



Daffodil
International
University

Renter Management App

Submitted By:

Md. Mithu Hasan

ID: 163-35-1784

Department of Software Engineering

Submitted By:

Md. Anwar Hossen

Lecturer (Senior Scale)

Department of Software Engineering

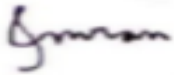
This Project report has been submitted in fulfillment of the requirements for
the Degree of Bachelor of Science in Software Engineering

©Daffodil International University

APPROVAL

This project titled on “**Renter Management app**”, submitted by **Md. Mithu Hasan (ID: 163-35-1784)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS



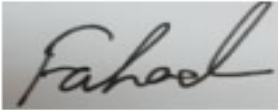
Chairman

Dr. Imran Mahmud
Associate Professor and Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University



Internal Examiner 1

K. M. Imtiaz-Ud-Din
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University



Internal Examiner 2

Md Fahad Bin Zamal
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University



External Examiner

Professor Dr. Md. Nasim Akhtar
Professor
Department of Computer Science and Engineering
Dhaka University of Engineering and Technology, Gazipur

DECLARATION

I hereby declare that project titled “Renter Management App” is an original record done by me under the supervision of Md. Anwar Hossen, Lecturer, Department of Software Engineering, Daffodil International University, towards the fulfillment of the requirement for the award of the degree of Bachelor of Science in Software Engineering during the period of 2016-2020. I also state that this project has not been submitted anywhere for any degree of this or any other University.



Md. Mithu Hasan

ID: 163-35-1784

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

Certified by:



Mr Md. Anwar Hossen

Lecturer Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

ACKNOWLEDGEMENT

I am thankful to Allah for giving me the opportunity to complete this project work and learn a lot. I am thanked to my supervisor, Mr Md. Anwar Hossen, for providing careful guidance starting from selecting the project scope to successfully finalizing the project work. Finally, I like to express my gratitude to Dr. Imran Mahmud, head of the Software Engineering faculty, for inspiring us in all means.

ABSTRACT

Renter Management app is a web-based application. The main outcome of the application is to maintain records of all users, billing, details of flat availability, and booking. Renter and another user without admin comment on their problems by select the problem category. Landlord's current financial position with the home. This will be an effective way to manage the home. Every landlord can use this system. This project will help them a lot.

Table of Contents

APPROVAL	ii
DECLARATION	iii
ACKNOWLEDGEMENT	iv
ABSTRACT	v
Chapter 1	1
Introduction.....	1
1.1 Purpose	1
1.2 Product Scope	1
1.3 Overview	1
1.4 References	2
1.5 Stockholder	2
1.6 Project Schedule	3
1.6.1 Grant Chart	3
2 Chapter	4
User Classes & Characteristics	4
3 Chapter	5
Design & Implementation Constraints	5
4 Chapter	6
Software Requirement Specification	6
4.1 Requirement Specification	6
4.1.1 Functional Requirement.....	6
4.1.2 Non-Functional Requirement.....	8
5 Chapter	10
System Analysis.....	10
5.1 Use Case Diagram	10
5.2 Use Case Description	16
5.2.1 Use case: Authentication	16
5.2.2 Use case: Account Status	17
5.2.3 Use case: Bill generation	18
5.2.4 Use case: Manage all user	19
5.2.5 Use case: Create renter information	20
5.2.6 Use case: Check users complain	21
5.2.7 Use case: Create proposal & agreement	22
5.2.8 Use case: Create flat information	23

5.2.9	Use case: Get flat booking request	24
5.2.10	Use case: Complain for any problem	25
5.2.11	Use case: View renter information	26
5.2.12	Use case: Send flat booking request to admin or manager.....	27
5.2.13	Use case: Complain for any problem	28
5.2.14	Use case: View security information	29
5.2.15	Use case: Complain for any problem	30
5.2.16	Use case: View mechanic information.....	31
5.2.17	Use case: Complain for any problem	32
5.2.18	Use case: View driver information.....	33
5.3	Activity Diagram.....	34
5.3.1	Activity diagram for Access to the server	34
5.3.2	Activity diagram for Authentication.....	35
5.3.3	Activity diagram for Account status	36
5.3.4	Activity diagram for Bill generation	37
5.3.5	Activity diagram for Manage all user	38
5.3.6	Activity diagram for Create renter info.....	39
5.3.7	Activity diagram for Check user complain.....	40
5.3.8	Activity diagram for Create proposal & agreements	41
5.3.9	Activity diagram for Get flat booking request from renter	42
5.3.10	Activity diagram for complain for any problem.....	43
5.3.11	Activity diagram for View user info.....	44
5.3.12	Activity diagram for Send flat booking request	45
5.4	Sequence Diagram.....	46
5.4.1	Sequence Diagram for Authentication.....	46
5.4.2	Sequence Diagram for Account status	47
5.4.3	Sequence Diagram for Bill generation	48
5.4.4	Sequence Diagram for Manage all employee.....	49
5.4.5	Sequence Diagram for Create Renter info.....	50
5.4.6	Sequence Diagram for Check user complain.....	51
5.4.7	Sequence Diagram for Create proposal & agreements	52
5.4.8	Sequence Diagram for Create booking info.....	53
5.4.9	Sequence Diagram for Check booking request.....	54
5.4.10	Sequence Diagram for Create user info.....	55
5.4.11	Sequence Diagram for Check user complain.....	56

5.4.12	Sequence Diagram for Create flat booking info	57
5.4.13	Sequence Diagram for Create billing info	58
5.4.14	Sequence Diagram for Complain for any problem.....	59
5.4.15	Sequence Diagram for View user info.....	60
5.4.16	Sequence Diagram for Send flat booking request	61
5.4.17	Sequence Diagram for Complain for any problem.....	62
6	Chapter	63
6.1	Entity Relationship Diagram	63
6.2	Class Diagram	64
6.3	7.3 Development of tools And Technology.....	64
6.3.1	User Interface Technology.....	64
6.3.2	Implementation Tools & Platforms.....	64
7	Chapter	65
	System Testing.....	65
7.1	Testing Features	65
7.1.1	Features to be tested	65
7.1.2	Features not to be tested.....	65
7.2	Testing Strategies	65
7.2.1	Test Approach	65
7.2.2	Pass/Fail Criteria.....	65
7.2.3	Suspension and Resumption.....	65
7.2.4	Testing Schedule.....	66
7.3	Test case.....	66
7.3.1	Test case 01	66
7.3.2	Test Case: 02.....	67
8	Chapter	69
	User Manual.....	69
8.1	Welcome page 01.....	69
8.2	Welcome page 02.....	69
8.3	Login page	70
8.4	Registration page.....	70
8.5	Admin dashboard page	71
8.6	Account status page.....	71
8.7	User complain page.....	72
8.8	User complain delete page.....	72

8.9	User complain Update page	73
8.10	Renter flat info page.....	73
8.11	Renter flat info delete page	74
8.12	Billing info page.....	74
8.13	Renter info page	75
8.14	Employee info page	75
8.15	Manager dashboard page	76
8.16	Renter dashboard page.....	76
8.17	Flat booking page.....	77
8.18	Booking request send page.....	77
8.19	Complain page.....	78
8.20	Security dashboard page.....	78
8.21	Mechanic dashboard page	79
8.22	Driver dashboard page.....	79
8.23	Logout page	80
9	Chapter	81
	Conclusion	81
10	Chapter	82
10.1	Limitations	82
10.2	Obstacles & Achievements.....	82
10.3	Future Scope.....	82

LIST OF FIGURES

Figure 1: Use case for Admin	10
Figure 2: Use case for Manager	11
Figure 3: Use case for Renter	12
Figure 4: Use case for Security	13
Figure 5: Use case for Mechanic	14
Figure 6: Use case for Driver	15
Figure 7: 6.3.1 Activity diagram for Access to the server	34
Figure 8: 6.3.2 Activity diagram for Authentication	35
Figure 9: 6.3.3 Activity diagram for Account status	36
Figure 10: 6.3.4 Activity diagram for Bill generation	37
Figure 11: 6.3.5 Activity diagram for Manage all user	38
Figure 12: 6.3.6 Activity diagram for Create renter info	39
Figure 13 : 6.3.7 Activity diagram for Check user complain	40
Figure 14 : 6.3.8 Activity diagram for Create proposal & agreements	41
Figure 15 : 6.3.9 Activity diagram for Get flat booking request from renter	42
Figure 16 : 6.3.10 Activity diagram for complain for any problem	43
Figure 17 : 6.3.11 Activity diagram for View user info	44
Figure 18 : 6.3.12 Activity diagram for Send flat booking request	45
Figure 19 : 6.4.1 Sequence Diagram for Authentication	46
Figure 20 : 6.4.2 Sequence Diagram for Account status	47
Figure 21: 6.4.3 Sequence Diagram for Bill generation	48
Figure 22 : 6.4.4 Sequence Diagram for Manage all employee	49
Figure 23 : 6.4.5 Sequence Diagram for Create Renter info	50
Figure 24 : 6.4.6 Sequence Diagram for Check user complain	51
Figure 25 : 6.4.7 Sequence Diagram for Create proposal & agreements	52
Figure 26 : 6.4.8 Sequence Diagram for Create booking info	53
Figure 27 : 6.4.9 Sequence Diagram for Check booking request	54
Figure 28 : 6.4.10 Sequence Diagram for Create user info	55
Figure 29 : 6.4.11 Sequence Diagram for Check user complain	56
Figure 30 : 6.4.12 Sequence Diagram for Create flat booking info	57
Figure 31 : 6.4.13 Sequence Diagram for Create billing info	58
Figure 32 : 6.4.14 Sequence Diagram for Complain for any problem	59
Figure 33 : 6.4.15 Sequence Diagram for View user info	60
Figure 34 : 6.4.16 Sequence Diagram for Send flat booking request	61
Figure 35 : 6.4.17 Sequence Diagram for Complain for any problem	62
Figure 36 : Entity Relationship Diagram	63
Figure 37 : Class Diagram	64
Figure 38 : 9.1 Welcome page 01	69
Figure 39 : 9.2 Welcome page 02	69
Figure 40 : 9.3 Login page	70
Figure 41 : 9.4 Registration page	70

Figure 42 : 9.5	Admin dashboard page.....	71
Figure 43 : 9.6	Admin dashboard page.....	71
Figure 44 : 9.7	Admin dashboard page.....	72
Figure 45 : 9.8	Admin dashboard page.....	72
Figure 46 : 9.9	User complain Update page.....	73
Figure 47 : 9.10	Renter flat info page	73
Figure 48 : 9.11	Renter flat info delete page.....	74
Figure 49 : 9.12	Billing info page	74
Figure 50 : 9.13	Renter info page.....	75
Figure 51 : 9.14	Employee info page	75
Figure 52 : 9.15	Manager dashboard page	76
Figure 53 : 9.16	Renter dashboard page.....	76
Figure 54 : 9.17	Flat booking page.....	77
Figure 55 : 9.18	Booking request send page	77
Figure 56 : 9.19	Complain page	78
Figure 57 : 9.20	Security dashboard page	78
Figure 58 : 9.21	Security dashboard page	79
Figure 59 : 9.22	Security dashboard page	79
Figure 60 : 9.23	Logout page	80

LIST OF TABLES

Table 1 : Grant Chart	3
Table 2 : 5.1.1 Functional Requirement (FR-01)	6
Table 3 : FR-02.....	6
Table 4 : FR-03.....	6
Table 5 : FR-04.....	6
Table 6 : RF-05.....	7
Table 7 : RF-06.....	7
Table 8 : FR-07.....	7
Table 9 : FR-08.....	7
Table 10 : FR-9.....	7
Table 11 : FR-10.....	8
Table 12 : FR-11.....	8
Table 13 : FR-12.....	8
Table 14 : Non-Functional Requirement (NFR-01)	8
Table 15 : NFR-02.....	8
Table 17 : NFR-03.....	9
Table 18 : UC-01 Success scenario	16
Table 19 : UC-01 Alternative flow.....	16
Table 20 : UC-01 Quality Requirements	17
Table 21 : UC-02 Success scenario	17
Table 22 : UC-03 Success scenario	18
Table 23 : UC-04 Success scenario	19
Table 24 : UC-05 Success scenario	20
Table 25 : UC-06 Success scenario	21
Table 26 : UC-07 Success scenario	22
Table 27 : UC-08 Success scenario	23
Table 28 : UC-09 Success scenario	24
Table 29 : UC-10 Success scenario	25
Table 30 : UC-11 Success scenario	26
Table 31 : UC-12 Success scenario	27
Table 32 : UC-13 Success scenario	28
Table 33 : UC-14 Success scenario	29
Table 34 : UC-15 Success scenario	30
Table 35 : UC-16 Success scenario	31
Table 36 : UC-17 Success scenario	32
Table 37 : UC-18 Success scenario	33
Table 38 : Testing Schedule (Programmer Form)	66
Table 39 : User login	66
Table 40 : User login	67
Table 41 : Check Eligibility	67
Table 42 : Check Eligibility	68

Chapter 1

Introduction

1.1 Purpose

The landlord easily manages his home by this system. This system will have six different panels. These are the admin panel, Manager Panel, Renter, Security guard, and Driver panel. Admin can only access in their panel. They can maintain an account, view account, bill generation, save the bill, add, view, edit, delete new tenant, security grad, driver and macaronic record, search renter, check flat booking info and renter complain and create proposal & agreements. The manager can only access in his panel. He not able to account management of this system but he can able to another task like admin.

1.2 Product Scope

The proposed software product is the Renter Manage App (RMP) for the Institute of Information Technology, Daffodil International University. Home automation also helps consumers improve efficiency, increase benefits and reduce time. The landlord easily manages his home by this system. This system will have six different panels. These are the Admin panel, Manager Panel, Renter, Security guard, and Driver panel. Admin can only access in their panel. They can maintain an account, view account, bill generation, create renter info, security grad, driver and macaronic record, check flat availability and renter complaints, create proposal and agreements.

1.3 Overview

This web application made for the landlords. In this application, admin can create renter and another employee's info. He also can maintain account status and create proposal and agreements. Bill generation for renter of each flat. Manager pay a vital role for this system. Renter check flat availability, view their info and complain to admin if any problem. In other hand all employee's view their info and complain to admin if any complain. This project was developed by PHP language (Laravel Framework) and MySQL database system for server site and HTML, CSS, JS for client site.

1.4 References

Reference by my course teacher

Reference by my class friend

<https://laravel.com/docs/5.8>

<https://stackoverflow.com/>

<https://getbootstrap.com/>

<https://www.w3schools.com/default.asp>

<https://www.youtube.com/channel/UCrmHQdRbYKFsB602Duho4Tw>

https://www.youtube.com/channel/UCSMFY8_rooijS-Zv43tKCrQ

1.5 Stockholder

There are six types of stockholders:

- ✓ Administrator
- ✓ Manager
- ✓ Renter
- ✓ Security
- ✓ Mechanic
- ✓ Driver

1.6 Project Schedule

1.6.1 Grant Chart

Task	Start Date	End Date	Status	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Proposal	08-05-2020	05-06-2020	Complete	Proposal							
Requirements	05-06-2020	08-07-2020	Complete		Requirements						
Design	08-07-2020	14-09-2020	Complete			Design	Design				
Imple	14-09-2020	16-10-2020	Complete					Impl		Imple	
Testing	16-10-2020	25-11-2020	Complete								Testing
Doc	25-11-2020	30-12-2020	Complete								Doc

Table 1 : Grant Chart

2 Chapter

User Classes & Characteristics

There are six types of users. They are student, admin and teacher. User characteristics are given below.

Admin

- ✓ To keep accounts of renter payment
- ✓ To know the current position of the account
- ✓ Bill generation
- ✓ Create proposal & agreements
- ✓ Create flat offer for renter
- ✓ Manage all user
- ✓ Check renter complain

Manager

- ✓ Manage all the whole user
- ✓ Bill generation
- ✓ Create flat offer for renter
- ✓ Manage all the complains of manager, renter, security guard and driver comment their complain in comment box by select problem category

Renter

- ✓ Access to view all records of renter, security guard, and driver.
- ✓ To check flat availability and booking.
- ✓ Complain for any complain.

Driver

- ✓ Access to view all user records.
- ✓ Complain for any complain.

Security

- ✓ Access to view all user records.
- ✓ Complain for any complain.

Mechanics

- ✓ Access to view all user records.
- ✓ Complain for any complain.

3 Chapter

Design & Implementation Constraints

Operating Environment

Operating Environment: The EHP will be web-based system. Thus, anyone having a browser can hit the specific link and can get access to it. Thus, it will ensure its best usage and will ease the means of getting access to the system. Moreover, it will remove the complexities of running the system in multiple platforms as it will be deployed in a web server.

Software Language Used

For front-end development

- ✓ HTML
- ✓ CSS
- ✓ Bootstrap
- ✓ JavaScript

For back-end development

- ✓ Laravel framework using PHP language

Database

- ✓ MySQL

Development Tools

- ✓ Microsoft Visual Studio code Professional edition will be used.
- ✓ Database operations MySQL will be used.

Database Support

The database that will be used is MySQL. Entity framework will be used from the applications end to insert, update and delete the data.

4 Chapter

Software Requirement Specification

4.1 Requirement Specification

4.1.1 Functional Requirement

FR-01	Authentication
Description	User will be able to login and logout to the system with valid user credentials.
Stakeholders	Admin, Manager, Renter, Security, Mechanic and driver.
Priority	High

Table 2 : 5.1.1 Functional Requirement (FR-01)

FR-02	Account Status
Description	Admin will be able to create account information.
Stakeholders	Admin
Priority	High

Table 3 : FR-02

FR-03	Bill generation
Description	Admin and manager will be able to generate renter and employee bill.
Stakeholders	Admin, Manager
Priority	High

Table 4 : FR-03

FR-04	Manage all user
Description	Admin and manager will be able to manage all user.
Stakeholders	Admin, Manager
Priority	Medium

Table 5 : FR-04

FR-05	Create renter information
Description	Admin, Manager will be able to create renter information.
Stakeholders	Admin, Manager
Priority	Medium

Table 6 : RF-05

FR-06	Check renter or other complain
Description	Admin, Manager will be able check renter or another user complain.
Stakeholders	Admin, Manager, Renter, Security, Mechanic and driver.
Priority	High

Table 7 : RF-06

FR-07	Create proposal and agreements
Description	Admin will be able to create proposal and agreements.
Stakeholders	Admin and Renter
Priority	High

Table 8 : FR-07

FR-08	Create flat information
Description	Admin and Manager will be able to create flat information.
Stakeholders	Admin, Manager and Renter
Priority	High

Table 9 : FR-08

FR-09	Get flat booking request
Description	Admin and Manager will be able to get booking request.
Stakeholders	Admin, Manager and Renter
Priority	High

Table 10 : FR-9

FR-10	Complain for any problem
Description	Renter or another employee will be able to complain for any problem.
Stakeholders	Renter and Another employee
Priority	High

Table 11 : FR-10

FR-11	View user information
Description	All user will be able to view user information.
Stakeholders	Admin, Manager, Renter, Security, Mechanic and driver.
Priority	Medium

Table 12 : FR-11

FR-12	Send flat booking request
Description	Renter will be able to send flat booking request to admin or manager
Stakeholders	Renter, Admin and Manager
Priority	High

Table 13 : FR-12

4.1.2 Non-Functional Requirement

NFR-01	Security
Description	Using token-based authentication, Validation it will be secure from unauthorized access.
Stakeholders	Renter, Admin, Manager, Security, Mechanic and Driver.
Priority	High

Table 14 : Non-Functional Requirement (NFR-01)

NFR-02	Availability
Description	The system should work 24/7 as user can get access and service
Stakeholders	Renter, Admin, Manager, Security, Mechanic and Driver.
Priority	High

Table 15 : NFR-02

NFR-04	Maintainability
Description	It's a way how easy to support, change and enhance the system.
Stakeholders	Renter, Admin, Manager, Security, Mechanic and Driver.
Priority	High

Table 16 : NFR-03

5 Chapter

System Analysis

5.1 Use Case Diagram

Admin

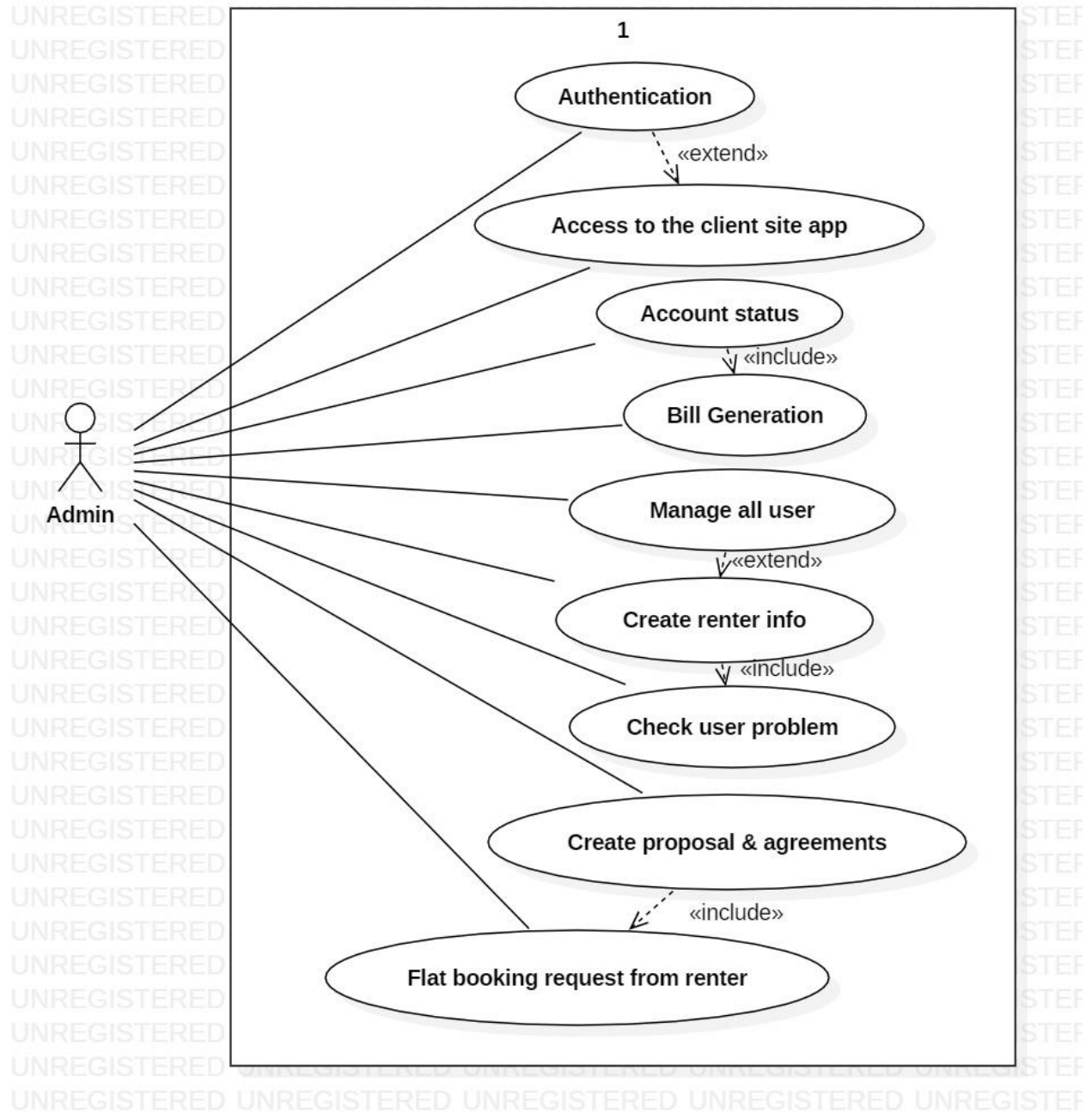


Figure 1: Use case for Admin

Manager

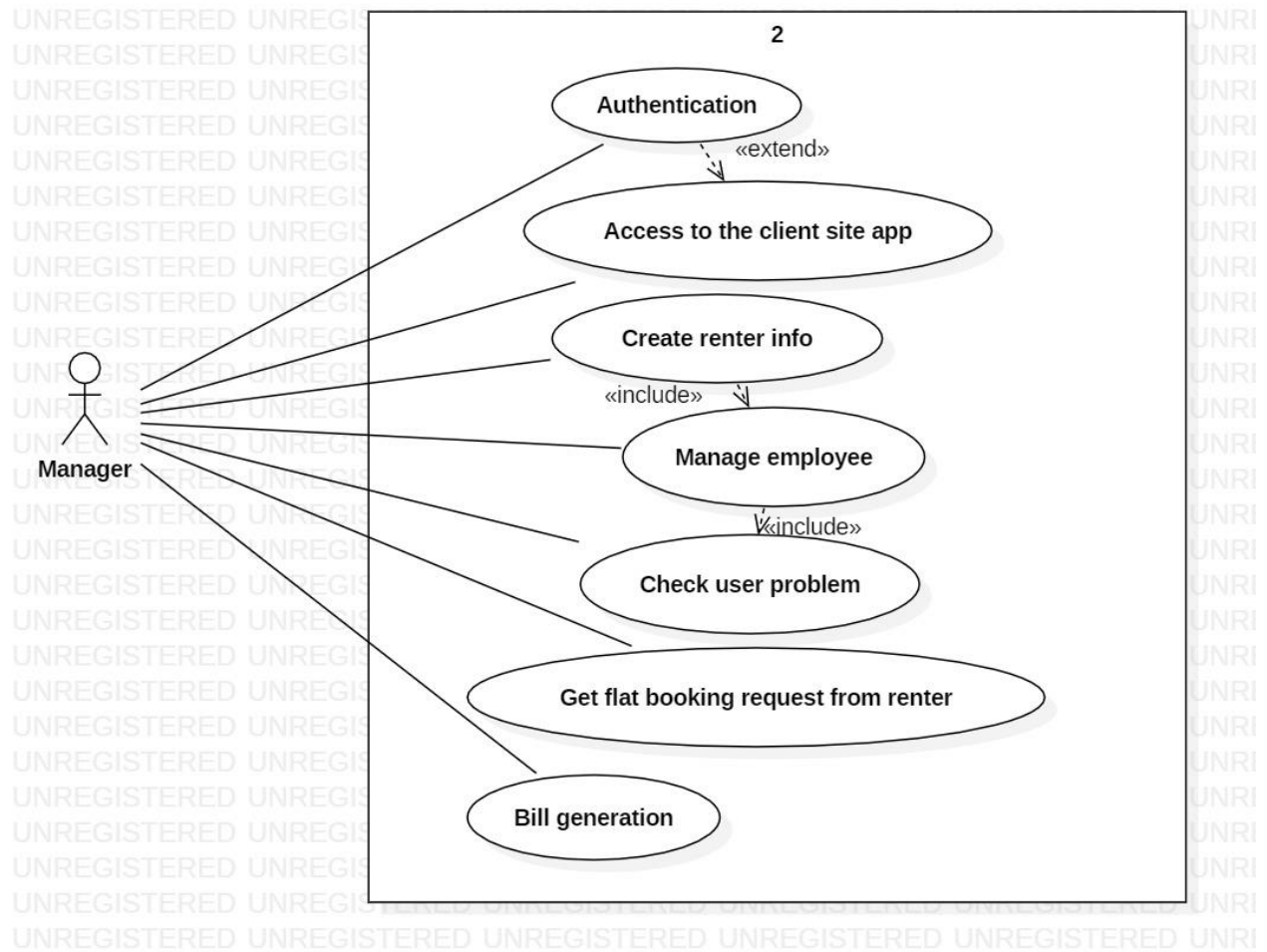


Figure 2: Use case for Manager

Renter

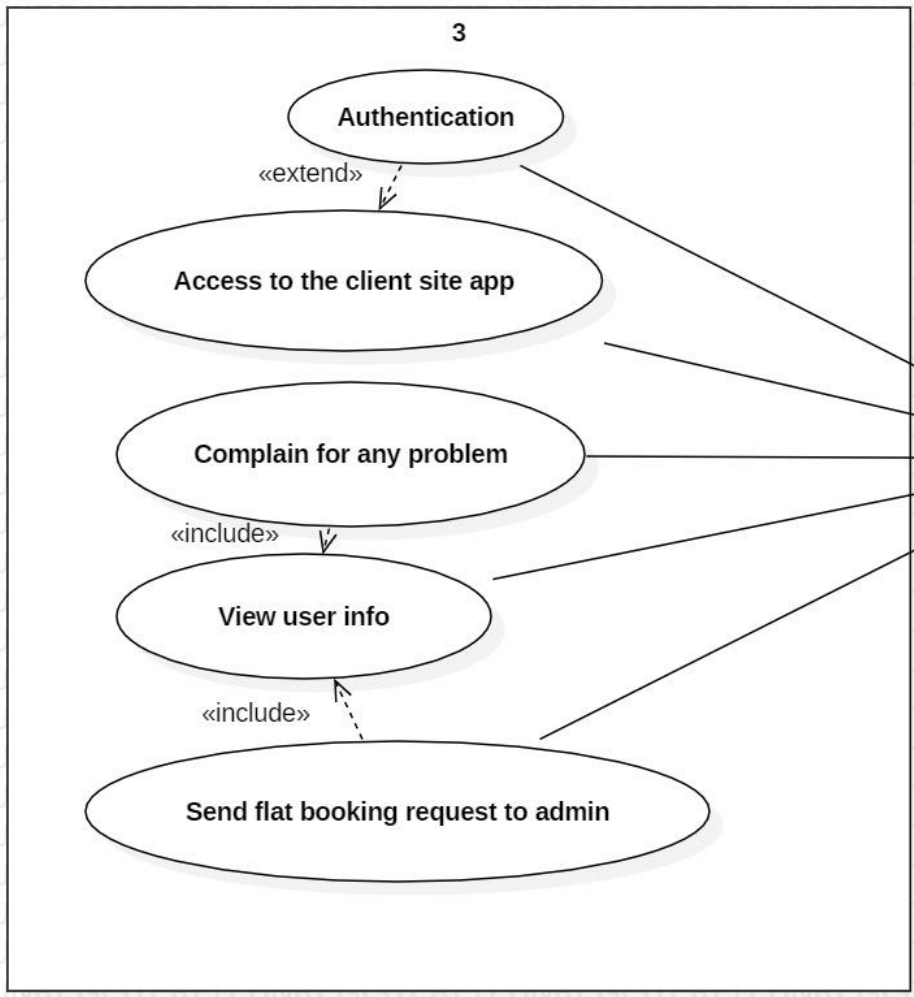


Figure 3: Use case for Renter

Security Guard

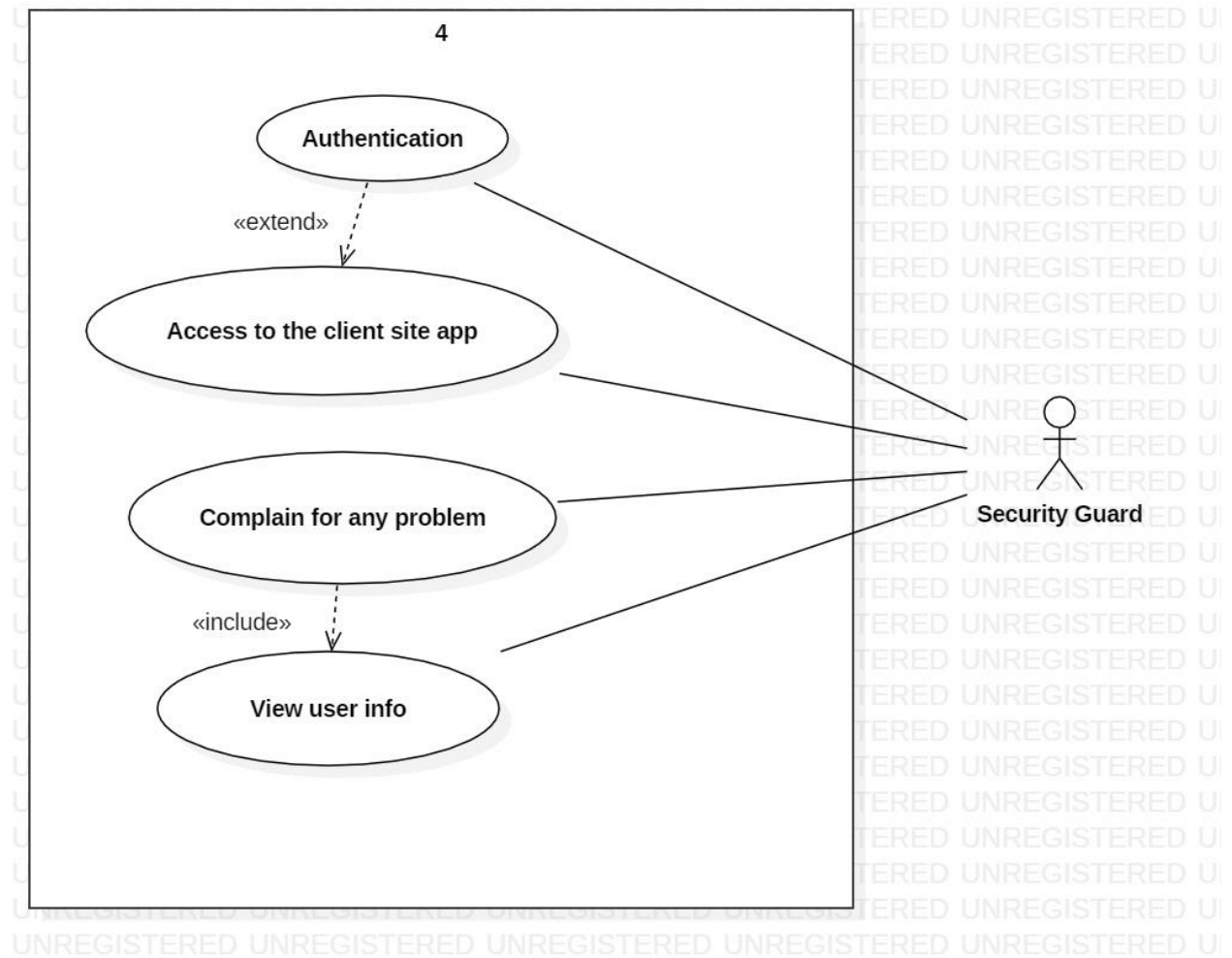


Figure 4: Use case for Security

Mechanic

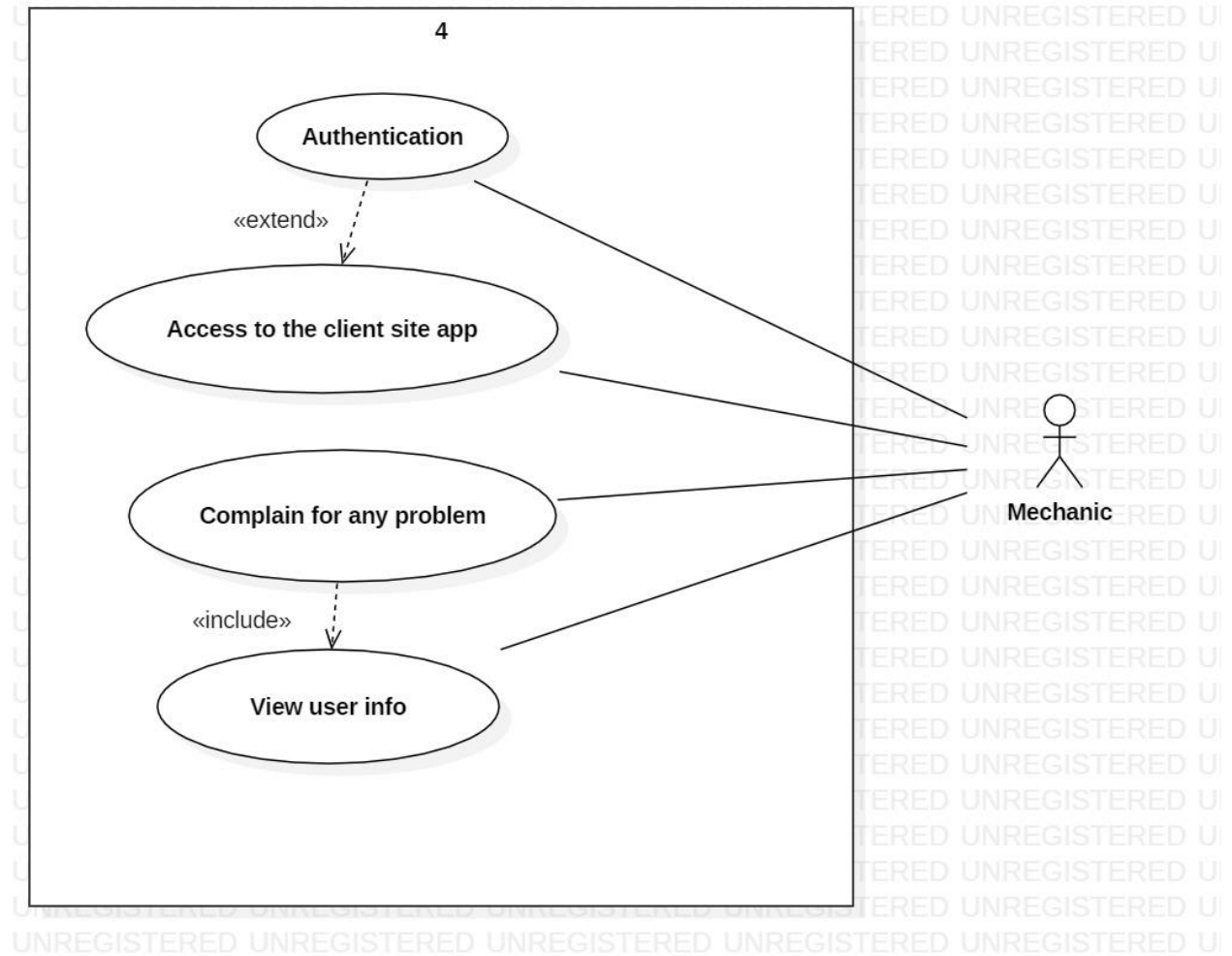


Figure 5: Use case for Mechanic

Driver

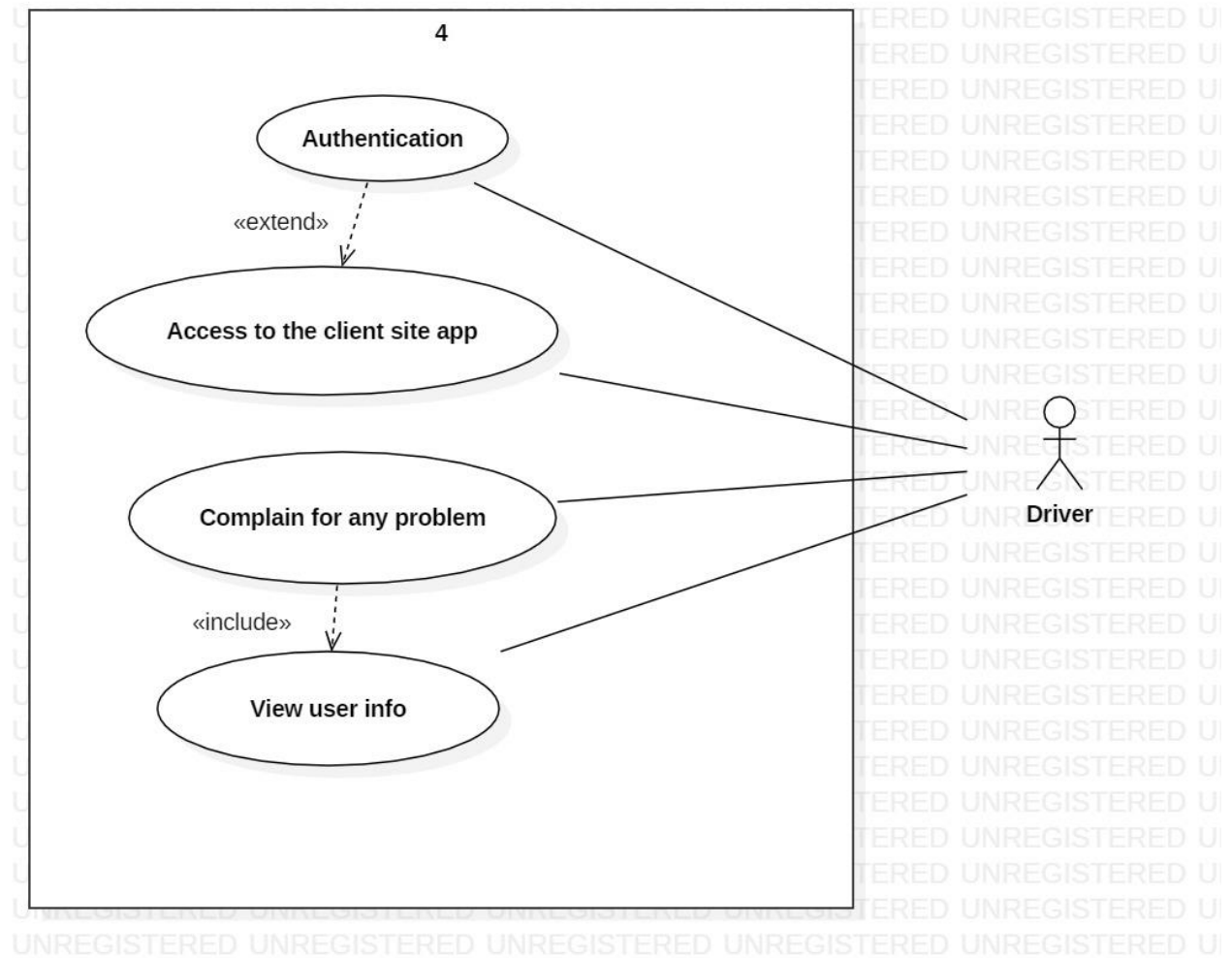


Figure 6: Use case for Driver

5.2 Use Case Description

5.2.1 Use case: Authentication

2. Goal:

Unregistered users register in the system. Registered user login into the system.

3. Pre-condition:

For authentication, at first user should submit the mail, which is store in user database. Admin and Manager have no pre-condition.

4. Post / end-condition:

- ✓ Success end: Successfully login or registration.
- ✓ Failure end: Login or registration failed.

5. Actors:

- ✓ Primary: Renters, Security, Mechanic and Driver.
- ✓ Secondary: Admin and Manager.

6. Trigger: Click on Login button

Success scenario:

Step	Action
01	User input user email
02	System verify this email with respect to user information.
03	User successfully logs into the system.

Table 17 : UC-01 Success scenario

Alternative flow:

Step	Branching action
1a	User is not a registered user
1a1	User give personal email for renter app
1b	User forget his/her password.
1b1	User recover his/her password by using email verification.

Table 18 : UC-01 Alternative flow

Quality Requirements:

Step	Requirements
1	Users should user at least 6-digit password with 2 special characters.
1a	Users can try 3 times wrong password. After that, the account will lock.

Table 19 : UC-01 Quality Requirements

5.2.2 Use case: Account Status

2. Goal:

Admin can add, delete and view the account information. Admin also can upload account information files.

3. Pre-condition:

Admin should registration for create account information.

4. Post / end-condition:

- ✓ Success end: Add, delete and update account information.
- ✓ Failure end: File cannot be upload

5. Actors:

- ✓ Primary: Admin
- ✓ Secondary: Renters, Security, Mechanic and Driver.

6. Trigger:

Click on add account information.

Success scenario:

Step	Action
01	Select update student information
02	System will insert, update and delete the account information

Table 20 : UC-02 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.3 Use case: Bill generation

2. Goal:

Admin and manager can add, delete and view the billing information. Admin also can upload billing information files.

3. Pre-condition:

Admin and manager should registration for create billing information.

4. Post / end-condition:

- ✓ Success end: Add, delete and update billing information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Admin and manager
- ✓ Secondary: Renters, Security, Mechanic and Driver.

6. Trigger:

Click on create billing information.

Success scenario:

Step	Action
01	Select update billing information
02	System will insert, update and delete the billing information

Table 21 : UC-03 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.4 Use case: Manage all user

2. Goal:

Admin and manager can add, delete and view the all-users information. Admin also can upload user's information files.

3. Pre-condition:

Admin and manager should registration for create user's information.

4. Post / end-condition:

- ✓ Success end: Add, delete and update user's information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Admin and manager
- ✓ Secondary: Renters, Security, Mechanic and Driver.

6. Trigger:

Click on create user's information.

Success scenario:

Step	Action
01	Select update user's information
02	System will insert, update and delete the user's information

Table 22 : UC-04 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.5 Use case: Create renter information

2. Goal:

Admin and manager can add, delete and view the renter's information. Admin also can upload renter's information files.

3. Pre-condition:

Admin and manager should registration for create renter's information.

4. Post / end-condition:

- ✓ Success end: Add, delete and update renter's information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Admin and manager
- ✓ Secondary: Renters, Security, Mechanic and Driver.

6. Trigger:

Click on create renter's information.

Success scenario:

Step	Action
01	Select update renter's information
02	System will insert, update and delete the renter's information.

Table 23 : UC-05 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.6 Use case: Check users complain

2. Goal:

Admin and manager can solve, delete and view the user's problem information. Admin also can upload solve user's complain information files.

3. Pre-condition:

Admin and manager should registration for solve user's complain information.

4. Post / end-condition:

- ✓ Success end: Add, delete and update user's solve information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Admin and manager
- ✓ Secondary: Renters, Security, Mechanic and Driver.

6. Trigger:

Click on solve users complain information.

Success scenario:

Step	Action
01	Select update user's complain information
02	System will delete the user's complain information.

Table 24 : UC-06 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.7 Use case: Create proposal & agreement

2. Goal:

Admin and manager can add, delete and view the proposal and agreements information.

3. Pre-condition:

Admin and manager should registration for create proposal and agreements information.

4. Post / end-condition:

- ✓ Success end: Add, delete and update create proposal and agreements information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Admin
- ✓ Secondary: Renters

6. Trigger:

Click on create proposal and agreements information.

Success scenario:

Step	Action
01	Select update create proposal and agreements information.
02	System will insert, update and delete the proposal and agreements information.

Table 25 : UC-07 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.8 Use case: Create flat information

2. Goal:

Admin and manager can add, delete and view the create flat information.

3. Pre-condition:

Admin and manager should registration for create flat information.

4. Post / end-condition:

- ✓ Success end: Add, delete and update flat information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Admin and manager
- ✓ Secondary: Renters

6. Trigger:

Click on create flat information.

Success scenario:

Step	Action
01	Select update flat information
02	System will insert, update and delete the flat information.

Table 26 : UC-08 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.9 Use case: Get flat booking request

2. Goal:

Admin and manager can accept and view the flat booking request from renters.

3. Pre-condition:

Admin and manager should registration for accept flat booking request.

4. Post / end-condition:

- ✓ Success end: Accept flat booking request.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Admin and manager
- ✓ Secondary: Renters

6. Trigger:

Click on view flat booking request.

Success scenario:

Step	Action
01	Select update flat booking request
02	System will view the flat booking request.

Table 27 : UC-09 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.10 Use case: Complain for any problem

2. Goal:

Renter can add, delete and view the create complain information for any user's problems.

3. Pre-condition:

Renter should registration for create complain information for any problems.

4. Post / end-condition:

- ✓ Success end: Add, delete and update complain information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Renter
- ✓ Secondary: Amin and manager

6. Trigger:

Click on create complain information.

Success scenario:

Step	Action
01	Select update complain information
02	System will insert, update and delete information.

Table 28 : UC-10 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.11 Use case: View renter information

2. Goal:

Renter can add, delete and view the view information.

3. Pre-condition:

Renter should registration for view user information.

4. Post / end-condition:

- ✓ Success end: View users' information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Renter
- ✓ Secondary: Admin and manager

6. Trigger:

Click on view user information.

Success scenario:

Step	Action
01	Select view user's information
02	System will view user's information.

Table 29 : UC-11 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.12 Use case: Send flat booking request to admin or manager

2. Goal:

Renter can add (send) and view the flat booking request information.

3. Pre-condition:

Renter should registration for send flat booking request information.

4. Post / end-condition:

- ✓ Success end: Add, delete and update flat booking request.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Renter
- ✓ Secondary: Admin and manager

6. Trigger:

Click on send flat booking request.

Success scenario:

Step	Action
01	Select update flat booking request.
02	System will insert, delete the flat request.

Table 30 : UC-12 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.13 Use case: Complain for any problem

2. Goal:

Security can add, delete and view the create complain information for any problems.

3. Pre-condition:

Security should registration for create complain information for any problems.

4. Post / end-condition:

- ✓ Success end: Add, delete and update complain information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Security
- ✓ Secondary: Amin and manager

6. Trigger:

Click on create complain information.

Success scenario:

Step	Action
01	Select update complain information
02	System will insert, update and delete information.

Table 31 : UC-13 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.14 Use case: View security information

2. Goal:

Security can add, delete and view the view information.

3. Pre-condition:

Security should registration for view user information.

4. Post / end-condition:

- ✓ Success end: View users' information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Security
- ✓ Secondary: Admin and manager

6. Trigger:

Click on view user information.

Success scenario:

Step	Action
01	Select view user's information
02	System will view user's information.

Table 32 : UC-14 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.15 Use case: Complain for any problem

2. Goal:

Mechanic can add, delete and view the create complain information for any problems.

3. Pre-condition:

Mechanic should registration for create complain information for any problems.

4. Post / end-condition:

- ✓ Success end: Add, delete and update complain information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Mechanic
- ✓ Secondary: Amin and manager

6. Trigger:

Click on create complain information.

Success scenario:

Step	Action
01	Select update complain information
02	System will insert, update and delete information.

Table 33 : UC-15 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.16 Use case: View mechanic information

2. Goal:

Mechanic can add, delete and view the view information.

3. Pre-condition:

Mechanic should registration for view user information.

4. Post / end-condition:

- ✓ Success end: View users' information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Mechanic
- ✓ Secondary: Admin and manager

6. Trigger:

Click on view user information.

Success scenario:

Step	Action
01	Select view user's information
02	System will view user's information.

Table 34 : UC-16 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.17 Use case: Complain for any problem

2. Goal:

Driver can add, delete and view the create complain information for any problems.

3. Pre-condition:

Driver should registration for create complain information for any problems.

4. Post / end-condition:

- ✓ Success end: Add, delete and update complain information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Driver
- ✓ Secondary: Amin and manager

6. Trigger:

Click on create complain information.

Success scenario:

Step	Action
01	Select update complain information
02	System will insert, update and delete information.

Table 35 : UC-17 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.2.18 Use case: View driver information

2. Goal:

Driver can add, delete and view the view information.

3. Pre-condition:

Driver should registration for view user information.

4. Post / end-condition:

- ✓ Success end: View users' information.
- ✓ Failure end: N/A

5. Actors:

- ✓ Primary: Driver
- ✓ Secondary: Admin and manager

6. Trigger:

Click on view user information.

Success scenario:

Step	Action
01	Select view user's information
02	System will view user's information.

Table 36 : UC-18 Success scenario

Alternative flow:

N/A

Quality Requirements:

N/A

5.3 Activity Diagram

5.3.1 Activity diagram for Access to the server

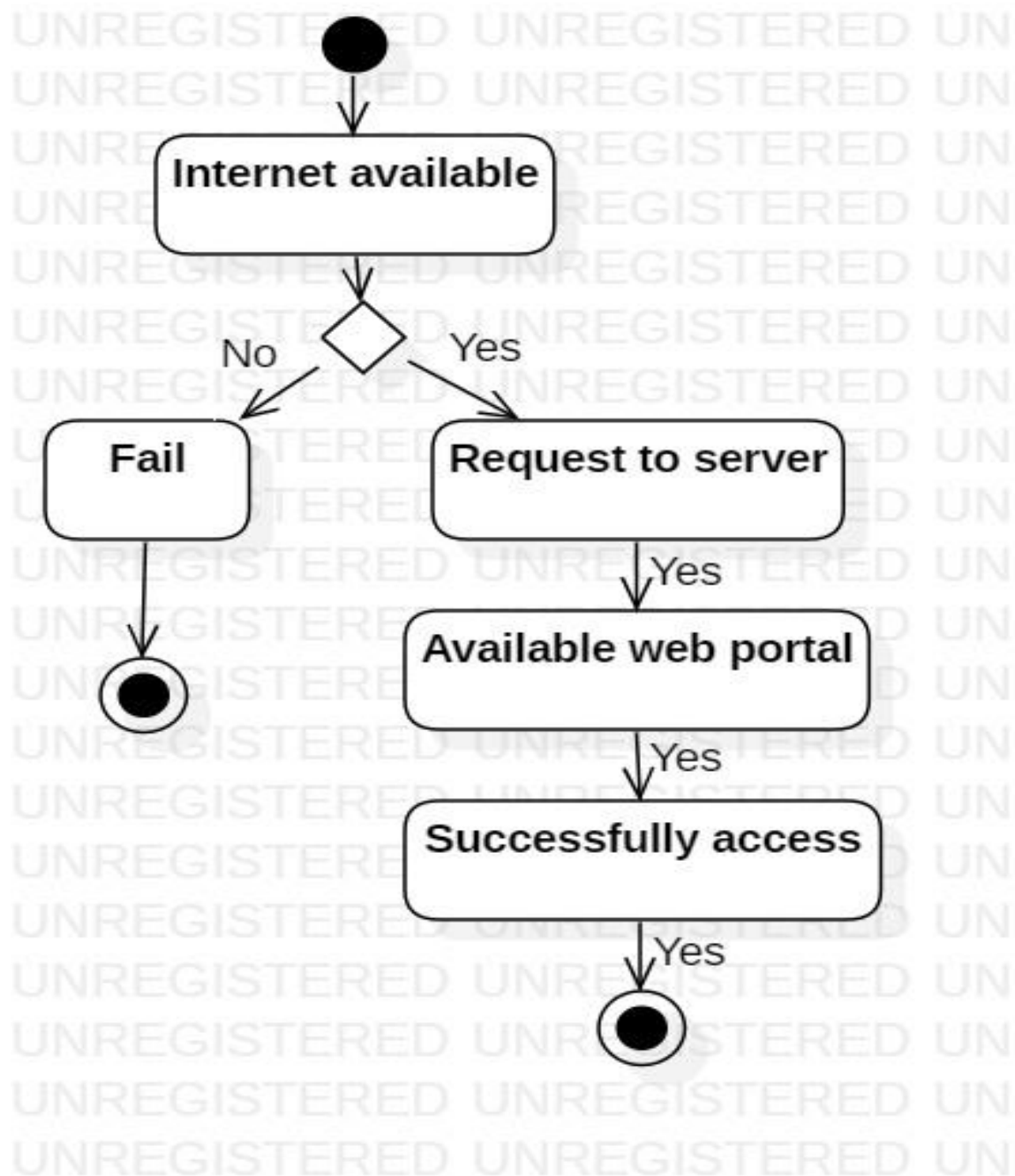


Figure 7: 6.3.1 Activity diagram for Access to the server

5.3.2 Activity diagram for Authentication

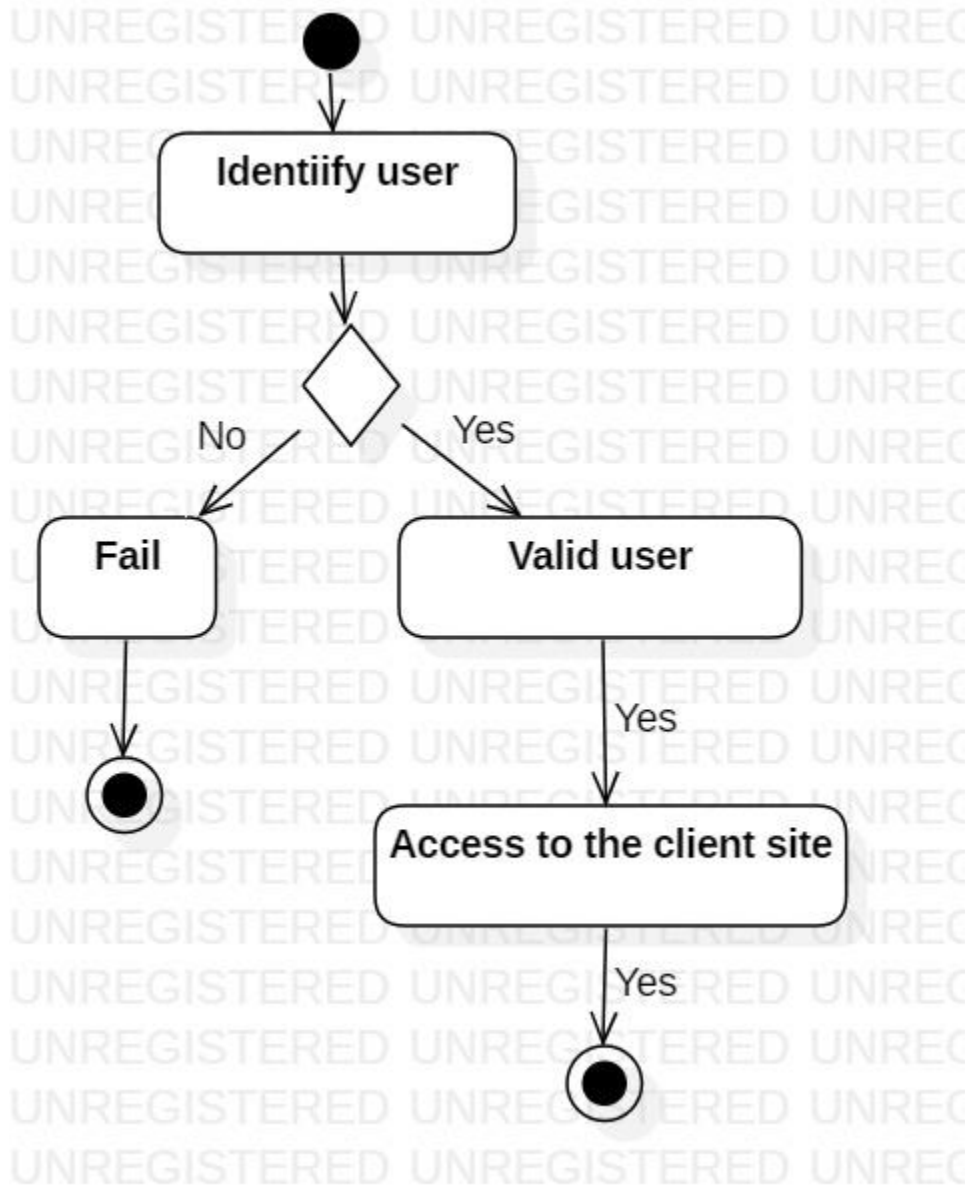


Figure 8: 6.3.2 Activity diagram for Authentication

5.3.3 Activity diagram for Account status

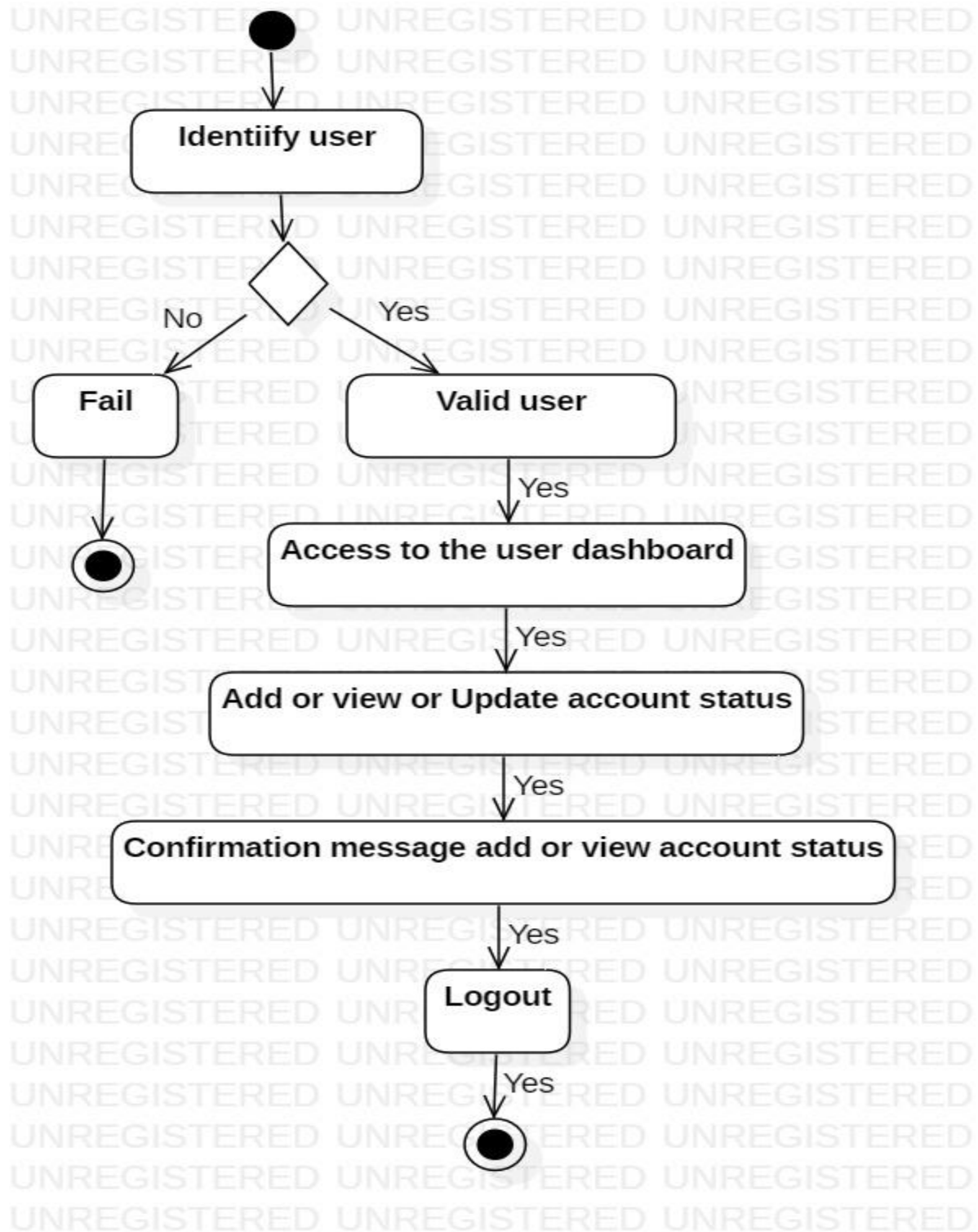


Figure 9: 6.3.3 Activity diagram for Account status

5.3.4 Activity diagram for Bill generation

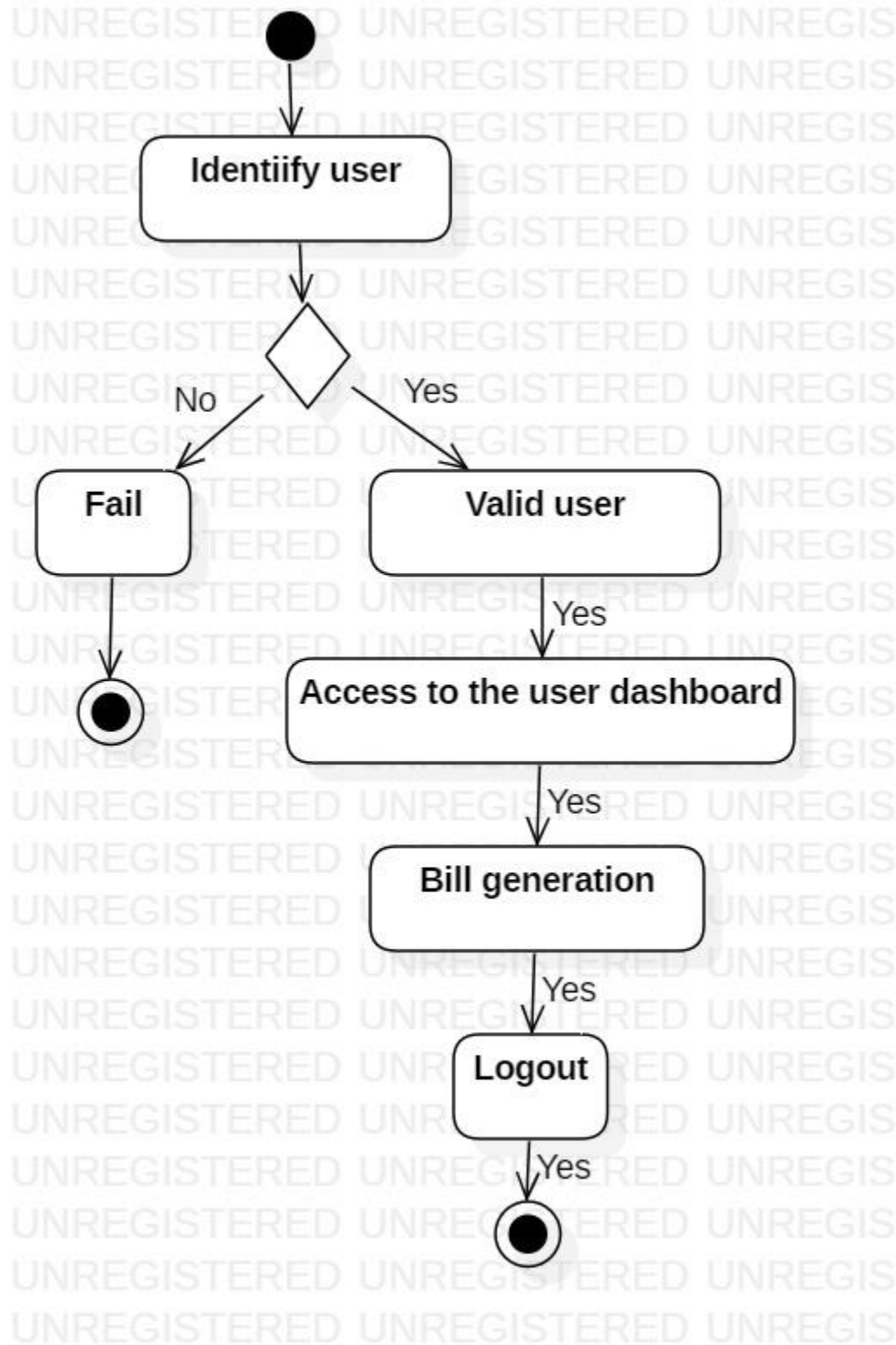


Figure 10: 6.3.4 Activity diagram for Bill generation

5.3.5 Activity diagram for Manage all user

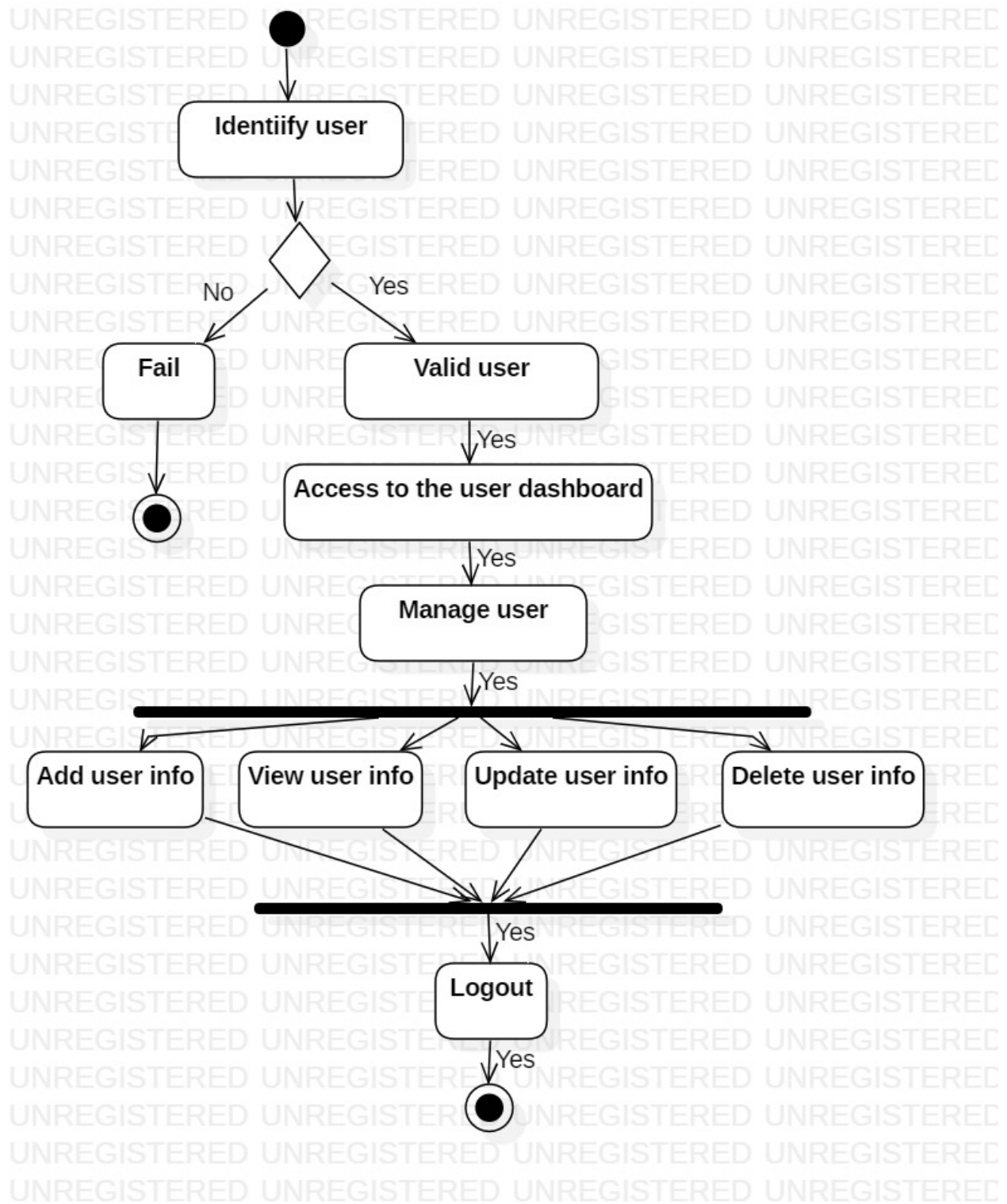


Figure 11: 6.3.5 Activity diagram for Manage all user

5.3.6 Activity diagram for Create renter info

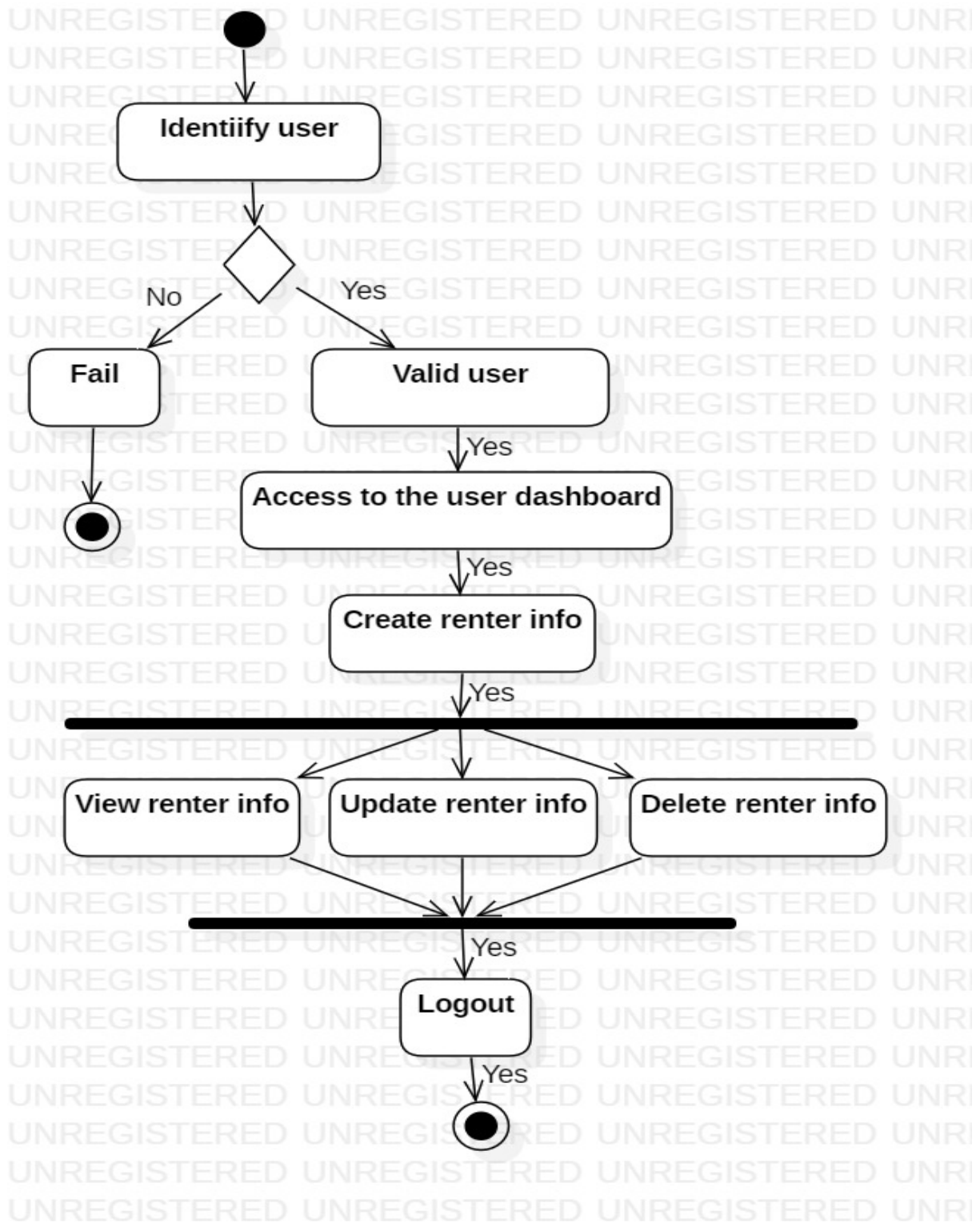


Figure 12: 6.3.6 Activity diagram for Create renter info

5.3.7 Activity diagram for Check user complain

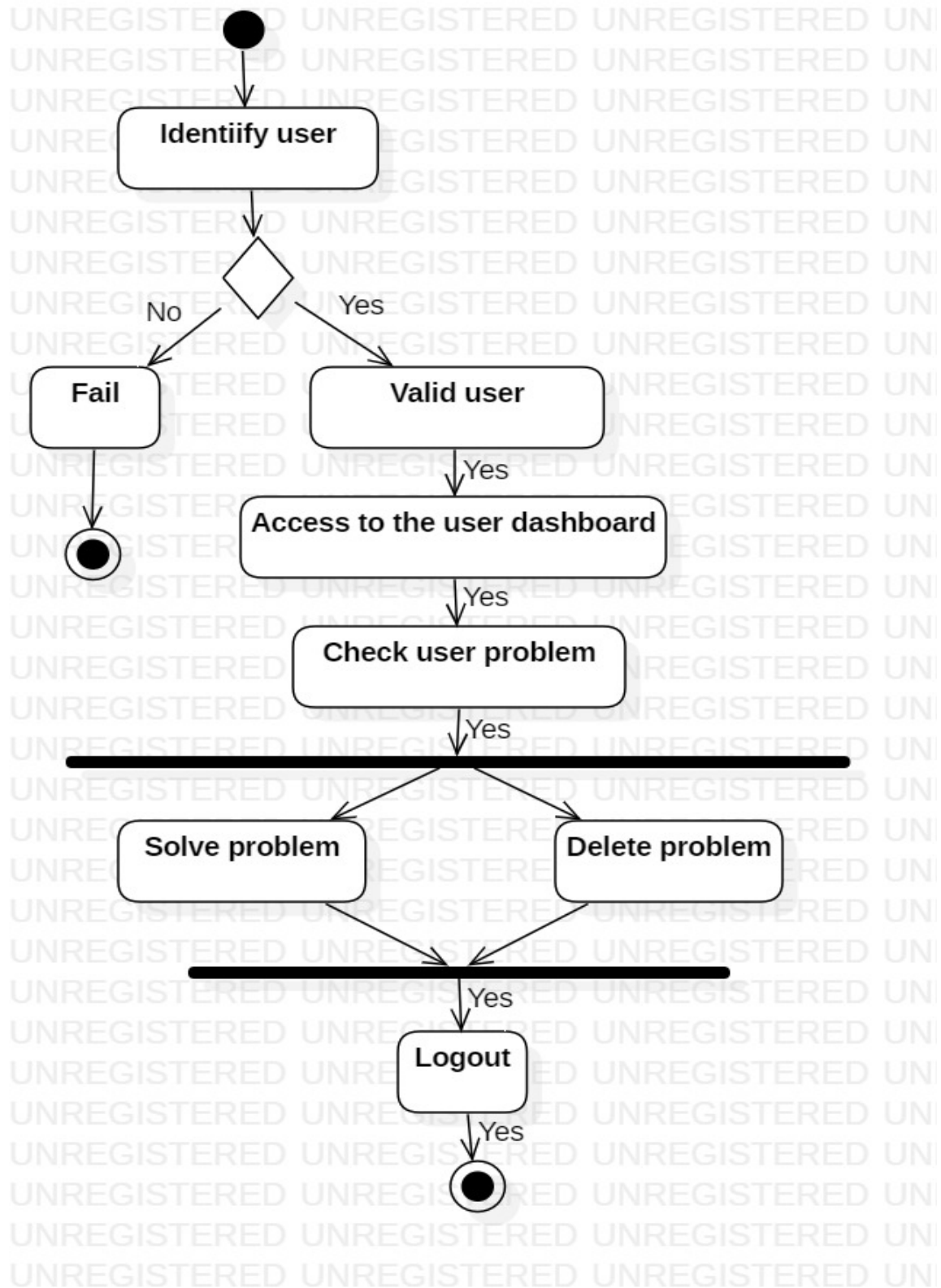


Figure 13 : 6.3.7 Activity diagram for Check user complain

5.3.8 Activity diagram for Create proposal & agreements

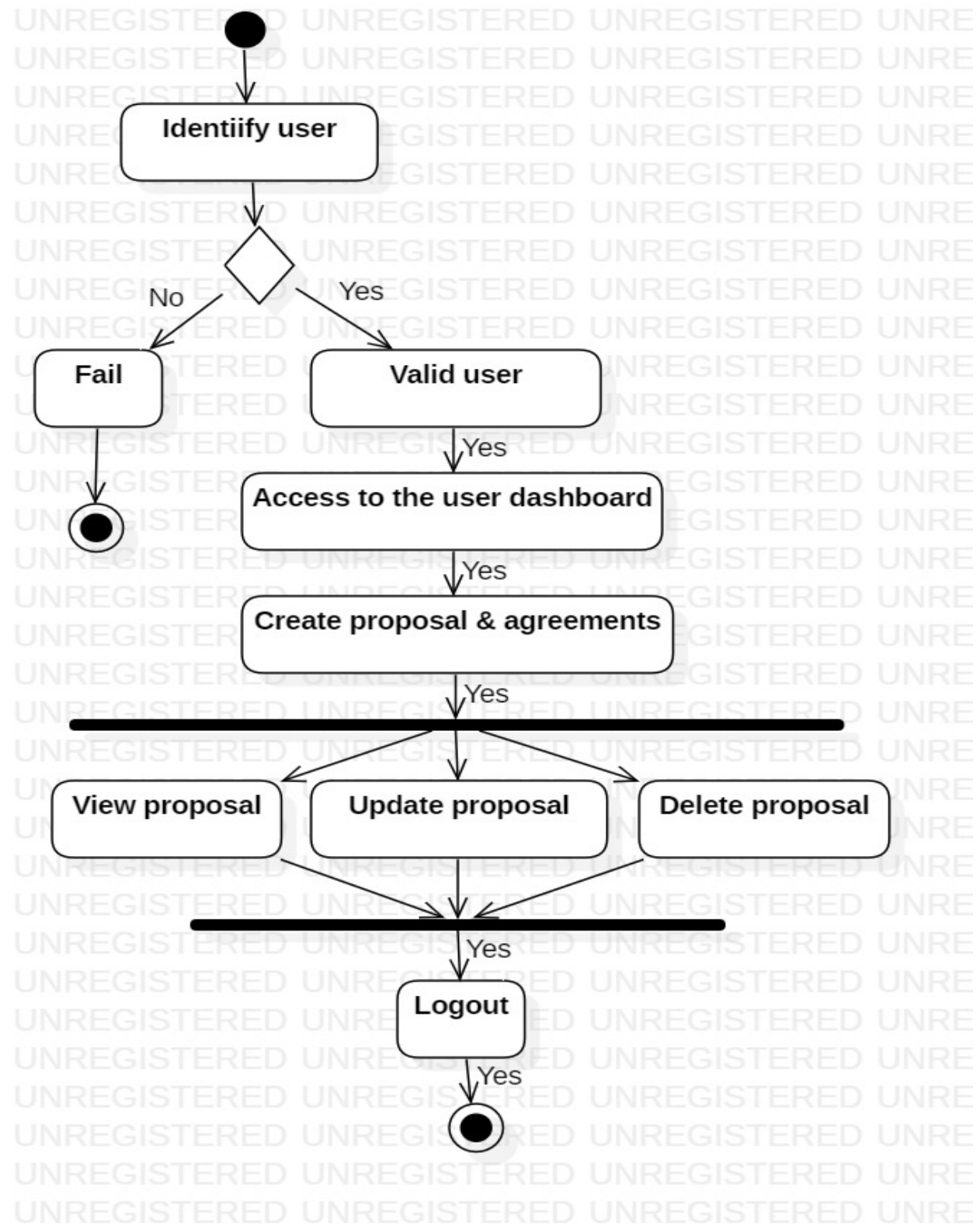


Figure 14 : 6.3.8 Activity diagram for Create proposal & agreements

5.3.9 Activity diagram for Get flat booking request from renter

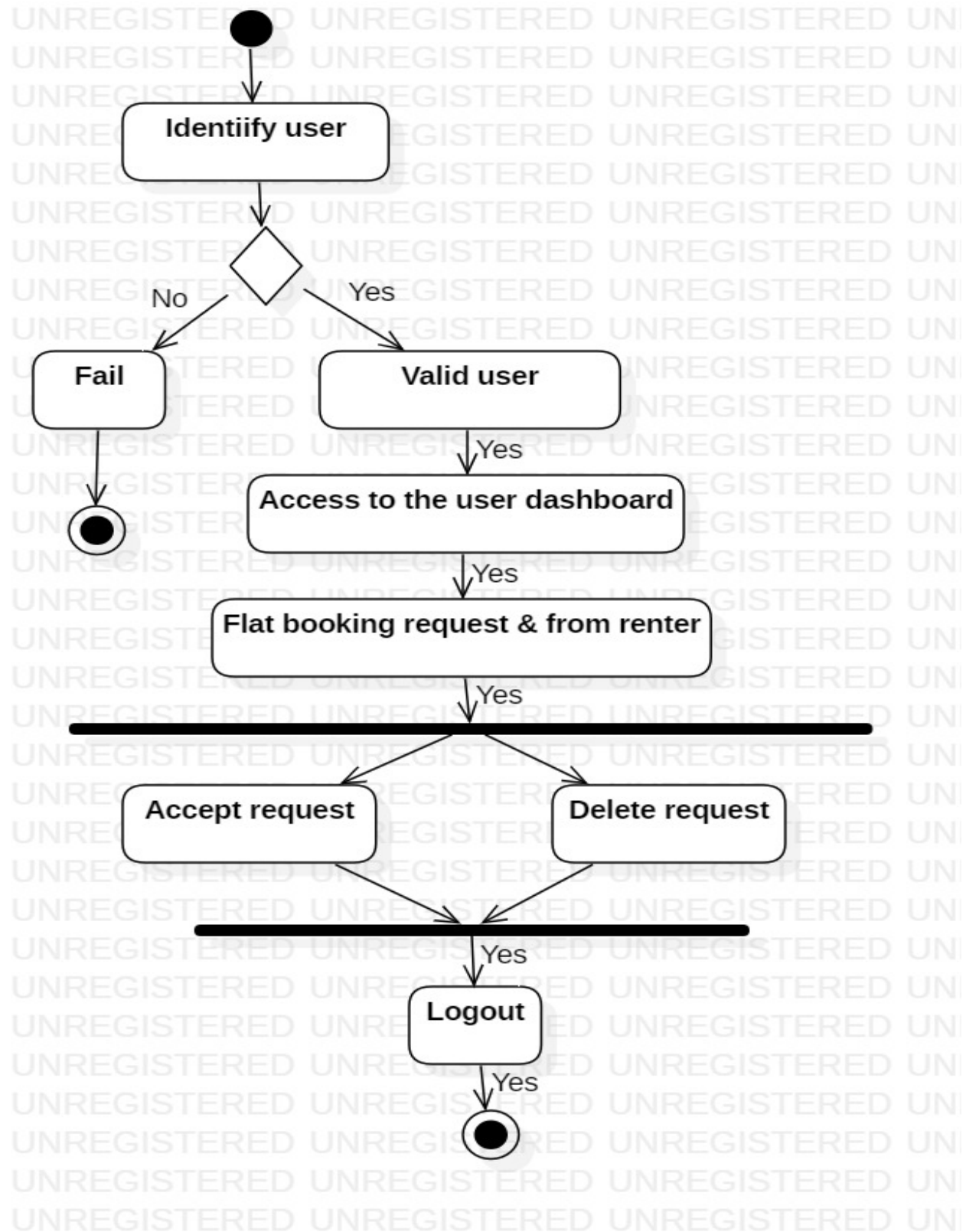


Figure 15 : 6.3.9 Activity diagram for Get flat booking request from renter

5.3.10 Activity diagram for complain for any problem

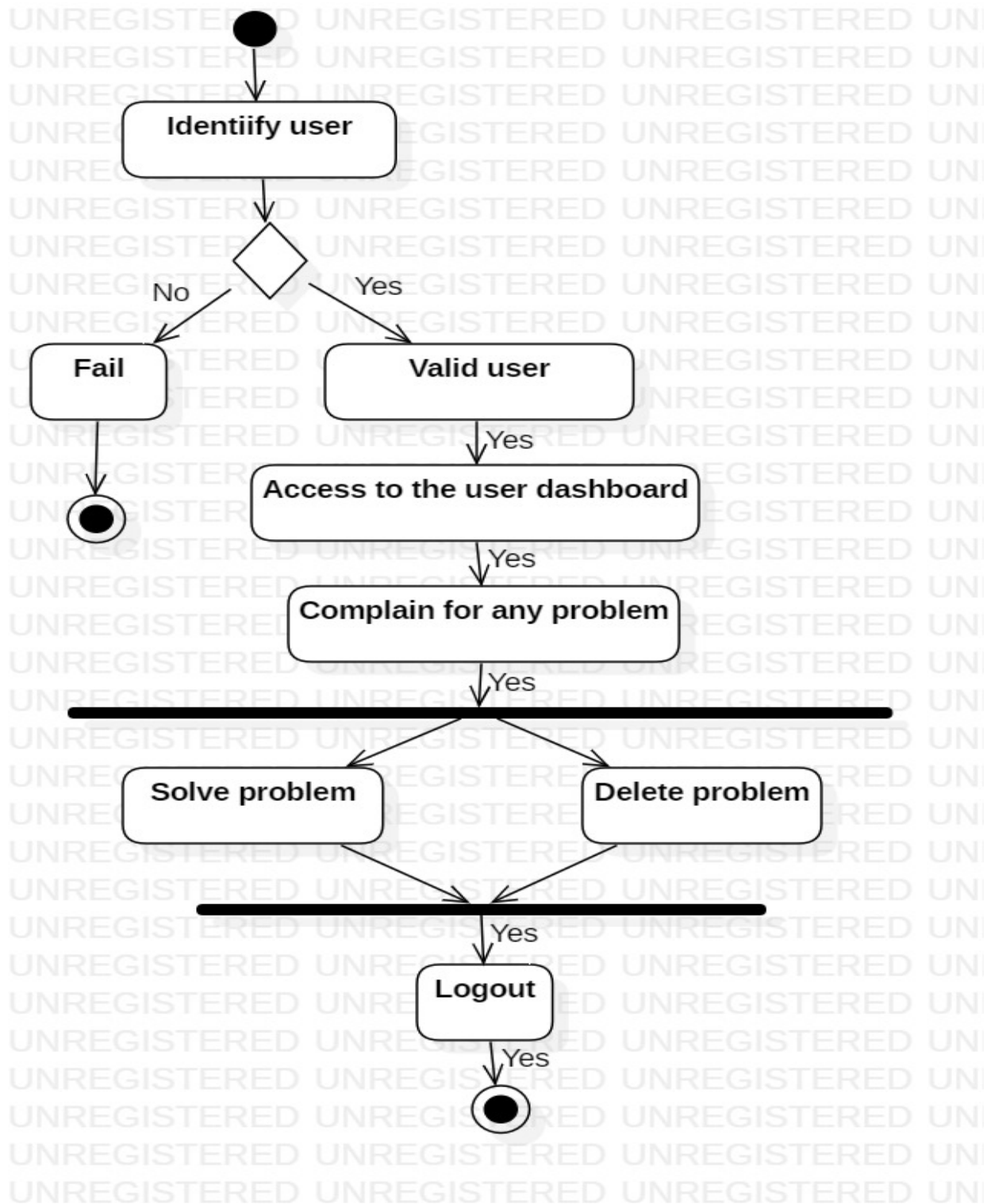


Figure 16 : 6.3.10 Activity diagram for complain for any problem

5.3.11 Activity diagram for View user info

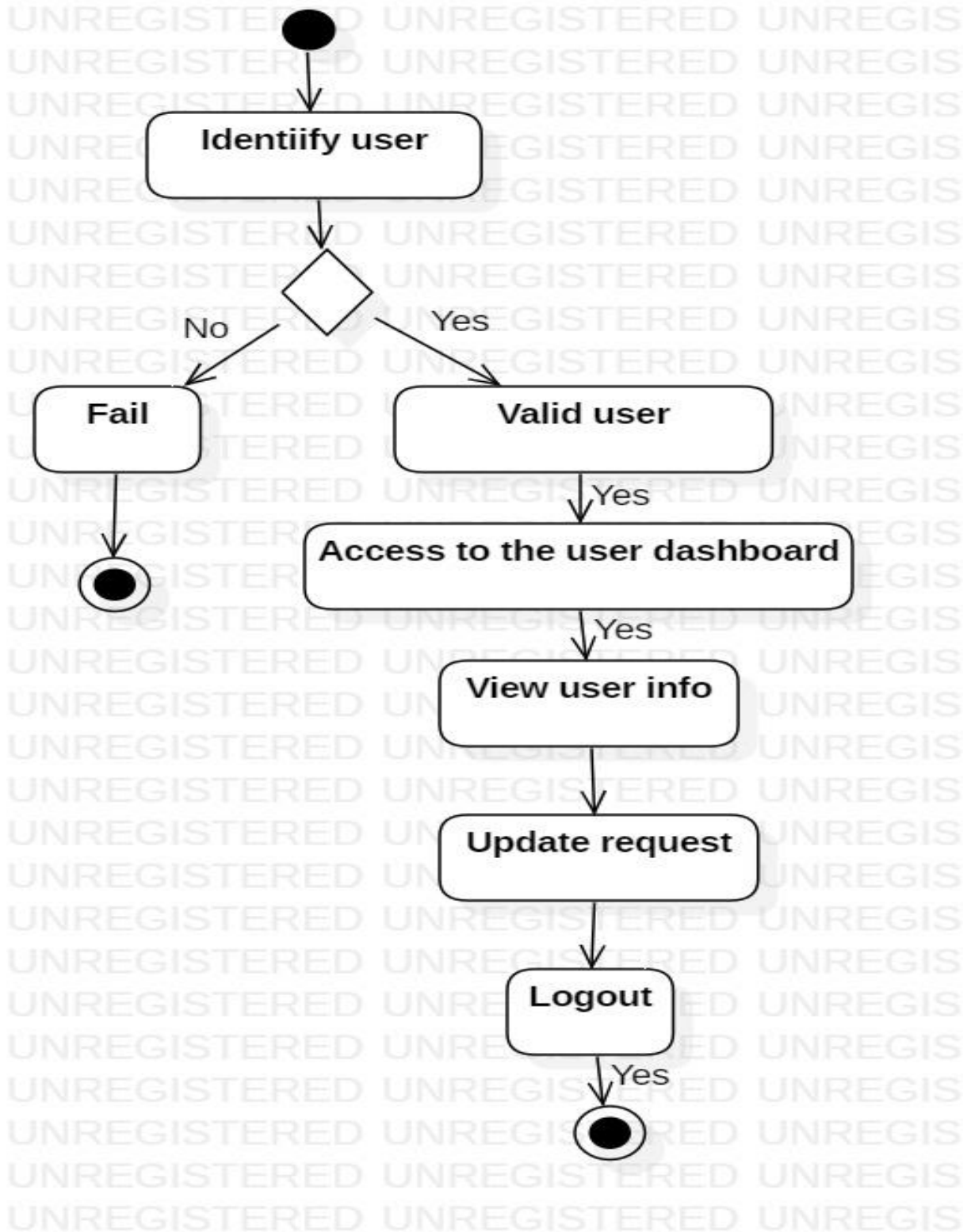


Figure 17 : 6.3.11 Activity diagram for View user info

5.3.12 Activity diagram for Send flat booking request

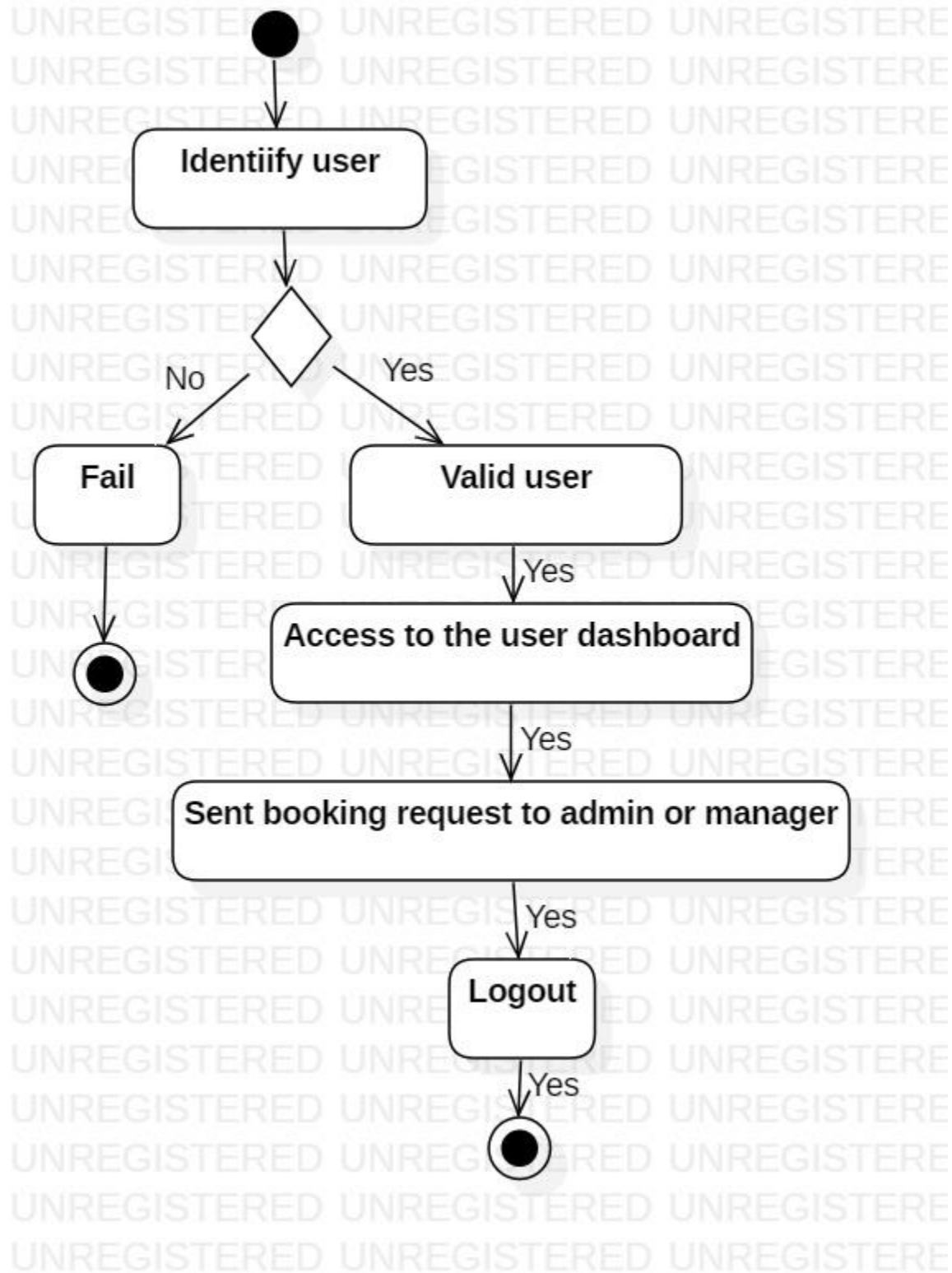


Figure 18 : 6.3.12 Activity diagram for Send flat booking request

5.4 Sequence Diagram

5.4.1 Sequence Diagram for Authentication

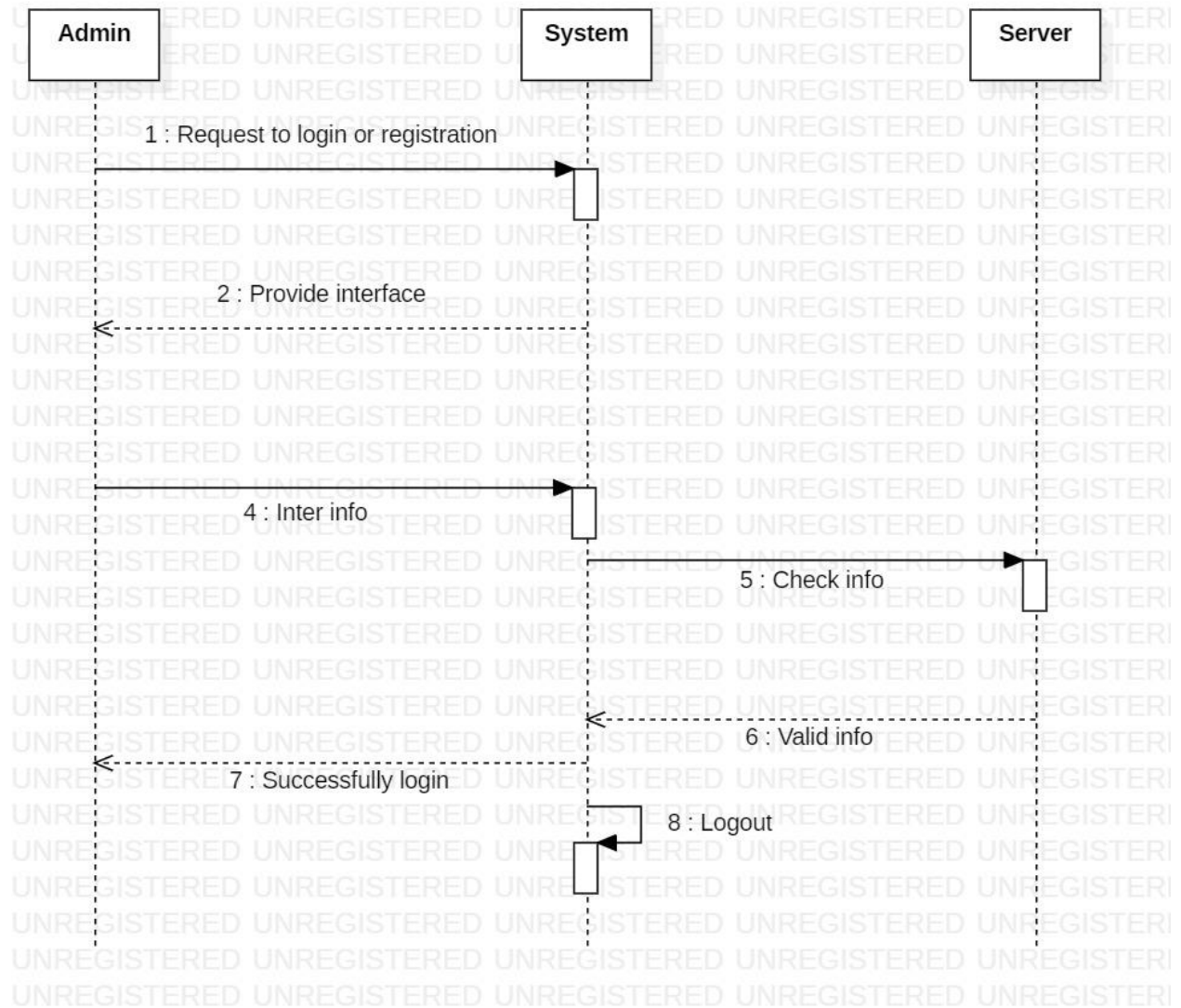


Figure 19 : 6.4.1 Sequence Diagram for Authentication

5.4.2 Sequence Diagram for Account status

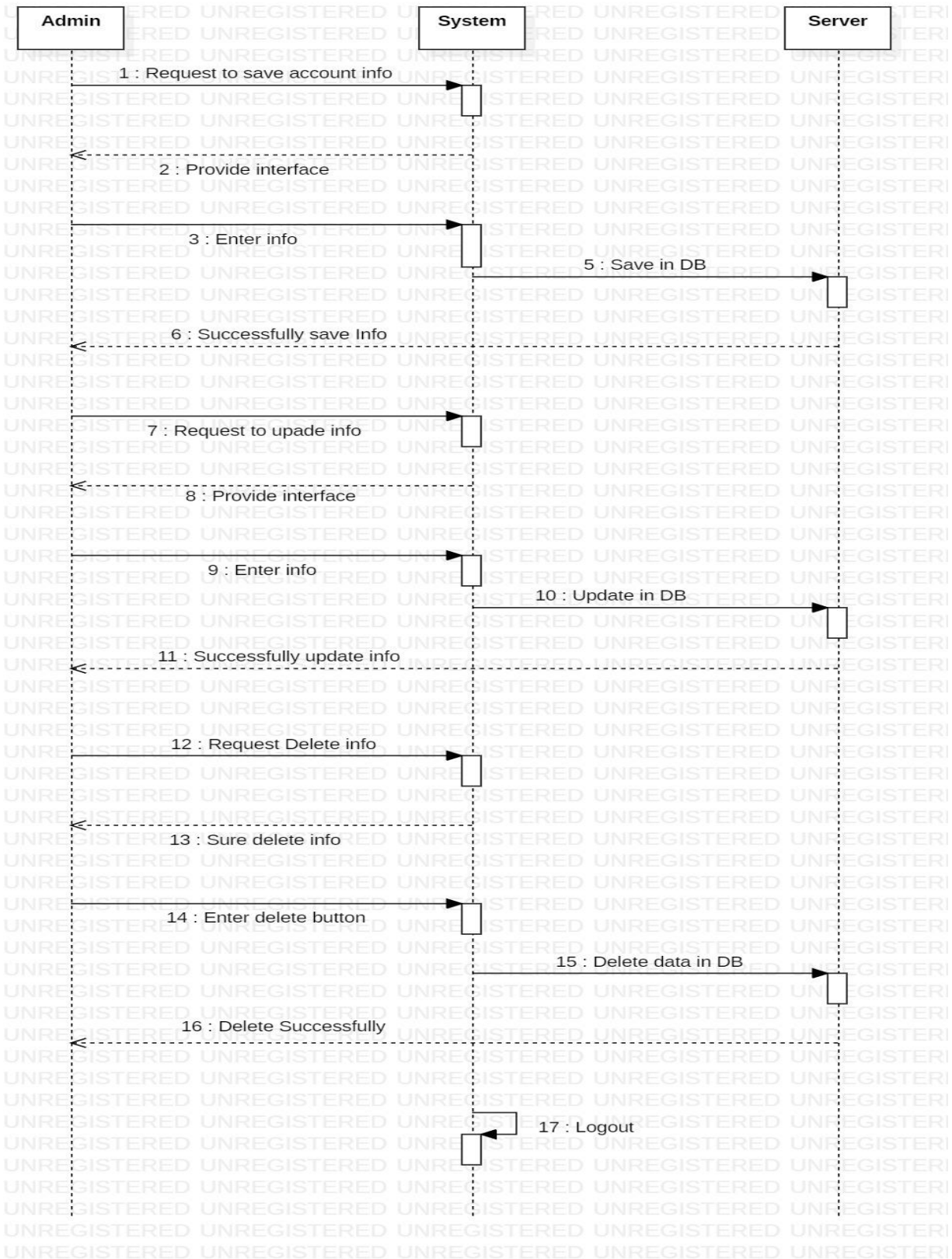


Figure 20 : 6.4.2 Sequence Diagram for Account status

5.4.3 Sequence Diagram for Bill generation

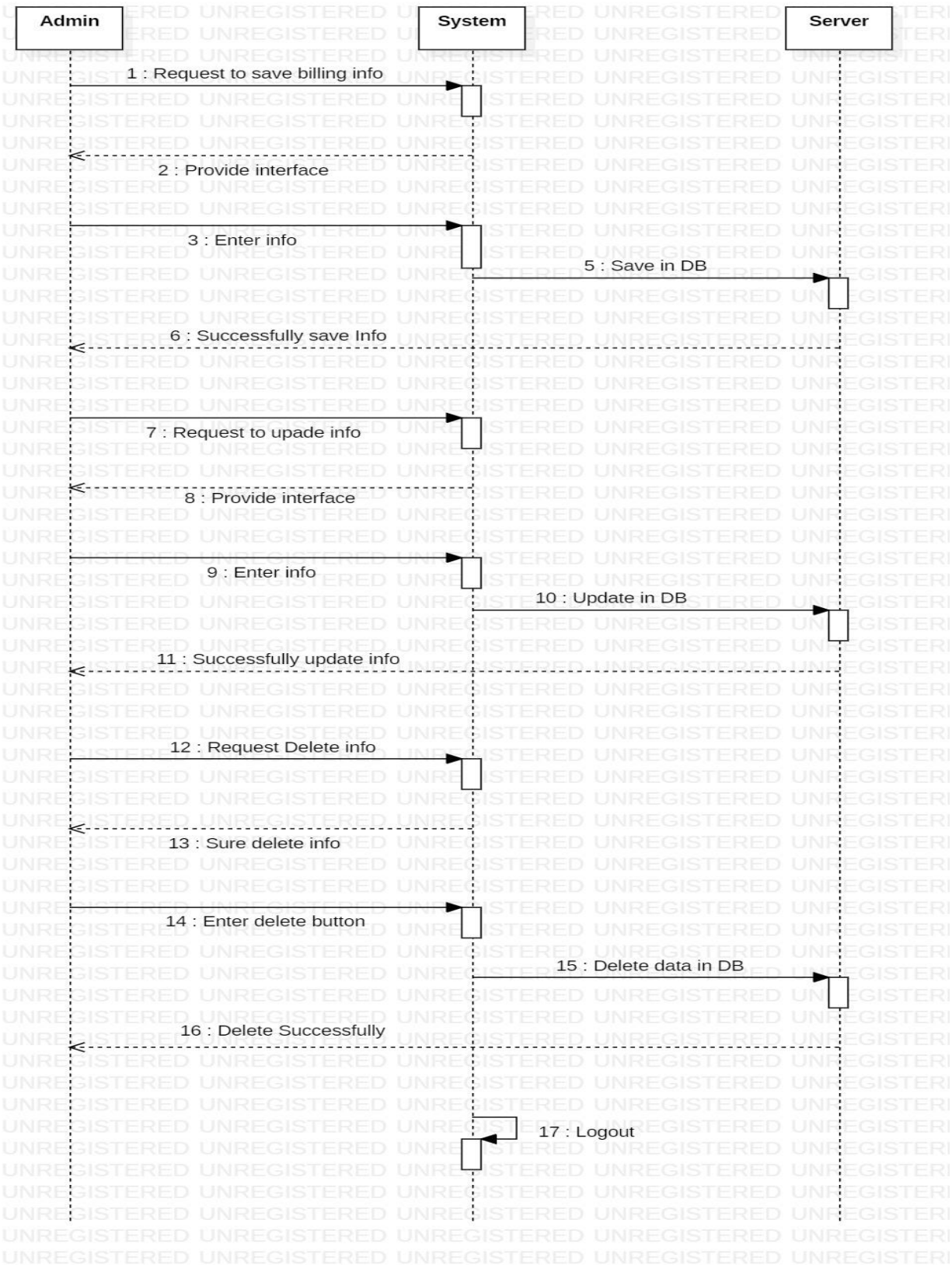


Figure 21: 6.4.3 Sequence Diagram for Bill generation

5.4.4 Sequence Diagram for Manage all employee

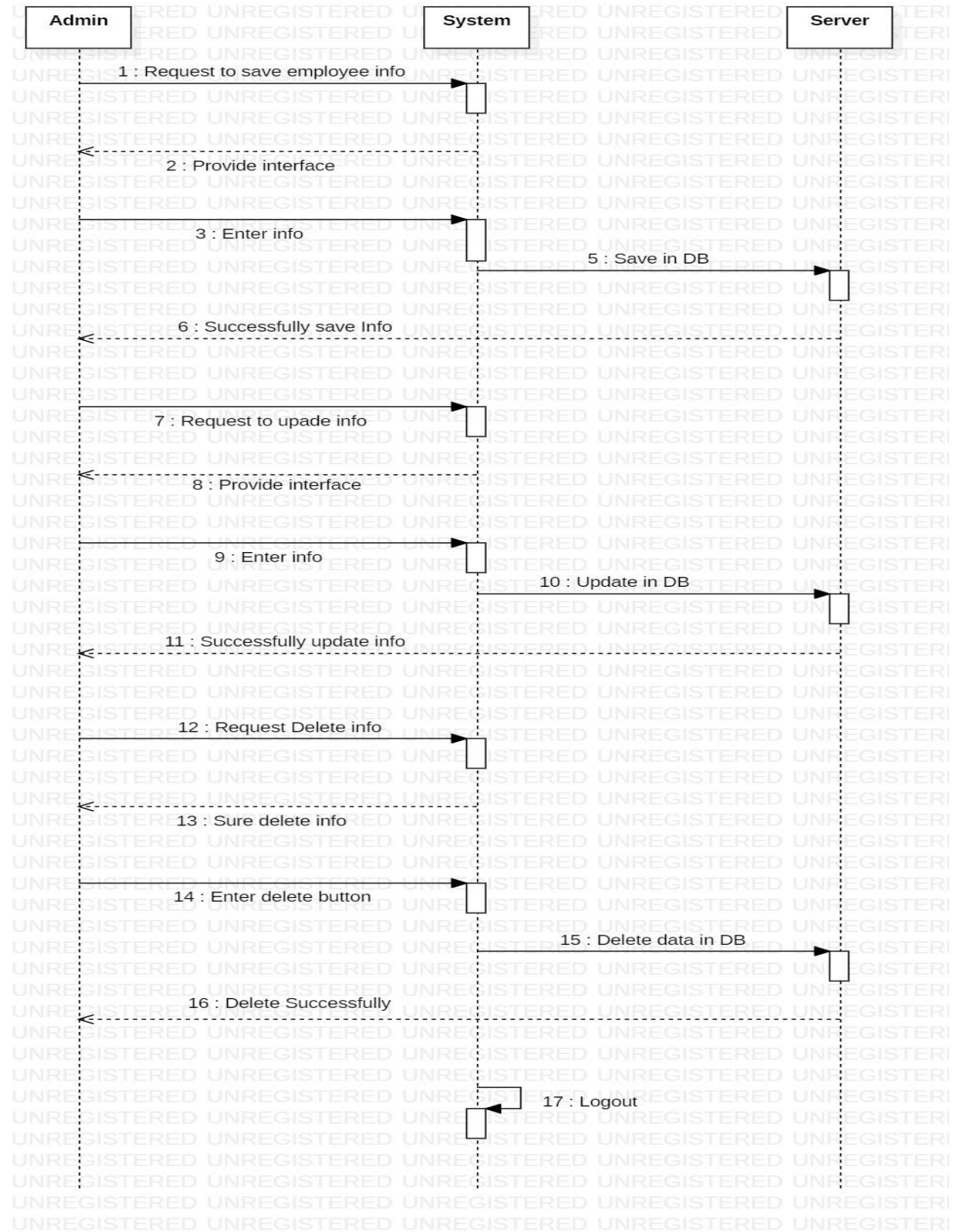


Figure 22 : 6.4.4 Sequence Diagram for Manage all employee

5.4.5 Sequence Diagram for Create Renter info

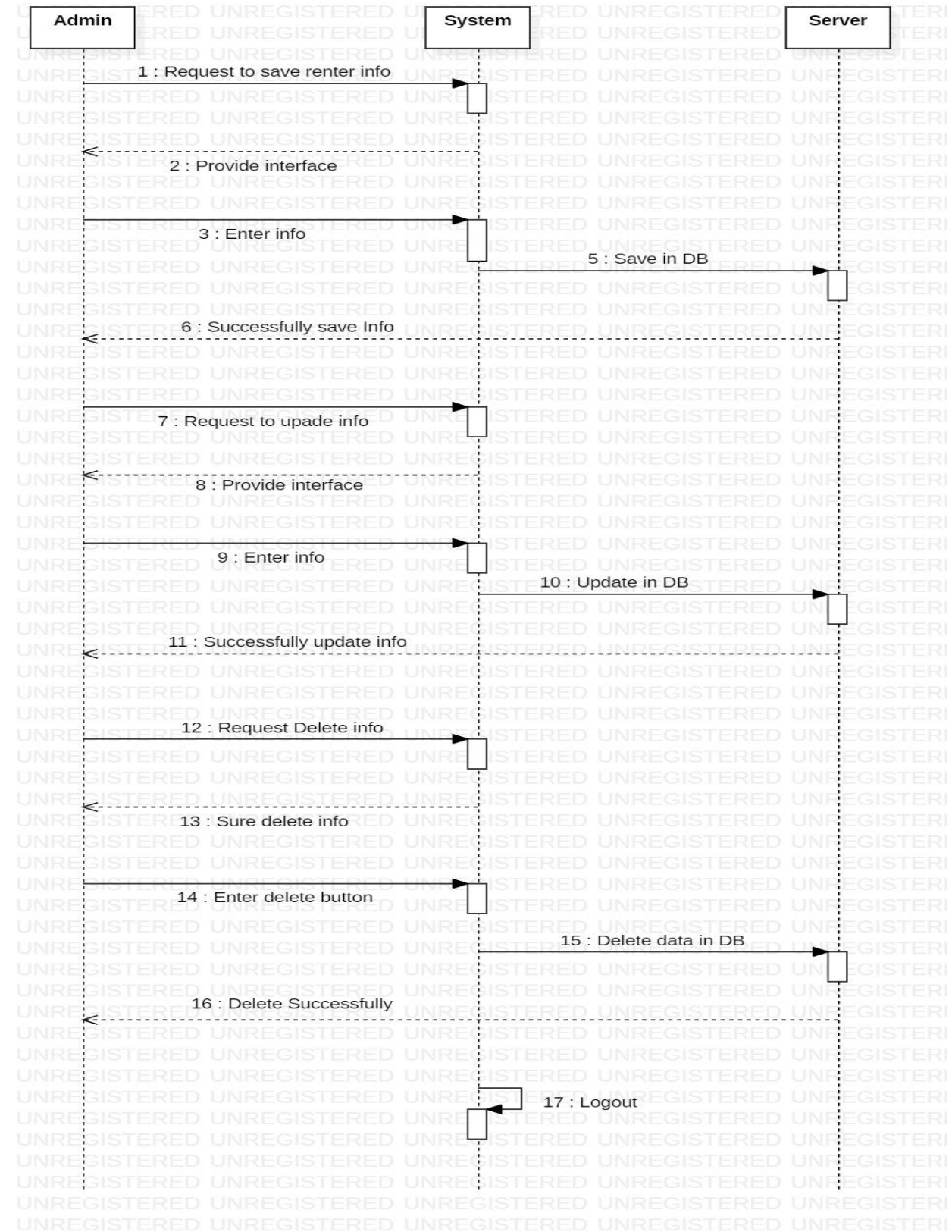


Figure 23 : 6.4.5 Sequence Diagram for Create Renter info

5.4.6 Sequence Diagram for Check user complain

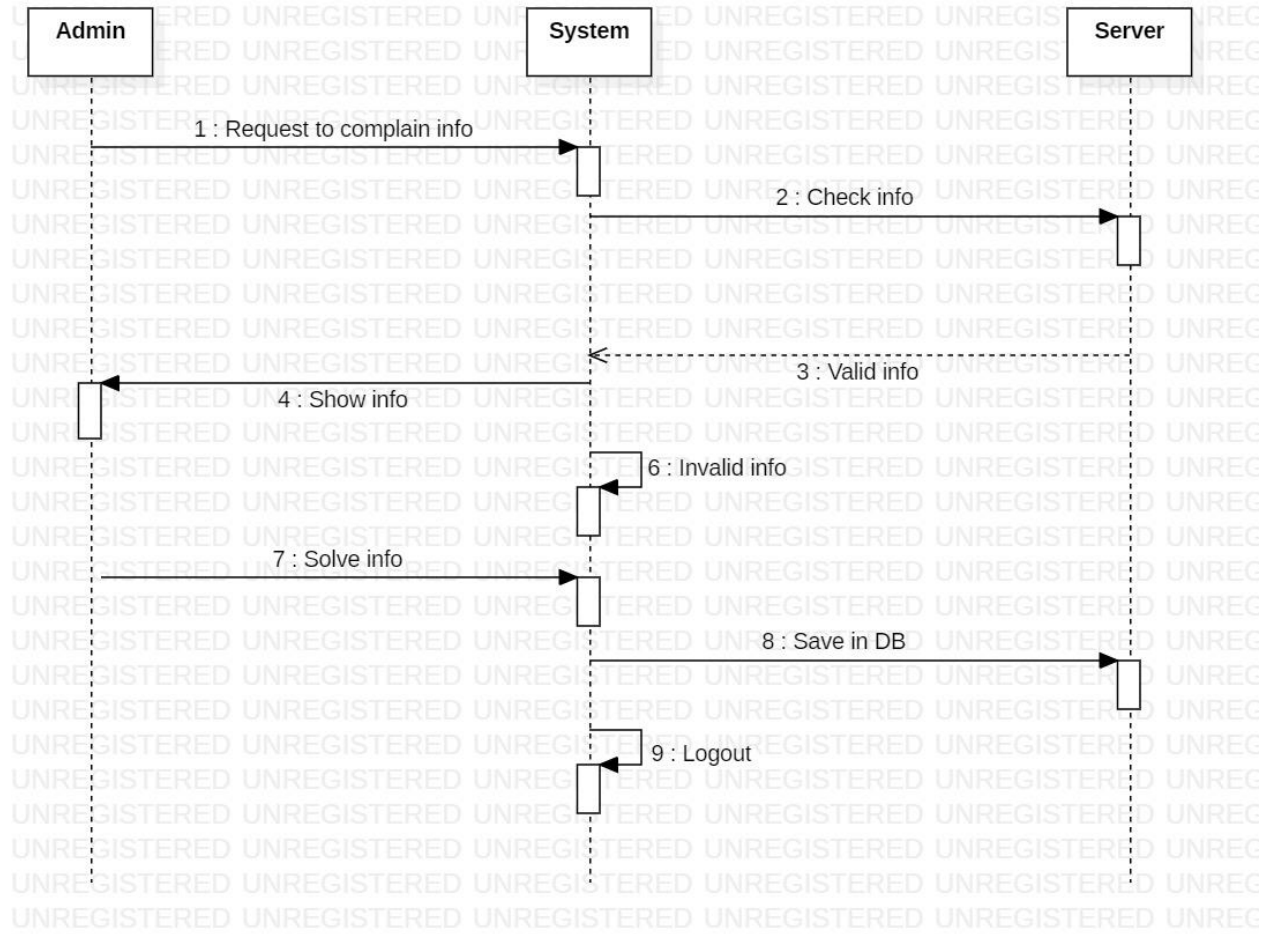


Figure 24 : 6.4.6 Sequence Diagram for Check user complain

5.4.7 Sequence Diagram for Create proposal & agreements

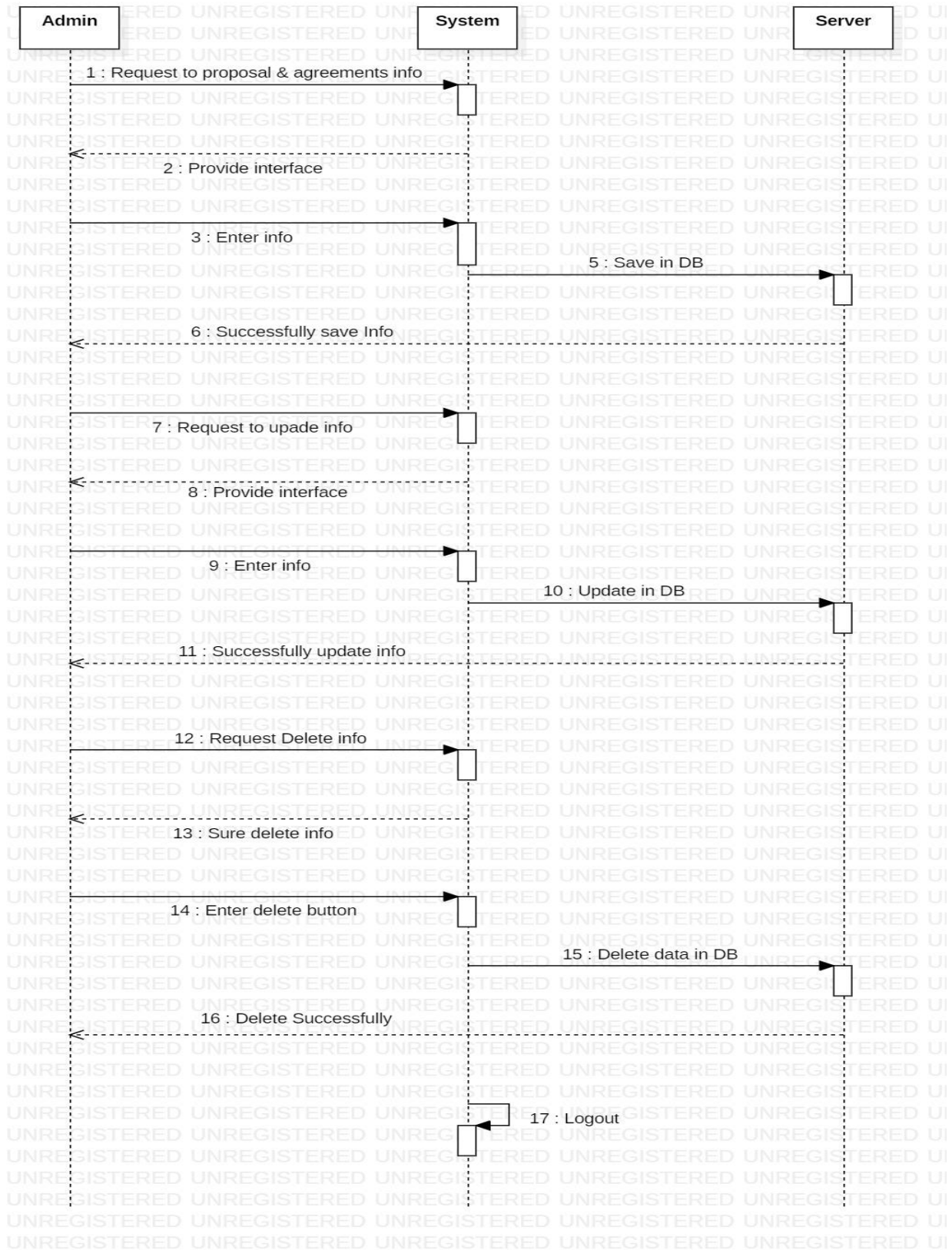


Figure 25 : 6.4.7 Sequence Diagram for Create proposal & agreements

5.4.8 Sequence Diagram for Create booking info

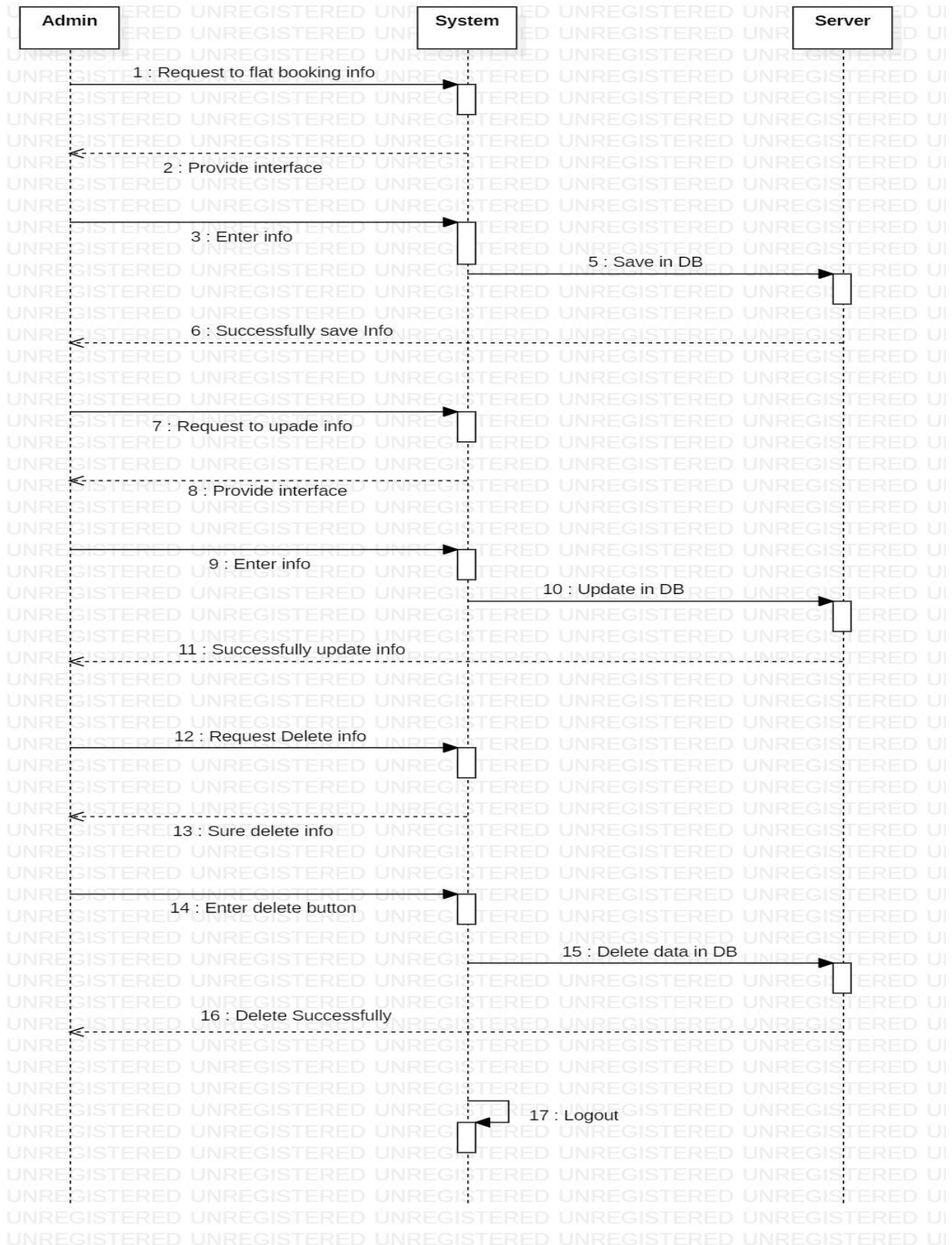


Figure 26 : 6.4.8 Sequence Diagram for Create booking info

5.4.9 Sequence Diagram for Check booking request

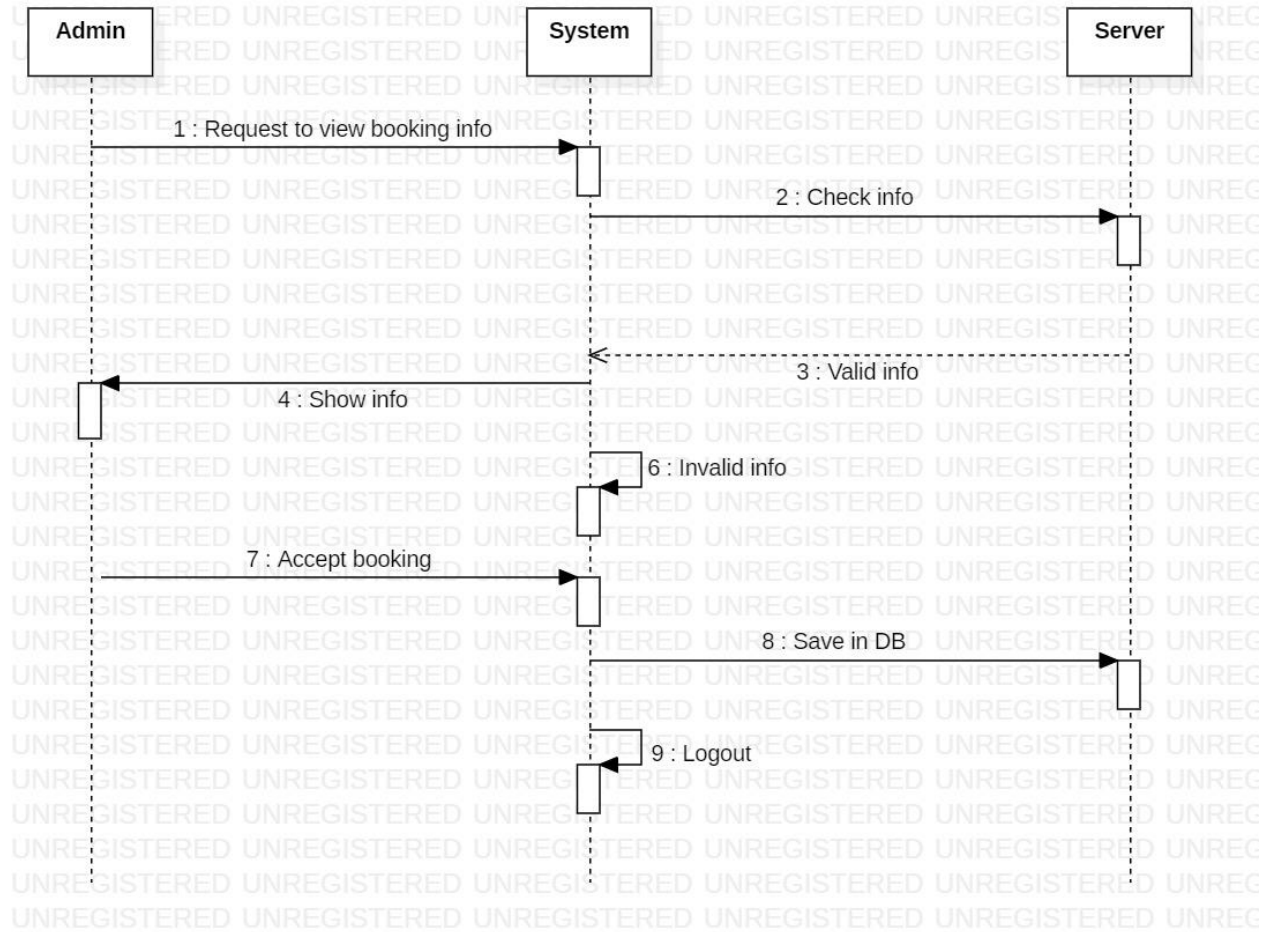


Figure 27 : 6.4.9 Sequence Diagram for Check booking request

5.4.10 Sequence Diagram for Create user info

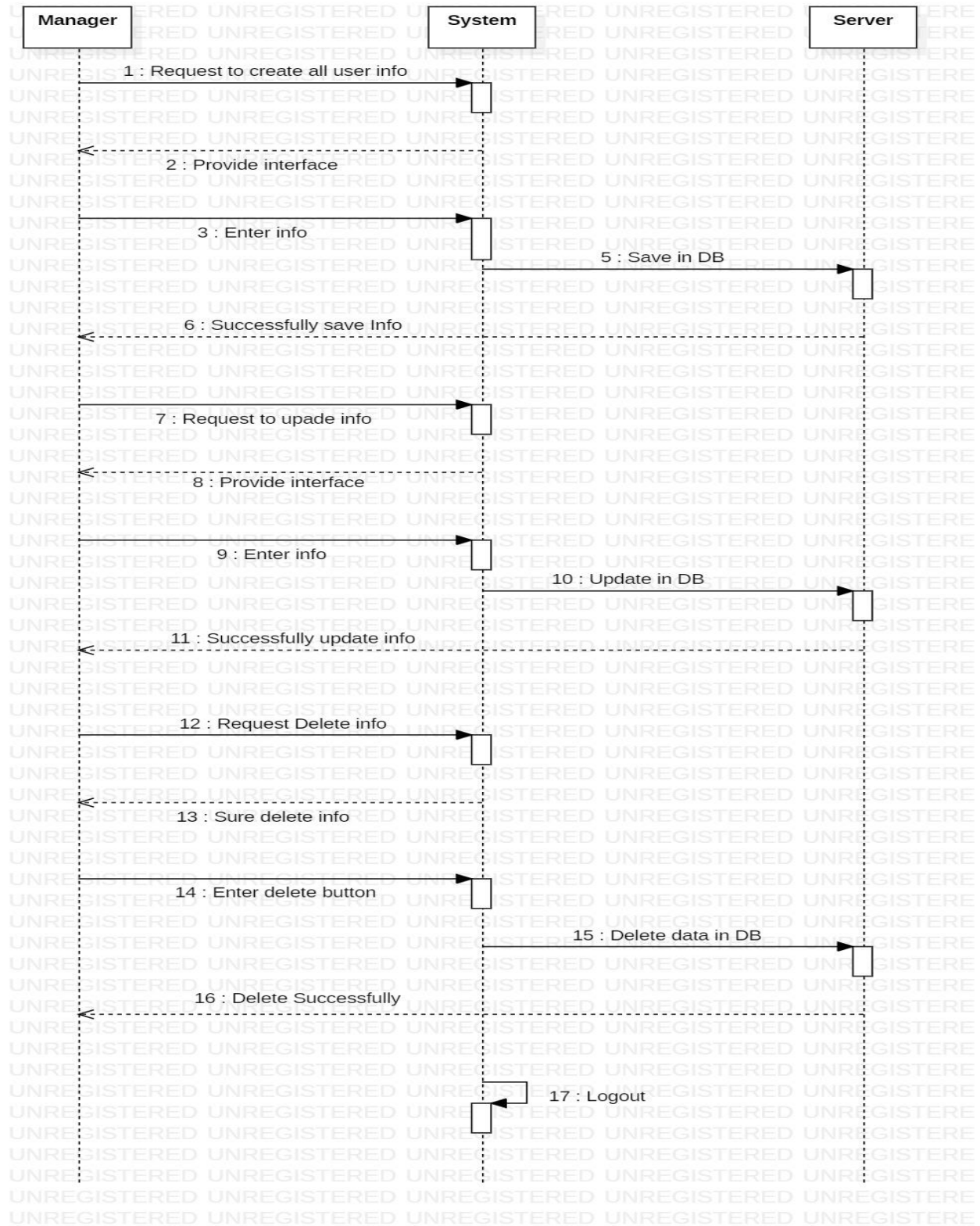


Figure 28 : 6.4.10 Sequence Diagram for Create user info

5.4.11 Sequence Diagram for Check user complain

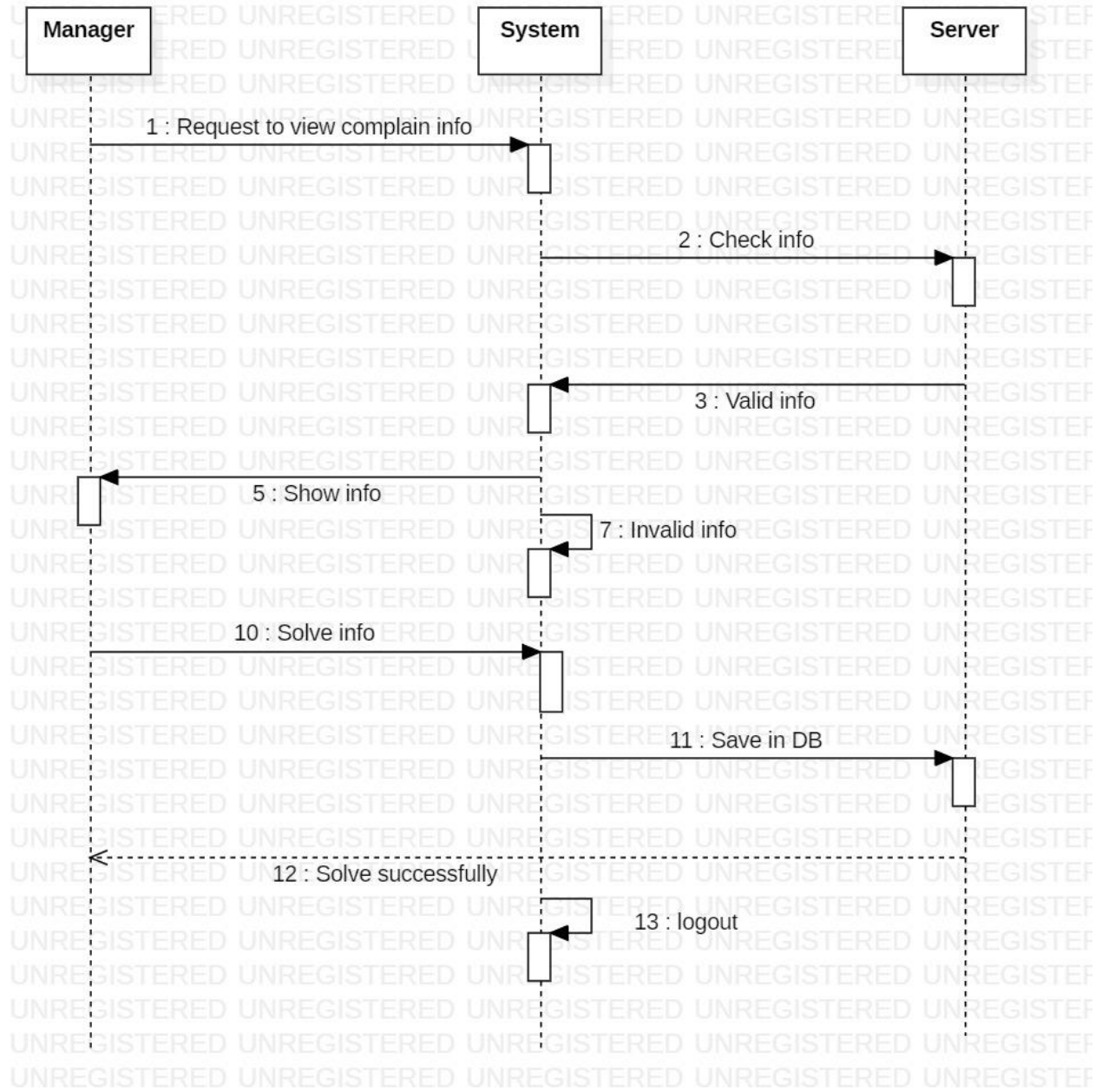


Figure 29 : 6.4.11 Sequence Diagram for Check user complain

5.4.12 Sequence Diagram for Create flat booking info

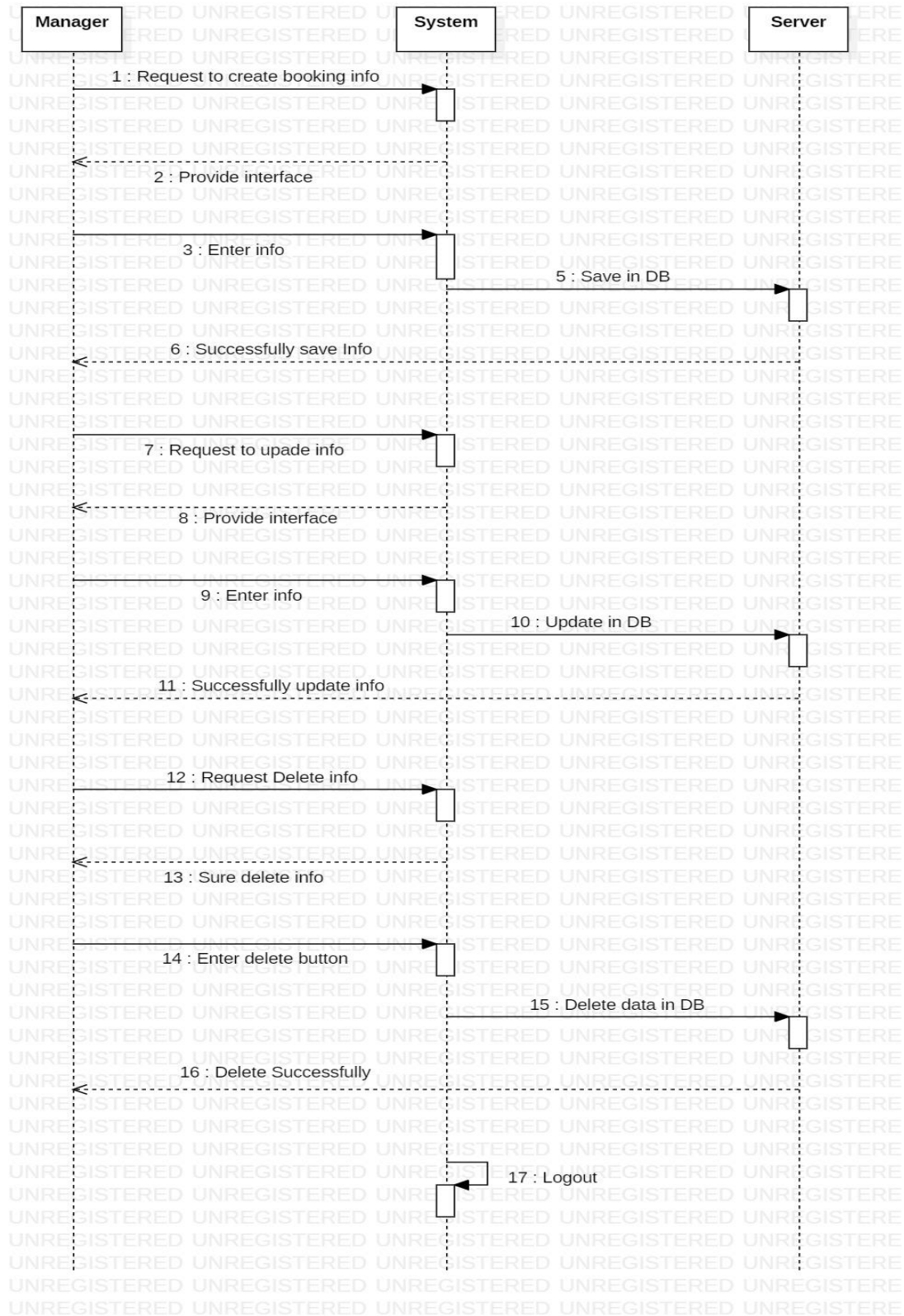


Figure 30 : 6.4.12 Sequence Diagram for Create flat booking info

5.4.13 Sequence Diagram for Create billing info

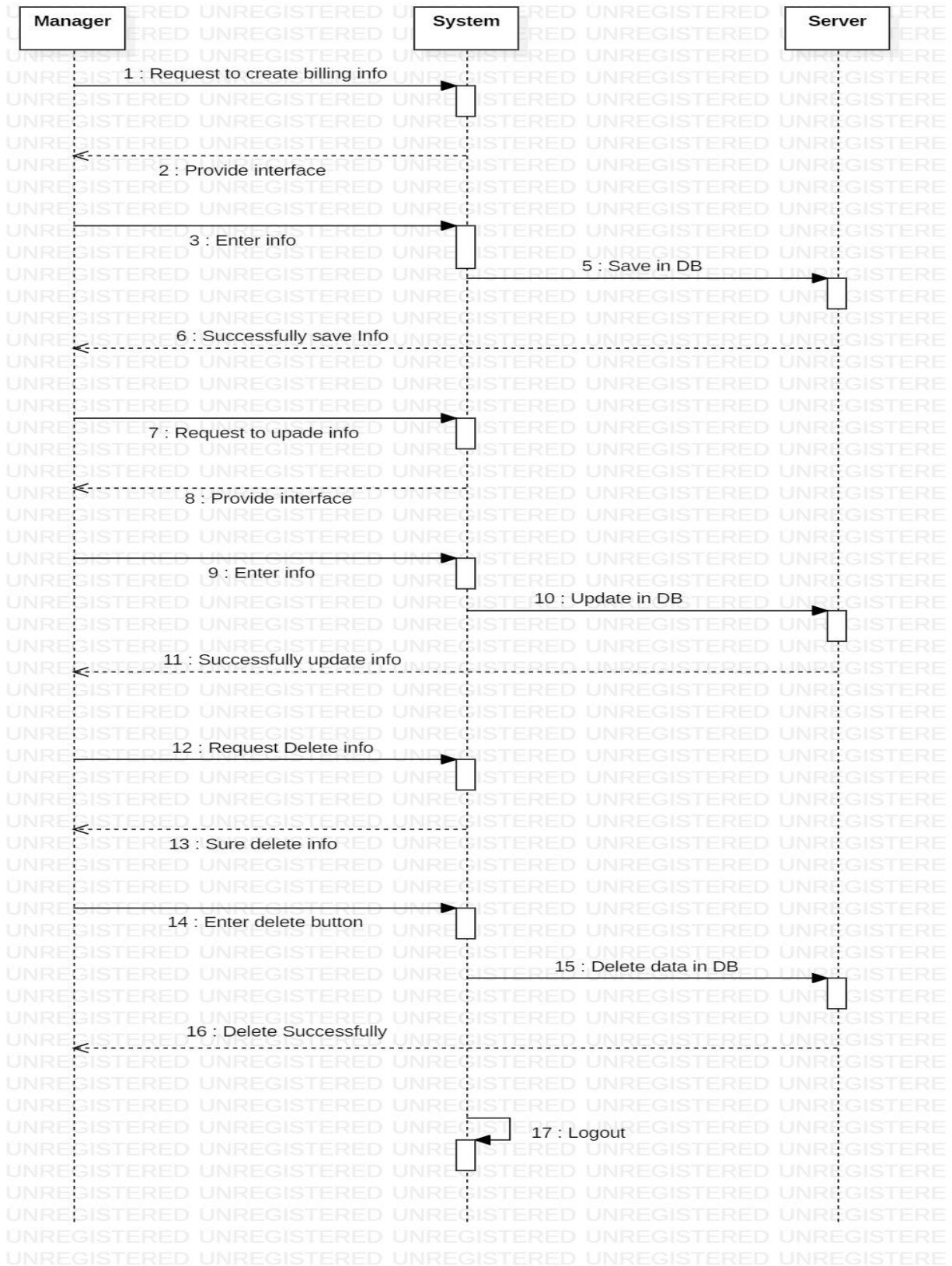


Figure 31 : 6.4.13 Sequence Diagram for Create billing info

5.4.14 Sequence Diagram for Complain for any problem

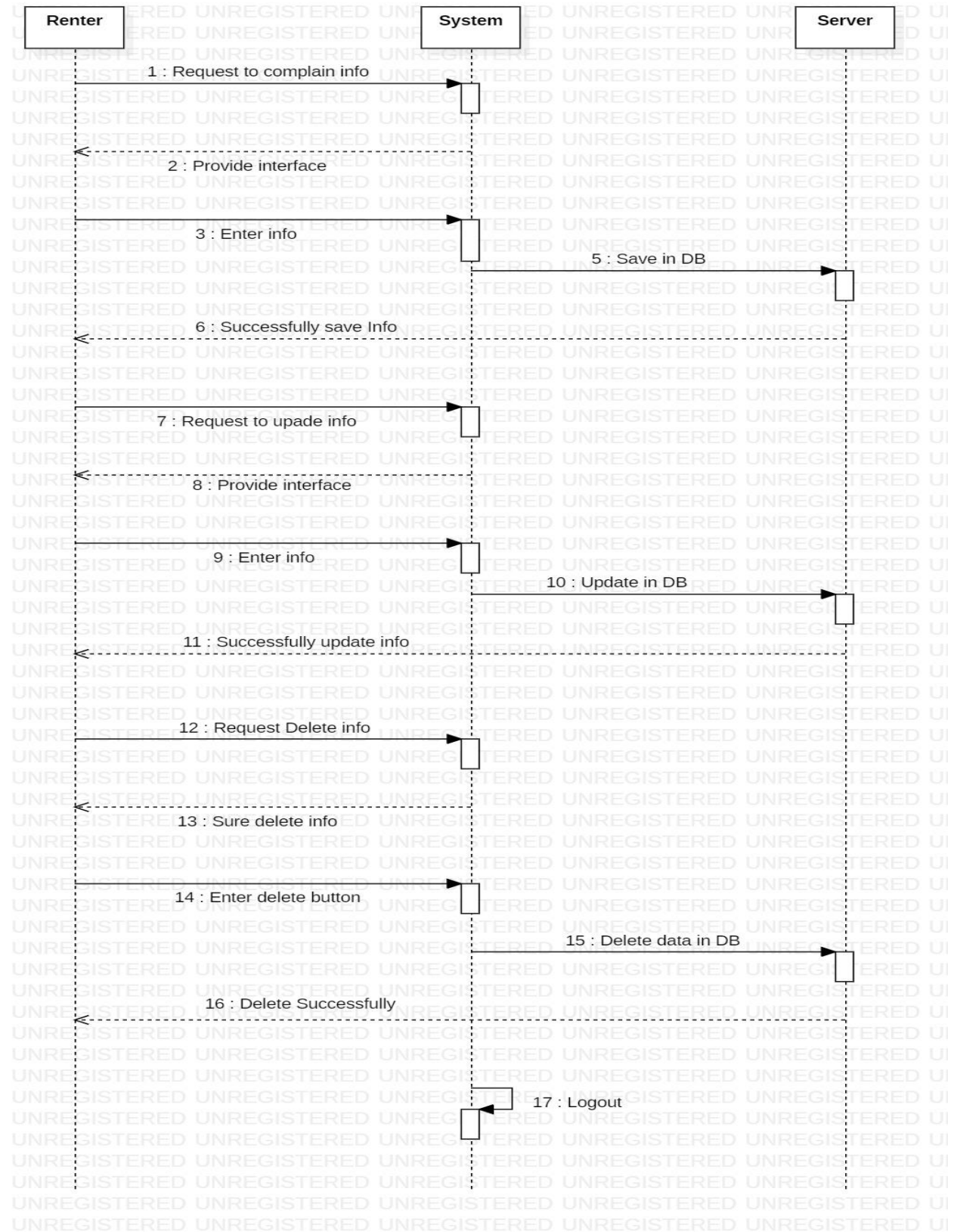


Figure 32 : 6.4.14 Sequence Diagram for Complain for any problem

5.4.15 Sequence Diagram for View user info

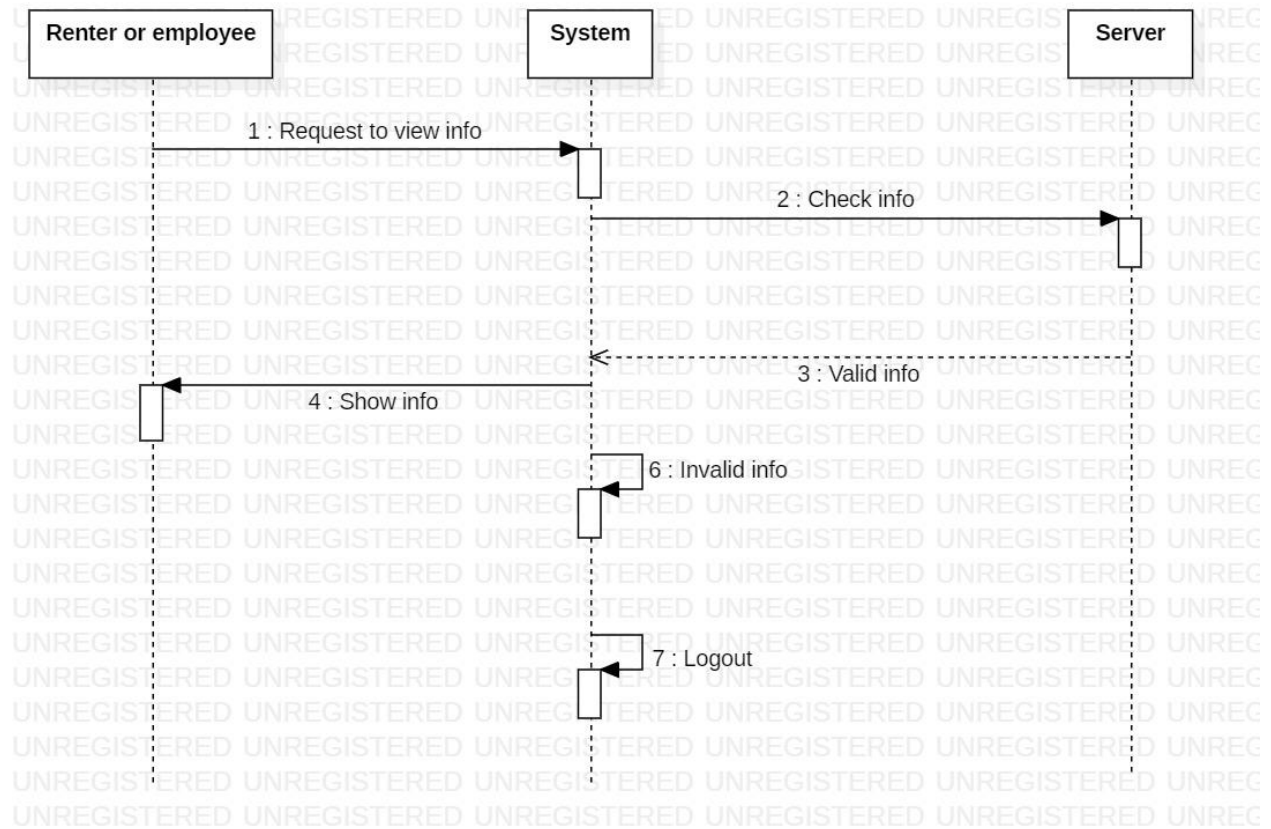


Figure 33 : 6.4.15 Sequence Diagram for View user info

5.4.16 Sequence Diagram for Send flat booking request

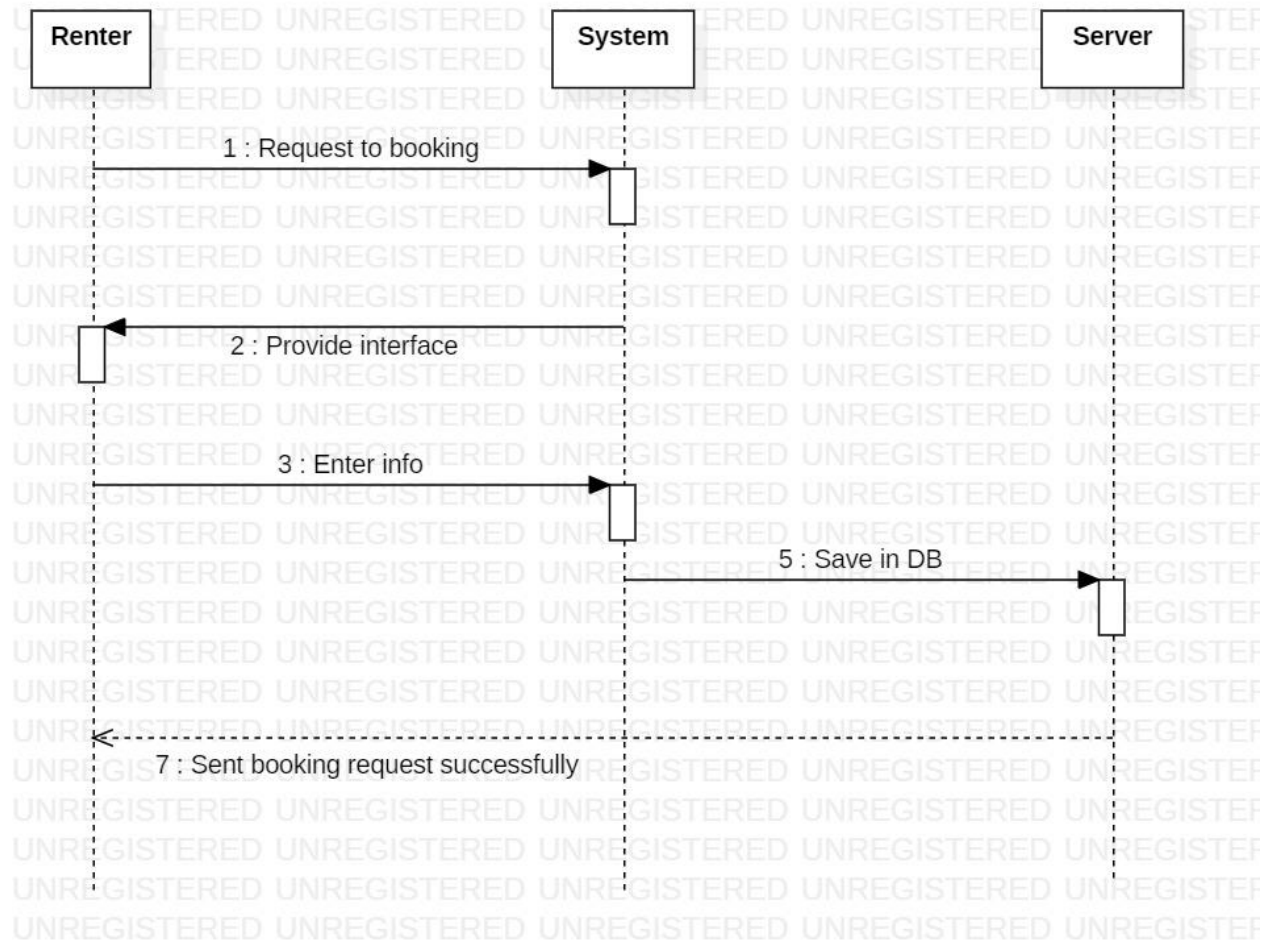


Figure 34 : 6.4.16 Sequence Diagram for Send flat booking request

5.4.17 Sequence Diagram for Complain for any problem

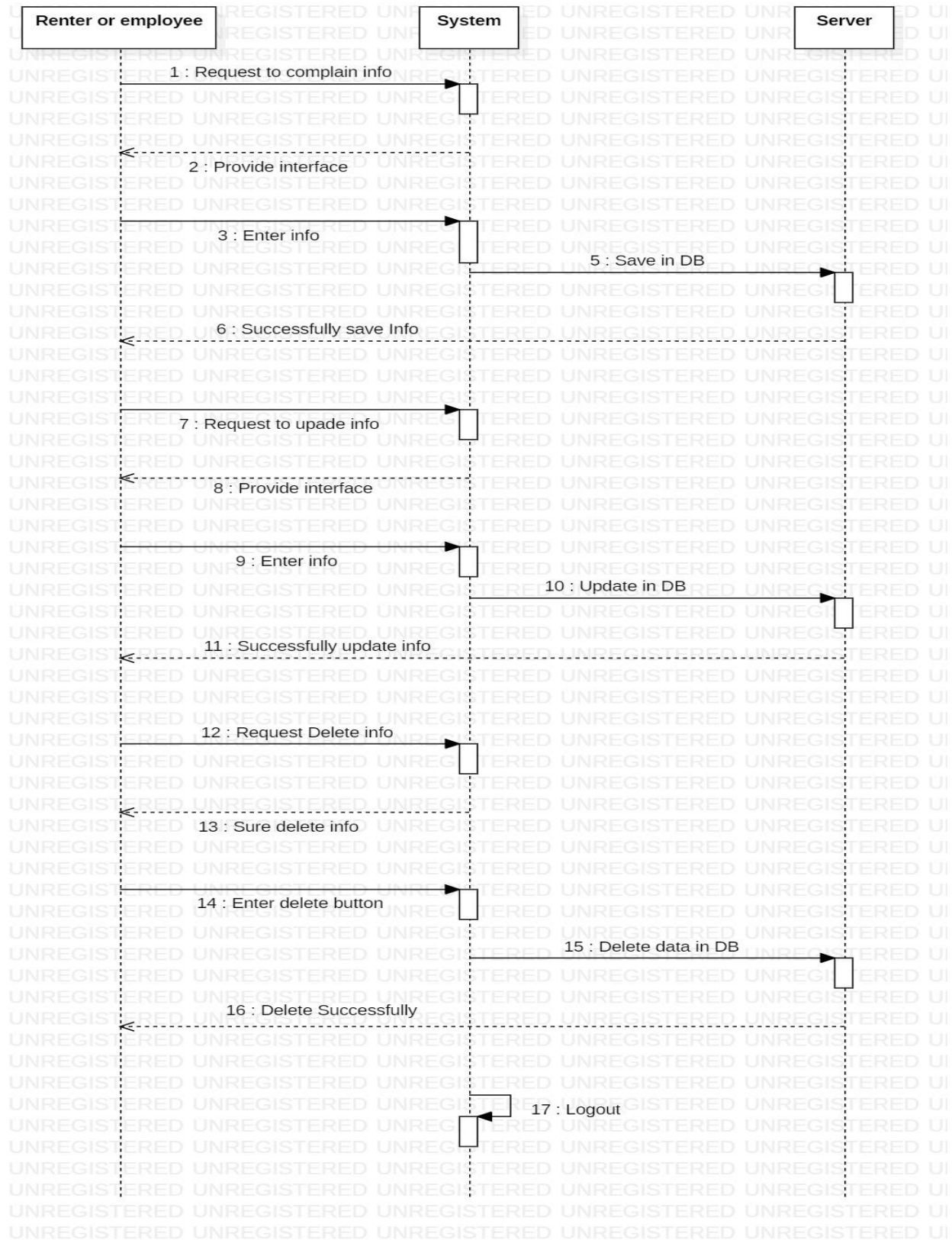


Figure 35 : 6.4.17 Sequence Diagram for Complain for any problem

6 Chapter

System Design Specification

6.1 Entity Relationship Diagram

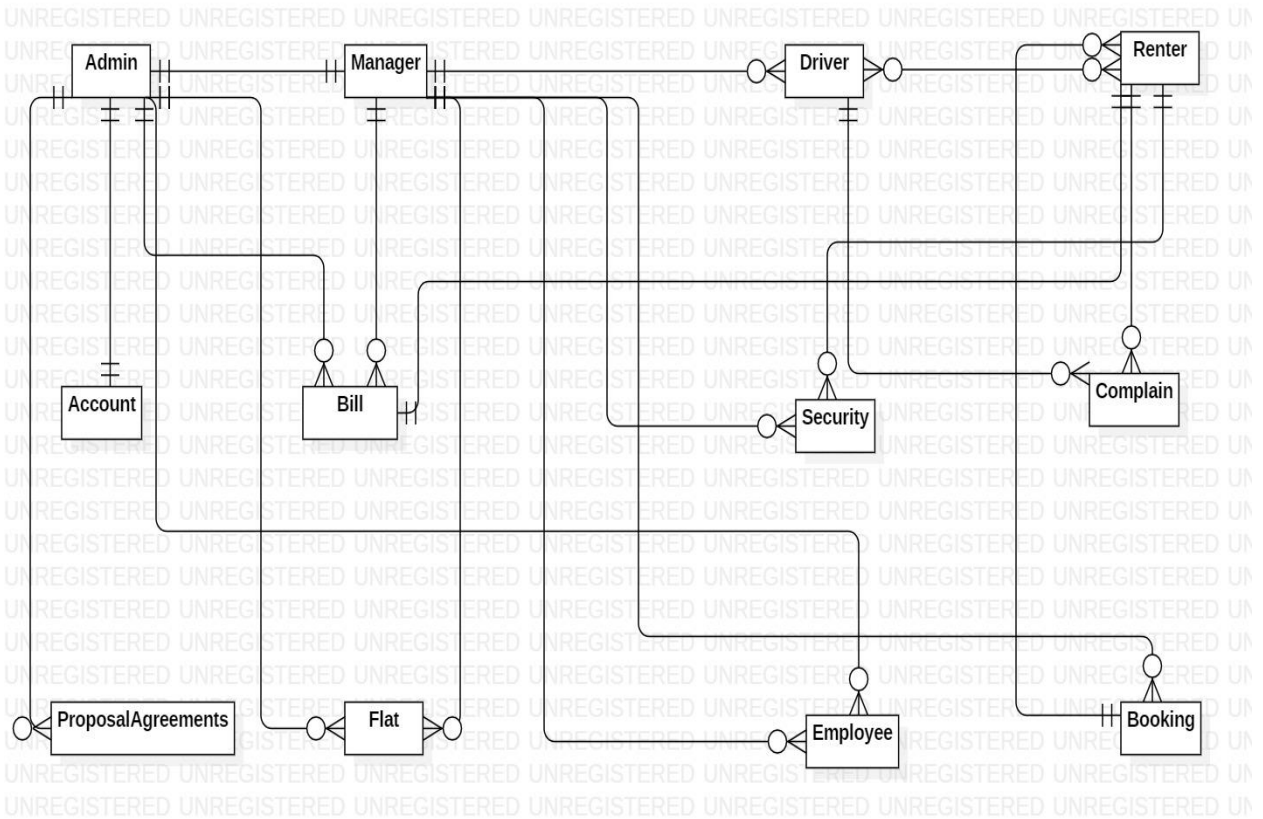


Figure 36 : Entity Relationship Diagram

6.2 Class Diagram

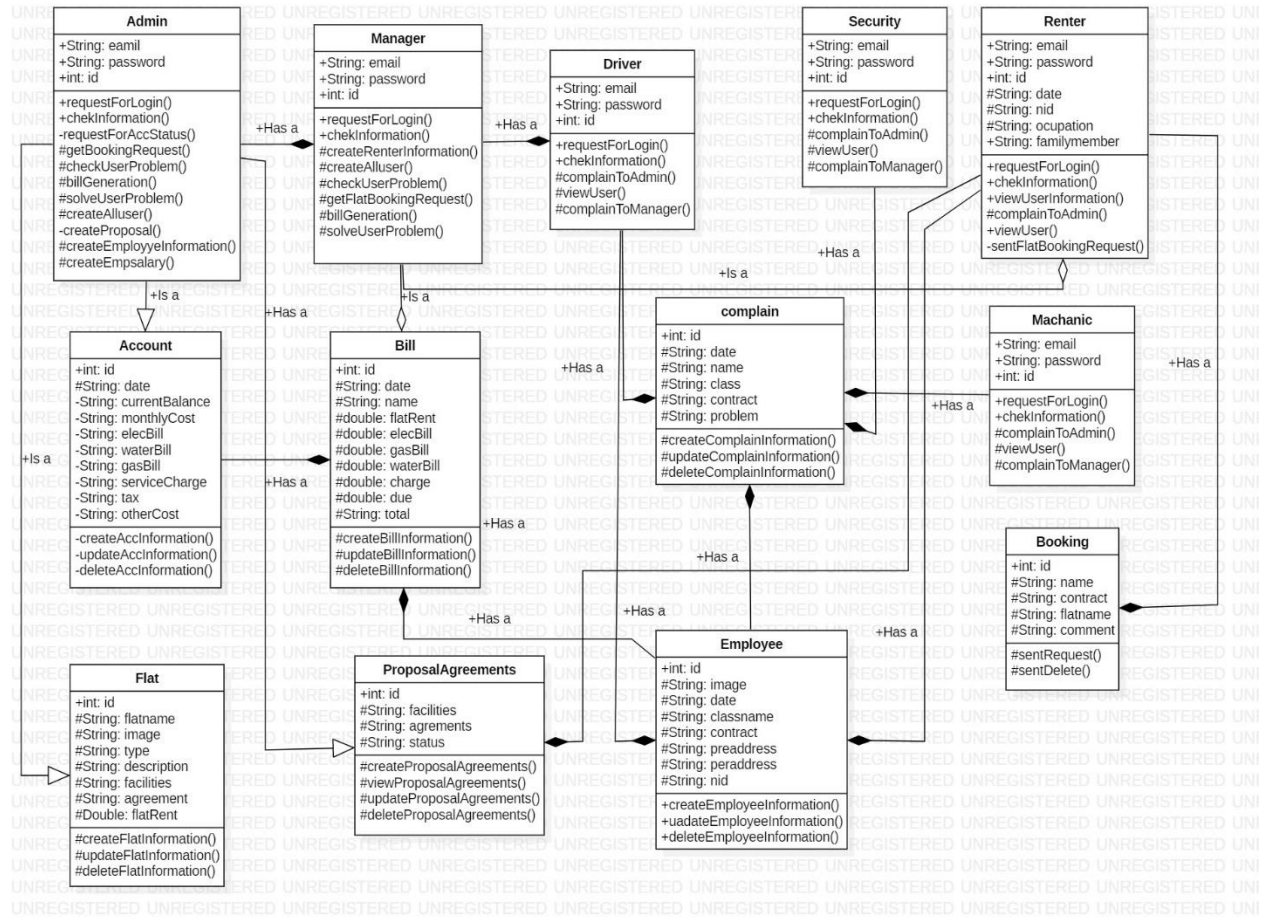


Figure 37 : Class Diagram

6.3 7.3 Development of tools And Technology

6.3.1 User Interface Technology

- ✓ HTML5
- ✓ CSS3
- ✓ JavaScript

6.3.2 Implementation Tools & Platforms

- ✓ Xampp (v3.2.2)
- ✓ Windows CMD
- ✓ Laravel v.6

7 Chapter

System Testing

7.1 Testing Features

7.1.1 Features to be tested

- ✓ Registration
- ✓ Login

7.1.2 Features not to be tested

- ✓ Rating System
- ✓ Project Comment
- ✓ IDE System

7.2 Testing Strategies

7.2.1 Test Approach

- ✓ The whole system will be tested manually
- ✓ System testing based on User acceptance.
- ✓ System testing based on admin activities

7.2.2 Pass/Fail Criteria

- ✓ Component Pass/Fail criteria – The test will pass if the case meet the object design requirement or fail if not.
- ✓ Integration Pass/Fail criteria – The test will pass if the case meets the object design architecture requirement or fail if not.

7.2.3 Suspension and Resumption

- ✓ Regression Testing – The system should work properly after each change on the system.
- ✓ Database change: The system will not work properly if we change database name.
- ✓ Build Acceptance Test – The system will pass the test if every build is successful if not then try the build again.
- ✓ System Design Changes – The system should work properly after each change in the design.

7.2.4 Testing Schedule

Test Phase	Time
Test Plan Creation	1 week
Test Specification Creation	1 week
Test Specification Team Review	2 weeks
Component Testing	2 weeks
Integration Testing	2 weeks
System Testing	3 weeks

Table 37 : Testing Schedule (Programmer Form)

7.3 Test case

7.3.1 Test case 01

Test case #: 01	Test case name: Sign up
System: User Login	Subsystem: N/A
Design by: Md. Mithu Hasan (163-35-1784)	Design Date:10-07-2020
Execute By: Md. Mithu Hasan	Execution date :15-07-2020

Table 38 : User login

Step	Action	Expected System Response	Pass/fail	comment
01	When a user fills up only Name field and clicks the login	Other fields are required	Pass	Other fields are required
02	When a user clicks only login button without a fill-up any field	Fill up the required field	Pass	Fill up the Required field
03	When a user enters email like abc.com	The system should display	Pass	Fill up the previous Required field
04	When a user enters email like abc@email.com	The system will display the email field is invalid	Pass	The valid email needs to register

Table 39 : User login

7.3.2 Test Case: 02

Test case #: 02	Test case name: Check Eligibility
System: Check Eligibility	Subsystem: N/A
Design by: Md. Mithu Hasan (163-35-1784)	Design Date:15-09-2020
Execute By: Md. Mithu Hasan	Execution date :20-09-2020

Table 40 : Check Eligibility

Step	Action	Expected System Response	Pass/fail	comment
01	When a user fills up only about user	Other fields are required	Pass	Other fields are required
02	When a user clicks only employment without a fill-up any field	Fill up the required field	Pass	Fill up the Required field
03	When user only choose banking relation from without previous from	The system should display	Pass	Fill up the previous Required field
04	When a user enters email like abc@email.com	The system will display the email field is invalid	Pass	The valid email needs to register

Table 41 : Check Eligibility

8 Chapter

User Manual

8.1 Welcome page 01

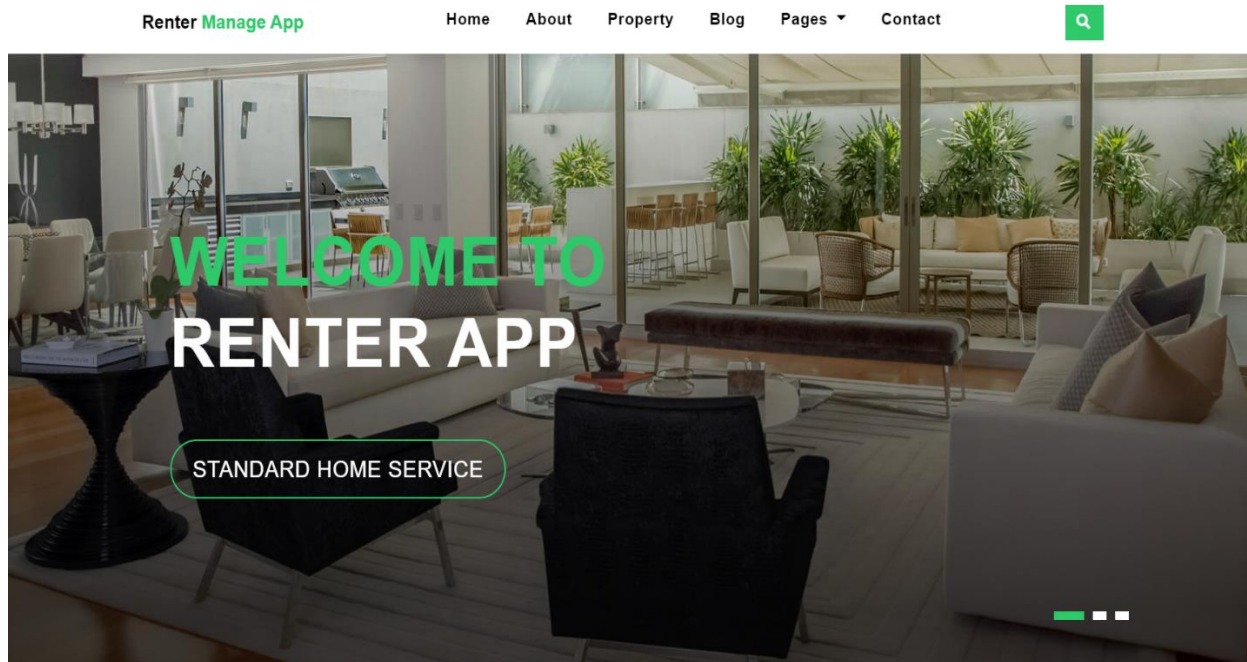


Figure 38 : 9.1 Welcome page 01

8.2 Welcome page 02

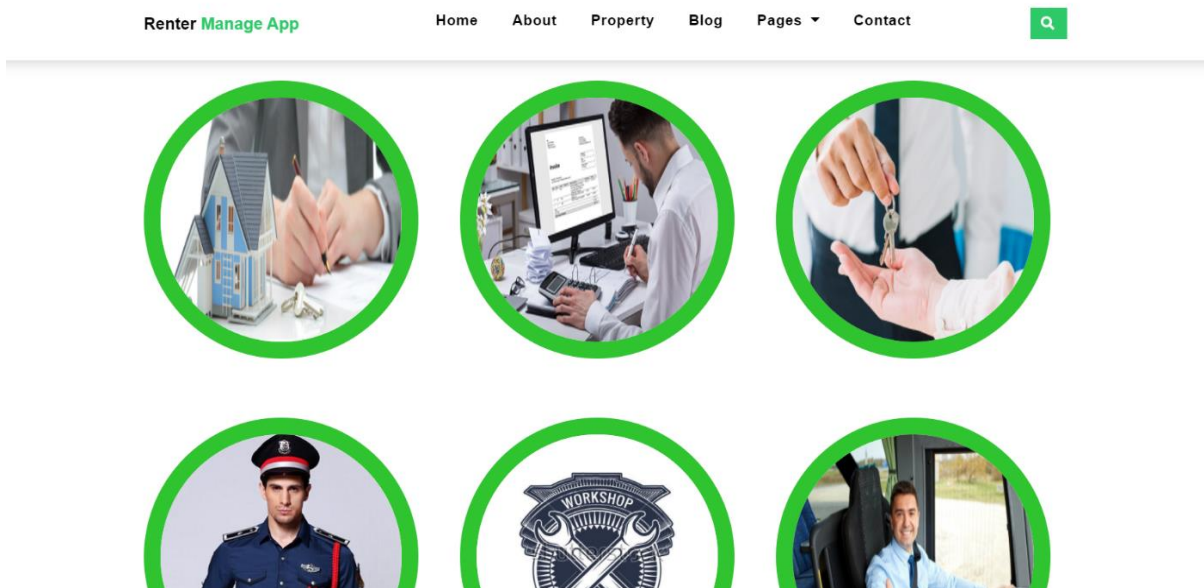
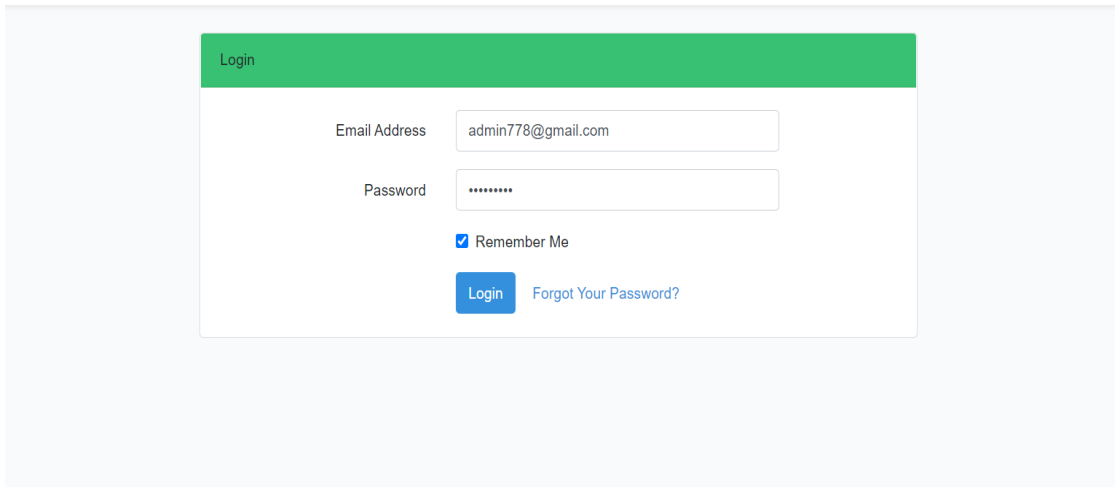


Figure 39 : 9.2 Welcome page 02

8.3 Login page

[Login](#) [Register](#)



Login

Email Address

Password

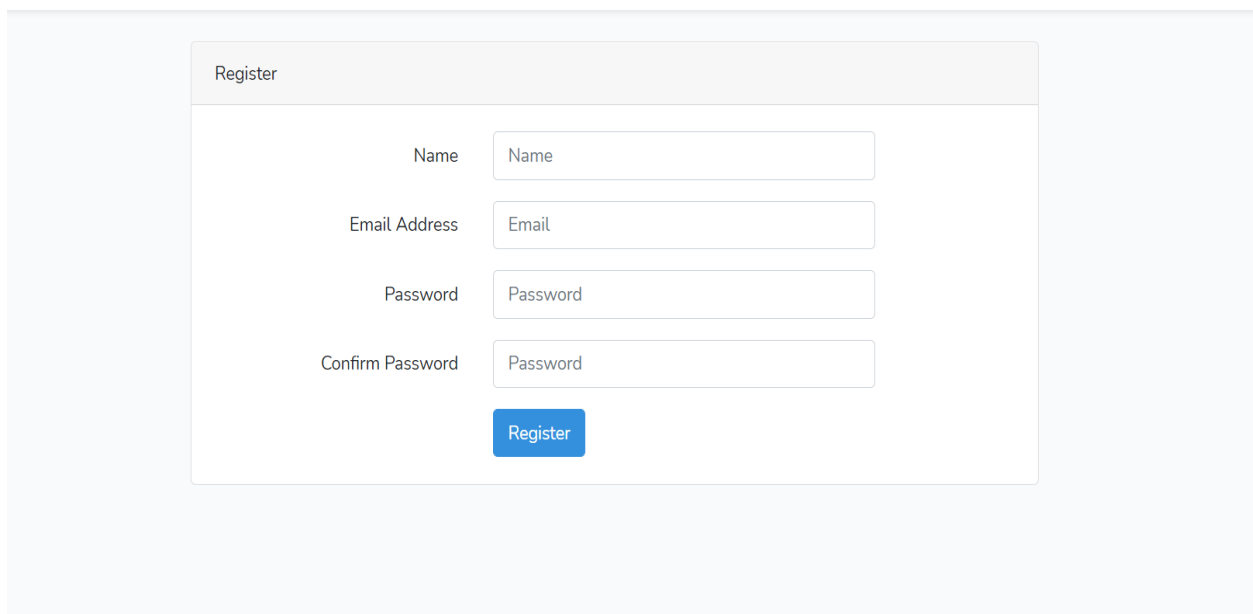
Remember Me

[Forgot Your Password?](#)

Figure 40 : 9.3 Login page

8.4 Registration page

[Login](#) [Register](#)



Register

Name

Email Address

Password

Confirm Password

Figure 41 : 9.4 Registration page

8.5 Admin dashboard page

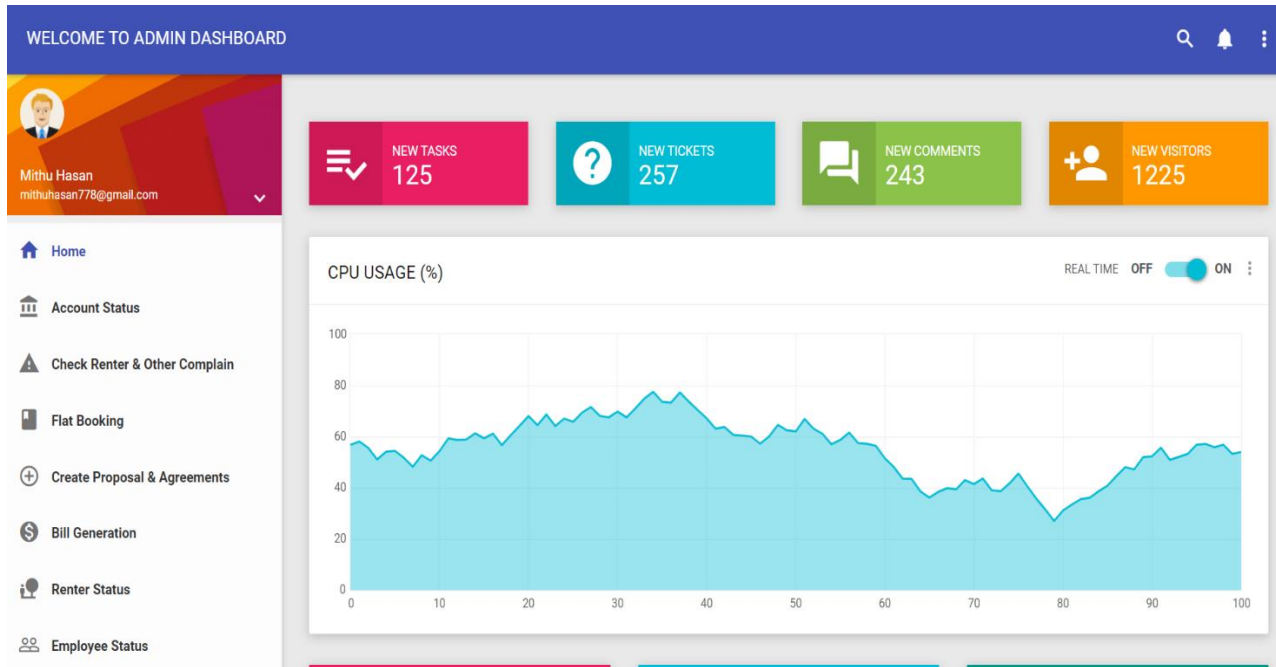


Figure 42 : 9.5 Admin dashboard page

8.6 Account status page

Admin ▾

[Back Dashboard](#)

All Account Information

Date	CurrentBalance	Cost	EmpSalary	ElectriBill	WaterBill	GasBill	ServiceCharge	Tax	Other Cost	Action
2019-05-17	222000	660000	3000	323	647467	7657568	457568	8777	600	
2019-06-02	278900	660000	3000	3230	647467	7600	10000	800	7000	
2019-07-02	4770000	660000	30000	3230	6000	6000	457568	400	6800	
2021-01-06	6	60000	30000	3230	2407	6000	10000	500	900	

Add Account Form

Date

Figure 43 : 9.6 Admin dashboard page

8.7 User complain page

Admin ▾

Complain					
Date	Name	Class	Contract	Problem	Action
2020-11-02	Mithu Hasan	Tenant	01656-654543	Gas line Problem	Delete Edit
2020-11-03	Jasim	Driver	01956-654543	My dress is needed	Delete Edit
2020-11-26	Rupa Khan	Tenant	01676-654543	Water line Problem	Delete Edit
2020-11-17	Rubel Islam	Tenant	01856-654543	Electricity line problem	Delete Edit
2020-11-02	Mithu Hasan	Tenant	01656-654543	Gas line Problemnbnvhn	Delete Edit

Figure 44 : 9.7 Admin dashboard page

8.8 User complain delete page

Admin ▾

Complain					
Date	Name	Class	Contract	Problem	Action
2020-11-02	Mithu Hasan	Tenant	01656-654543	Gas line Problem	Delete Edit
2020-11-03	Jasim	Driver	01956-654543	My dress is needed	Delete Edit
2020-11-26	Rupa Khan	Tenant	01676-654543	Water line Problem	Delete Edit
2020-11-17	Rubel Islam	Tenant	01856-654543	Electricity line problem	Delete Edit
2020-11-02	Mithu Hasan	Tenant	01656-654543	Gas line Problemnbnvhn	Delete Edit

Are you sure to delete ?

Don't recover this information.Deleted information are still intact until they are written over.

[Yes](#) [No](#)

Figure 45 : 9.8 Admin dashboard page

8.9 User complain Update page

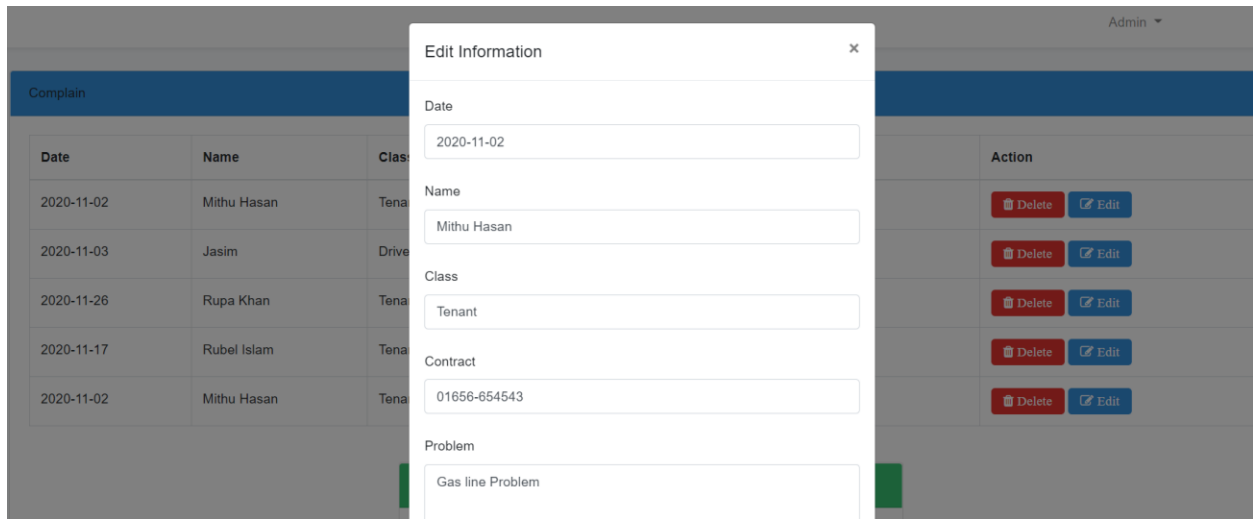


Figure 46 : 9.9 User complain Update page

8.10 Renter flat info page

[Add Flat](#)

-----All Flats-----

Flat	Image	Type	Description	Facilities	Agreement	Rent	Action
201A		Family	Two bedroom One diningroom Two bathroom One kitchen Two belkony	24 hour electricity supply No water & gas problem Strong security	No guest alow Don't waste water and gas Don't make noise Pay in 5 days envey month	6000	Delete Edit
201B		Bachelor/Students	Two bedroom One diningroom Two bathroom One kitchen Two belkony	24 hour electricity supply No water & gas problem Strong security	No guest alow Don't waste water and gas Don't make noise Pay in 5 days envey month	6000	Delete Edit
201C		Family	Two-bedroom One dining room Two bathroom One kitchen Two balcony	24-hour electricity supply No water & gas problem Strong security	No guest alow Don't waste water and gas Don't make noise Pay in 5 days every month	12000	Delete Edit
301A		Family	Two-bedroom One dining room Two bathroom One kitchen Two balcony	24-hour electricity supply No water & gas problem Strong security	No guest alow Don't waste water and gas Don't make noise Pay in 5 days every month	7600	Delete Edit

Figure 47 : 9.10 Renter flat info page

8.11 Renter flat info delete page

The screenshot shows a web interface for managing flats. A modal dialog box is open in the center, asking for confirmation to delete information. The dialog text reads: "Are you sure to delete ?" followed by "Don't recover this information. Deleted information are still intact until they are written over." There are "Yes" and "No" buttons at the bottom of the dialog. In the background, a table lists flats with columns for Flat, Image, Type, Amenities, Rules, Rent, and Action. The table contains four rows of data for flats 201A, 201B, 201C, and 301A.

Flat	Image	Type	Amenities	Rules	Rent	Action
201A		Family		Don't waste water and gas Don't make noise every month	6000	Delete Edit
201B		Bachelor/Students	Two bedroom One diningroom Two bathroom One kitchen Two belkony	24 hour electricity supply No water & gas problem Strong security	6000	Delete Edit
201C		Family	Two-bedroom One dining room Two bathroom One kitchen Two balcony	24-hour electricity supply No water & gas problem Strong security	12000	Delete Edit
301A		Family	Two-bedroom One dining room Two bathroom One kitchen Two balcony	24-hour electricity supply No water & gas problem Strong security	7600	Delete Edit

Figure 48 : 9.11 Renter flat info delete page

8.12 Billing info page

The screenshot shows a web interface for viewing billing information. At the top right, there is an "Admin" dropdown menu. Below it is a green header bar labeled "Billing Information". The main content is a table with columns for Date, Name, FlatRent, EleBill, GasBill, WaterBill, Charge, Due, Total, and Action. The table contains five rows of billing records. At the bottom left, there is a pagination control showing page 1 of 2.

Date	Name	FlatRent	EleBill	GasBill	WaterBill	Charge	Due	Total	Action
2019-08-09	Washim Akram	16000	700	500	300	500	400	890008	Delete Edit
2019-07-28	Ratre Islam	16000	700	500	300	500	400	800040	Delete Edit
2019-07-10	Mithu Hasan	16000	700	500	300	500	400	890000	Delete Edit
2019-07-10	Ektiyas Wahid	16000	700	570	300	5000	490	896900	Delete Edit
2019-08-02	Washim Akram	16000	700	500	390	500	0	789000	Delete Edit

Figure 49 : 9.12 Billing info page

8.13 Renter info page






All Renter Information									
Photo	Enrolled	Name	Contract	Present Address	Permanent Address	NID Number	Occupation	Member	Action
	2019-07-27	Md. Mithu Hasan	01726184147	Shukrabad 43/2,Dhaka-1207	Shukrabad 43/2,Dhaka-1207	876899999999	Student	6	Delete Edit
	2019-07-04	Halima Ratre	019567184147	Shukrabad 43/2,Dhaka-1207	Shukrabad 43/2,Dhaka-1207	876899999999	Student	3	Delete Edit
	2019-07-12	Farhat Anjum	01726-184147	Shukrabad 43/2,Dhaka-1207	Shukrabad 43/2,Dhaka-1207	1996675766888	Graphics Designer	5	Delete Edit
	2019-07-20	Subroto Karmoker	01726184149	Shukrabad 43/2,Dhaka-1207	Shukrabad 43/2,Dhaka-1207	8768999999990	Web developer	4	Delete Edit
	2019-	Samiul	019567184147	Surabad	Surabad	19966757668888899	Student	2	Delete Edit

Figure 50 : 9.13 Renter info page

8.14 Employee info page






All Employee Information							
Photo	Enrolled	Class & Name	Contract	Present Address	Permanent Address	NID Number	Action
	2019-08-10	Machanic: Mithu Kamal	017654545439	Shukrabad 43/2,Dhaka-1207	Shukrabad 43/2,Dhaka-1207	198000	Delete Edit
	2019-08-10	Security: Rahim Sarder	017654545439	Shukrabad 43/2,Dhaka-1207	Shukrabad 43/2,Dhaka-1207	198000667700	Delete Edit
	2019-08-17	Security: Khan	017654545439	Shukrabad 43/2,Dhaka-1207	Shukrabad 43/2,Dhaka-1207	1980006673330	Delete Edit
	2019-08-22	Driver: Sumon Kazi	017654545439	Shukrabad 43/2,Dhaka-1207	Shukrabad 43/2,Dhaka-1207	00	Delete Edit
	2019-08-05	Mithu Hasan	017654545439	Shukrabad 43/2,Dhaka-1207	Shukrabad 43/2,Dhaka-1207	19770667700	Delete Edit

Figure 51 : 9.14 Employee info page

8.15 Manager dashboard page

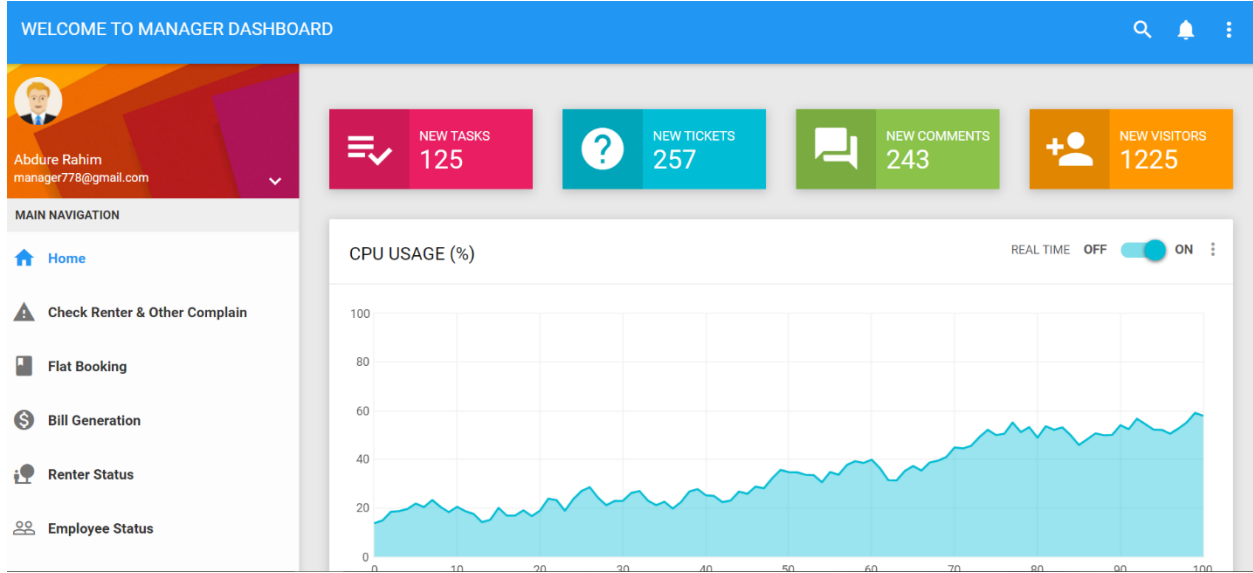


Figure 52 : 9.15 Manager dashboard page

8.16 Renter dashboard page

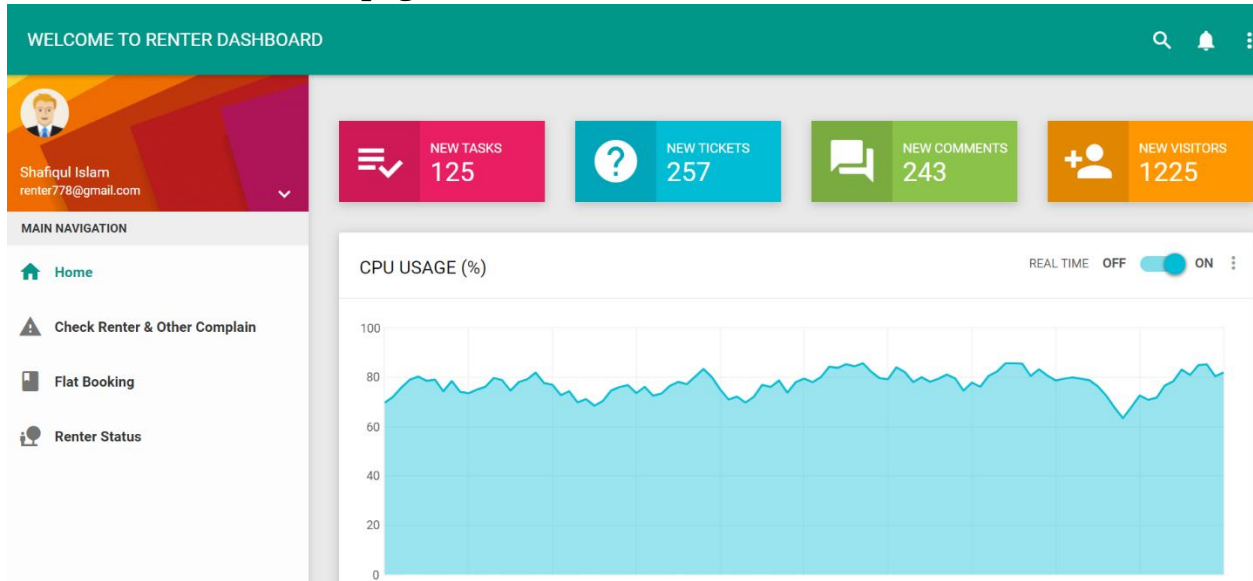


Figure 53 : 9.16 Renter dashboard page

8.17 Flat booking page

Flat	Image	Type	Description	Facilities	Agreement	Rent	Action
201A		Family	Two bedroom One diningroom Two bathroom One kitchen Two belkony	24 hour electricity supply No water & gas problem Strong security	No guest allow Don't waste water and gas Don't make noise Pay in 5 days envey month	6000	Booking
201B		Bachelor/Students	Two bedroom One diningroom Two bathroom One kitchen Two belkony	24 hour electricity supply No water & gas problem Strong security	No guest allow Don't waste water and gas Don't make noise Pay in 5 days envey month	6000	Booking
201C		Family	Two-bedroom One dining room Two bathroom One kitchen Two balcony	24-hour electricity supply No water & gas problem Strong security	No guest allow Don't waste water and gas Don't make noise Pay in 5 days every month	12000	Booking
301A		Family	Two-bedroom One dining room Two bathroom One kitchen Two balcony	24-hour electricity supply No water & gas problem Strong security	No guest allow Don't waste water and gas Don't make noise Pay in 5 days every month	7600	Booking
301B		Bachelor	Two-bedroom One dining room Two bathroom One kitchen Two balcony	24-hour electricity supply No water & gas problem Strong security	Two-bedroom One dining room Two bathroom One kitchen Two balcony	6700	Booking

Figure 54 : 9.17 Flat booking page

8.18 Booking request send page

201A		Family	Two bedroom One diningroom Two bathroom One kitchen Two belkony	24 hour electricity supply No water & gas problem Strong security	No guest allow Don't waste water and gas Don't make noise Pay in 5 days envey month	6000	Booking
201B		Bachelor/Students	Two bedroom One diningroom Two bathroom One kitchen Two belkony	24 hour electricity supply No water & gas problem Strong security	No guest allow Don't waste water and gas Don't make noise Pay in 5 days envey month	6000	Booking
201C		Family	Two-bedroom One dining room Two bathroom One kitchen Two balcony	24-hour electricity supply No water & gas problem Strong security	No guest allow Don't waste water and gas Don't make noise Pay in 5 days every month	12000	Booking
301A		Family	Two-bedroom One dining room Two bathroom One kitchen Two balcony	24-hour electricity supply No water & gas problem Strong security	No guest allow Don't waste water and gas Don't make noise Pay in 5 days every month	7600	Booking
301B		Bachelor	Two-bedroom One dining room Two bathroom One kitchen Two balcony	24-hour electricity supply No water & gas problem Strong security	Two-bedroom One dining room Two bathroom One kitchen Two balcony	6700	Booking

Booking Request X

Name

Contract

Flat

Comment

Close
Send Request

Figure 55 : 9.18 Booking request send page

8.19 Complain page

Tenant ▾

Complain					
Date	Name	Class	Contract	Problem	Action
2020-11-02	Mithu Hasan	Tenant	01656-654543	Gas line Problem	Delete Edit
2020-11-03	Jasim	Driver	01956-654543	My dress is needed	Delete Edit
2020-11-26	Rupa Khan	Tenant	01676-654543	Water line Problem	Delete Edit
2020-11-17	Rubel Islam	Tenant	01856-654543	Electricity line problem	Delete Edit
2020-11-02	Mithu Hasan	Tenant	01656-654543	Gas line Problemnbnvhn	Delete Edit

1 2

Figure 56 : 9.19 Complain page

8.20 Security dashboard page

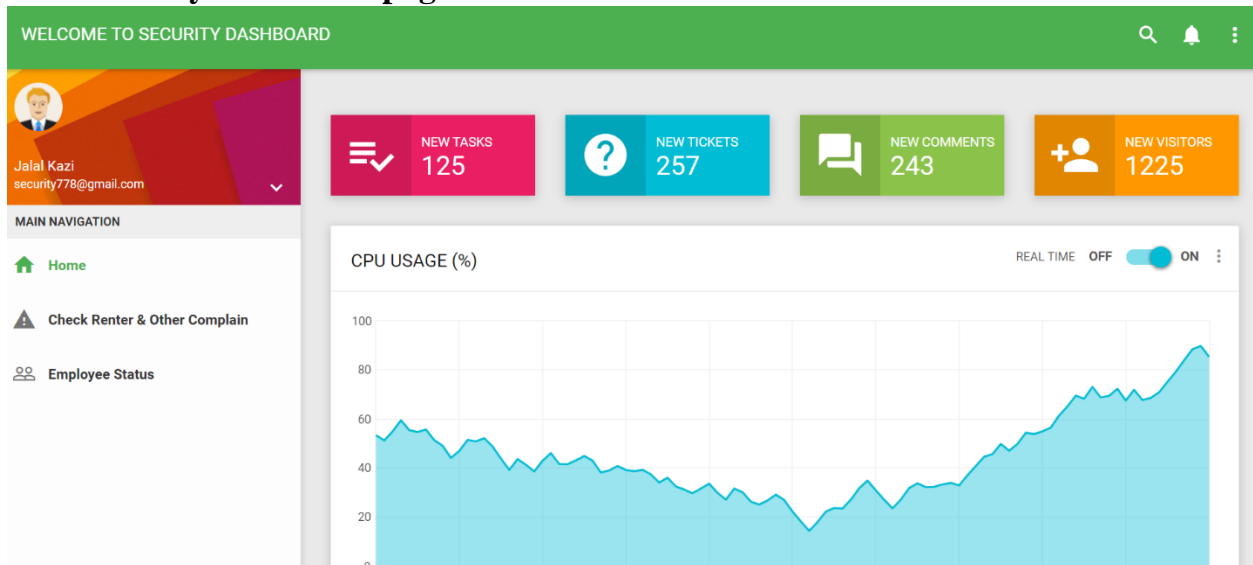


Figure 57 : 9.20 Security dashboard page

8.21 Mechanic dashboard page

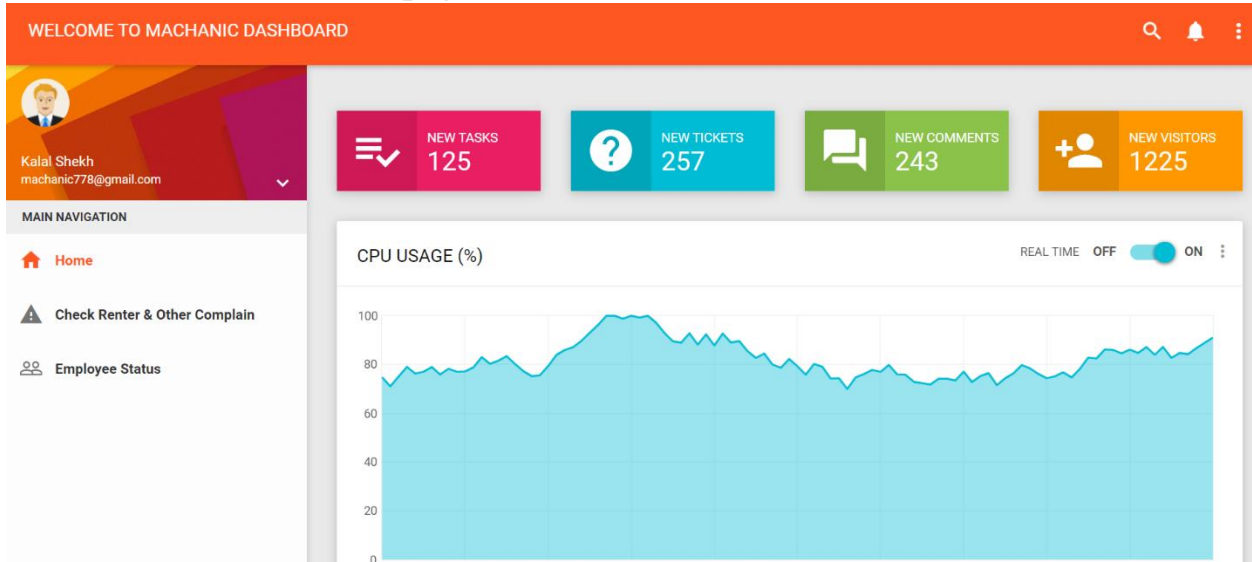


Figure 58 : 9.21 Security dashboard page

8.22 Driver dashboard page

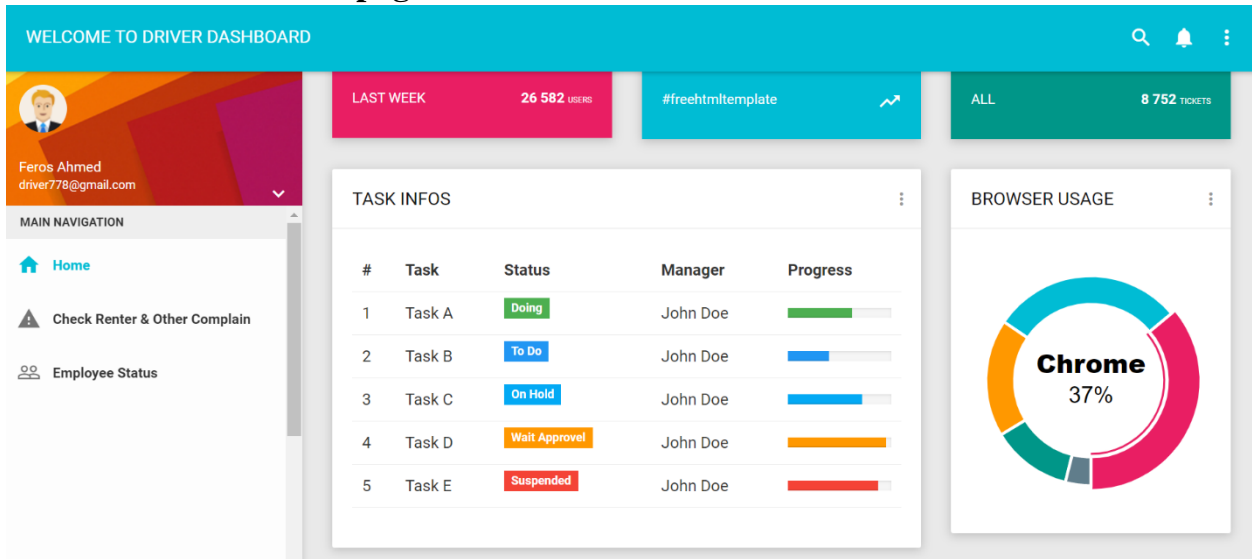


Figure 59 : 9.22 Security dashboard page

8.23 Logout page

Admin ▾

Logout





GasBill	ServiceCharge	Tax	Other Cost	Action
7657568	457568	8777	600	 
7600	10000	800	7000	 

Figure 60 : 9.23 Logout page

9 Chapter

Conclusion

A document will be provided that contains the full details of the code and the website. The main outcome of the application is to maintain records of all users, billing, details of flat availability, and booking. Renter and another user without Admin comment on their problems by select the problem category. Landlord's current financial position with the home. This will be an effective way to manage the home. Every landlord can use this system.

10 Chapter

10.1 Limitations

- ✓ Those people can't be benefited who are not eligible.
- ✓ Limited knowledge about technology.
- ✓ Time shortage.
- ✓ Limited knowledge about process to build complete project.

10.2 Obstacles & Achievements

Day by day technologies are updated. It is quite tough to work with new technology but not so hard. New packages and new library functions are added regularly. So, I feature there are some technology can be eliminating and the project can be implemented with new technology. There are too many errors while I am developing the system.

10.3 Future Scope

Every project has some future scope so that in near future the application adds some new feature

- ✓ Paperless work
- ✓ Increase Platform
- ✓ Getting visa and credit card

Plagiarism Report

2/2/2021

Turnitin

Turnitin Originality Report					
Processed on: 02-Feb-2021 10:58 +06 ID: 1499673005 Word Count: 7455 Submitted: 1					
163-35-1784 By Md. Mithu Hasan	<table border="1"><tr><td>Similarity Index</td><td>9%</td></tr><tr><td>Similarity by Source</td><td>Internet Sources: 5% Publications: 1% Student Papers: 7%</td></tr></table>	Similarity Index	9%	Similarity by Source	Internet Sources: 5% Publications: 1% Student Papers: 7%
Similarity Index	9%				
Similarity by Source	Internet Sources: 5% Publications: 1% Student Papers: 7%				

1% match (Internet from 06-Jan-2020) http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/3547/P13646%20%2824%25%29.pdf?isAllowed=y&sequence=1
1% match (Internet from 15-Mar-2020) http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/3553/P13659%20%2829%25%29.pdf?isAllowed=y&sequence=1
1% match (Internet from 01-Apr-2020) https://www.slideshare.net/RaihanMahmud5/remote-doctor-project-report
1% match (student papers from 21-Jan-2014) Submitted to University of Greenwich on 2014-01-21
1% match (student papers from 06-Apr-2019) Submitted to Gujarat Technological University on 2019-04-06
< 1% match (student papers from 16-May-2016) Submitted to University of Westminster on 2016-05-16
< 1% match (student papers from 15-Jun-2011) Submitted to University of Teesside on 2011-06-15
< 1% match (student papers from 15-Oct-2015) Submitted to Study Group Australia on 2015-10-15
< 1% match (student papers from 24-Apr-2013) Submitted to University of Greenwich on 2013-04-24
< 1% match (Internet from 09-Oct-2018) http://dspace.library.daffodilvarsity.edu.bd:8080/bitstream/handle/20.500.11948/3070/FinalDefenceDocumentation_1435-583%2c141-35-593.pdf?isAllowed=y&sequence=1
< 1% match (student papers from 21-Nov-2012) Submitted to IAT Nairobi (ncc) on 2012-11-21
< 1% match (student papers from 03-May-2019) Submitted to HELP UNIVERSITY on 2019-05-03
< 1% match (Internet from 05-Jun-2020) http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/3927/P148856%20%2828_%29.pdf?isAllowed=y&sequence=1
< 1% match (student papers from 01-Jun-2011) Submitted to Middle East College of Information Technology on 2011-06-01
< 1% match (student papers from 30-Oct-2017) Submitted to Gavatri Vidya Parishad College of Engineering (Autonomous) on 2017-10-30
< 1% match (student papers from 05-May-2017) Submitted to Glyndwr University on 2017-05-05
< 1% match (student papers from 28-May-2010) Submitted to University of Luton on 2010-05-28

https://www.turnitin.com/newreport_printview.asp?eq=1&eb=1&esm=10&oid=1499673005&sid=0&n=0&m=2&svr=22&r=0.39066145033386235&lang... 1/11