

HANOI UNIVERSITY OF SCIENCES AND TECHNOLOGY

School of Information and communications technology

Personal Report

EcoBike Rental

Subject: ITSS Software Development

Group Number 6

Member name: Phạm Vũ Minh

Member ID: 20176818

Lecturer: PhD. Trinh Tuan Dat

Hanoi, 12/2020

1 Table of contents

Table of contents	1
1 Introduction	3
2 Use Case Analysis.....	4
3 Architectural Design.....	6
4 Interface Design	8
5 Class Design	9
6 Test Plan	11

1 Introduction

1.1 *Objective*

This report details my personal contributions to the project “EcoBike Rental”

1.2 *Scope*

In this project, I'm in charge of the composite use case “Manage bikes” for admin, whose main functions include:

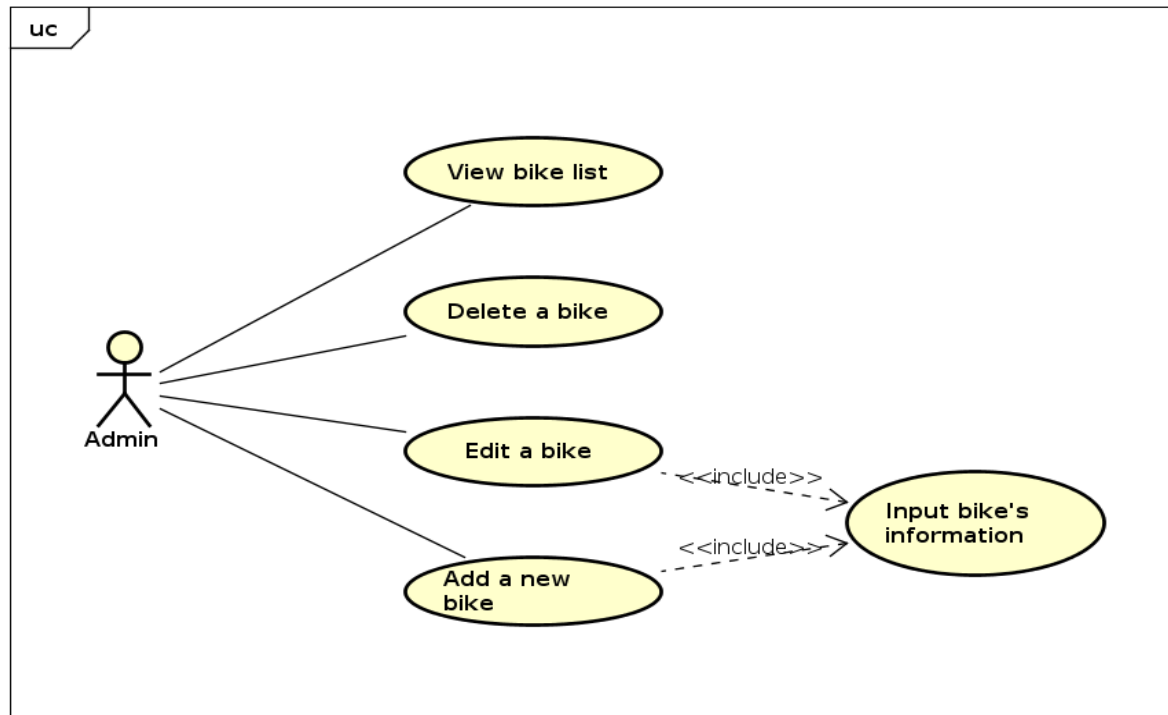
- Presenting a list of all bikes
- Delete a selected bike
- Edit an existing bike
- Add a new bike
- Validate user's input and prompt error message if necessary

Additional contributions include:

- Design and maintain database: create ER diagram and help generate data for database

2 Use case Analysis

2.1 Use case diagram



2.2 Use case Specification

Specification for the use case “Add new bike”

Use case ID	UC001	Use case name	Add New Bike
Actors	Admin, System		
Pre-Condition(s)	- Admin successfully logged in. - Admin has modify rights.		
Basic Path (Success)	No	Proceeded by	Actions
	1.	Admin	Click “Add New” button
	2.	System	Display bike information screen
	3.	System	Generate and display new bike’s ID
	4.	Admin	Input new bike’s information (input description below)
	5.	Admin	Click “Add” button
	6.	System	Validate if the new bike name is unique
	7.	Admin	Validate if the new bike license plate is unique
	8.	System	Add new bike to database

	9.	System	Update bike list
	10.	System	Show updated bike list
Alternative Paths	No	Proceeded by	Actions
	4a.	System	IF any input field is empty, display error message "Invalid Input"
	4b.	System	IF input for weight is not a float, display error message "Invalid Input"
	6a.	System	IF new bike name is not unique, display error message
	7a.	System	IF new license plate is not unique, display error message..
Post-Condition(s)	None		

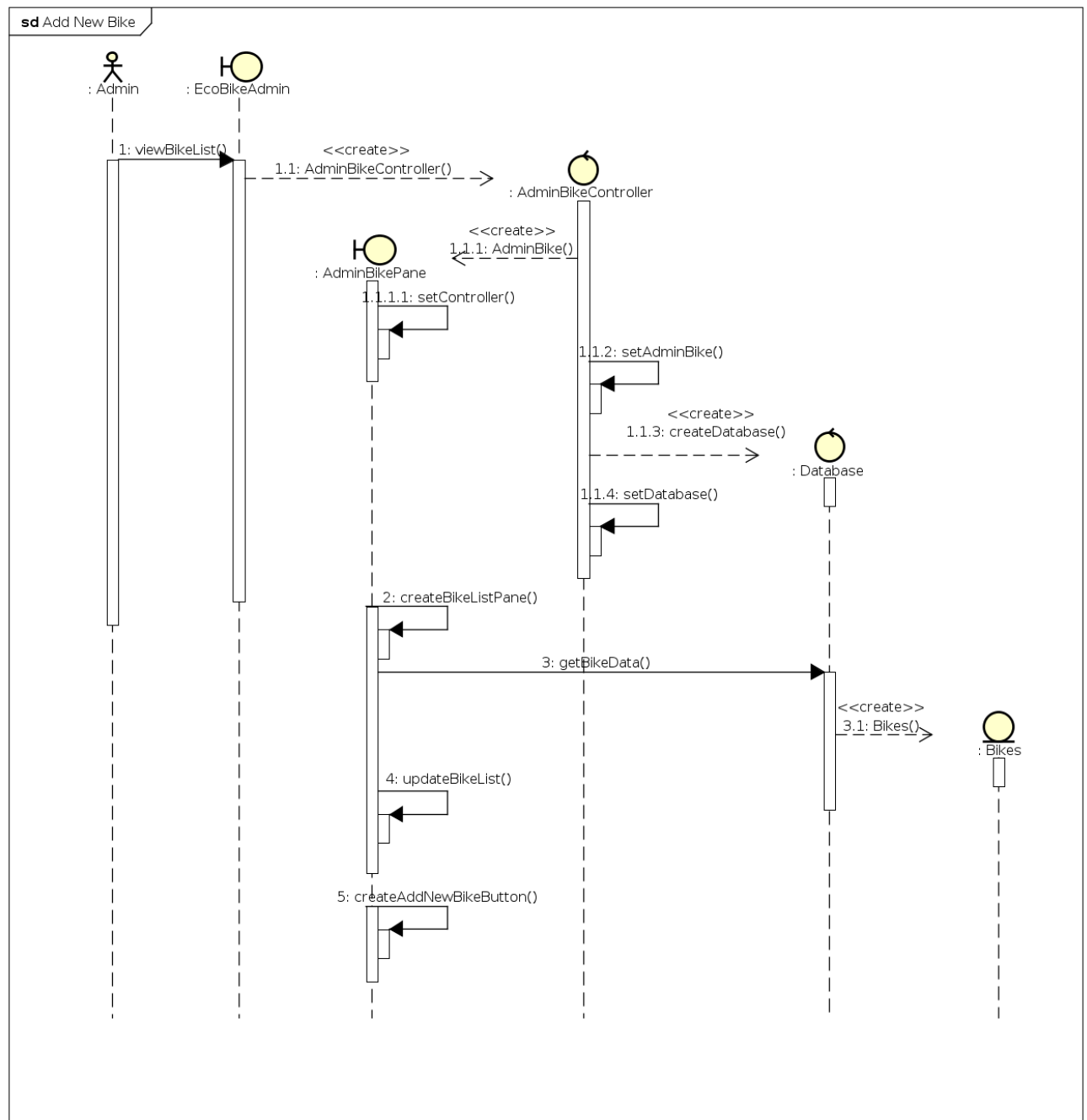
* Input data of added bike information includes these following fields:

ID	Data field	Description	Mandatory	Valid condition	Example
1.	Name	Bike name	Yes	Must not exist in database	Argon 18
1.	Type	Bike type	Yes	Must be "NormalBike", "EcoBike" or "TwinBike"	EcoBike
3.	License Plate	Bike's license plate	Yes	Must not exist in database	068581324-0
4.	Weight	Bike's weight	Yes	Must be a float	13.2
5.	Manufacturing date	Bike's date of manufacturing	Yes	Must be a valid date	2020-12-01
6.	Producer	Bike's producer	Yes	None	Trek
7.	StationID	Station ID of which bike is added to	Yes	Selected station's dock must not be full	1

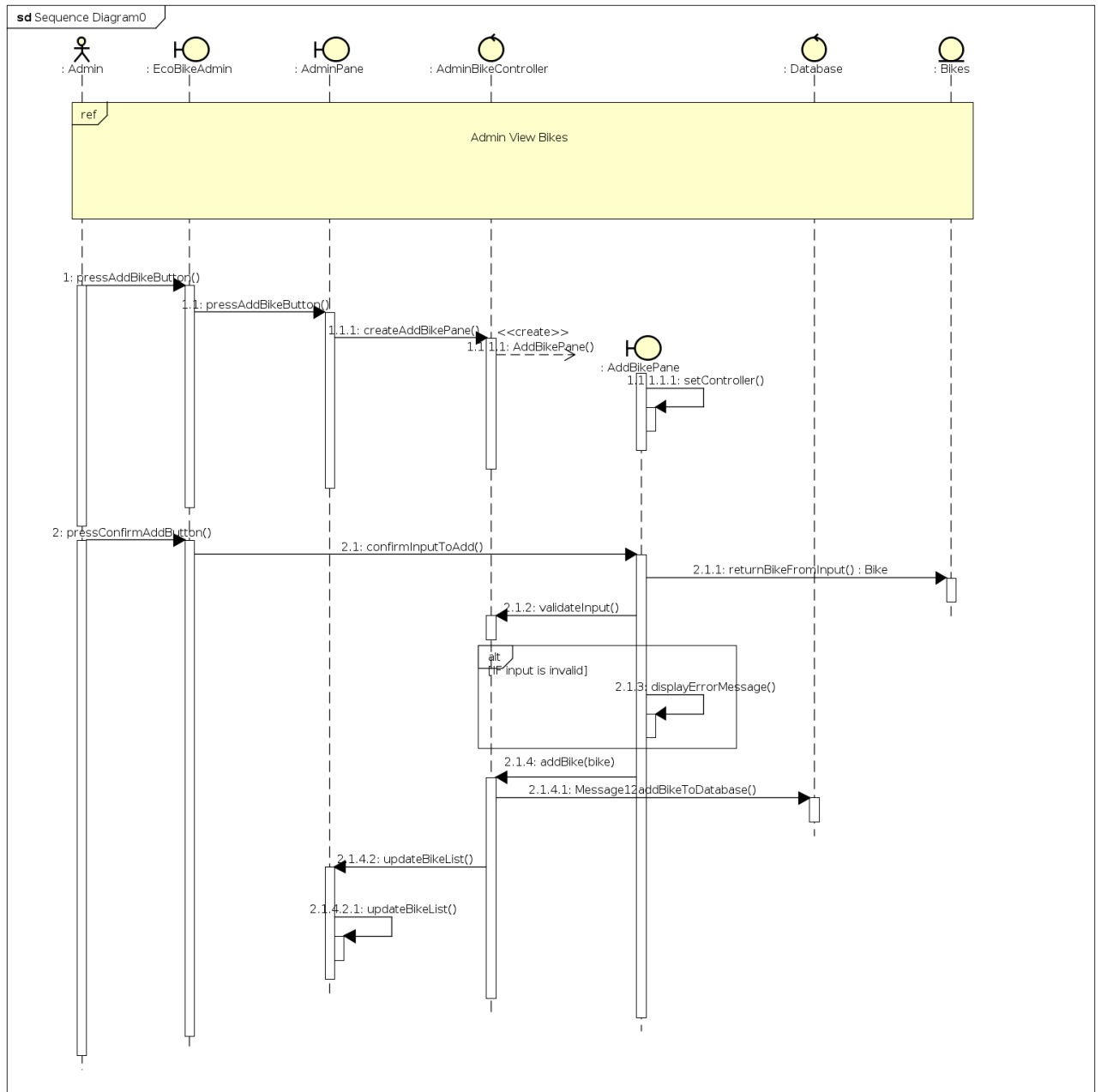
3 Architectural Design

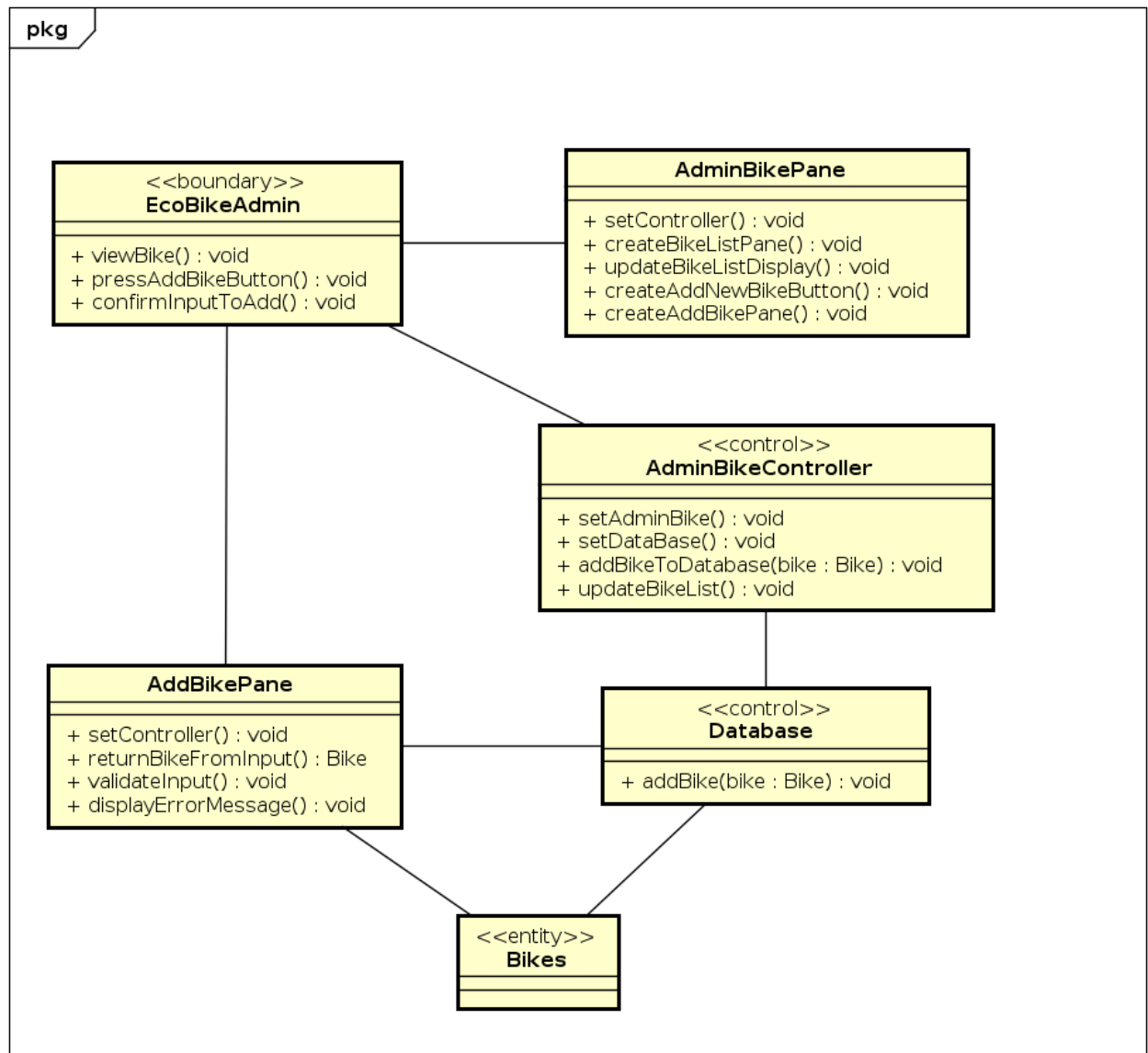
3.1 Sequence Diagram

3.1.1 Admin View All Bikes

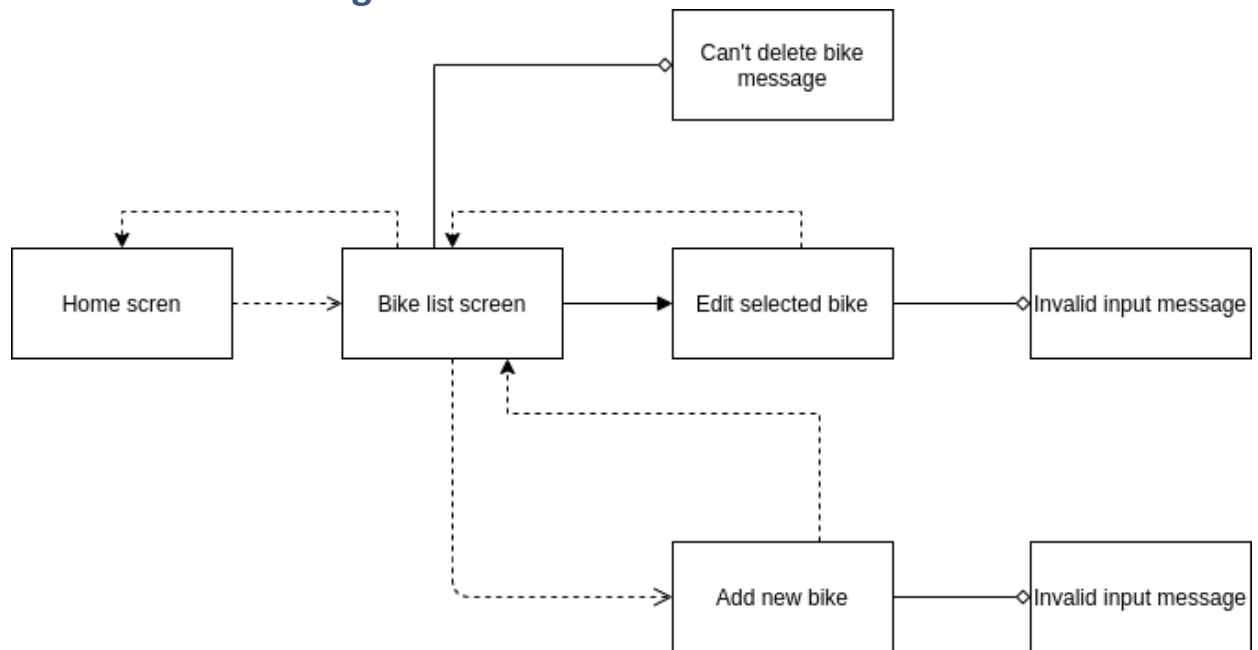


3.1.2 Admin Add Bike





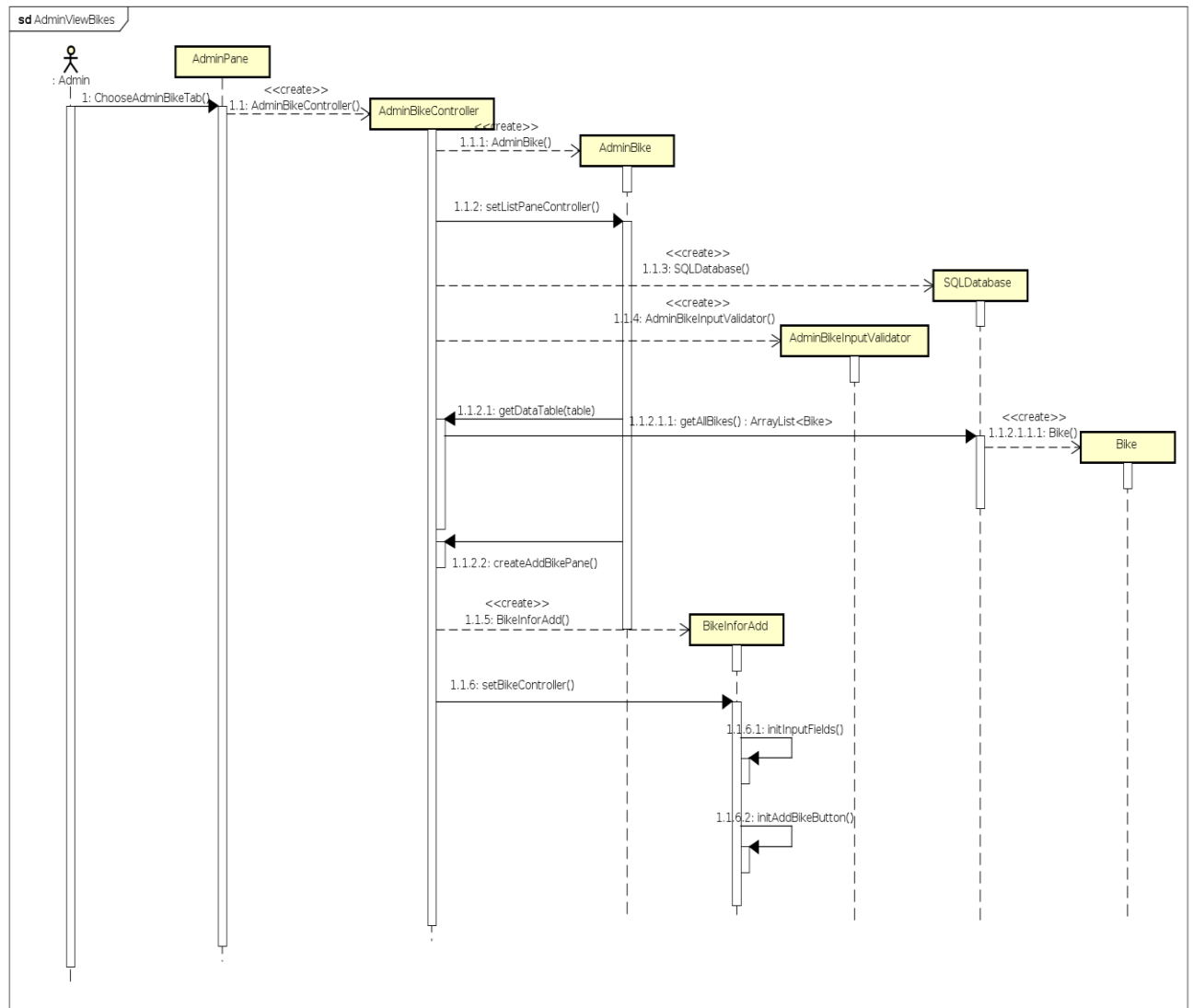
4 Interface Design



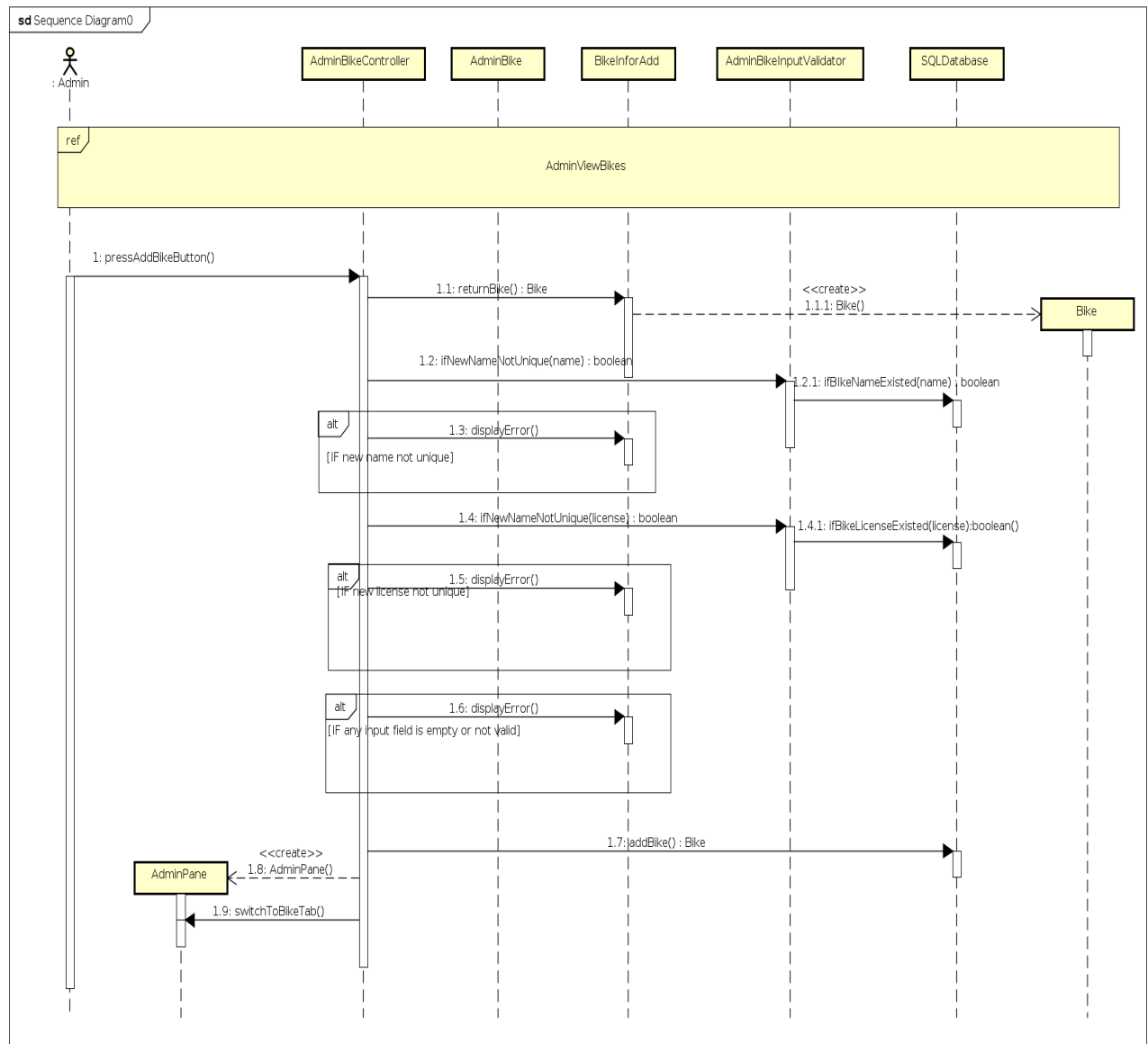
5 Class Design

5.1 Sequence Diagram

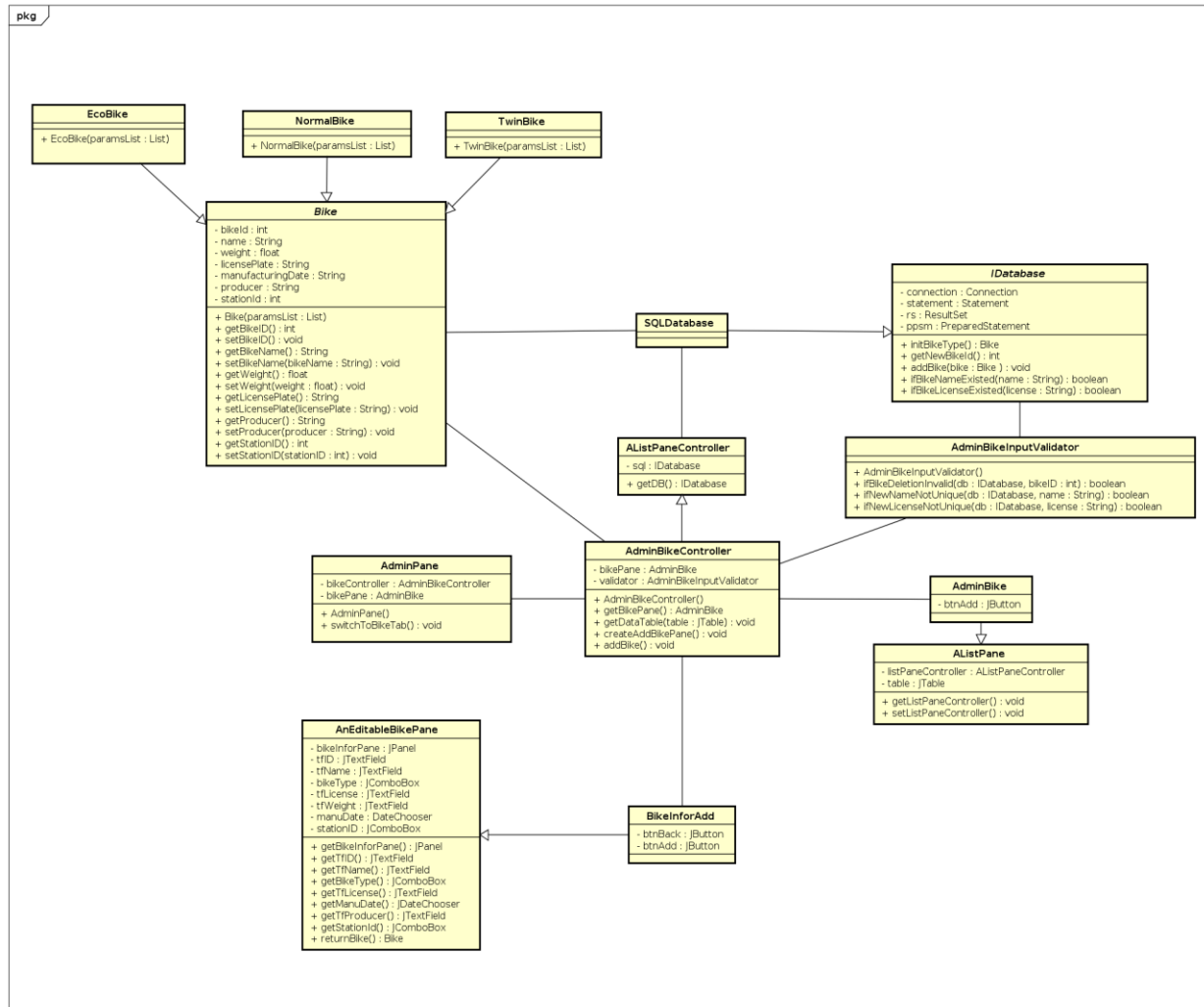
5.1.1 Admin View Bikes



5.1.2 Admin Add Bike



5.2 Class Diagram



6. Test Plan

In this use case, input from user (admin) has to be validated and tested. These validate methods are included in the AdminBikeInputValidator class. I design test cases for the methods of this class using JUnit 4.

6.1 Test case specification for “testBikeDeletion”

Test Case ID	TC001	Test Case Description	Check if a bike selected for deletion is being rented or not		
Created By	Pham Vu Minh	Reviewed By	Pham Vu Minh	Version	1.0

#	Prerequisites:
1	List of rented bike IDs is loaded from database
2	List of unrented bike IDs is loaded from database

#	Test Data	Value
1	rented bike IDs	<loaded from database>
2	unrented bike IDs	<loaded from database>

Test Scenario	Pass a bike ID as an argument, verify whether or not bike is currently being rented
Expected result	Rented bikes should return true, unrented bikes should return false

6.2 Test case specification for “testUniqueNameAfterEdit”

Test Case ID	TC002	Test Case Description	Check if a bike selected for deletion is being rented or not		
Created By	Pham Vu Minh	Reviewed By	Pham Vu Minh	Version	1.0

#	Prerequisites:
1	List of existed bike names is loaded from database

#	Test Data	Value
1	existedNames	<loaded from database>
2	uniqueName	!@#%\$!@#%^@#\$\$@#!#@#\$

Test Scenario	Pass two bike names, if they differ then check whether or not new name already existed in database
Expected result	Return false for all cases

6.3 Test case specification for “testUniqueLicenseAfterEdit”

Test Case ID	TC002	Test Case Description	Check if a bike selected for deletion is being rented or not		
Created By	Pham Vu Minh	Reviewed By	Pham Vu Minh	Version	1.0

#	Prerequisites:
1	List of existed bike licenses is loaded from database

#	Test Data	Value
1	existedLicenses	<loaded from database>
2	uniqueLicense	!@#%\$!@#%^@#\$\$@#!#@#\$

Test Scenario	Pass two bike licenses, if they differ then check whether or not
---------------	--

	new license already existed in database
Expected result	Return false for all cases