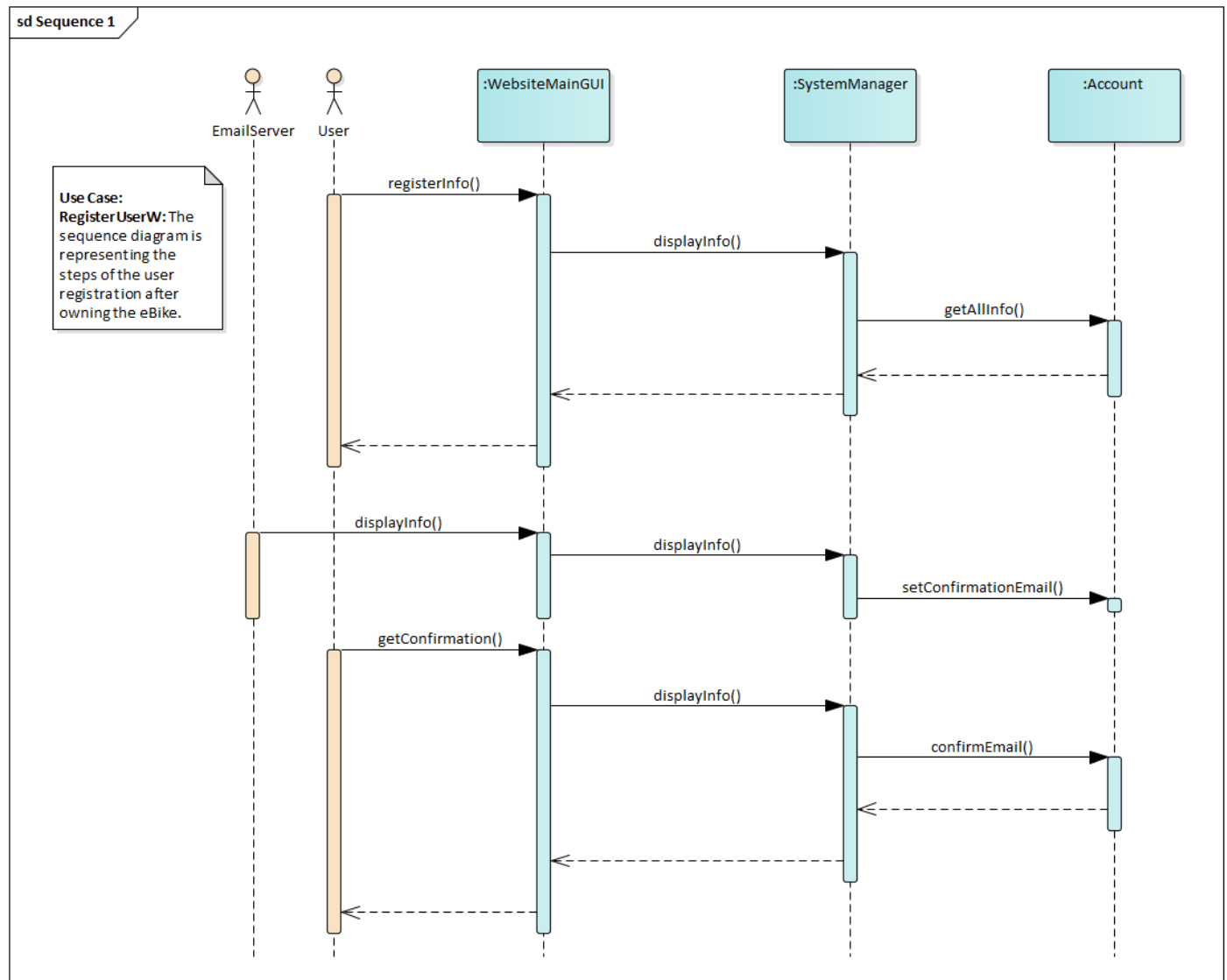
6. Design

6.1 Classes Design Level

class class

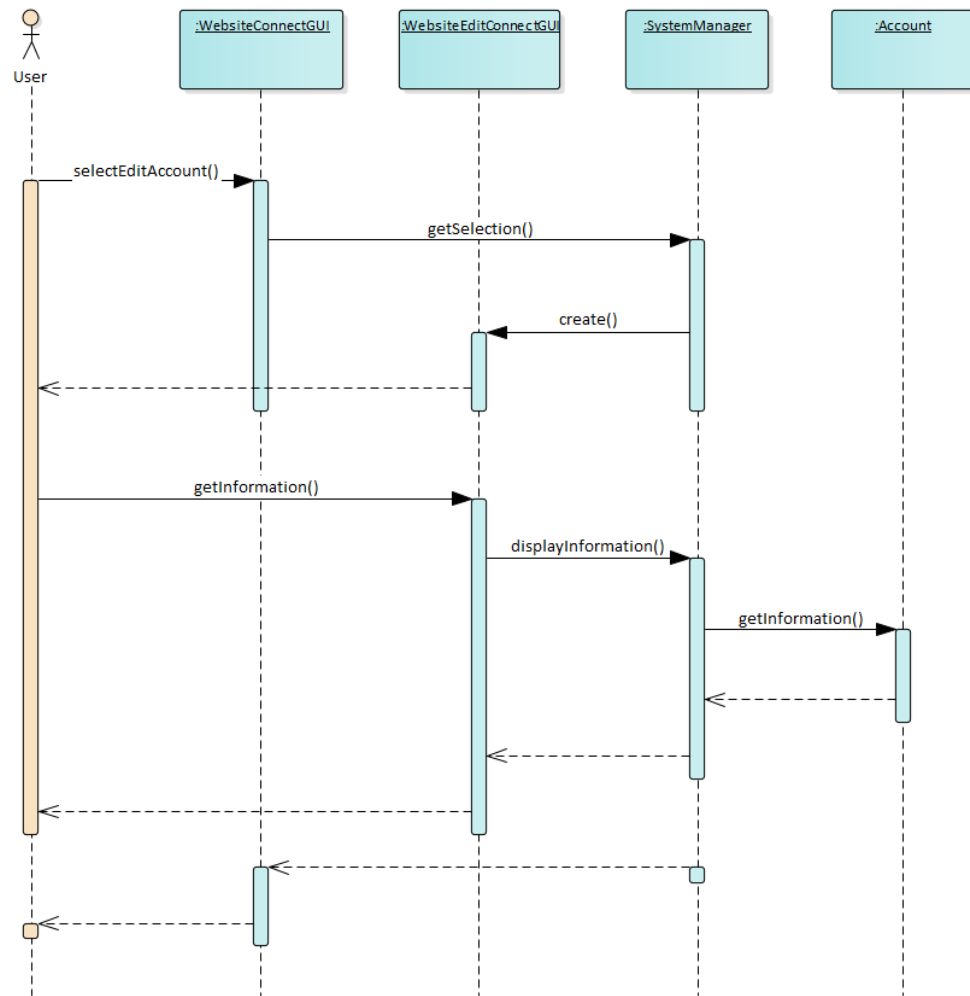


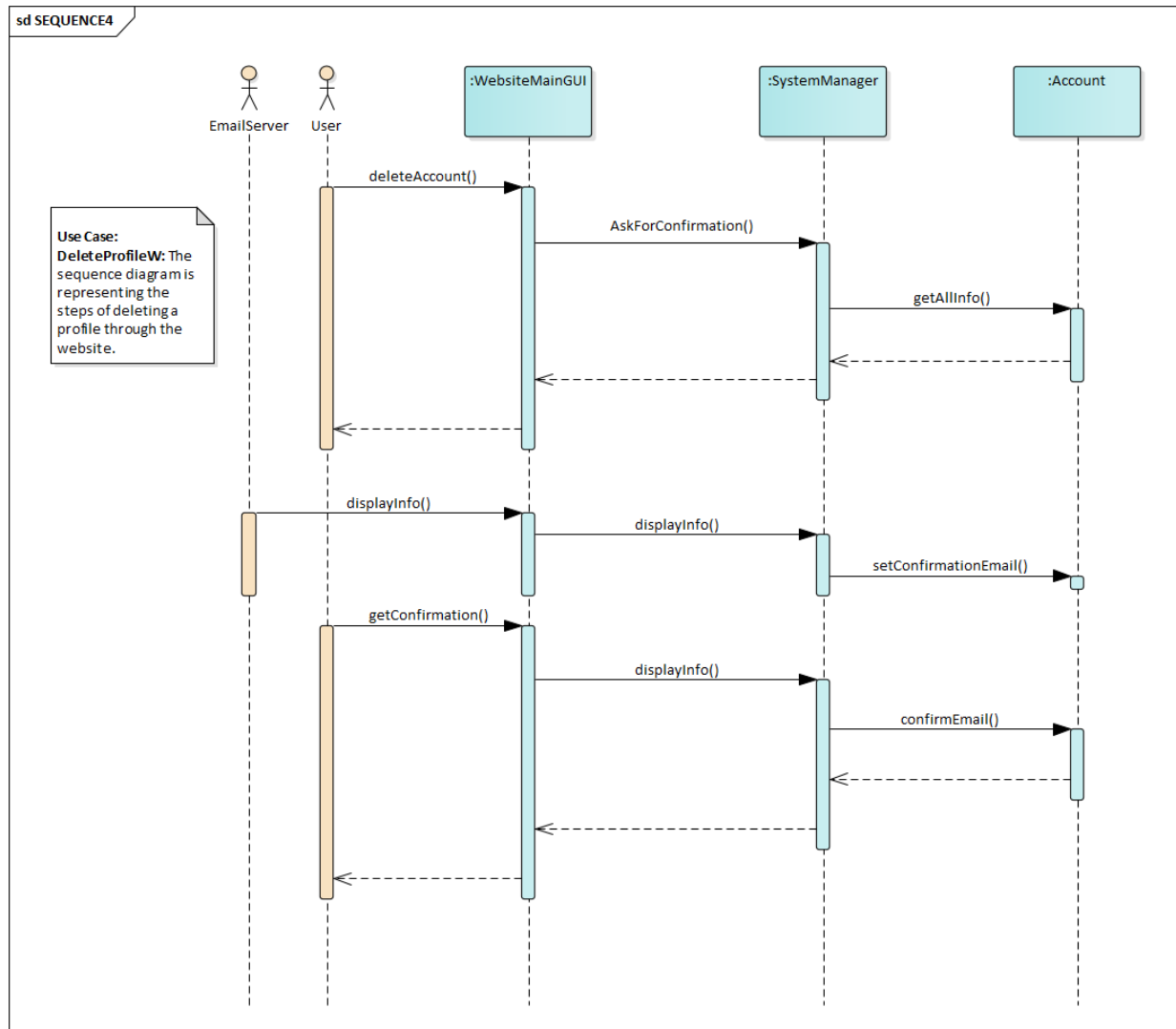
6.2 Use Case Realizations at Design Level

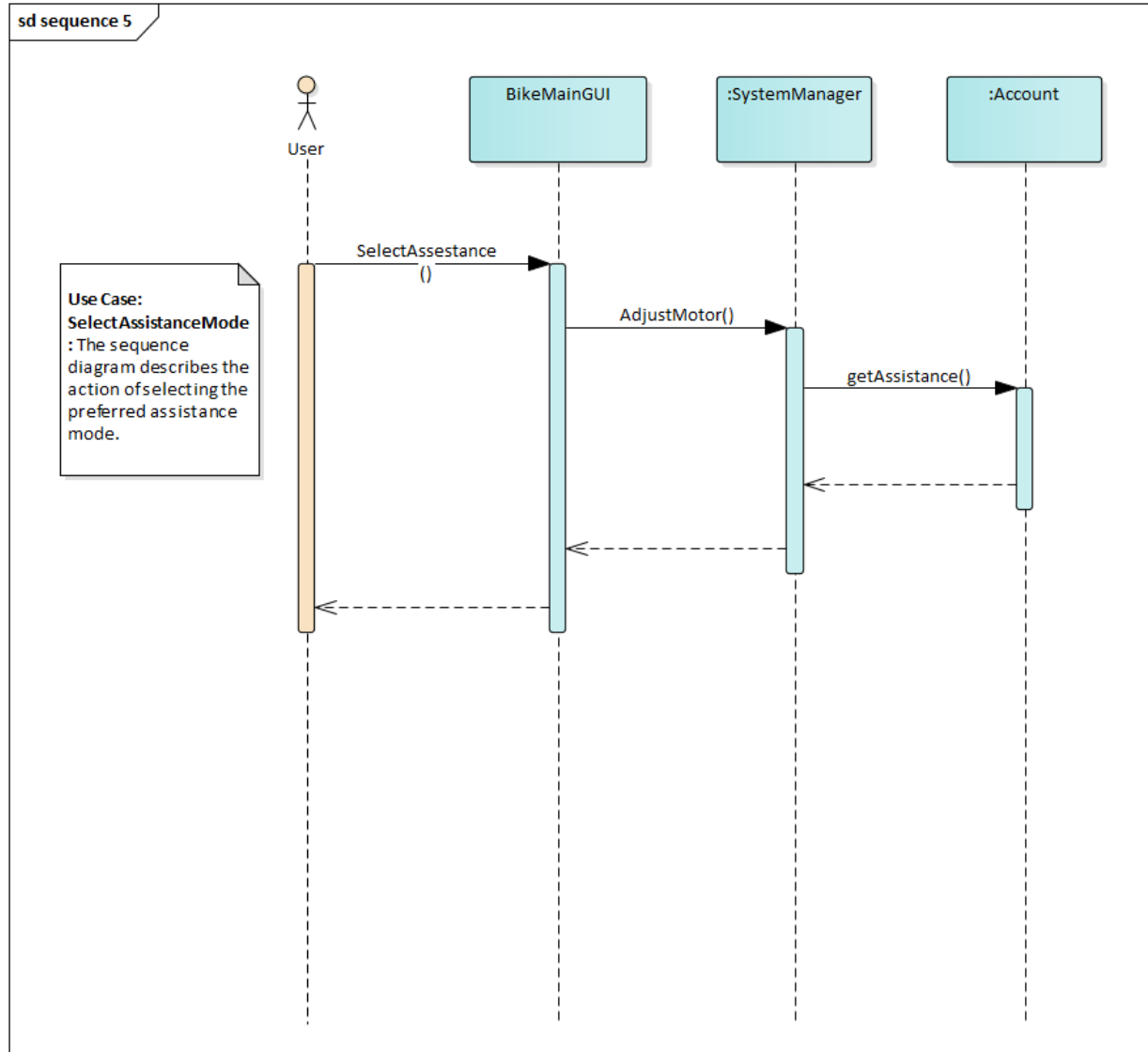


sd sequence 5

Use Case:EditConnectAccountW: The sequence diagram describes the action of editing an account from the Website.

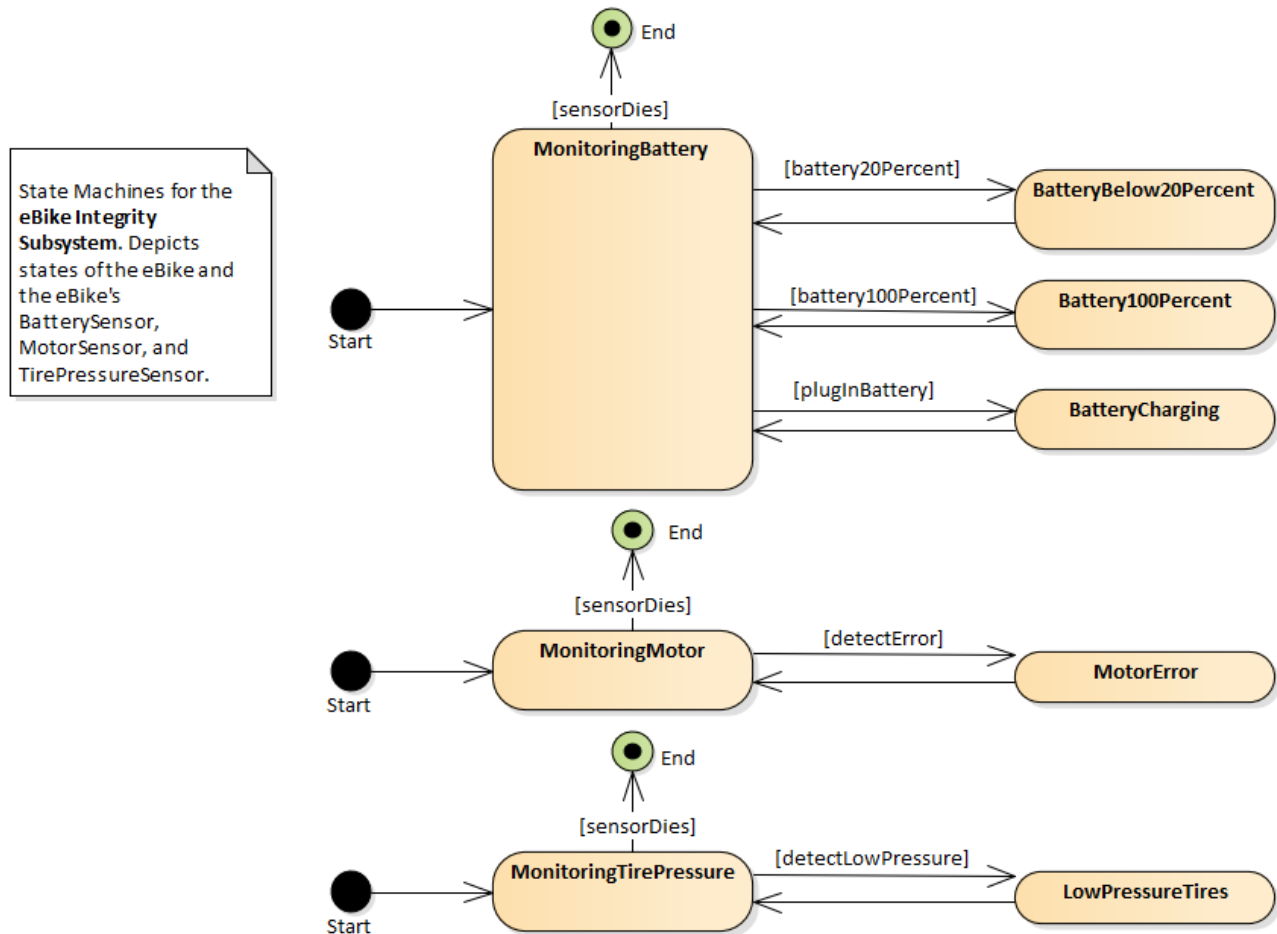




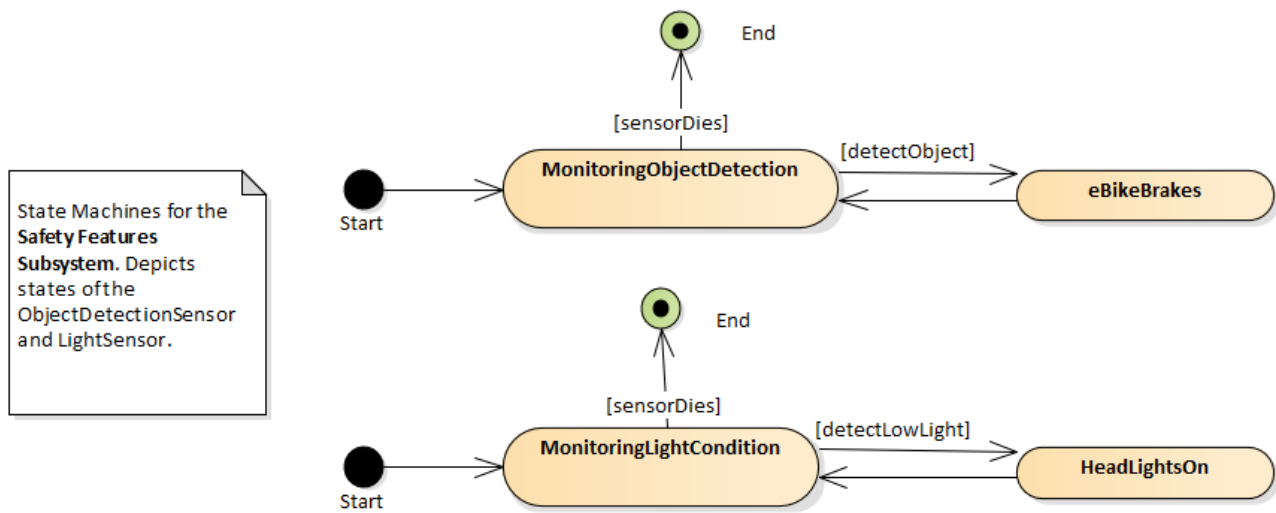


6.2 State Machines

6.2.1 eBike Integrity Subsystem State Machines

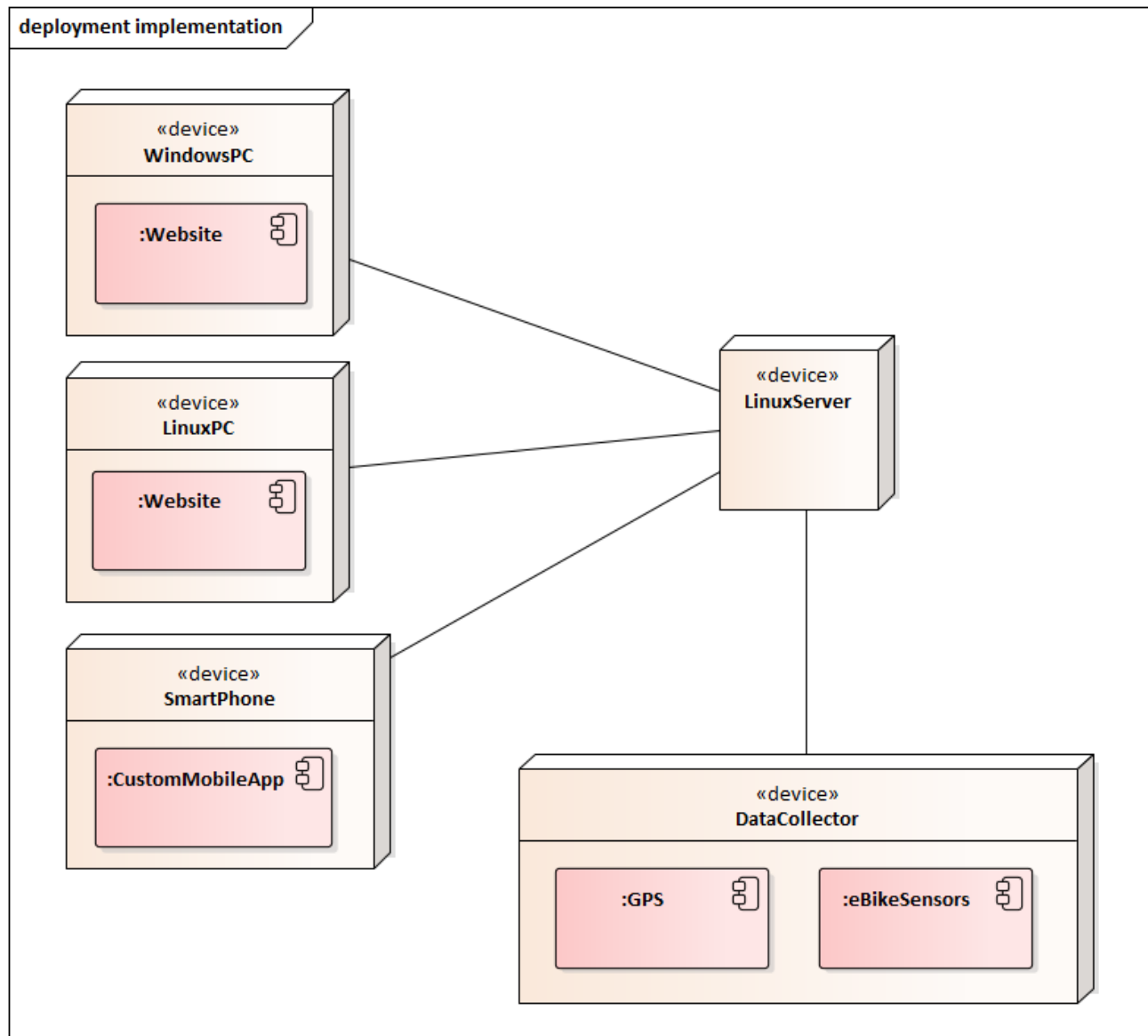


6.2.2 Safety Features Subsystem State Machines

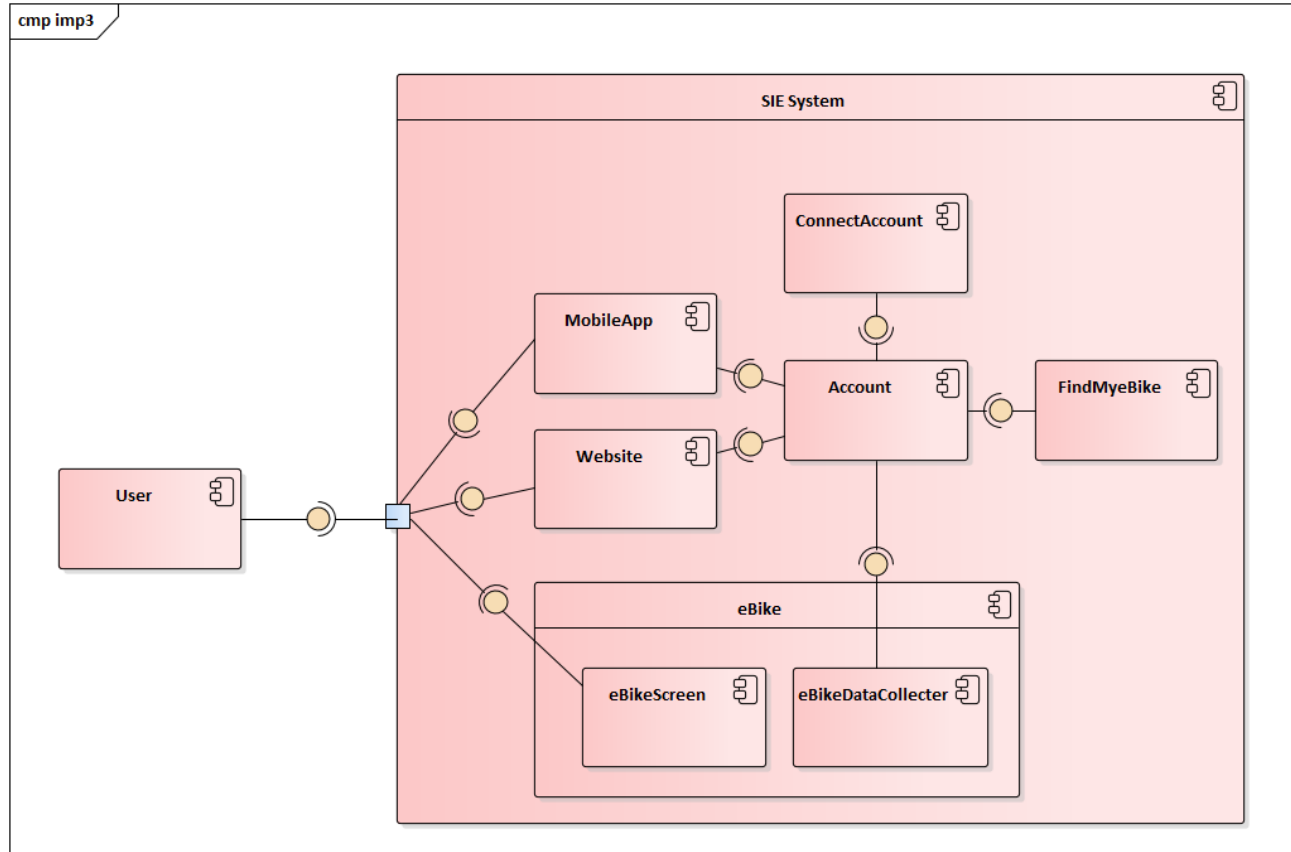


7. Implementation

7.1 System for Intelligent eBikes Deployment Model



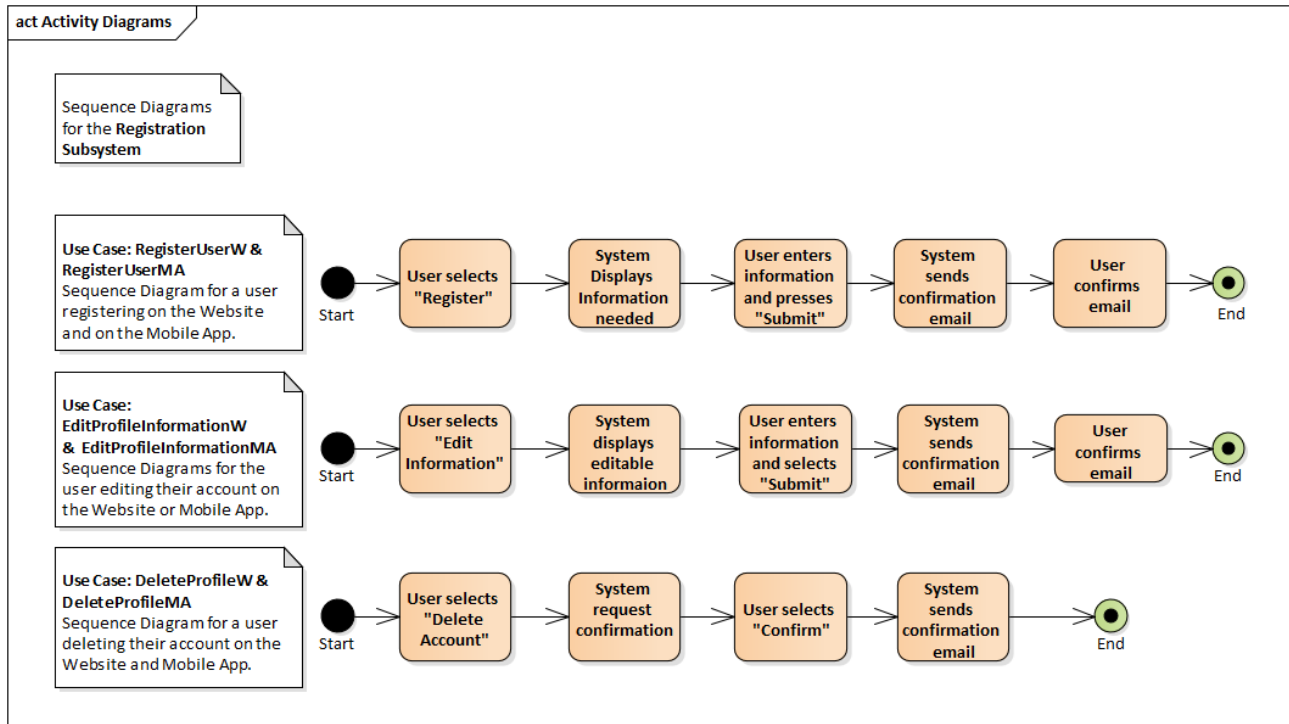
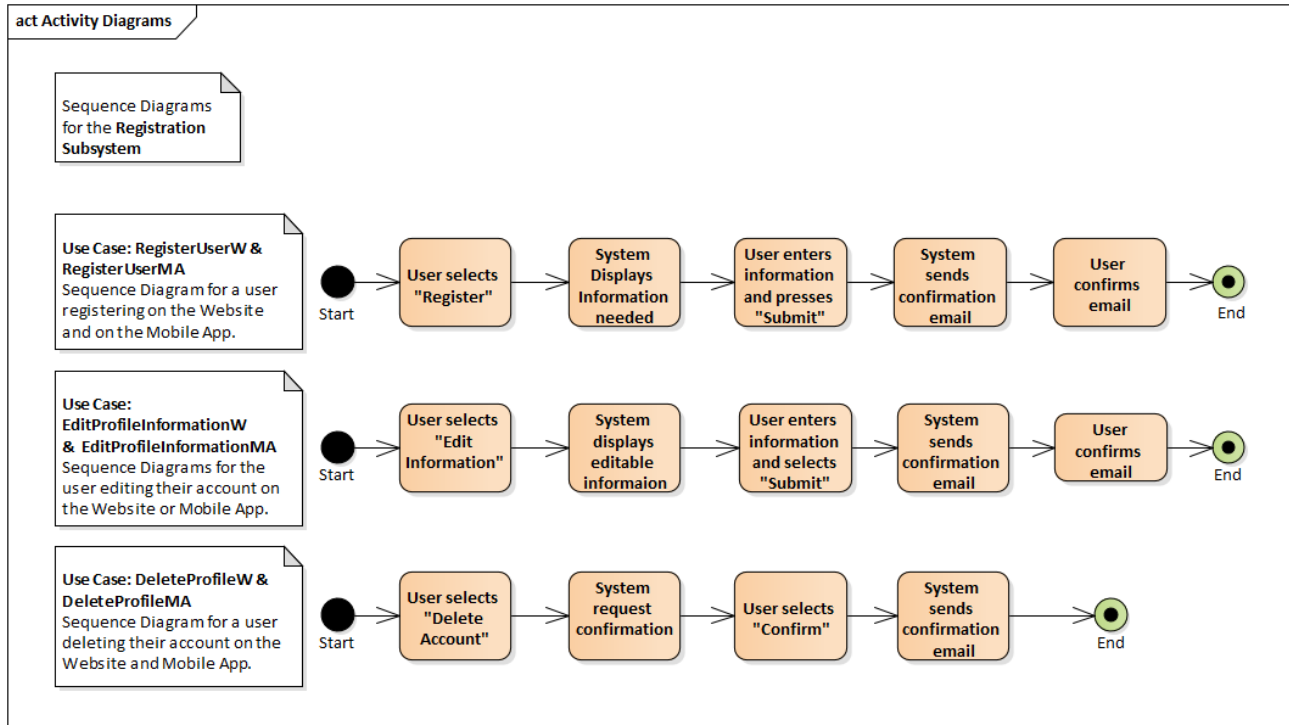
7.2 System for Intelligent eBikes Component Model



- W - Website
- MA - Mobile App

Use Cases	Requirements
Registration Subsystem	Registration Subsystem
4.1.1.1.1	4.1.1.1.1
4.1.1.1.2	4.1.1.1.2
4.1.1.1.3	4.1.1.1.3
4.1.1.1.4	4.1.1.1.4
4.1.1.1.5	4.1.1.1.5
4.1.1.1.6	4.1.1.1.6
User Statistics Subsystem	User Statistics Subsystem
4.1.1.2.1	4.1.1.2.1
4.1.1.2.2	4.1.1.2.2
4.1.1.2.3	4.1.1.2.3
4.1.1.2.4	4.1.1.2.4
4.1.1.2.5	4.1.1.2.5
eBike Integrity Subsystem	eBike Integrity Subsystem
4.1.1.3.1	4.1.1.3.1
4.1.1.3.2	4.1.1.3.2
4.1.1.3.3	4.1.1.3.3
4.1.1.3.4	4.1.1.3.4
Safety Features Subsystem	Safety Features Subsystem
4.1.1.4.1	4.1.1.4.1
4.1.1.4.2	4.1.1.4.2
4.1.1.4.3	4.1.1.4.3
4.1.1.4.4	4.1.1.4.4
Security Subsystem	Security Subsystem
4.1.1.5.1	4.1.1.5.1
4.1.1.5.2	4.1.1.5.2
Biking Control Subsystem	Biking Control Subsystem
4.1.1.6.1	4.1.1.6.1
eBike Connect Subsystem	eBike Connect Subsystem
4.1.1.7.1	4.1.1.7.1
4.1.1.7.2	4.1.1.7.2
4.1.1.7.3	4.1.1.7.3
4.1.1.7.4	4.1.1.7.4
4.1.1.7.5	4.1.1.7.5

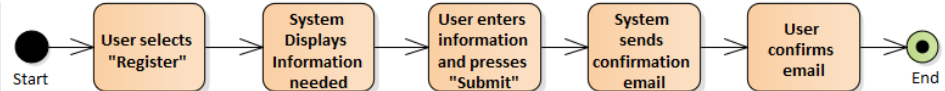
Appendix C: Analysis Models



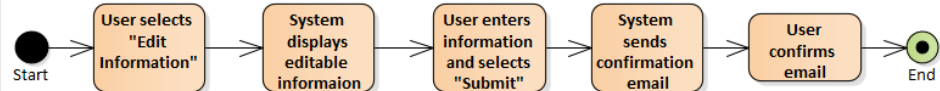
act Activity Diagrams

Sequence Diagrams
for the Registration
Subsystem

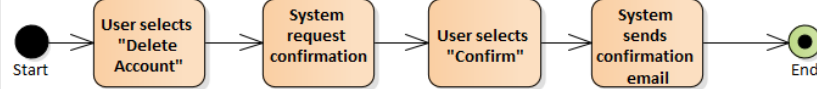
Use Case: RegisterUserW & RegisterUserMA
Sequence Diagram for a user registering on the Website and on the Mobile App.



Use Case: EditProfileInformationW & EditProfileInformationMA
Sequence Diagrams for the user editing their account on the Website or Mobile App.



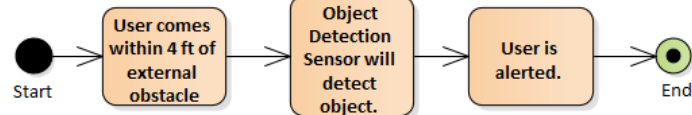
Use Case: DeleteProfileW & DeleteProfileMA
Sequence Diagram for a user deleting their account on the Website and Mobile App.



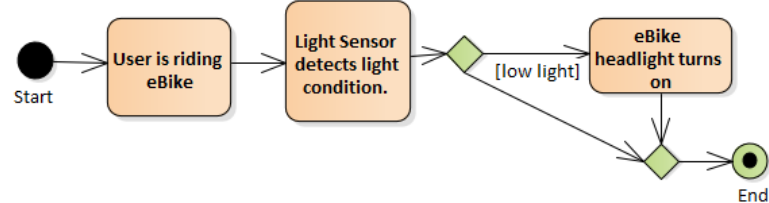
act Safety Features

Sequence Diagrams
for the Safety Features
Subsystem

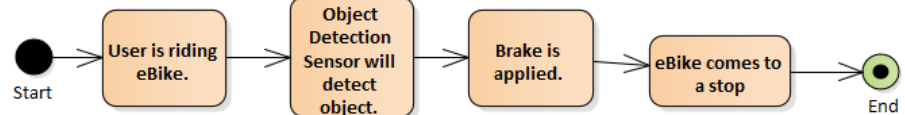
Use Case: ObstacleAlert
Sequence Diagram for the system detecting and alerting the User of obstacles.



Use Case: HeadlightControl
Sequence Diagram for the system detecting lighting and controlling eBike headlight.



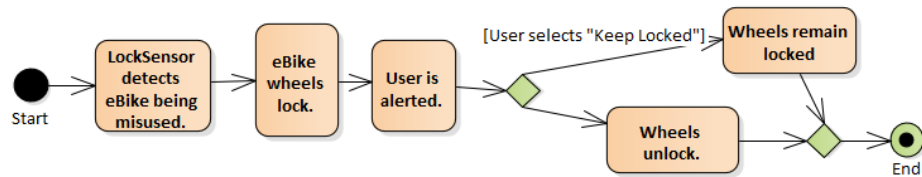
Use Case: Frontbrake
Sequence Diagram for the system detecting and braking for obstacles.



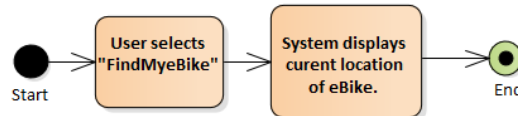
act Security

Sequence Diagrams
for the Security
Subsystem

Use Case: TheftAlert
Sequence Diagram for the
system detecting and alerting
the User of potential theft.



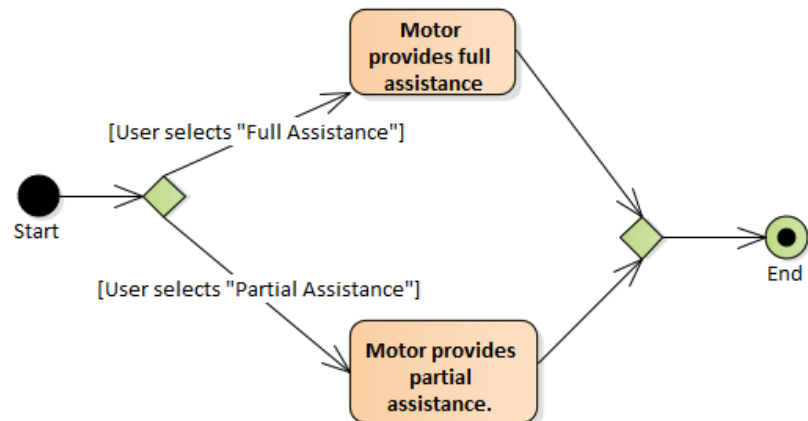
Use Case: FindMyeBike
Sequence Diagram for the user
locating their eBike.

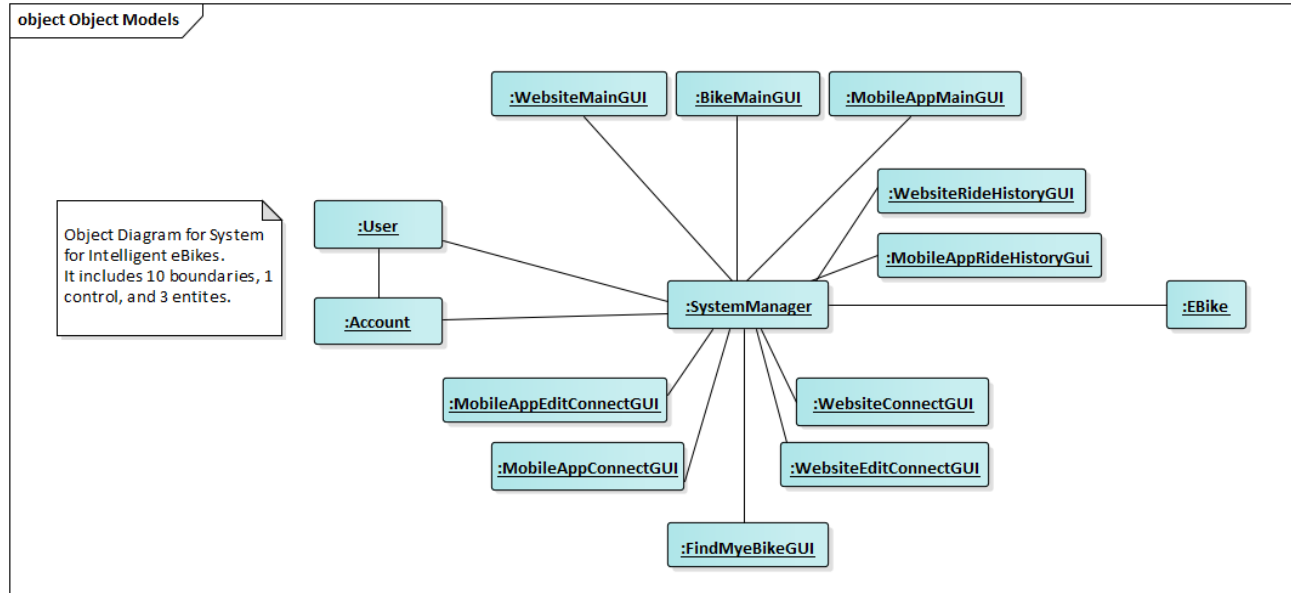
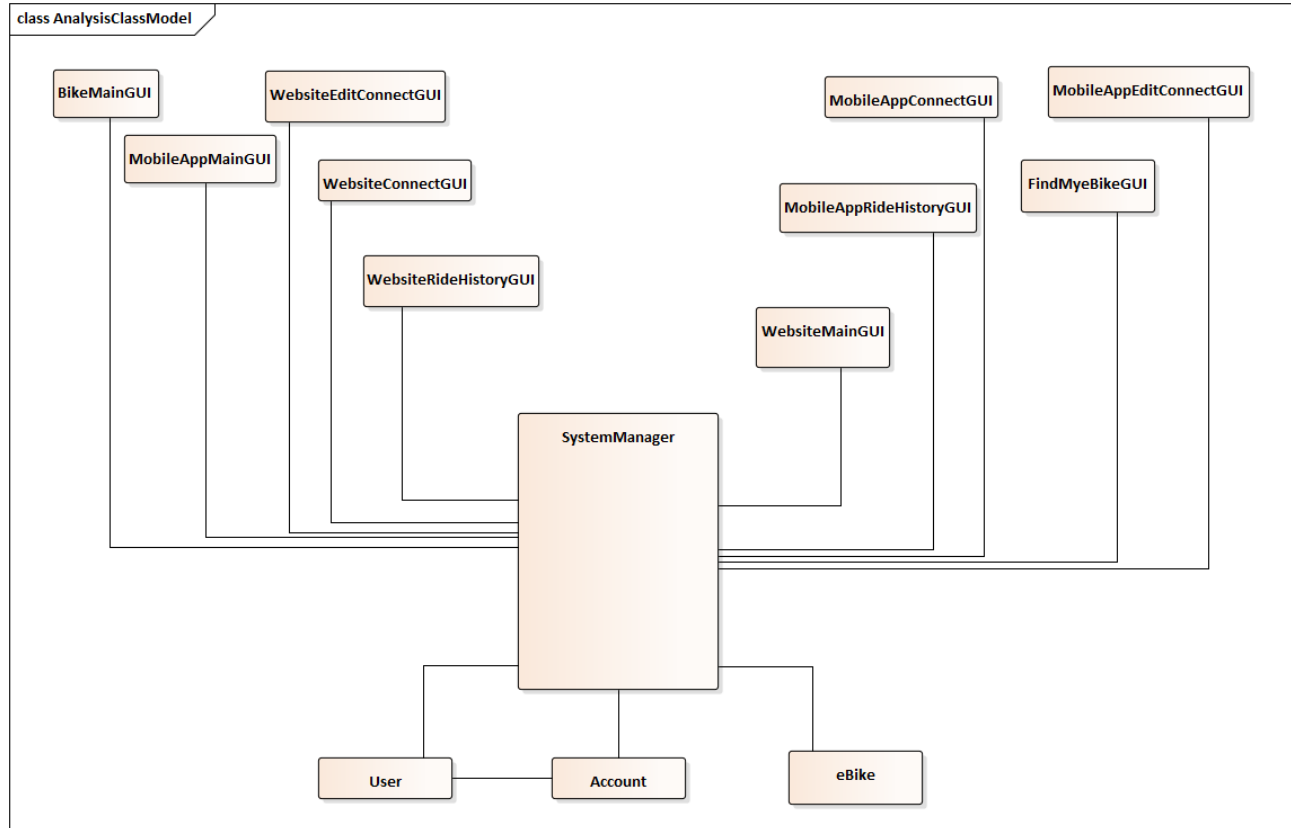


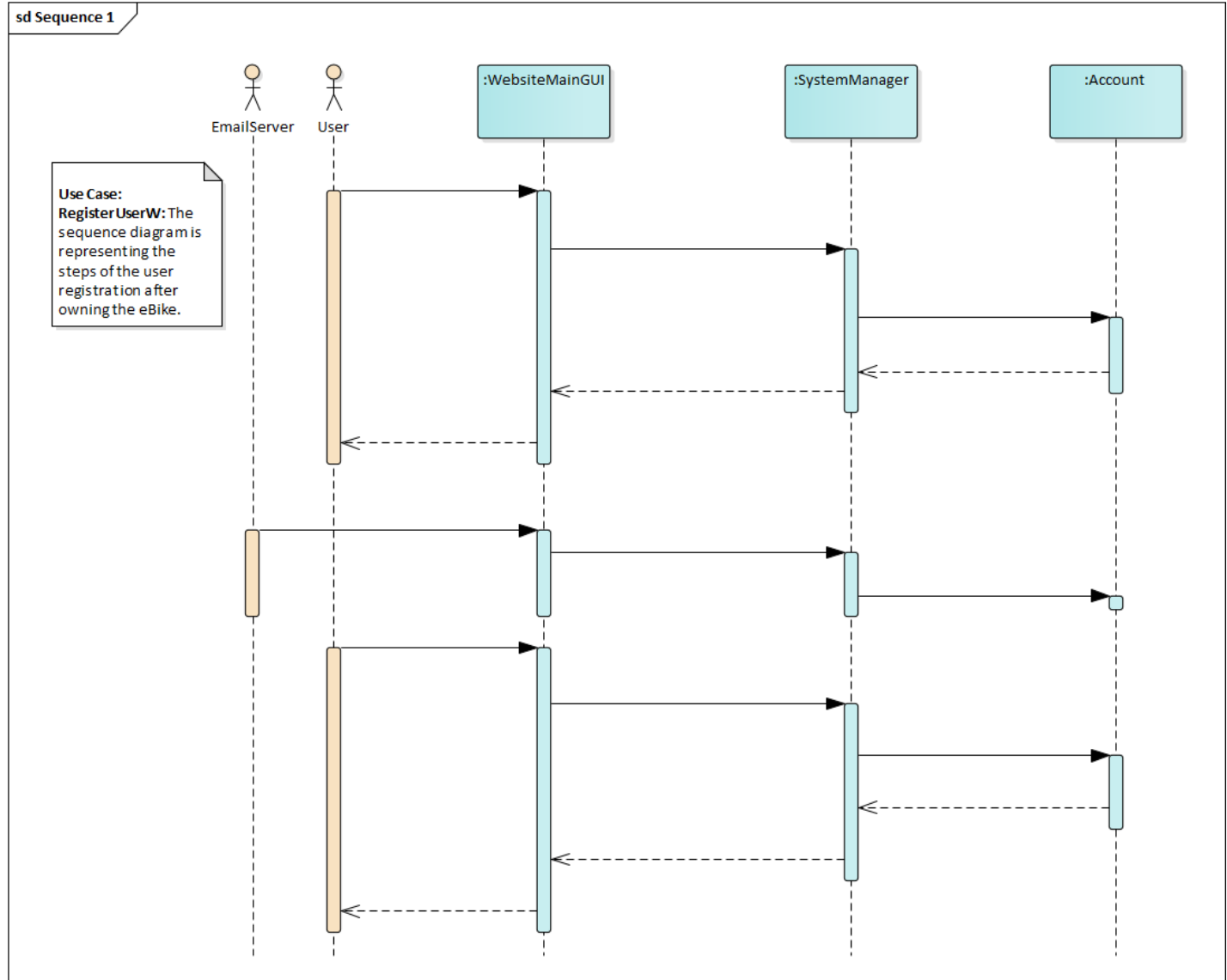
act Biking Control

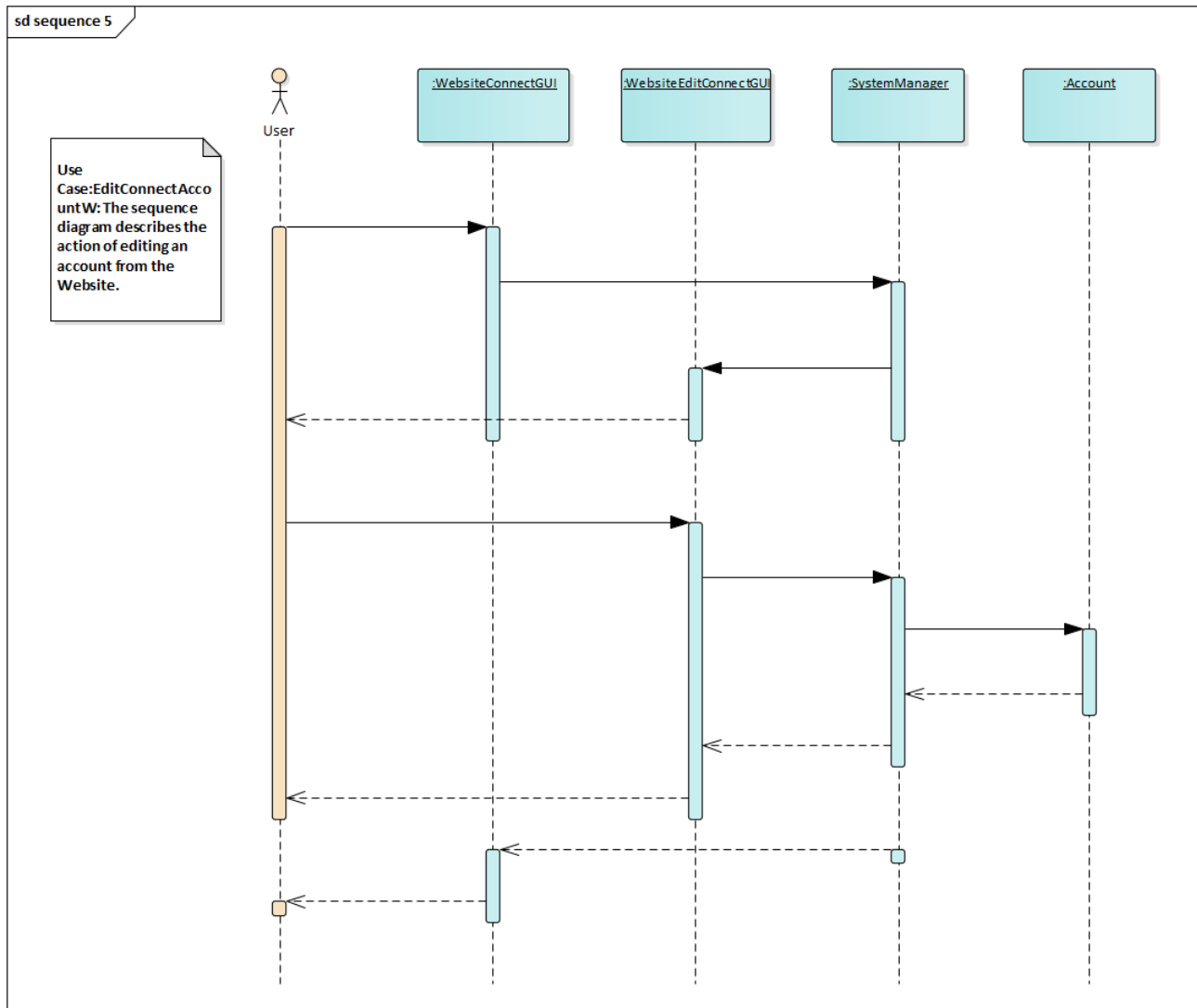
Sequence Diagrams
for the Biking Control
Subsystem

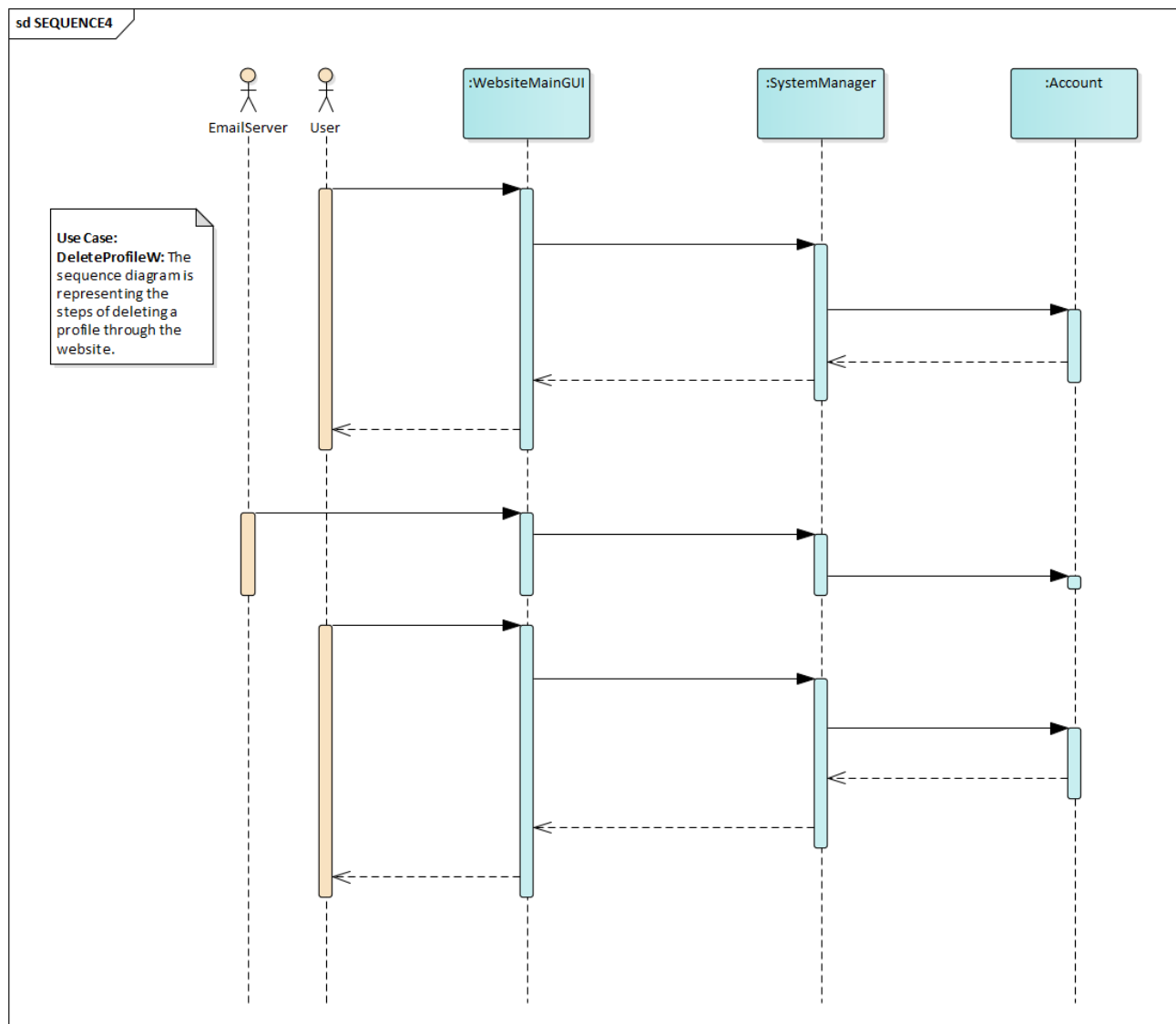
Use Case: FindMyeBike
Sequence Diagram adjusting
the motor assistance.

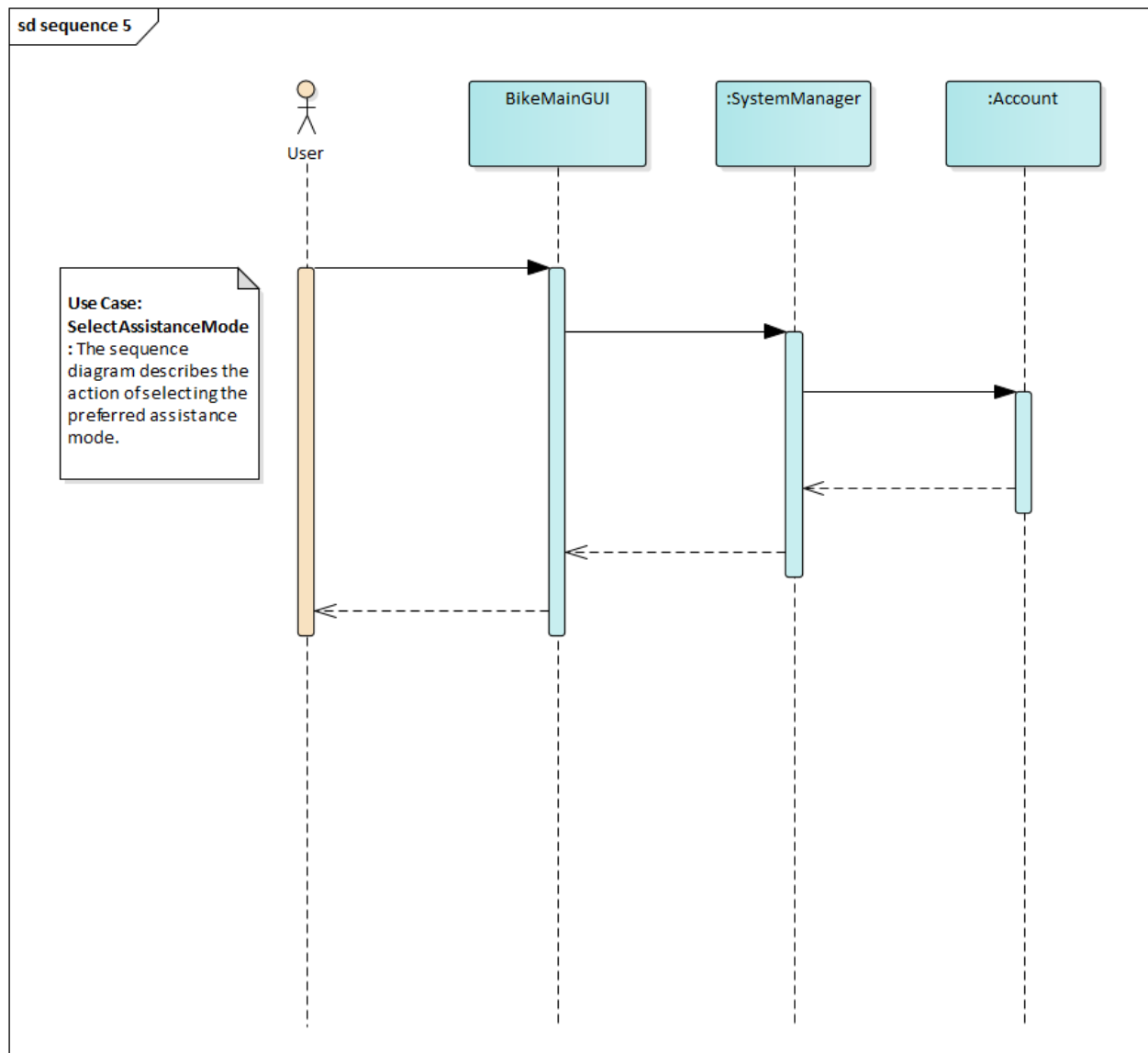






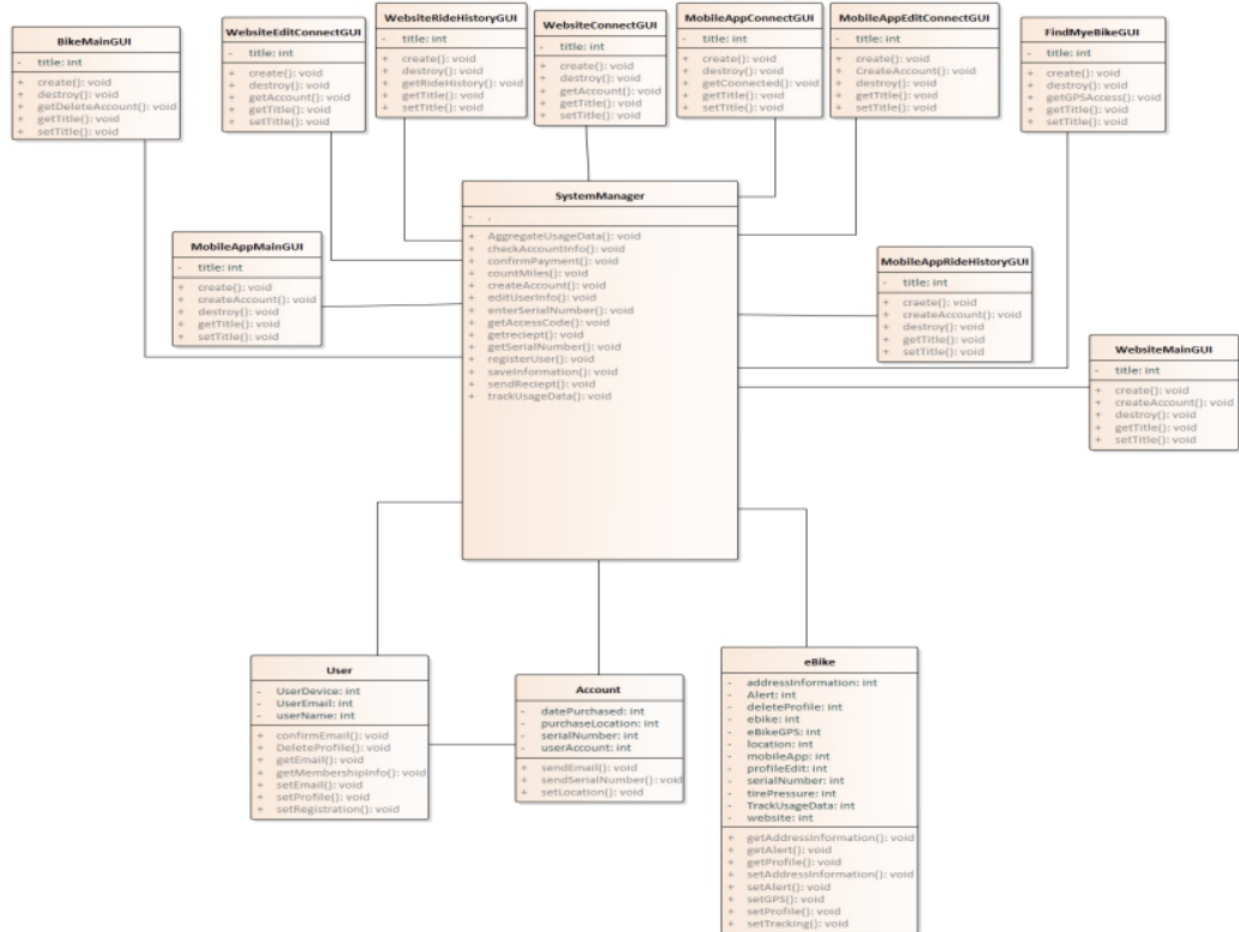




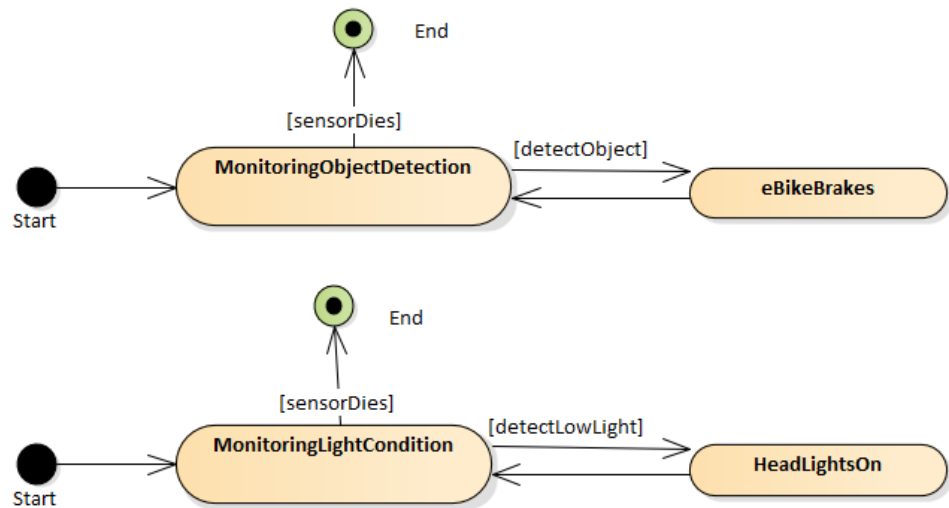


Appendix D: Design Models

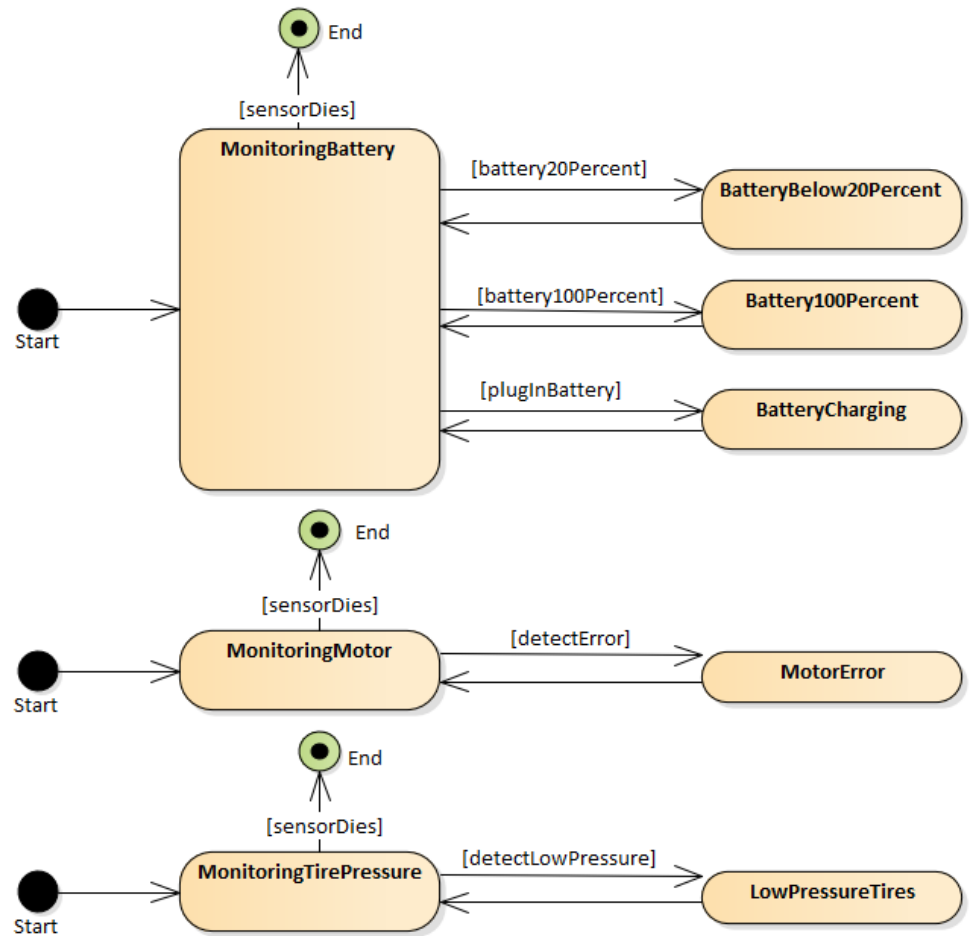
class class

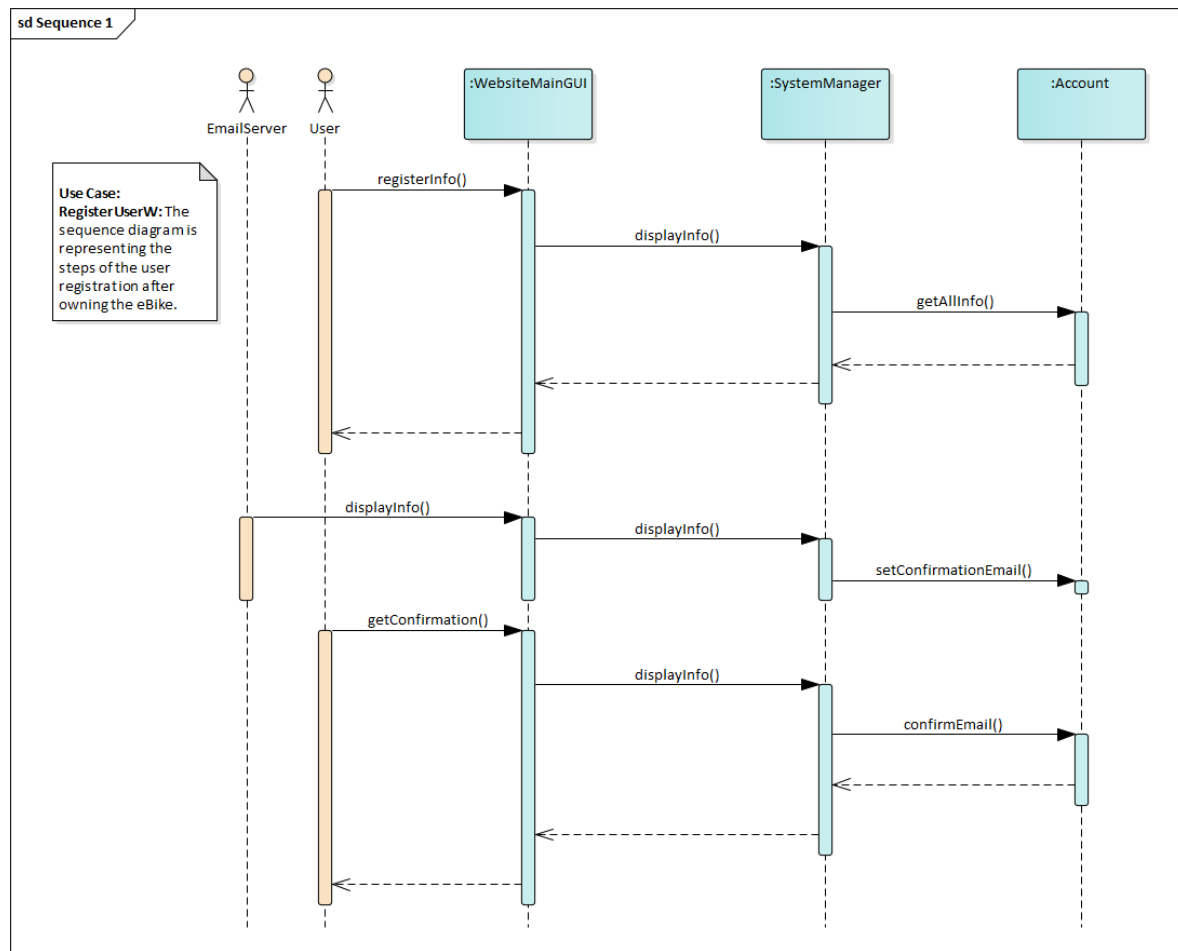


State Machines for the **Safety Features Subsystem**. Depicts states of the ObjectDetectionSensor and LightSensor.



State Machines for the eBike Integrity Subsystem. Depicts states of the eBike and the eBike's BatterySensor, MotorSensor, and TirePressureSensor.





sd sequence 5

Use Case: EditConnectAccountW: The sequence diagram describes the action of editing an account from the Website.

