HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

School of Information and communications technology

Software Requirement Specification

Version 1.0

EcobikeRental

Subject: Software Design and Construction

Group 17

Student Name StudentID

Tran Thi Hong Nhung 20183965

Vu Thi Ngoc Lan 20183939

Duong Hue Linh 20183942

*Hanoi,* *10-2021*

Table of contents

Table of contents 1

1 Introduction 2

1.1 Objective 2

1.2 Scope 2

1.3 Glossary 2

1.4 References 2

2 Overall Description 3

3 Detailed Requirements 4

4 Supplementary specification 6

4.1 Functionality 6

4.2 Usability 6

4.3 Reliability 6

4.4 Performance 6

4.5 Supportability 6

4.6 Other requirements 6

# Introduction

*<The following subsections of the Software Requirements Specifications (SRS) document should provide an overview of the entire SRS. The thing to keep in mind as you write this document is that you are telling what the system must do – so that designers can ultimately build it. Do not use this document for design!!!>*

## Objective

https://www.geeksforgeeks.org/how-to-write-a-good-srs-for-your-project/

<*Identify the purpose of this SRS and its intended audience. In this subsection, describe the purpose of the particular SRS and specify the intended audience for the SRS*>

## Scope

<*In this subsection:*

1. *Identify the software product(s) to be produced by name*
2. *Explain what the software product(s) will, and, if necessary, will not do*
3. *Describe the application of the software being specified, including relevant benefits, objectives, and goals*
4. *Be consistent with similar statements in higher-level specifications if they exist*

*This should be an executive-level summary. Do not enumerate the whole requirements list here*>

## Glossary

*<Listing and explaining the terms appearing in the software’s profession and this documents. Any assumption of the reader’s prior knowledge or experience on the subject is ill advised>*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Term** | **Explanation** | **Example** | **Note** |
| **1** | token | A piece of data created by server, and contains the user's information, as well as a special token code that user can pass to the server with every method that supports authentication, instead of passing a username and password directly. | JSON Web Token (JWT) | Compact, URL-safe and usable especially in web browser single sign-on (SSO) context. |
| **2** | … |  |  |  |

## References

*<Listing the referenced material used in this documents, including the one related to the project>*

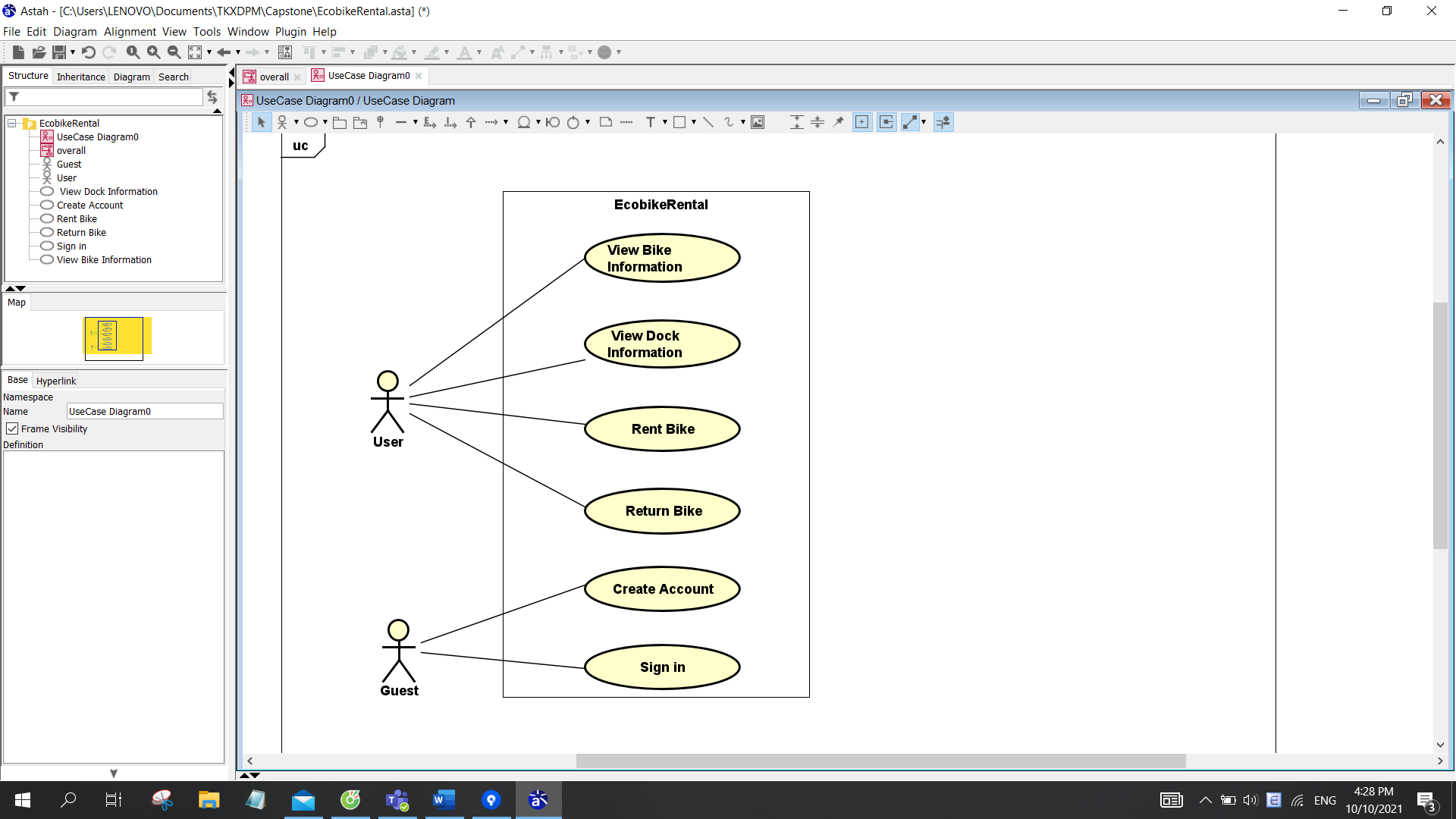
# Overall Description

## Survey

* Ecopark township has an hourly bike rental service with lots of docking stations (i.e., docks) for users to rent or return bikes automatically. EcobikeRental is a software developed for this service. This software helps users to rent and return bikes, view status of docking stations, view information of a bike.
* The software has one actor: User (Guest is not considered because in this project we focus on features related to bike renting and return). User is a role of customer when they had account in system and signed in successfully.

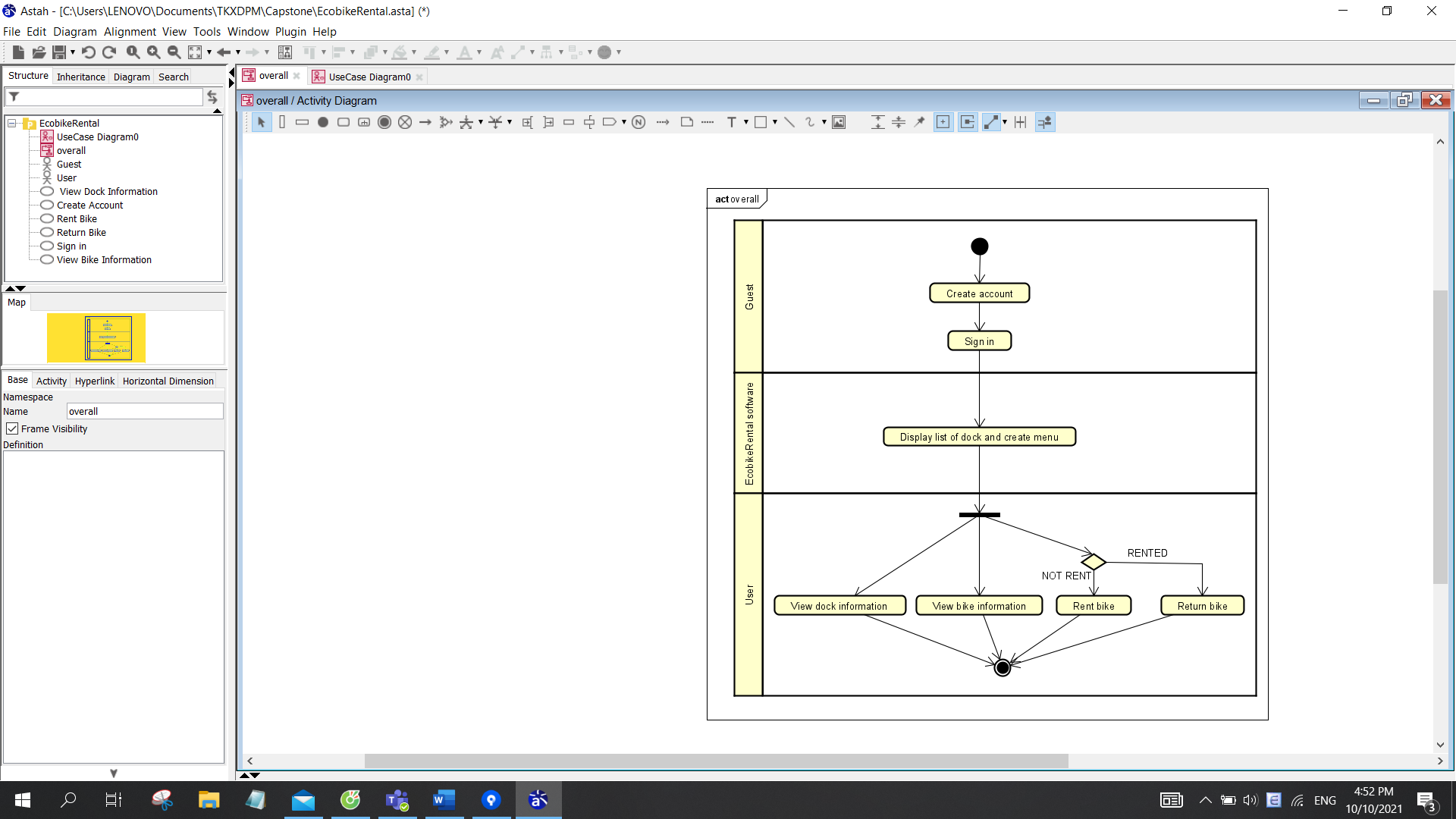
## Overall requirements

In EcobikeRental system, user has got an account with payment information. When user signs in successfully, system displays a list of docks. User can rent bike, return bike, view information of bikes and available dock stations. With one payment method, user can only rent 01 bike. When user wants to return bike, user can choose a dock from a list that system provides.



## Business process

In the first time user opens application, user must create an account with payment information for rental service. After having an account, user sign in and start to access all functions of application.



# Detailed Requirements

## Use case “View Bike Information”

**1. Use case code**

UC001

**2. Brief Description**

This use case describes the interaction between users and EcobikeRental software when the user wishes to view bike information.

**3. Actors**

3.1 User

3.2 EcobikeRental software

**4. Preconditions**

User logged in on system.

**5. Basic Flow of Events**

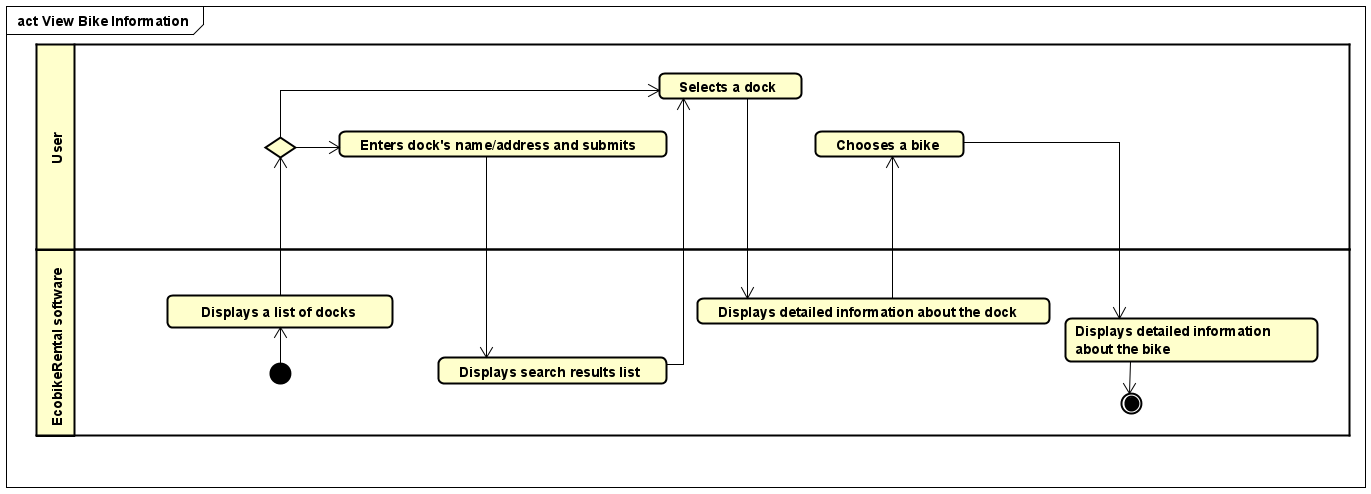
Step 1. The EcobikeRental software displays a list of docks

Step 2. User selects a dock

Step 3. The EcobikeRental software displays detailed information about the dock

Step 4. The customer chooses a bike

Step 5. The EcobikeRental software displays detailed information about the bike



**6. Alternative flows**

N/A

**7. Input data**

*Table 1-Input data of search form*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Data fields | Description | Mandatory | Valid condition | Example |
| 1 | Dock name |  | Yes |  | HBT1 |
| 2 | Address |  | Yes |  | 12 Street, 3 district, HN |

**8. Output data**

*Table 2-Output data of dock information*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Data fields | Description | Display format | Example |
| 1 | Dock name |  |  | HBT1 |
| 2 | Dock ID |  |  | A1234 |
| 3 | Dock address |  |  | 12 Street, 3 district, HN |
| 4 | Dock area |  |  | 50m² |
| 5 | Number of available bikes |  | ▪ positive integer | 20 |
| 6 | Number of empty docking points |  | ▪ positive integer | 30 |
| 7 | Distance |  |  | 200m |
| 7 | Walking time | Walking time from user’s location to this dock (minute) | ▪ positive integer | 30 |

*Table 2-Output data of bike information*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Data fields | Description | Display format | Example |
| 1 | Bike ID |  |  | A1233 |
| 2 | Bike type |  | ▪ standard bicycle / standard e-bike /   twin bike | twin bike |
| 3 | Battery percentage | Battery percentage of e-bikes | ▪ positive integer | 20 |
| 4 | Time limit | Remaining battery time of e-bikes (minute) | ▪ positive integer | 10 |
| 7 | Deposit | 40% of the value of the bike (VND) | ▪ positive integer | 400 000 |

**9. Postconditions**

N/A

## Use case “Rent Bike”

Use case specification

## Use case “Return Bike”

**1. Use case code**

UC004

**2. Brief Description**

This use case describes the interaction between users and EcobikeRental software when the user wishes to return a bike in a dock.

**3. Actors**

3.1 User

**4. Preconditions**

User rented a bike in system. There is an active network connection to the Internet.

**5. Basic Flow of Events**

Step 1. The customer requests to return bike

Step 2. The EcobikeRental software search for available dock points

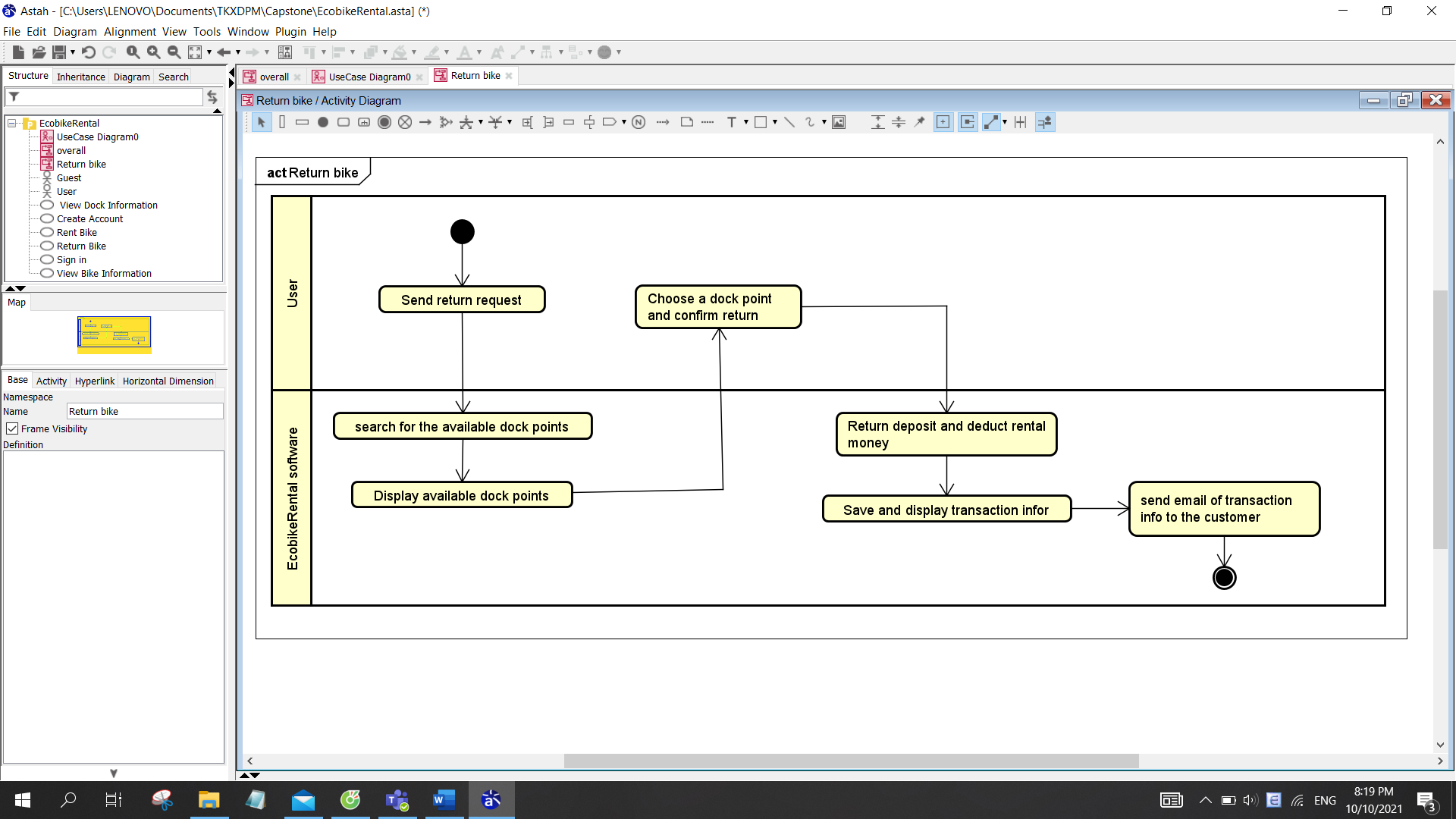
Step 3. The EcobikeRental software displays a list of available dock points

Step 4. The customer chooses a dock point and confirm return bike.

Step 5. The EcobikeRental software returns deposit and deducts rental money

Step 6. The EcobikeRental software saves and displays rental transaction information

Step 7. The EcobikeRental software sends an email of transaction info to the customer



**6. Alternative flows**

N/A

**7. Input data**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Data fields | Description | Mandatory | Valid condition | Example |
| 1 | Dock point ID | Choose from a list | Yes |  | A1233 |
| 2 | Dock point address |  | 12 Street, 3 district, HN |
| 3 | Bike ID |  | Yes |  | A1233 |

**8. Output data**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Data fields | Description | Display format | Example |
| 1 | Cardholder name |  | * Case sensitive | TRAN NHUNG |
| 2 | Card number |  | * Positive integer | 9403 2357 4568 9123 |
| 3 | Issuing bank | Name of the bank issue the card |  | TPBank |
| 4 | Expiration date |  | * mm/yy | 12/26 |
| 5 | Security code |  | * Positive integer | 123456 |
| 6 | Transaction description |  |  | Rent bike |
| 7 | Bike ID | ID of bike rented |  | A1233 |
| 8 | Dock point ID | ID of dock that user return bike |  | A1233 |
| 9 | Return time |  | * hh:mm dd/mm/yy | 12:30 31/01/22 |
| 10 | Rental period | Rental period (minute) | * positive integer | 30 |
| 11 | Currency |  |  | VND |
| 12 | Total | Total rental money | * positive integer | 150,000 |

**9. Postconditions**

N/A

# Supplementary specification

## Functionality

Users firstly need to create an account on theEcoBikeRental application,validate information, set up access permissions of the application, and set up at least onepayment method to pay charges (by linking to interbank or e-wallet)

## Usability

EcoBikeRentalis a 24/7 platform-independent system which allows novice users to use without any training.

## Reliability

The system can be repaired within 2 hours after any typical failure.

## Performance

It is expected to serve 100 users at the same time without noticeable loss of performance and to operate in an average of 200 hours without failure. The response time for the system is 1 second at normal and 2 seconds during a peak load if it is not explicitly stated.

## Supportability

N/A

## Other requirements

N/A