

islington college
(इस्लिंग्टन कलेज)

CS6PO5NI, Final Year Project Report – 2019/20

Waste Management System

Student Name: Pranesh Chhetri

London Met ID: 17031141

College ID: NP01CP4A170223

Internal Supervisor: Subeksha Shrestha

External Supervisor: Ishwor Shrestha

Assignment Due Date: 5th June 2020

Assignment Submission Date: 10th June 2020

Words Count: 11834 words

Acknowledgement

I do have to express my collective gratitude towards my supervisors *Mr. Ishwor Shrestha and Mrs. Subeksha Shrestha*. I say a ‘big thanks’ not only for their guidance and patience during this project but also for their mentoring throughout all my brief encounters with the research world in the past.

Special acknowledgements to my module leader *Mr. Monil Adhikari* whose exemplary guidance, valuable feedbacks and suggestions greatly influenced my formal approach towards the project.

I would also like to thank *Islington College* for providing necessary resources for the continuous progress of this project. I am thankful and fortunate enough to get constant encouragement, support and guidance from all the teaching staffs of computing who helped me in completing this project.

Summary/Abstract:

This report provides the overview in the progress of the final year project of **Waste Management System** based on mobile and web. This project is a react-native based mobile application that allows users to buy and sell precious different kinds of wastes, raw materials, and waste made products. It has a management tool or admin console which helps to control or boost the mobile application.

This report includes the detail description of the progress of the project. This project is divided into five different chapters which are Introduction, Background, Development, Testing and Analysis, and Conclusion. Introduction section has the brief information of the topic whereas background/literature review has detail information of which technology and methodology used and it also includes the review and analysis of similar projects.

Development include the progress of the development of the project. This section also includes the results of the survey which is taken before and after development. This section also addresses the features of both web and mobile applications.

Testing and analysis include the planning before testing, types of testing approaches, or methodologies used. A whole system testing will be documented in this section. In this section, all the individual modules, APIs, or integrated modules are tested. This section also includes the analysis of the testing.

A conclusion is a final section of the report which briefly summarizes the main subject and purpose of the project. This section includes the social, ethical and legal issues raised by the system and how social, ethical and legal system is. This section also includes limitations of the project and future possibilities of the project.

Table of Contents

1 INTRODUCTION	1
<i>1.1 PROJECT DESCRIPTION.....</i>	<i>1</i>
<i>1.2 CURRENT SCENARIO.....</i>	<i>1</i>
<i>1.3 PROBLEM DOMAIN.....</i>	<i>2</i>
<i>1.4 PROJECT AS A SOLUTION</i>	<i>5</i>
<i>1.5 AIMS AND OBJECTIVES</i>	<i>6</i>
<i>1.6 STRUCTURE OF THE REPORT.....</i>	<i>7</i>
<i>1.6.1 Background.....</i>	<i>7</i>
<i>1.6.2 Development</i>	<i>7</i>
<i>1.6.3 Testing and Analysis</i>	<i>7</i>
<i>1.6.4 Conclusion</i>	<i>7</i>
2 BACKGROUND	8
<i>2.1 ABOUT THE END-USERS</i>	<i>8</i>
<i>2.2 UNDERSTANDING THE SOLUTION</i>	<i>9</i>
<i>2.3 SIMILAR PROJECTS</i>	<i>10</i>
<i>2.3.1 Mobile Application</i>	<i>10</i>
<i>2.3.2 Web application</i>	<i>12</i>
<i>2.4 COMPARISON WITH SIMILAR PROJECTS.....</i>	<i>15</i>
3 DEVELOPMENT	16
<i>3.1 CONSIDERED METHODOLOGIES</i>	<i>16</i>
<i>3.2 SELECTED METHODOLOGY</i>	<i>16</i>
<i>3.3 PHASES OF METHODOLOGY.....</i>	<i>18</i>
<i>3.3.1 Backlog</i>	<i>18</i>
<i>3.3.2 Design and Prototyping.....</i>	<i>18</i>
<i>3.3.3 Software Development.....</i>	<i>18</i>
<i>3.3.4 Testing</i>	<i>18</i>
<i>3.3.5 Deployment</i>	<i>18</i>
<i>3.4 SURVEY RESULTS</i>	<i>19</i>
<i>3.4.1 Pre-survey Results.....</i>	<i>19</i>
<i>3.4.2 Post-survey Results.....</i>	<i>20</i>
<i>3.5 REQUIREMENT ANALYSIS</i>	<i>21</i>
<i>3.6 DESIGN</i>	<i>21</i>

3.6.1 Wireframes.....	21
3.6.2 Use Case Diagram (Web Application).....	35
3.6.3 High Level Use Case Description (Web Application).....	36
3.6.4 Use Case Diagram (Mobile Application).....	40
3.6.5 High Level Use Case Description (Mobile Application)	41
3.6.6 Level 2 Data Flow Diagram.....	43
3.6.7 ER Diagram.....	44
3.7 CORE DEVELOPMENT FEATURES IMPLEMENTATION	45
3.7.1 Mobile Application Features.....	47
3.7.2 Web Application Features	48
4 TESTING AND ANALYSIS.....	49
4.1 TEST PLAN	49
4.2 UNIT TESTING.....	50
4.2.1 Web Application	50
4.2.2 Mobile Application	74
4.3 INTEGRATION TESTING	78
4.3.1. Mobile application sign-up (customer).....	78
4.3.2 Mobile application – view products list (customer).....	82
4.3.3 Order products (customer)	84
4.3.4 Waste collection schedule (customer)	86
4.3.5 Shopping history (customer).....	88
4.3.6 View customer's waste collection schedule (superhero)	89
4.3.7 Assign employees – waste-collector (superhero)	90
4.3.8 Assign employees – delivery-person (superhero)	92
4.3.9 View orders (superhero)	94
4.3.10 Collect waste (waste collector).....	94
4.3.11 Deliver products (delivery person)	97
4.3.12 Profile – waste containers details (customer)	98
4.3.13 Push notification (customer)	99
4.3.14 Push notification (waste collector)	102
4.3.15 Push notification (delivery person)	104
4.4 CRITICAL ANALYSIS	106
5 CONCLUSION.....	107
5.1 LEGAL, SOCIAL AND ETHICAL ISSUES	108

<i>5.1.1 Legal Issues</i>	108
<i>5.1.2 Social Issues</i>	108
<i>5.1.3 Ethical Issues</i>	108
<i>5.2 LIMITATIONS</i>	109
<i>5.3 FUTURE WORK</i>	109
6 REFERENCES	110
7 BIBLIOGRAPHY	111
8 APPENDIX	112
<i>8.1 APPENDIX A: PRE-SURVEY</i>	112
<i>8.1.1 Pre-Survey Form</i>	112
<i>8.1.2 Pre-Survey Result</i>	116
<i>8.2 APPENDIX B: POST-SURVEY</i>	122
<i>8.2.1 Post-Survey Form</i>	122
<i>8.2.2 Post-Survey Result</i>	126
<i>8.3 APPENDIX C: SAMPLE CODES</i>	134
<i>8.3.1 Sample Code of the UI</i>	134
<i>8.3.2 Sample Code for the Automation Script</i>	134
<i>8.4 APPENDIX D: DESIGNS</i>	135
<i>8.4.1 Gantt Chart</i>	135
<i>8.4.2 Work Breakdown Structure</i>	136
<i>8.4.3 Expanded Use Case Description (Web Application)</i>	142
<i>8.4.4 Expanded Use Case Description (Mobile Application)</i>	149
<i>8.5 APPENDIX E: TESTING AND ANALYSIS</i>	154
<i>8.5.1 Compatibility Testing</i>	154
<i>8.5.2 Acceptance Testing</i>	162

Table of Figures

<i>Figure 1 - Increase of global plastic production, measured in tons per year, from 1950 through to 2015 (Ritchie & Roser, 2018)</i>	2
<i>Figure 2 - Estimated share of global plastic waste that is discarded, recycled or incinerated from 1980 through to 2015 (Ritchie & Roser, 2018)</i>	3
<i>Figure 3 - Composition of household waste in the 58 municipalities (Asian Development Bank, 2013)</i>	4
<i>Figure 4 - Review on recycling coach mobile app - events and collection schedule (Google Play, 2020)</i>	10
<i>Figure 5 - Review on recycling coach mobile app - drop off location (Google Play, 2020)</i>	11
<i>Figure 6 - Review on eBay - add product (eBay, 2020)</i>	12
<i>Figure 7 - Review on eBay - add product details (eBay, 2020)</i>	13
<i>Figure 8 - Review on x-cart - dashboard and all (X-Cart , 2020)</i>	14
<i>Figure 9 - Kanban board for Laravel Authentication</i>	17
<i>Figure 10 - Loading and Login page for customers</i>	21
<i>Figure 11 - Forget password page for customers</i>	22
<i>Figure 12 - Register and Choose waste containers page for customers</i>	23
<i>Figure 13 - Register address details and set home location page for customers</i>	24
<i>Figure 14 - Dashboard and product details page for customers</i>	25
<i>Figure 15 - Shopping cart and waste collection time for customers</i>	26
<i>Figure 16 - Notifications and user profile page for customers</i>	27
<i>Figure 17 - Blogs and Articles page for customers</i>	28
<i>Figure 18 - Loading and log in page for waste collectors</i>	29
<i>Figure 19 - Customer list and customer detail page for waste collectors</i>	30
<i>Figure 20 - Map view and notification page for waste collectors</i>	31
<i>Figure 21 - Loading and log in page for delivery person</i>	32
<i>Figure 22 - Dashboard and order detail page for delivery persons</i>	33
<i>Figure 23 - Map view and notifications page for delivery persons</i>	34
<i>Figure 24 - User case diagram - web application</i>	35
<i>Figure 25 - Use case diagram - mobile application</i>	40
<i>Figure 26 - Level 2 data flow diagram of the system</i>	43
<i>Figure 27 - ER Diagram of the system</i>	44
<i>Figure 28 - Set waste collection schedule page</i>	45
<i>Figure 29 - Queue is running in the background</i>	46
<i>Figure 30 - Notifying users manually using this command</i>	46
<i>Figure 31 - Test case TC1 when user enters valid username and password</i>	50
<i>Figure 32 - Test case TC1 with valid username and password user successfully log into the application</i>	51

Figure 33 - Test case TC2 showing error message when user enters invalid username and password	51
Figure 34 - Test case TC3 showing error message when user enters invalid username and password	52
Figure 35 - Test case TC4 when user sign out from an application.....	53
Figure 36 - Test case TC5 when a user tries to access the application after signing out from the system.....	54
Figure 37 - Test case TC5 showing that system redirects the log in page.....	54
Figure 38 - Test case TC6 when user add category with details.....	55
Figure 39 – Test case TC6 where category wood is added in the list	56
Figure 40 - Test case TC7 when a user doesn't give inputs	56
Figure 41 - Test case TC8 when a user edits category.....	57
Figure 42 - Test case TC8 where category name to woods	57
Figure 43 - Test case TC9 when user delete a category wood.....	58
Figure 44 - Test case TC9 where category wood is deleted from the list	58
Figure 45 - Test case TC10 when user add recycled product with details	59
Figure 46 - Test case TC10 where recycled product Rough Sawn Shelf 2x4 is added.....	60
Figure 47 - Test case TC11 showing error message where invalid inputs were filled.....	60
Figure 48 - Test case TC12 where product name edited to Rough Sawn Shelf	61
Figure 49 - Test case TC12 where product name updated to Rough Sawn Shelf	61
Figure 50 - Test case TC13 when user delete a recycled product.....	62
Figure 51 - Test case TC13 recycled product Rough Sawn Shelf is deleted	62
Figure 52 - Test case 14 when user adds the waste type.....	63
Figure 53 - Test case TC14 where Plastic waste type is added.....	64
Figure 54 - Test case TC15 when user edit container color to blue	64
Figure 55 - Test case TC15 where waste type's container color updated to blue	65
Figure 56 - Test case TC16 when a user deletes a waste type	65
Figure 57 - Test case TC16 where plastic waste type was deleted.....	65
Figure 58 - Test case TC17 when user adds a waste container.....	66
Figure 59 - Test case TC17 where medium size waste container added	67
Figure 60 - Test case TC18 when a user doesn't give an input.....	67
Figure 61 - Test case TC19 when user edits a waste container size to large	68
Figure 62 - Test case TC19 where waste container size updated to large	68
Figure 63 - Test case TC20 when user deletes a waste container	69
Figure 64 - Test case TC20 where waste container is deleted.....	69
Figure 65 - Test case TC21 when a user adds an employee	70
Figure 66 - Test case 21 when a user added an employee.....	71
Figure 67 - Test case TC22 when a user doesn't give input.....	71

Figure 68 - Test case TC23 when a user edits employee name to Bill Bahadur Gates	72
Figure 69 - Test case TC23 where an employee name is updated to Bill Bahadur Gates.....	72
Figure 70 - Test case 24 when a user deletes an employee	73
Figure 71 - Test case TC24 where a user deleted an employee.....	73
Figure 72 - Test case T25 when a user enters a valid username and password.....	74
Figure 73 - Test case TC26 showing error message when a user enters invalid or empty data	75
Figure 74 - Test case TC27 where a user logged out from application	76
Figure 75 - Test case TC28 where customers can read articles and blogs related to waste management and recycling.....	77
Figure 76 - Test case TC29 when a user signs up with valid data	79
Figure 77 - Test case TC29 where a user successfully created	79
Figure 78 - Test case TC29 where a customer data is shown in the admin panel.....	80
Figure 79 - Test case TC30 showing an error message when a user enters a invalid or empty data	80
Figure 80 - Test case TC31 where password and confirm password are not same and TC32 when a user enters the username which is already exists.....	81
Figure 81 - Test case TC32 products list in the web application.....	82
Figure 82 - Test case TC32 all the products are displayed	83
Figure 83 - Test case TC33 where note was searched from the product list	83
Figure 84 - Test case TC35 when a user ordered a recycled product	84
Figure 85 - Test case TC35 where an ordered recycled product is displayed in the admin panel	85
Figure 86 - Test case TC36 when a user ordered a waste container.....	85
Figure 87 - Test case TC36 where an ordered waste container is displayed in the admin panel	86
Figure 88 - Test case TC37 where a superhero set waste collection schedule to streets	86
Figure 89 - Test case TC37 where a waste collection for New Baneshwor street was displayed.....	87
Figure 90 - Test case T38 where a customer can view the shopping history	88
Figure 91 - Test case TC39 where superhero can view the customer's waste collection schedule according to the street.....	89
Figure 92 - Test case TC40 where a superhero search for street to assign waste collectors.....	90
Figure 93 - Test case TC40 where a superhero assigned street (New Baneshwor) to the waste collector (Tim Cook & Elon Musk)	91
Figure 94 - Test case TC40 where a customer's list of New Baneshwor street is displayed in the waste collector mobile app	91
Figure 95 - Test case TC41 where a superhero assigns orders to the delivery person (Donald Trump)	92
Figure 96 - Test case TC41 where superhero assigned orders to the delivery person (Donald Trump).....	93
Figure 97 - Test case TC41 where a list of order assigned to a delivery person (Donald Trump) is shown	93

Figure 98 - Test case TC42 where the pending order list is displayed	94
Figure 99 - Test case TC43 where customer's list is displayed, and garbage weight collected is updated	95
Figure 100 - Test case TC44 when a waste collector input more weight than waste container capacity.....	96
Figure 101 - Test case TC45 where a delivery person successfully delivered a product (Bucket) of Jack Ma.....	97
Figure 102 - Test case TC45 where customer's order details is displayed in the admin panel.....	97
Figure 103 - Test case TC46 showing waste containers and total waste collected details	98
Figure 104 - Test case TC47 where customer (Jack Ma) log in to the mobile app	99
Figure 105 - Test case TC47 where new expo toke for Jack Ma (customer) is generated.....	99
Figure 106 - Test case TC48 where waste collection schedule for New Baneshwor street is set to 9:12 pm.....	100
Figure 107 - Test case TC48 where waste collection schedule for Jack Ma is set for 21:12 pm	100
Figure 108 - Test case TC48 where a customer (Jack Ma) received a notification before one hour of waste collection time.....	101
Figure 109 - Test case TC49 where waste collectors assigned for New Baneshwor are Tim Cook and Elon Musk ...	102
Figure 110 - Test case TC49 where waste collection day for New Baneshwor street is Sunday	102
Figure 111 - Test case TC49 where a waste-collector (Tim Cook) received a notification on Sunday.....	103
Figure 112 - Test case TC50 where a customer made order of Wireless Bluetooth Headphone.....	104
Figure 113 - Test case TC50 where a superhero assigned an order to a delivery person (Donald Trump).....	104
Figure 114 - Test case TC50 where a delivery person (Donald Trump) received a notification for a new order	105
Figure 115 - Queries related to general details of users	112
Figure 116 - Queries related to household waste generation and disposal.....	113
Figure 117 – Queries related to garbage collection services.....	114
Figure 118 - Queries related to recycling	115
Figure 119 – Respondent's locations in Kathmandu Valley	116
Figure 120 - Total members living in the house	116
Figure 121 - Consequences of waste in the respondent area.....	117
Figure 122 - Storage method using and how often does it gets emptied in a week	117
Figure 123 - Separation of waste at home.....	118
Figure 124 - Separation of waste if waste collection service provider tells to separate	118
Figure 125 - Regular garbage collection services in the respondent's area.....	119
Figure 126 - Garbage collection services in the respondent's area if they collect regular	119
Figure 127 - Waste collection services using respondents	120
Figure 128 - Type of waste collection services using respondents	120
Figure 129 - Type of recycled materials respondent normally use.....	121
Figure 130 - Respondents opinion for effective waste management.....	121
Figure 131 - Queries related to general details.....	122

<i>Figure 132 –Queries related to web application</i>	122
<i>Figure 133 - Queries related to web application II.....</i>	123
<i>Figure 134 - Queries related to mobile application.....</i>	124
<i>Figure 135 - Queries related to mobile application II.....</i>	125
<i>Figure 136 - Web application design.....</i>	126
<i>Figure 137 - Navigation between pages</i>	126
<i>Figure 138 - Catalog management feature.....</i>	127
<i>Figure 139 - Order management feature.....</i>	127
<i>Figure 140 - User management feature.....</i>	128
<i>Figure 141 - Extra data in the dashboard</i>	128
<i>Figure 142 - Settings of the web application.....</i>	129
<i>Figure 143 – Suggestions to improve the admin panel.....</i>	129
<i>Figure 144 - Mobile application design.....</i>	130
<i>Figure 145 - Functionalities of the mobile application.....</i>	130
<i>Figure 146 – Order products from a mobile app</i>	131
<i>Figure 147 - Garbage collection schedule feature</i>	131
<i>Figure 148 - Employees (waste-collectors and delivery persons) app.....</i>	132
<i>Figure 149 - Overall experience of the mobile app</i>	132
<i>Figure 150 - Recommendation of this app to others.....</i>	133
<i>Figure 151 - Respondents opinion for mobile app</i>	133
<i>Figure 152 - Project Gantt chart</i>	135
<i>Figure 153 - Web application work-break down structure</i>	136
<i>Figure 154 - Web application work-breakdown structure I</i>	137
<i>Figure 155 - Web application work-breakdown structure II</i>	138
<i>Figure 156 - Mobile application work-breakdown structure.....</i>	139
<i>Figure 157 - Mobile application work-breakdown structure I.....</i>	140
<i>Figure 158 - Mobile application work-breakdown structure II.....</i>	141
<i>Figure 159 – Web application compatibility testing on windows operating system.....</i>	154
<i>Figure 160 – Web application compatibility testing on macOS Catalina 10.15.4</i>	155
<i>Figure 161 – Web application compatibility testing on google chrome browser.....</i>	155
<i>Figure 162 – Web application compatibility testing on safari browser</i>	156
<i>Figure 163 – Web application compatibility testing on opera web browser</i>	156
<i>Figure 164 – Web application compatibility testing on Microsoft edge</i>	157
<i>Figure 165 – Web application compatibility testing on iPhone and android phone</i>	157
<i>Figure 166 – Web application compatibility testing on iPad</i>	158

<i>Figure 167 – Mobile application compatibility testing on iPhone 11 Pro Max and iPhone SE (2nd generation)</i>	159
<i>Figure 168 - Mobile application compatibility testing on iPad Pro (9.7 inch)</i>	160
<i>Figure 169 - Mobile application compatibility testing on Android Phone (Samsung).....</i>	161

Table of Tables

<i>Table 1 - Test cases for admin login.....</i>	50
<i>Table 2 - Test cases for admin logout</i>	53
<i>Table 3 - Test cases for catalog management - categories.....</i>	55
<i>Table 4 - Test cases for catalog management - recycle products</i>	59
<i>Table 5 - Test cases for catalog management - waste types</i>	63
<i>Table 6 - Test cases for catalog management - waste container</i>	66
<i>Table 7 - Test cases for user management – employees.....</i>	70
<i>Table 8 - Test case for Mobile application login (Customer, Waste Collector and Delivery Person).....</i>	74
<i>Table 9 – Test case for Mobile application logout (Customer, Waste Collector and Delivery Person).....</i>	76
<i>Table 10 - Test case for Blogs and Articles.....</i>	77
<i>Table 11 - Test case for Mobile application sign-up (customer)</i>	78
<i>Table 12 - Test case for Mobile application – view products list (customer)</i>	82
<i>Table 13 - Test case for Order products (customer).....</i>	84
<i>Table 14 - Test case for Waste collection schedule (customer).....</i>	86
<i>Table 15 - Test case for Shopping history (customer)</i>	88
<i>Table 16 - Test case for View customer's waste collection schedule (superhero)</i>	89
<i>Table 17 - Test case for Assign employees – waste-collector (superhero)</i>	90
<i>Table 18 - Test case for Assign employees – waste-collector (superhero)</i>	92
<i>Table 19 - Test case for View orders (superhero).....</i>	94
<i>Table 20 - Test case for Collect waste (waste collector)</i>	94
<i>Table 21 – Test case for Deliver products (delivery person).....</i>	97
<i>Table 22 - Test case for Profile – waste containers details (customer)</i>	98
<i>Table 23 - Test case for Push notification (customer)</i>	99
<i>Table 24 - Test case for Push notification (waste collector).....</i>	102
<i>Table 25 - Test case for Push notification (delivery person).....</i>	104
<i>Table 26 - Test cases of web application compatibility testing.....</i>	154

1 Introduction

1.1 Project Description

This project is a react-native based mobile application that allows users to buy and sell precious different kinds of wastes, raw materials, and waste made products. It has a management tool or admin console which helps to control or boost the mobile application.

The customer mobile app allows individual households to buy and sell different kinds of wastes and waste made products and to manage waste collection services from a handy smartphone. The waste collector app allows to collect garbage from customers in different location. Similarly, delivery person app allows to deliver order made by customers.

I've chosen agile software development methodology Kanban. I have also used digital project management tool which helps me to visualize work, and limit in progress. Everyone's shifting mobile nowadays, getting things done on the go - communication, purchases, job, etc. Mobile phones are easily accessible to the people. That's why I thought that mobile application would be the better option for this project.

1.2 Current Scenario

Around the world, waste generation rates are rising. In 2016, the worlds' cities generated 2.01 billion tons of solid waste, amounting to a footprint of 0.74 kilograms per person per day. With rapid population growth and urbanization, annual waste generation is expected to increase by 70% from 2016 levels to 3.40 billion tons in 2050 (World Bank Group, 2019).

Managing solid waste is one of the major challenges in urbanization. According to a survey conducted by the *Asian Development Bank* in all 58 municipalities of Nepal in 2012 found that the average municipal solid waste generation was 317 grams per capita per day. This translates into 1,435 tons per day or 524,000 tons per year of municipal solid waste generation in Nepal. Many of these technically and financially constrained municipalities are still practicing roadside waste pickup from the open piles and open dumping, creating major health risks (Asian Development Bank, 2013).

1.3 Problem Domain

Today, the world produces about 300 million metric tons of plastics every year and this is a lot of numbers. It is still growing every year because we keep producing more new plastics. These plastics wind up negatively affecting wildlife habitat, and humans by affecting lands, oceans, and waterways (Opsomer & Pennington, 2020). Marine plastic litter pollution is already affecting more than 800 marine species through ingestion, entanglement, and habitat change. Current estimates show that at last 8 million pieces of plastic are entering the oceans every single day. With 1 in 3 fish caught for human consumption now containing plastic, the question is no longer are we eating plastic but how bad for us is that. In fact, researchers say that less than 10 percent of our plastics actually get recycled, so all the new plastics we make a lot of it ends up in the wrong place. If we still don't think about plastic. Then, by 2050, the oceans will contain more plastic than fish by weight (Mwamba, 2020). Below figures shows that how plastic production is increasing globally from 1950 to 2015.

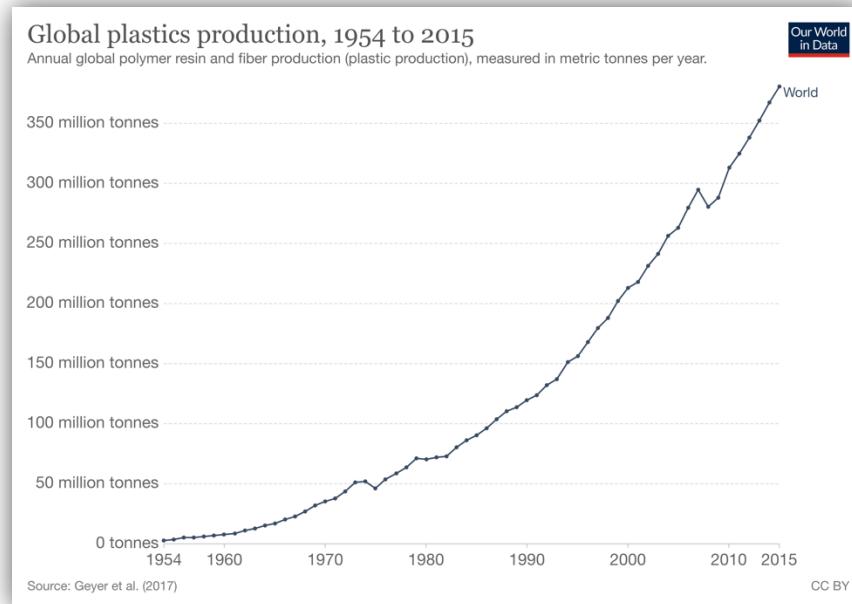


Figure 1 - Increase of global plastic production, measured in tons per year, from 1950 through to 2015 (Ritchie & Roser, 2018)

Sadly, to say that food waste is one of the biggest problems facing mankind today. Between 33 – 80% of all food produced is never eaten, and the value of this wasted food is worth over \$1 trillion. This is not only an economic problem, but it is also an environmental problem (WASTE WISE

PRODUCTS INC., 2020). It takes a landmass larger than China to grow the food each year that is ultimately never eaten, land that has been deforested, species that have been driven to extinction, indigenous population that has been moved, soil that has been degraded, all to produce food that we then just thrown away. The food in landfills decomposes and emits methane, a poisonous greenhouse gas. Globally, methane gas from food waste makes up seven percent of total greenhouse emissions (WASTE WISE PRODUCTS INC., 2020). Prior to 1980, recycling and incineration of plastic was negligible; 100 percent was therefore discarded. From 1980 for incineration, and 1990 for recycling, rates increased on average by about 0.7 percent per year.⁴ In 2015, an estimated 55 percent of global plastic waste was discarded, 25 percent was incinerated, and 20 percent recycled (Ritchie & Roser, 2018).

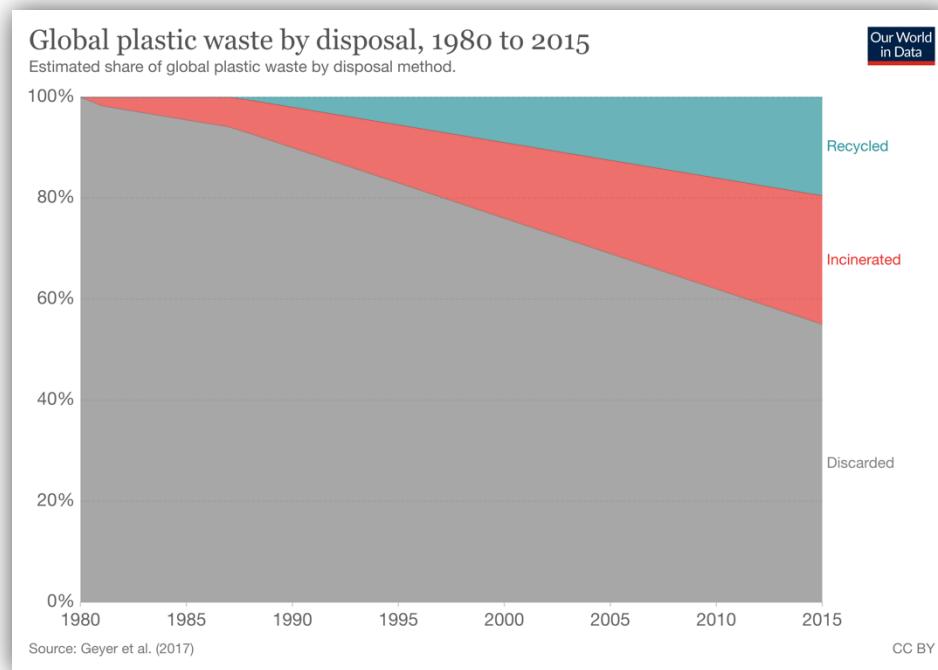


Figure 2 - Estimated share of global plastic waste that is discarded, recycled or incinerated from 1980 through to 2015 (Ritchie & Roser, 2018)

Metals and Steels have been also a big problem in an environment and to human beings. Scrap metal in the landfill is a wasted resource. It is estimated that we consume about 200 billion of beer and soda can every year. Throwing away a single aluminum can waste energy equivalent to the same can filled with gasoline.

In Nepal, urbanization is increasing at an alarming rate putting immense pressure on municipal services, particularly to manage the ever-increasing amounts of waste. At present most of the wastes generated in municipalities are not being adequately managed thereby creating serious health and environmental hazard, particularly in the slum and squatter areas, where the residents have less capacity to pay for better services and are often ignored by the official agencies. Therefore, poor urban settlements are most affected because of indiscriminate dumping and lack of open spaces. The current situation in Nepal is unpleasant. It is estimated that between 800 and 1000 tons of plastics are produced in the Kathmandu Valley each day and tons of recyclable waste still ends up in the Landfills every year (Azoulay, 2020).

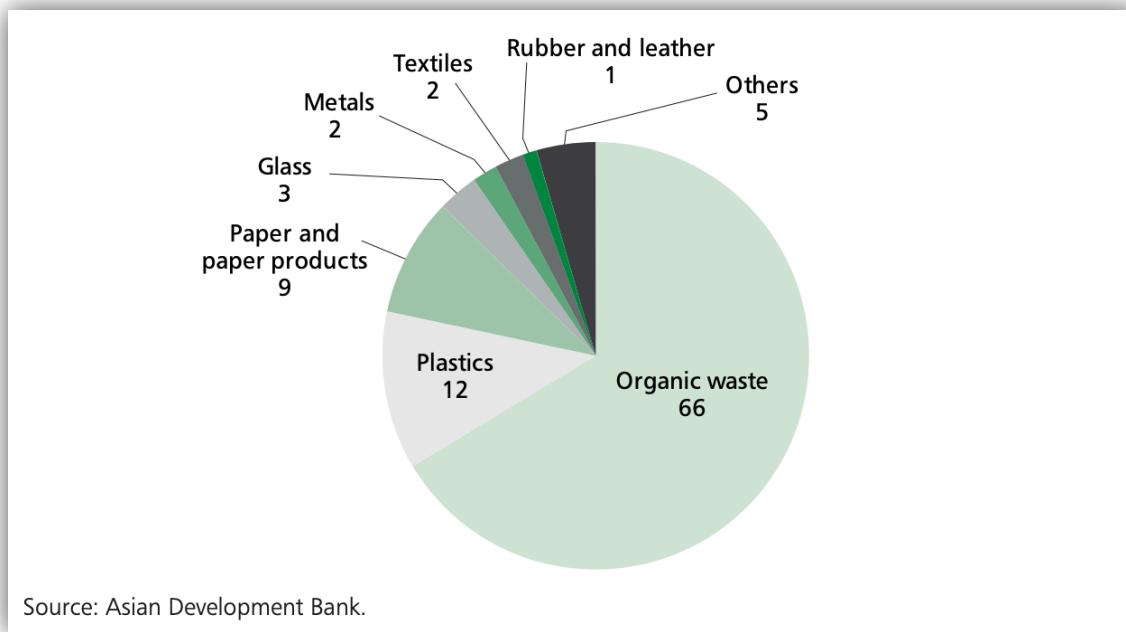


Figure 3 - Composition of household waste in the 58 municipalities (Asian Development Bank, 2013)

The waste composition analysis indicates that the highest waste fraction is organic matter (66%), followed by plastics (12%), paper and paper products (9%), others (5%), and glass (3%). Metal, textiles, rubber and leather each accounted for 2% or less (Asian Development Bank, 2013).

1.4 Project as a Solution

Recycling plastic helps reduce landfill space, helps protect animals, decreased pollution, reduced greenhouse gas emissions, and helps conserve natural resources and energy. Although, almost every kind of metal can be recycled again and again without degradation of properties, currently, only 30 percent of metal is recycled.

Recycling metals can help save up to 75% of the energy. Air pollution is also reduced by almost 90% (Leblanc, 2020). Food waste is composed of organic matter which can be used for composting to make fertilizer. It is an effective and eco-friendly way of disposing of food waste. By using leftovers and other food waste, we can convert these smelly items into a highly organic product rich in nutrients that we can use to grow vegetables or flowers with it.

In order to maximize waste recycling the private sector should be encourage to setup and operate waste recycling and composting facilities. So, I have decided to create a mobile application that allows the customers to buy and sell precious different kinds of wastes, raw materials, and waste made products. This app allows individual households to throw away the right waste, helping them with a simple interface organized by categories and weekly schedule for garbage collection.

This application also offers helpful information about recycling, and waste management. This application also connects waste collectors and delivery persons to customers where waste collectors collects garbage from customers in different location and delivery persons delivers orders made by customers.

1.5 Aims and Objectives

The main goal of this project is to solve environmental problems with the power of technology. This project will not only help in recycling but also in managing the waste. The project primarily focused on developing a mobile application for people to manage and recycle waste. One can easily buy and sell precious different kinds of wastes, raw materials, and waste made products.

Objectives of this project are pointed out below:

- a) Try to solve the global environmental problem with the power of technology.
- b) To make the waste management system process more efficient.
- c) To do the right thing for the customers, reduce their waste and help them to earn money out of trash, and save their time, and help them meet sustainable goals by keeping more trash out of landfills and oceans.
- d) To help to create a more efficient and effective society.

1.6 Structure of the Report

1.6.1 Background

A project background includes an overview of the project's activities and goals. This section also includes what problems and needs will be addressed once it's been implemented successfully. This section also describes the end-user of the project. In this section, the similar features of this project will be compared with other projects.

1.6.2 Development

A development section includes the software methodology used to develop this project and describes the phases of the development. This section also includes the results of the survey which is taken before and after development. This section also addresses the features of both web and mobile applications.

1.6.3 Testing and Analysis

Testing and analysis include the planning before testing, types of testing approaches, or methodologies used. A whole system testing will be documented in this section. In this section, all the individual modules, APIs, or integrated modules are tested. This section also includes the analysis of the testing.

1.6.4 Conclusion

A conclusion is a final section of the report which briefly summarizes the main subject and purpose of the project. This section includes the social, ethical and legal issues raised by the system and how social, ethical and legal system is. This section also includes limitations of the project and future possibilities of the project.

2 Background

This project takes a different approach on a solid waste management system and recycling. This project used react native as a mobile application framework and Laravel as a php web framework. Mobile application allows users to buy and sell precious different kinds of wastes, raw materials, and waste made products where admin console which helps to control or boost the mobile application.

The customer mobile app allows users to buy and sell different kinds of wastes and waste made products and to manage waste collection services. The waste collector app allows to collect garbage from customers in different location.

Similarly, delivery person app allows to deliver orders made by customers. This application helps to manage waste and recycling efficiently and meet the sustainability goals. This application is primarily designed for the local people of Kathmandu.

2.1 About the End-users

This project is mainly designed for individual households/customers, waste collectors and delivery persons. The customer mobile app allows users to sell different kinds of wastes and waste made products. It also helps users to manage waste collection services according to their street. Customers can also read blogs and articles related to waste management and recycling. Customer can also monitor the summary related to waste disposal and history of their purchased orders.

The waste collector app allows waste collector to collect garbage from customers in different location. Similarly, delivery person app allows delivery person to deliver orders made by customers. All the users received notification where customers and waste collectors get notification related to waste collection schedule and delivery person gets notification related to orders which were assigned by superhero.

2.2 Understanding the Solution

The technical terms related to web application are as below:

1. **Blade Template:** a templating engine to design a layout.
2. **Laravel Authentication:** a process of identifying the user credentials.
3. **CRUD Operation:** to create, read, update, and delete operation
4. **Laravel Passport:** to apply authentication over an API for mobile application
5. **jQuery:** to create application functionality that functions smoothly
6. **Ajax:** to send and receive data using ajax request from the server
7. **Laravel Queue and Jobs:** to queue notifications is to create job that sends notifications to the notifiers

The technical terms related to mobile application are as below:

1. **React Native Redux:** to manage state of the application and to implement actions in redux
2. **React Native Axios:** to manage API requests in react-native
3. **Expo:** is a framework and a platform to build a react-native application
4. **Expo Push Notification:** provides access to push notifications and local notifications (scheduling and immediate) related functions.
5. **Expo Constants:** provides system information that remains constant throughout the lifetime of the app's install
6. **React Native Responsive Screen:** to make mobile app responsive

2.3 Similar Projects

This section is divided into two different parts: review on similar mobile application projects and review on similar web application projects.

2.3.1 Mobile Application

2.3.1.1 Recycle Coach

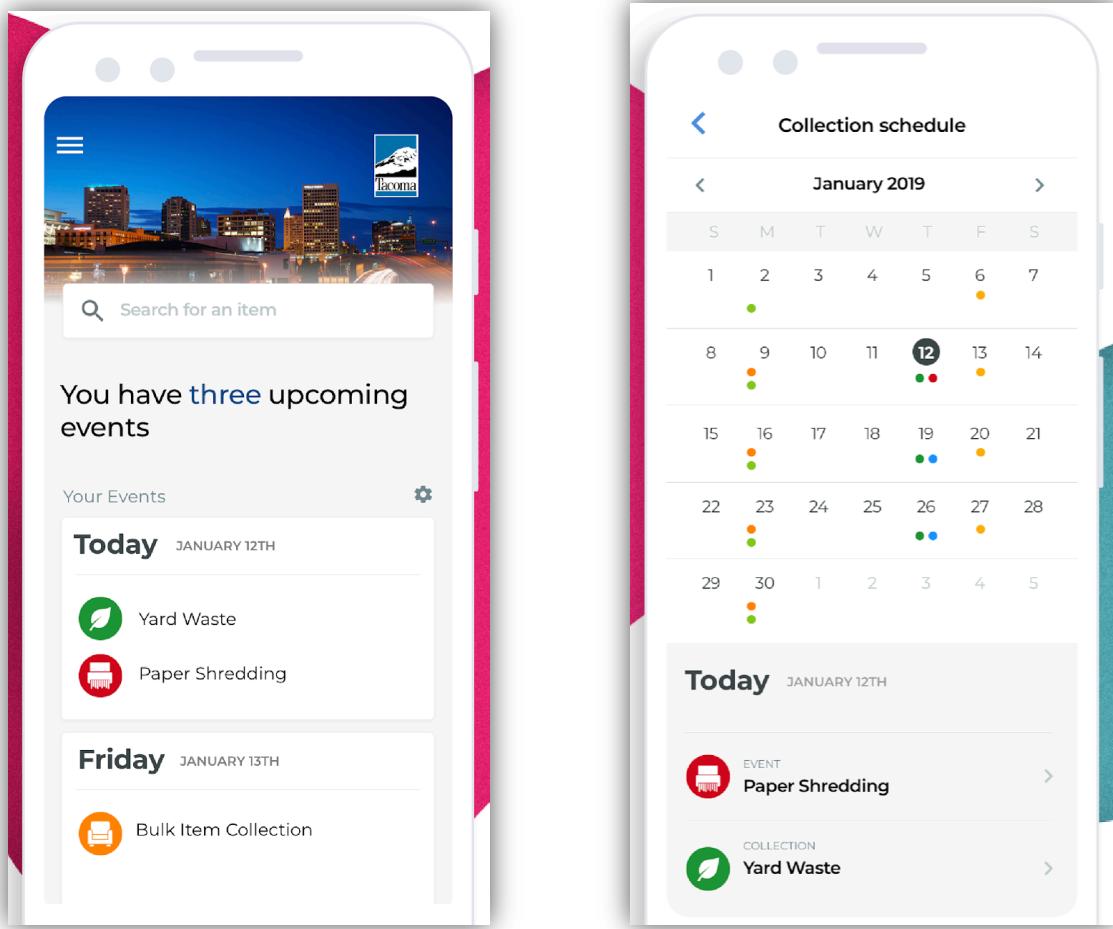


Figure 4 - Review on recycling coach mobile app - events and collection schedule (Google Play, 2020)

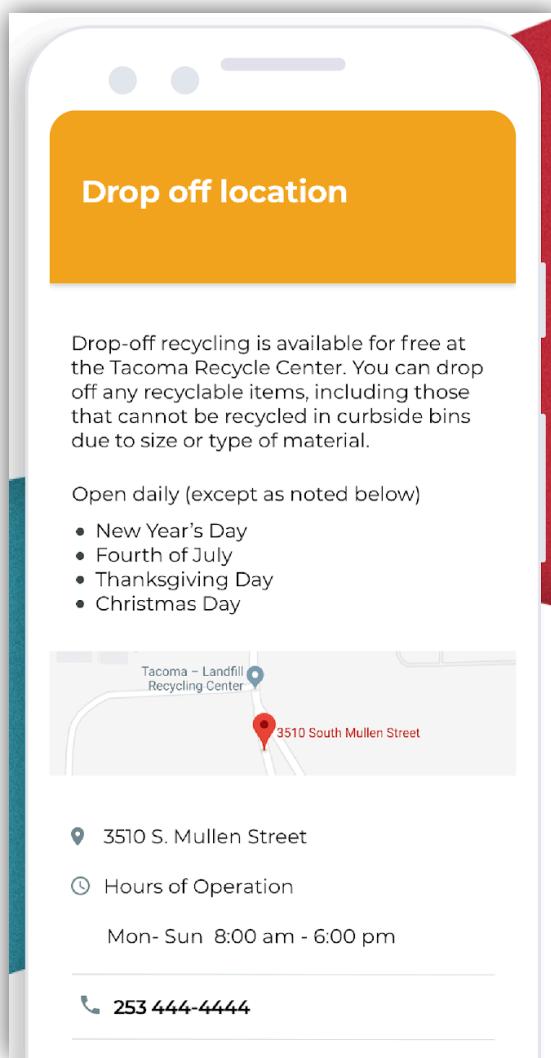


Figure 5 - Review on recycling coach mobile app - drop off location (Google Play, 2020)

Recycle coach is a Central Okanagan's based mobile application where we can get instant access to all waste disposal information for our community. This application is available for both iOS and Android. This application has simplified communication with real-time reporting of missed collections, graffiti, illegal dumping and more. We can get recycling tips and as well as collection updates. We get personalized calendars and reminders through mobile device. This application also helps to get accurate recycling and disposal information for thousands of household items. This application also helps to know about the local waste and recycling events (Recycle Coach, 2020).

2.3.2 Web application

2.3.2.1 eBay

Listing details

*Title [\(i\)](#)

Stand out with a bold title in search results (\$3.00) 80 character(s) left

Subtitle (\$1.00) [\(i\)](#) 55 character(s) left

Second category (fees may apply) [Add a second category](#)

Variations We don't support variations for auction-style listings. To use variations, change the [Format](#) to fixed price.

*Condition [\(i\)](#) -

*Additional photos (0) We recommend adding 3 more photos [Delete all](#) | [Import from web](#)

[Add photos](#)

Add up to 12 photos. We don't allow photos with extra borders, text or artwork.
You can also [copy your photos from a web address](#).

Figure 6 - Review on eBay - add product (eBay, 2020)

The screenshot shows the 'Additional item specifics' section of the eBay product listing form. It is divided into two main sections: 'Required' and 'Additional'.

Required: This section is labeled 'Buyers need these item specifics about your item'. It contains a dropdown menu for 'Brand' with options like 'JBL', 'Yamaha', and 'Electro-Voice'. Below the dropdown, it says 'Frequent: JBL, Yamaha, Electro-Voice'.

Additional: This section is labeled 'Buyers may also be interested in these item specifics'. It includes dropdown menus for 'RMS Power' (with 2.9K searches) and 'Custom Bundle'. There is also a dropdown for 'Modified Item' and a text area for 'California Prop 65 Warning' with a character limit of 800.

Below these fields are dropdown menus for 'Country/Region of Manufacture' and 'Additional item description'.

A button '+ Add custom item specific' is located at the bottom left of the specific section.

Figure 7 - Review on eBay - add product details (eBay, 2020)

eBay Inc. is an American multinational ecommerce corporation based in San Jose, California, that facilities consumer-to-consumer and business-to-consumer sales through its website (eBay Inc., 2020). This is a very big company and has many more features regarding ecommerce. From this application, I have known about the number of fields required while adding products.

2.3.2.2. X-Cart:

Order #	Date	Customer	Payment status	Fulfilment status	Amount
#00005	Jan 12, 2016 16:18	Guest Guest bit-bucket@example.com	Paid	Delivered	\$352.81 Qty: 19
#00004	Jan 12, 2016 16:13	Guest Guest bit-bucket@example.com	Declined	Will Not Deliver	\$38.01 Qty: 1
#00003	Aug 2, 2014 22:58	Guest Guest bit-bucket@example.com	Paid	Delivered	\$158.94 Qty: 7
#00002	Mar 28, 2014 15:24	Guest Guest bit-bucket@example.com	Paid	Delivered	\$88.67 Qty: 5
#00001	Mar 5, 2014 11:45	Guest Guest bit-bucket@example.com	Paid	Delivered	\$18.49 Qty: 1

Search total: \$656.92

Figure 8 - Review on x-cart - dashboard and all (X-Cart , 2020)

X-Cart is an open-source shopping cart demo software. It is an ecommerce website builder that give the full control over every element for online store. This software allows us to view the details of products, customers, and orders. This software helps us in tracking and managing orders and also manage multiple orders at once. From this software, we can also add unlimited products with ease. This software also helps in tracking product inventory one at a time or manage multiple products at once. This is the complete ecommerce application which has almost every feature that should be in the ecommerce application (X-Cart , 2020).

2.4 Comparison with similar projects

Recycle coach is a Central Okanagan's based mobile application where we can get instant access to all waste disposal information for our community. We can get recycling tips and as well as collection updates. We get personalized calendars and reminders through mobile device. This application also helps to get accurate recycling and disposal information for thousands of household items.

The mobile application has features of waste collection schedule similar to Recycle Coach. Customer who're using waste container can view waste collection schedule from their mobile app. They can also receive notification regarding waste collection schedule according to the streets. Another similar features to Recycle coach is to get recycling tips where from the mobile application customers can read articles and blogs related to waste management system and recycling.

X-Cart is an open-source shopping cart demo software. This software helps us in tracking and managing orders and also manage multiple orders at once. From this software, we can also add unlimited products with ease. This software also helps in tracking product inventory one at a time or manage multiple products at once. This is the complete ecommerce application which has almost every feature that should be in the ecommerce application.

eBay and X-Cart is comparable to web application or admin console where different e-commerce like features are similar. From admin console superhero can manage catalog, orders and users like X-Cart. The admin console has a similar feature like eBay where adding products with details. And, other different system was researched to gain a detailed analysis of this topic. The similar projects really helped me doing the project.

3 Development

3.1 Considered Methodologies

Software methodology plays a vital role in the software development process to manage the project efficiently. All methodologies have different strengths and weaknesses and exist for different reasons. Before implementing a framework for choosing the right software methodology, different software development life cycle models like agile, iterative, and incremental, waterfall model were researched. I was confused between agile frameworks; scrum and Kanban to choose the right software model for this project. Finally, I thought that Kanban would be the right methodology for this project.

3.2 Selected Methodology

In order to achieve the expected outcome, the software methodology used in this project is the Agile Kanban Methodology. Kanban, which literally means billboard in Japanese, is a visual approach to project management. As the name suggests, Kanban workflows are arranged on a Kanban board, where the team can visualize them as a group. The agile methodology is an incremental and iterative mobile application development approach, where the complete app development process is divided into multiple sub-modules, considered as mini projects (Packt Publishing Ltd, 2020).

I've chosen the Agile Kanban methodology because, in the case of the agile development model, the complete mobile app project is divided into smaller modules that are treated like independent sub-projects. Kanban doesn't come with the size limitations whereas scrum recommends the development team to have at least 3 to 9 members.

On the other hand, Kanban is focused on work in progress. There are no sprints in Kanban. This methodology fosters testing of every single module at the primitive level. This reduces the risk of encountering a bug at the time of quality testing of the complete project. For this project, I've used a software development tool Jira to plan and track the project progress. Below shows the screenshot of the Jira Kanban board which is used for this project.

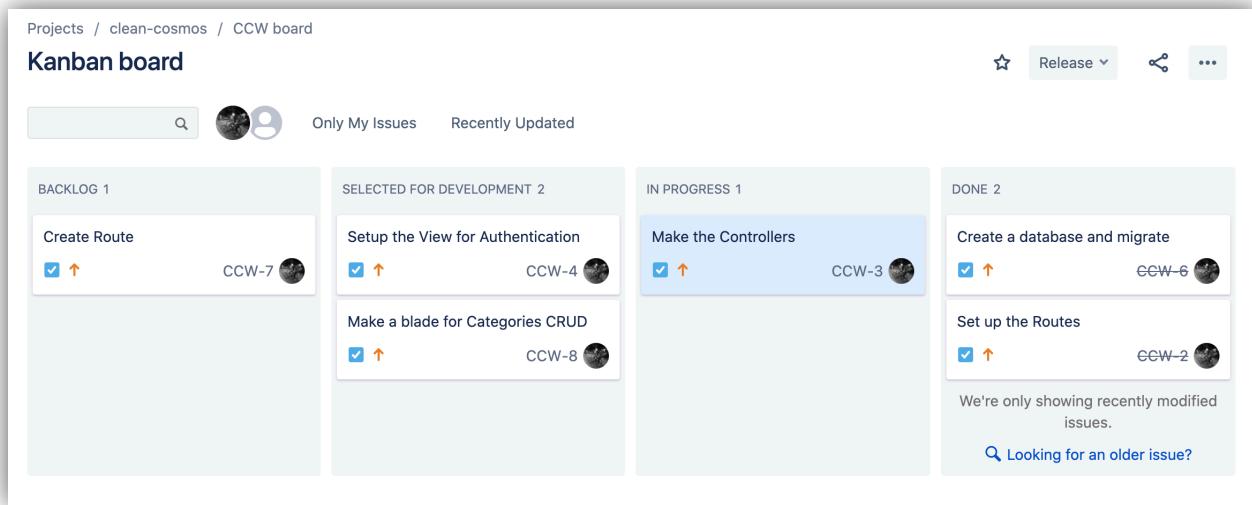


Figure 9 - Kanban board for Laravel Authentication

3.3 Phases of Methodology

The primary requirement of Kanban methodology is to divide the project into three big parts: To Do, In Progress and Done. Brief explanation of activities of each software development phase are as below:

3.3.1 Backlog

The technical requirements of the project were gathered based on the product backlog and put into the Kanban board.

3.3.2 Design and Prototyping

With the requirements in place, the software was designed to find what this software will look like and how it will function. Wireframes, UML diagrams, ER diagrams were designed in this phase.

3.3.3 Software Development

This phase is obviously the hardest and potentially riskiest stage of the project where the software was designed and developed.

3.3.4 Testing

The testing of the system was performed using different software testing approaches and methodologies.

3.3.5 Deployment

After development and testing phase, all the modules were integrated, and software was launched.

3.4 Survey Results

This section includes the results of the surveys which were taken before and after development with the help of google forms. The main purpose of doing surveys is to obtain information from people regarding waste management and recycling in their area.

3.4.1 Pre-survey Results

Pre-survey was taken before the development which helps me to identify the waste management problems facing people in their area. This survey is divided into three different sections; household waste generation and disposal, garbage collection services, and recycling where a total of twenty-one people responded.

3.4.1.1 Household Waste Generation and Disposal

In this section queries related to waste generation and disposal for an individual household were asked. Many people use medium size dustbin as a storage method which gets emptied twice a week. Total 61.9% of people doesn't separate waste types at their home and if their collection provider would tell to separate then 90.5% people will do it. A maximum number of people said that priority concern about waste in their will effects on the environment. This specific section helps me in choosing waste types and waste containers.

3.4.1.2 Garbage Collection Services

In this section questions related to garbage collection services were asked. A total of 100%, only 57.1% have regular garbage collection services in their area. And, if garbage collection providers collect garbage then almost everyone will use it. 61.9% uses collection services once a week, 33.3% uses collection services twice a week where only 4.8% uses collection services daily. Everyone uses public garbage services where only 33.3% are satisfied with their current service. This section helps me in developing logic for the garbage collection schedule.

3.4.1.3 Recycling

In this section queries related to recycling were asked. A lot of people use paper and plastic recycled materials. Some people reply about strategies for effective waste management. They said that the smart waste management system should be implemented, and garbage collection bins should be available in every local area.

3.4.2 Post-survey Results

The post-survey was taken after the development to determine whether the developed project met expectations of people or not. This survey is divided into two different sections; web application and mobile. All the user feedback was performed remotely where because of the global pandemic situation only nine people responded.

3.4.2.1 Web Application

In this section user feedback related to web application was gathered. A total of 66.7 % liked the web application design and many of them think that navigation between pages is simple and uncluttered. Out of nine responses, 44.4% rate five on catalog management feature, 33.3% rate five on order management feature, and user management feature. Out of nine responses, 4 people liked the settings of the application. Some people suggest adding extra charts on the admin dashboard.

3.4.2.2 Mobile Application

In this section user feedback related to the mobile application was gathered. Total 44.4% liked the mobile application design and the same percentage liked the functionalities of the mobile app. Out of nine responses, 44.4% rate five on ordering products from mobile app, 33.3% rate five on garbage collection features. Only 22% of the people rate five on employee app. Overall 66.7% liked and mobile application and all nine people recommend this app to others.

Screenshots for both pre-survey and post-survey forms and results are displayed in Appendix A and Appendix B.

3.5 Requirement Analysis

There are different requirement analysis techniques used to analyze the project requirement which is listed out below:

3.6 Design

3.6.1 Wireframes

3.6.1.1 Customers



Figure 10 - Loading and Login page for customers

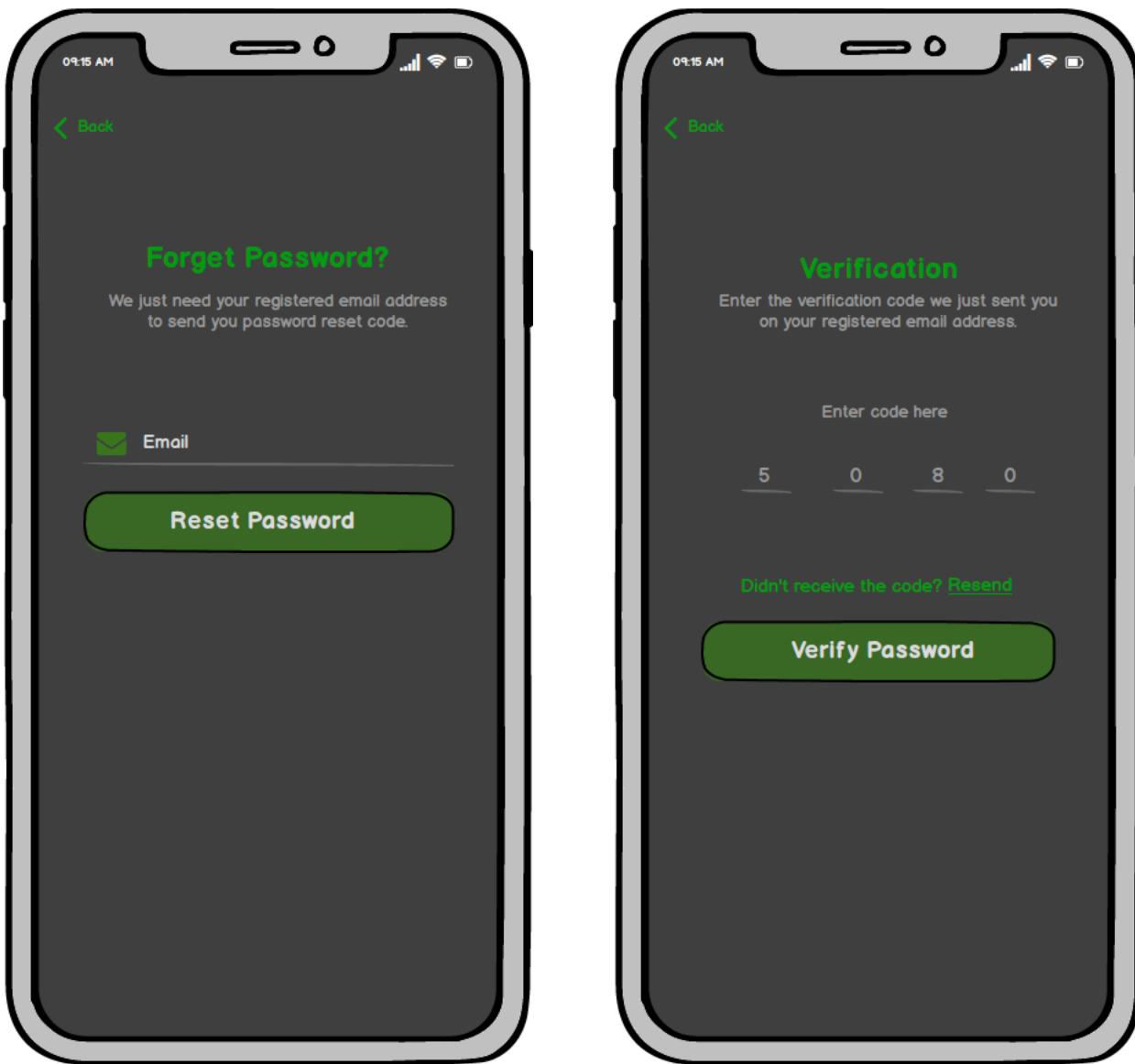


Figure 11 - Forget password page for customers

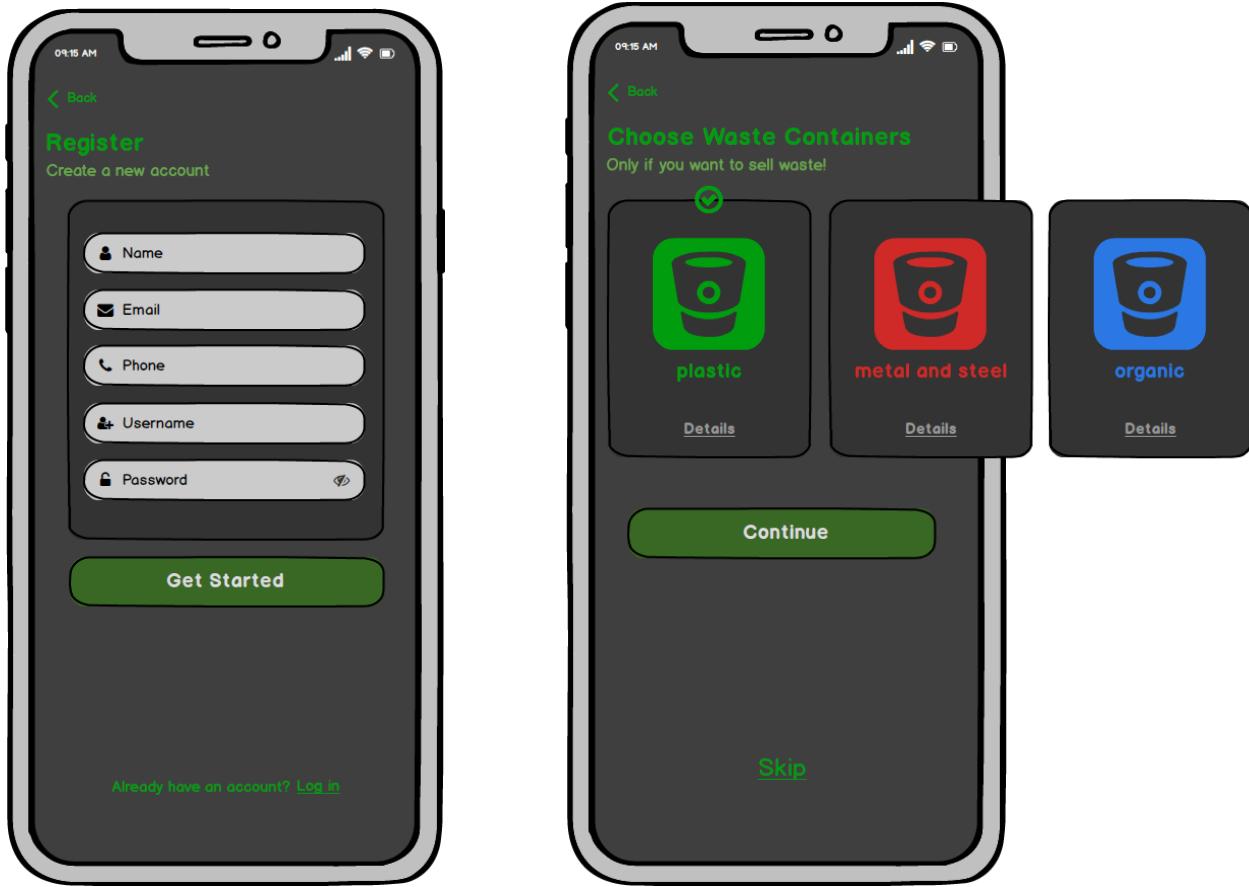


Figure 12 - Register and Choose waste containers page for customers

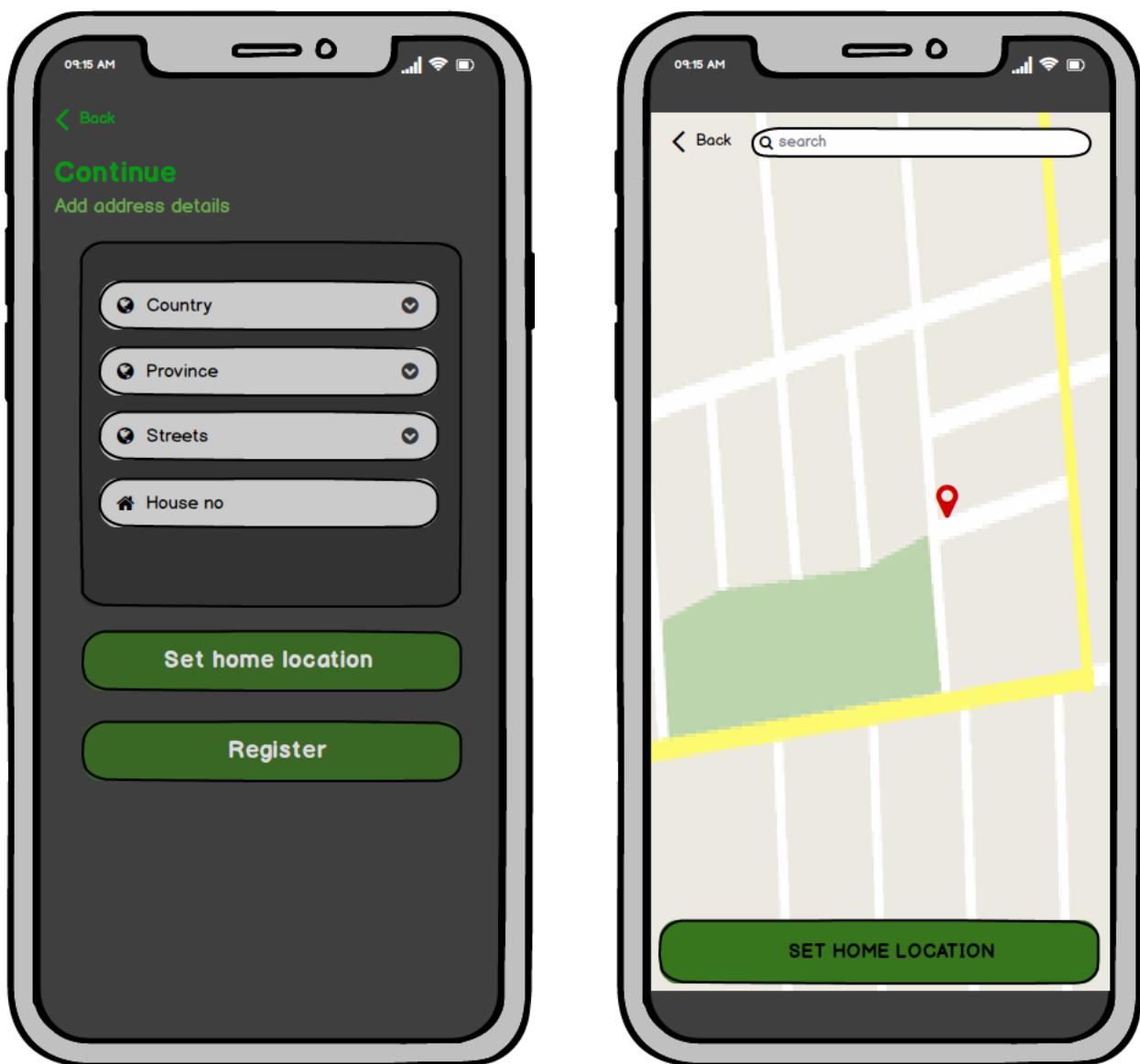


Figure 13 - Register address details and set home location page for customers

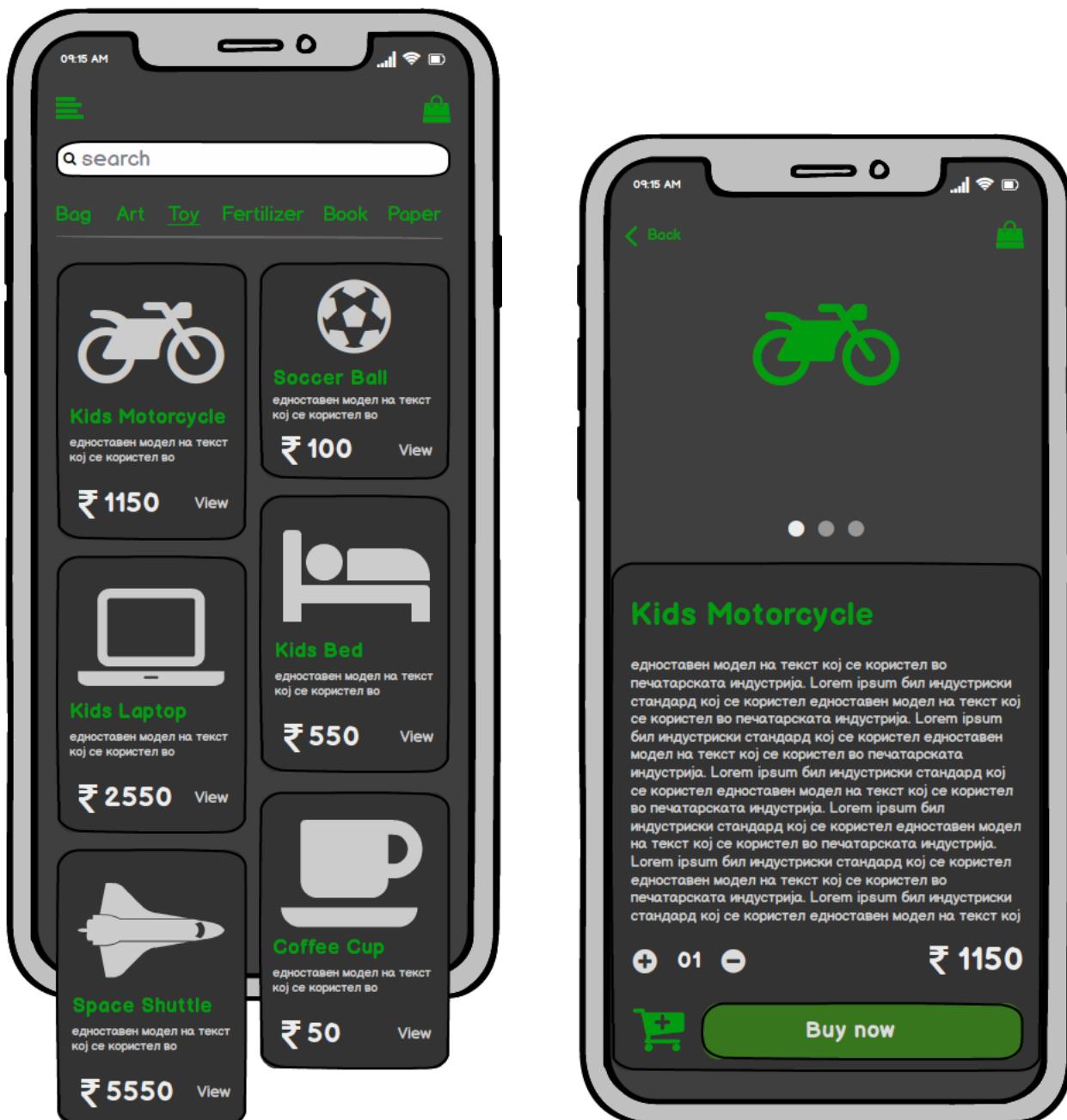


Figure 14 - Dashboard and product details page for customers

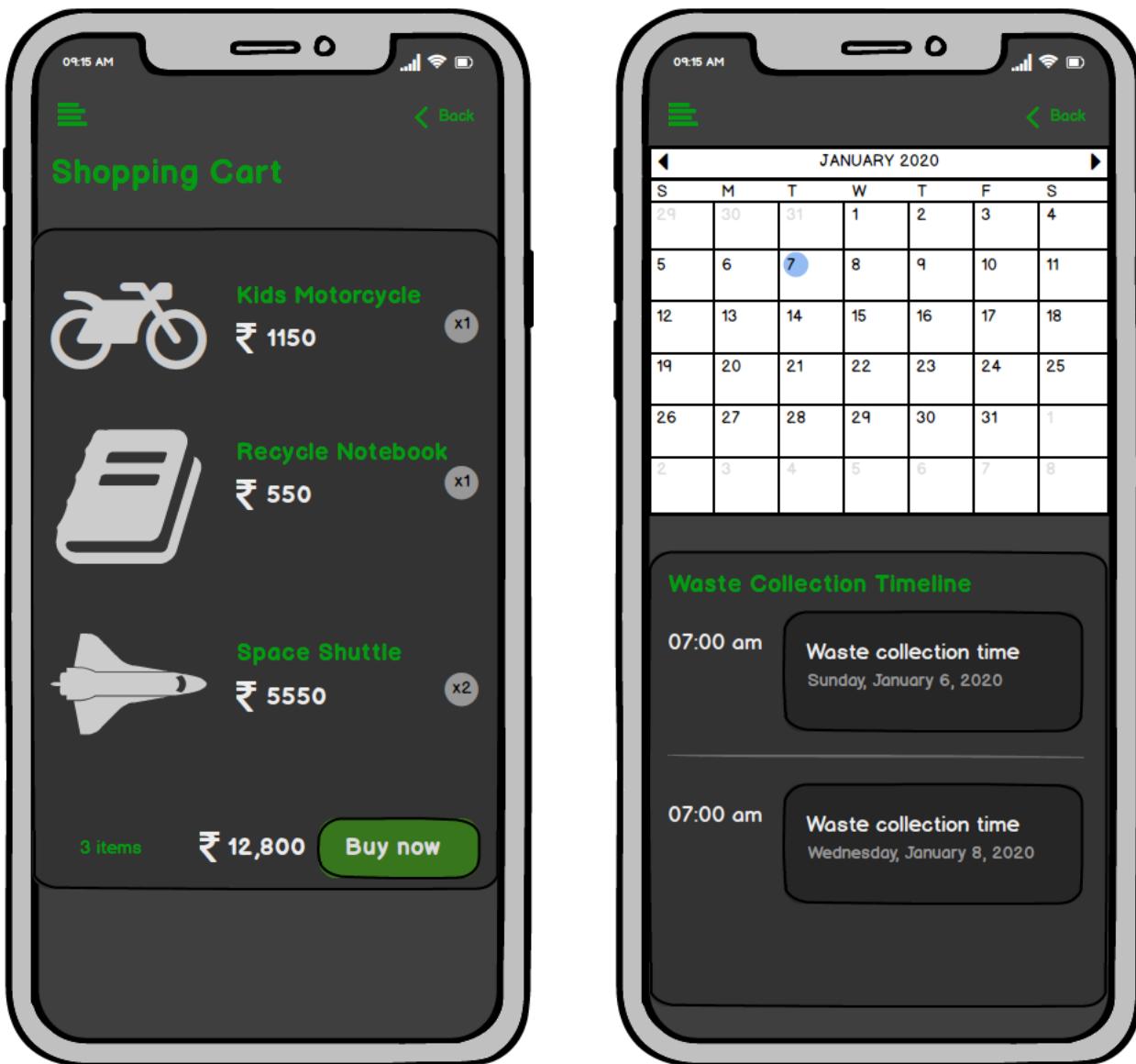


Figure 15 - Shopping cart and waste collection time for customers

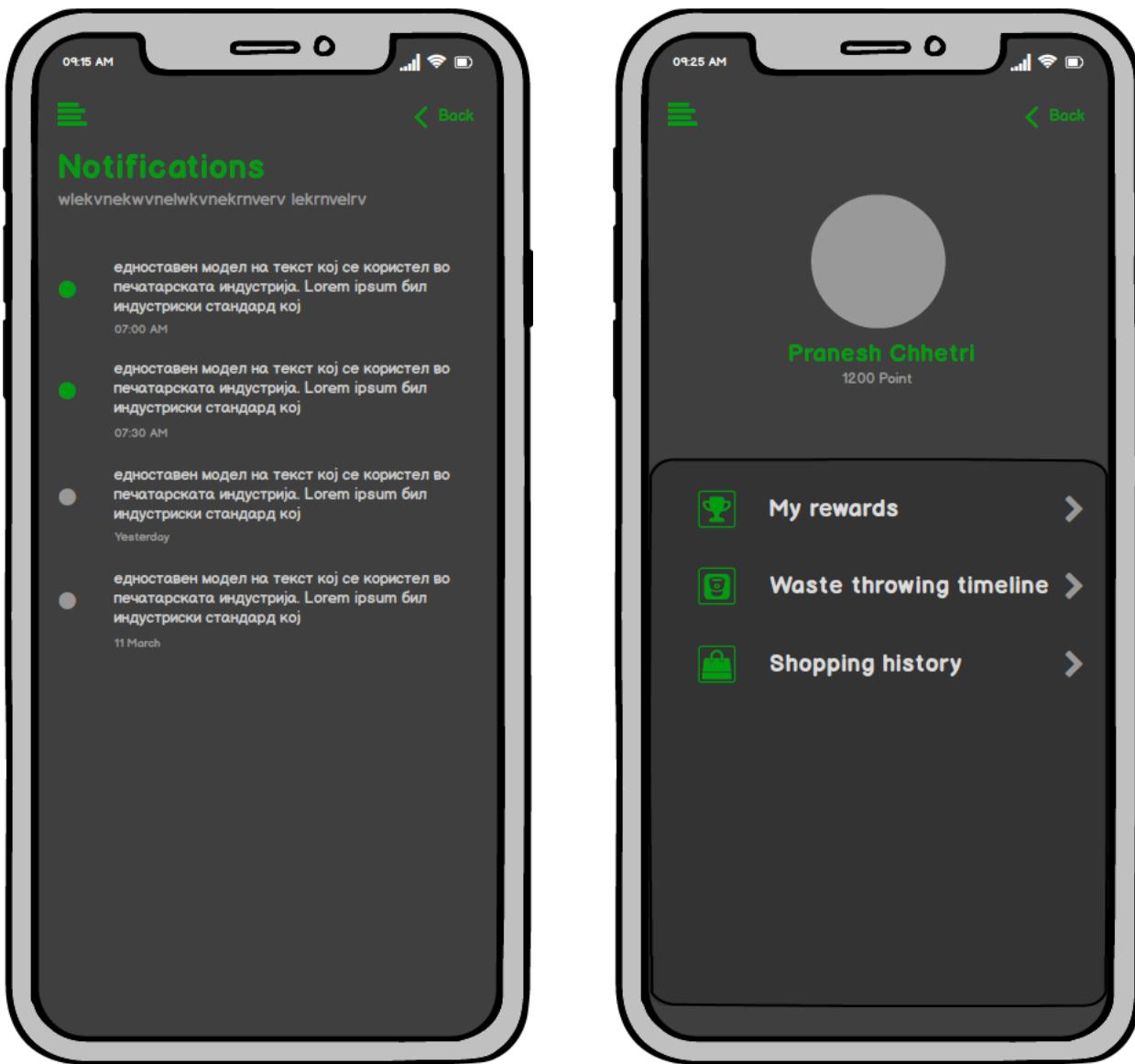


Figure 16 - Notifications and user profile page for customers



Figure 17 - Blogs and Articles page for customers

3.6.1.2 Waste collectors

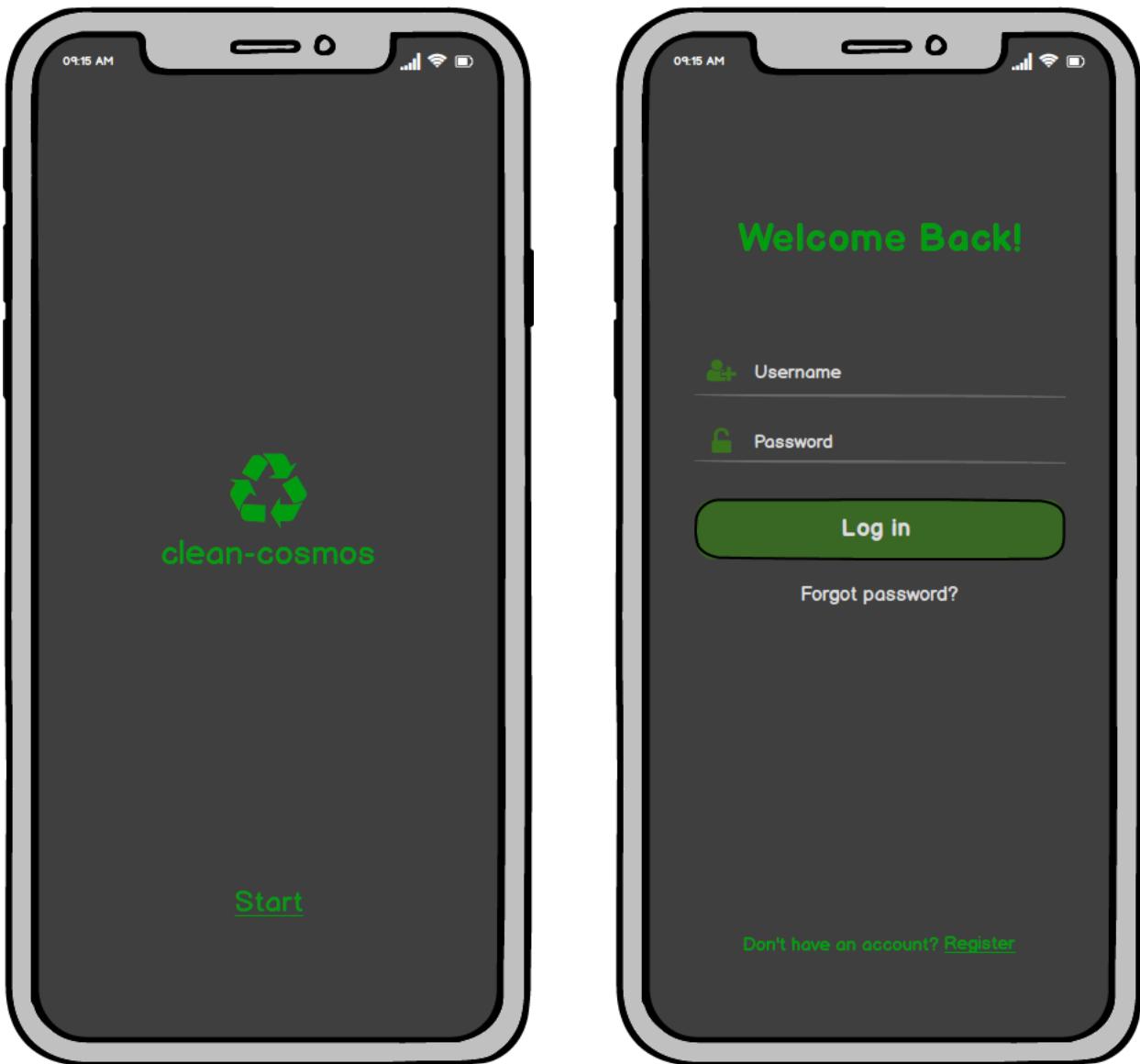


Figure 18 - Loading and log in page for waste collectors

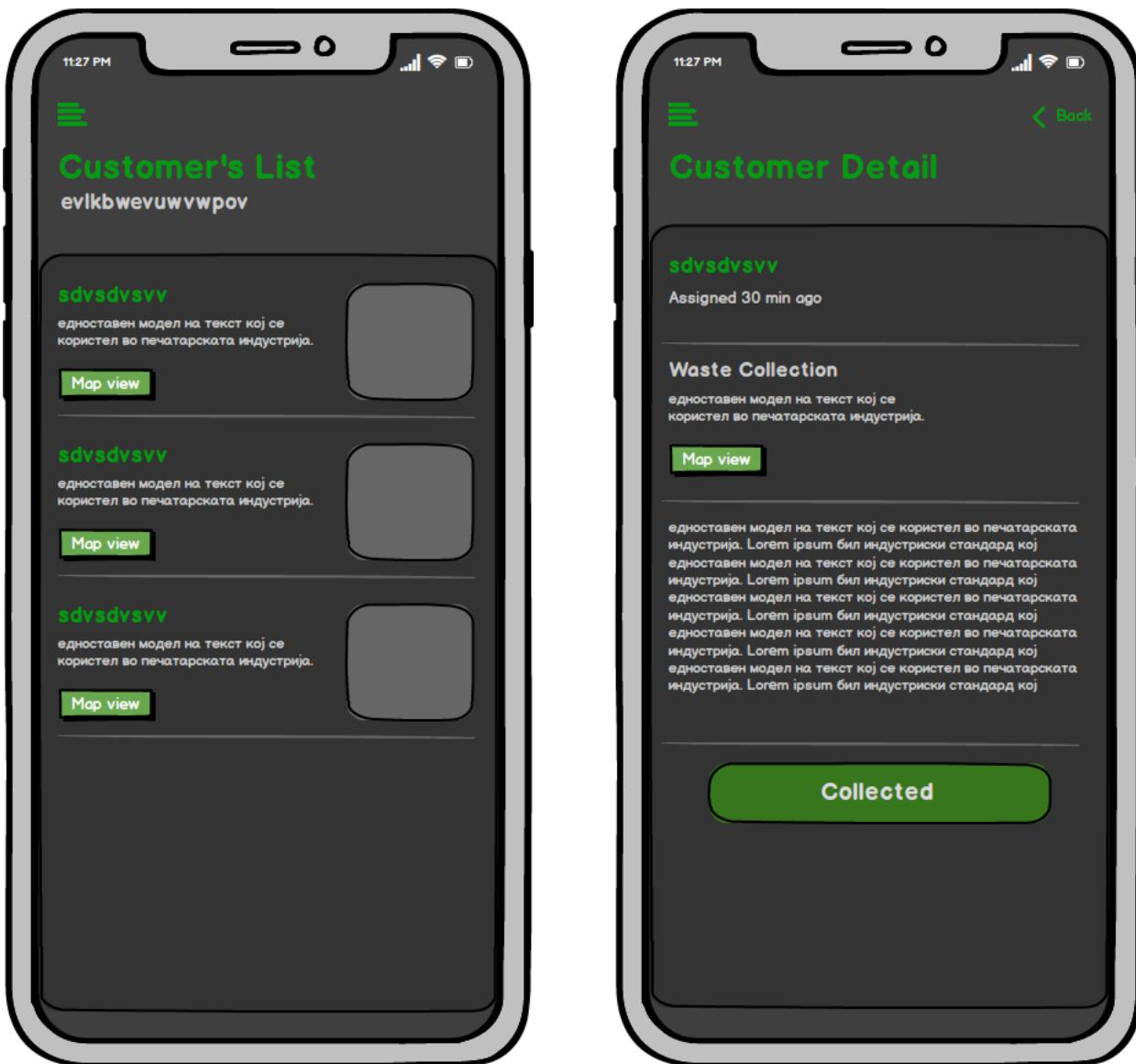


Figure 19 - Customer list and customer detail page for waste collectors

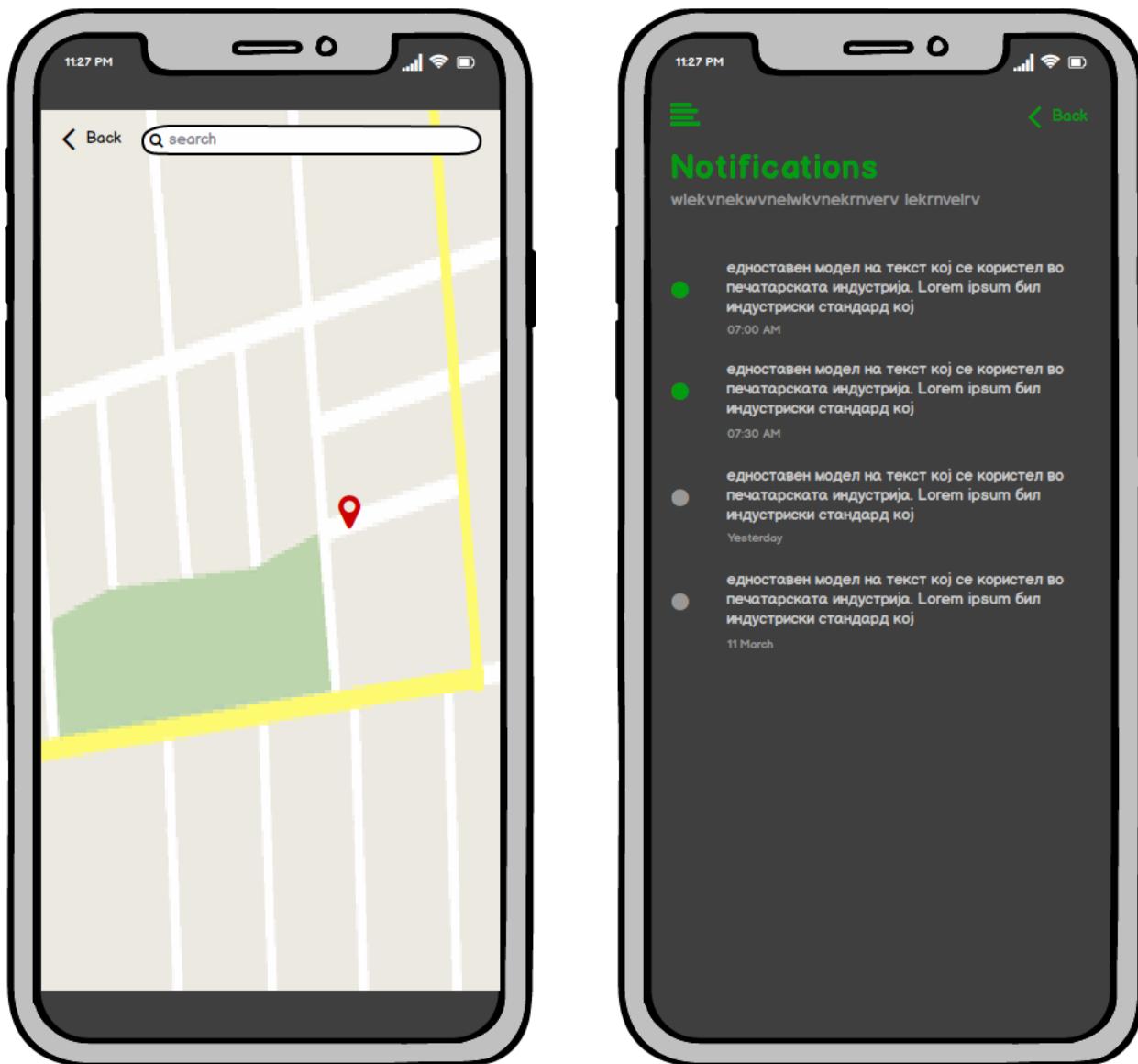


Figure 20 - Map view and notification page for waste collectors

3.6.1.3 Delivery person

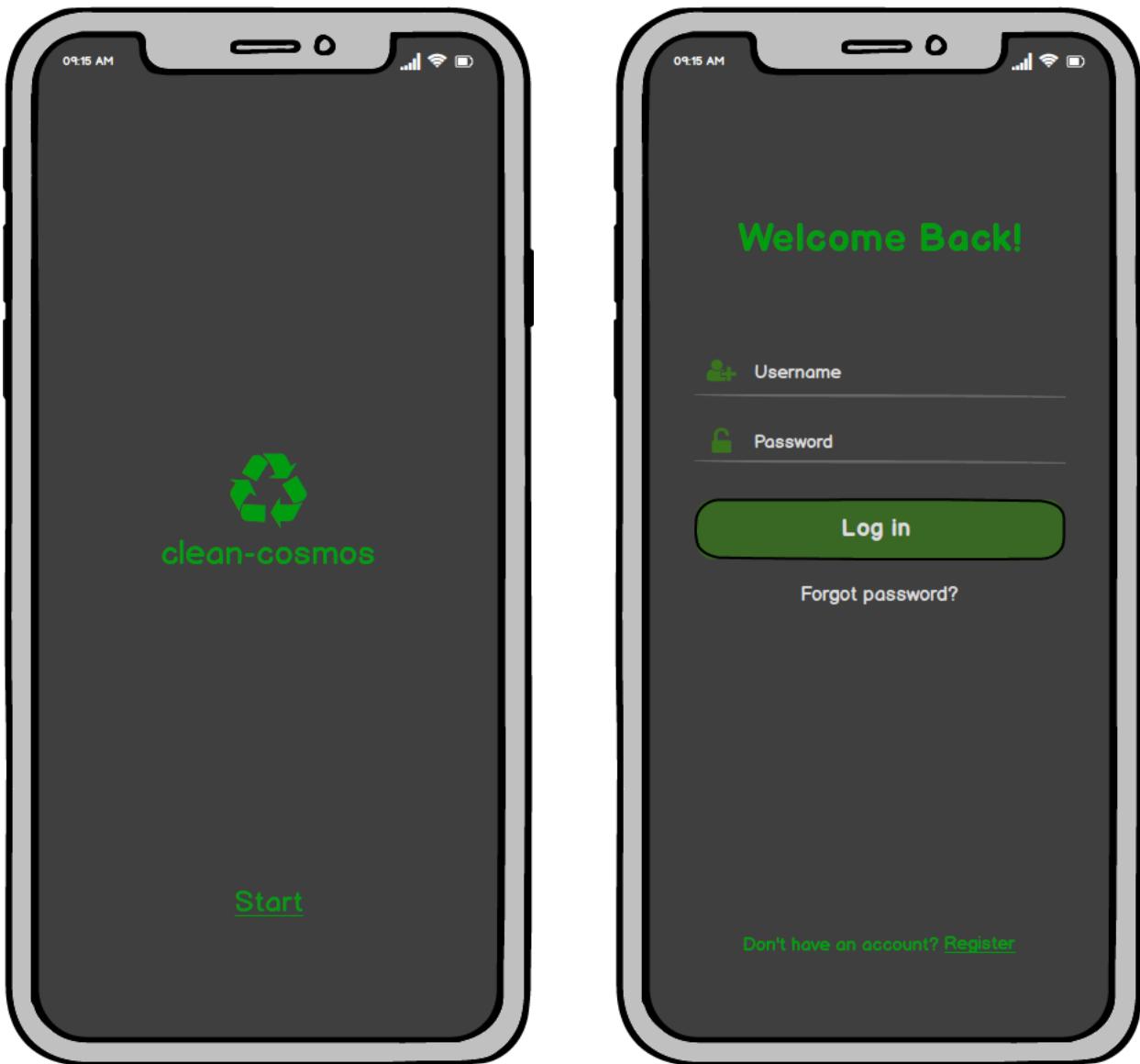


Figure 21 - Loading and log in page for delivery person

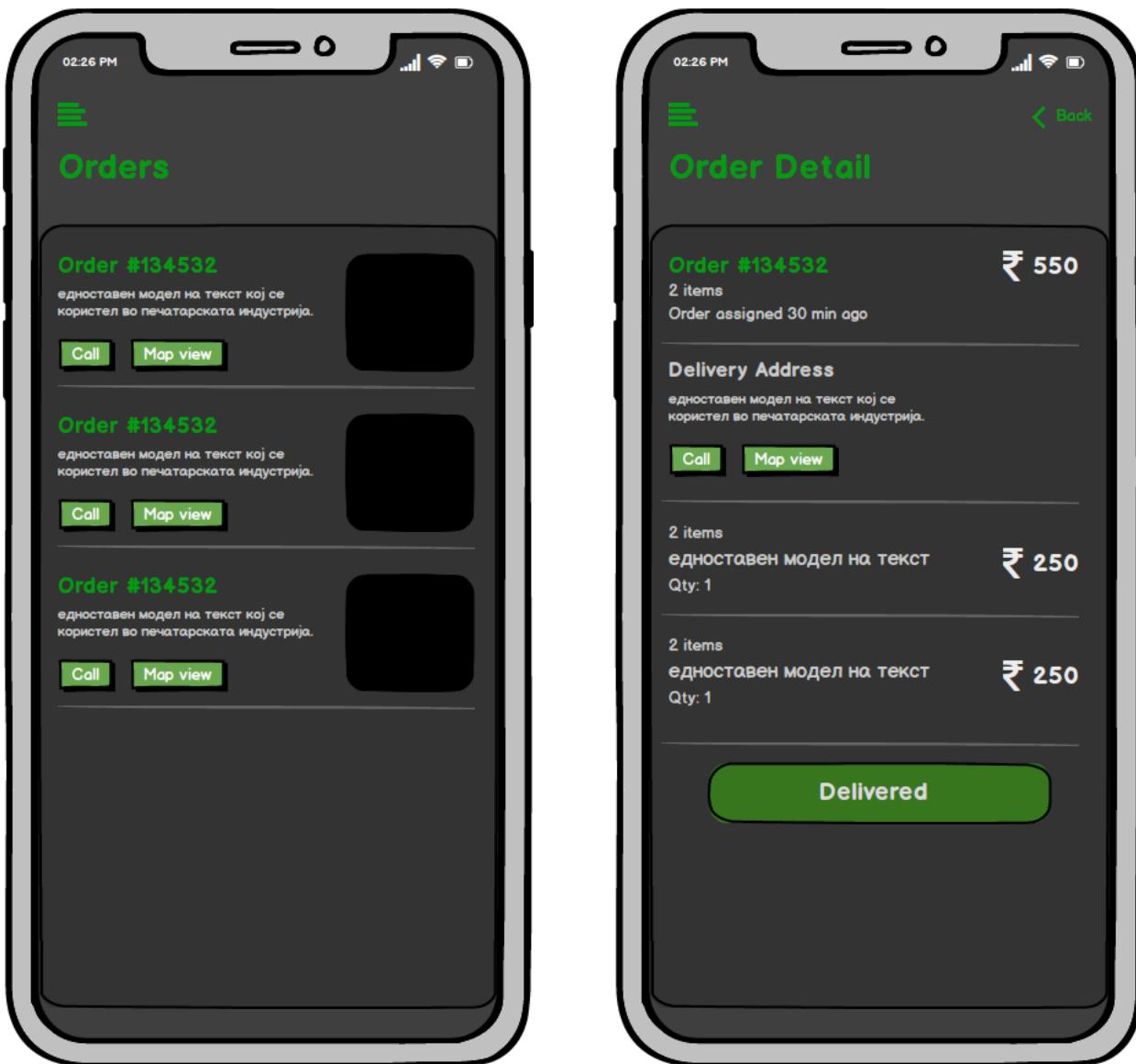


Figure 22 - Dashboard and order detail page for delivery persons

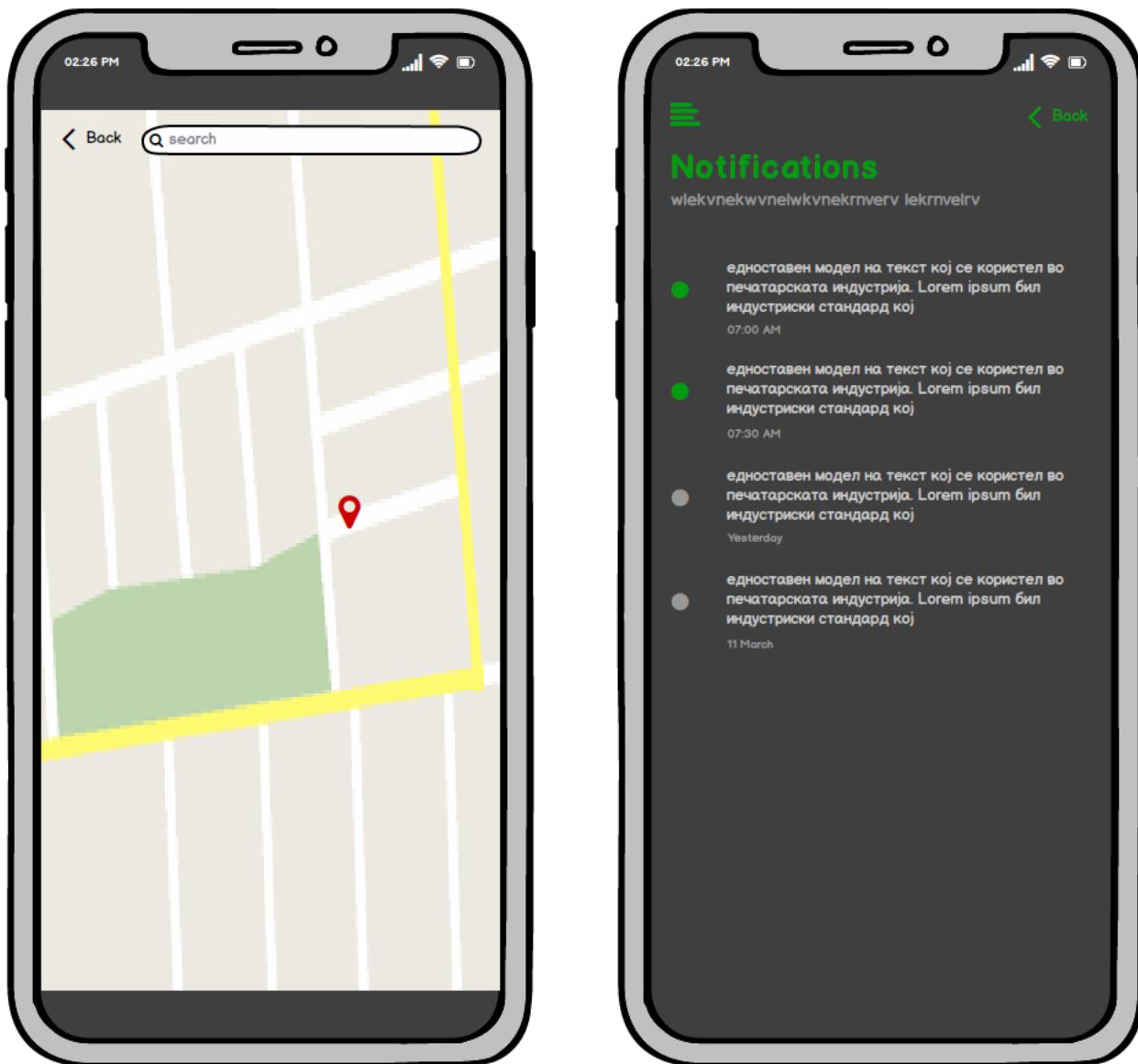


Figure 23 - Map view and notifications page for delivery persons

3.6.2 Use Case Diagram (Web Application)

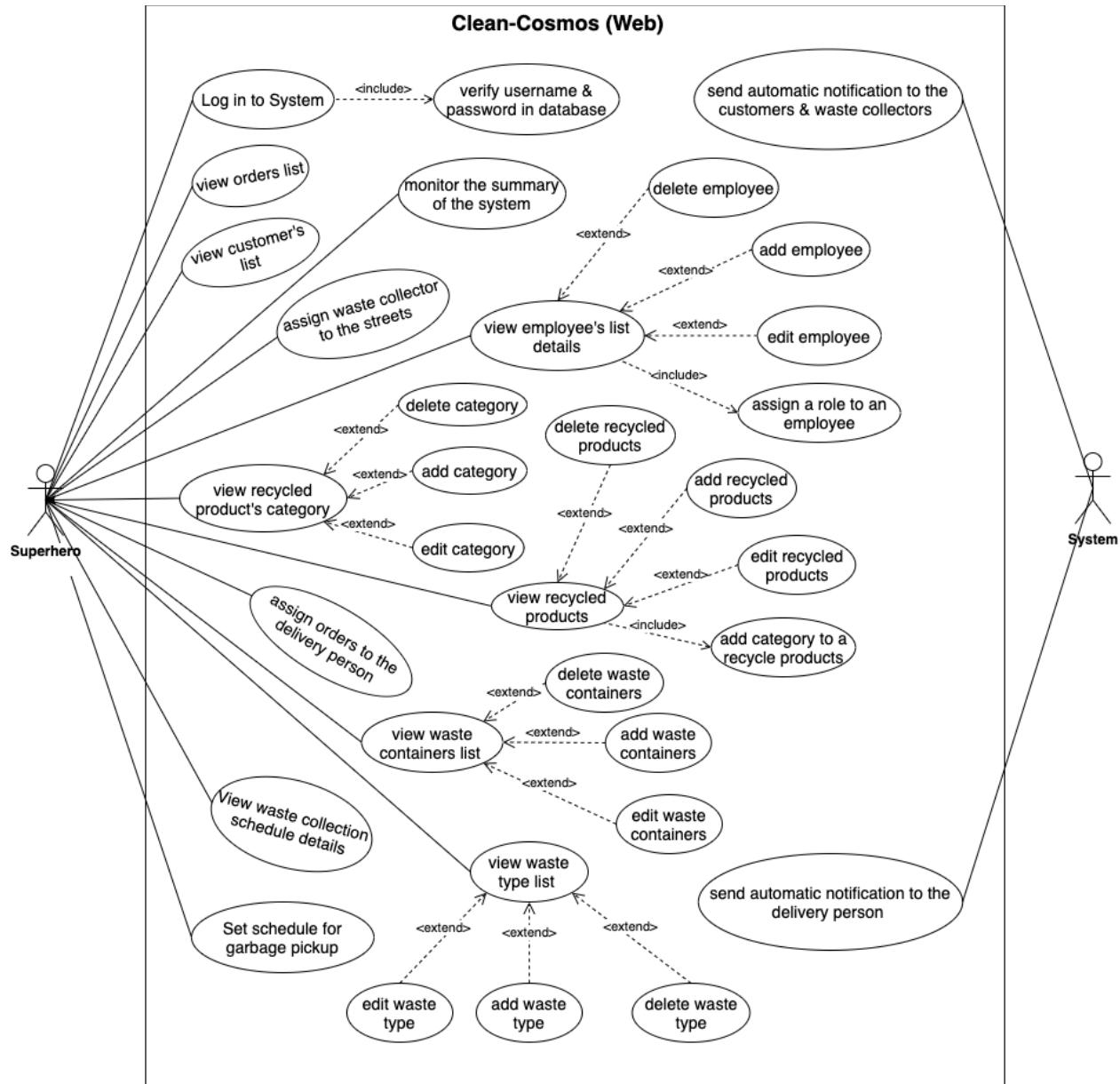


Figure 24 - User case diagram - web application

The figure above shows a use diagram for the web application. The use case model shows the use of include and extend relationship types. Besides, there are associations that connect between actors and use cases. In this project, super-admin is represented as a *superhero*.

3.6.3 High Level Use Case Description (Web Application)

3.6.3.1 Log in to System

Use Case: Log in to system

Actors: Superhero

Descriptions: In order to access the system, superhero have to log in to the system. A log in generally requires username and password and need to verify with database.

3.6.3.2 View orders list

Use Case: View orders list

Actors: Superhero

Descriptions: Superhero can view the list of total orders with details which were ordered by the customers until the present. Superhero can also view the status of the order which is pending or delivered.

3.6.3.3 Monitor the summary of the system

Use Case: Monitor the summary of the system

Actors: Superhero

Descriptions: Superhero can monitor the summary of the system through dashboard. He can view the total number of orders from customers, total revenue collected, total waste collected, and total products sold until the present. He can also view the recent orders from dashboard.

3.6.3.4 View customer's list

Use Case: View customer's list

Actors: Superhero

Descriptions: Superhero can view the list of customers with details. He can also search the data from the table.

3.6.3.5 View employee's list

Use Case: View employee's list

Actors: Superhero

Descriptions: Superhero can view the employee's list details. He can add, edit and delete the employees. He can also assign role to an employee.

3.6.3.6 View recycle product's category

Use Case: View recycle product's category

Actors: Superhero

Descriptions: Superhero can view the category of recycle products. He can add, edit and delete a category.

3.6.3.7 View recycled products

Use Case: View recycled products

Actors: Superhero

Descriptions: Superhero can view the list of totals recycled products. He can also add, edit and delete recycled products. He can also add category to a recycled product.

3.6.3.8 View waste type list

Use Case: View waste type list

Actors: Superhero

Descriptions: Superhero can view the list of waste types. He can also add, edit and delete waste type.

3.6.3.9 View waste containers list

Use Case: View waste containers list

Actors: Superhero

Descriptions: Superhero can view the list of waste containers. He can also add, edit and delete waste containers.

3.6.3.10 Assign waste collector to the streets

Use Case: Assign waste collector to the streets

Actors: Superhero

Descriptions: Superhero can assign waste collector to the multiple streets when it's time to collect waste. All the customers of that streets were assigned to that waste collector.

3.6.3.11 Assign orders to the delivery person

Use Case: Assign orders to the delivery person

Actors: Superhero

Descriptions: Superhero can view the list of orders with details. Superhero assign orders to the delivery person which were pending. And, after order assigned, delivery person delivered the products.

3.6.3.12 Set schedule for garbage pickup

Use Case: Set schedule for garbage pickup

Actors: System

Descriptions: Superhero can set schedule for garbage pickup according to streets. He can also search data from the table.

3.6.3.13 View waste collection schedule details

Use Case: view waste collection schedule details

Actors: System

Descriptions: Superhero can view the waste collection schedule details of customers. He can also view the waste container used, next collection date and last waste collection date for individual customers according to streets.

3.6.3.14 Send automatic notifications to the customers & waste collectors

Use Case: Send automatic notifications to the customers

Actors: System

Descriptions: System sends automatic notification to the customers & waste collectors when it's time to collect waste. The notification day and collection day is set by superhero.

3.6.3.15 Send automatic notifications to the delivery persons

Use Case: Send automatic notifications to the delivery persons

Actors: System

Descriptions: System sends automatic notifications to the delivery persons when superhero assign orders to the delivery person.

3.6.4 Use Case Diagram (Mobile Application)

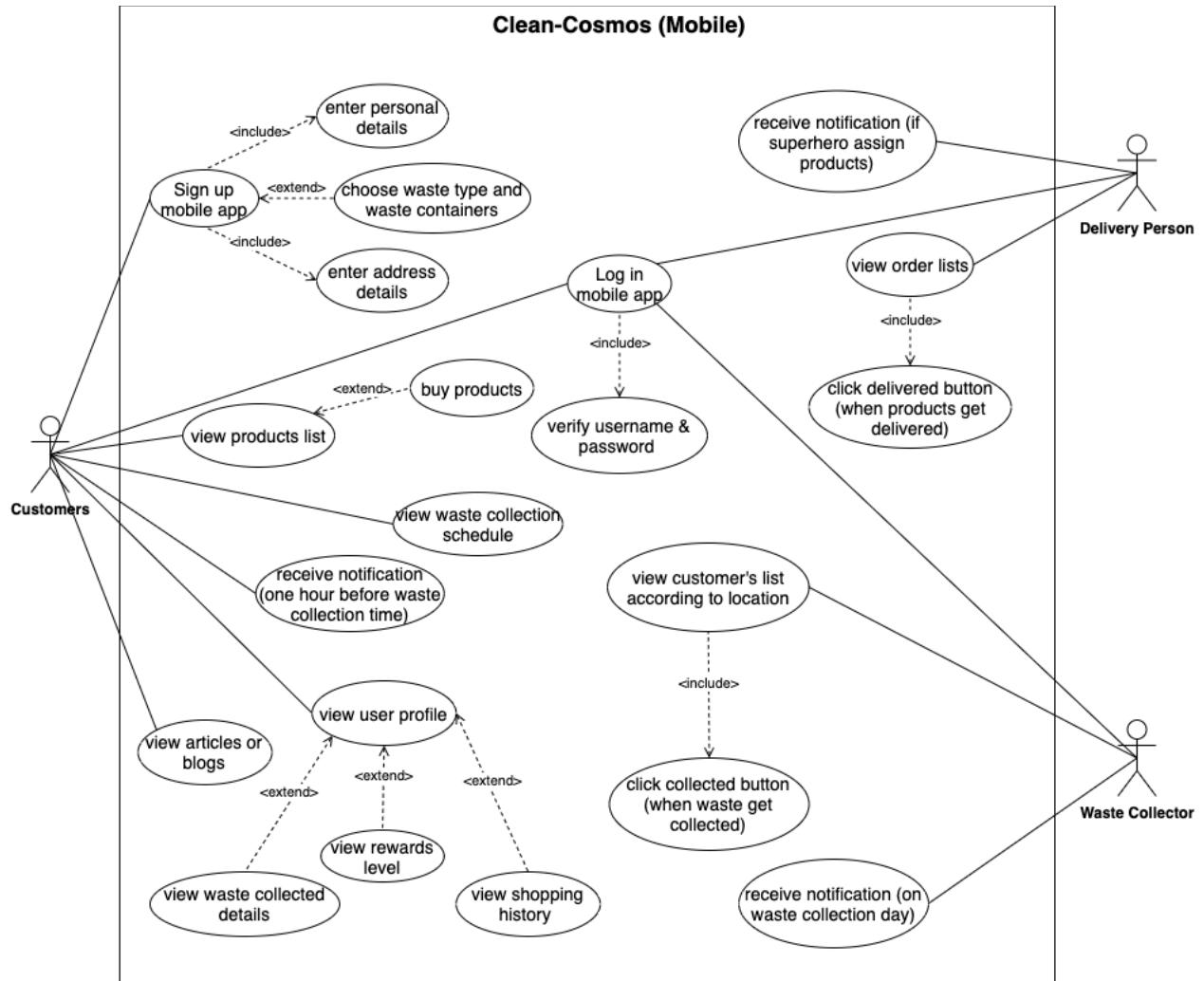


Figure 25 - Use case diagram - mobile application

The above figure shows a use diagram for the mobile application. The use case model shows the use of include and extend relationship types. Besides, there are associations that connect between actors and use cases.

3.6.5 High Level Use Case Description (Mobile Application)

3.6.5.1 Sign up mobile app

Use Case: Sign up mobile app

Actors: Customers

Descriptions: Customers need to sign up in the system in order to use the application. They need to enter their personal details, and their address details before signing up in the system. They need to choose waste type and waste container only if they want to sell waste.

3.6.5.2 Log in mobile app

Use Case: Log in mobile app

Actors: Customers, Waste collectors, and Delivery person

Descriptions: In order to access the system all end-user have to login to the system. System verify username and password after user log in to the mobile app.

3.6.5.3 View products list

Use Case: View products list

Actors: Customers

Descriptions: Customers can view list of products which can be recycle products or waste container with all the details which was recorded from admin panel. He can also buy the products. To buy waste container, customer needs to choose waste type too.

3.6.5.4 View waste collection schedule

Use Case: View waste collection schedule

Actors: Customers

Descriptions: Customers can view their waste collection schedule of their street.

3.6.5.5 View user profile

Use Case: View user profile

Actors: Customers

Descriptions: Customer can view their profile. They can view the list of waste container they are using, and total garbage collected in their profile. They can also view the history of their purchased orders and rewards level.

3.6.5.6 View articles and blogs

Use Case: View articles and blogs

Actors: Customers

Descriptions: Customers can view the articles and blogs related to waste management system and recycling.

3.6.5.7 View order list

Use Case: View order list

Actors: Delivery Person

Descriptions: Delivery person can view the list of orders with details which were assigned by superhero. Delivery person needs to click delivered button when orders gets delivered.

3.6.5.8 View customer's list according to location

Use Case: View customer's list according to location

Actors: Waste Collectors

Descriptions: Waste collectors can view the list of customers of their assigned streets. They can also sort out the customer according to street. They can also input the total waste collected from individual customers.

3.6.5.9 Receive notifications

Use Case: Receive notifications

Actors: Customers, Waste Collectors, Delivery Persons

Descriptions: All the user receive notifications and can view the notification list in their mobile application. Customers receive notification before one hour of waste collection time. Waste collector receive notification on waste collection day. Similarly, delivery person receive notification after superhero assigned orders to them.

3.6.6 Level 2 Data Flow Diagram

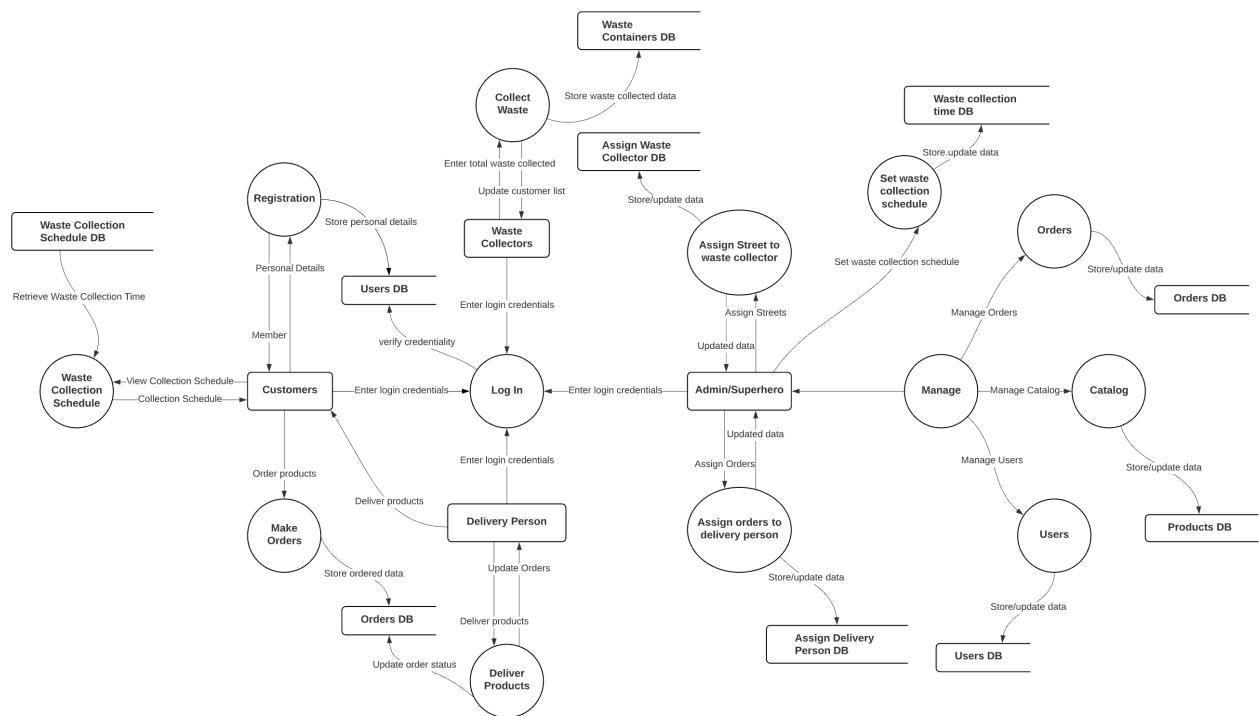


Figure 26 - Level 2 data flow diagram of the system

Based on the diagram, every external entity can log in to the system where further it verifies login credentials with a database. Customer can register to the system and view the waste collection schedule. Admin/superhero manages the orders, catalog, and users. Superhero can assign employees where waste collector can collect waste from customers and deliver person can deliver orders to customers.

3.6.7 ER Diagram

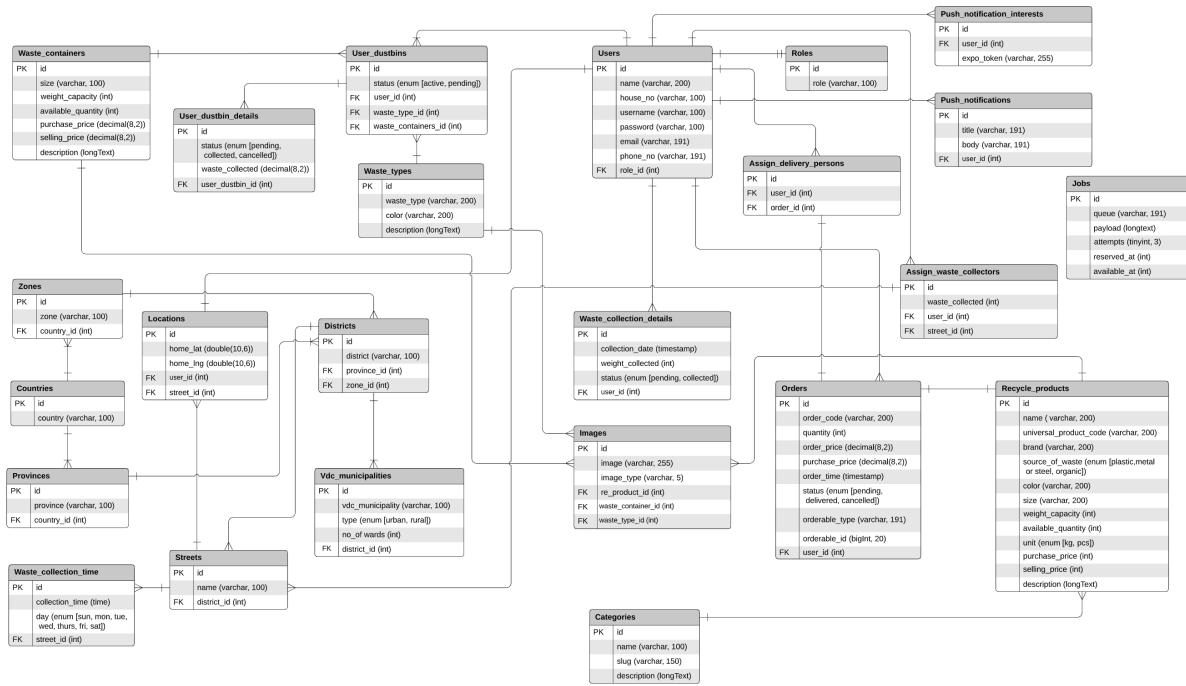


Figure 27 - ER Diagram of the system

Based on the above diagram, it shows different entity within a system and how they relate to each other. Database seed made to store the local address of Nepal in the database. Admin can assign employees; delivery person to deliver orders and waste collectors to collect waste. Customers can make multiple orders and can have multiple waste containers. Images for recycle products, waste containers and waste types are stored in the images table.

Entities push_notifications, push_notification_interests and jobs are used for notifications where push_notification_interests stores the unique expo token for each new device, push_notifications stores the title, body and time of notifications for users. Jobs table which stores Laravel queue which helps to queued notifications and find available time to notify users.

The Gantt chart, work break down structure, and expanded use case description for each use case is described below in Appendix D: Designs.

3.7 Core Development Features Implementation

1. Set garbage collection schedule:

A superhero selects the province which sends ajax requests to fetch districts and then renders to a select tag. After that, a superhero select district and get the street name. Then, the superhero enters garbage collection time for the streets. Here, I've used jQuery that process data in the application layer where only inputted data are stored in the database as time datatype. This helps to reduce the server load.

The screenshot shows a web-based scheduler interface titled "SCHEDULER". At the top, it says "Set time for garbage pickup." Below this, there are dropdown menus for "Province" (set to "Province 3") and "District" (set to "Kathmandu"). There are also buttons for "Show 25 entries" and a search bar with the placeholder "Search: ne".

The main area displays a grid of street names and their collection times for each day of the week. The columns are labeled "Street Name" and "Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", and "Saturday". The rows list four streets: "Mid Baneshwor", "New Baneshwor", "New Road", and "Old Baneshwor". Each row contains seven time slots corresponding to the days of the week, with each slot containing a time value (e.g., "07:30:00 AM") and a small circular icon.

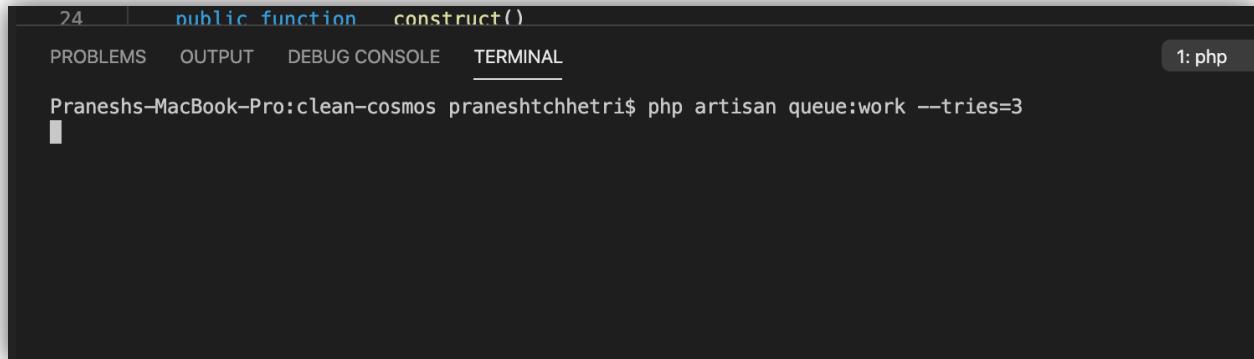
At the bottom left, a button labeled "Submit" is visible. At the bottom right, there are navigation buttons for "Previous", "1", and "Next", along with a message indicating "Showing 1 to 4 of 4 entries (filtered from 20 total entries)".

Figure 28 - Set waste collection schedule page

2. Push notification

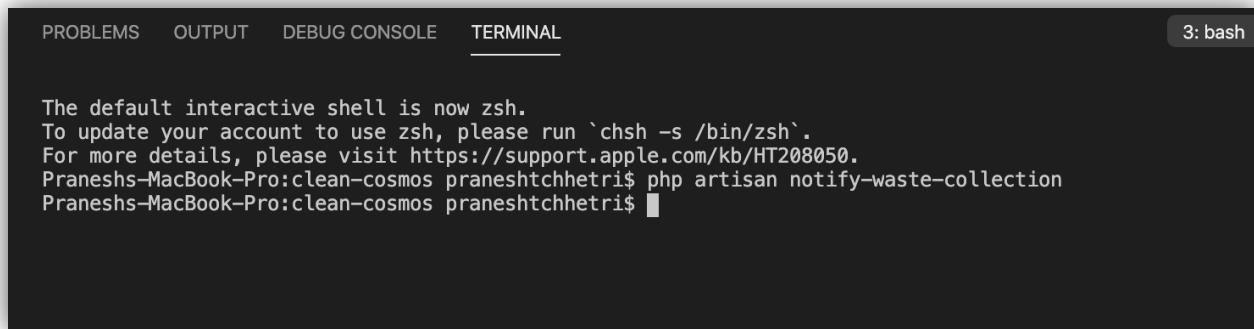
Initially, the `php artisan queue:work` command should be run in the background which trigger the jobs. Then, run command `php artisan notify-waste-collection` for sending notification.

Then a system finds out the users who has waste collection schedule for today. For each user it receives the waste collection time. Customer receive a notification before one hour of waste collection time. System call the function that set a notification time before one hour of collection time. Then it creates a Laravel job which triggers notification time. After that it push the job to the Laravel queue waiting to send notification on time. Waste collector receive notifications on waste collection day where delivery receive notification after superhero assigned orders to them.



```
24 | public function __construct()
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: php
Praneshs-MacBook-Pro:clean-cosmos praneshtchhetri$ php artisan queue:work --tries=3
```

Figure 29 - Queue is running in the background



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
3: bash
The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
Praneshs-MacBook-Pro:clean-cosmos praneshtchhetri$ php artisan notify-waste-collection
Praneshs-MacBook-Pro:clean-cosmos praneshtchhetri$
```

Figure 30 - Notifying users manually using this command

3.7.1 Mobile Application Features

The mobile application features for different user types; customers, waste collectors and delivery person are pointed out below:

3.7.1.1 Customers

- a. Customers can easily login & register into the mobile application.
- b. The navigation of the mobile application is very simple and as much uncluttered as it can be. It delivers seamless navigation and displays all the products in the most impressive manner.
- c. Customers can simply buy and sell precious different kinds of wastes, raw materials, and waste made products.
- d. Customers can easily view the product gallery and search for products from the product gallery by using a search bar at the top of the screen.
- e. Customers can check the order history effortlessly which includes the list of the past purchased products.
- f. Customer mobile app helps individual households to throw away the right waste, helping them with a simple interface organized by categories and weekly schedule for garbage collection.
- g. Customers receive notification of the garbage collection schedule.
- h. Customers can read the blogs and articles related to recycling and waste management.
- i. Customers can quickly check out the rewards and their waste-related data from the user profile.

3.7.1.2 Employees

- a. Waste collectors and delivery persons can easily log into the mobile application anytime they want and also stays logged in as well.
- b. The delivery person can check the number of orders which were allocated to them.
- c. Waste collectors can easily check the customer's list and filter or sort out customers according to the streets.
- d. A delivery person receives a notification when the order is placed or is out for delivery.
- e. The waste collector receives the notification of the garbage collection schedule.

3.7.2 Web Application Features

The web application features for admin or superhero are pointed out below:

- a. A superhero can easily log into the application anytime they want and also stays logged in as well.
- b. The navigation of the web application is very simple and as much uncluttered as it can be which helps the superhero to easily get to where he needs to be.
- c. The web application allows the superhero to add, edit and, delete recycled products, waste types, waste containers and, employees. He can easily view the list of recycled products, waste types, waste containers and, employees with detailed information.
- d. A superhero can easily reset his and employee's password from the admin console.
- e. A superhero can easily view the customer details and the details of their waste containers.
- f. A superhero can simply view the ordered product details and sort out the pending and delivered orders.
- g. The web application has built-in analytics that helps the superhero keep track of the orders, waste collected, total products sold, and total revenue collected.
- h. The web application allows a superhero to place orders to the delivery person and assign locations to the waste collectors.
- i. The web application allows a superhero to set the time for garbage collection and view the customer's garbage pickup schedule.

4 Testing and Analysis

4.1 Test Plan

Software testing is a crucial part of software development, it ensures that the software developed performs all functional requirements and is free from any form of defect and errors (Ghahrai, 2020). The test plan describes the different approaches and methodologies that applied to the unit, integration, system and, compatibility testing. The plan identified the features to be tested, the types of testing to be performed, the resources and schedule required to complete testing, and the risks associated with the plan. The popular agile framework Kanban was used to organize tasks and make the processes clearer and more consistent.

Types of testing performed:

1. **Unit Testing:** where individual application component or module are tested
2. **Integration Testing:** all the APIs or integrated module of the application are tested
3. **Compatibility Testing:** where the application is tested in different hardware to validates how the system behaves and runs in a different environment
4. **Acceptance Testing:** performed this testing via friends and others to identify whether the end to end flow of the application is as per the needs of the end-user or not.

The main objective of this testing is to define the various testing strategies which verify the functionality of the application to complete the testing life cycle of this project. It also helps in finding defects while developing the application and finding the quality of the project.

4.2 Unit Testing

4.2.1 Web Application

Below show all the test cases which are being tested using unit testing approach.

4.2.1.1 Admin Login

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC1	Login with valid username and password	1. Enter valid username and password 2. Click login button	User should be logged into an application	Pass
TC2	Login with invalid username and password	1. Enter invalid username and password 2. Click login button	User shouldn't be logged into an application and error message should be shown.	Pass
TC3	Login with empty field	1. Leave any of the field empty 2. Click login button	User shouldn't be logged into an application and error message should be shown.	Pass

Table 1 - Test cases for admin login

Below show the screenshots for test case **TC1** (login with valid username and password):

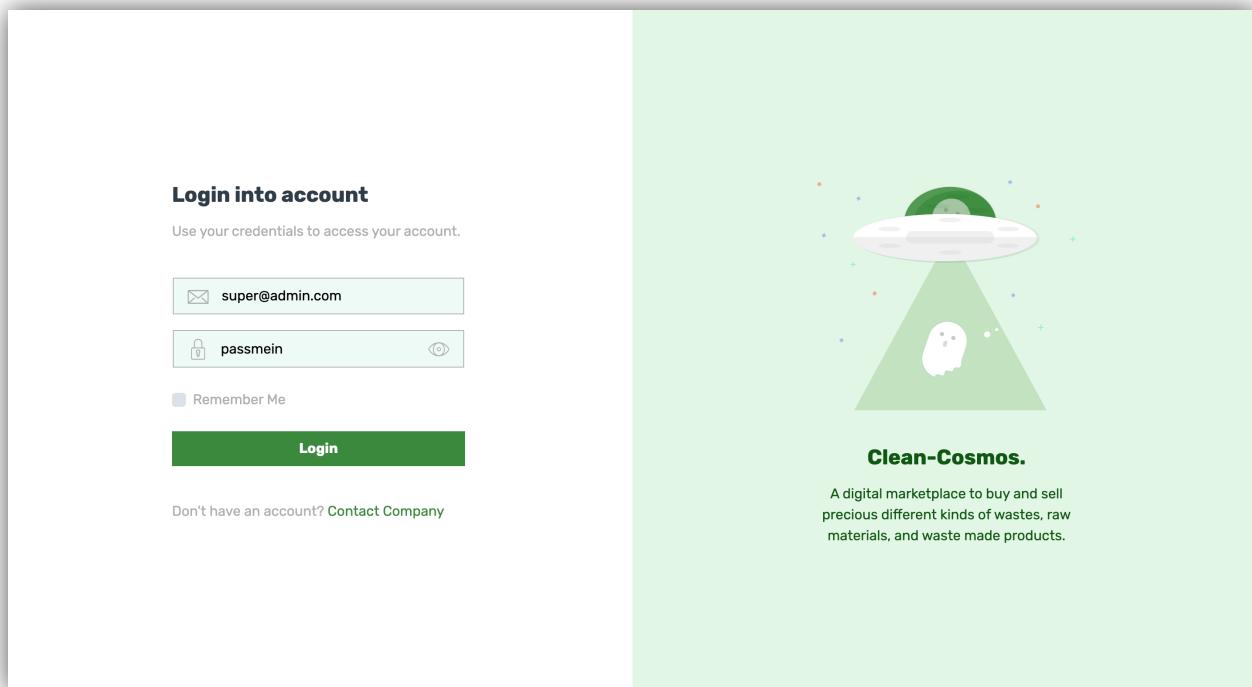


Figure 31 - Test case TC1 when user enters valid username and password

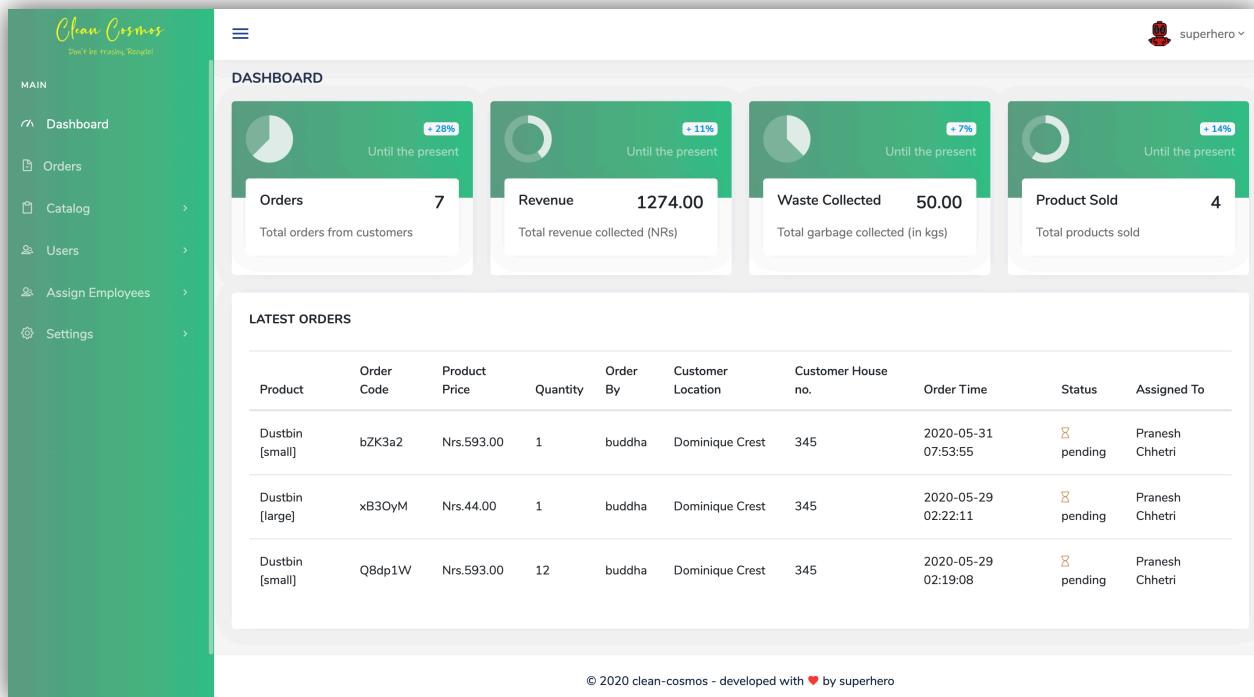


Figure 32 - Test case TC1 with valid username and password user successfully log into the application

Below show the screenshots for test case **TC2** (login with invalid username and password):

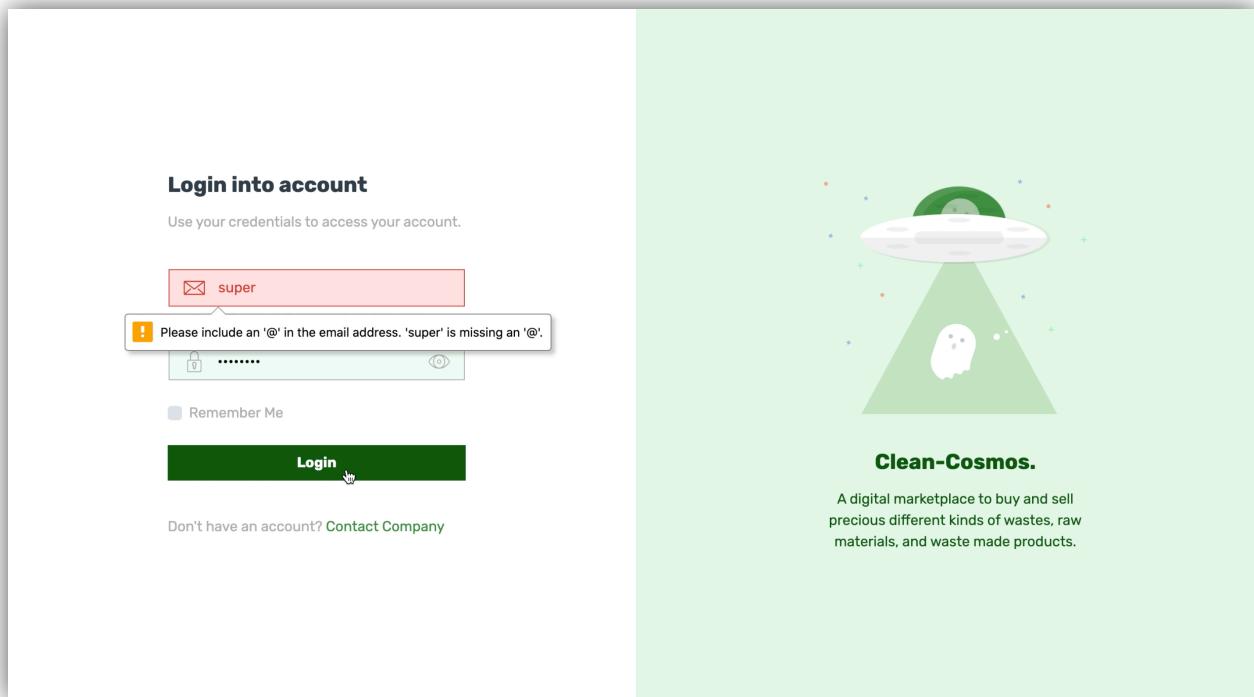


Figure 33 - Test case TC2 showing error message when user enters invalid username and password

Below show the screenshots for test case **TC3** (login with empty field):

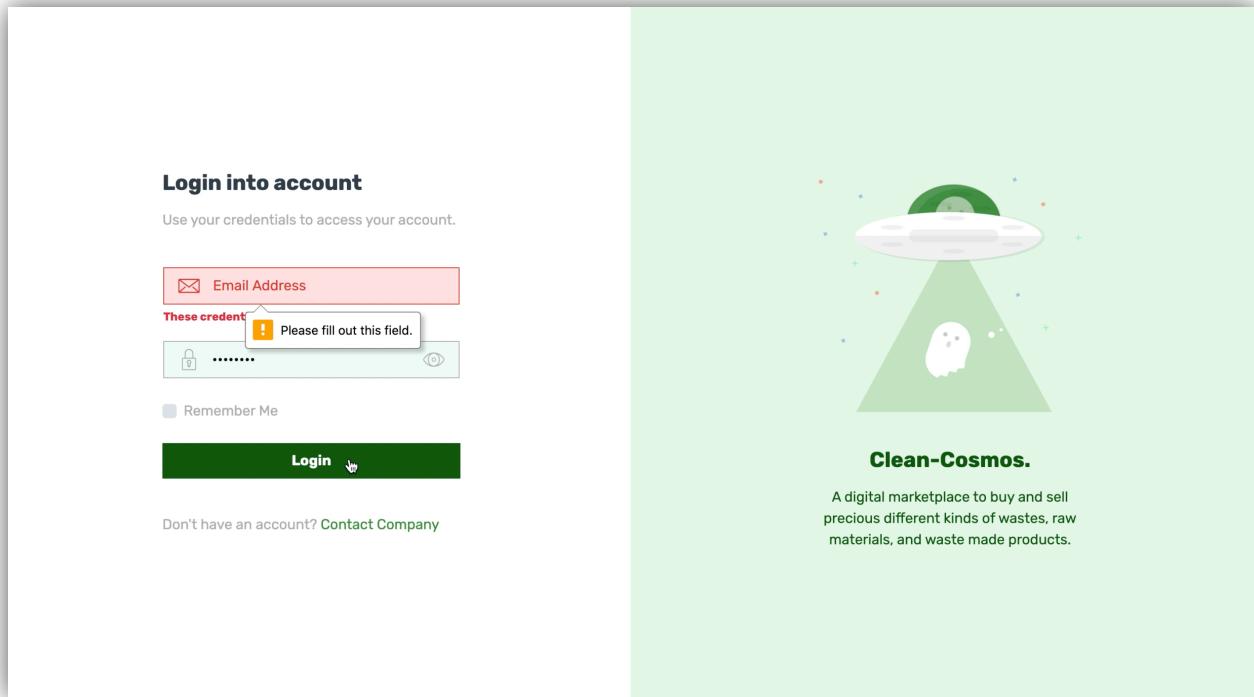


Figure 34 - Test case TC3 showing error message when user enters invalid username and password

4.2.1.2 Admin logout

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC4	Verify by clicking on sign out button	1. Click sign out button	User should be logged out from an application	Pass
TC5	Test if the user is authenticated or not	1. Click sign out button 2. Try to access application without login	So, user shouldn't be able to access application	Pass

Table 2 - Test cases for admin logout

Below show the screenshots for test case TC4 (verify by clicking on sign out button):

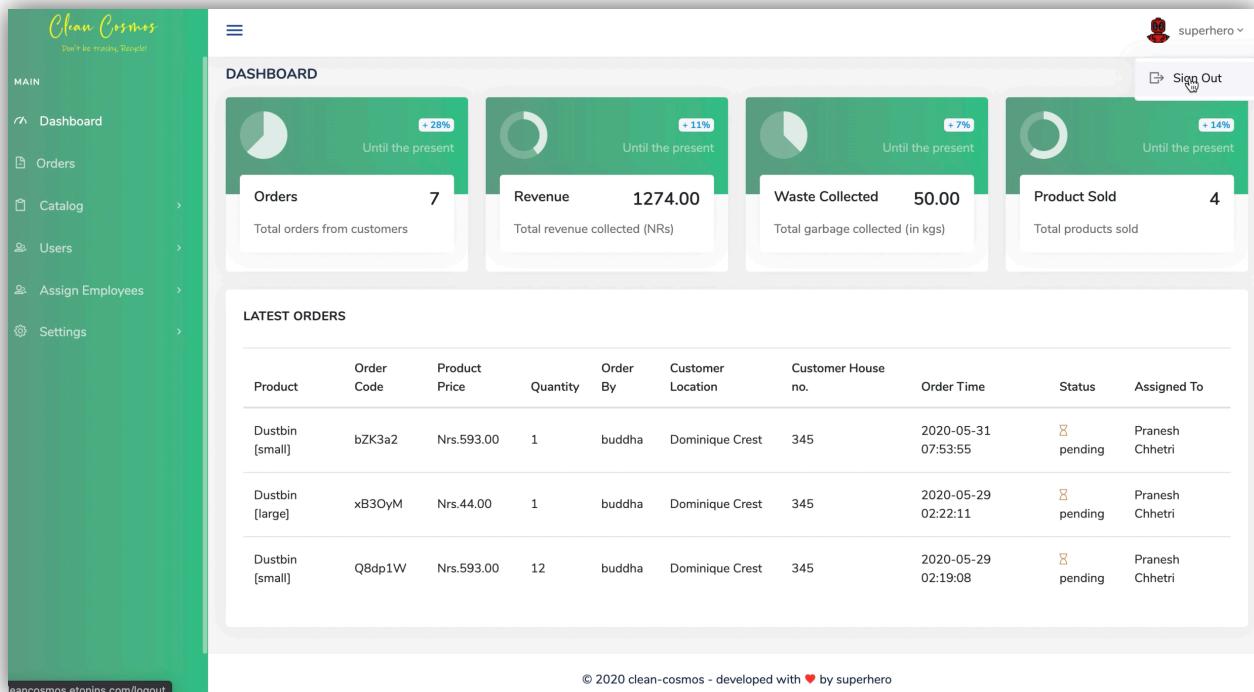


Figure 35 - Test case TC4 when user sign out from an application

Below show the screenshots for test case **TC5** (test if the user is authenticated or not):

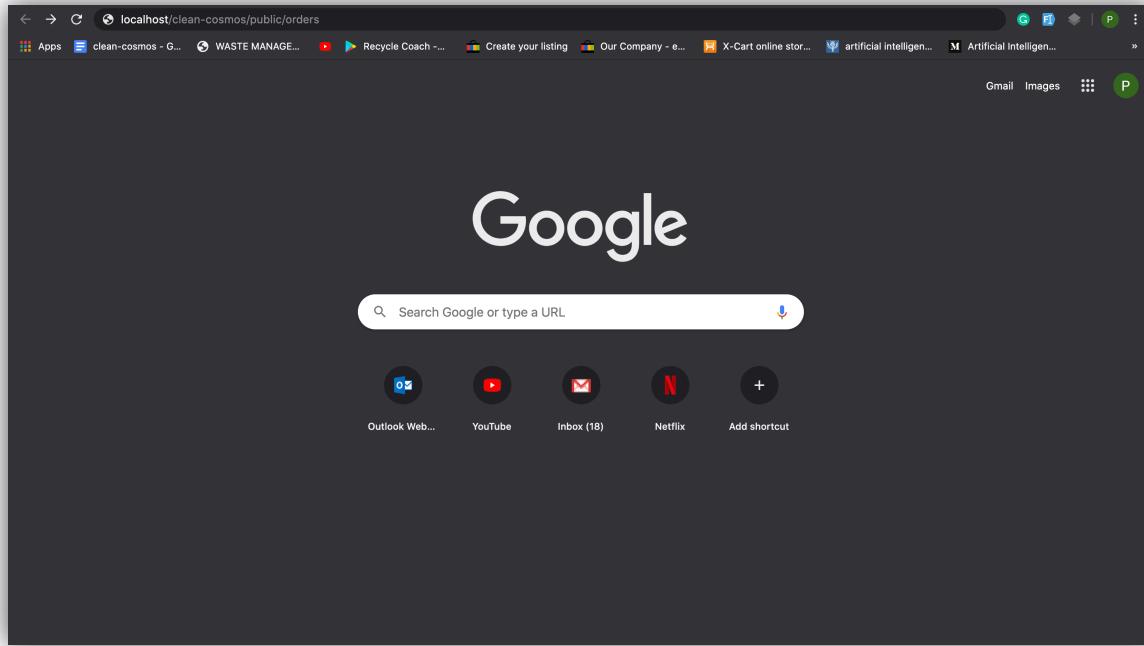


Figure 36 - Test case TC5 when a user tries to access the application after signing out from the system

When a user tries to access the application after signing out from the system then the system redirects the login page.

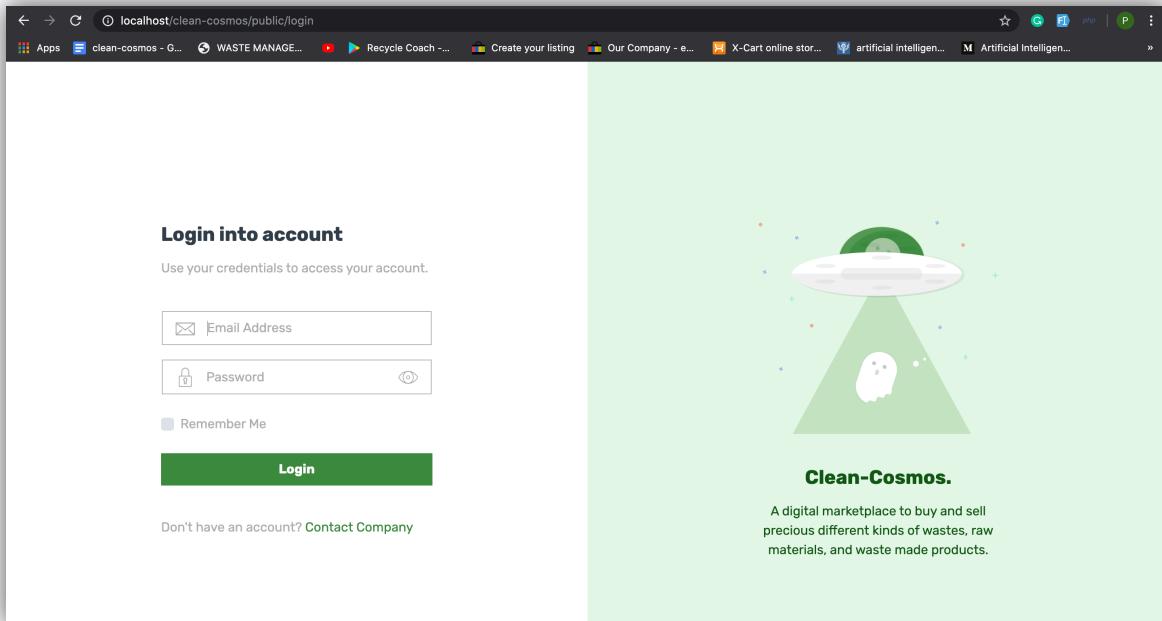


Figure 37 - Test case TC5 showing that system redirects the log in page

4.2.1.3 Catalog management - categories

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC6	Add category	1. Add category with details 2. Click submit button	Category should be added in the list	Pass
TC7	Field validation testing	1. Don't input the data 2. Click submit button	Category shouldn't be added, and error message should be shown	Pass
TC8	Edit category	1. Click edit button 2. Edit the details 3. Click submit button	Category should be updated	Pass
TC9	Delete category	1. Click delete button 2. System shows confirm message 3. Click OK	Category should be deleted	Pass

Table 3 - Test cases for catalog management - categories

Below show the screenshots for test case **TC6** (add category):

The screenshot shows a web-based form titled "ADD CATEGORIES". The form has the following fields:

- Category Name:** A text input field containing "Wood".
- Slug:** A text input field containing "wood".
- Description:** A rich text editor with a toolbar above it. The toolbar includes icons for bold, italic, underline, font size, alignment, lists, tables, and other common rich text functions. The description area contains the text "Reclaimed wood products.".
- Buttons:** At the bottom left are two buttons: "Submit" (highlighted in green) and "Cancel".

Figure 38 - Test case TC6 when user add category with details

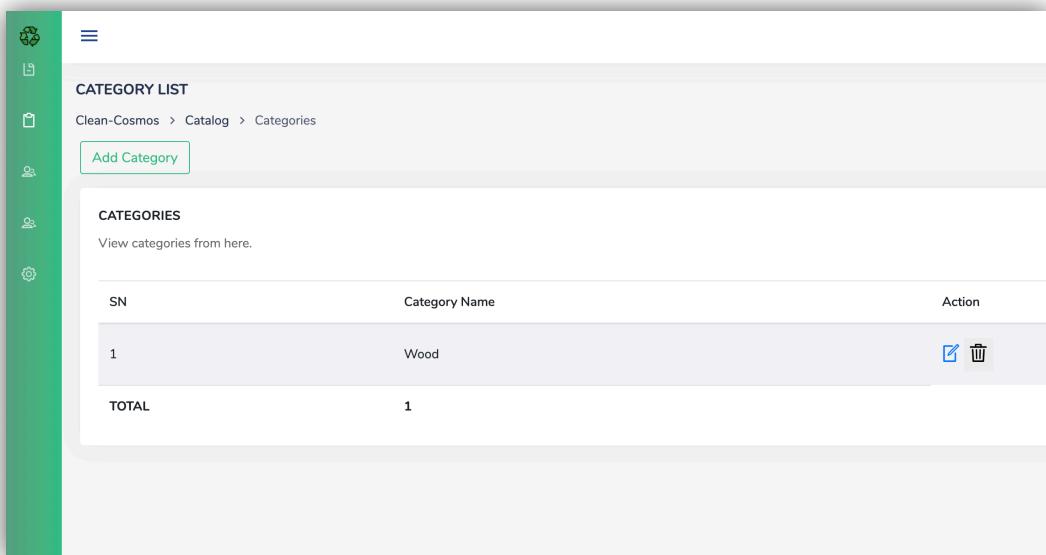


Figure 39 – Test case TC6 where category wood is added in the list

Below show the screenshots for test case **TC7** (field validation testing):

ADD CATEGORIES

Add category from here!

Category Name

Bag, Bottle, Laptop.

Slug

Bag, Bottle, Laptop.

The slug field is required.

Description

The description field is required.

Submit Cancel

Figure 40 - Test case TC7 when a user doesn't give inputs

Below show the screenshots for test case **TC8** (edit category):

ADD CATEGORIES

Add category from here!

Category Name

Slug

Description

Reclaimed wood products.

Rich text editor toolbar:

- B** (Bold)
- U** (Underline)
- A** (Font Style)
- Nunito
- Font Size
- Align Left
- Align Center
- Align Right
- Align Justify
- Image
- Link
- Unlink
- Cross
- Code View
- Help

Submit **Cancel**

Figure 41 - Test case TC8 when a user edits category

Updated Successfully

CATEGORY LIST

Clean-Cosmos > Catalog > Categories

Add Category

CATEGORIES

View categories from here.

SN	Category Name	Action
1	Woods	
TOTAL	1	

Figure 42 - Test case TC8 where category name to woods

Below show the screenshots for test case **TC9** (delete category):

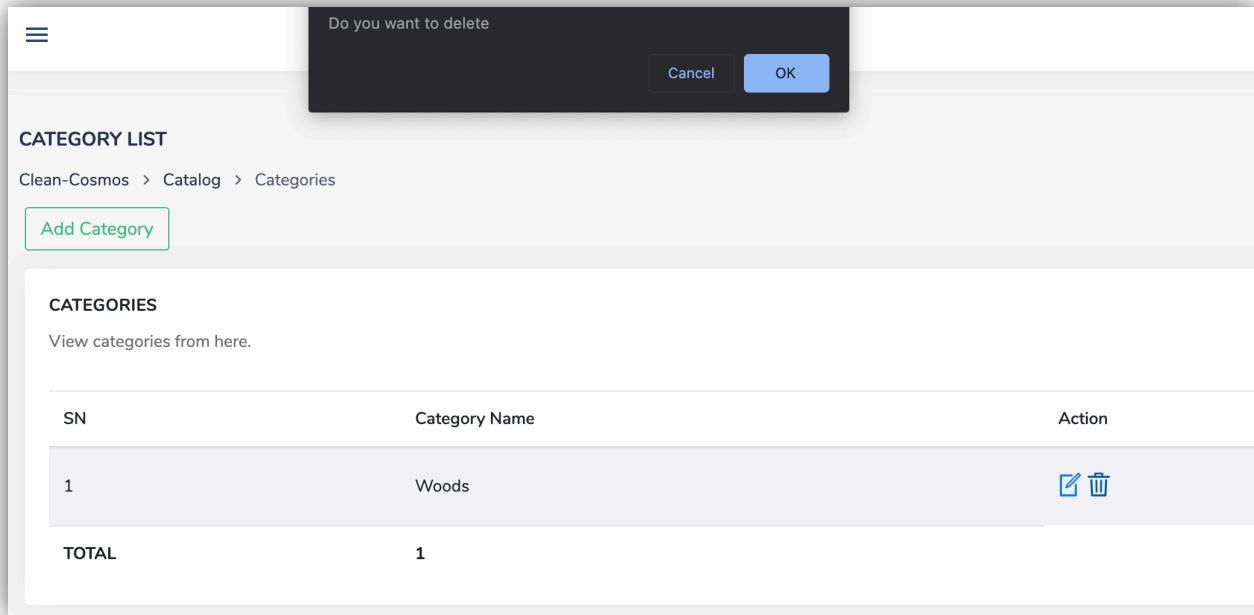


Figure 43 - Test case TC9 when user delete a category wood

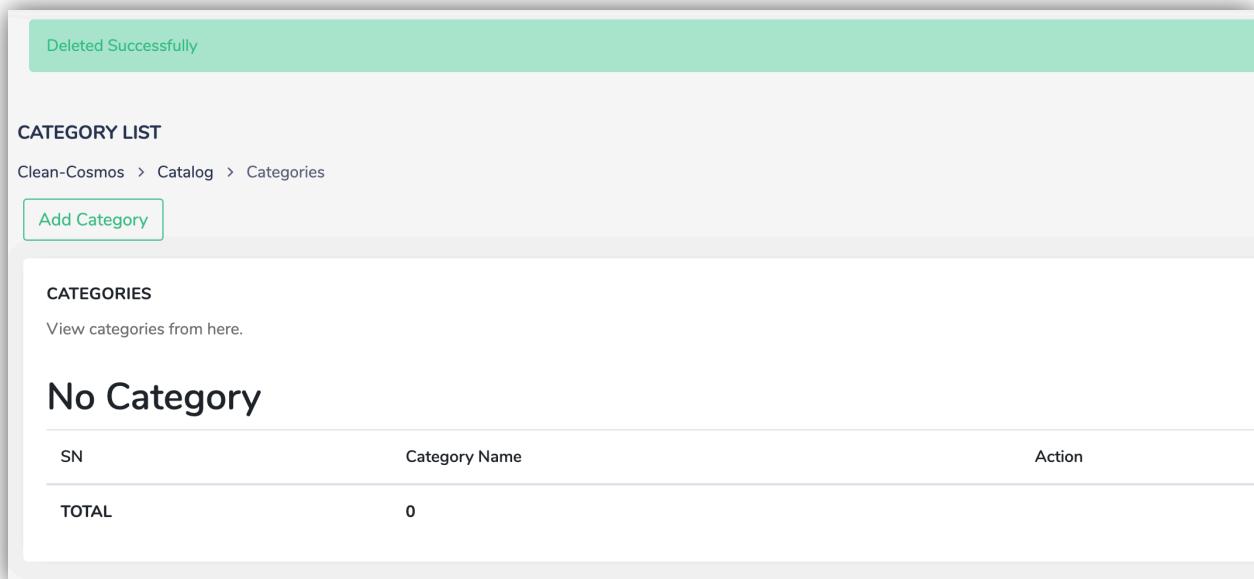


Figure 44 - Test case TC9 where category wood is deleted from the list

4.2.1.4 Catalog management – recycled products

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC10	Add recycled product	1. Add recycled product with details 2. Click submit button	Recycled product should be added in the list	Pass
TC11	Field validation testing	1. Don't give any input or fill invalid inputs 2. Click submit button	Recycled product shouldn't be added, and error message should be shown	Pass
TC12	Edit recycled product	1. Click edit button 2. Edit the details 3. Click submit button	Recycled product should be updated	Pass
TC13	Delete recycled product	1. Click delete button 2. System shows confirm message 3. Click OK	Recycled product should be deleted	Pass

Table 4 - Test cases for catalog management - recycle products

Below show the screenshots for test case **TC10** (add recycled product):

The screenshot displays a user interface for adding a new product. The form consists of several input fields and dropdown menus. Key data points include:

- Product Name:** ROUGH SAWN SHELF 2X4
- Category:** Wood
- Universal Product Code:** 03600029145x12
- Brand:** ROUGH SAWN SHELVES
- Color:** Red
- Source of waste:** Wood
- Size:** Medium
- Units:** Pcs
- Weight:** 100
- Available Quantity:** 25
- Purchase Price:** 3500
- Selling Price:** 5000
- Description:** Antique Rough Sawn Beam Shelf - 2-inch thickness x 4-inch depth. Antique Reclaimed Rough Sawn beam shelves provide a

Figure 45 - Test case TC10 when user add recycled product with details

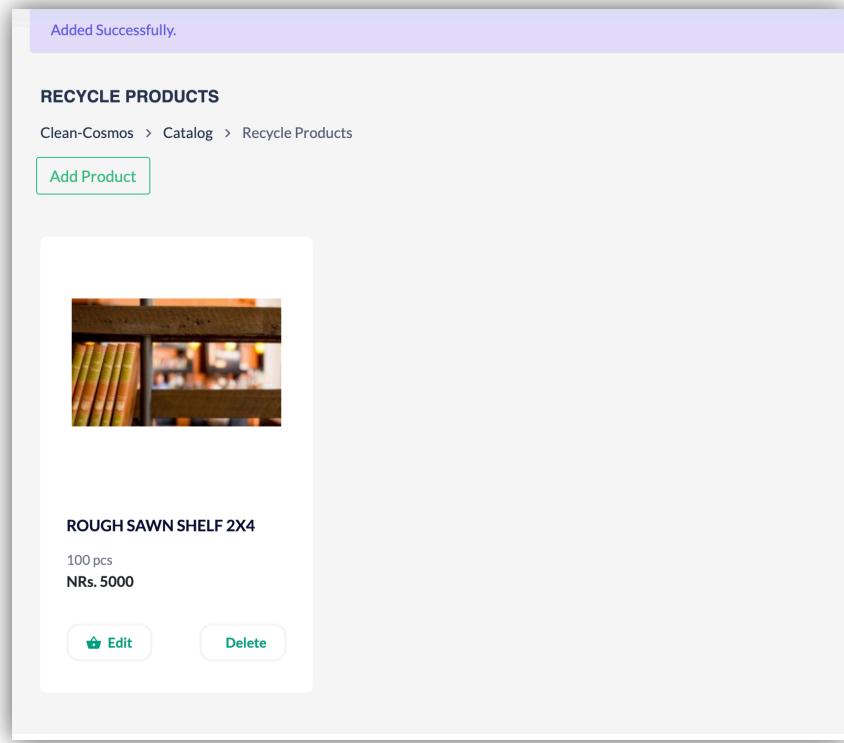


Figure 46 - Test case TC10 where recycled product Rough Sawn Shelf 2x4 is added

Below show the screenshots for test case **TC11** (field validation testing):

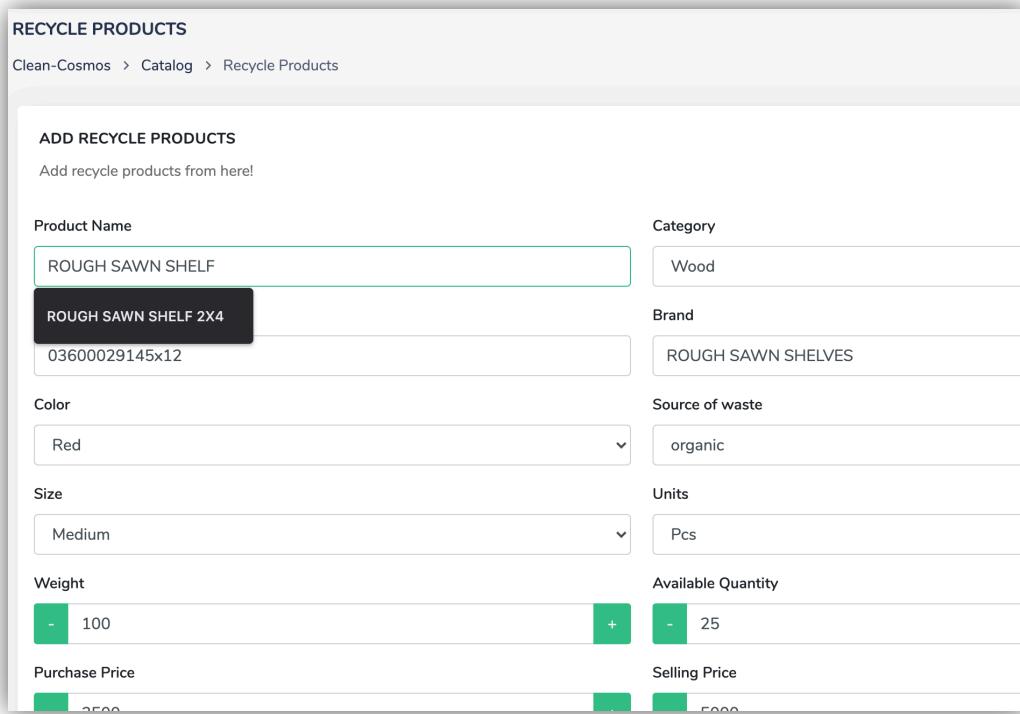
Here the unique field validation, mandatory field validation and invalid field validation was tested.

The screenshot shows a form for adding a new product. Several fields contain error messages:

- Product Name:** BROWN BARN WOOD DOOR (no error)
- Category:** Wood (no error)
- Universal Product Code:** 03600029145x12 (error: The universal product code has already been taken.)
- Brand:** BARN WOODS (no error)
- Color:** Select (error: The selected color is invalid.)
- Source of waste:** Select (error: The selected source of waste is invalid.)
- Size:** Medium (no error)
- Units:** Pcs (no error)
- Weight:** (input field with error: The weight capacity must be an integer.)
- Available Quantity:** 35 (input field with error: The weight capacity must be an integer.)
- Purchase Price:** 5000 (input field with error: The weight capacity must be an integer.)
- Selling Price:** 10000 (input field with error: The weight capacity must be an integer.)
- Upload Image:** Choose Files (No file chosen) (error: The images field is required.)
- Description:** (no error)

Figure 47 - Test case TC11 showing error message where invalid inputs were filled

Below show the screenshots for test case **TC12** (edit recycled product):



RECYCLE PRODUCTS

Clean-Cosmos > Catalog > Recycle Products

ADD RECYCLE PRODUCTS

Add recycle products from here!

Product Name	Category
ROUGH SAWN SHELF	Wood
ROUGH SAWN SHELF 2X4	Brand
03600029145x12	ROUGH SAWN SHELVES
Color	Source of waste
Red	organic
Size	Units
Medium	Pcs
Weight	Available Quantity
- 100	+ 25
Purchase Price	Selling Price
- 3500	+ 5000

Figure 48 - Test case TC12 where product name edited to Rough Sawn Shelf

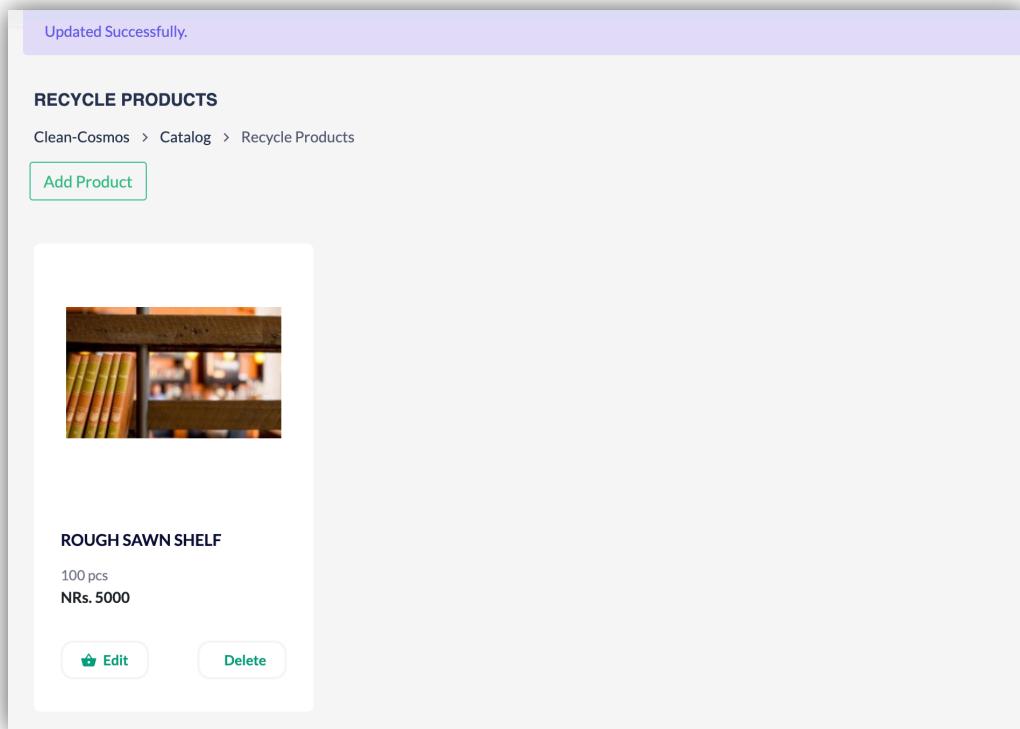


Figure 49 - Test case TC12 where product name updated to Rough Sawn Shelf

Below show the screenshots for test case **TC13** (delete recycle product):

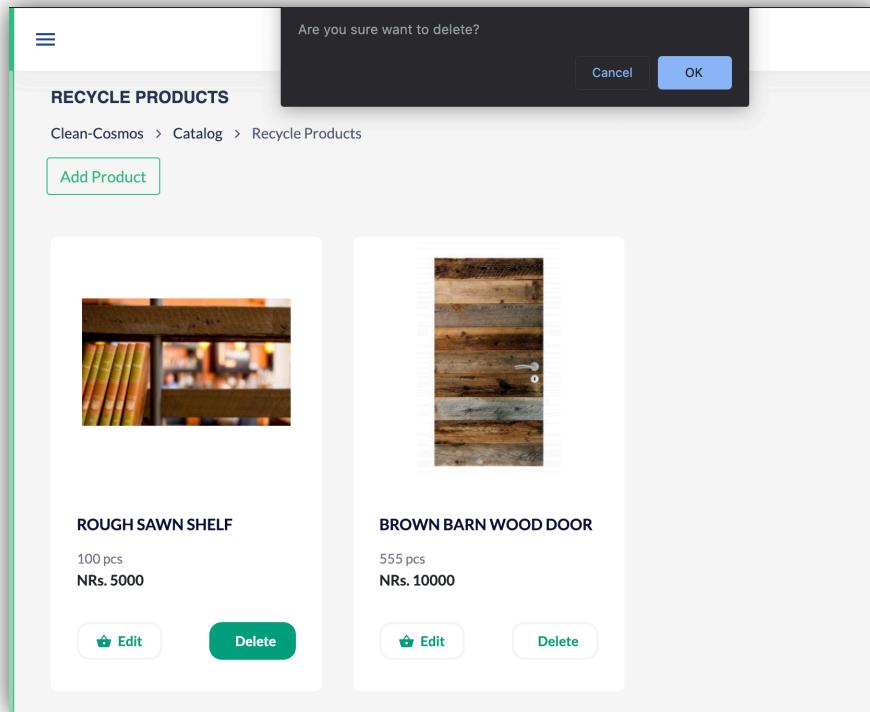


Figure 50 - Test case TC13 when user delete a recycled product

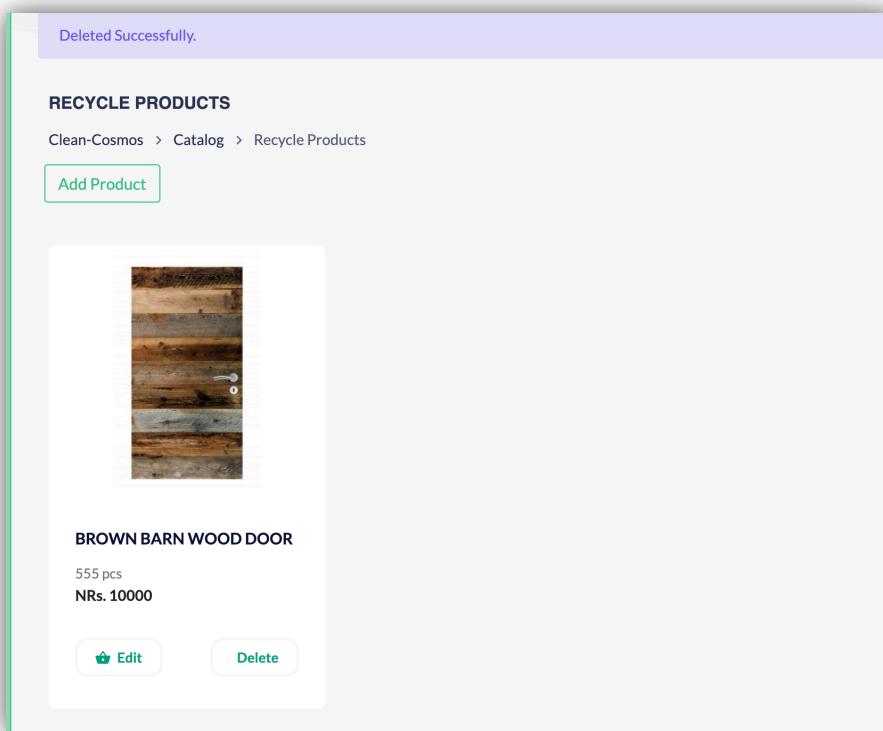


Figure 51 - Test case TC13 recycled product Rough Sawn Shelf is deleted

4.2.1.5 Catalog management – waste types

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC14	Add waste type	1. Add waste type with details 2. Click submit button	Waste type should be added in the list	Pass
TC15	Edit waste type	1. Click edit button 2. Edit the details 3. Click submit button	Waste type should be updated	Pass
TC16	Delete waste type	1. Click delete button 2. System shows confirm message 3. Click OK	Waste type should be deleted	Pass

Table 5 - Test cases for catalog management - waste types

Below show the screenshots for test case **TC14** (add waste type):

The screenshot shows a modal window titled "ADD WASTE TYPE". It contains the following fields:

- Waste Type:** A dropdown menu set to "Plastic Waste".
- Container Color:** A dropdown menu set to "Red".
- Description:** A rich text editor with a toolbar containing icons for bold, italic, underline, etc. The editor's content area displays the text: "Plastic is the general common term for a wide range of synthetic or semi-synthetic organic amorphous solid materials derived from oil and natural gas. Here, plastic waste is represented as a red color."
- Buttons:** At the bottom left are "Submit" and "Cancel" buttons.

Figure 52 - Test case 14 when user adds the waste type

Added Successfully.

WASTE TYPE

Clean-Cosmos > Catalog > Waste Type

Add Waste Type

S.N	Waste Type	Container Color	Description	Action
1	Plastic Waste	red	Plastic is the general common term for a wide range of synthetic or semi-synthetic organic amorphous solid materials derived from oil and natural gas. Here, plastic waste is represented as a red color.	

Figure 53 - Test case TC14 where Plastic waste type is added

Below show the screenshots for test case **TC15** (edit waste type):

ADD WASTE TYPE

Add Waste Type from here!

Waste Type Container Color

Plastic Waste Blue

Description

Plastic is the general common term for a wide range of synthetic or semi-synthetic organic amorphous solid materials derived from oil and natural gas. Here, plastic waste is represented as a red color.

Submit Cancel

Figure 54 - Test case TC15 when user edit container color to blue

Updated Successfully.

WASTE TYPE

Clean-Cosmos > Catalog > Waste Type

Add Waste Type

S.N	Waste Type	Container Color	Description	Action
1	Plastic Waste	blue	Plastic is the general common term for a wide range of synthetic or semi-synthetic organic amorphous solid materials derived from oil and natural gas. Here, plastic waste is represented as a red color.	

Figure 55 - Test case TC15 where waste type's container color updated to blue

Below show the screenshots for test case **TC16** (delete waste type):

Are you sure want to delete?

Cancel **OK**

WASTE TYPE

Clean-Cosmos > Catalog > Waste Type

Add Waste Type

S.N	Waste Type	Container Color	Description	Action
1	Plastic Waste	blue	Plastic is the general common term for a wide range of synthetic or semi-synthetic organic amorphous solid materials derived from oil or natural gas. Here, plastic waste is represented as a red color.	

Figure 56 - Test case TC16 when a user deletes a waste type

Deleted Successfully.

WASTE TYPE

Clean-Cosmos > Catalog > Waste Type

Add Waste Type

S.N	Waste Type	Container Color	Description	Action
No Waste Types. Please Add!!				

Figure 57 - Test case TC16 where plastic waste type was deleted

4.2.1.6 Catalog management – waste containers

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC17	Add waste container	1. Add waste container with details 2. Click submit button	Waste container should be added in the list	Pass
TC18	Field validation testing	1. Don't input the data 2. Click submit button	Waste container shouldn't be added, and error message should be shown	Pass
TC19	Edit waste container	1. Click edit button 2. Edit the details 3. Click submit button	Waste container should be updated	Pass
TC20	Delete waste container	1. Click delete button 2. System shows confirm message 3. Click OK	Waste container should be deleted	Pass

Table 6 - Test cases for catalog management - waste container

Below show the screenshots for test case **TC17** (add waste container):

The screenshot shows a web-based application interface for adding a waste container. The title bar says "ADD WASTE CONTAINERS". Below it, a sub-instruction says "Add Waste Containers from here!". The form has several input fields:

- "Containers Size": A dropdown menu showing "Medium".
- "Container Weight/Capacity": An input field containing "50".
- "Purchase Price": An input field containing "500".
- "Selling Price": An input field containing "750".
- "Available Quantity": An input field containing "30".
- "Description": A rich text editor toolbar with various icons for bold, italic, underline, etc. Below the toolbar, the text area contains: "The 2 Wheel Waste Disposal Containers guarantee maximum functionality in use, with ad hoc connection for such distinct essentials as handling and storing small metal products or exquisite objects. Various types of ideas ensure excellent space usage. Lastly, the modular assembled parts are made for easy conveying."

At the bottom of the form are two buttons: "Submit" and "Cancel".

Figure 58 - Test case TC17 when user adds a waste container

Added Successfully.

WASTE CONTAINERS

Clean-Cosmos > Catalog > Waste Containers

Add Waste Containers

S.N	Container Size	Weight Capacity	Purchase Price	Selling Price	Available Quantity	Description	Action
1	medium	50	500.00	750.00	30	The 2 Wheel Waste Disposal Containers guarantee maximum functionality in use, with ad hoc connection for such distinct essentials as handling and storing small metal products or exquisite objects. Various types of ideas ensure excellent space usage. Lastly, the modular assembled parts are made for easy conveying.	

Figure 59 - Test case TC17 where medium size waste container added

Below show the screenshots for test case **TC18** (field validation testing)

ADD WASTE CONTAINERS

Add Waste Containers from here!

Containers Size	Container Weight/Capacity	Purchase Price
Medium	<input type="text"/>	<input type="text"/>
The selected size is invalid.		
Selling Price	Available Quantity	
<input type="text"/>	<input type="text"/>	
The selling price field is required.		
The available quantity field is required.		
Description 		

Figure 60 - Test case TC18 when a user doesn't give an input

Below show the screenshots for test case **TC19** (edit waste container):

The screenshot shows a form titled "ADD WASTE CONTAINERS" with the sub-instruction "Add Waste Containers from here!". It contains several input fields and a rich text editor.

Containers Size	Container Weight/Capacity	Purchase Price
Large	50	500.00

Selling Price	Available Quantity
750.00	30

Description:

The 2 Wheel Waste Disposal Containers guarantee maximum functionality in use, with ad hoc connection for such distinct essentials as handling and storing small metal products or exquisite objects. Various types of ideas ensure excellent space usage. Lastly, the modular assembled parts are made for easy conveying.

Buttons:

- Submit (highlighted in green)
- Cancel

Figure 61 - Test case TC19 when user edits a waste container size to large

The screenshot shows a catalog page for "WASTE CONTAINERS". At the top, there is a success message: "Updated Successfully." Below it, the breadcrumb navigation shows "Clean-Cosmos > Catalog > Waste Containers". A green button labeled "Add Waste Containers" is visible.

WASTE CONTAINERS

S.N	Container Size	Weight Capacity	Purchase Price	Selling Price	Available Quantity	Description	Action
1	large	50	500.00	750.00	30	The 2 Wheel Waste Disposal Containers guarantee maximum functionality in use, with ad hoc connection for such distinct essentials as handling and storing small metal products or exquisite objects. Various types of ideas ensure excellent space usage. Lastly, the modular assembled parts are made for easy conveying.	

Figure 62 - Test case TC19 where waste container size updated to large

Below show the screenshots for test case **TC20** (delete waste container):

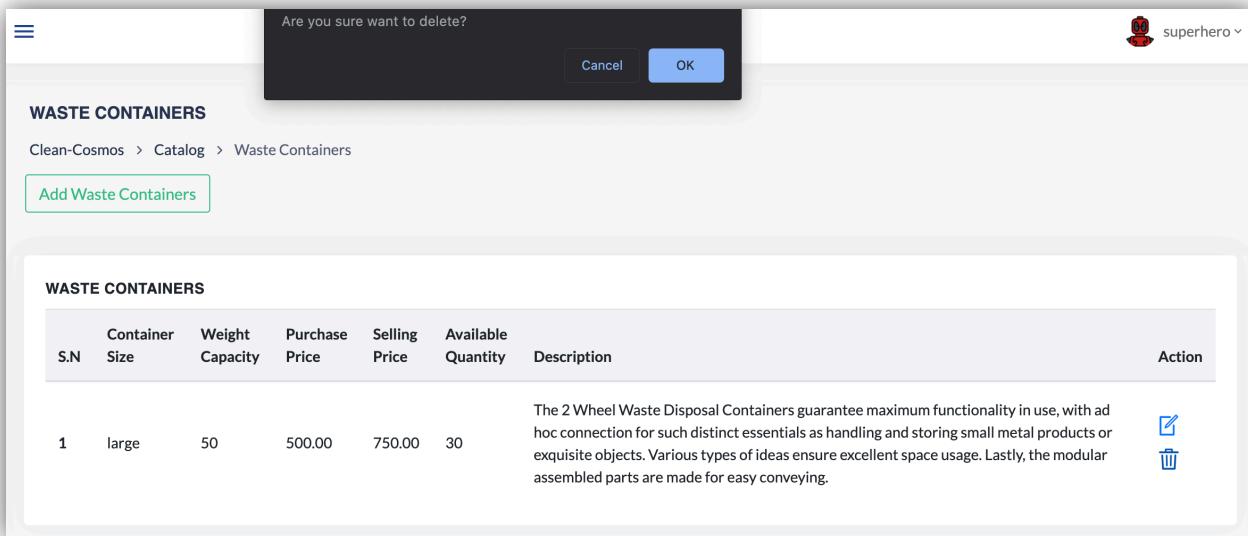


Figure 63 - Test case TC20 when user deletes a waste container

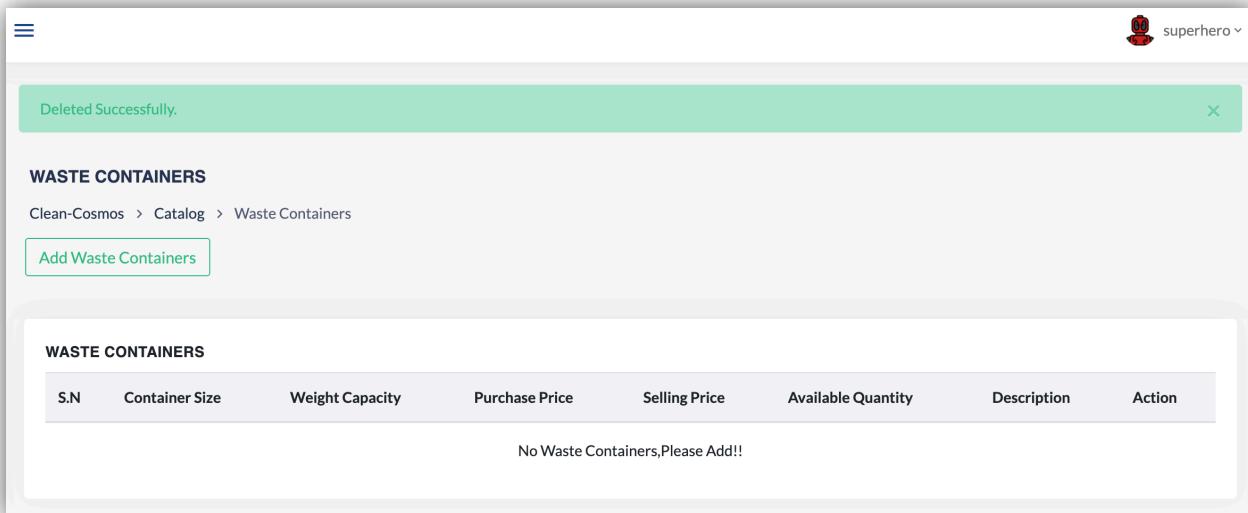


Figure 64 - Test case TC20 where waste container is deleted

4.2.1.7 User management – employees

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC21	Add employee	1. Add employee with details 2. Select role of an employee 3. Click submit button	Employee should be added in the list	Pass
TC22	Field validation testing	1. Don't input the data 2. Click submit button	Employee shouldn't be added, and error message should be shown	Pass
TC23	Edit employee	1. Click edit button 2. Edit the details 3. Click submit button	Employee should be updated	Pass
TC24	Delete employee	1. Click delete button 2. System shows confirm message 3. Click OK	Employee should be deleted	Pass

Table 7 - Test cases for user management – employees

Below show the screenshots for test case **TC21** (add employee):

The screenshot shows a web-based application interface titled "ADD EMPLOYEES". The page instructs the user to "Add employees from here!". It contains several input fields and dropdown menus:

- Name:** Bill Gates
- Contact:** 9856875467
- Email:** billgates@microsoft.com
- Role:** waste collector
- Username:** billgates
- Password:** ***** (represented by five asterisks)

At the bottom of the form are two buttons: "Submit" and "Cancel".

Figure 65 - Test case TC21 when a user adds an employee

Created Successfully

EMPLOYEES

Clean-Cosmos > Users > Employees

[Add Employee](#)

EMPLOYEES

View employees from here.

Name	Position	Username	Contact	Email	Action
Bill Gates	waste collector	billgates	9856875467	billgates@microsoft.com	

Figure 66 - Test case 21 when a user added an employee

Below show the screenshots for test case **TC22** (field valid testing):

ADD EMPLOYEES

Add employees from here!

Name	Contact
<input type="text"/>	<input type="text"/>
The name field is required.	The phone no field is required.
Email	Role
<input type="text"/>	Select
The email field is required.	The role id field is required.
Username	Password
<input type="text"/>	<input type="password"/>
The username field is required.	The password field is required.

[Submit](#) [Cancel](#)

Figure 67 - Test case TC22 when a user doesn't give input

Below show the screenshots for test case **TC23** (edit employee):

ADD EMPLOYEES

Add employees from here!

Name	Contact
Bill Bahadur Gates	9856875467
Email	Role
billgates@microsoft.com	waste collector
Username	Password
billgates	

Submit **Cancel**

Figure 68 - Test case TC23 when a user edits employee name to Bill Bahadur Gates

Updated successfully

EMPLOYEES

Clean-Cosmos > Users > Employees

Add Employee

EMPLOYEES

View employees from here.

Name	Position	Username	Contact	Email	Action
Bill Bahadur Gates	waste collector	billgates	9856875467	billgates@microsoft.com	

Figure 69 - Test case TC23 where an employee name is updated to Bill Bahadur Gates

Below show the screenshots for test case **TC24** (delete employee):

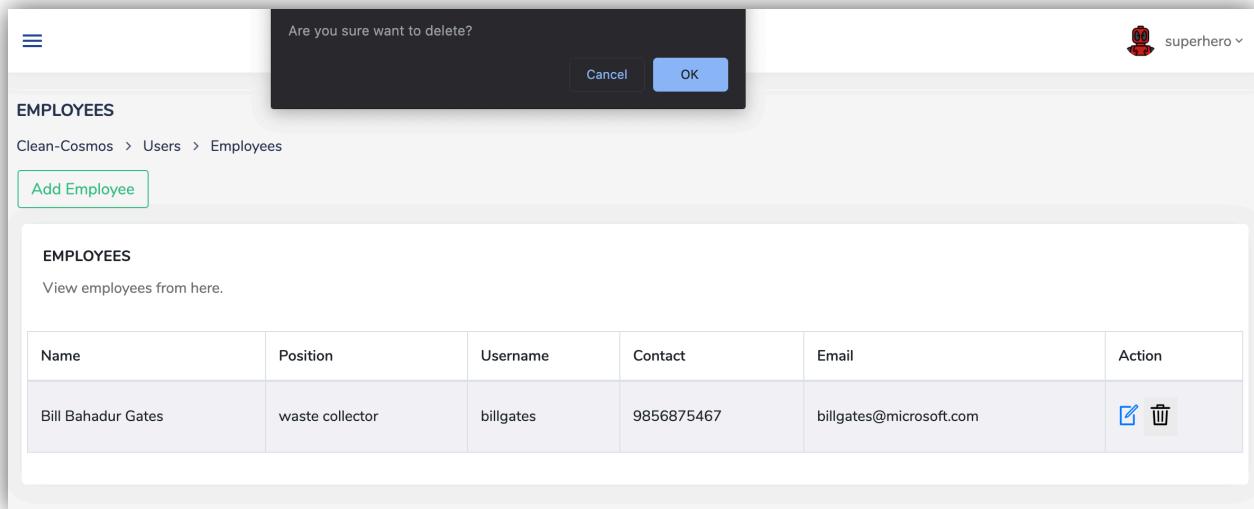


Figure 70 - Test case 24 when a user deletes an employee

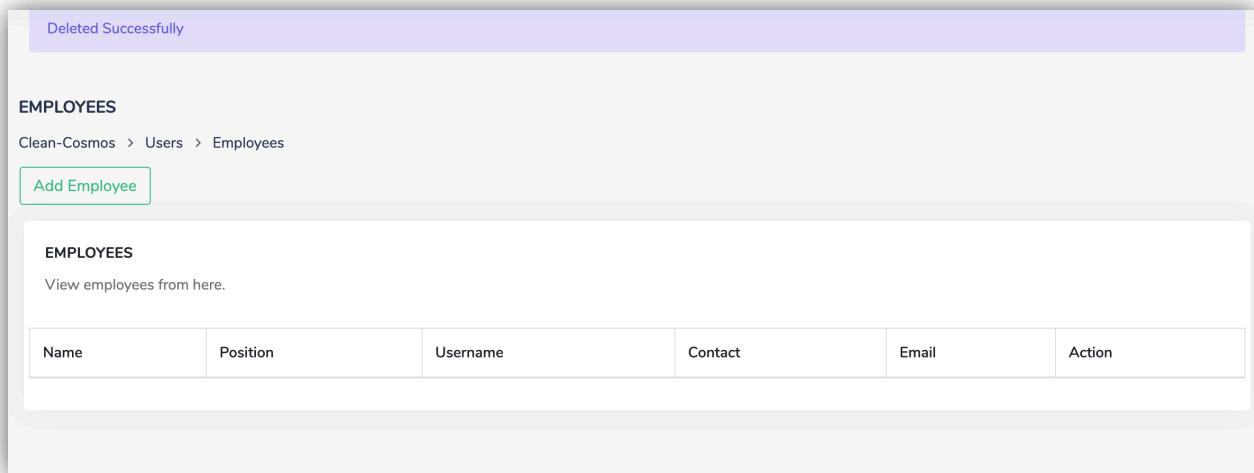


Figure 71 - Test case TC24 where a user deleted an employee

4.2.2 Mobile Application

4.2.2.1 Mobile application login (Customer, Waste Collector and Delivery Person)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC25	User login with valid data	1. Enter username and password 2. Click login button	User should be logged into the mobile application	Pass
TC26	User login with invalid or empty data	1. Enter invalid or empty data 2. Click login button	User shouldn't be logged into the mobile application and error message should be shown	Pass

Table 8 - Test case for Mobile application login (Customer, Waste Collector and Delivery Person)

Base URL for log in API: baseURL/api/v1/login (Method: **POST**)

Below show the screenshots for test case **TC25** (user login with valid data):

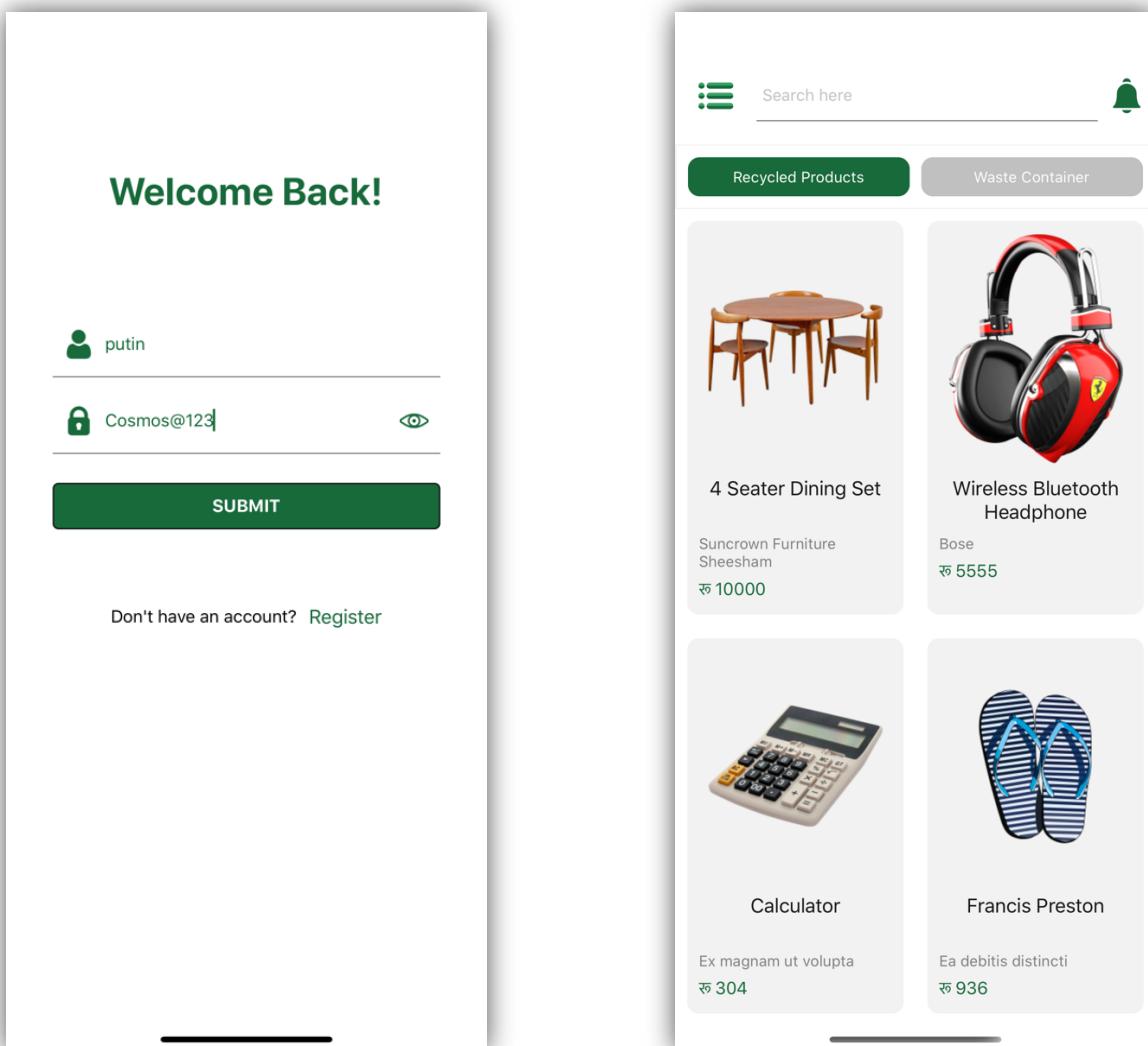


Figure 72 - Test case T25 when a user enters a valid username and password

Below show the screenshots for test case **TC26** (user login with invalid or empty data):

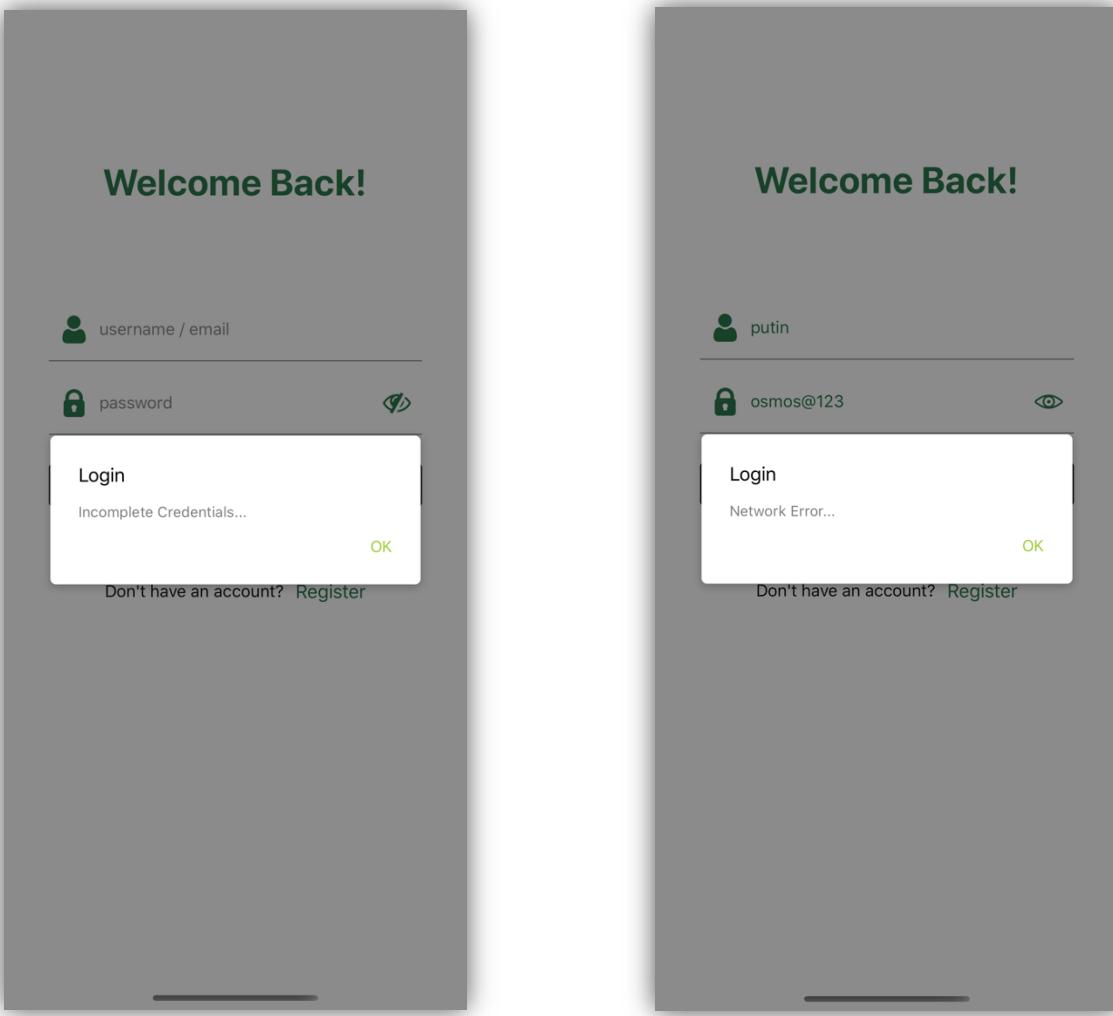


Figure 73 - Test case TC26 showing error message when a user enters invalid or empty data

4.2.2.2 Mobile application logout (Customer, Waste Collector and Delivery Person)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC27	User logout from mobile application	1. Click logout button on navigation drawer 2. Click Yes	User should be logged out from the mobile application	Pass

Table 9 – Test case for Mobile application logout (Customer, Waste Collector and Delivery Person)

Base URL for log in API: baseURL/api/v1/logout (Method: **POST**)

Below show the screenshots for test case **TC27** (user logout from mobile application):

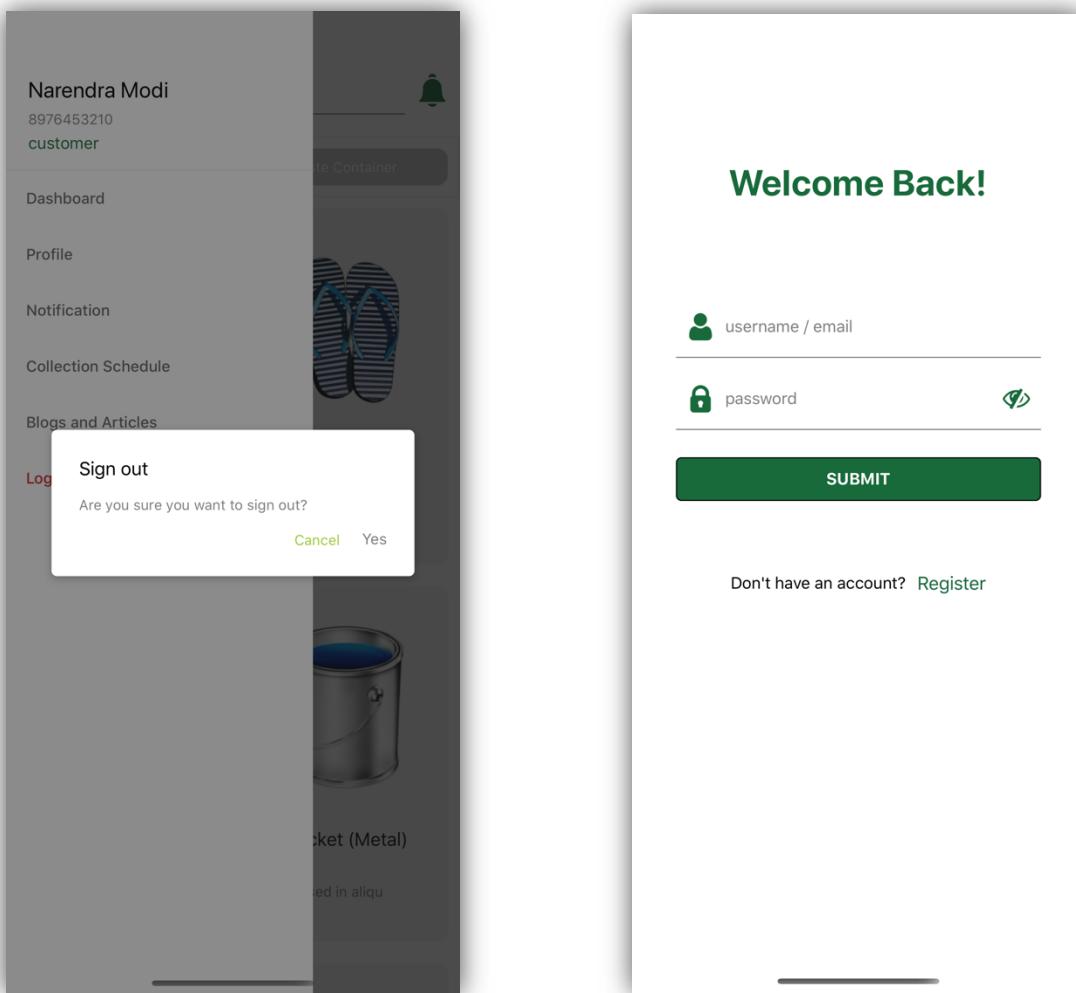


Figure 74 - Test case TC27 where a user logged out from application

4.2.2.3 Blogs and Articles

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC28	Customers can read articles and blogs related to waste management and recycling	1. Click the Blogs and Articles on navigation drawer. 2. Click any of the Blogs or Articles	Blogs or Articles should be displayed on the screen	Pass

Table 10 - Test case for Blogs and Articles

Below show the screenshots for test case **TC28** (Customers can read articles and blogs related to waste management system and recycling):

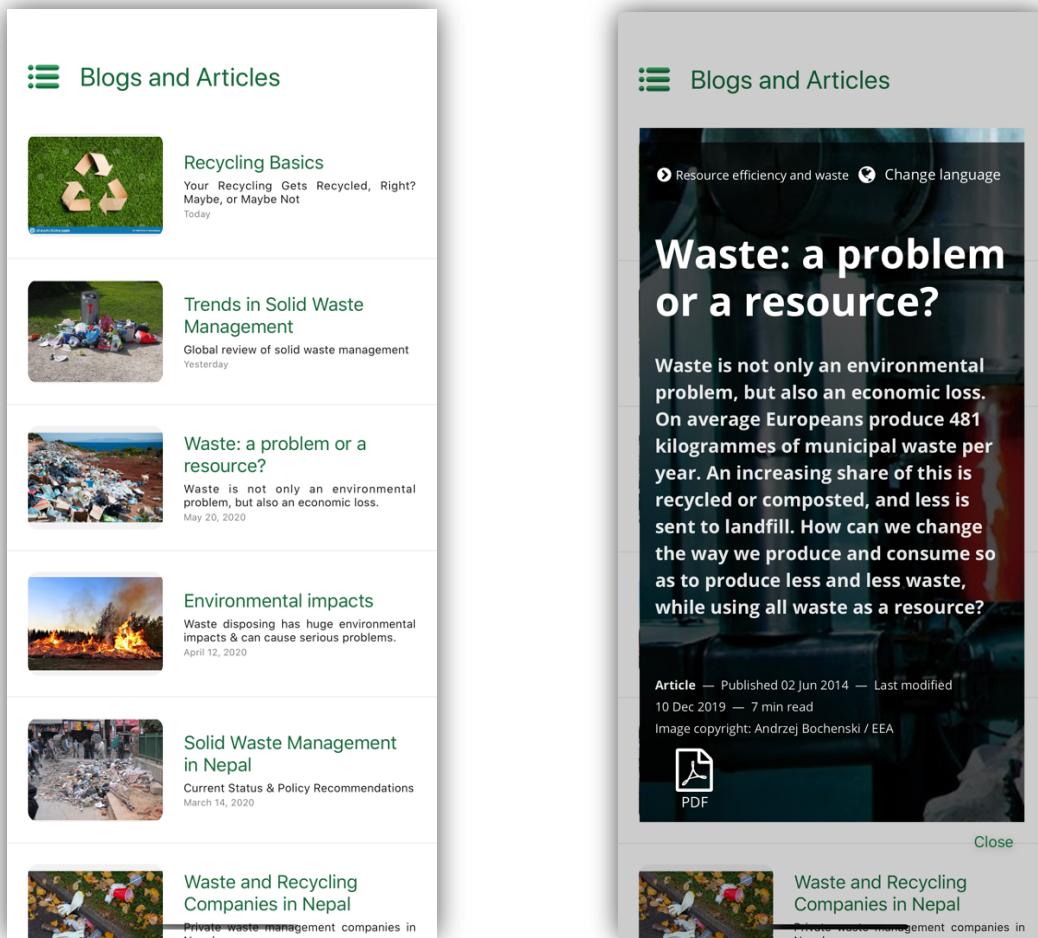


Figure 75 - Test case TC28 where customers can read articles and blogs related to waste management and recycling

4.3 Integration Testing

Below show all the test cases where all the APIs or integrated module of the application are tested.

4.3.1. Mobile application sign-up (*customer*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC29	Customer Sign Up with valid data	1. Enter the valid details of yours 2. Click submit button 3. Choose waste type and waste container or skip 4. Click submit button 5. Enter house number and search street 6. Click register button	Customer should be register to the database and customer data should be shown in the web application	Pass
TC30	Customer Sign Up with invalid or empty data	1. Enter invalid or empty data 2. Click register button	Customer shouldn't be register and error message should be shown	Pass
TC31	Verify whether the password and confirm password are same or not	1. Enter different password 2. Click submit button	Customer shouldn't be register and error message should be shown	Pass
TC32	Verify whether the username is unique or not	1. Enter the username which is already exists 2. Click register button	Customer shouldn't be register and error message should be shown	Pass

Table 11 - Test case for Mobile application sign-up (*customer*)

Base URL for sign up APIs are as below:

- 1. General details:** baseURL/api/v1/signup (Method: **POST**)
- 2. Choose waste types:** baseURL/api/v1/resource/waste-types (Method: **GET**)
- 3. Choose waste container:** baseURL/api/v1/resource/waste-containers (Method: **GET**)
- 4. Search streets:** baseURL/api/v1/resource/street/search (Method: **GET**)

Below show the screenshots for test case **TC29** (Customer sign up with valid data):

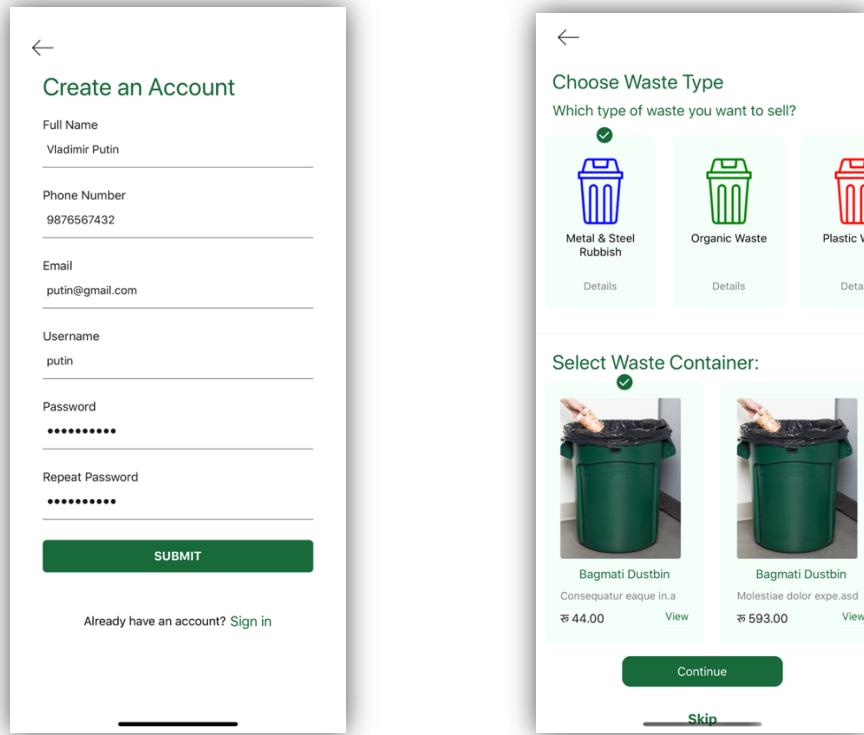


Figure 76 - Test case TC29 when a user signs up with valid data

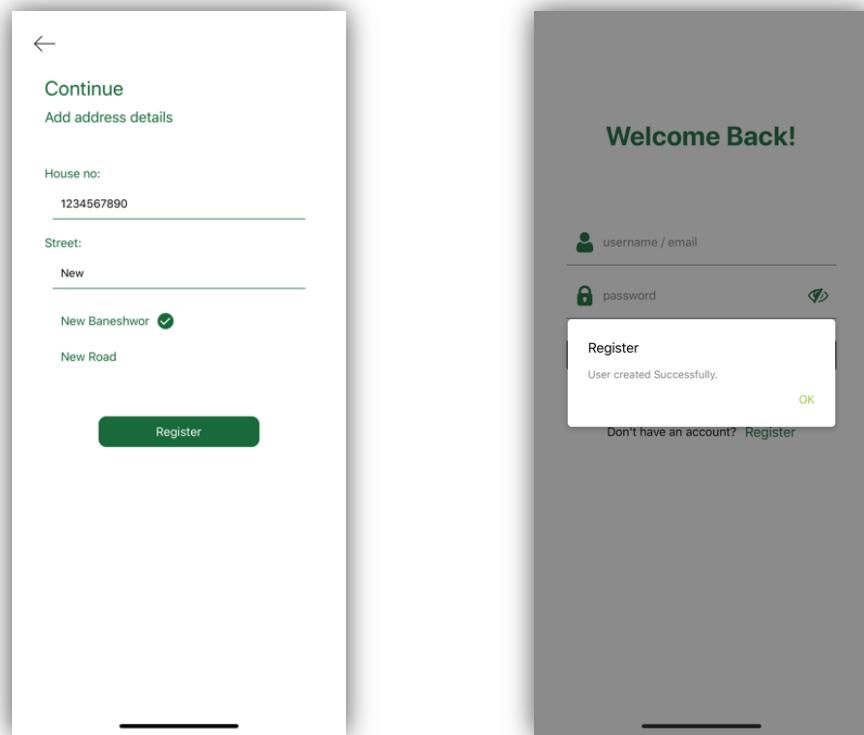


Figure 77 - Test case TC29 where a user successfully created

CUSTOMERS						
View customers from here.						
Show 10 entries						
Search: <input type="text"/>						
Name	Username	Email	Phone no.	Location	House no.	Waste Container
Jeff Bezos	jbezos	jbezos@amazon.com	9876452310	New Baneshwor	1234567890	Metal & Steel Rubbish[small]
Vladimir Putin	putin	putin@gmail.com	9876567432	New Baneshwor	1234567890	Metal & Steel Rubbish[small]

Showing 1 to 2 of 2 entries

Previous 1 Next

Figure 78 - Test case TC29 where a customer data is shown in the admin panel

Below show the screenshots for test case **TC30** (Customer sign up with invalid or empty data):

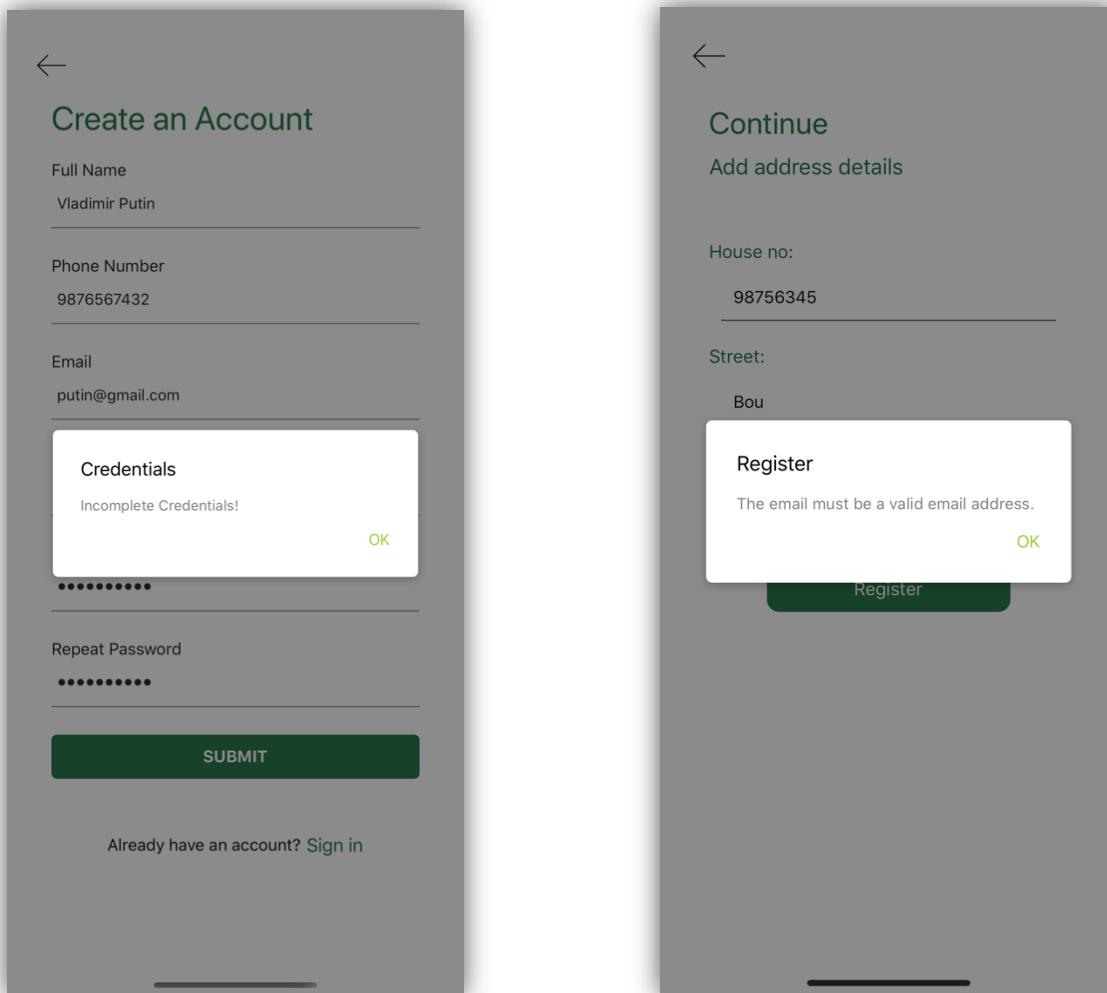


Figure 79 - Test case TC30 showing an error message when a user enters a invalid or empty data

Below show the screenshots for test case **TC31** (Verify whether the password and confirm password are same or not) and **TC32** (Verify whether the username is unique or not):

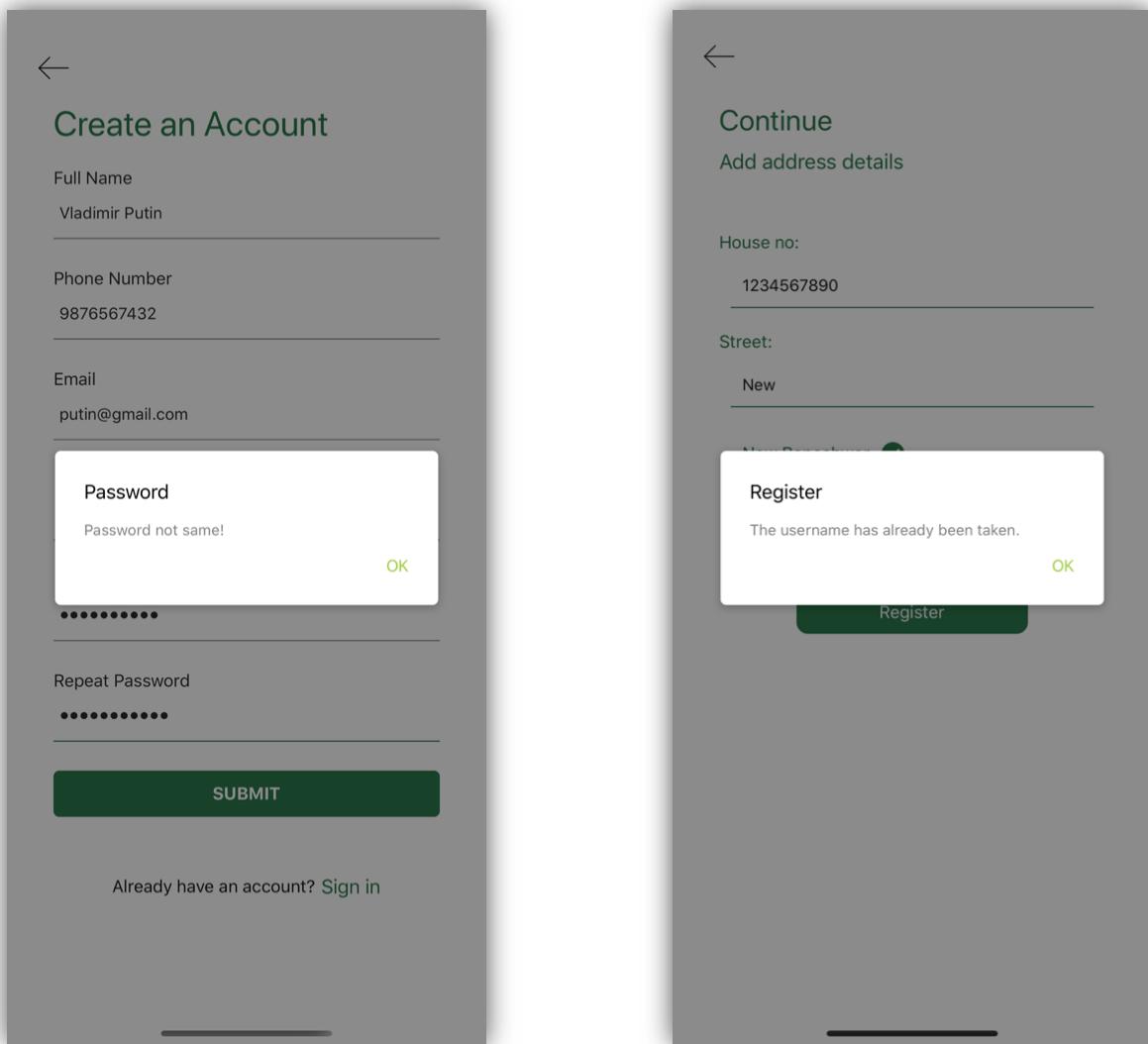


Figure 80 - Test case TC31 where password and confirm password are not same and TC32 when a user enters the username which is already exists

4.3.2 Mobile application – view products list (*customer*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC33	Verify the product list which were added from web application	1. Login to the application 2. View products list	All the products which were added should be displayed	Pass
TC34	Search product	1. Enter product name in the search bar	Search results displayed should be relevant to search keyword	Pass

Table 12 - Test case for Mobile application – view products list (*customer*)

Base URL for recycle products APIs are as below:

1. **Recycle products list:** baseURL/api/v1/recycle-product (Method: **GET**)
2. **Search recycle products:** baseURL/api/v1/recycle-product/search?keyword=drake (Method: **GET**)

Below show the screenshots for test case **TC33** (verify the product which were added from the web application):

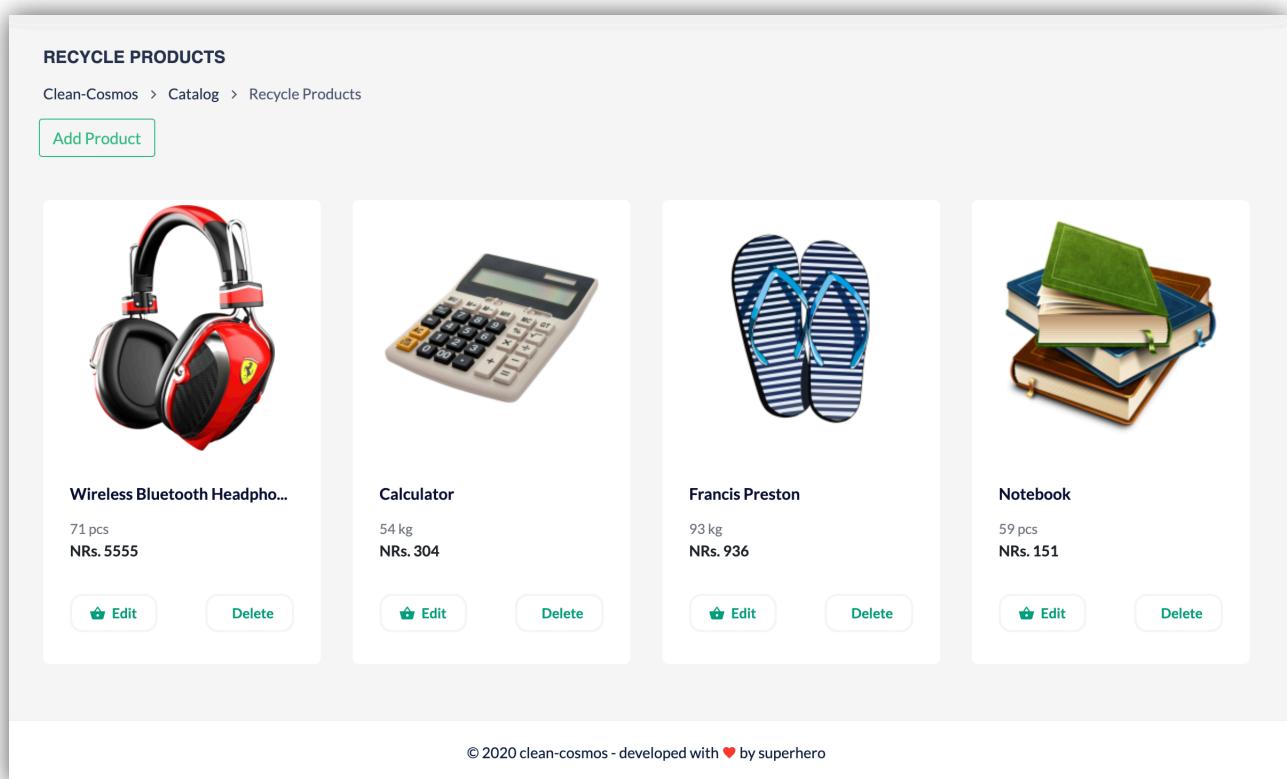


Figure 81 - Test case TC32 products list in the web application

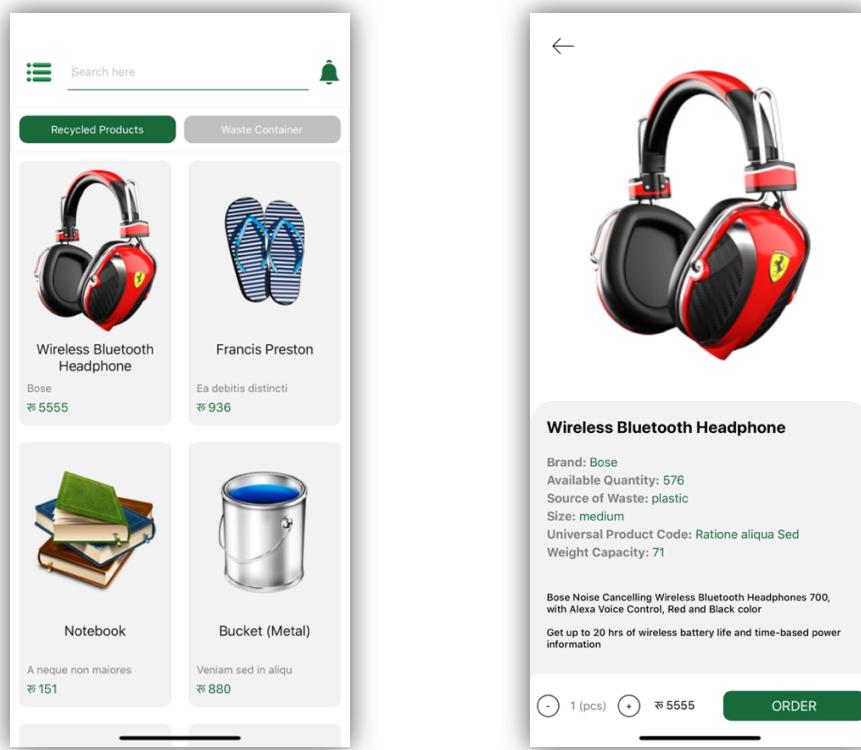


Figure 82 - Test case TC32 all the products are displayed

Below show the screenshots for test case **TC34** (search product):

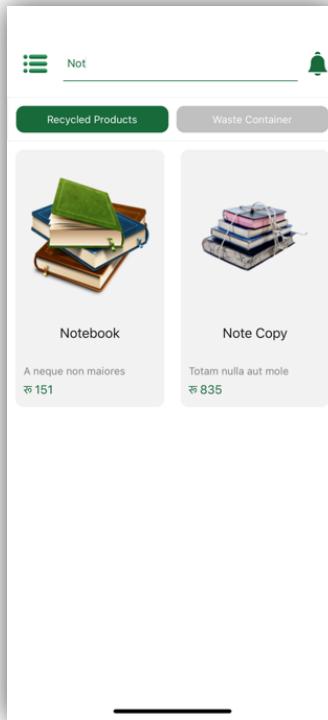


Figure 83 - Test case TC33 where note was searched from the product list

4.3.3 Order products (*customer*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC35	Customer can order recycled products from mobile application	1. Click on the product 2. Choose quantity 3. Click order button	Ordered recycled product should be displayed in the admin panel	Pass
TC36	Customer can order waste container from mobile application	1. Click waste container 2. Choose waste type 3. Choose waste container 4. Click order button	Ordered waste container should be displayed in the admin panel	Pass

Table 13 - Test case for Order products (*customer*)

Below show the screenshots for test case TC35 (order products from mobile application):

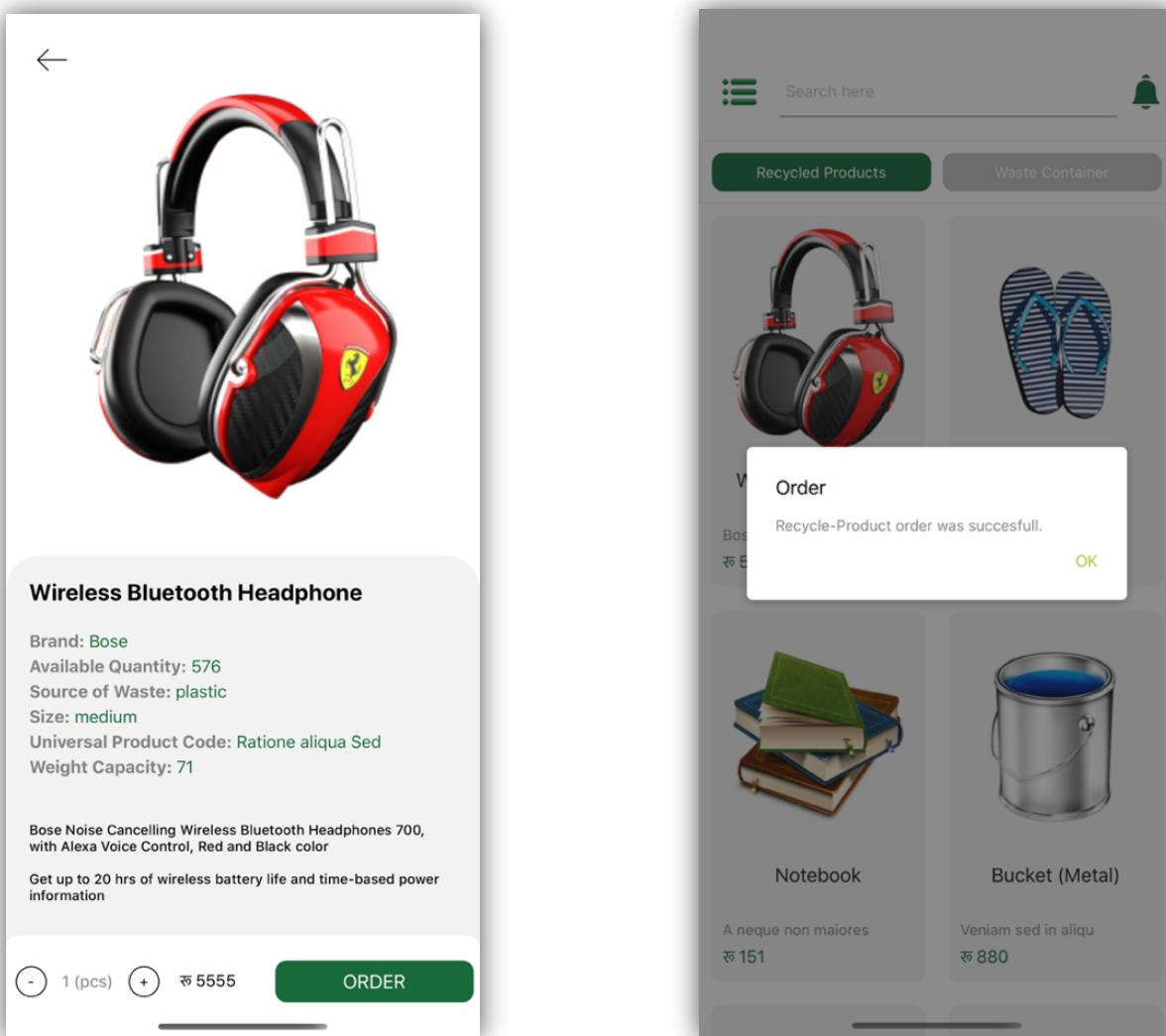


Figure 84 - Test case TC35 when a user ordered a recycled product

ORDER											
View orders made by customers. And, assign delivery-persons to orders.											
Order Status: Pending Orders											
Show 10 entries											
Search:											
Product	Order Code	Product Price	Quantity	Order By	Customer Location	Customer House no.	Order Time	Status	Assigned To	Action	
Wireless Bluetooth Headphone	vFtes1	Nrs.5555.00	1	Vladimir Putin	New Baneshwor	1234567890	2020-06-05 11:54:37	Pending		Assign	

Showing 1 to 1 of 1 entries

Previous 1 Next

Figure 85 - Test case TC35 where an ordered recycled product is displayed in the admin panel

Below show the screenshots for test case **TC36** (customer can order waste container from mobile application):

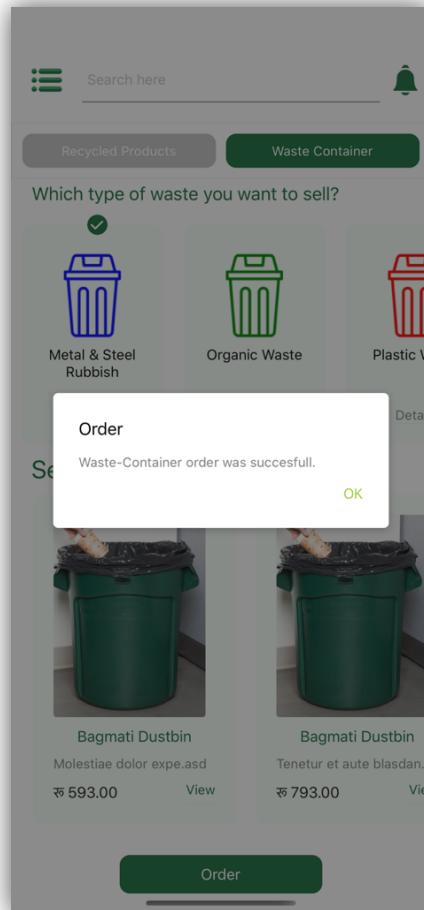


Figure 86 - Test case TC36 when a user ordered a waste container

ORDER										
View orders made by customers. And, assign delivery-persons to orders.										
Order Status: Pending Orders										
Show 10 entries										
Search:										
Product	Order Code	Product Price	Quantity	Order By	Customer Location	Customer House no.	Order Time	Status	Assigned To	Action
Dustbin [small]	egmKxP	Nrs.593.00	1	Vladimir Putin	New Baneshwor	1234567890	2020-06-05 12:01:23	pending		Assign
Wireless Bluetooth Headphone	vFtes1	Nrs.5555.00	1	Vladimir Putin	New Baneshwor	1234567890	2020-06-05 11:54:37	pending		Assign

Showing 1 to 2 of 2 entries

Previous 1 Next

Figure 87 - Test case TC36 where an ordered waste container is displayed in the admin panel

4.3.4 Waste collection schedule (customer)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC37	Customer who bought waste container can view the waste collection schedule of their street	1. Superhero set the waste collection schedule from admin panel to the streets 2. Customer click the collection schedule on the navigation drawer	Waste collection schedule should be displayed on the customer's mobile screen	Pass

Table 14 - Test case for Waste collection schedule (customer)

Below show the screenshots for test case **TC37** (customer who bought waste container can view the waste collection schedule of their street):

SET SCHEDULE

SCHEDULER
Set time for garbage pickup.

Province	District
Province 3	Kathmandu

Show 25 entries

Search: new b

Street Name	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
New Baneshwor	07:30:00 AM <input type="button" value=""/>	--:--:-- <input type="button" value=""/>	--:--:-- <input type="button" value=""/>	07:30:00 AM <input type="button" value=""/>	--:--:-- <input type="button" value=""/>	--:--:-- <input type="button" value=""/>	07:30:00 AM <input type="button" value=""/>

Showing 1 to 1 of 1 entries (filtered from 20 total entries)

Previous 1 Next

Figure 88 - Test case TC37 where a superhero set waste collection schedule to streets

Base URL for waste collection schedule: baseURL/api/v1/resource/time-schedule (Method: **GET**)

Here, the superhero set waste collection schedule for New Baneshwor where a customer also belongs to that street. Therefore, waste collection schedule was displayed on the customer's mobile screen.

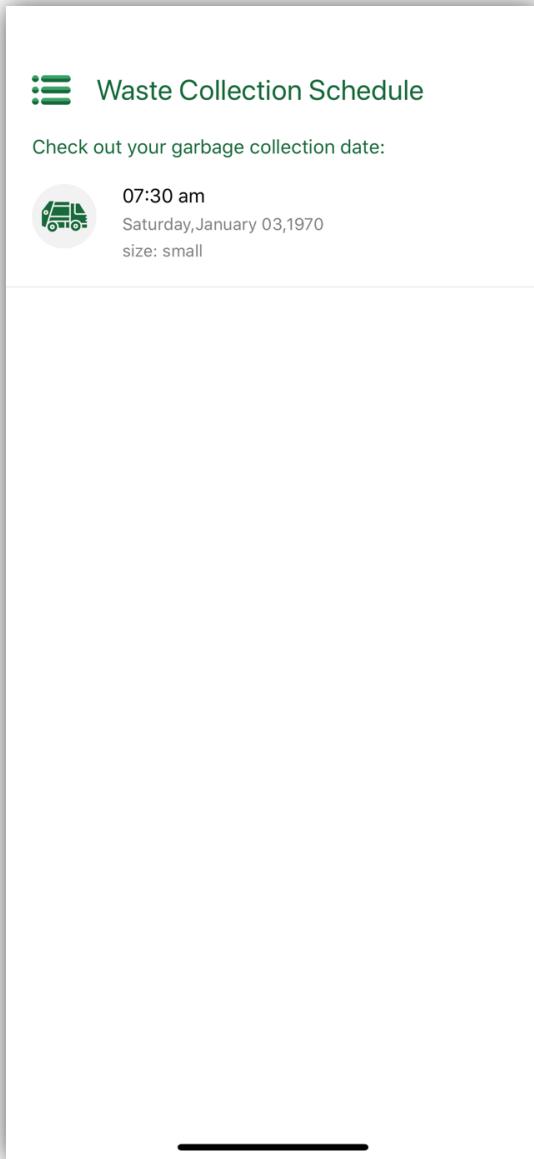


Figure 89 - Test case TC37 where a waste collection for New Baneshwor street was displayed

4.3.5 Shopping history (*customer*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC38	Customer can view the history of their purchased items	1. Click my profile on the navigation drawer 2. Click shopping history	History of the purchased order should be displayed	Pass

Table 15 - Test case for Shopping history (*customer*)

Base URL for shopping history API: baseURL/api/v1/customer/shopping/history/ (Method: **GET**)

Below show the screenshots for test case **TC38** (customer can view the history of their purchased items):

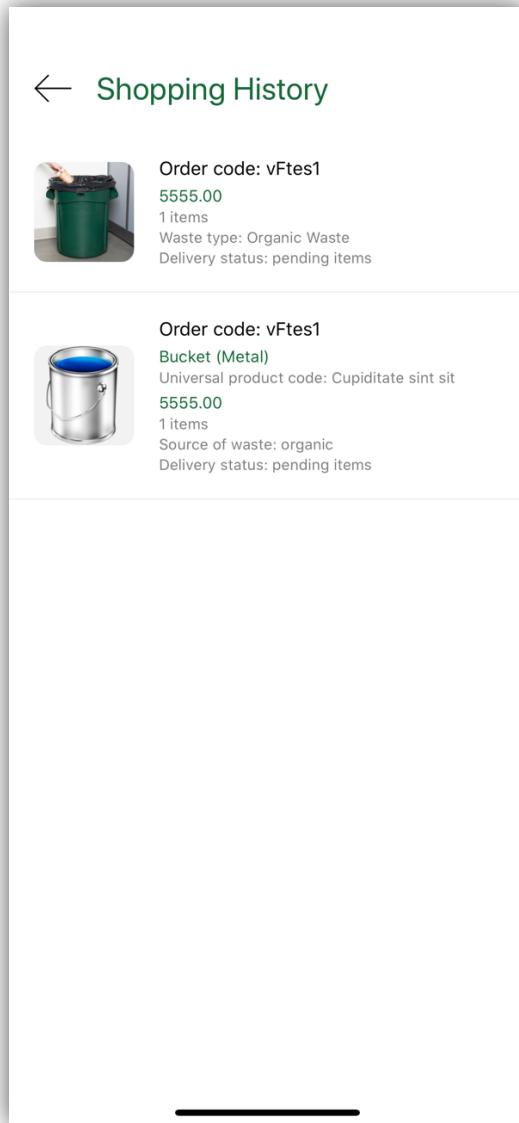


Figure 90 - Test case T38 where a customer can view the shopping history

4.3.6 View customer's waste collection schedule (*superhero*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC39	Superhero can view the customer's waste collection schedule according to street	1. Click settings – collection schedule 2. Select province, district and street	Customer's waste collection details for that street should be displayed	Pass

Table 16 - Test case for View customer's waste collection schedule (*superhero*)

Base URL for customer's waste collection schedule APIs is as below:

- 1. Waste containers details:** baseURL/api/v1/waste-collection/customer/waste-containers (Method: **GET**)
- 2. Update waste container details:** baseURL/ api/v1/waste-collection/customer/waste-containers/update (Method: **POST**)

Below show the screenshots for test case **TC39** (superhero can view the customer's waste collection schedule according to street):

Customer Name	Schedule
Jack Ma	Dustbin: Organic Waste [small] CollectionTime: 07:30 am Collection Date: Saturday,January 03,1970 Last Collected At: Thursday,January 01,1970
Jeff Bezos	Dustbin: Organic Waste [small] CollectionTime: 07:30 am Collection Date: Saturday,January 03,1970 Last Collected At: Thursday,January 01,1970
Narendra Modi	
Pope Francis	
Vladimir Putin	Dustbin: Organic Waste [small] CollectionTime: 07:30 am Collection Date: Saturday,January 03,1970 Last Collected At: Thursday,January 01,1970
Warren Buffet	

Figure 91 - Test case TC39 where superhero can view the customer's waste collection schedule according to the street

4.3.7 Assign employees – waste-collector (*superhero*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC40	Superhero can assign waste collector to the street for waste collection	1. Click assign employees – waste collectors 2. Search location or street 3. Select waste collector 4. Click assign button	Street should be assigned to the waste collector and all the customers to that street should be shown in the waste collectors mobile app	Pass

Table 17 - Test case for Assign employees – waste-collector (*superhero*)

Below show the screenshots for test case **TC40** (superhero can assign waste collector to the street for waste collection):

Here, superhero can assign street to the waste collector where all the customers who belong to that street should be shown in the waste collector's mobile app. And, superhero can also assign multiple waste collector to a street.

The screenshot shows a user interface titled "WASTE-COLLECTOR" with a sub-section titled "ASSIGN WASTE-COLLECTORS". Below this, a message says "Assign waste-collectors to the streets." A "Search Location" input field contains the text "new". To its right is a blue "Look For Collector" button. Below the input field is a dropdown menu with three options: "New Baneshwor" (which is highlighted in blue), "New Road", and another partially visible option.

Figure 92 - Test case TC40 where a superhero search for street to assign waste collectors

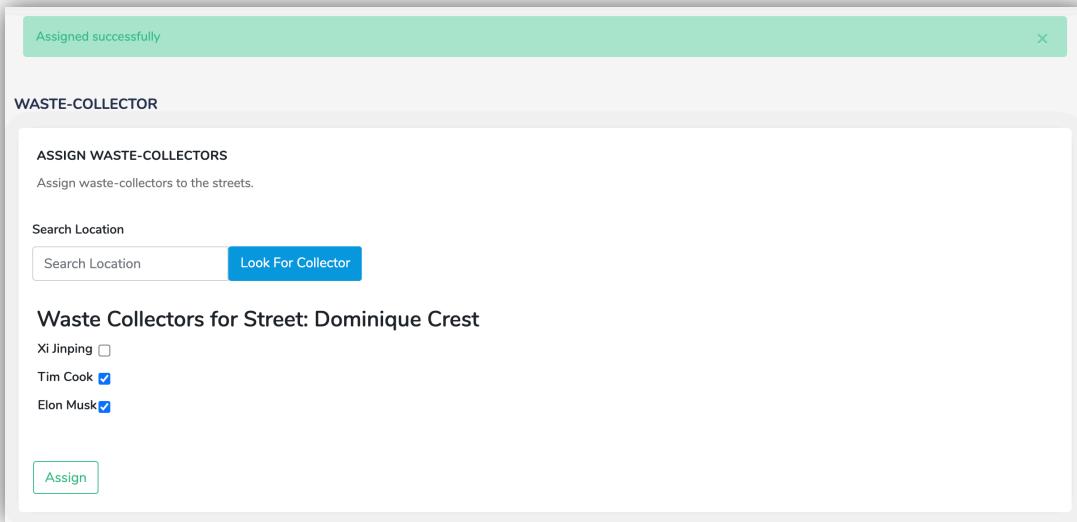


Figure 93 - Test case TC40 where a superhero assigned street (New Baneshwor) to the waste collector (Tim Cook & Elon Musk)

Base URL for waste collector APIs is as below:

- 1. Get location:** baseURL/api/v1/waste-collection/assigned-location (Method: **GET**)
- 2. Customer list:** baseURL/api/v1/waste-collection/customer/list (Method: **GET**)

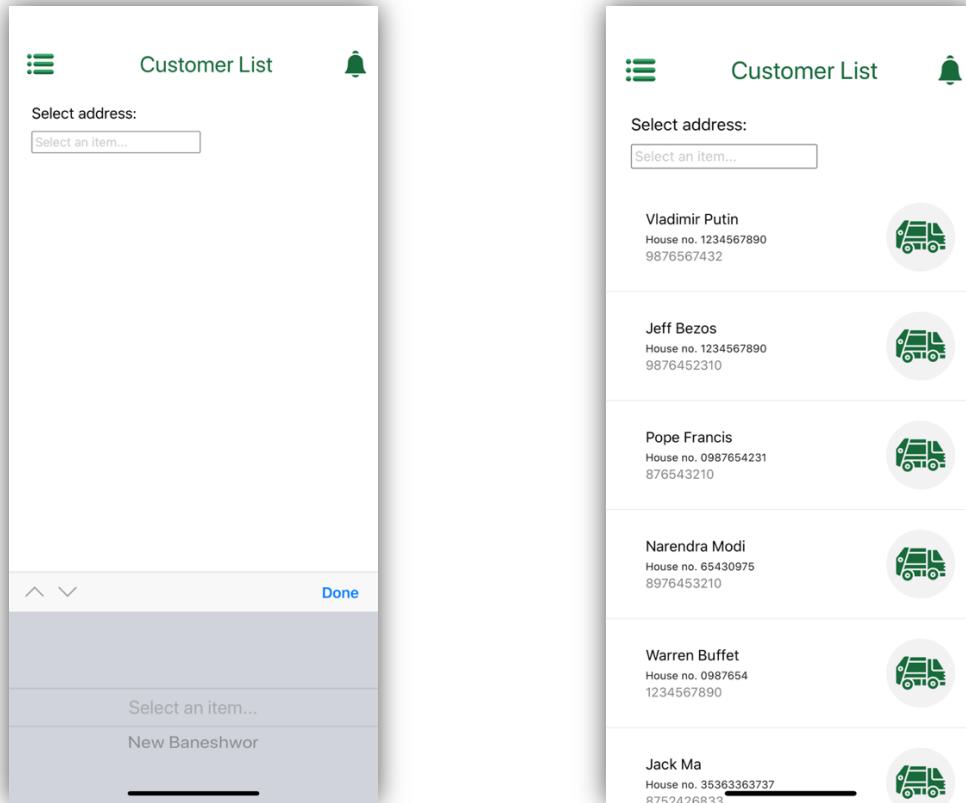


Figure 94 - Test case TC40 where a customer's list of New Baneshwor street is displayed in the waste collector mobile app

4.3.8 Assign employees – delivery-person (*superhero*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC41	Superhero can assign delivery person to the orders	1. Click assign employees – delivery persons 2. Click assign button 3. Select delivery person 4. Click assign button	Orders should be assigned to the delivery person and all the orders should be shown in the delivery person mobile app	Pass

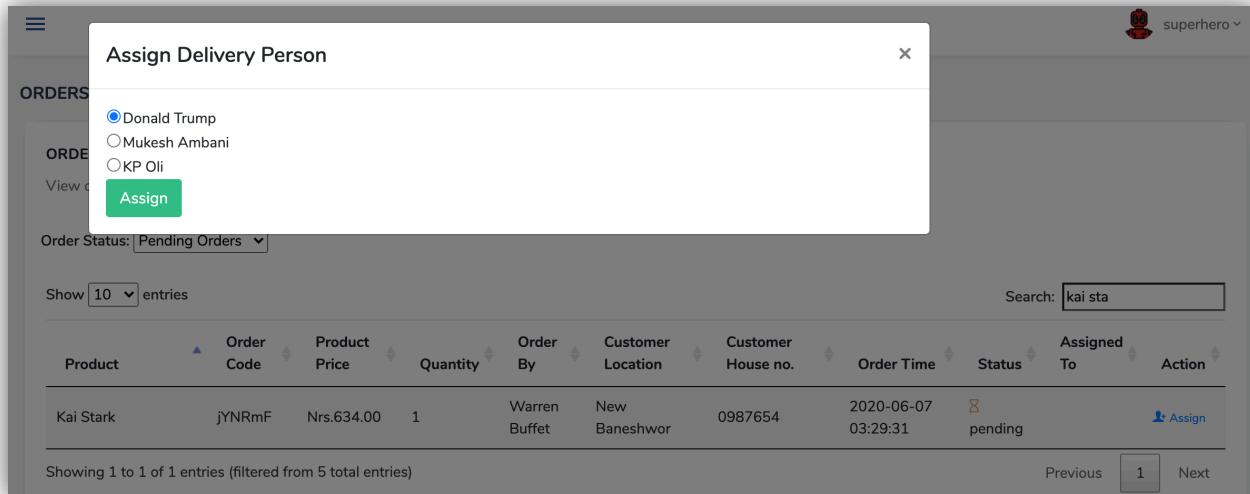
Table 18 - Test case for Assign employees – waste-collector (*superhero*)

Base URL for order APIs is as below:

1. **Order list:** baseURL/ api/v1/order/list (Method: **GET**)
2. **Order update status:** baseURL/api/v1/order/update-status (Method: **POST**)
3. **Order store:** baseURL/api/v1/order/store (Method: **POST**)

Below show the screenshots for test case **TC41** (superhero can assign delivery person to the orders):

Here, superhero can assign multiple orders to the delivery person which were made by customer. And, delivery person can view all the orders which were assigned to them in their mobile application.



The screenshot shows a mobile application interface for managing orders. At the top, there's a navigation bar with icons for menu, search, and account ('superhero'). Below it, a modal dialog titled 'Assign Delivery Person' is open. Inside the dialog, there's a list of delivery persons with radio buttons: 'Donald Trump' (selected), 'Mukesh Ambani', and 'KP Oli'. Below the list is a green 'Assign' button. In the background, a main screen displays a table of orders. The table has columns: Product, Order Code, Product Price, Quantity, Order By, Customer Location, Customer House no., Order Time, Status, Assigned To, and Action. One row is visible: 'Kai Stark' with Order Code 'jYNRmF', Product Price 'Nrs.634.00', Quantity '1', Order By 'Warren Buffet', Customer Location 'New Baneshwor', Customer House no. '0987654', Order Time '2020-06-07 03:29:31', Status 'pending', Assigned To 'Donald Trump', and Action 'Assign'. At the bottom of the main screen, it says 'Showing 1 to 1 of 1 entries (filtered from 5 total entries)' and has 'Previous' and 'Next' buttons.

Figure 95 - Test case TC41 where a superhero assigns orders to the delivery person (Donald Trump)

Assigned Delivery Person sucessfully X

ORDERS

ORDER

View orders made by customers. And, assign delivery-persons to orders.

Order Status: Pending Orders ▼

Show 10 entries ▼ Search:

Product	Order Code	Product Price	Quantity	Order By	Customer Location	Customer House no.	Order Time	Status	Assigned To	Action
Kai Stark	jYNRmF	Nrs.634.00	1	Warren Buffet	New Baneshwor	0987654	2020-06-07 03:29:31	pending	Donald Trump	Assign

Showing 1 to 1 of 1 entries (filtered from 5 total entries) Previous 1 Next

Figure 96 - Test case TC41 where superhero assigned orders to the delivery person (Donald Trump)

Order List

Order code: jYNRmF
Warren Buffet , New Baneshwor

Order Detail

Order code: jYNRmF
1 items ₹ 634.00
Order status: pending

Delivery Address:
House no: 0987654
Street: New Baneshwor Warren Buffet

Call

Item:

1 items
Kai Stark
Brand: Id porro pariatur C
Color: red
Size: medium
Source of Waste: plastic
Universal product code: Autem laboriosam ne
Available stock: 818 ₹ 634

Delivered

Figure 97 - Test case TC41 where a list of order assigned to a delivery person (Donald Trump) is shown

4.3.9 View orders (*superhero*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC42	Superhero can view orders and check by order status - pending or delivered	1. Click orders 2. Select order status	Orders should be displayed in the admin panel and superhero should view by order status	Pass

Table 19 - Test case for View orders (*superhero*)

Below show the screenshots for test case **TC42** (superhero can view orders and check by order status - pending or delivered):

ORDER										
View orders made by customers. And, assign delivery-persons to orders.										
Order Status: Pending Orders										
Show 10 entries	Search:									
Product	Order Code	Product Price	Quantity	Order By	Customer Location	Customer House no.	Order Time	Status	Assigned To	Action
Bucket (Metal)	kWD6sR	Nrs.880.00	1	Narendra Modi	New Baneshwor	65430975	2020-06-05 22:35:10	X pending	KP Oli	Assign
Bucket (Metal)	MgrfaB	Nrs.880.00	1	Jack Ma	New Baneshwor	35363363737	2020-06-05 12:59:47	X pending	Mukesh Ambani	Assign
Dustbin [small]	RtMPNT	Nrs.593.00	1	Jack Ma	New Baneshwor	35363363737	2020-06-05 12:34:05	X pending	Mukesh Ambani	Assign
Dustbin [small]	egmKxP	Nrs.593.00	1	Vladimir Putin	New Baneshwor	1234567890	2020-06-05 12:01:23	X pending	KP Oli	Assign
Notebook	gdPHxM	Nrs.151.00	1	Narendra Modi	New Baneshwor	65430975	2020-06-05 22:35:48	X pending	KP Oli	Assign
Wireless Bluetooth Headphone	vFtes1	Nrs.5555.00	1	Vladimir Putin	New Baneshwor	1234567890	2020-06-05 11:54:37	X pending	Mukesh Ambani	Assign

Figure 98 - Test case TC42 where the pending order list is displayed

4.3.10 Collect waste (*waste collector*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC43	Waste collector can view the list of customers of their assigned area	1. Login to the mobile app 2. Click on customer's list 3. Update the waste collected weight 4. Click update button	Customer's list should be displayed in the waste collector's mobile app and waste collected weight should be updated	Pass
TC44	Verify waste collected to waste container weight capacity	1. Input more weight collected than waste container weight capacity 2. Click update button	Weight capacity shouldn't be updated, and error message should be shown	Pass

Table 20 - Test case for Collect waste (*waste collector*)

Below show the screenshots for test case **TC43** (waste collector can view the list of customers of their assigned area):

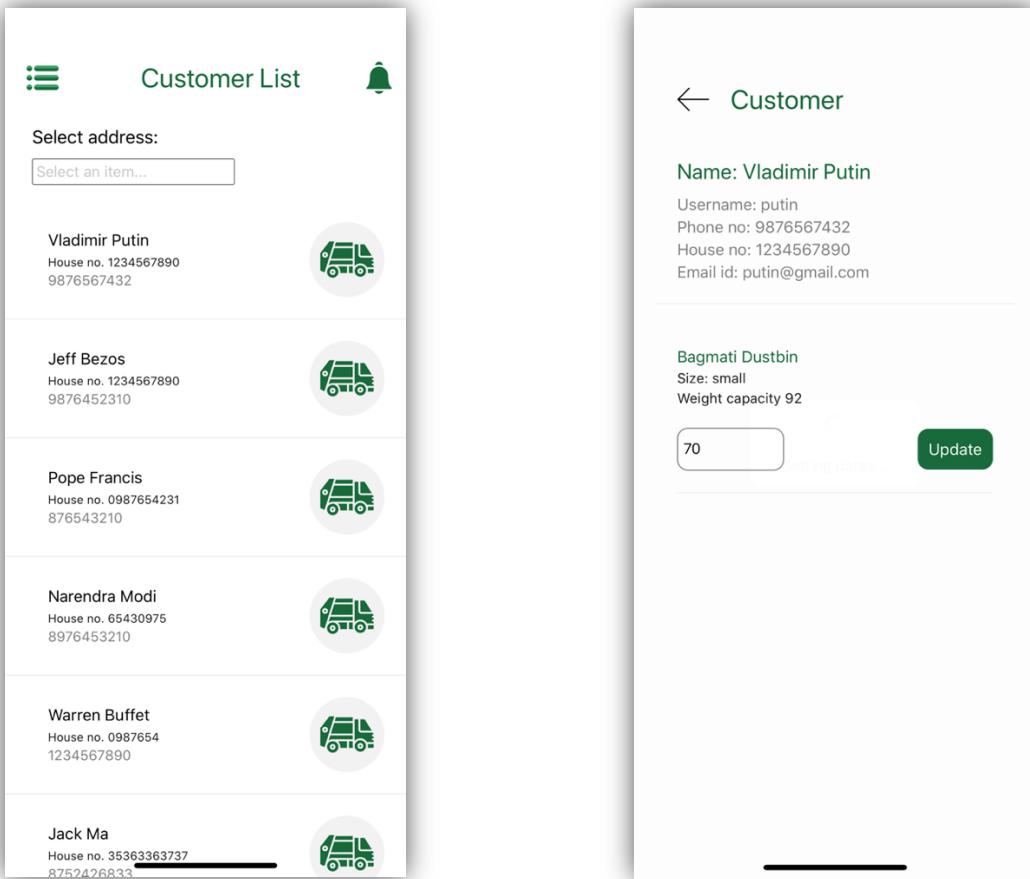


Figure 99 - Test case TC43 where customer's list is displayed, and garbage weight collected is updated

Below show the screenshots for test case **TC44** (verify waste collected to waste container weight capacity):

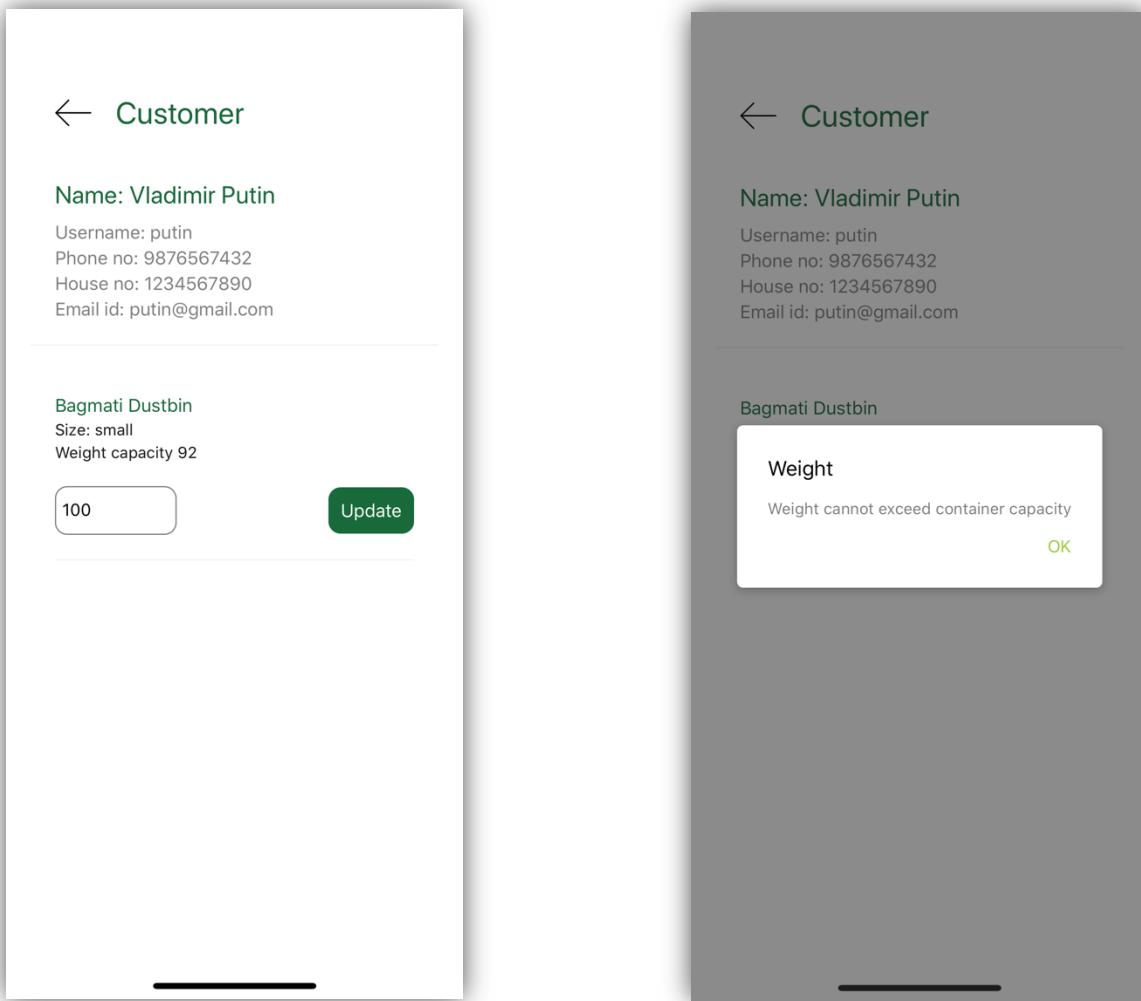


Figure 100 - Test case TC44 when a waste collector input more weight than waste container capacity

4.3.11 Deliver products (*delivery person*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC45	Delivery person can all the orders which were assigned by superhero	1. Login to the mobile app 2. Click on orders list 3. Click delivered button	Order should be delivered, and order's list should be displayed in the admin panel	Pass

Table 21 – Test case for Deliver products (*delivery person*)

Below show the screenshots for test case **TC45** (verify waste collected to waste container weight capacity):

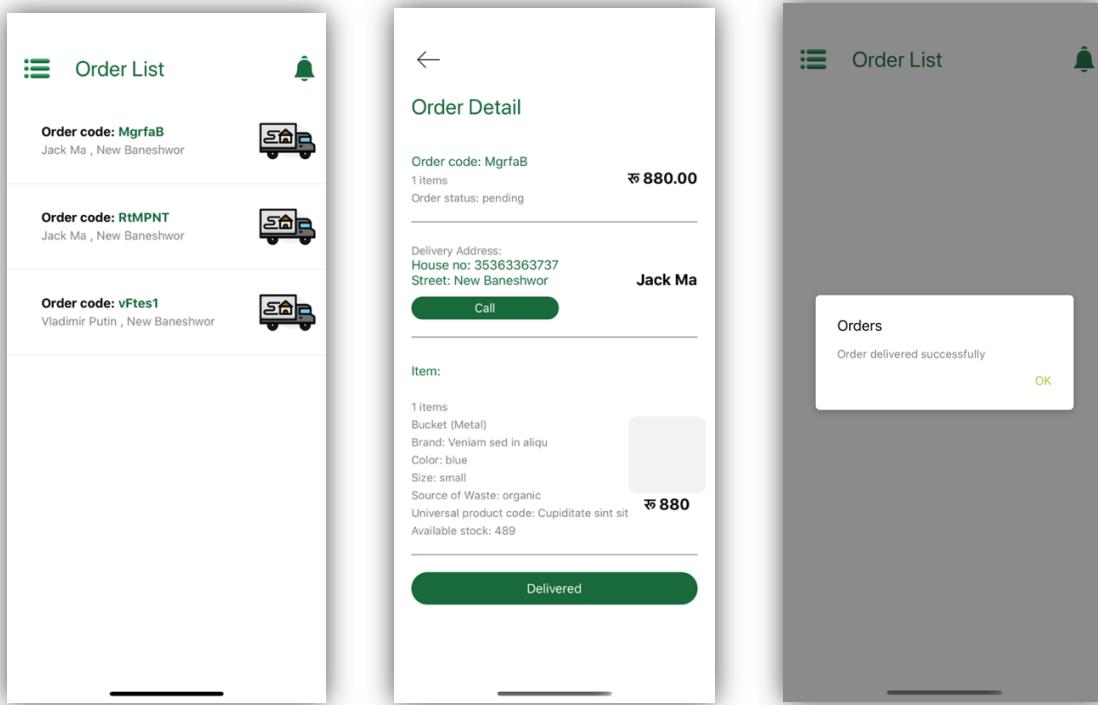


Figure 101 - Test case TC45 where a delivery person successfully delivered a product (Bucket) of Jack Ma

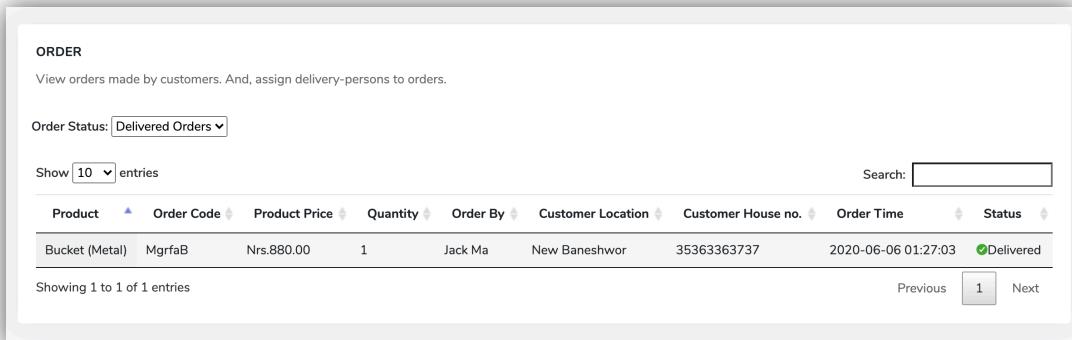


Figure 102 - Test case TC45 where customer's order details is displayed in the admin panel

4.3.12 Profile – waste containers details (*customer*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC46	Verify if the waste container details and waste collected is showing	1. Click the profile on navigation drawer 2. Click My Achievements button	Waste container's details and total waste collected weight should be shown on the customer's profile	Pass

Table 22 - Test case for Profile – waste containers details (*customer*)

Base URL for waste container details APIs is as below:

1. **Customer waste container:** baseURL/ api/v1/customer/waste-container (Method: **GET**)
2. **Total waste collected:** baseURL/api/v1/customer/waste-collected/total (Method: **GET**)

Below show screenshots for test case **TC46** (verify if the waste collected weight is updated):

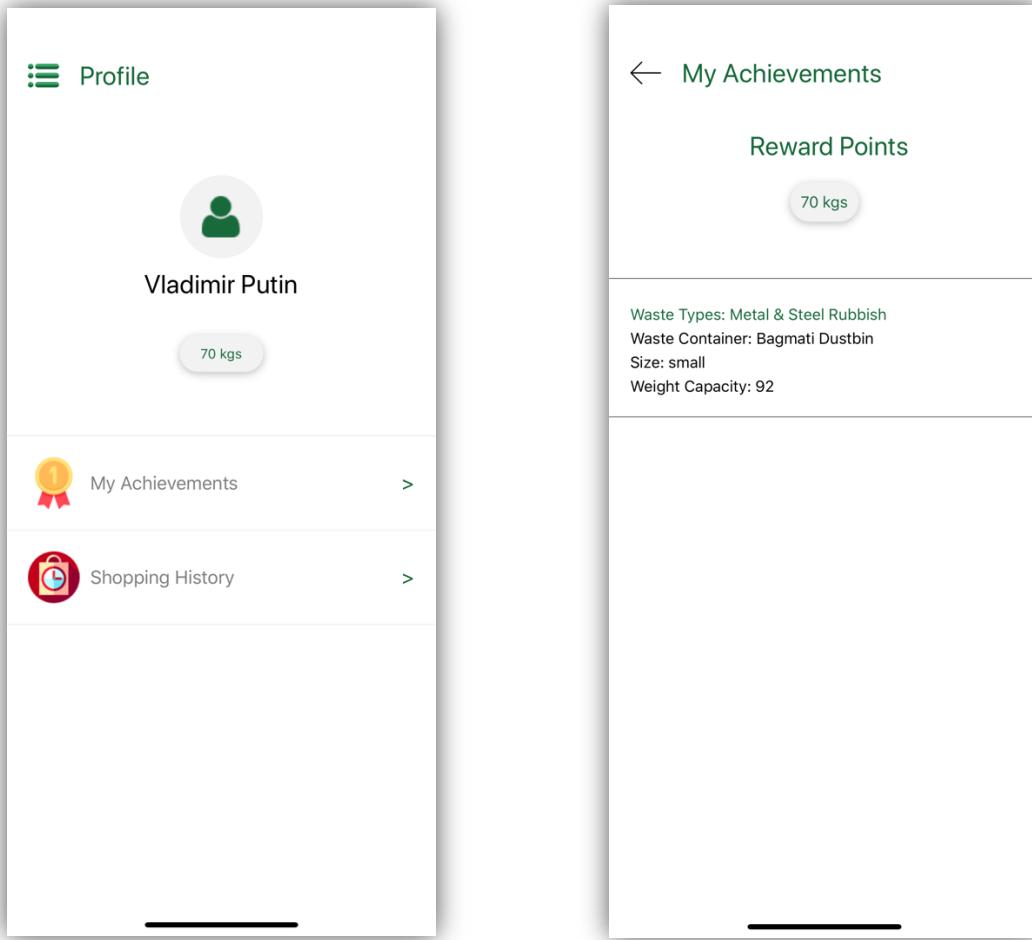


Figure 103 - Test case TC46 showing waste containers and total waste collected details

4.3.13 Push notification (*customer*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC47	Verify expo token of user's device	1. User login to the mobile app 2. Check database	New expo token for that user's device should be displayed in the database	Pass
TC48	Verify if the notification received before one hour of waste collection time	1. Monitor job is processed from queue	Customer should receive notification and displayed on app	Pass

Table 23 - Test case for Push notification (*customer*)

Below show the screenshots for test case **TC47** (verify expo token of user's device):

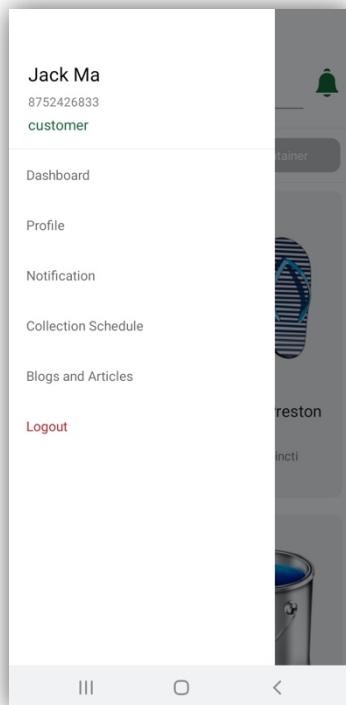


Figure 104 - Test case TC47 where customer (Jack Ma) log in to the mobile app

+ Options		id	user_id	expo_token	created_at	updated_at
<input type="checkbox"/>	Show all	25	Filter rows:	Search this table		
<input type="checkbox"/>	Edit	Copy	Delete	30	ExponentPushToken[4CuLQuHPDqEYufh39TMACT]	2020-06-07 14:15:36
<input type="checkbox"/> Check all With selected: <input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete <input type="checkbox"/> Export						
<input type="checkbox"/> Show all Number of rows: 25 Filter rows: Search this table						
Query results operations						

Figure 105 - Test case TC47 where new expo token for Jack Ma (customer) is generated

Below show the screenshots for test case **TC48** (verify if the notification received before one hour of waste collection time):

SCHEDULER
Set time for garbage pickup.

Province: Province 3 | District: Kathmandu

Show 25 entries | Search: new b

Street Name	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
New Baneshwor	09:12:00 PM	12:00:00 PM	--:--:--	07:30:00 AM	--:--:--	--:--:--	07:30:00 AM

Showing 1 to 1 of 1 entries (filtered from 20 total entries) | Previous 1 Next

Submit

Figure 106 - Test case TC48 where waste collection schedule for New Baneshwor street is set to 9:12 pm

WASTE COLLECT SCHEDULER
View customer's garbage pickup schedule.

Province: Province 3 | District: Kathmandu | Street: New Baneshwor

Show 10 entries | Search: jac

Customer Name	Schedule
Jack Ma	Dustbin: Organic Waste [small] CollectionTime: 21:12 pm Collection Date: Sunday,June 07,2020 Last Collected At: Friday,June 05,2020

Showing 1 to 1 of 1 entries (filtered from 6 total entries) | Previous 1 Next

Figure 107 - Test case TC48 where waste collection schedule for Jack Ma is set for 21:12 pm

Base URL for notification list API for all users: baseURL/api/v1/notification/list (Method: **GET**)

Waste collection schedule for New Baneshwor street was set for 9:12 pm which means customer should receive notification before one hour of waste collection time i.e. 8:12 pm.

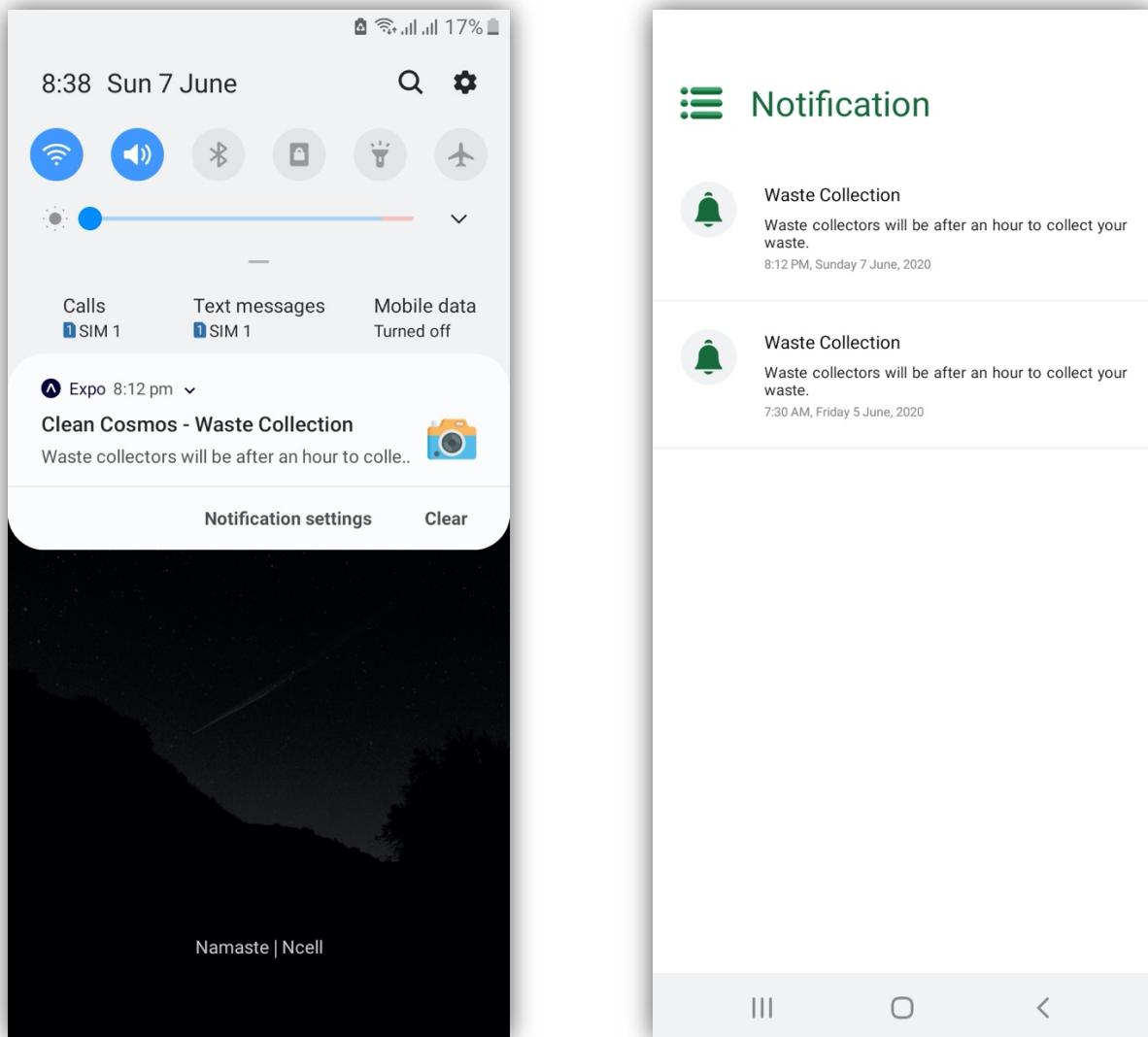


Figure 108 - Test case TC48 where a customer (Jack Ma) received a notification before one hour of waste collection time

4.3.14 Push notification (*waste collector*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC49	Verify if the notification received in the day of waste collection	1. Monitor job is processed from queue	Waste collector should receive notification and displayed on mobile app	Pass

Table 24 - Test case for Push notification (*waste collector*)

Below show the screenshots for test case **TC49** (verify if the notification received in the day of waste collection):

The screenshot shows the 'WASTE-COLLECTOR' application interface. Under the 'ASSIGN WASTE-COLLECTORS' section, there is a note: 'Assign waste-collectors to the streets.' Below this is a search bar labeled 'Search Location' with a placeholder 'Search Location' and a blue button labeled 'Look For Collector'. A list of waste collectors is shown with checkboxes next to their names: 'Xi Jinping' (unchecked), 'Tim Cook' (checked), and 'Elon Musk' (checked). At the bottom is a green 'Assign' button.

Figure 109 - Test case TC49 where waste collectors assigned for New Baneshwor are Tim Cook and Elon Musk

The screenshot shows the 'SCHEDULER' application interface. It has sections for 'Province' (Province 3) and 'District' (Kathmandu). Below these are dropdown menus for 'Show 25 entries' and a search bar with the text 'new b'. A table lists collection times for 'Street Name' 'New Baneshwor' on days of the week: Sunday (09:12:00 PM, 12:00:00 PM, 07:30:00 AM, 07:30:00 AM), Monday (09:12:00 PM, 12:00:00 PM, 07:30:00 AM, 07:30:00 AM), Tuesday (09:12:00 PM, 12:00:00 PM, 07:30:00 AM, 07:30:00 AM), Wednesday (09:12:00 PM, 12:00:00 PM, 07:30:00 AM, 07:30:00 AM), Thursday (09:12:00 PM, 12:00:00 PM, 07:30:00 AM, 07:30:00 AM), Friday (09:12:00 PM, 12:00:00 PM, 07:30:00 AM, 07:30:00 AM), and Saturday (09:12:00 PM, 12:00:00 PM, 07:30:00 AM, 07:30:00 AM). At the bottom, it says 'Showing 1 to 1 of 1 entries (filtered from 20 total entries)' and has 'Previous', 'Next', and a page number '1'.

Figure 110 - Test case TC49 where waste collection day for New Baneshwor street is Sunday

The waste collector assigned to New Baneshwor is Tim Cook. The waste collection day for New Baneshwor is Sunday where waste collector (Tim Cook) received a notification on Sunday.

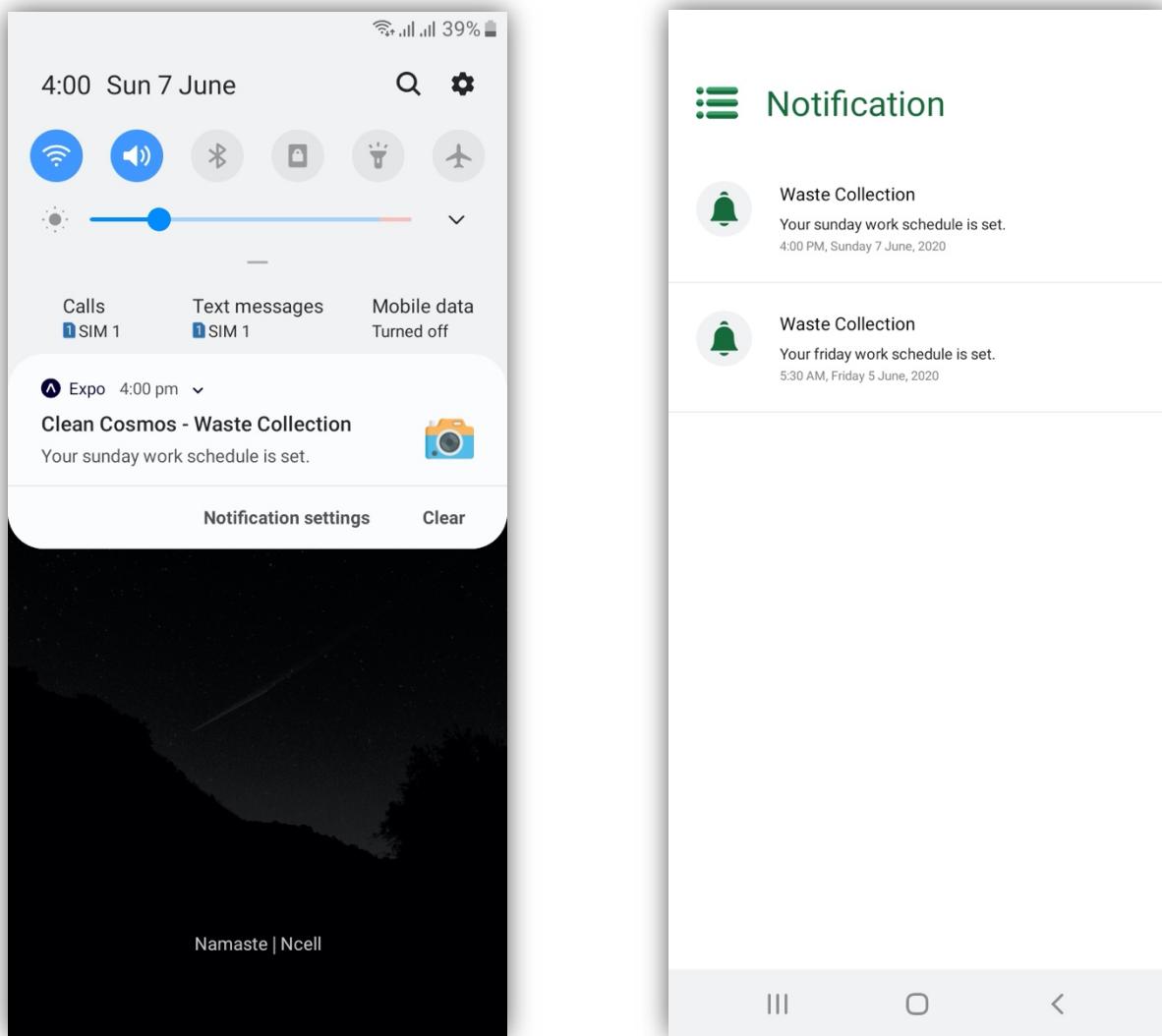


Figure 111 - Test case TC49 where a waste-collector (Tim Cook) received a notification on Sunday

4.3.15 Push notification (*delivery person*)

ID	Test Case Description	Test Case Procedure	Expected Output	Result
TC50	Verify if the notification received after orders assigned to a delivery person	1. Assign orders to a delivery person	Delivery person should receive notification and displayed on mobile app	Pass

Table 25 - Test case for Push notification (*delivery person*)

Below show the screenshots for test case **TC50** (verify if the notification received in the day of waste collection):

The screenshot shows a table of orders. One row is highlighted, representing a customer order for a Wireless Bluetooth Headphone. The order details are: Product: Wireless Bluetooth Headphone, Order Code: vFtes4, Price: Nrs.5555.00, Quantity: 1, Order By: Narendra Modi, Customer Location: New Baneshwor, Customer House no.: 65430975, Order Time: 2020-06-05 23:39:37, Status: pending, Assigned To: (empty), Action: [Assign](#). The table has columns for Product, Order Code, Product Price, Quantity, Order By, Customer Location, Customer House no., Order Time, Status, Assigned To, and Action.

Figure 112 - Test case TC50 where a customer made order of Wireless Bluetooth Headphone

The screenshot shows a success message "Assigned Delivery Person sucessfully" at the top. Below it is the same order table as Figure 112. The order for the Wireless Bluetooth Headphone now has an assigned delivery person: Donald Trump. The status is still pending. The rest of the table and interface are identical to Figure 112.

Figure 113 - Test case TC50 where a superhero assigned an order to a delivery person (Donald Trump)

When a superhero assigned orders to a delivery person, delivery person gets notification about that assigned orders.

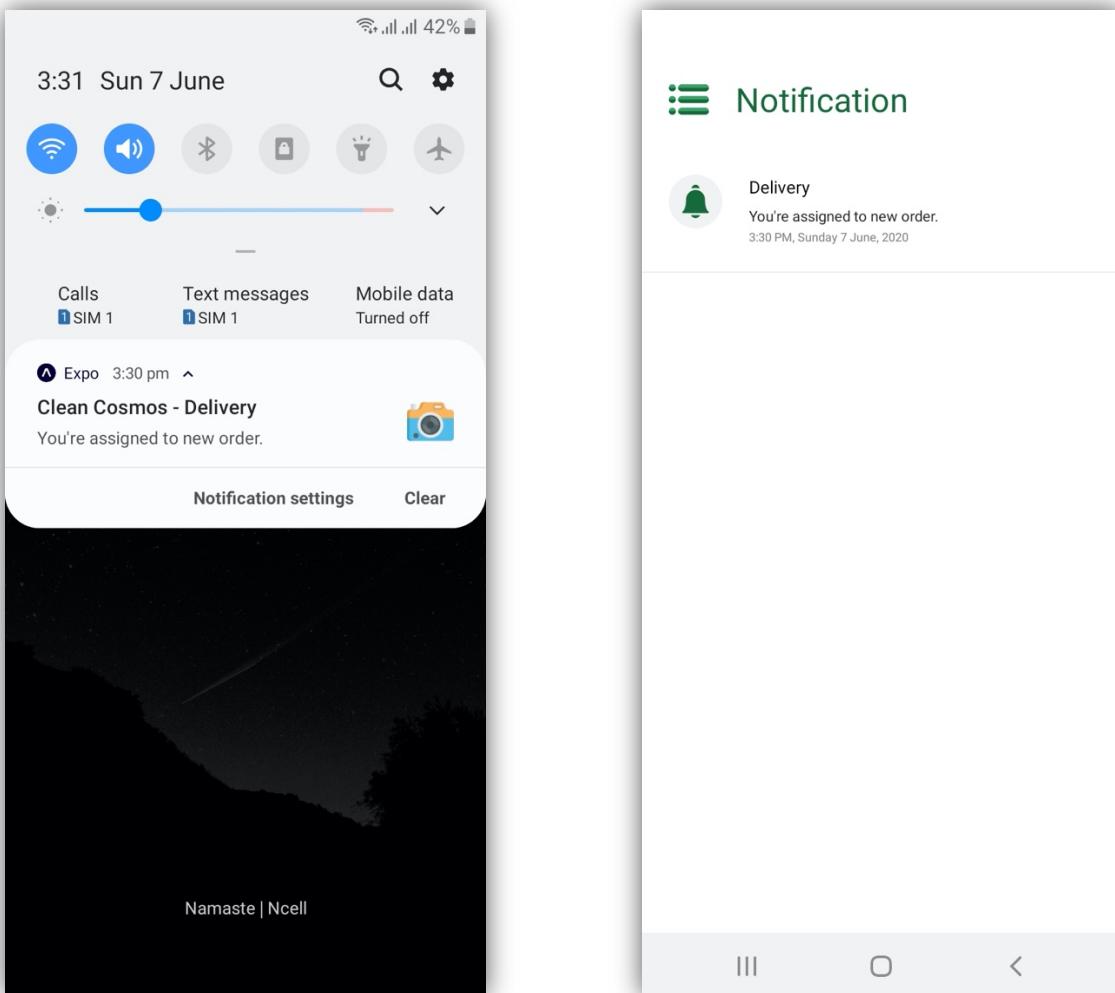


Figure 114 - Test case TC50 where a delivery person (Donald Trump) received a notification for a new order

4.4 Critical Analysis

The whole system was tested via different testing approaches and methodologies which are unit, integration, compatibility, and user acceptance testing. After doing testing of both web and mobile application I found that it is a crucial part of software development which ensures me to verify that the software been developed performs all functional requirements and is free from any form of defect and errors.

All the individual system components or modules were tested during the development phase via unit testing which helps me to develop the system faster and debugging the application gone easy. All the APIs or integrated modules were tested which helps me to identify whether the logic implemented in multiple integrated modules is as per the expectations or not. This approach also helps me to detect the errors related to the interface.

Compatibility testing was performed to check whether the application performs or operates as expected for all the intended users across multiple platforms. Integration testing was performed via friends and others to identify whether the end to end flow of the application is as per the needs of the end-user or not. Basically, the web and mobile application were tested by the users to ensure that the system is working in the way it has to. This testing also helps in determining how efficiently all the previous testing was performed.

The compatibility testing and user acceptance testing for both web and mobile applications are displayed in Appendix E: Testing and Analysis.

5 Conclusion

With all the accumulated effort invested in this project, there are reasons to believe that at the end of the project, this application finds itself in a much better shape and very closer to goal of the project. When gathering information from different sources, as covered by the first objective, I expected a lack of response where my supervisors helped me. There are many problems regarding this topic where I tried to find a solution via this project. Different types of UML diagrams were created and took surveys to obtain information from people regarding waste management and recycling.

All the tasks or backlog were successfully developed using Kanban methodology. After the software development, system was completely tested using four different testing approaches. Due to an unforeseen pandemic situation, I'd have a little bit of a hard time doing my project. However, learning online and doing assignments online also helped in gaining knowledge in an efficient way.

I viewed the project as a journey where I learned many lessons and gained some insights to the subjects which I tried to share in this report and summarized in this chapter. I tried to look at the problems from many points of view which generated some new ideas that could be explored in the future.

5.1 Legal, Social and Ethical Issues

5.1.1 Legal Issues

Legal issue or issue of law is a legal question or laws that are set according to the government laws. It requires a court's decision. It can also refer to a point on which the evidence is undisputed, the outcome of which depends on the court's interpretation of the law. According to the constitution of Nepal, when users use the mobile application fills in with all their personal data shouldn't be share for personal use. Any time the system collects the personal identifiable information (PII) from the users, it risks of violating privacy laws.

5.1.2 Social Issues

Every technological invention has got both positive and negative impacts on the society. Social issues can be defined as problems or issues that affect the significant number of people in a society. It is often the result of factors that extend beyond the control of an individual and is the source of a conflicting opinion on the basis of what is perceived as a morally fair personal life or societal order. If the mobile application posts fake news and information regarding waste management system and recycling, users will get misinformation.

5.1.3 Ethical Issues

Ethical issues in general terms can be stated as the issue generated or raised in the context of people doing their works. It is a problem or situation that requires a person or organization to choose between alternatives that must be evaluated as ethical or unethical. There are some ethical issues that are closely connected to this project, such as trust, knowledge, privacy, and individual autonomy.

5.2 Limitations

Even the project was completed successfully, there are some limitation I faced during this project. This project is an individual type project where I can't be able to learn in a team. It is difficult to find the latest solid waste management data of Nepal in the internet.

Due to an unforeseen pandemic situation, I'd have a little bit of a hard time doing my project. It was difficult to focus on project during this time. However, learning online and doing assignments online also helped in gaining knowledge in an efficient way.

5.3 Future Work

I tried to look at the problems from many points of view which generated some new ideas that could be explored in the future. During the development phase, different new ideas which will be implementation in future. Future work concerns deeper analysis of particular mechanisms, new proposals to try different methods, or simply curiosity. There are some ideas I would like to try after the completion of the project.

I would like to add a business account feature for merchants or companies. This feature will help companies like office buildings, schools, stores, hotels, restaurants and other commercial and institutional buildings who generate significant amounts of materials and waste to improve waste management via this application.

I would also like to add donate waste feature where organizations can donate products or materials to others who need and can use the items. For example, restaurants, hotels and cafeterias promptly distribute perishable and prepared foods to hungry people in their communities. Many local food banks will pick up food donations free of charge, saving you storage and disposal costs. I would also like to add some extra charts to track waste in an admin console.

6 References

Asian Development Bank. (2013). *Solid Waste Management in Nepal: Current Status and Policy Recommendations*. Retrieved May 11, 2020, from

<https://www.adb.org/sites/default/files/publication/30366/solid-waste-management-nepal.pdf>

Azoulay, J. “. (2020, January 02). *waste-management-kathmandu-littered-journey*. Retrieved from spotlightnepal: <https://www.spotlightnepal.com/2018/07/02/waste-management-kathmandu-littered-journey/>

Computer Hope. (2020, January 02). *What is a Web Application?* Retrieved from computerhope: <https://www.computerhope.com/>

eBay. (2020, January 6). *Create your listing*. Retrieved from ebay: <https://bulksell.ebay.com/ws/eBayISAPI.dll?SingleList&&DraftURL=http://my.ebay.com/ws/eBayISAPI.dll?MyEBayBeta¤tpage=MyeBayNextSavedDrafts>

eBay Inc. (2020, January 6). *Our Company - eBay Inc.* Retrieved from ebayinc: <https://www.ebayinc.com/company/>

Ghahrai, A. (2020). *Software Testing Fundamentals*. Retrieved May 20, 2020, from <https://devqa.io/software-testing-fundamentals/>

Google Play. (2020, January 6). *Recycle Coach - Apps on Google Play*. Retrieved from google play: <https://play.google.com/store/apps/details?id=mobi.recyclecoach.worldster.pack>

7 Bibliography

Leblanc, R. (2020, January 03). *an-introduction-to-metal-recycling*. Retrieved from thebalancesmb: <https://www.thebalancesmb.com/an-introduction-to-metal-recycling-4057469>

Mwamba, S. (2020, January 02). *plastic-pollution-facts*. Retrieved from globalcitizen: <https://www.globalcitizen.org/en/content/plastic-pollution-facts/>

Opsomer, R., & Pennington, J. (2020, January 02). *what-are-the-drawbacks-of-todays-plastics-economy*. Retrieved from weforum: <https://www.weforum.org/agenda/2016/03/what-are-the-drawbacks-of-todays-plastics-economy/>

Packt Publishing Ltd. (2020, January 04). *9-reasons-to-choose-agile-methodology-for-mobile-app-development*. Retrieved from packtpub: <https://hub.packtpub.com/9-reasons-to-choose-agile-methodology-for-mobile-app-development/>

Recycle Coach. (2020, January 6). *Residents | Recycling at home made easier*. Retrieved from recyclecoach: <https://recyclecoach.com/>

Ritchie, H., & Roser, M. (2018). *Plastic Pollution*. Retrieved May 12, 2020, from <https://ourworldindata.org/plastic-pollution>

WASTE WISE PRODUCTS INC. (2020, January 02). *food-waste-an-economic-and-environmental-problem/*. Retrieved from wastewiseproductsinc: <https://www.wastewiseproductsinc.com/blog/food-waste/food-waste-an-economic-and-environmental-problem/>

World Bank Group. (2019). *Solid Waste Management*. Retrieved May 11, 2020, from <https://www.worldbank.org/en/topic/urbandevelopment/brief/solid-waste-management>

X-Cart . (2020, January 7). *X-Cart online store builder - X-Cart Demo store company*. Retrieved from x-cart: <https://demostore.x-cart.com/admin/admin.php>

8 Appendix

8.1 Appendix A: Pre-Survey

This section includes the questions asked and the results of the surveys which was taken before with the help of google forms.

8.1.1 Pre-Survey Form

Name *

Your answer

Email *

Your answer

Where do you live in the Kathmandu Valley? *

Your answer

Total members in the house * 1 point

Your answer

Figure 115 - Queries related to general details of users

Household Waste Generation and Disposal

In your opinion which of these is a priority concern about waste in the area? *

Littering and looks bad
 Effects on human health
 Effect on environment
 Other: _____

Which size of storage method (plastic bags, dustbin) do you use? And, how often does it gets emptied in a week? *

Your answer _____

Do you separate different types of waste at your home? *

Choose ▾

Would you do so if you are told by your collection service provider? *

Choose ▾

Figure 116 - Queries related to household waste generation and disposal

Garbage Collection Services

Do you have regular garbage collection in your area? *

Choose ▾

If yes, do you use it?

Choose ▾

How often do you use the collection service? *

Once a week
 Twice a week
 Other: _____

Which collection service do you use? *

Public
 Private

Figure 117 – Queries related to garbage collection services

Recycling

What recycled materials do you typically use? *

Paper
 Plastic
 Metal and Steel
 Food Waste
 Other: _____

Any strategies for effective waste management?

Your answer _____

Back **Submit**

Figure 118 - Queries related to recycling

8.1.2 Pre-Survey Result

8.1.2.1 General Details

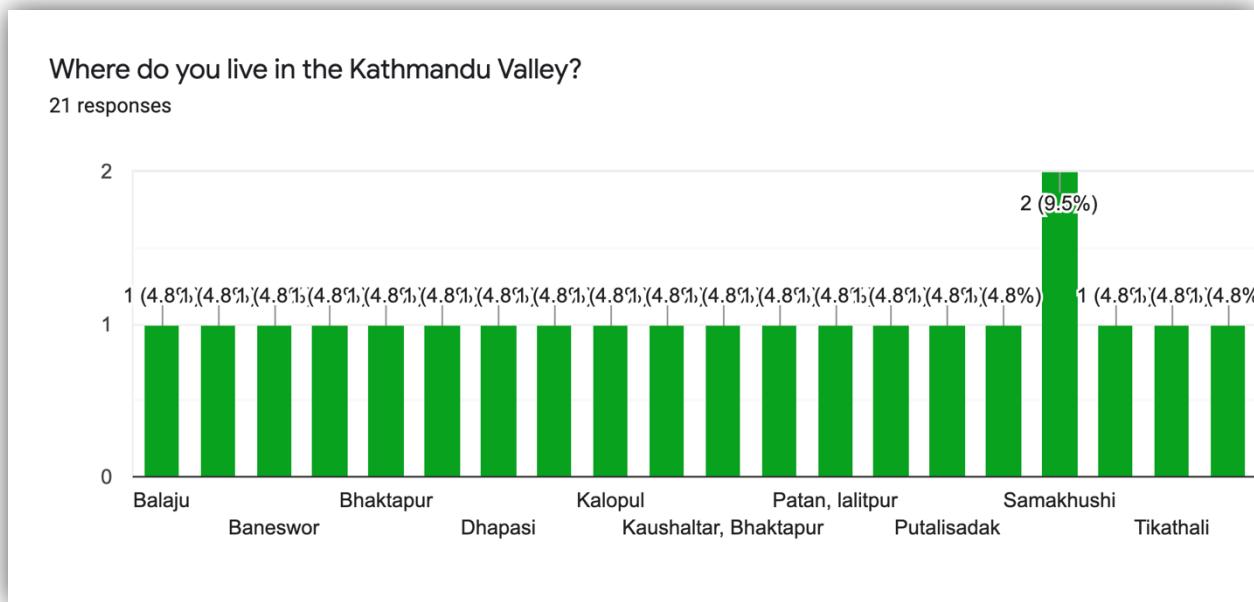


Figure 119 – Respondent's locations in Kathmandu Valley

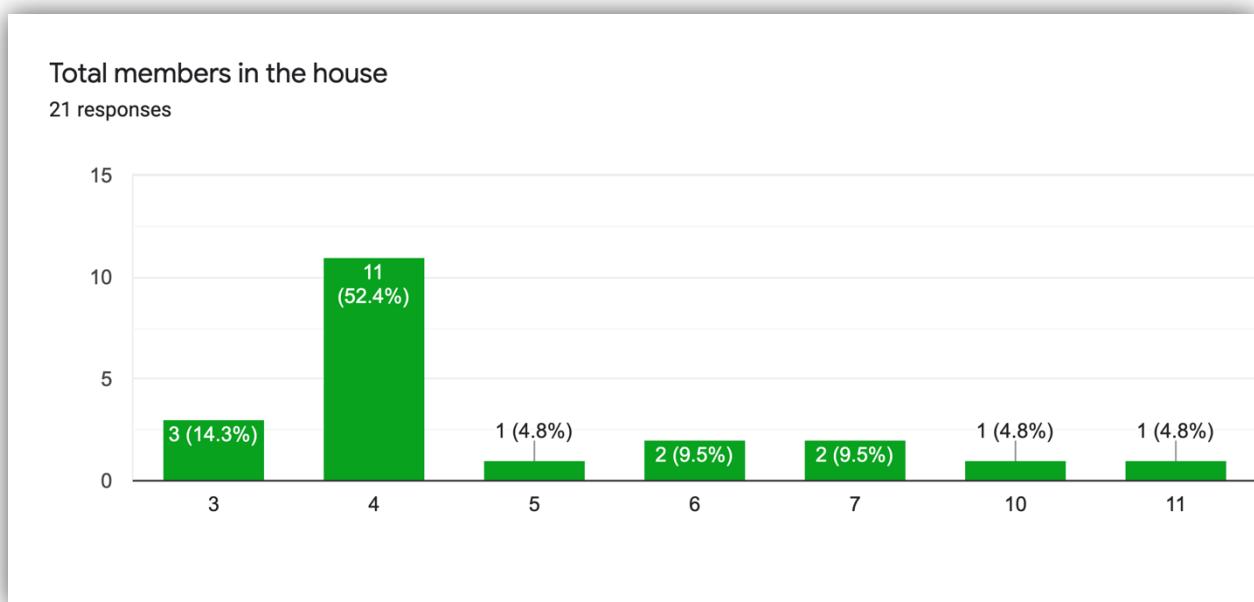


Figure 120 - Total members living in the house

8.1.2.2 Household Waste Generation and Disposal

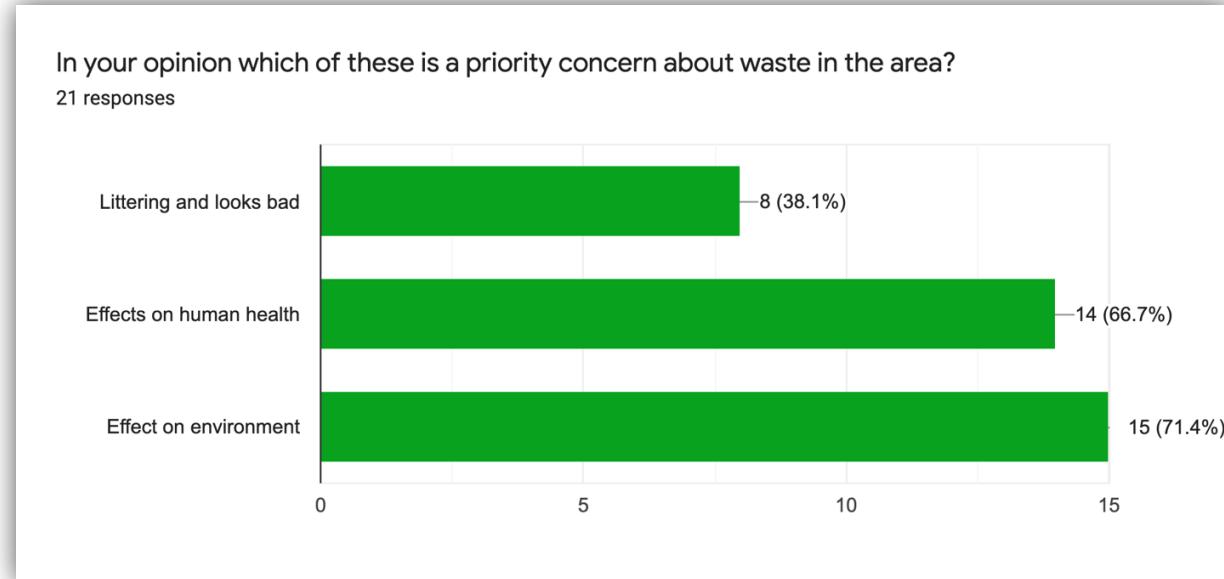


Figure 121 - Consequences of waste in the respondent area

Which size of storage method (plastic bags, dustbin) do you use? And, how often does it gets emptied in a week?

21 responses

plastic bags, once a week

I use size of about 50litre dustbin and it gets emptied in about few weeks.

dustbin

Dustbin, personally in my room, around once a week

I used 8 * 12 size of dustbin at my home and it get emptied twice a week

Plastic bags. By the disposal in municipality truck

I use medium size dustbin which gets emptied twice a week.

Dustbin, twice a week

Dustbin, once a week

Figure 122 - Storage method using and how often does it gets emptied in a week

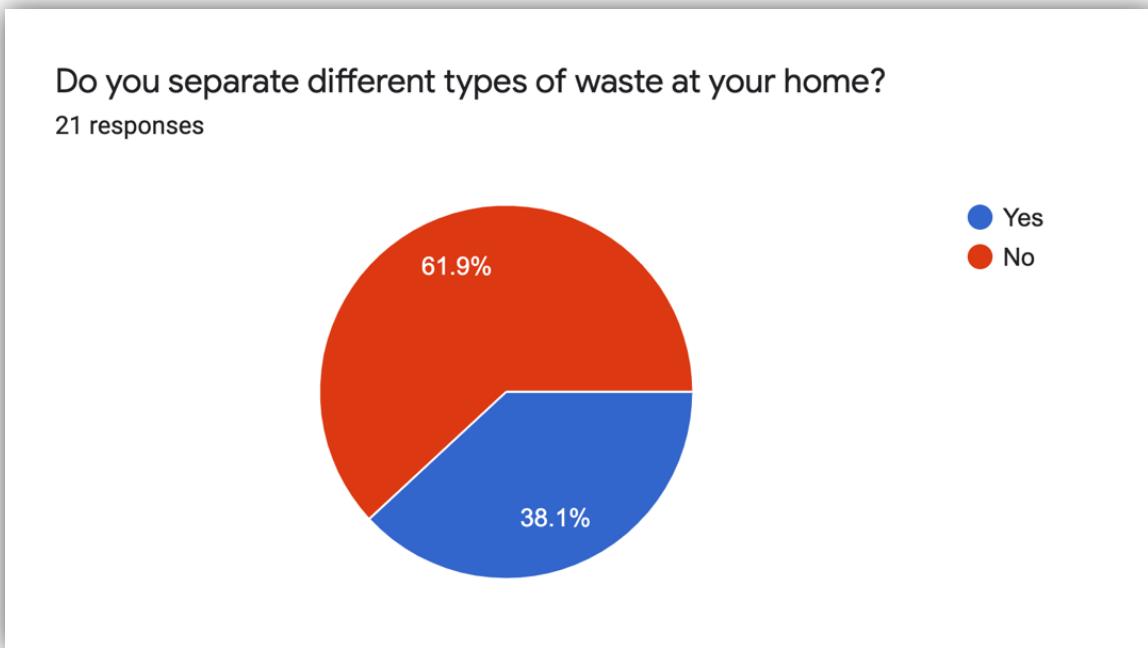


Figure 123 - Separation of waste at home

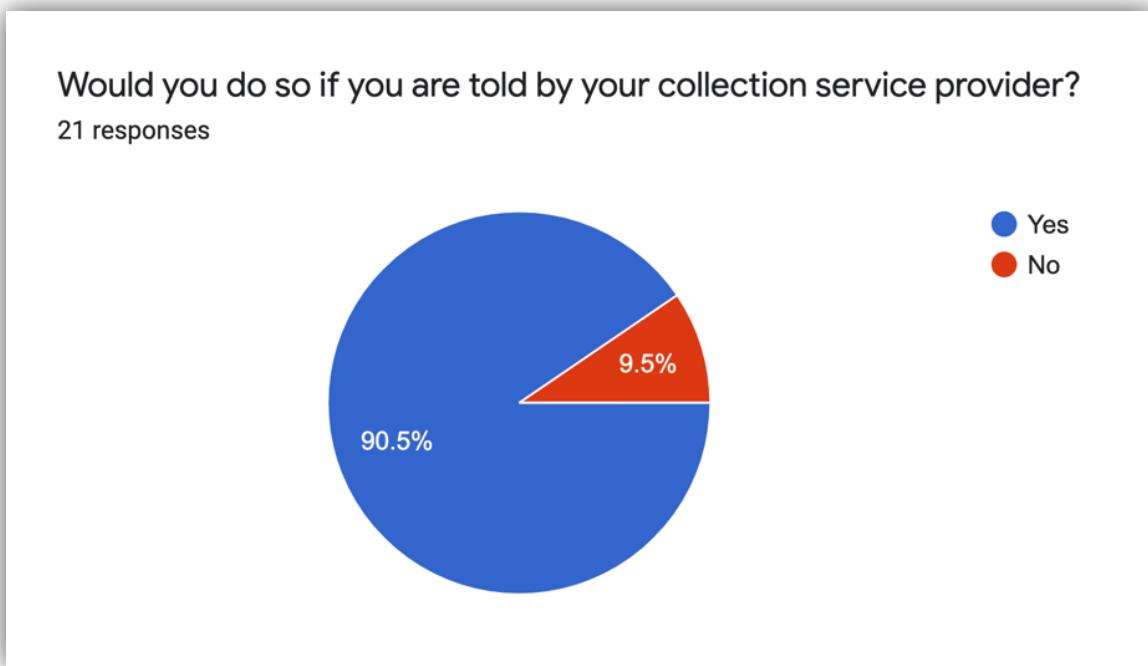


Figure 124 - Separation of waste if waste collection service provider tells to separate

8.1.2.3 Garbage Collection Services

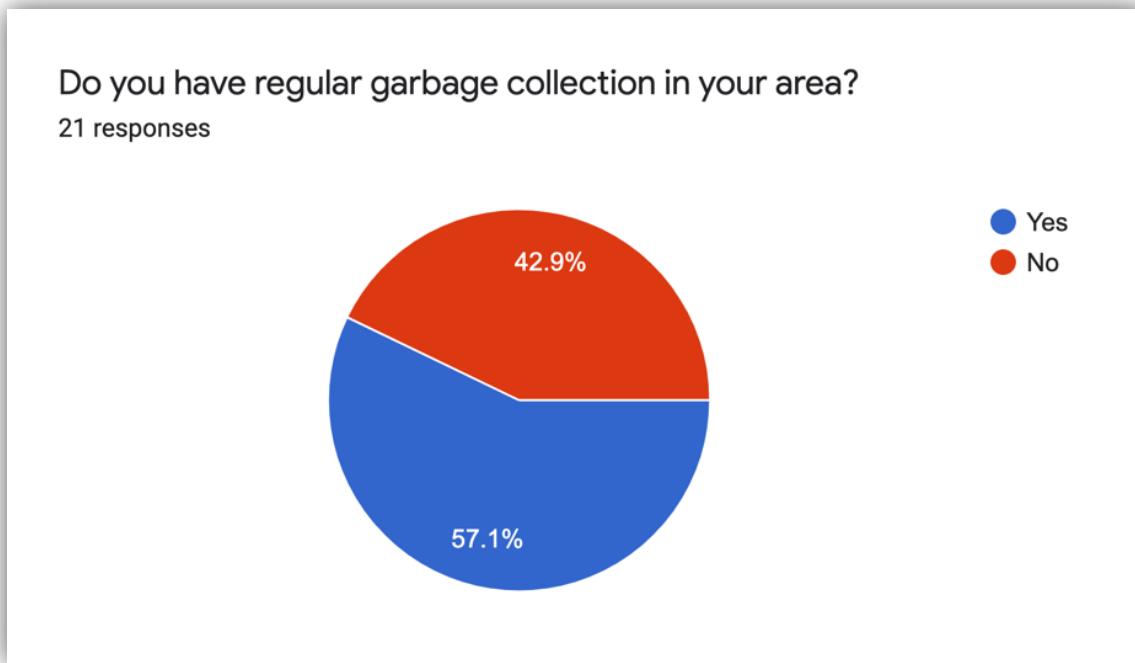


Figure 125 - Regular garbage collection services in the respondent's area

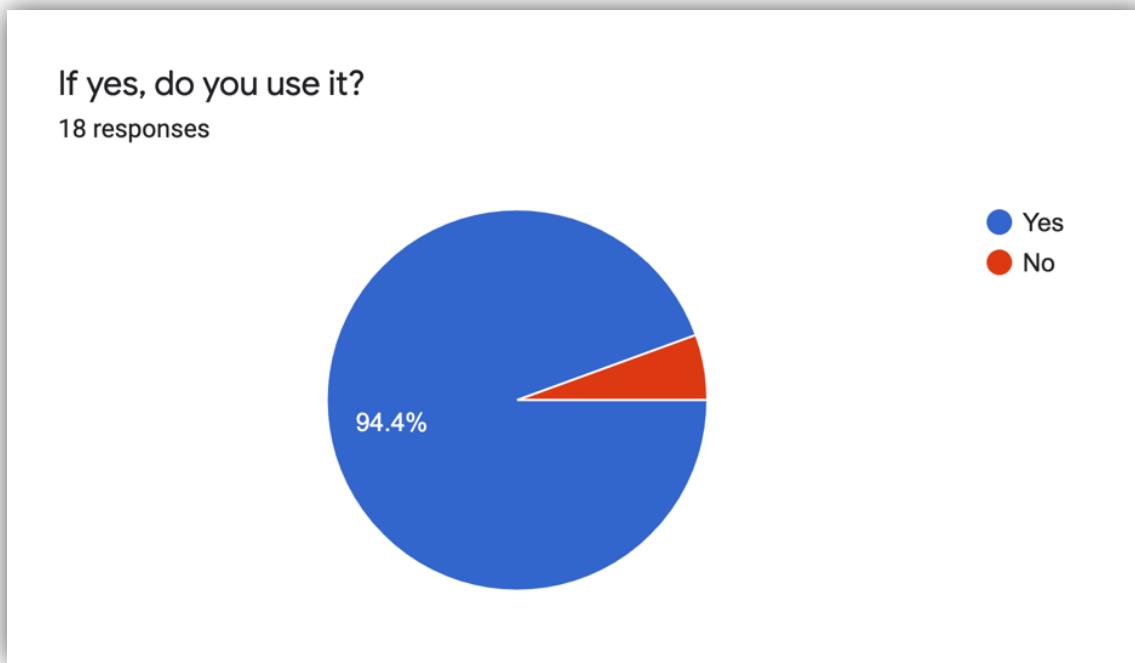


Figure 126 - Garbage collection services in the respondent's area if they collect regular

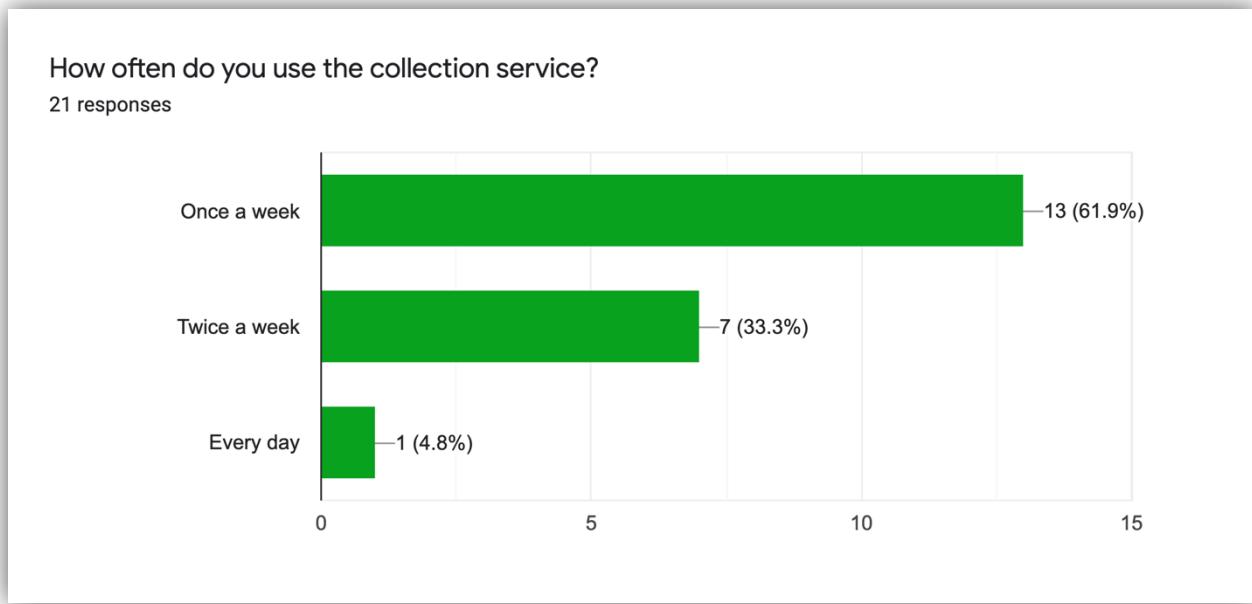


Figure 127 - Waste collection services using respondents

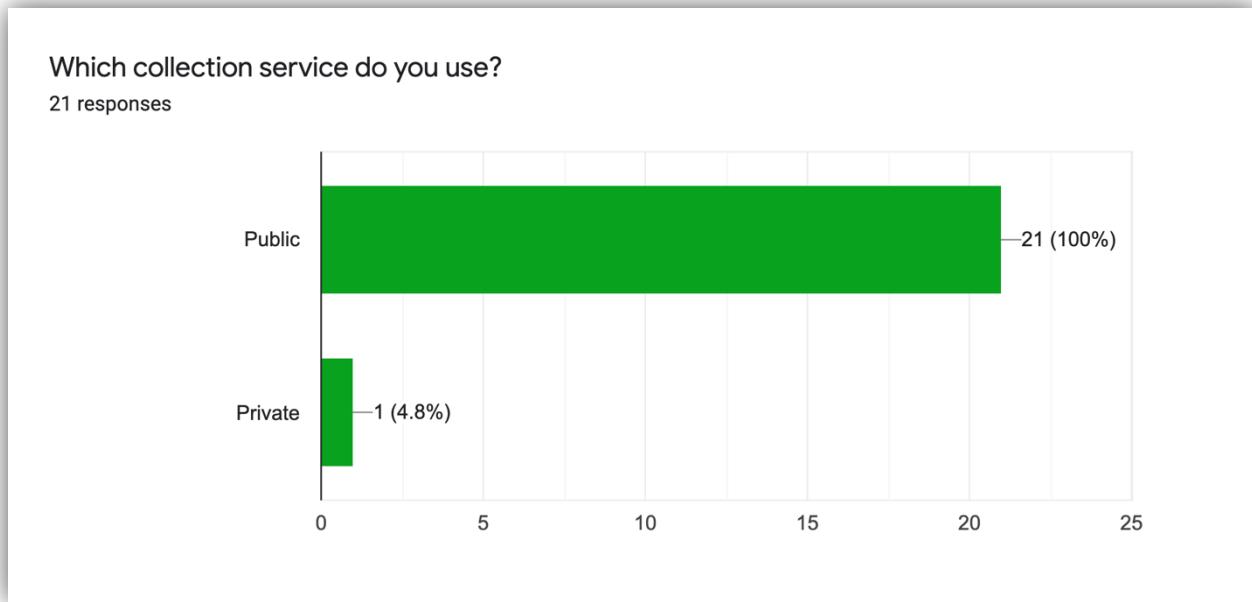


Figure 128 - Type of waste collection services using respondents

8.1.2.4 Recycling

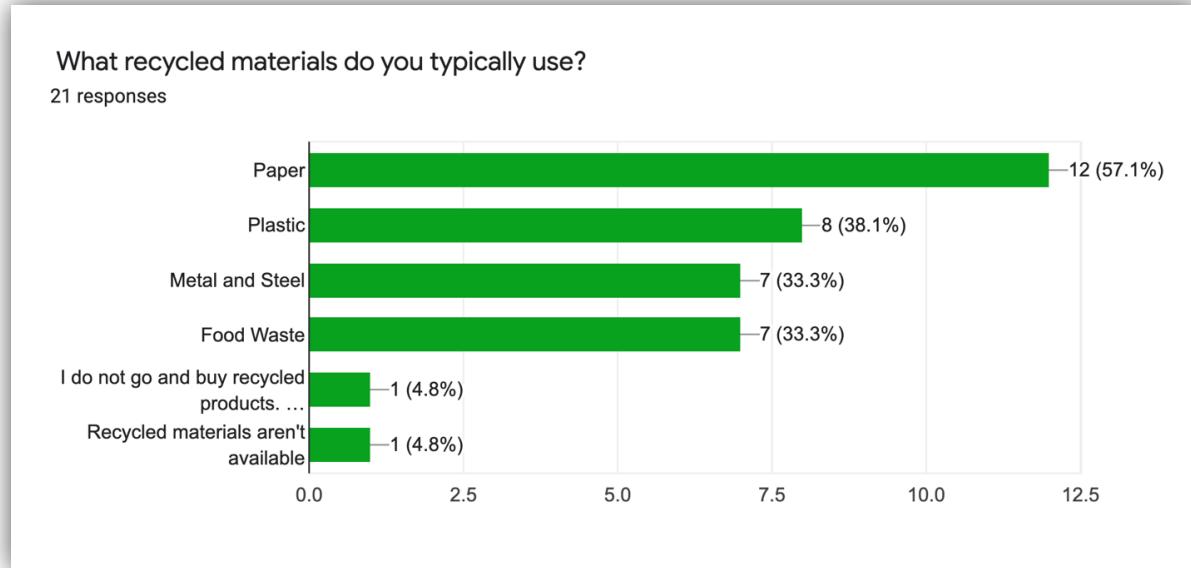


Figure 129 - Type of recycled materials respondent normally use

Any strategies for effective waste management?

11 responses

Garbage collection bins should be available in every local area.

proper management

Collection of waste should be in daily basis with proper dumping area

Using the decomposing wastes as manure

Recycling all the waste & proper management...

Better cleaning

Separation of waste products and recycling those than can be recycled

Right now, no idea

I think we should avoid creating waste in the first place.

Figure 130 - Respondents opinion for effective waste management

8.2 Appendix B: Post-Survey

This section includes the questions asked and the results of the surveys which was taken after the development with the help of google forms.

8.2.1 Post-Survey Form

The screenshot shows a portion of a Google Form. It features two stacked text input fields. The top field is labeled "Name *". Below it is a placeholder "Your answer" followed by a horizontal line for input. The bottom field is labeled "Email *". Below it is a placeholder "Your answer" followed by a horizontal line for input.

Figure 131 - Queries related to general details

The screenshot shows a Google Form with four distinct sections. The first section, titled "Web Application" and "Admin Console", contains a question "How do you like the app design? *". It includes a scale from 1 to 5 with radio buttons, where 1 is "Very Unlikely" and 5 is "Very Likely". The second section, titled "Do you think navigation between pages is simple and uncluttered? *", has a dropdown menu with options "Choose" and " ". The third section, titled "How do you like the catalog management feature? *", also includes a scale from 1 to 5 with radio buttons, ranging from "Very Unlikely" to "Very Likely". The fourth section, titled "How do you like the order management feature? *", follows the same pattern. All sections include a red asterisk indicating they are required fields.

Figure 132 –Queries related to web application

How do you like the user management feature? *

1 2 3 4 5

Very Unlikely Very Likely

What should I add extra data to the dashboard? *

Your answer

How do you like the settings of the application? *

1 2 3 4 5

Very Unlikely Very Likely

Do you have any suggestions to improve the admin panel?

Your answer

Figure 133 - Queries related to web application II

Mobile Application

For Customers, Waste Collectors and Delivery Person

How do you like the app design? *

1	2	3	4	5	
Very Unlikely	<input type="radio"/> Very Likely				

How do you like the functionalities of the mobile app? *

1	2	3	4	5	
Very Unlikely	<input type="radio"/> Very Likely				

How do you like ordering products from a mobile app? *

1	2	3	4	5	
Very Unlikely	<input type="radio"/> Very Likely				

How do you like the garbage collection features? *

Figure 134 - Queries related to mobile application

How do you like the employees (Waste Collectors and Delivery Person) app? *

1 2 3 4 5

Very Unlikely Very Likely

How would you rate the overall experience of the app? *

1 2 3 4 5

Very Unlikely Very Likely

Would you recommend this app to others? *

Choose ▾

Do you have any suggestions to improve the mobile app?

Your answer

Figure 135 - Queries related to mobile application II

8.2.2 Post-Survey Result

8.2.2.1 Web Application

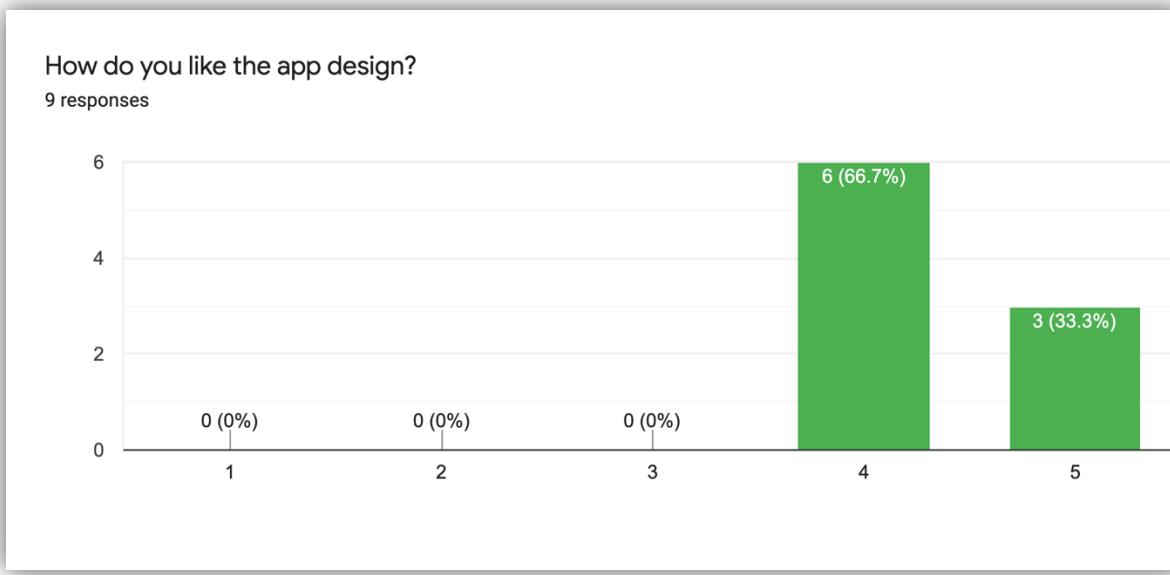


Figure 136 - Web application design

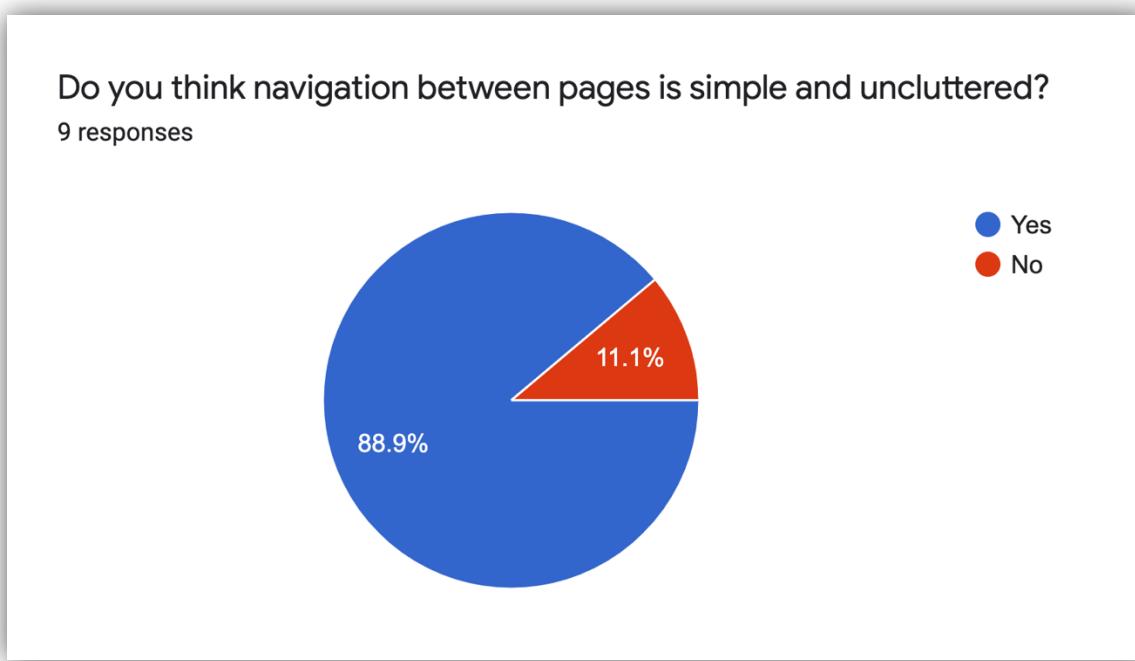


Figure 137 - Navigation between pages

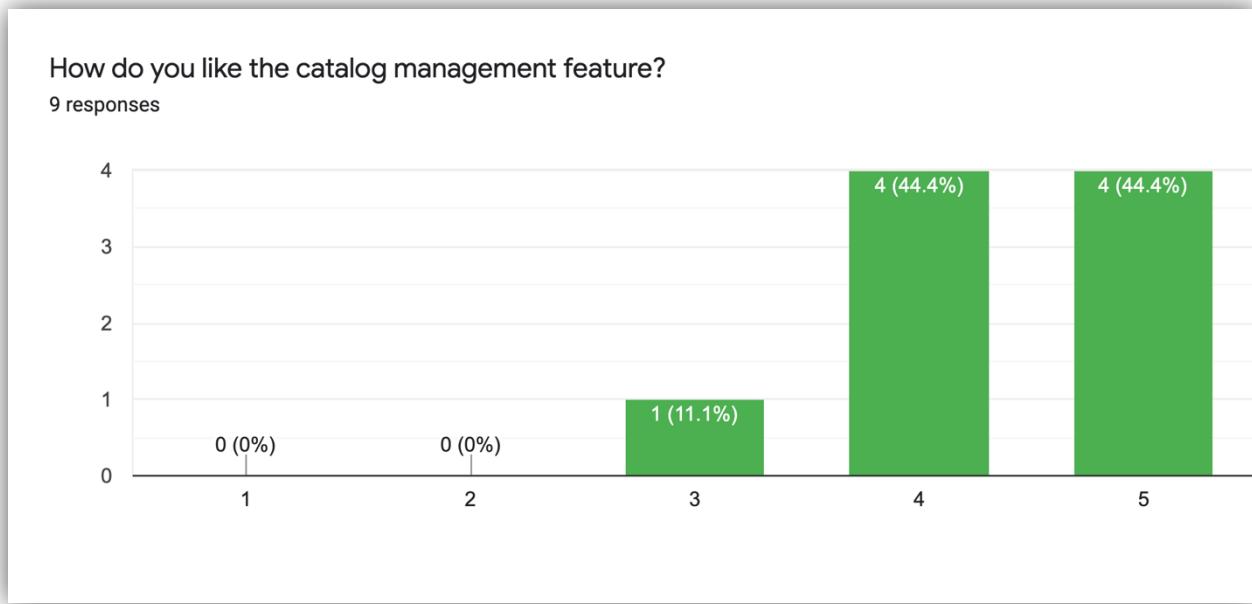


Figure 138 - Catalog management feature

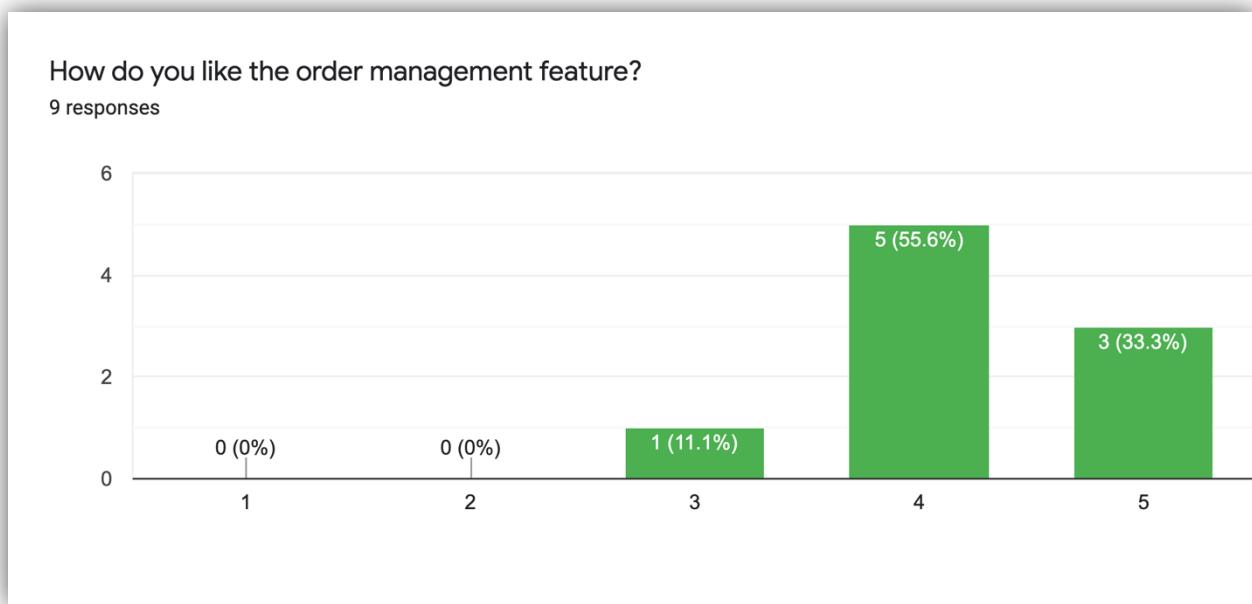


Figure 139 - Order management feature

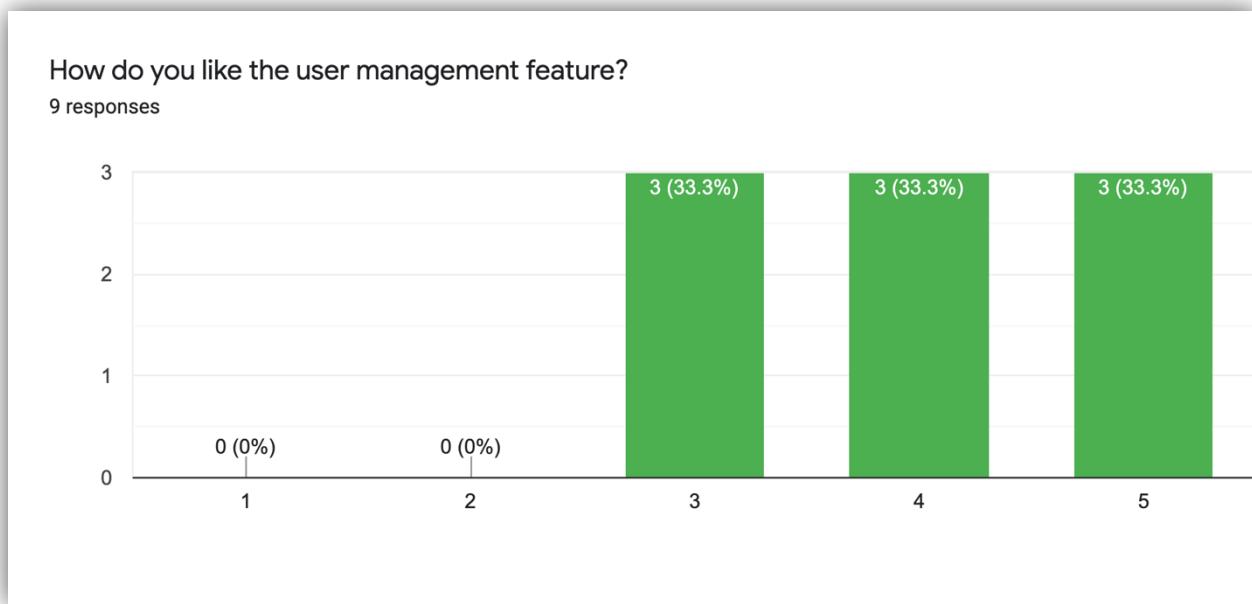


Figure 140 - User management feature

What should I add extra data to the dashboard?

9 responses

Visualization on how the waste is being managed in the city would be a good piece of data

I really loved the design aspect.

Add Graphs

No idea related to this topic

IDK

Not sure

None

Customer's Individual dustbin data

Extra charts to represent the data seamlessly.

Figure 141 - Extra data in the dashboard

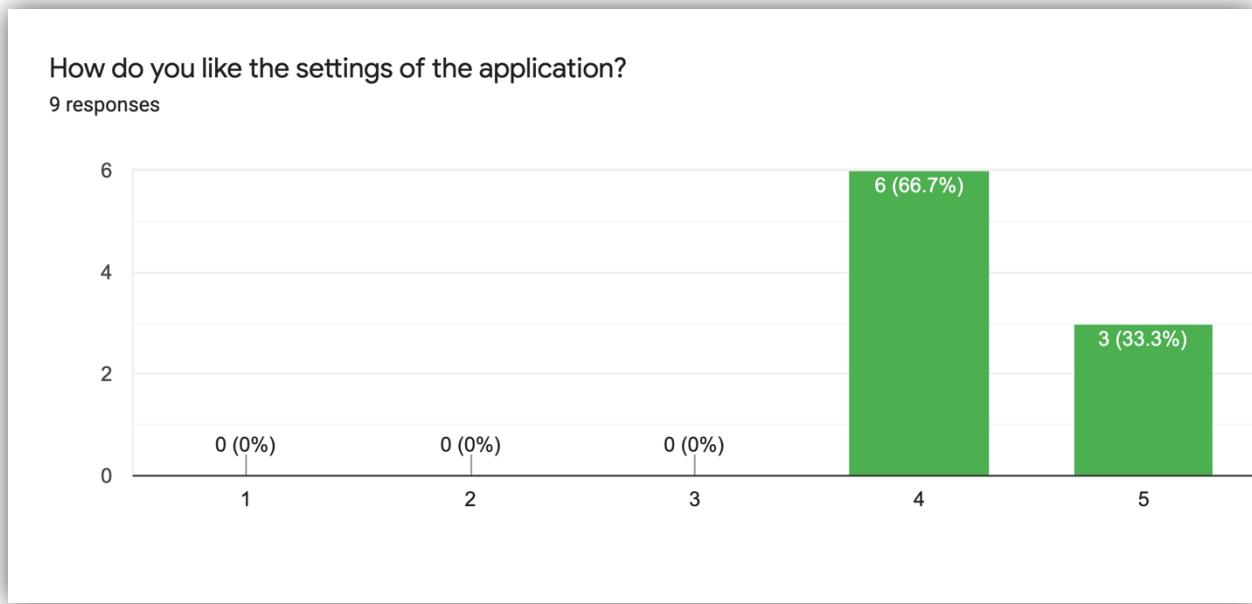


Figure 142 - Settings of the web application

Do you have any suggestions to improve the admin panel?

5 responses

The admin panel is great.

Its great

It would be more nice if the UI is more expanded

The admin panel is clean and minimal.

right now, no!!

Figure 143 – Suggestions to improve the admin panel

8.2.2.2 Mobile Application

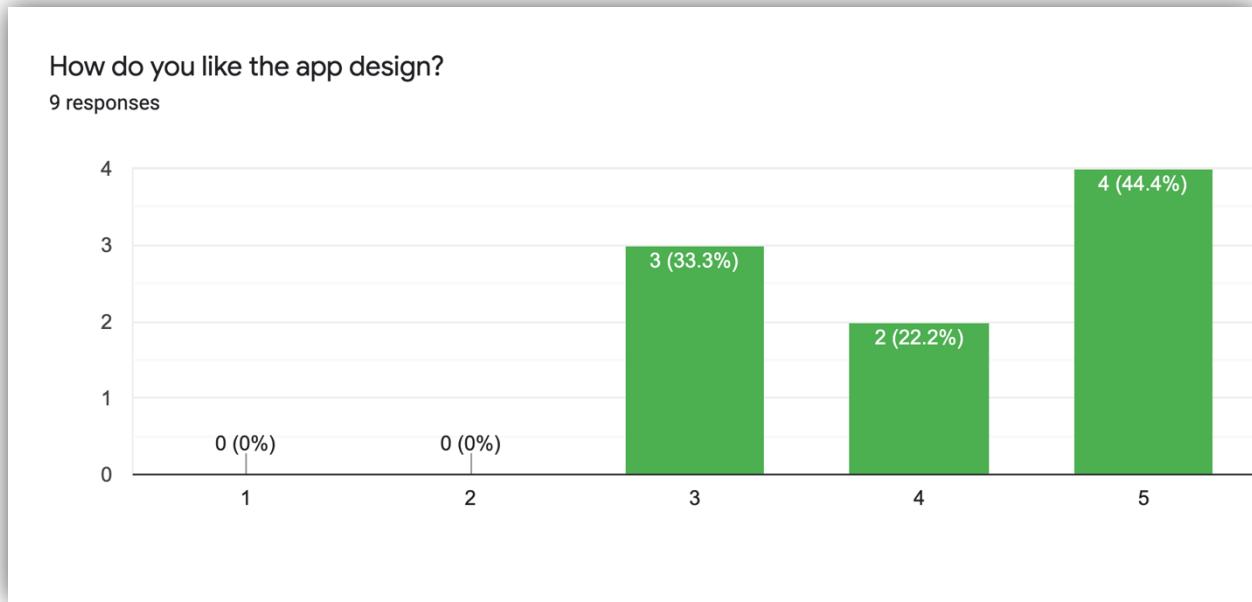


Figure 144 - Mobile application design

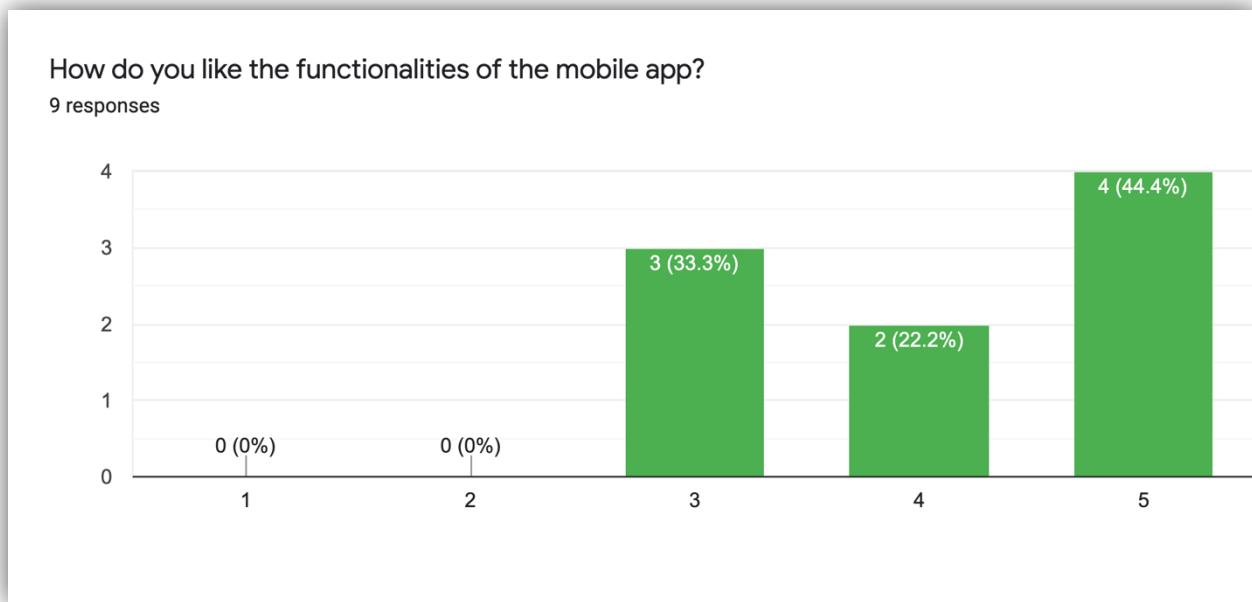


Figure 145 - Functionalities of the mobile application

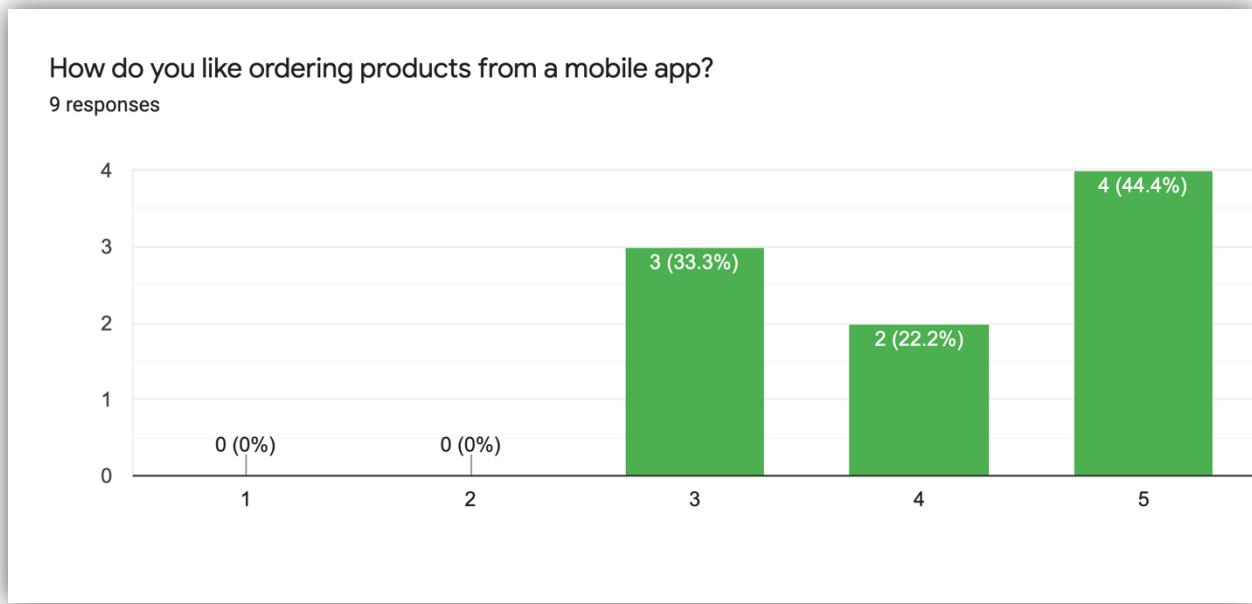


Figure 146 – Order products from a mobile app

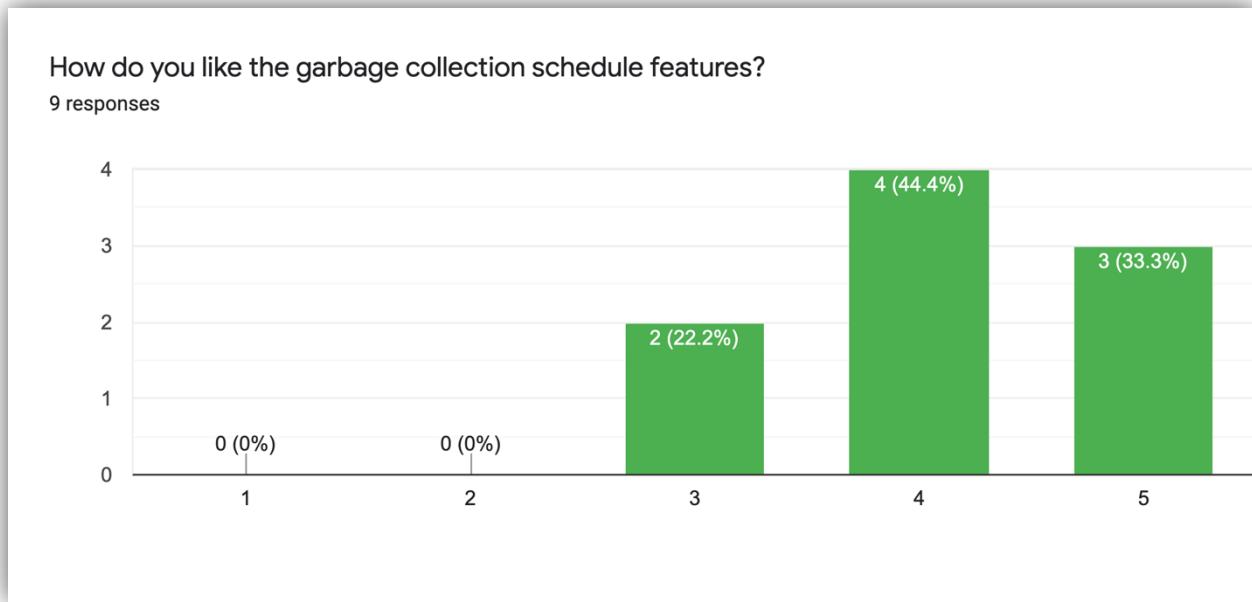


Figure 147 - Garbage collection schedule feature



Figure 148 - Employees (waste-collectors and delivery persons) app

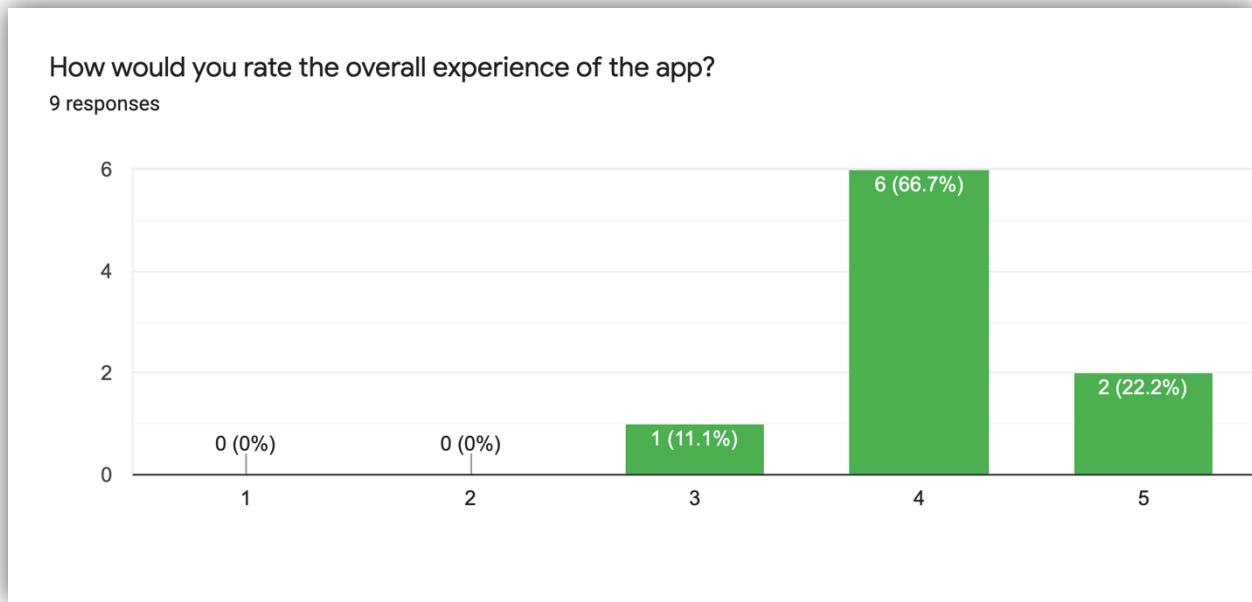


Figure 149 - Overall experience of the mobile app

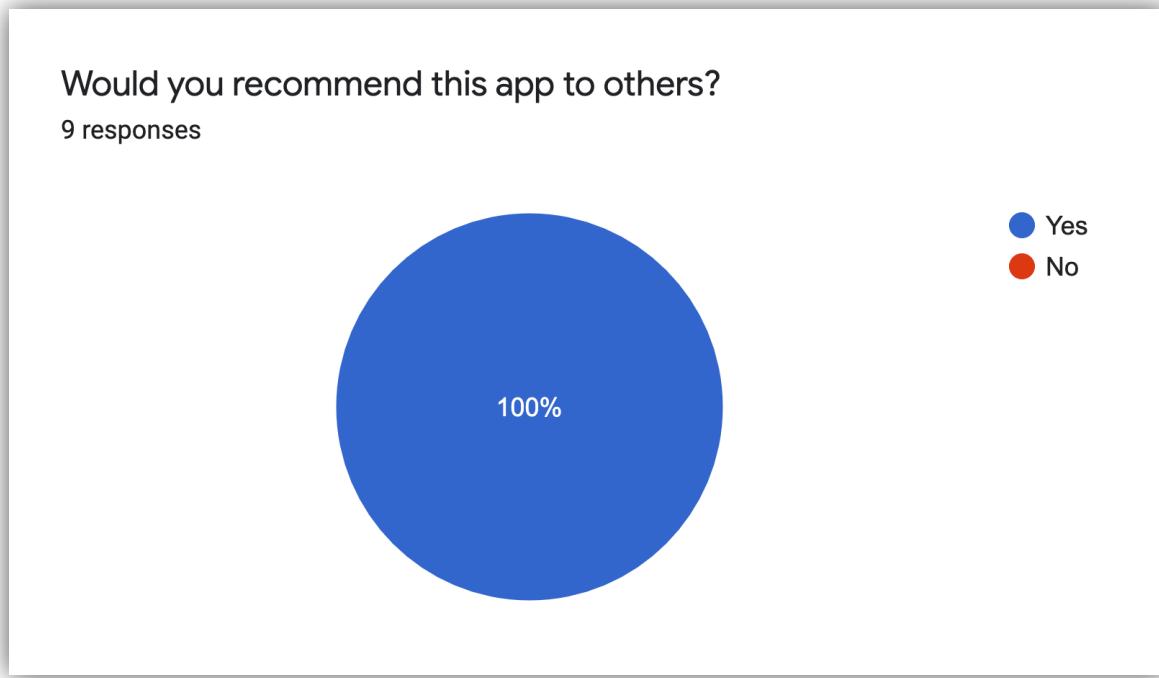


Figure 150 - Recommendation of this app to others

Do you have any suggestions to improve the mobile app?

3 responses

It will be great if user can select multiple Waste Type

no

None

Figure 151 - Respondents opinion for mobile app

8.3 Appendix C: Sample Codes

8.3.1 Sample Code of the UI

8.3.2 Sample Code for the Automation Script

8.4 Appendix D: Designs

8.4.1 Gantt Chart

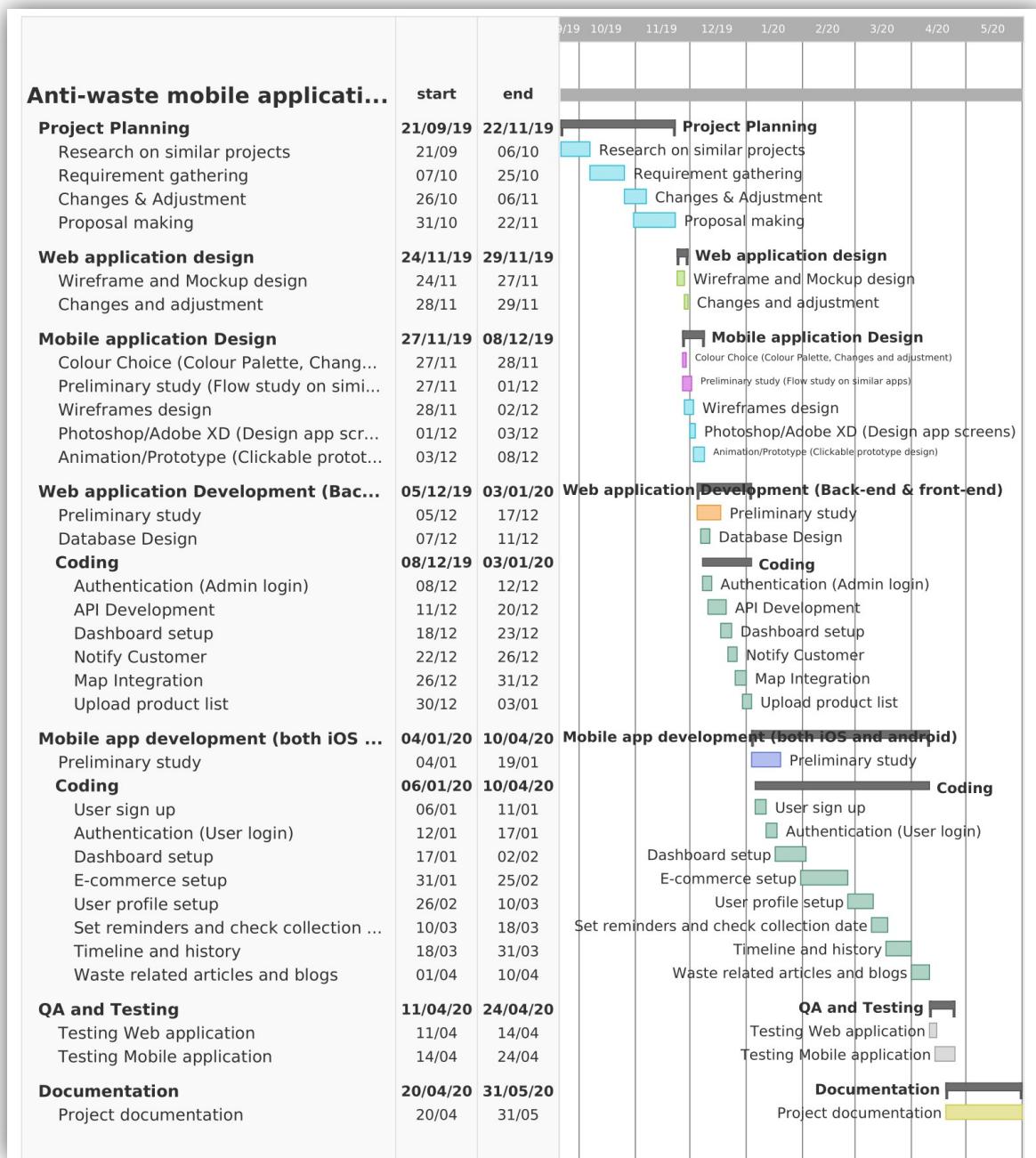


Figure 152 - Project Gantt chart

8.4.2 Work Breakdown Structure

A work-breakdown structure is divided into two sections; web application and mobile application. Essentially, using a work breakdown structure enables us to take a top-down look at the project and break it into the tasks and subtasks that will get to completion.

8.4.2.1 Web Application:

Web Application To Do		
✓	Task	Features
<input type="checkbox"/>	1. Market Research 2. Define functionality of the application 3. Plan the workflow of the webapp 4. Sitemap and wireframe creation 5. Blade templating 6. Navigation	Web Application Designing
<input type="checkbox"/>	1. Requirements gathering and understand the proposed system 2. Database analysis 3. Make an ER-Diagram 4. Create a database and migrate 5. Create a seed data for Locations	Database Designing
<input type="checkbox"/>	<u>Web</u> 1. Set up Laravel Authentication for login and sign up 2. Add form validation 3. Implement in front-end 4. Apply auth middleware for all web request	Laravel Login Authentication
<input type="checkbox"/>	<u>APIs</u> 1. Use laravel passport for token authentication 2. Make three API - signup, login, and logout 3. Add validation for APIs 4. Apply auth middleware for all APIs	
<input type="checkbox"/>	<u>Categories (Recycle Product):</u> Add, edit, delete, and view categories <u>Recycle Product:</u> Add, edit, delete, and view recycle products <u>Waste Type:</u> Add, edit, delete, and view waste type <u>Waste Container:</u> Add, edit, delete, and view waste container <u>Employees:</u> Add, edit, delete, and view employees (Waste Collectors, and Delivery Person)	CRUD Operation

Figure 153 - Web application work-break down structure

Web Application To Do		
✓	Task	Features
<input checked="" type="checkbox"/>	<p>Customers</p> <ul style="list-style-type: none"> 1. Get waste type details while signing up 2. Get waste container details while signing up 3. Search streets while signing up 4. Product list 5. Search product 6. Order create 7. Shopping history 8. User profile - waste container's details 9. Total weight of garbage collected 	
<input type="checkbox"/>	<p>Waste Collectors</p> <ul style="list-style-type: none"> 1. Show locations to waste-collector app 2. Customer's list per location to associated to waste collector <ul style="list-style-type: none"> - Show customer's waste container detail - Post weight of waste collected for individual waste container 3. Get garbage collection schedule 	API Development
<input type="checkbox"/>	<p>Delivery Persons</p> <ul style="list-style-type: none"> 1. Order list assigned to delivery person 2. Order update status (delivery person) 	
<input type="checkbox"/>	<ul style="list-style-type: none"> 1. Design a flow for setting time for each street 2. Use jQuery to manipulate data for storing in the server 3. Show updated data in front-end 	Predifined time setting
<input type="checkbox"/>	<ul style="list-style-type: none"> 1. Assign Delivery Person if the order status is pending <ul style="list-style-type: none"> - Use jQuery ajax to fetch delivery in assigned list 	List orders according to order status
<input type="checkbox"/>	<ul style="list-style-type: none"> 1. Design a flow for assigning waste collectors to multiple streets 2. Use jQuery ajax autocomplete to search for streets 3. Post selected waste collectors to that streets 4. Show updated data in front-end 	Assign Waste Collectors
<input type="checkbox"/>	<ul style="list-style-type: none"> 1. View Customer's List 	Customer's List

Figure 154 - Web application work-breakdown structure I

Web Application To Do		
✓	Task	Features
<input type="checkbox"/>	Create a logic to generate schedule based on waste container's size and last waste collected date for each waste container	View Customer's garbage pickup schedule
<input type="checkbox"/>	1. Recent order list 2. Statistics - Total orders from customers - Total revenue collected - Total garbage collected - Total products sold	Dashboard
	Garbage Collection Schedule Customers 1. Create a logic to notify customers (before 1 hour of collection time) 2. Create laravel job and assign timestamp for triggering that job 3. Put jobs to the queue 4. Run php artisan notify-waste-collection to notify customers	
<input type="checkbox"/>	Waste Collectors 1. Create a logic to notify waste collector (start of the day) 2. Create laravel job and put it to the queue for triggering that job 3. Run php artisan notify-waste-collection to notify waste collectors	Send expo push notifications
	When superhero assigned orders to delivery person Delivery Persons 1. Create laravel job and put it to the queue for triggering that job	
<input type="checkbox"/>		

Figure 155 - Web application work-breakdown structure II

8.4.2.2 Mobile Application

Mobile Application To Do		
✓	Task	Features
<input type="checkbox"/>	1. Market Research 2. Define functionality of the application 3. Plan the workflow of the mobile app 4. Sitemap and wireframe creation	Mobile Application Designing
<input type="checkbox"/>	1. Create a new app 2. Add icon and splash screens 3. App name and package name setup	Application Setup
<input type="checkbox"/>	1. Redux actions setup 2. Redux state setup 3. Make state and actions global setting up redux store	Redux setup
<input type="checkbox"/>	1. Expo notification package installation 2. Extract device ID and store in redux state	Expo Notification Setup
<input type="checkbox"/>	1. Make navigation global 2. Add screens 3. Different navigation for different user login state 4. Different navigation for different login role 5. Logout navigation change setup	React Navigation Setup
<input type="checkbox"/>	Customers 1. Login and register design 2. Waste container design and (API Implementation) 3. Pass register data to waste container selection screen 4. Waste container selection logic 5. Address selection screen design and (API Implementation) 6. Implement register validation logic 7. Register (API Implementation) 8. Login (API Implementation) 9. Notification list (API Implementation)	Login and Register

Figure 156 - Mobile application work-breakdown structure

Mobile Application To Do		
✓	Task	Features
<input type="checkbox"/>	1. Navigation drawer integration 2. Create custom navigation drawer design 3. Product gallery design 4. Product gallery (API Implementation) 5. Product detail design 6. Product order selection logic 7. Waste container selection design and (API Implementation) 8. Order (API Implementation) logic for both product and waste container	Customer's Dashboard
<input type="checkbox"/>	1. Blogs and articles screen design 2. Blogs and articles item design 3. Implement web view	Blogs and Articles
<input type="checkbox"/>	1. Profile design and menu listing 2. My Achievements screen design 3. Shopping history design 4. Shopping history API redux action & redux state data (API Implementation) 5. Waste container API redux action & redux state data (API Implementation) 6. Total garbage weight collected (API Implementation) & display in a profile 7. Implement Reward level logic with total garbage weight collected data	Customer Profile
<input type="checkbox"/>	1. Waste collection schedule screen design 2. Waste collection (API Implementation) 3. Item design	Waste Collection Schedule
<input type="checkbox"/>	1. Logout modal box design 2. Logical coding 3. Expo token retrieved 4. Logout (API Implementation) 5. Clear saved datas in logout	Logout
<input type="checkbox"/>	Dashboard a. View list of orders which were assigned by super-admin	Dashboard

Figure 157 - Mobile application work-breakdown structure I

Mobile Application To Do		
	Task	Features
✓	<p>Delivery Persons</p> <ul style="list-style-type: none"> 1. Navigation drawer and custom drawer design 2. Order list (API Implementation) 3. Order list item design 4. Order list detail navigation 5. Pass parameters to screen 6. Add Call feature 7. Order delivered (API Implementation) 8. Navigate dashboard on success 9. Notification list (API Implementation) 	
☐		Delivery Person Dashboard Screen
☐	<p>Waste Collectors</p> <ul style="list-style-type: none"> 1. Location List (API Implementation) 2. Location list picker integrated 3. Customer list (API Implementation) on Location picked 4. Customer list design 5. Customer detail design 6. Waste collection weight submit item design 7. Waste container weight submit (API Implementation) 8. Notification list (API Implementation) 	
☐		Waste Collector Dashboard Screen
☐		

Figure 158 - Mobile application work-breakdown structure II

8.4.3 Expanded Use Case Description (Web Application)

8.4.3.1 Log in to System

The detailed step by step description for use case ‘log in to system’ is as below:

Actor Action	System Response
1. Superhero enters username and password to login to the system.	
2. Superhero click the login button to login to the system.	
	3. System verify the username and password in the database.
	4. System displays the dashboard.

Alternative Action:

Line 2. If superhero enters username and password incorrect. Error message will be displayed.

8.4.3.2 View orders list

The detailed step by step description for use case ‘view recent order list’ is as below:

Actor Action	System Response
1. Superhero click the Orders button in the menu bar.	
	2. System displays the list of orders with details.
3. Superhero view list of total orders with details.	

8.4.3.3 Monitor the summary of the system

The detailed step by step description for use case ‘monitor the summary of the system’ is as below:

Actor Action	System Response
1. Superhero click the login button to login the system.	
	2. System displays the dashboard.
	3. System shows the statistics data in the dashboard.
	4. System shows the recent orders list in the dashboard.
4. Superhero view statistics data and can also view the recent orders list in the dashboard.	

8.4.3.4 View customer’s list

The detailed step by step description for use case ‘view customer’s list’ is as below:

Actor Action	System Response
1. Superhero click Customers in the menu bar.	
	2. System displays the list of total customers with details.
3. Superhero view the list of total customers with details.	

8.4.3.5 View employee's list

The detailed step by step description for use case ‘view employee’s list’ is as below:

Actor Action	System Response
1. Superhero click Employees in the menu bar.	2. System displays the list of total employees.
3. Superhero view the employee’s list details.	
4. Superhero assign a role to an employee.	
5. Superhero add an employee.	6. System add an employee in that table.
7. Superhero edit an employee.	8. System update the data of that employee.
9. Superhero delete an employee.	10. System delete the data of that employee.

8.4.3.6 View recycle product’s category

The detailed step by step description for use case ‘view recycle product’s category’ is as below:

Actor Action	System Response
1. Superhero click Categories in the menu bar.	
	2. System displays the list of total categories.
3. Superhero view the category list details.	
4. Superhero add a category.	5. System add a category in that table.
6. Superhero edit a category.	7. System update the data of that category.
8. Superhero delete a category.	9. System delete the data of that category.

8.4.3.7 View recycled products

The detailed step by step description for use case ‘view recycled products’ is as below:

Actor Action	System Response
1. Superhero click Recycle Products in the menu bar.	
	2. System displays the list of totals recycled products.
3. Superhero view the list of totals recycled products.	
4. Superhero assign category to a recycled product.	
5. Superhero add a recycled product.	6. System add a recycled product in that table.
7. Superhero edit a recycled product.	8. System update the data of that recycled product.
9. Superhero delete a recycled product.	10. System delete the data of that recycled product.

8.4.3.8 View waste type list

The detailed step by step description for use case ‘view waste type list’ is as below:

Actor Action	System Response
1. Superhero click Waste Types in the menu bar.	
	2. System displays the list of total waste types.
3. Superhero view the list of total waste types.	
4. Superhero add a waste type.	5. System add a waste type in that table.
6. Superhero edit a waste type.	7. System update the data of that waste type.
8. Superhero delete a waste type.	9. System delete the data of that waste type.

8.4.3.9 View waste containers list

The detailed step by step description for use case ‘view waste containers list’ is as below:

Actor Action	System Response
1. Superhero click Waste Containers in the menu bar.	2. System displays the list of total waste containers.
3. Superhero view the list of total waste containers.	
4. Superhero add a waste container.	5. System add a waste container in that table.
6. Superhero edit a waste container.	7. System update the data of that waste container.
8. Superhero delete a waste container.	9. System delete the data of that waste container.

8.4.3.10 Assign waste collector to the streets

The detailed step by step description for use case ‘assign waste collector to the streets’ is as below:

Actor Action	System Response
1. Superhero click Assign Employees in the menu bar.	2. System displays the Assign Waste Collector page.
3. Superhero search for the streets.	
4. Superhero select streets.	
5. Superhero choose the waste collector.	
6. Superhero assign the waste collector to the streets.	7. System assign the waste collector to that street.

8.4.3.11 Assign orders to the delivery person

The detailed step by step description for use case ‘assign orders to the delivery person’ is as below:

Actor Action	System Response
1. Superhero click Assign Employees in the menu bar.	
	2. System displays the Assign Delivery Person page.
3. Superhero select the order.	
4. Superhero choose the delivery person.	
5. Superhero assign pending orders to the delivery person.	6. System assign orders to the delivery person.

8.4.3.12 Set schedule for garbage pickup

The detailed step by step description for use case ‘set schedule for garbage pickup’ is as below:

Actor Action	System Response
1. Superhero click Settings in the menu bar.	2. System displays the Set Schedule page.
3. Superhero select the province and district.	
	4. System displays the list of streets.
5. Superhero set the time to the district.	6. System save the time.

Alternative Action:

Line 3: Superhero can also search streets directly.

8.4.3.13 View waste collection schedule details

The detailed step by step description for use case ‘view waste collection schedule details’ is as below:

Actor Action	System Response
1. Superhero click Settings in the menu bar.	2. System displays the Collection Schedule page.
3. Superhero select the province and district.	
	4. System displays the collection schedule details for individual customers according to the streets.
5. Superhero view the collection schedule details for individual customers according to the streets.	

Alternative Action:

Line 3: Superhero can also search streets directly.

8.4.4 Expanded Use Case Description (Mobile Application)

8.4.4.1 Sign up mobile app

The detailed step by step description for use case ‘sign up mobile app’ is as below:

Actor Action	System Response
1. Customer click the register button to sign up mobile app.	2. System displays the sign-up page.
3. Customer enters the personal details.	
4. Customer click the Continue button.	5. System display the choose waste type and waste container page.
6. Customer choose waste type and waste container.	
7. Customer click the Continue button	8. System display the address details page.
9. Customer enters the address details.	
10. Customer click the Register button.	11. Customer registered in the application.

Alternative Action:

Line 6: Customer doesn't need to choose waste type and waste container if he/she don't want to sell the waste.

8.4.4.2 Log in mobile app

The detailed step by step description for use case ‘log in mobile app’ is as below:

Actor Action	System Response
	1. System displays the log-in page.
2. Users enters the username and password.	
3. Users clicks the log in button to log in mobile app.	4. System verify the username and password in the database.
	5. System displays the dashboard.

Alternative Action:

Line 2. If Customer, Waste Collector or Delivery Person enters username and password incorrect. Error message will be displayed.

8.4.4.3 View products list

The detailed step by step description for use case ‘view products list’ is as below:

Actor Action	System Response
	1. System displays the dashboard page.
2. Customer view the products from dashboard.	
3. Customers clicks the details button to view the details of the products.	
4. Customer choose quantity of the items.	
5. Customer click Order button to buy the product.	6. System update the order data.

8.4.4.4 View waste collection schedule

The detailed step by step description for use case ‘view waste collection schedule’ is as below:

Actor Action	System Response
1. Customer click Collection Schedule on navigation drawer.	2. System displays the waste collection schedule page.
3. Customers view the waste collection schedule of their street.	

8.4.4.5 View user profile

The detailed step by step description for use case ‘view user profile’ is as below:

Actor Action	System Response
1. Customer click profile on navigation drawer.	2. System displays the profile page.
3. Customers view the total garbage collected and rewards level.	
4. Customer click My Achievements.	5. System displays My Achievements page.
6. Customer view the waste containers list which they are using.	
7. Customer click Shopping History.	8. System displays Shopping History page.
9. Customer view the history of their purchased orders.	

8.4.4.6 View articles and blogs

The detailed step by step description for use case ‘view articles and blogs’ is as below:

Actor Action	System Response
1. Customer click Articles and Blogs on navigation drawer.	2. System displays the waste article and blogs page.
3. Customers view the article and blogs related to waste management and recycling.	

8.4.4.7 View order list

The detailed step by step description for use case ‘view order list’ is as below:

Actor Action	System Response
1. Delivery person login to the mobile app	2. System displays order list page.
3. Delivery person view the list of orders which were assigned to them.	
4. Delivery person click on order list.	
5. Delivery person view the orders details.	
6. Delivery person click the delivered button when order gets delivered.	7. System update the order status.

8.4.4.8 Receive notifications

The detailed step by step description for use case ‘receive notifications’ is as below:

Actor Action	System Response
1. User click Notifications on navigation drawer.	2. System displays the notification page.
3. User view the notification list.	

8.4.4.9 View customer's list according to location

The detailed step by step description for use case ‘view customer’s list according to location’ is as below:

Actor Action	System Response
1. Waste collector login to the mobile app	
2. Waste collector click the dropdown button to view the streets which were assigned to them.	
3. Waste collector click on the street name.	4. System displays the customer list of that street.
5. Waste collector view the list of customers of that street.	
6. Waste collector click on customer list.	
7. Waste collector view the customer details.	
8. Waste collector update the total waste collected of that customer.	9. System update the total waste collected of that customer.

Alternative Action:

Line 8. If waste collector enters more weight than waste container capacity, error message will be shown.

8.5 Appendix E: Testing and Analysis

8.5.1 Compatibility Testing

8.5.1.1 Web Application

The compatibility of the web application was tested on different operating system, browsers and mobile platforms to validates how the system behaves and runs in a different environment.

ID	Test Case Description	Environment	Expected Output	Result
TC51	Testing application on different operating system	1. Windows 10 Operating System 2. MacOS Catalina 10.15.4	Application should work seamlessly on both operating system	Pass
TC52	Testing application on different web browser	1. Chrome 2. Safari 3. Opera web browser 4. Microsoft Edge	Application should work seamlessly on all web browsers	Pass
TC53	Testing application on mobile platforms	1. iPhone 2. Android phone 3. iPad	Application should work seamlessly on mobile platforms	Pass

Table 26 - Test cases of web application compatibility testing

Below show the screenshots for test case **TC51** (testing application on different operating system):

Windows 10 Operating System

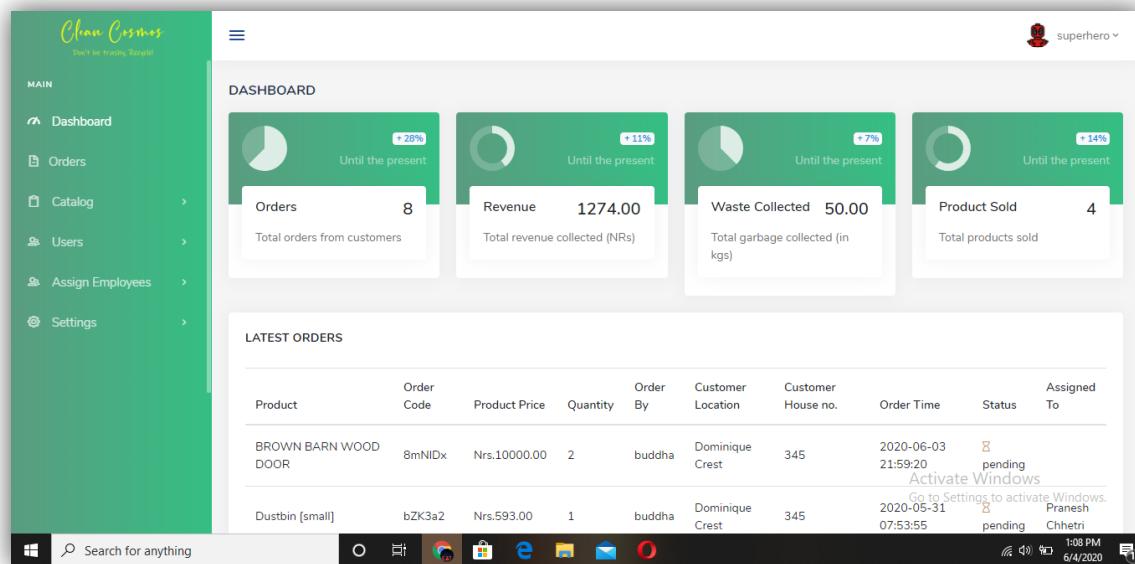


Figure 159 – Web application compatibility testing on windows operating system

MacOS Catalina 10.15.5

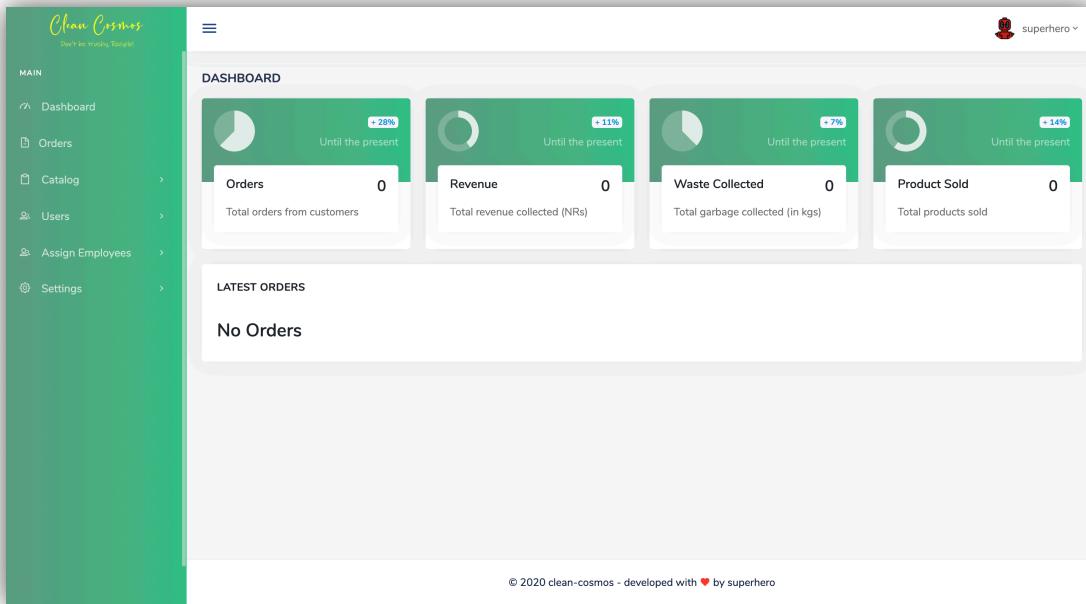


Figure 160 – Web application compatibility testing on macOS Catalina 10.15.4

Below show the screenshots for test case **TC52** (testing application on different web browser):

Google Chrome

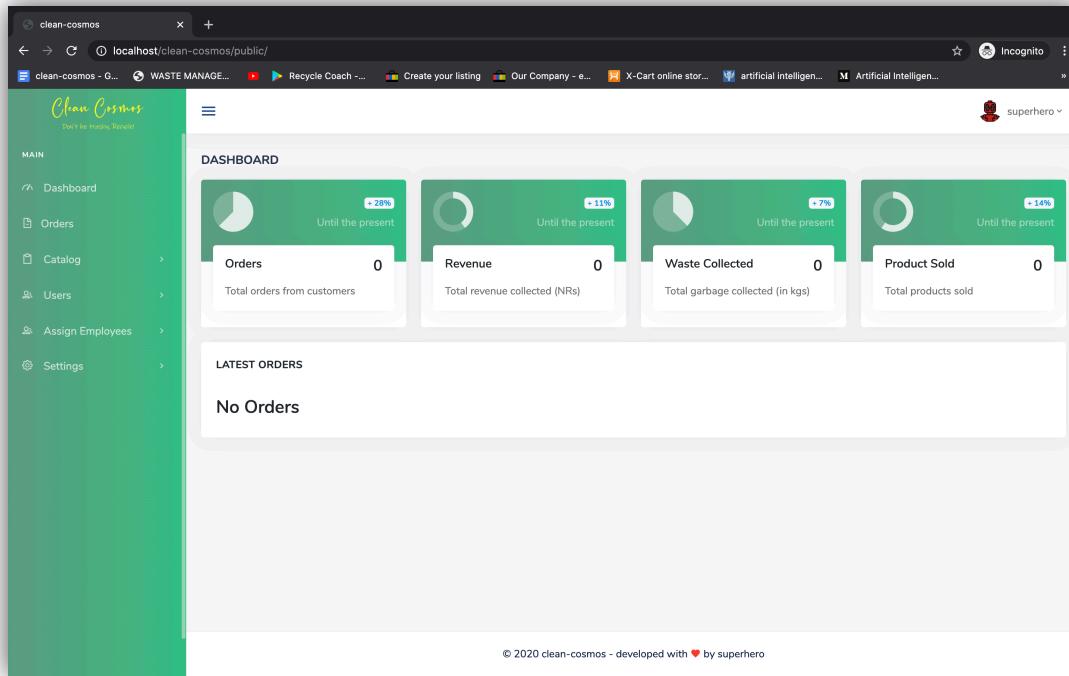


Figure 161 – Web application compatibility testing on google chrome browser

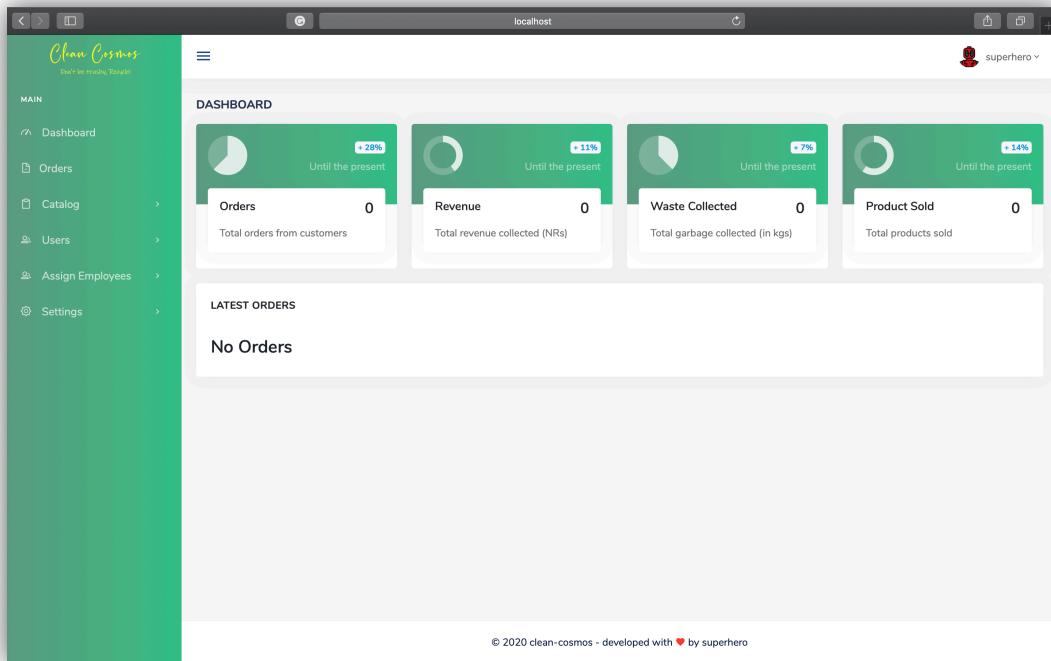
Safari

Figure 162 – Web application compatibility testing on safari browser

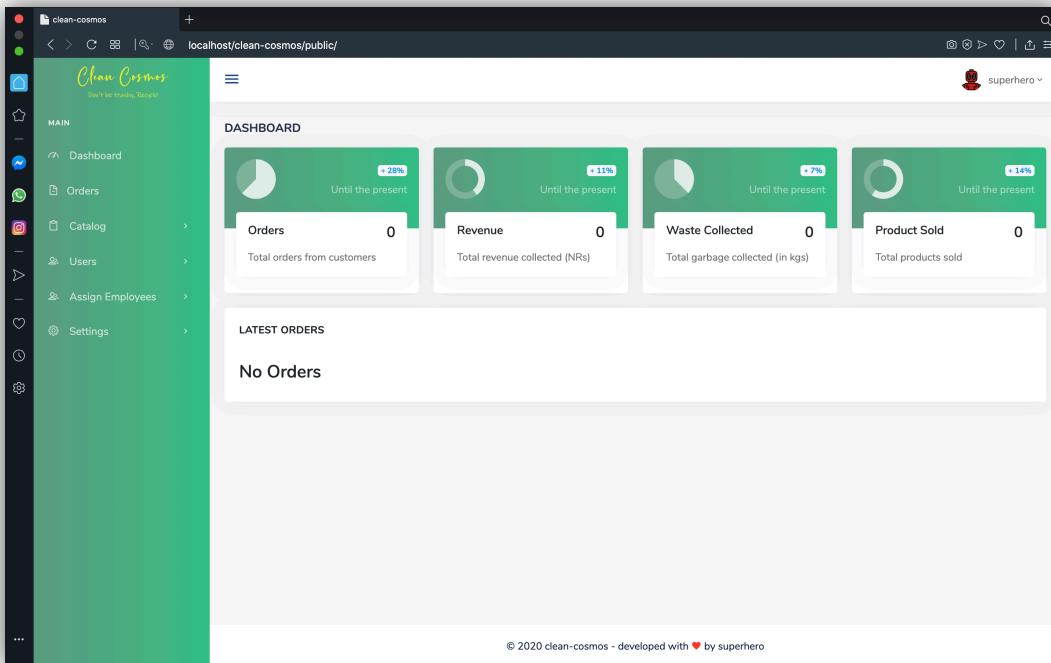
Opera Web Browser

Figure 163 – Web application compatibility testing on opera web browser

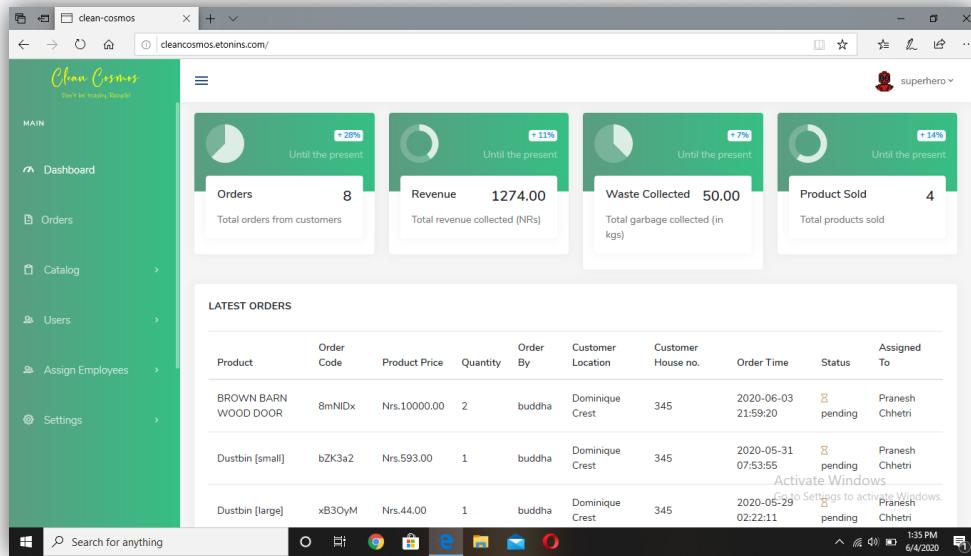
Microsoft Edge

Figure 164 – Web application compatibility testing on Microsoft edge

Below show the screenshots for test case **TC53** (testing application on mobile platform):

iPhone, Android Phone and iPad

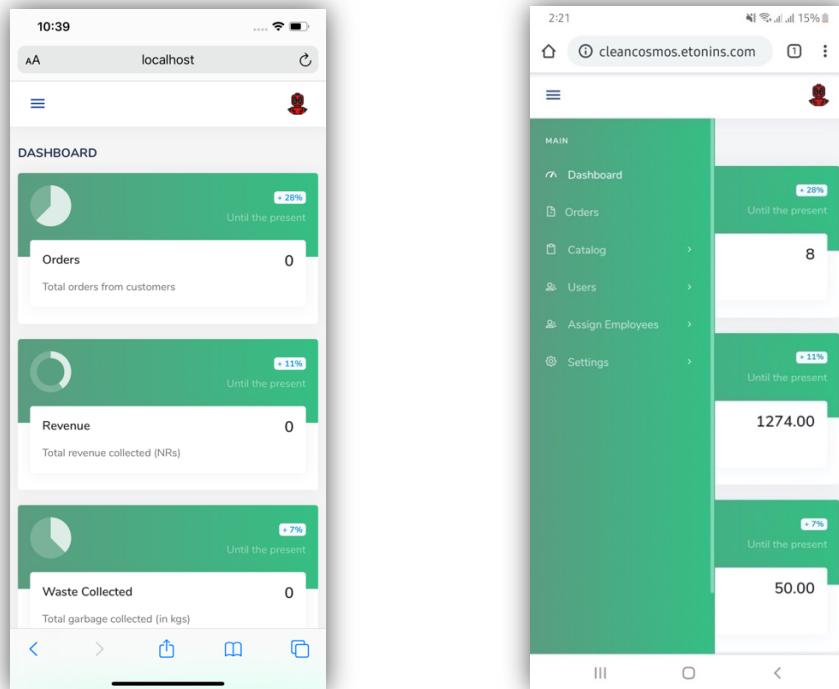


Figure 165 – Web application compatibility testing on iPhone and android phone

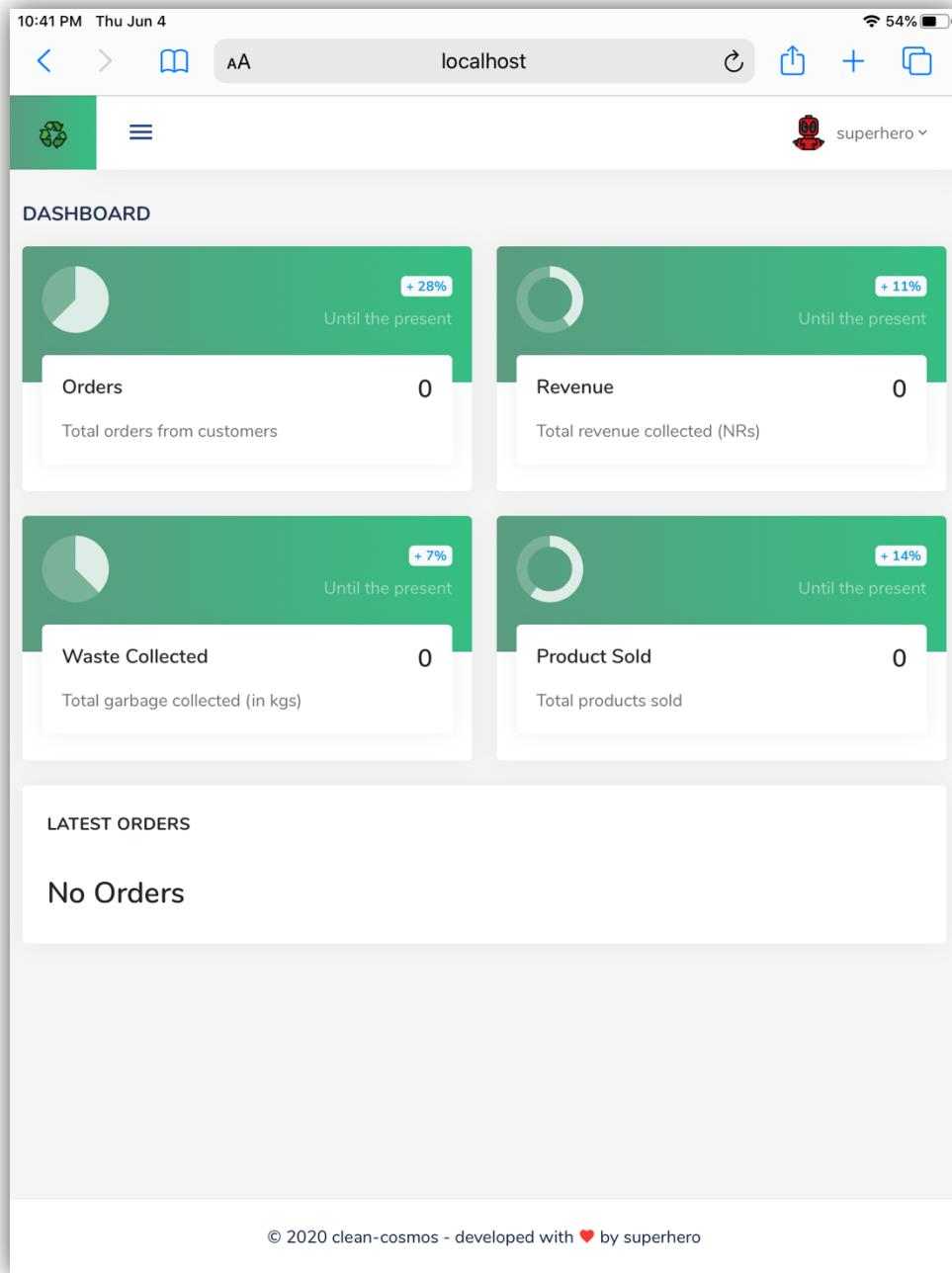


Figure 166 – Web application compatibility testing on iPad

8.5.1.2 Mobile Application

The compatibility of the mobile application was tested on different mobile devices to validate how the mobile application behaves and runs in a different screen, size and resolution.

ID	Test Case Description	Environment	Expected Output	Result
TC54	Testing mobile app on different mobile devices	1. iPhone 11 Pro Max 2. iPhone SE (2 nd generation) 3. iPad Pro (9.7 inch) 4. Android Phone (Samsung)	User interface of the mobile app should be as per the screen size of the device	Pass

Below show the screenshots for test case **TC54** (testing mobile app on different mobile devices):
iPhone 11 Pro Max and iPhone SE (2nd generation)

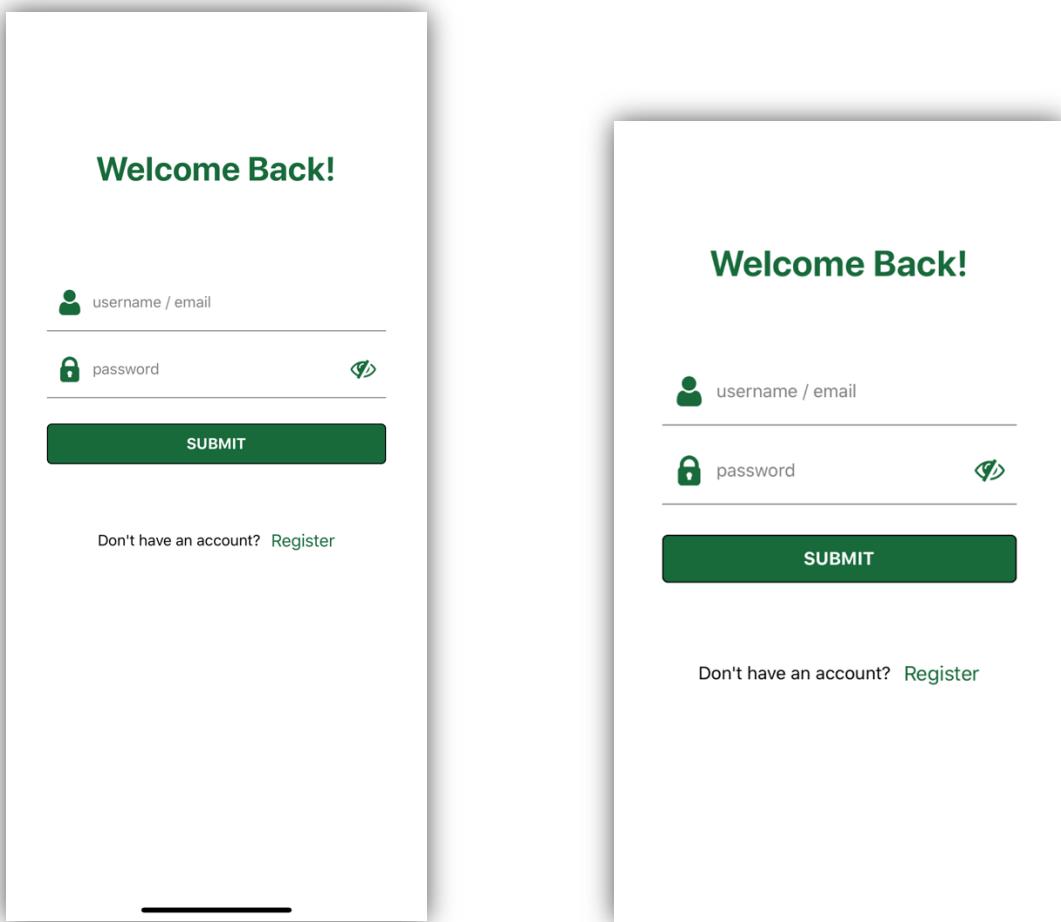


Figure 167 – Mobile application compatibility testing on iPhone 11 Pro Max and iPhone SE (2nd generation)

iPad Pro 9.7 inch

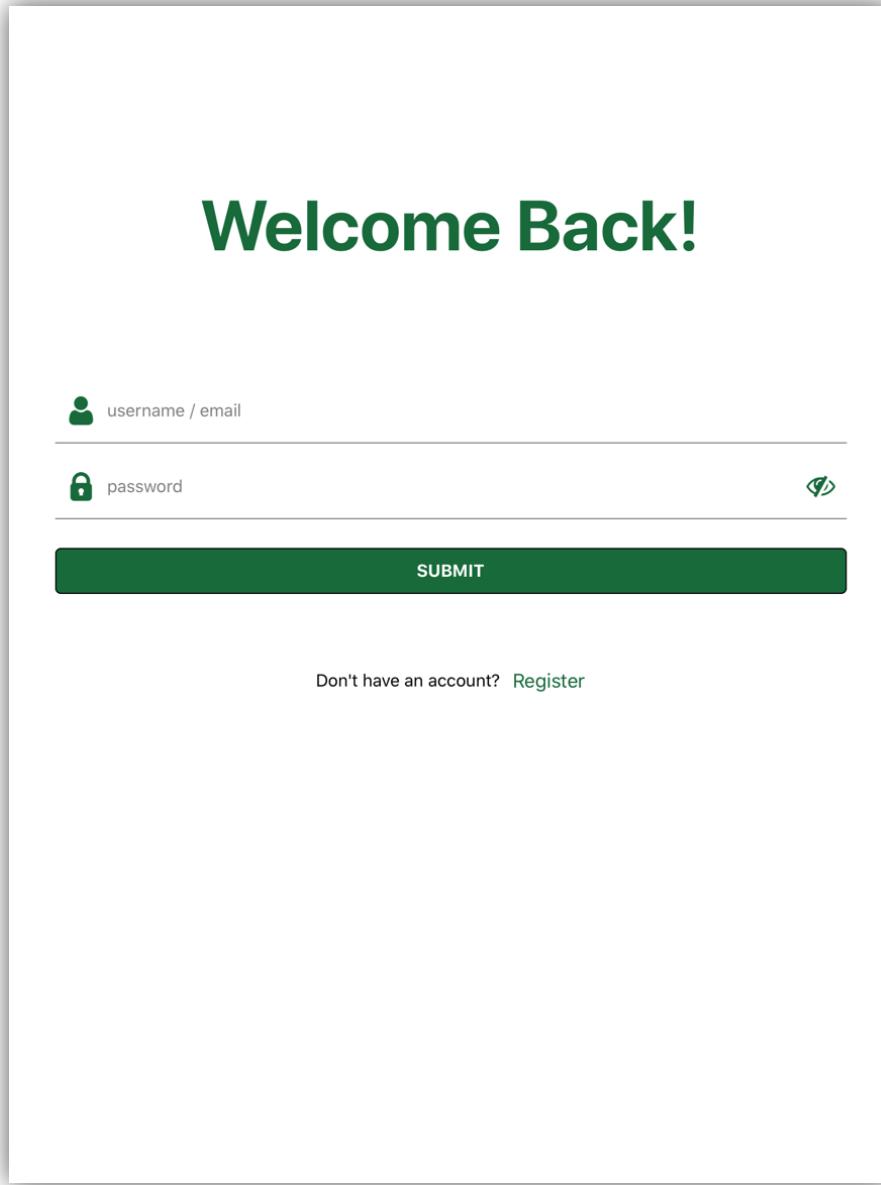


Figure 168 - Mobile application compatibility testing on iPad Pro (9.7 inch)

Android Phone (Samsung)

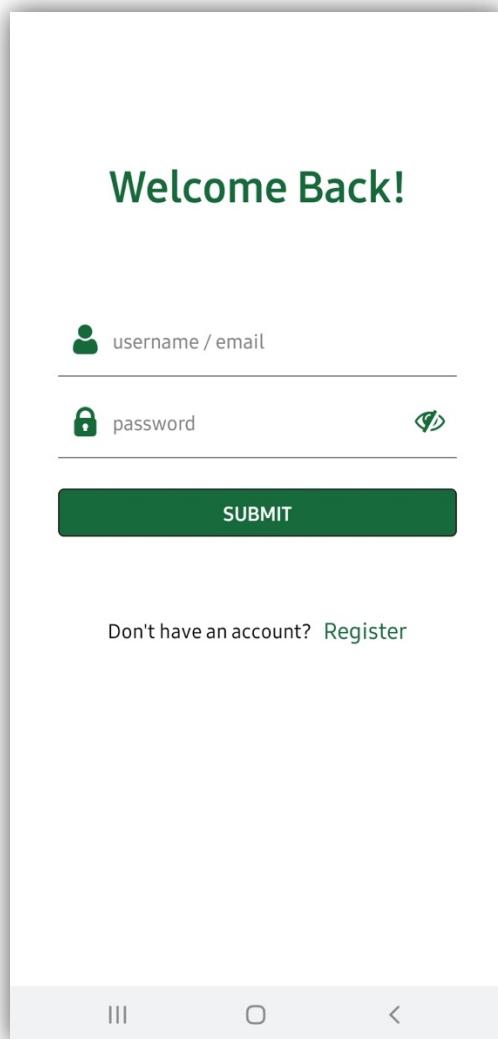


Figure 169 - Mobile application compatibility testing on Android Phone (Samsung)

8.5.2 Acceptance Testing

Once the unit and integration testing were completed, user acceptance testing was done. This testing was performed via friends and others to identify whether the end to end flow of the application is as per the needs of the end-user or not. Basically, the web and mobile applications were tested by the users to ensure that the system is working in the way it has to.

In this testing phase, some of the design or UI bugs were encountered in the mobile application which later fixed and re-tested. This testing helps me in finding appropriate data to show in the customer's waste collection schedule table which later added to that table. This testing also helps in determining how efficiently all the previous testing was performed.