



FPT UNIVERSITY

Capstone Project Document

Rheinol Quick Service

Group 15 - IS	
Group members	Lê Đình Thiện Vũ – SE62590 (Leader) Phan Ánh Phúc – SE62543 Đặng Hữu Phương Nam – SE62167
Supervisor	Nguyễn Thị Cẩm Hương
Ext. Supervisor	N/A
Project Code	RQS

– Ho Chi Minh City, 20th August, 2019 –

A. Introduction.....	15
1. Project Information.....	15
2. Introduction	15
3. Current Situation	15
4. Problem Definition.....	15
5. Proposed Solution.....	15
5.1 Main features	16
5.2 Advantages and disadvantages	16
5.2.1 Advantages.....	16
5.2.2 Disadvantages	16
6. Functional Requirement.....	16
6.1 Buying products and accessories:.....	16
6.2 Booking the services:.....	16
6.3 Launch new product, service and promotion:.....	17
6.4 Notifying maintenance schedule:.....	17
6.5 Statistic report regarding all of the services:.....	17
7. Role and Responsibility	17
B. Software Project Management Plan	17
1. Problem Definition.....	17
1.1 Problem Abstract.....	18
1.2 Project Overview	18
1.2.1 Current situation	18
1.2.2 The proposed system	18
1.2.3 Boundaries of the system	18
1.2.4 Future plans	19
1.2.5 Development environment.....	19
1.2.5.1 Hardware requirements	19
2. Project Organization	20
2.1 Software Process Model.....	20
2.2 Roles and responsibilities	21
2.3 Tools and Techniques	22
3. Project Management Plan	22
3.1 Requirement.....	22
3.2 Mockup UI	22
3.3 All meeting minutes.....	22
4. Coding Convention.....	22
C. Software Requirement Specification.....	23
1. User Requirement Specification	23
1.1 Guest Requirement	23
1.2 Member Requirement.....	23
1.3 Car Owner Requirement.....	23

1.4	Sales clerk Requirement	24
1.5	Director Requirement.....	24
2.	System Requirement Specification	24
2.1	External Interface Requirement	24
2.1.1	User Interface	24
2.1.2	Hardware Interface	24
2.1.3	Software Interface	24
2.1.4	Communication Protocol	24
2.2	System Overview Use Case.....	24
2.3	List of use case.....	26
2.3.1	<Guest> Overview Use Case	26
2.3.1.1	Sign in	26
2.3.1.2	Sign up	29
2.3.2	<Authorized user> Overview use case	31
2.3.2.1	Register a car	31
2.3.2.2	Edit a car.....	33
2.3.2.3	Remove a car	36
2.3.2.4	Register bank card	37
2.3.2.5	Cash in wallet.....	40
2.3.2.6	Cash-out wallet.....	42
2.3.2.7	View promotion detail.....	44
2.3.2.8	Change Password	46
	46
2.3.2.9	Edit an account	48
2.3.2.10	Create the order	50
2.3.3	<Technical> Overview use case.....	52
2.3.3.1	Verify service order	53
2.3.3.2	Complete task.....	55
2.3.4	< Sales clerk > Overview use case	57
2.3.4.1	Confirm to do Order	57
2.3.4.2	Add account.....	59
2.3.4.3	Change status account	62
2.3.4.4	Edit order.....	63
	64
2.3.5	<Director> Overview use case.....	66
2.3.6	<Admin> Overview use case	68
2.3.6.1	Add a product into the system	69
2.3.6.2	Update existed product.....	71
2.3.6.3	Add a category into the system.....	74
2.3.6.4	Update a category	76
2.3.6.5	Add a promotion into the system.....	78
2.3.6.6	Delete a promotion.	81
2.3.6.7	Add a service.	83
2.3.6.8	Edit a service.....	86
2.3.7	<System> Overview use case	88
2.3.7.1	Notify to customer	89
2.3.7.2	Filter order timeout.....	91
2.3.7.3	Collect Data	93

2.3.7.4	Suggest promotion	94
2.3.7.5	Filter hot service	96
2.3.7.6	Filter hot product	97
2.3.7.7	Remove an Account.....	99
3.	Software System Attribute	100
3.1	Usability	100
3.2	Reliability	100
3.3	Availability	101
3.4	Security.....	101
3.5	Maintainability.....	101
3.6	Portability	101
3.7	Performance	101
4.	Conceptual Diagram.....	101
D.	Software Design Description	103
1.	Design Overview	103
2.	System Architectural Design.....	103
2.1	Web application architecture description	103
3.	Component Diagram.....	104
4.	Detail Description	106
4.1	Class Diagram	106
4.2	Class Diagram Explanation.....	106
4.2.1	Employee	107
4.2.2	Appointment	108
4.2.3	AppointmentService	109
4.2.4	Car	109
4.2.5	CarRegistration	110
4.2.6	Category	110
4.2.7	Garage	111
4.2.8	Image	111
4.2.9	Notification	111
4.2.10	Order	112
4.2.11	OrderProduct	113
4.2.12	Product.....	113
4.2.13	Promotion	114
4.2.14	PromotionCombo.....	114
4.2.15	PromotionRole	115
4.2.16	PromotionType.....	115
4.2.17	Role	115
4.2.18	Service	116
4.2.19	StatusAppointment	116
4.2.20	StatusOrder	117
4.2.21	Transaction.....	117
4.2.22	TransactionType	117
4.3	Interaction Diagram.....	118
4.3.1	Login.....	118

4.3.2	Add new Order.....	119
4.3.3	Update an Order	120
4.3.4	Add new Category.....	121
4.3.5	Update Category	121
4.3.6	Add new Service.....	122
4.3.7	Update a Service	122
4.3.8	Add new Product.....	123
4.3.9	Update a product	123
4.3.10	Add new an Account	124
4.3.11	Add new a Promotion	124
4.3.12	Filter hot product	125
4.3.13	Filter hot service.....	125
4.3.14	Notify customer.....	126
4.3.15	Filter order timeout.....	126
4.3.16	Register card.....	127
4.3.17	Payment order	127
4.3.18	Book service	128
4.3.19	Buy product.....	129
	130
4.3.20	Register account.....	130
4.3.21	Edit account.....	131
5.	Interface.....	131
5.1	User Interface Design	131
5.1.1	Web application Screen	131
5.1.1.1	Login	131
5.1.1.2	Menu Admin	132
5.1.1.3	Manage Product	133
5.1.1.4	Manage service.....	134
5.1.1.5	Manage promotion.....	136
5.1.1.6	Manage category.....	137
5.1.1.7	Manage order	138
5.1.1.8	Manage account	140
5.1.1.9	View revenue report.....	142
5.1.1.10	View rating report	143
5.1.2	Mobile Screen	144
5.1.2.1	Login	144
5.1.2.2	Home Screen	145
5.1.2.3	List Categories of Service.....	146
5.1.2.4	List of promotion	147
5.1.2.5	Promotion's detail	148
5.1.2.6	Cart	150
5.1.2.7	Book time	151
5.1.2.8	Customer	152
6.	Database Design	154
6.1	Entity Relationship Diagram (ERD)	154
6.2	Data Dictionary	154
E.	System Implementation & Test.....	158

1. Introduction	158
1.1 Overview.....	158
1.2 Test Approach.....	158
2. Data Relationship Diagram.....	159
2.1 Physical Diagram.....	159
2.2 Data Dictionary	159
3. Test Plan.....	160
3.1 Features to be tested	160
3.2 Features not to be tested	161
4. System Testing Test Case.....	161
4.1 Authorized User Test Case.....	161
4.1.1 Buy Product Test	162
4.1.2 Book Service Test	163
4.1.3 Suggest Promotion Test	164
4.1.4 Register Card Test	165
4.1.5 Withdraw Money Test	166
4.1.6 Recharge Money Test	166
4.1.7 Scheduler Maintain Test	167
4.1.8 Upgrade Account Test.....	168
4.1.9 Notification Test.....	170
4.1.10 Register Car Test	171
4.2 Technical Employees Test Case	172
4.2.1 Recognize Voice Test.....	172
F. Software User's Manual.....	175
1. Installation Guide	175
1.1 Setting up the environment at the server-side	175
1.1.1 Hardware requirements.....	175
1.1.2 Software requirements	175
1.2 Deployment at server side.....	176
1.2.1 Prepare deployment package	176
1.2.2 Configure server before deploy	176
1.3 Setting up the environment at the client-side.....	176
2. User Guide.....	176
2.1 Admin	176
2.1.1 Product.....	176
2.1.1.1 Add product.....	176
2.1.1.2 Edit product.....	178
2.1.1.3 Search product	179
2.1.1.4 Disable product	179
2.1.1.5 Add the product by import excel.....	180
2.1.2 Service	181
2.1.2.1 Add service	181
2.1.2.2 Edit service	182
2.1.2.3 Search service	184
2.1.3 Promotion	184

2.1.3.1	Add promotion	184
2.1.3.2	Notify promotion.....	186
2.1.3.3	Delete promotion	186
2.1.4	Category.....	187
2.1.4.1	Add category	187
2.1.4.2	Disable category	188
2.1.5	Garage	189
2.1.5.1	Edit information garage.....	189
2.2	Sales clerk	190
2.2.1	Account	190
2.2.1.1	Add account.....	190
2.2.1.2	Disable account	191
2.2.2	Order.....	192
2.2.2.1	Add order.....	192
2.2.2.2	Edit order.....	193
2.2.2.3	Cancel order	194
2.3	Customer	194
2.3.1	Create order have service	194
2.3.2	Buy product.....	199
2.4	Technical.....	202
2.4.1	Confirm order.....	202

List Table

Table 1 - Role and responsibility	17
Table 2 - Hardware requirements for server.....	19
Table 3 - Hardware requirement for PC.....	20
Table 4 - Software requirement	20
Table 5 - Role and responsibilities	22
Table 6 - Tools and techniques.....	22
Table 7 - Sign in use case.....	28
Table 8 - Sign Up.....	30
Table 9 - Register a car	33
Table 10 - Edit car.....	35
Table 11 - Delete car	37
Table 12 - Register bank card	39
Table 13 - Cash in Wallet.....	41
Table 14 - Cash out wallet	43
Table 15 - View promotion detail	45
Table 16 - Change Password.....	47
Table 17 - Edit account	50
Table 18 - Create order	52
Table 19 - verify services	54
Table 20 - Complete task.....	56
Table 21 - Confirm to do order	59
Table 22 - Add account	62
Table 23 - Change status account	63
Table 24 - Edit order.....	66
Table 25 - View report	68
Table 26 - Add product	71
Table 27 - Update product.....	74
Table 28 - Add category.....	76
Table 29 - Update category	78
Table 30 - Add promotion.....	81
Table 31 - Delete promotion.....	83
Table 32 - Add service	85
Table 33 - Edit service.....	88
Table 34 - Notify to customer.....	91
Table 35 - Filter order timeout.....	92
Table 36 - Generate report	94
Table 37 - Suggest promotion	96
Table 38 - Filter hot service	97
Table 39 - Filter hot product	99
Table 40 - Remove an account	100
Table 41 - Data dictionary	103

Table 42 - Describes components.....	105
Table 43 - Describes components.....	106
Table 44 - Class Diagram Explanation	107
Table 45 - <Class diagram attribute> Employee.....	108
Table 46 - <Class diagram method> Employee	108
Table 47 - <Class diagram attribute> Appointment.....	108
Table 48 - <Class diagram method> Appointment	109
Table 49 - <Class diagram attribute> AppointmentService	109
Table 50 - <Class diagram method> AppointmentService	109
Table 51 - <Class diagram attribute> Car	110
Table 52 - <Class diagram method> Car	110
Table 53 - <Class diagram attribute> CarRegistration.....	110
Table 54 - <Class diagram method> CarRegistration	110
Table 55 - <Class diagram attribute> Category.....	110
Table 56 - <Class diagram method> Category	111
Table 57 - <Class diagram attribute> Garage.....	111
Table 58 - <Class diagram method > Garage	111
Table 59 - <Class diagram attribute> Image	111
Table 60 - <Class diagram method> Image.....	111
Table 61 - <Class diagram attribute> Notification.....	112
Table 62 - <Class diagram method> Notification	112
Table 63 - <Class diagram attribute> Order.....	112
Table 64 - <Class diagram method> Order	112
Table 65 - <Class diagram attribute> OrderProduct.....	113
Table 66 - <Class diagram method> OrderProduct.....	113
Table 67 - <Class diagram attribute> Product	113
Table 68 - <Class diagram method> Product	113
Table 69 - <Class diagram attribute> Promotion	114
Table 70 - <Class diagram method> Promotion	114
Table 71 - <Class diagram attribute> Promotion Combo.....	114
Table 72 - <Class diagram method> PromotionCombo	115
Table 73 - <Class diagram attribute> Promotion role.....	115
Table 74 - <Class diagram method> Promotion role	115
Table 75 - <Class diagram attribute> Promotion type	115
Table 76 - <Class diagram method> Promotion type	115
Table 77 - <Class diagram attribute> Role.....	115
Table 78 - <Class diagram method> Role	116
Table 79 - <Class diagram attribute> Service	116
Table 80 - <Class diagram method> Service	116
Table 81 - <Class diagram attribute> StatusAppoinment.....	116
Table 82 - <Class diagram method> StatusAppointment.....	117
Table 83 - <Class diagram attribute> StatusOrder	117

Table 84 - <Class diagram method> StatusOrder.....	117
Table 85 - <Class diagram attribute> Transaction	117
Table 86 - <Class diagram method> Transaction.....	117
Table 87 - <Class diagram attribute> TransactionType	118
Table 88 - <Class diagram method> TransactionType.....	118
Table 89 - <Fields> Login.....	132
Table 90 - <Button> Login.....	132
Table 91 - <Button> Menu admin.....	133
Table 92 - <Fields> Manage product.....	134
Table 93 - <Button> Manage product	134
Table 94 - <Fields> Manage service.....	135
Table 95 - <Button> Manage service.....	135
Table 96 - <Fields> Manage promotion	136
Table 97 - <Button> Manage promotion.....	137
Table 98 - <Fields> Manage category.....	138
Table 99 - <Button> Manage category.....	138
Table 100 - <Fields> Manage order	139
Table 101 - <Button> Manage order	140
Table 102 - <Fields> Manage account.....	141
Table 103 - <Button> Manage account.....	141
Table 104 - <Fields> View report.....	142
Table 105 - <Button> View report	142
Table 106 - <Fields> View rating report.....	143
Table 107 - <Button> View rating report	144
Table 108 - <Fields> Login	144
Table 109 - <Button> Login	145
Table 110 - <Button> Home screen	146
Table 111 - <Button> List category	147
Table 112 - <Fields> List of promotion	148
Table 113 - <Button> List of promotion	148
Table 114 - <Fields> Promotion detail	149
Table 115 - <Button> Promotion detail	149
Table 116 - <Fields> Cart.....	151
Table 117 - <Button> Cart	151
Table 118 - <Fields> Book time.....	152
Table 119 - <Button> Book time	152
Table 120 - <Fields> Customer.....	153
Table 121 - <Button> Customer	154
Table 122 - <Entity> Dictionary.....	155
Table 123 - <Entity> Description.....	158
Table 124 - <Physical> Data dictionary	160
Table 125 - <Test case> Buy product.....	163

Table 126 - <Test case> Book Service.....	164
Table 127 - <Test case> Suggest promotion	165
Table 128 - <Test case> Register card	165
Table 129 - <Test case> Withdraw Money.....	166
Table 130 - <Test case> Recharge money	167
Table 131 - <Test case> Schedule Maintain	168
Table 132 - <Test case> Upgrade account.....	170
Table 133 - <Test case> Notification.....	171
Table 134 - <Test case> Register car.....	172
Table 135 - <Test case> Recognize voice	174
Table 136 - Hardware requirement.....	175
Table 137 - Software requirement.....	175
Table 138 - Add product steps.....	178
Table 139 - Edit product steps.....	179
Table 140 - Search product steps	179
Table 141 - Disable product steps.....	180
Table 142 - Add the product by excel steps.....	181
Table 143 - Add service steps	182
Table 144 - Edit service steps	183
Table 145 - Search service steps	184
Table 146 - Add promotion steps	186
Table 147 - Notify promotion steps.....	186
Table 148 - Delete promotion steps	187
Table 149 - Add category steps	188
Table 150 - Disable category steps.....	188
Table 151 - Edit garage steps	189
Table 152 - Add account steps.....	191
Table 153 - Disable account steps.....	191
Table 154 - Add order steps	192
Table 155 - Edit order steps	193
Table 156 - Cancel order steps	194
Table 157 - Create order steps for customer steps.....	199
Table 158 - Buy product steps	201
Table 159 - Confirm service steps	203

List Image

Figure 1 - Waterfall model.....	21
Figure 2 - System overview use case.....	25
Figure 3 - Guest overview use case	26
Figure 4 - Sign in use case.....	26
Figure 5 - Sign up use case	29

Figure 6 - <Authorized user> Overview use case	31
Figure 7 - Register a car.....	31
Figure 8 - Edit a car	34
Figure 9 - Remove a car	36
Figure 10 - Register bank card.....	37
Figure 11 - Cash in wallet	40
Figure 12 - Cash out wallet.....	42
Figure 13 - View promotion.....	44
Figure 14 - Change password.....	46
Figure 15 - Edit account	48
Figure 16 - Create order	50
Figure 17 - <Technical> Overview use case	52
Figure 18 – Verify service order.....	53
Figure 19 - Complete task.....	55
Figure 20 - <Sales clerk> Overview use case	57
Figure 21 - Confirm to do order	57
Figure 22 - Add account.....	59
Figure 23 - Change status account.....	62
Figure 24 - Edit order	64
Figure 25 - View reports	66
Figure 26 - <Admin> Over view use case	69
Figure 27 - Add product	69
Figure 28 - Update product.....	72
Figure 29 - Add category.....	74
Figure 30 - Update category.....	76
Figure 31 - Add promotion.....	79
Figure 32 - Delete promotion	81
Figure 33 - Add service	83
Figure 34 - Edit service	86
Figure 35 - <System> Over view	89
Figure 36 - Notify to customer.....	89
Figure 37 - Filter order timeout	91
Figure 38 – Collect data	93
Figure 39 - Suggest promotion	94
Figure 40 - Filter hot service	96
Figure 41 - Filter hot product.....	97
Figure 42 - Remove an account	99
Figure 43 - Conceptual Diagram.....	102
Figure 44 - Web application architecture description	104
Figure 45 - Component Diagram.....	104
Figure 46 - Component Diagram.....	105
Figure 47 - Class diagram	106

Figure 48 - <Sequence Diagram> Login	118
Figure 49 - <Activity Diagram> Login	119
Figure 50 - <Sequence Diagram> Add new order	120
Figure 51 - <Sequence Diagram> Update order	120
Figure 52 - <Sequence Diagram> Add new category	121
Figure 53 - <Sequence Diagram> Update category	121
Figure 54 - <Sequence Diagram> Add new service	122
Figure 55 - <Sequence Diagram> Update service	122
Figure 56 - <Sequence Diagram> Add new product	123
Figure 57 - <Sequence Diagram> Update product	123
Figure 58 -- <Sequence Diagram> add new account	124
Figure 59 - <Sequence Diagram> Add new promotion	124
Figure 60 - <Sequence Diagram> Filter hot service	125
Figure 61 - <Sequence Diagram> Filter hot product	125
Figure 62 - <Sequence Diagram> Notify customer	126
Figure 63 - <Sequence Diagram> Filter order timeout	126
Figure 64 - <Activity Diagram> Register card	127
Figure 65 - <Activity Diagram> Payment order	128
Figure 66 - <Activity Diagram> Book service	129
Figure 67 - <Activity diagram> Buy product	130
Figure 68 - <Activity Diagram> Register account	130
Figure 69 - <Activity Diagram> Edit account	131
Figure 70 - <Interface> Login	131
Figure 71 - <Interface> Menu admin	132
Figure 72 - <Interface> Manage product	133
Figure 73 - <Interface> Manage service	134
Figure 74 - <Interface> Manage promotion	136
Figure 75 - <Interface> Manage category	137
Figure 76 - <Interface> Manage order	138
Figure 77 - <Interface> Manage account	140
Figure 78 - <Interface> View report	142
Figure 79 - <Interface> View rating report	143
Figure 80 - <Mobile> Login	144
Figure 81 - <Mobile> Home screen	145
Figure 82 - <Mobile> List category of service	146
Figure 83 - <Mobile> List of promotion	147
Figure 84 - <Mobile> Promotion detail	148
Figure 85 - <Mobile> Cart	150
Figure 86 - <Mobile> Book time	151
Figure 87 - <Mobile> Customer	153
Figure 88 - ERD - Entity Relationship Diagram	154
Figure 89 - Physical Diagram	159

Figure 90 - Add product	177
Figure 91 - Add product detail.....	177
Figure 92 - Edit product	178
Figure 93 - Edit product detail.....	178
Figure 94 - Search product detail	179
Figure 95 - Disable product.....	180
Figure 96 - Add the product by excel	180
Figure 97 - Add service	181
Figure 98 - Add service detail	181
Figure 99 - Edit service	182
Figure 100 - Edit service detail.....	183
Figure 101 - Search service.....	184
Figure 102 - Add promotion	184
Figure 103 - Add promotion detail.....	185
Figure 104 - Notify promotion.....	186
Figure 105 - Delete promotion	186
Figure 106 - Add category	187
Figure 107 - Add category detail.....	187
Figure 108 - Disable category	188
Figure 109 - Edit garage	189
Figure 110 - Add account.....	190
Figure 111 - Add account detail	190
Figure 112 - Disable account.....	191
Figure 113 - Add order	192
Figure 114 - Add order detail.....	192
Figure 115 - Edit order	193
Figure 116 - Edit order detail.....	193
Figure 117 - Cancel order	194
Figure 118 - Create order have service	195
Figure 119 - Create order have service	195
Figure 120 - Create order have service	196
Figure 121 - Create order have service	196
Figure 122 - Create order have service	196
Figure 123 - Create order have service	197
Figure 124 - Create order have service	197
Figure 125 - Create order have service	198
Figure 126 - Buy product.....	199
Figure 127 - Buy product.....	200
Figure 128 - Buy product.....	201
Figure 129 - Confirm service	202
Figure 130 - Confirm service	202

A.Introduction

1. Project Information

- Project name: **Rheinol Quick Service**
- Project Code: **RQS**
- Product Type: **Website Application and Mobile Application**
- Start Date: **May 13rd, 2019**
- End Date: **August 20th, 2019**

2. Introduction

Swd is a medium-sized independent company, appearing from former FINA Schmierstoffwerk Duisburg. Swd has been developing, manufacturing and distributing high-quality lubricants and specialty for over 50 years. The Swd-Rheinol® brand has existed since 1998. Do not stop there, they are trying to extend business marketing by providing services, maintenance, repair and so on.

With the development of the technology revolution, we introduce two tools: first is a website administrator system for managing the business and the second is a mobile application. Customers can book any services by touching on their smartphone, hence implicitly making convenient and proactive.

3. Current Situation

When customers visit the Rheinol's website, they can only see some product's information or introduction to the company. Customers can't buy products or book services through the website. Moreover, they do not know the available time of the garage to reserve. And with the current system, customers are not informed about the time to return to maintain replacement or used equipment or services.

4. Problem Definition

Below are the disadvantages of Rheinol website:

- The website just displays Rheinol's information and products.
- Customers cannot buy products or use Rheinol's services through the website.
- The system has no control over the services or products they are trading.
- The system does not manage loyal customers to offer promotions.

5. Proposed Solution

We provide a solution about building web and mobile application to manage Rheinol quick service. The purpose of Rheinol is selling their products and services regarding luxury cars such as selling accessories, interior decoration, outside decoration, quickly maintenance and basic services. The system should be the following features:

5.1 Main features

- Buying and selling accessories.
- Supply repairing, maintenance.
- Launch new products, services, and promotions for customers.
- Notify a maintenance schedule for a customer's car.
- Statistic report regarding all of the services.
- Customer can book services base on the available slot.

5.2 Advantages and disadvantages

5.2.1 Advantages

- Help customer improve user experiment in using Rheinol quick service.
- Admin manages product, service, and the system more easily.
- Resolve the issue of warranty scheduling for Rheinol.
- Optimize time for the amount of space in the garage
- Manage customer accounts and offer specific promotions exactly.

5.2.2 Disadvantages

- Only support the Vietnamese language for web and mobile app.
- Not ready to support the franchise.

6. Functional Requirement

Function requirements of the system are listed below:

6.1 Buying products and accessories:

- Customer must be authorized by logging in.
- The customer could buy multiple products or accessories.
- The customer could view all products and accessories if they are available.
- The customer could add products and accessories into the cart.
- The customer could pay the order by using the wallet in the application as long as the amount in the wallet is greater than or equal to the amount in the bill.
- If payment is successful, the system shall save purchase history into a customer's account.

6.2 Booking the services:

- Customer must be authorized by logging in.
- The customer could book multiple services.
- The customer could view all services if they are available.
- The customer could add services into the cart.
- The customer could select a time that they want to use services.
- If the customer selects an inappropriate time, the system shall notify.

6.3 Launch new product, service and promotion:

- Admin could add new product and service.
- Admin could create a new promotion for the customers.

6.4 Notifying maintenance schedule:

- The system shall inform customers on the mobile application about the warranty schedule for the services used before 7 days.

6.5 Statistic report regarding all of the services:

- The system shall generate a report about revenue services are used.
- Director could find report follow week, month, year.

7. Role and Responsibility

No	Full Name	Role	Position	Contact
1	Nguyen Thi Cam Huong	Project Manager	Supervisor	huongntc2@fe.edu.vn
2	Le Dinh Thien Vu	Developer	Leader	vuldtse62590@fpt.edu.vn
3	Phan Anh Phuc	Developer	Member	phucpase62543@fpt.edu.vn
4	Dang Huu Phuong Nam	Developer	Member	namdhpse62167@fpt.edu.vn

Table 1 - Role and responsibility

B. Software Project Management Plan

1. Problem Definition

Name of this Capstone Project

- **Official name:** Rheinol Quick Service
- **Vietnamese name:** Hệ thống dịch vụ Rheinol

- **Abbreviation:** RQS

1.1 Problem Abstract

The project is a concern about providing a website for Admin that can manage its systems such as products, services, user members and promotions. Also, the director can see sales and membership reports at time intervals. In addition, the project provides customers with a mobile application that can buy products or reserve time to use the service.

Rheinol's company uses only one website as the main channel to provide information about lubricant products or introduce the company. With the current website, customers cannot directly buy products or use services from companies on the internet. Therefore, causing many difficulties for businesses and customers.

The system will be a useful help for businesses as well as customers. It will help the interactive functions between the two sides convenient, easy and intuitive.

1.2 Project Overview

1.2.1 Current situation

Below are the problems encountered in this project:

- Limit time and human resource: Team has only Modeling member and time for all project is about 13 weeks for writing a document, implementing the products and testing.
- Schedule of team member: team members can have a conflict in meeting schedule because of sick, or class and work schedule, etc.
- Framework study: team members have a problem when applying the play framework into the project. The team needs an amount of time to get familiar with new techniques.
- New technique: Some team members are new to the techniques used in the project. The team needs an amount of time to get familiar with those techniques.
- Lack of UI (user interface), UX (user experience) design skill: Our team members all study IS major, and no one has studied UI, UX design. Therefore, some UI may misunderstand or hard to use with normal users.

1.2.2 The proposed system

Building a system where the customer could buy products or accessories from the trusted provider. On the other hand, the customer also could book car services from the mobile application. Moreover, they could set when to start the service, but if any trouble like the garage full slots happens, the system shall suggest time.

1.2.3 Boundaries of the system

The system can:

- Allow admin to manage member accounts, products, services, promotion.
- Allow the sales clerk to manage orders request from customers.
- Allow the director to view the report.
- Allow customer to buy products or accessories, book services.
- Allow customer can pay orders by the Rheinol's wallet.

The system cannot:

- The system cannot manage the human resource.
- The system cannot manage devices.
- The system cannot manage inventory.

1.2.4 Future plans

- Develop the franchise system.
- Apply Elastic search
- Automatically move data was disabled to another storage
- Apply schedule algorithms

1.2.5 Development environment

1.2.5.1 Hardware requirements

For server

Hardware	Minimum Requirement	Recommended
Internet Connection	Cable, Wi-Fi (8 Mbps)	Cable, Wi-Fi (50 Mbps or more)
Operation System	XP, Vista, 7, 10, Window Server 2008	10, Window server 2008
Computer Processor	Intel® Xeon ® 1.4GHz	Intel® Xeon ® Quad Core (12M Cache, 2.50 GHz)
Computer Memory	4GB RAM	32 GB RAM or more
Storage Space	1GB	5GB or more

Table 2 - Hardware requirements for server

For PC

PC	Minimum Requirements	Recommended
Internet Connection	Cable, Wi-Fi (4 Mbps)	Cable, Wi-Fi (8 Mbps)
Operating System	Window 7	Window 7 or more.

Computer Processor	Intel® Core i3 1.4GHz	Intel® Core i5 2.50GHz
Computer Memory	1GB RAM	2GB RAM or more
Web Browser	Chromes (v42 or higher)	Chrome latest stable version

Table 3 - Hardware requirement for PC

Software requirement

Software	Name / Version	Description
Operation System	Windows Server 2014	Operating system and platform for development
Environment	Java EE 8.0, Node v10, NPM v6	Specification for developing web application
Modeling tool	StarUML	Used to design diagram
IDE	IntelliJ IDEA 2019.1.2, Visual Studio Code 1.35.1	Programming tools
DBMS	MySQL 5.7	Used to create & manage the database for system
Source control	Git on IDE (Git lab)	Used for source control
Web browser	Chrome 69 or above	Testing browser

Table 4 - Software requirement

2. Project Organization

2.1 Software Process Model

This project is developed by using the waterfall model. We apply customized waterfall model to capable with the current situation in our team and our customer. We choose this model because of the following reasons:

- Based on the reality surveys, research, software Lucky Gara, the requirements of this project are stable, clear, fixed and well understood by all members.
- This project does not have anything to change the requirement from the customer.
- We do not meet the customer regularly.

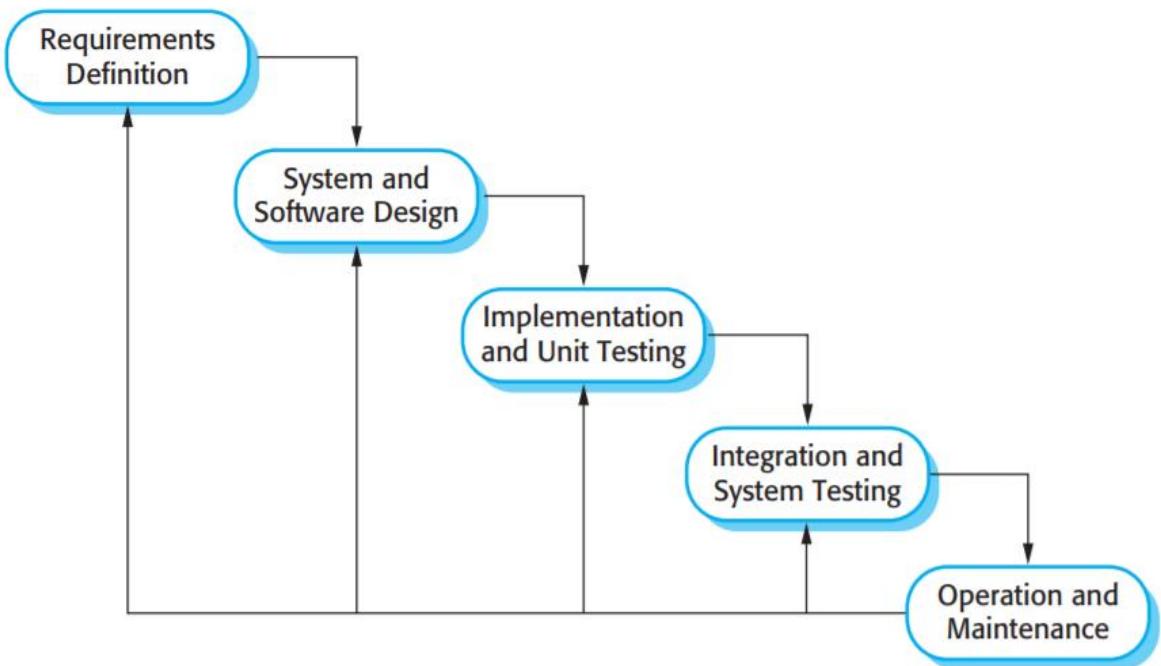


Figure 1 - Waterfall model

2.2 Roles and responsibilities

No	Full name	Role in Group	Responsibilities
1	Nguyễn Thị Cẩm Hương	Project manager, BA	Specify user requirement Control the development process Give out technique and business analysis support
2	Lê Đình Thiện Vũ	Team leader, DEV, BA, Tester	Managing process Designing software architect Designing database Clarifying requirements Prepare documents GUI Design Create a test plan Coding Testing Arrange Meeting Risk Management Designing database

3	Phan Ánh Phúc	DEV, BA, Tester	Designing database Clarifying requirements Prepare documents GUI Design Creating a test plan Coding Testing
4	Đặng Hữu Phương Nam	DEV, BA, Tester	Designing database Clarifying requirements Prepare documents GUI Design Creating a test plan Coding Testing

Table 5 - Role and responsibilities

2.3 Tools and Techniques

Tool/Technique	Name
Front-end	HTML5, CSS3, JavaScript, Bootstrap 4, ReactJS
Back-end	JavaEE, Spring Boot 2 framework, JPA, Hibernate
Mobile	Expo(react native)
DBMS	MySQL
Source Control	GitLab
Modeling tool	StarUML
IDE	IntelliJ IDEA

Table 6 - Tools and techniques

3. Project Management Plan

3.1 Requirement

In attached files

3.2 Mockup UI

In attached files

3.3 All meeting minutes

In attached files

4. Coding Convention

Summary:

- **Naming Convention:**

- Name of variables, properties of the tag should be applied camel notation.
- Variable, classes, component names should be short and meaningful.
- Methods should be verbs.
- Name of class and component should be uppercase for the first letter
- **Indentation:**
 - One declaration per line is recommended since it encourages commenting.
 - In absolutely no case should variables and functions be declared on the same line
- **Declarations Convention:**
 - One declaration per line is recommended since it encourages commenting
 - Using Java Code Convention from
<http://www.oracle.com/technetwork/java/codeconvtoc-136057.html>

C. Software Requirement Specification

1. User Requirement Specification

1.1 Guest Requirement

A guest is a person who does not have access to the system. A Guest can use some functions in the system. To use all functions, guest must log in. These are functions guest can use:

- Register.
- Login.
- View the list of products and accessories.
- View the list book services.

1.2 Member Requirement

A member is an authorized user how has permission to login into the system. There are some functions a member can use:

- Buy products or accessories.
- Get discounts
- Earn better discounts through member level system.
- Register car information.

1.3 Car Owner Requirement

A car owner is an authorized user and had been registered for the car information. There are some functions a car owner can use:

- Book services from Rheinol mobile application.
- Setup schedule timing by their self.
- Receive notification that reminds maintenance service

1.4 Sales clerk Requirement

A sales clerk is an authorized user that belongs to Rheinol organization:

- Control orders request from customers.
- Create an order if necessary.
- Create a customer account.

1.5 Director Requirement

A director is an authorized user. A director can view all of the reports

2. System Requirement Specification

2.1 External Interface Requirement

2.1.1 User Interface

- GUI should be simple, clear, intuitive and reminiscent.
- The interface design is an iterate process includes design, sketching, prototyping, user assessment.
- The system was developed uses language is Vietnamese.

2.1.2 Hardware Interface

- Desktop PC.

2.1.3 Software Interface

- Web application: any web browser that supports HTML5 & CSS3

2.1.4 Communication Protocol

- Use HTTP protocol 1.1 for communication between web application and web server

2.2 System Overview Use Case

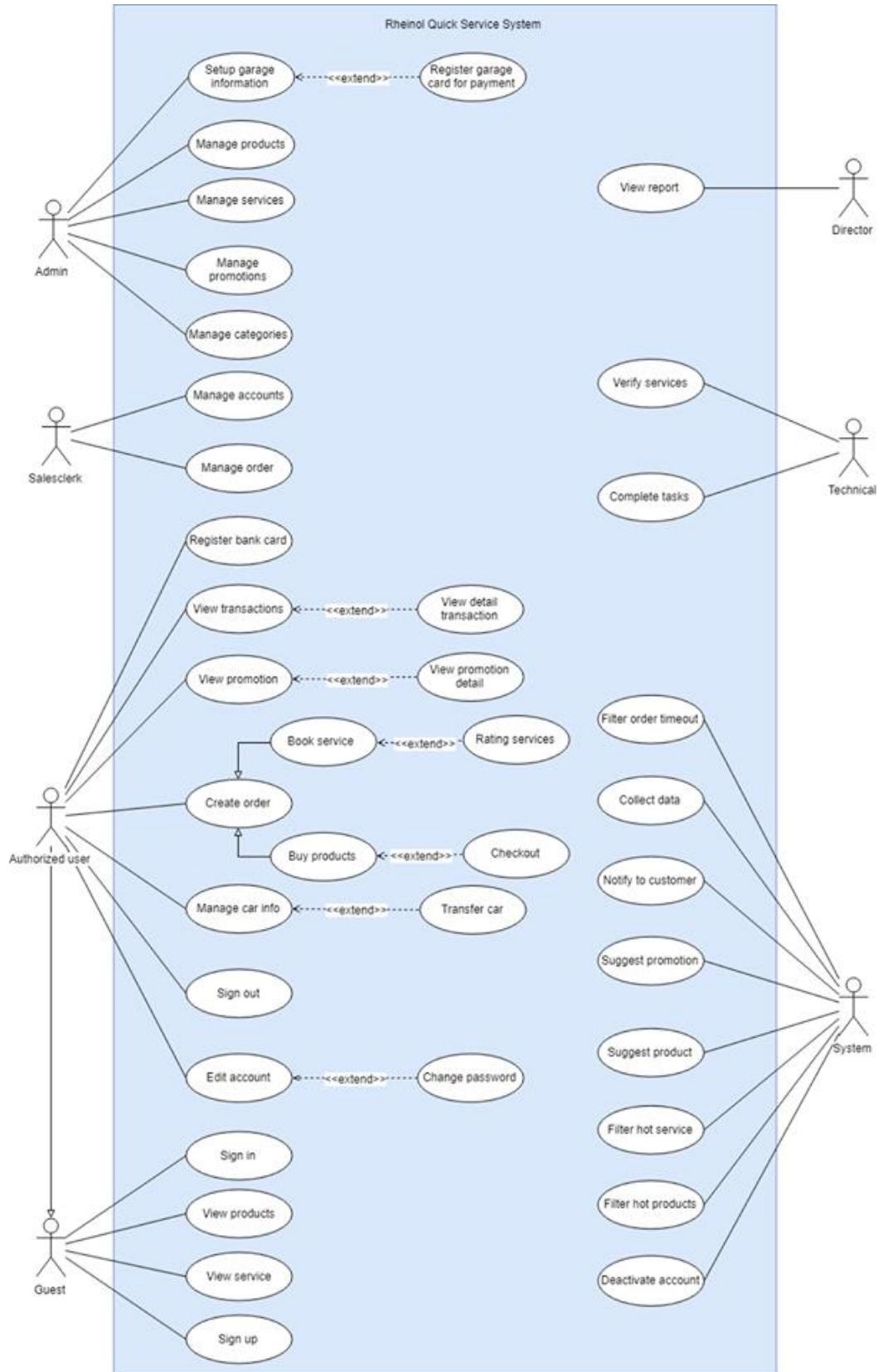


Figure 2 - System overview use case

2.3 List of use case

2.3.1 <Guest> Overview Use Case

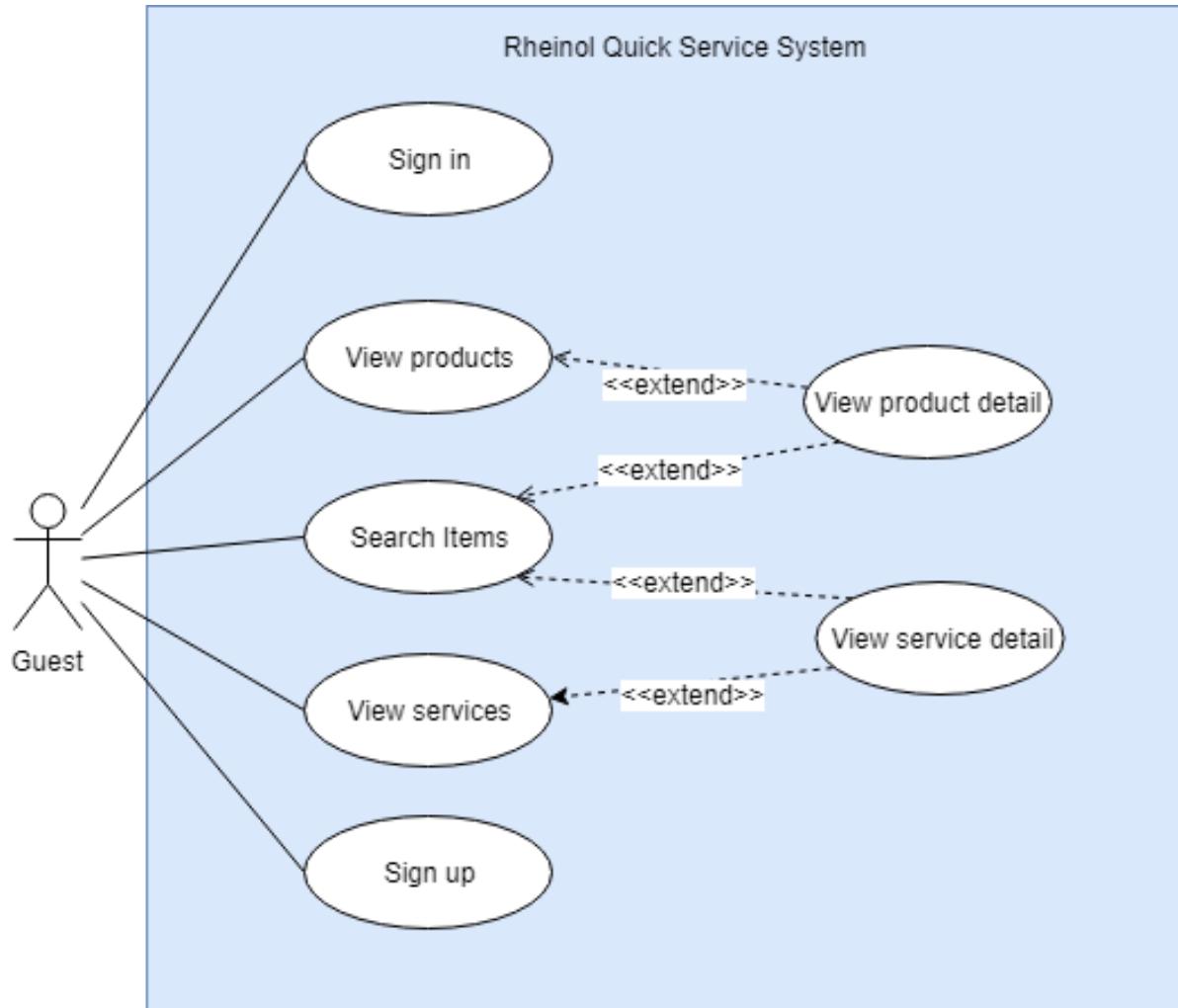


Figure 3 - Guest overview use case

2.3.1.1 Sign in

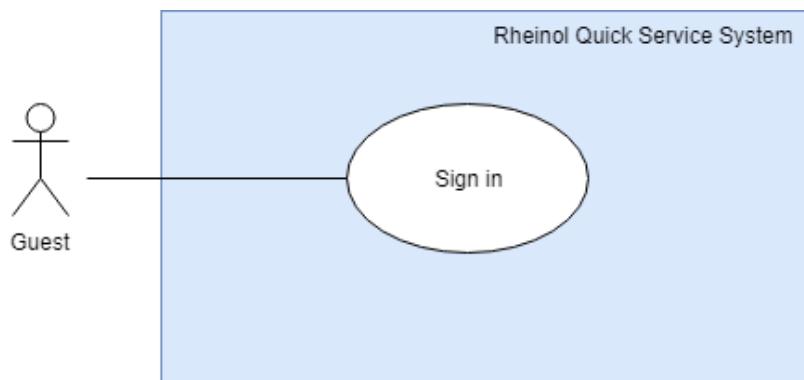


Figure 4 - Sign in use case

USE CASE - RQS_UC_1			
Use Case No.	RQS_UC_1	Use Case Version	2.0
Use Case Name	Sign in		
Author	NamDHP		
Date	02/06/2019	Priority	High

Actor:

Guest

Summary:

This use case allows guest could sign in the system.

Goal:

Guest could sign in the system with a specific role.

Triggers:

The guest wants to sign in the system.

Preconditions:

Guest signed up the account before.

Postconditions:

Success: the guest is authorized with a specific role in the system.

Fail: the guest cannot sign in the system.

Main Success Scenario:

Step	Actor Action	System Response
1	The guest goes to “Sign in” view.	The system requires information from a guest: Username: input text type. Password: input text type.

2	The guest input information							
3.1	The guest sends the command to sign in the system.	The system will display an error message if the information does not match. [Exception 1]						
3.2		The system will show a successful message and redirect to "User" view..						
Exceptions:								
<table border="1"> <thead> <tr> <th>No</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>The guest sends the command to sign in the system.</td><td>The system shows an error message to ask user input correct data.</td></tr> </tbody> </table>			No	Actor Action	System Response	1	The guest sends the command to sign in the system.	The system shows an error message to ask user input correct data.
No	Actor Action	System Response						
1	The guest sends the command to sign in the system.	The system shows an error message to ask user input correct data.						
Business rules:								
<p>Password is encrypted before being sent to the server.</p> <p>After login to the system, the guest will be redirected to a specific view based on their role.</p> <p>On the mobile application system: an authorized user.</p> <ul style="list-style-type: none"> ▪ The role is "Authorized user", the system will display to the home view. <p>On the web system: a director, a sales clerk, an admin.</p> <ul style="list-style-type: none"> ▪ If the role is "Director", the system will display to Director view. ▪ If the role is "Sales clerk", the system will display to Sales clerk Dashboard view. ▪ If the role is "Admin", the system will display to Admin Dashboard view. 								

Table 7 - Sign in use case

2.3.1.2 Sign up

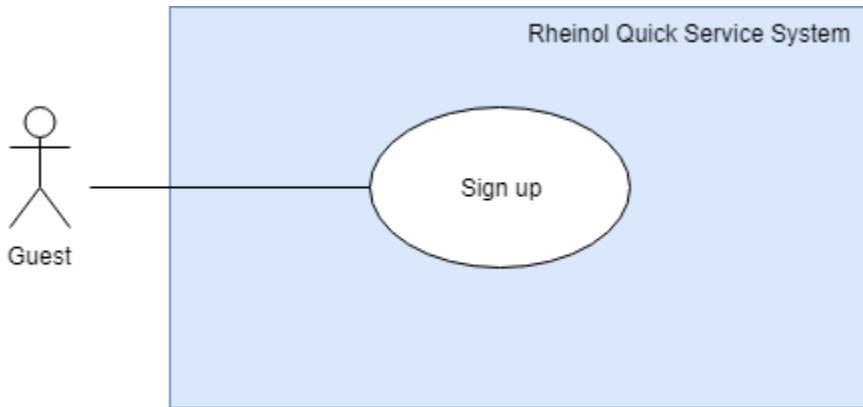


Figure 5 - Sign up use case

USE CASE – RQS_UC_2			
Use Case No.	RQS_UC_2	Use Case Version	2.0
Use Case Name	Sign up		
Author	PhucPA		
Date	02/06/2019	Priority	Normal
Actor:			
Guest			
Summary:			
This use case allows guest could sign up an account.			
Goal:			
Guest could sign up an account successfully, and the new account is added into the system.			
Triggers:			
The guest wants to sign up an account.			
Preconditions:			
N/A			
Postconditions:			

Success: a new account is saved in the system.

Fail: a new account is not added to the system and shows a notification.

Main Success Scenario:

Step	Actor Action	System Response
1	The guest goes to "Sign up" view.	The system requires information from a guest : Username: input text type, the length is greater than 6 digits. Password: input text type. Retype password: input text type must be the same as password.
2	The guest input information	
3.1	The guest sends the command to sign in the system.	The system will display an error message if the information does not match. [Exception 1]
3.2		Guest logsins system with his specific role.

Exceptions:

No	Actor Action	System Response
1	The guest sends the command to sign in the system. the account's information.	The system shows an error message to ask user input missing required fields or condition fields.

Business rules:

Password is encrypted before being sent to the server.

Table 8 - Sign Up

2.3.2 <Authorized user> Overview use case



Figure 6 - <Authorized user> Overview use case

2.3.2.1 Register a car

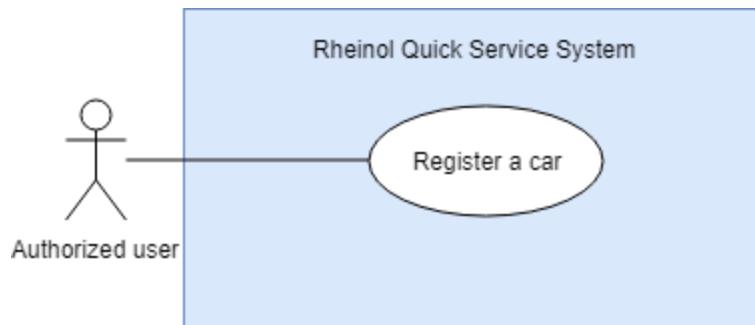


Figure 7 - Register a car

USE CASE – RQS_UC_3			
Use Case No.	RQS_UC_3	Use Case Version	2.0
Use Case Name	Register a Car		
Author	VuLDT		
Date	22/05/2019	Priority	Normal
<p>Actor: Authorized user</p> <p>Summary: This use case allows authorized user could register car information.</p> <p>Goal: When the authorized user added car successfully, this car is saved in System.</p> <p>Triggers: There is a new car that needs to be added to the system.</p> <p>Preconditions: The guest must log in into the mobile application with the role authorized user.</p> <p>Postconditions: Success: a car is saved in the system. Fail: a car is not added to the System and shows a notification.</p> <p>Main Success Scenario:</p>			
Step	Actor Action	System Response	
1	The authorized user goes to new car view.	Car information screen display fields data for user input: License plates: textbox, required not null Capacity: textbox, required not null Name: textbox, required.	

		Manufacture: textbox, required. Category: combo box.
2	The authorized user inputs information	
3.1	The authorized user sends a command to add a new car request.	The system validates information, the system will display a message if the information is not matched. [Exception 1,2]
3.2		The car is added to the database and system will show message successfully.
Exceptions:		
No	Actor Action	System Response
1	The authorized user sends a command to add a new car request.	The system shows an error message to ask user input missing required fields or condition fields.
Business rules: The car belongs to many authorized user accounts. Only authorized user can add new a car to the system. The new car is saved with QR code.		

Table 9 - Register a car

2.3.2.2 Edit a car

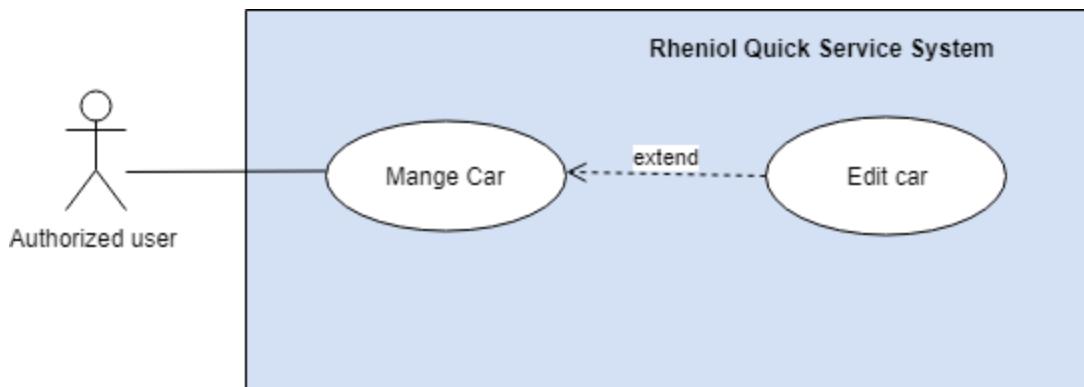


Figure 8 - Edit a car

USE CASE - RQS_UC_4			
Use Case No.	RQS_UC_4	Use Case Version	2.0
Use Case Name	Edit Car		
Author	VuLDT		
Date	22/05/2019	Priority	Normal
Actor:			
Authorized user			
Summary:			
This use case allows authorized user could edit car information.			
Goal:			
When authorized user edited car successfully, this car's information is updated in System.			
Triggers:			
There is a car that needs to be edited into System.			
Preconditions:			
The guest must log in into the mobile application with the role authorized user.			
Postconditions:			
Success: the new car's information is updated in the System.			
Fail: the car is not edited to the System and shows the notification.			

Main Success Scenario:		
Step	Actor Action	System Response
1	The authorized user goes to edit car view.	<p>Car information screen display fields data which information for user edit:</p> <p>License plates: textbox, required not null</p> <p>Capacity: textbox, required not null</p> <p>Name: textbox, required.</p> <p>Manufacture: textbox, required.</p> <p>Category: combo box.</p>
2	The authorized user edits information	
3.1	The authorized user sends a command to edit car request.	<p>The system validates information, the system will display a message if the information is not matched.</p> <p>[Exception 1]</p>
3.2		The car is edited to the database and system will show message successfully.
Exceptions:		
No	Actor Action	System Response
1	The authorized user sends a command to edit car request.	The system shows an error message to ask user input missing required fields or condition fields.
Business rules:		
N/A		

Table 10 - Edit car

2.3.2.3 Remove a car

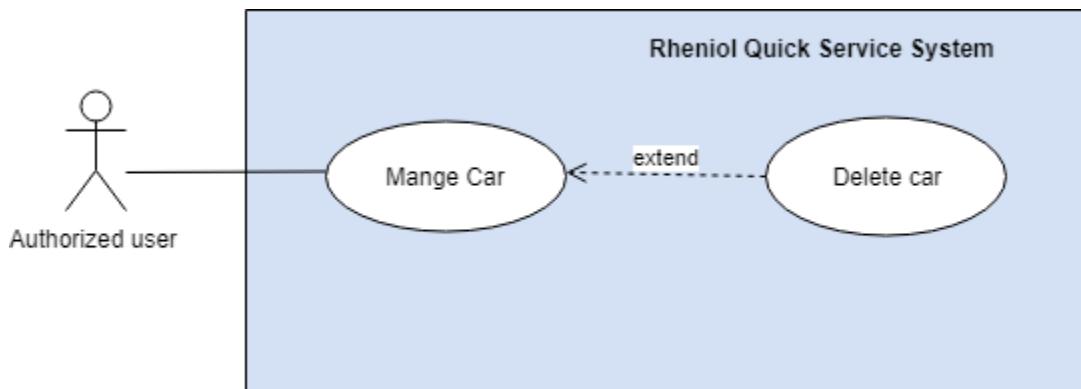


Figure 9 - Remove a car

USE CASE – RQS_UC_5			
Use Case No.	RQS_UC_5	Use Case Version	2.0
Use Case Name	Delete Car		
Author	VuLDT		
Date	22/05/2019	Priority	Normal
Actor:	Authorized user		
Summary:	This use case allows authorized user could delete a car.		
Goal:	When an authorized user deletes a car successfully, the car's status is disabled.		
Triggers:	There is a car that needs to be removed from the customer's list.		
Preconditions:	The guest must log in into the mobile application with the authorized user role.		
Postconditions:	Success: a car's status is disabled in the System. Fail: a car's status is not changed and shows notification message.		

Main Success Scenario:		
Step	Actor Action	System Response
1	The authorized user goes to list car view.	List car is displayed with the delete buttons.
2.1	The authorized user sends a command to delete car request.	The car's status is not changed and the system shows the notification.
2.2		The car's status is disabled and the system will show a message successfully.
Exceptions:		
No	Actor Action	System Response
1	The authorized user sends a command to delete car request.	The system shows an error message when a car's status cannot be changed.
Business rules:		
The only authorized user can delete the car.		

Table 11 - Delete car

2.3.2.4 Register bank card

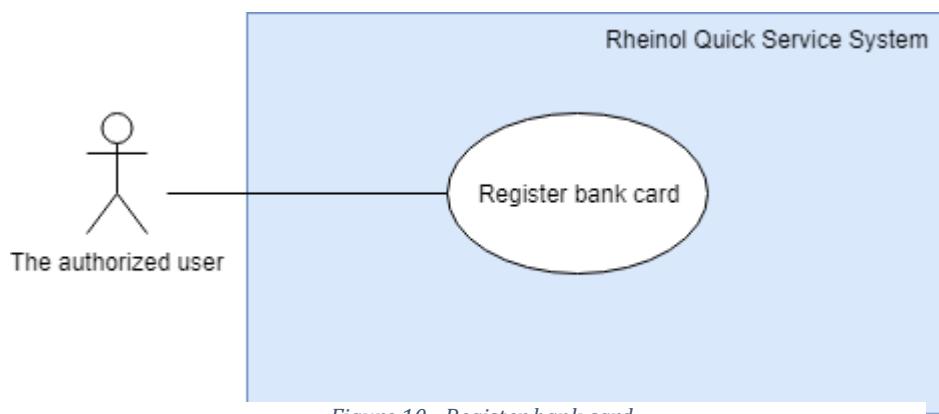


Figure 10 - Register bank card

USE CASE – RQS_UC_6			
Use Case No.	RQS_UC_6	Use Case Version	2.0
Use Case Name	Register bank card		
Author	VuLDT		
Date	02/06/2019	Priority	High
Actor:			
Authorized user			
Summary:			
This use case allows authorized user could register bank card in the mobile application.			
Goal:			
When the authorized user register bank card successfully, the authorized user can manipulate with wallet in the application.			
Triggers:			
The Authorized user wants to use the wallet in the application.			
Preconditions:			
Guest must log in into the mobile application with role authorized user.			
Postconditions:			
Success: the bank card is registered successfully.			
Fail: the bank card is not registered successfully.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	The authorized user goes to manage bank card view.	The system requires information: Card number: a sequence of digits with 16 numbers.	

		Expiry: input text, type number. CVC/CCV: input text, type number. Cardholder's name: textbox, length more than 6 characters.
2	The authorized user input information	The system validates the information.
3.1	The authorized user sends the command to register bank card.	The system will display a message if the information is not matched. [Exception 1]
3.2		The system will display a successful message if the bank card is valid.

Exceptions:

No	Actor Action	System Response
1	The authorized user sends the command to register bank card.	The system shows an error message to ask user input wrong information.

Business rules:

The authorized user could register only one bank card in application.

The bank card accepted by the system is the visa card and master card.

The bank card is managed by a third party.

Table 12 - Register bank card

2.3.2.5 Cash in wallet

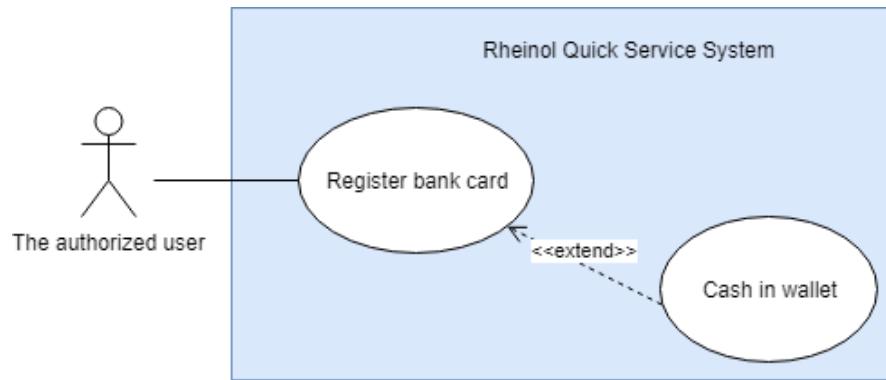


Figure 11 - Cash in wallet

USE CASE – RQS_UC_7			
Use Case No.	RQS_UC_7	Use Case Version	2.0
Use Case Name	Cash in wallet		
Author	VuLTD		
Date	02/06/2019	Priority	Normal
Actor:	Authorized user		
Summary:	This use case allows authorized user could cash in the mobile application's wallet.		
Goal:	When the authorized user cashes in successfully, balance in wallet increased.		
Triggers:	The Authorized user wants to cash in the application's wallet.		
Preconditions:	Guest must log in into the mobile application and there is bank card in application.		
Postconditions:	Success: the balance in the wallet increased.		

Fail: the balance in the wallet remains the same.

Main Success Scenario:

Step	Actor Action	System Response
1	The authorized user goes to “cash in wallet” view.	The system requires information: Amount of money: input type, type number.
2	The authorized user input information	
3.1	The authorized user sends the command to cash in the wallet.	The system will display a message if the information is not matched. [Exception 1]
3.2		The system will display a successful message if the cashing in the wallet is valid.

Exceptions:

No	Actor Action	System Response
1	The authorized user sends the command to cash in the wallet	The system shows an error message if the bank card has a problem.

Business rules:

The authorized user could cash in the wallet any time.

Table 13 - Cash in Wallet

2.3.2.6 Cash-out wallet

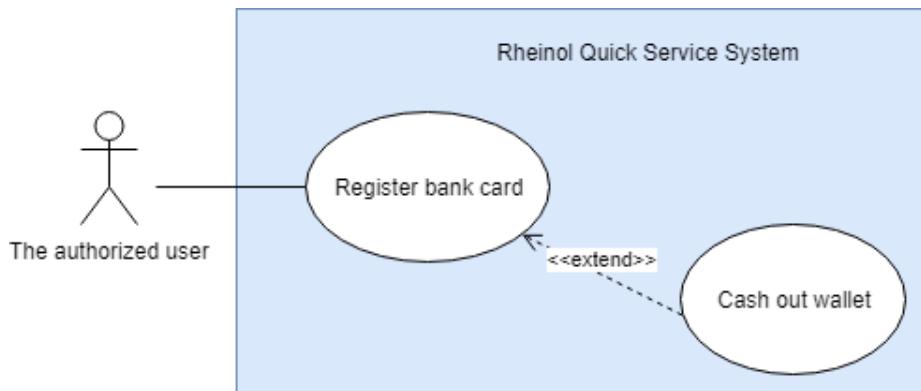


Figure 12 - Cash out wallet

USE CASE – RQS_UC_8			
Use Case No.	RQS_UC_8	Use Case Version	2.0
Use Case Name	Cash-out wallet		
Author	NamDHP		
Date	02/06/2019	Priority	Normal
Actor:	Authorized user		
Summary:	This use case allows authorized user could cash out the mobile application's wallet.		
Goal:	When the authorized user cashes out successfully, balance in wallet decreased.		
Triggers:	The Authorized user wants to cash out the application's wallet.		
Preconditions:	Guest must log in into the mobile application and the balance still has money.		
Postconditions:			

Success: the balance in the wallet decreased.

Fail: the balance in the wallet remains the same.

Main Success Scenario:

Step	Actor Action	System Response
1	The authorized user goes to “cash out wallet” view.	The system requires information: Amount of money: input type, type number.
2	The authorized user input information	
3.1	The authorized user sends the command to cash out the wallet.	The system will display a message if the information is not matched. [Exception 1]
3.2		The system will display a success message if the cashing out the wallet is valid.

Exceptions:

No	Actor Action	System Response
1	The authorized user sends the command to cash out the wallet	The system shows an error message if the bank card has a problem.

Business rules:

The authorized user could cash out the wallet any time.

Table 14 - Cash out wallet

2.3.2.7 View promotion detail

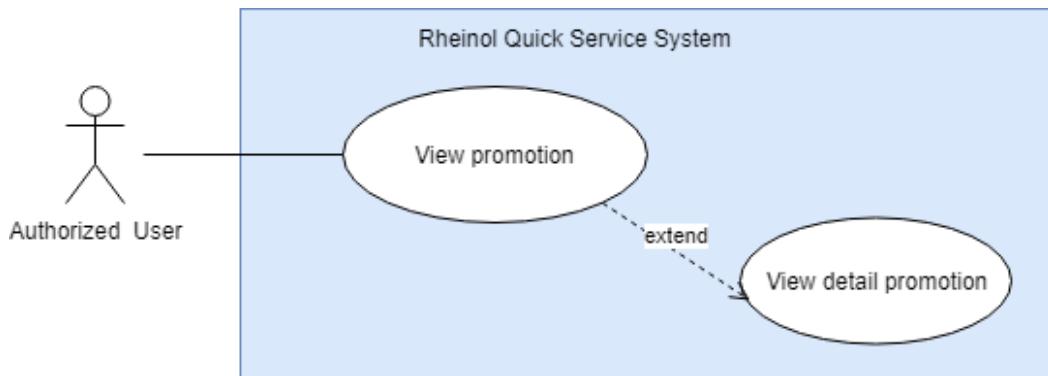


Figure 13 - View promotion

USE CASE – RQS_UC_9			
Use Case No.	RQS_UC_9	Use Case Version	2.0
Use Case Name	View promotion detail		
Author	NamDHP		
Date	02/06/2019	Priority	Normal
Actor:	Authorized user		
Summary:	This use case allows authorized user could view the promotions in the mobile.		
Goal:	When the authorized user could view the promotion.		
Triggers:	The Authorized user wants to view the promotion in the application.		
Preconditions:	Guest must log in into the mobile application with role authorized user.		
Postconditions:	Success: the authorized user views all promotion detail.		

Fail: the authorized user cannot view promotion.

Main Success Scenario:

Step	Actor Action	System Response
1	The authorized user goes to promotion view.	The system shows list promotion.
2	The authorized user selects one promotion to see detail.	The information about the promotion's detail.

Exceptions:

No	Actor Action	System Response
1	The authorized user selects one promotion to see detail.	The system cannot show information about the promotion's detail.

Business rules:

N/A

Table 15 - View promotion detail

2.3.2.8 Change Password

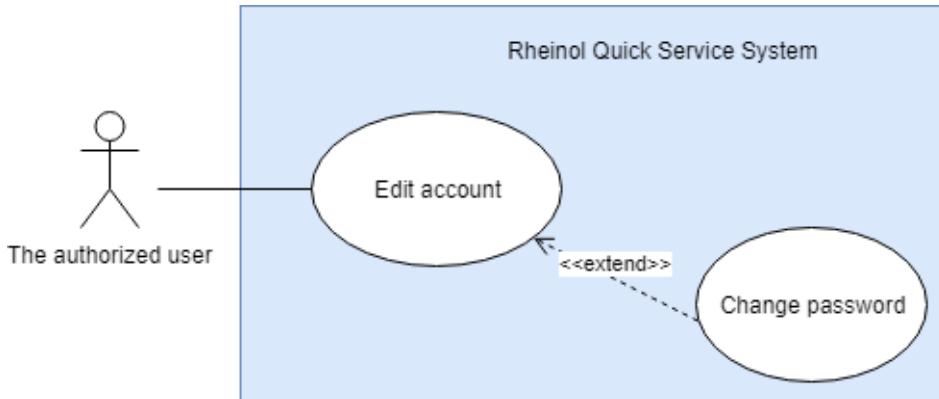


Figure 14 - Change password

USE CASE – RQS_UC_10			
Use Case No.	RQS_UC_10	Use Case Version	2.0
Use Case Name	Change password		
Author	VuLTD		
Date	02/06/2019	Priority	Normal
Actor:	Authorized user		
Summary:	This use case allows authorized user could change password.		
Goal:	When the authorized user changes password successfully, the system saves the new password.		
Triggers:	The Authorized user wants to change the password in the application.		
Preconditions:	Guest must log in into the mobile application.		
Postconditions:			

Success: the system saves the new password.

Fail: the password cannot reset.

Main Success Scenario:

Step	Actor Action	System Response
1	The authorized user goes to “update password” view.	The system requires information: The old password: input text type. The new password: input text type, a length is greater 6 digits. Re-type new password: input type text must be same the new password.
2	The authorized user input information	
3.1	The authorized user sends the command to change the password.	The system will display a message if the information is not matched. [Exception 1]
3.2		The system will return a “manage information account” view and change password.

Exceptions:

No	Actor Action	System Response
1	The authorized user sends the command to change the password	The system shows an error message if the old password does not match.

Business rules:

The account is registered with Facebook that cannot change password.

Table 16 - Change Password

2.3.2.9 Edit an account

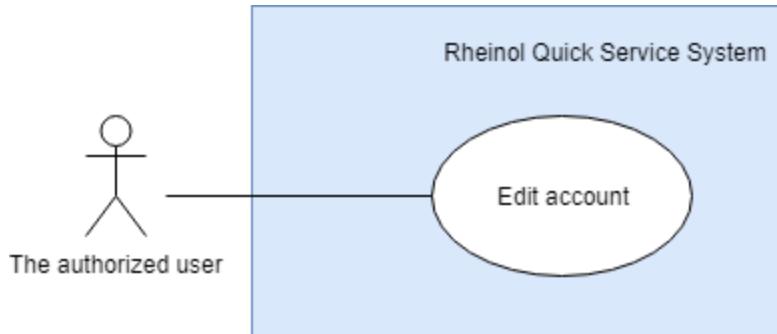


Figure 15 - Edit account

USE CASE – RQS_UC_11			
Use Case No.	RQS_UC_11	Use Case Version	2.0
Use Case Name	Edit the account		
Author	NamDHP		
Date	02/06/2019	Priority	Normal
Actor: Authorized user			
Summary: This use case allows authorized user could edit the account.			
Goal: When the authorized user edits the account successfully, the system saves the new account's information.			
Triggers: The Authorized user wants to edit the account's information.			
Preconditions: Guest must log in into the mobile application.			
Postconditions:			

Success: the system saves the new information's account.

Fail: the new information's account cannot save in the system.

Main Success Scenario:

Step	Actor Action	System Response
1	The authorized user goes to "manage account" view.	<p>The system requires information:</p> <p>Account's name: input text type, the length is greater than 6 digits.</p> <p>Account's date of birth: input date type.</p> <p>Account's email: input text type, follow email format.</p> <p>Account's address: input text type, the length is greater than 6 digits.</p> <p>Account's phone: input text type, follow phone format.</p>
2	The authorized user input information	
3	The authorized user sends the command to edit the account's information.	The system will display a message if the information is not matched. [Exception 1]
		The system will return a "manage information account" view and update the new information.

Exceptions:

No	Actor Action	System Response
1	The authorized user sends the command to edit	The system shows an error message to ask user input missing required fields or condition fields.

	the account's information.	
Business rules: The authorized user cannot edit the account's username.		

Table 17 - Edit account

2.3.2.10 Create the order

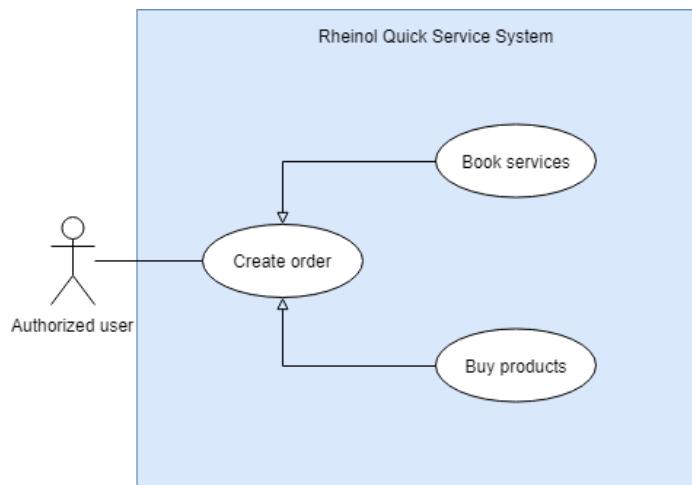


Figure 16 - Create order

USE CASE – RQS_UC_12			
Use Case No.	RQS_UC_12	Use Case Version	2.0
Use Case Name	Create Order		
Author	PhucPA		
Date	02/06/2019	Priority	High
Actor: Authorized User Summary:			

The authorized user create order by themselves.

Goal:

When an authorized user creates order successfully, the order is created in system and system will assign a technical employee to take care of this order if the order includes doing service at the garage.

Triggers:

The user has an account and wants to buy some product or do some service or both at the garage.

Preconditions:

The user has an account and logged in on the mobile app.

If the order includes service, service must be booked on the free time of garage.

Postconditions:

Success: order is created in the system. If the order includes service, the system will send a notification to the technical employee who assigns for this service.

Fail: order is not created and the system sends fail notification to the client.

Main Success Scenario:

Step	Actor Action	System Response
1	User choose the product and add to cart	Cart is updated with the amount of item in cart and display in screen mobile.
2	User choose the service and add to cart [Alternative 1]	Request choose Date and choose the free time slot
3	User can choose the promotion that the system suggests for them [Alternative 2]	Display information of promotion in cart
4	The user chooses to submit to create order	System process request create order [Exception 1,2]

Alternative Flow:		
No	Actor Action	System Response
1	User don't choose service	
2	User don't choose promotion	
Exceptions:		
No	Actor Action	System Response
1	Request create order is successfully	Order is created successfully and show detail order on screen
2	Request create order is failed	Network error message
Business rules:		
Book service must be including date and time choice.		

Table 18 - Create order

2.3.3 <Technical> Overview use case

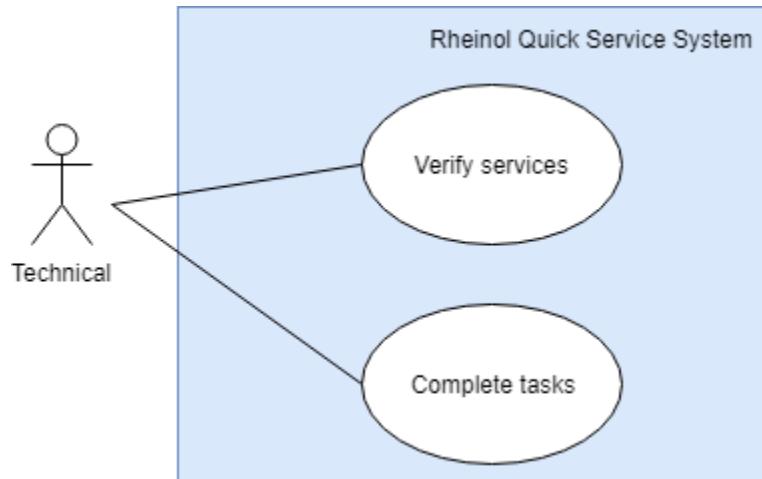


Figure 17 - <Technical> Overview use case

2.3.3.1 Verify service order

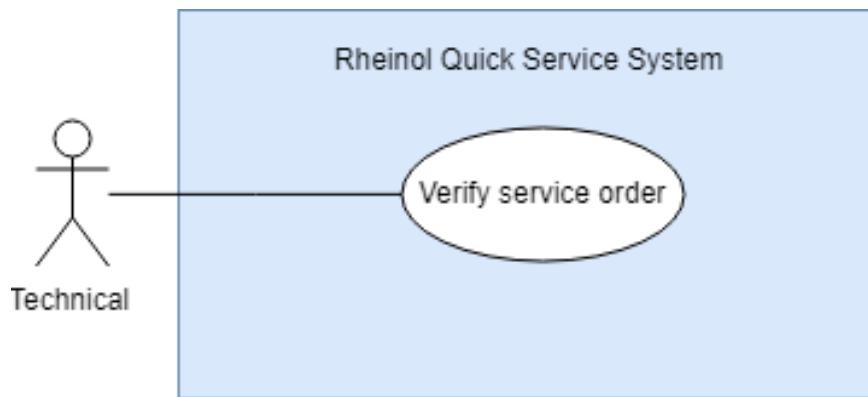


Figure 18 – Verify service order

USE CASE – RQS_UC_13			
Use Case No.	RQS_UC_13	Use Case Version	2.0
Use Case Name	Verify services		
Author	NamDHP		
Date	02/06/2019	Priority	Normal
Actor: Technical			
Summary: This use case allows the technical could change order's status to verify services.			
Goal: The technical could change order's status to verify services successfully, and the notification is sent to the sales clerk.			
Triggers: There is the order has service which was booked time before.			
Preconditions: Guest must log in into the web system with the role technical.			

Postconditions:

Success: the order is changed status, and the system sends the notification to the sales clerk.

Fail: the order cannot be changed status, and the system shows the error message.

Main Success Scenario:

Step	Actor Action	System Response
1	The technical go to "Verify service" view	The system shows the order which is waited to verify.
2	The technical select the order	The system shows order's detail and display field data for technical input: Description: input type text.
3	The technical inputs description	
4.1	The technical send command to change status.	The system will display an error message if the order's status cannot change. [Exception 1]
4.2		The order's status is changed to verify the service.

Exceptions:

No	Actor Action	System Response
1	The technical confirms to change the order's status.	The system will display an error message to notify the error.

Business rules:

Table 19 - verify services

- Only technical can change status to verify service.

2.3.3.2 Complete task

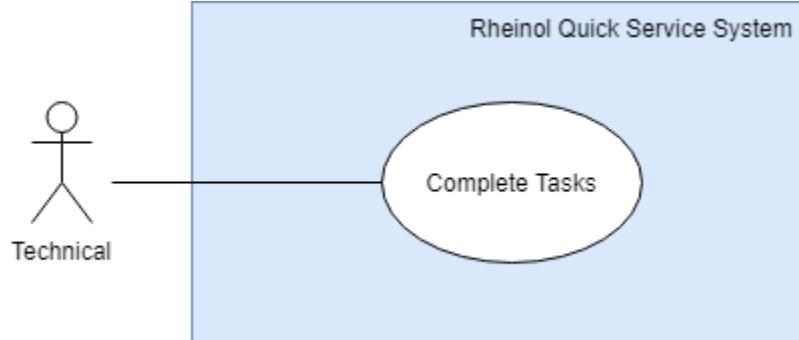


Figure 19 - Complete task

USE CASE - RQS_UC_15			
Use Case No.	RQS_UC_15	Use Case Version	2.0
Use Case Name	Complete tasks		
Author	VuLTD		
Date	02/06/2019	Priority	Normal
Actor:	Technical		
Summary:	This use case allows the technical could change an order's status to complete tasks.		
Goal:	The technical could change an order's status to complete tasks successfully, and the notification is sent to the customer.		
Triggers:	The technical has finished the tasks in order.		
Preconditions:			

Guest must log in into the web system with the role Technical.

The order's status is in progress.

Postconditions:

Success: the order is changed status, and the system sends the notification to the customer.

Fail: the order cannot be changed status, and the system shows the error message.

Main Success Scenario:

Step	Actor Action	System Response
1	The technical go to "Doing" view	The system shows the order which is in progress.
2	The technical sends a command to change the order's status request through the button.	The system shows confirm pop up.
3.1	The technical confirms to change the order's status.	The system will display an error message if the order's status cannot change. [Exception 1]
3.2		The order's status is changed to confirm completing tasks.

Exceptions:

No	Actor Action	System Response
1	The technical confirms to change the order's status.	The system will display an error message to notify the error.

Table 20 - Complete task

Business rules:

Only technical can change status to complete task.

2.3.4 < Sales clerk > Overview use case

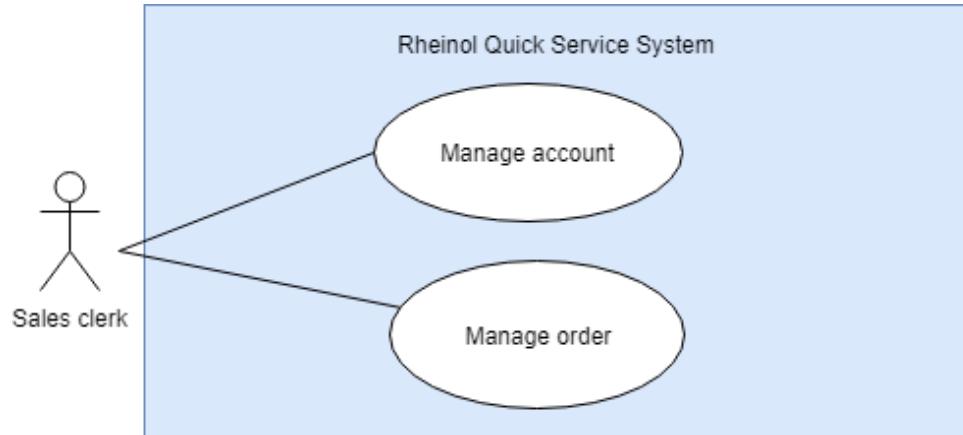


Figure 20 - <Sales clerk> Overview use case

2.3.4.1 Confirm to do Order

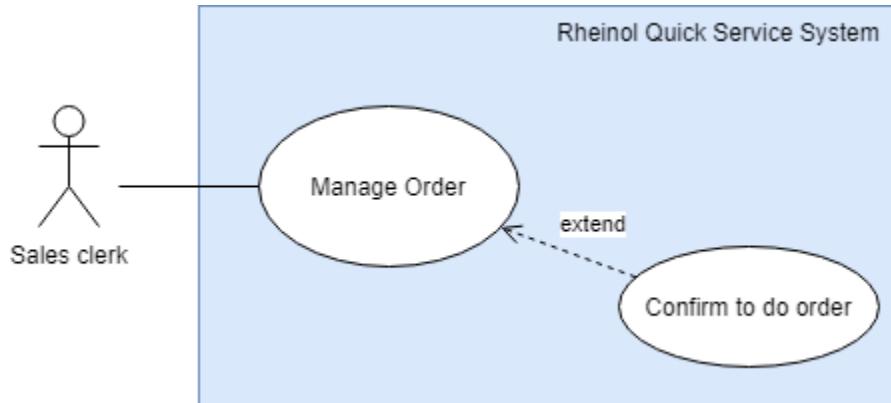


Figure 21 - Confirm to do order

USE CASE – RQS_UC_16			
Use Case No.	WCS_UC_16	Use Case Version	2.0
Use Case Name	Confirm to do order		
Author	NamDHP		

Date	02/06/2019	Priority	Normal
Actor:			
Sales clerk			
Summary:			
This use case allows the sales clerk could change order's status to confirm to do order.			
Goal:			
The sales clerk could change order's status to confirm successfully, and the notification is sent to technical to do order.			
Triggers:			
After the technical checks order and change the order's status to waiting.			
Preconditions:			
Guest must log in into the web system with the role Sales clerk.			
Postconditions:			
Success: the order is changed status, and the system sends the notification to technical to do order.			
Fail: the order cannot be changed status, and the system shows the error message.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	The sales clerk goes to "Order" view.	The system shows the list order with "change status" button in each order.	
2	The sales clerk sends a command to change the order's status request through the button.	The system shows confirm pop up.	
3.1	The sales clerk confirms to	The system will display an error message if the order's status cannot change.	

	change the order's status.	[Exception 1]
3.2		The order's status is changed to confirm doing the order.
Exceptions:		
No	Actor Action	System Response
1	The sales clerk confirms to change the order's status.	The system will display an error message to notify the error.

Business rules:

The orders with a description and previously confirmed by the technical before, the sales clerk will be able to change the order's status to confirm doing the order.

The order's changed status is in progress.

Table 21 - Confirm to do order

2.3.4.2 Add account

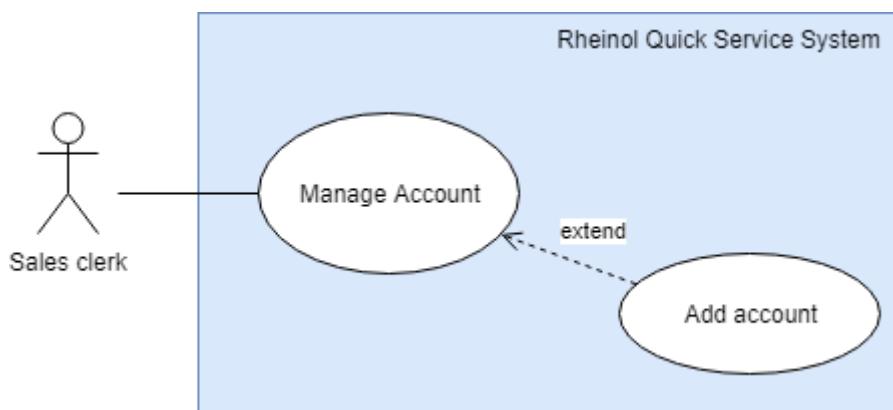


Figure 22 - Add account

USE CASE – RQS_UC_17			
Use Case No.	RQS_UC_17	Use Case Version	2.0

Use Case Name	Add Account		
Author	NamDHP		
Date	02/06/2019	Priority	Normal
Actor:			
Sales clerk			
Summary:			
This use case allows Sales clerk could add a new account.			
Goal:			
When Sales clerk added account successfully, the account is saved in System.			
Triggers:			
There is a new account that needs to be added into System.			
Preconditions:			
Guest must log in into the web application with the role Sales clerk.			
Postconditions:			
Success: Account is saved in System.			
Fail: Account is not added to the System and shows a notification.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Sales clerk goes to new account view.	Account information screen display fields data for user input: Name: textbox, required, length more than 6 characters. Email: textbox, required, format (abc@email.com). Phone: textbox, required, format (begin is 0, length 10 integer).	

		<p>Address: textbox, length more than 6 characters.</p> <p>Date of birth: input type date.</p> <p>Username: textbox, length more than 6 characters.</p> <p>Password: text box, length more than 6 characters.</p> <p>Retype-Password: text box, must be the same Password.</p> <p>Role: combo box, required, select from a list role.</p>
2	Sales clerk inputs information	
3	Sales clerk sends a command to add a new account request.	<p>The system validates information, the system will display a message if the information is not matched.</p> <p>[Exception 1,2]</p>
4		<p>The account is added to the database and system will show message successfully. All input text will reset.</p>
Exceptions:		
No	Actor Action	System Response
1	Sales clerk sends a command to add a new account request.	The system shows an error message to ask user input missing required fields or condition fields.
2	Email exists in the system	The message will appear to notify admin that email has existed.

Business rules:

The account belongs to only one role.

When the new account is added to the System, the account's status is always enabled.

Table 22 - Add account

2.3.4.3 Change status account

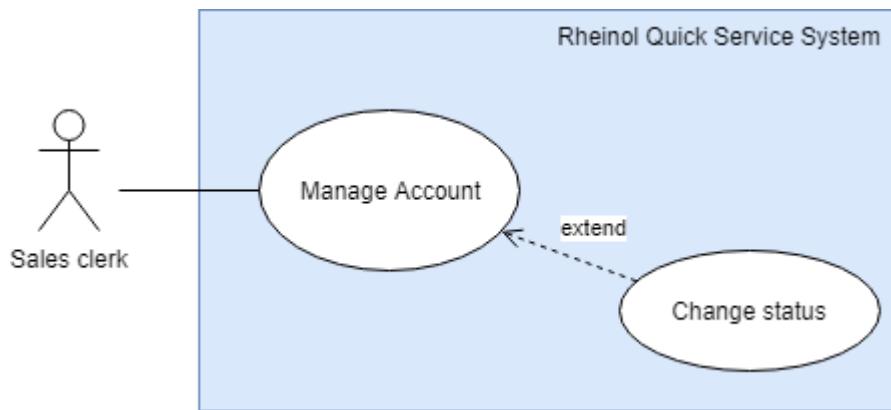


Figure 23 - Change status account

USE CASE – RQS_UC_18			
Use Case No.	RQS_UC_18	Use Case Version	2.0
Use Case Name	Change status		
Author	NamDHP		
Date	02/06/2019	Priority	Normal
Actor:			
Sales clerk			
Summary:			
This use case allows sales clerk could change an account's status.			
Goal:			
When sales clerk change account's status successfully, this account is changed in the system.			

Triggers:

There is an account that needs to change status.

Preconditions:

Guest must log in into the web application with role sales clerk.

Postconditions:

Success: an account is changed in the system.

Fail: account is not changed to the system and shows a notification.

Main Success Scenario:

Step	Actor Action	System Response
1	Sales clerk goes to list account view.	List accounts displayed with the change button
2	Sales clerk sends a command to change the account's status request.	The system will show a message change successfully.
3		The account will show a failed message if the account's status does not change

Business rules:

Just account has the status 'disable' which can change status.

Only sales clerk can change an account's status to the system.

Table 23 - Change status account

2.3.4.4 Edit order

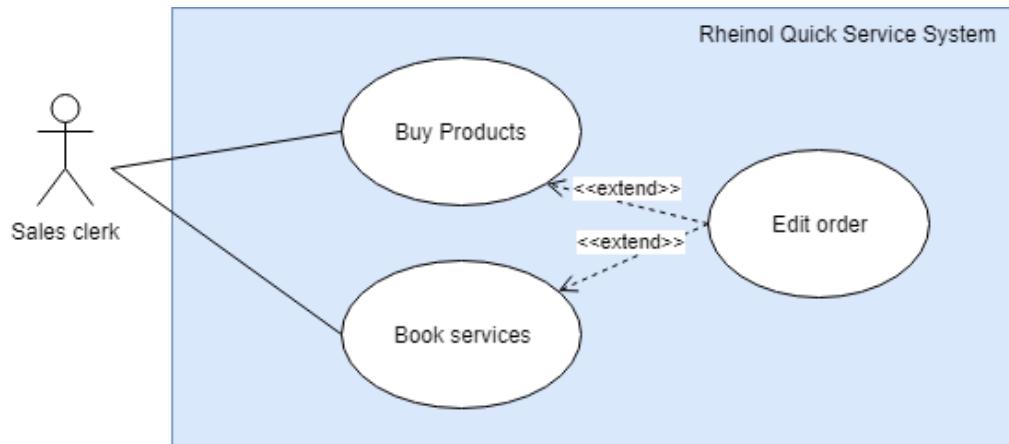


Figure 24 - Edit order

USE CASE – RQS_UC_19						
Use Case No.	RQS_UC_19	Use Case Version	2.0			
Use Case Name	Edit Order					
Author	PhucPA					
Date	02/06/2019	Priority	Normal			
Actor:						
Sales clerk						
Summary:						
Sales clerk edit order base on customer's demand or switch status for technical employee know when to start doing service.						
Goal:						
When Sales clerk edit order successfully, the order is edited in the system. When Sales clerk changes the status of the order from idle to doing, a technical employee will receive notification about this change.						
Triggers:						
The user wants to buy more product or book more service or both.						
Technical send notification about description of the order included service						
Preconditions:						

Sales clerk must log in into the web application with the role of the Sales clerk.

Order must be created before.

Postconditions:

Success: order is edited in the system. The technical employee will receive notification about a change of status order from idle to doing.

Fail: order is not edited and system send fail notification to Sales clerk's screen.

Main Success Scenario:

Step	Actor Action	System Response
1	Sales clerk logged in and access to edit order page	Edit order page include: + Name customer: textbox, required, auto complete + Product: modal popup + Service: modal popup + Total money: label + Submit: button + Cancel: button
2	Sales clerk input name customer and choose product base on customer's opinion	
3	Sales clerk choose service base on customer's opinion [Alternative 1]	Request choose Date and choose the free time slot
4	A sales clerk can pick promotion suitable for this order	

5	Sales clerk choose submit to process edit order	System process request edit order [Exception 1,2]
Alternative Scenario:		
No	Actor Action	System Response
1	Sales clerk don't add more service	
Exceptions:		
No	Actor Action	System Response
1	Request edit order is successfully	Order is edited successfully and back to list order
2	Request edit order is failed	Network error message
Business rules:		
The only Sales clerk can edit information of order		
Anonymous account is for the guest that use rheinol's service		

Table 24 - Edit order

2.3.5 <Director> Overview use case

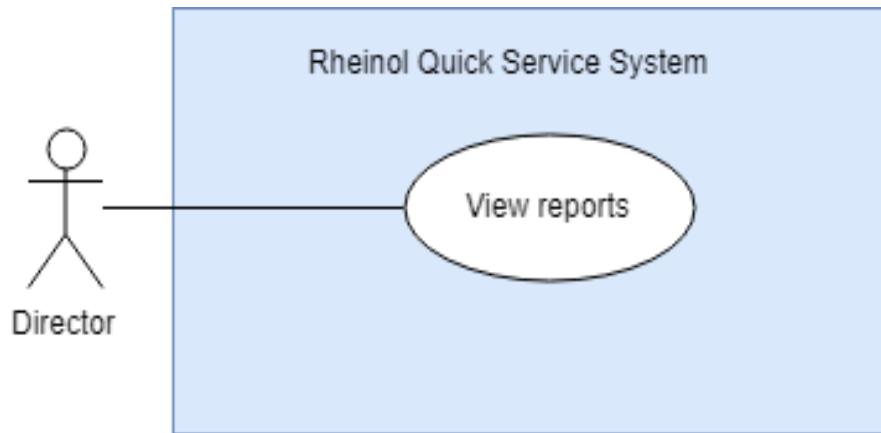


Figure 25 - View reports

USE CASE – RQS_UC_20			
Use Case No.	RQS_UC_20	Use Case Version	2.0
Use Case Name	View Report		
Author	NamDHP		
Date	02/06/2019	Priority	Normal
Actor:	Director		
Summary:	This use case allows the director could view the report.		
Goal:	When the director selects the report's type and chooses a suitable timeline, the director can view report chart.		
Triggers:	The director wants to view the report.		
Preconditions:	Guest must log in into the web application with role Director.		
Postconditions:	Success: report chart will be shown. Fail: report chart will not be shown and the failed message.		
Main Success Scenario:			
Step	Actor Action	System Response	
1	The director goes to report view.	Report screen display fields data which information for the user: Type Timeline: select box, select one type. Time input: input type date, required.	

2	Director select and input information										
3.1	Director sends a command to view the report request.	The system check information, the system will display a message if the information is not matched. [Exception 1]									
3.2		The report chart will be shown.									
Exceptions:											
<table border="1"> <thead> <tr> <th>No</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>Director sends a command to view the report request.</td><td>The system shows an error message to ask user input missing required fields or condition fields.</td></tr> <tr> <td>2</td><td>The time is not suitable</td><td>The message will appear to notify admin that time is invalid.</td></tr> </tbody> </table>			No	Actor Action	System Response	1	Director sends a command to view the report request.	The system shows an error message to ask user input missing required fields or condition fields.	2	The time is not suitable	The message will appear to notify admin that time is invalid.
No	Actor Action	System Response									
1	Director sends a command to view the report request.	The system shows an error message to ask user input missing required fields or condition fields.									
2	The time is not suitable	The message will appear to notify admin that time is invalid.									
Business rules: When the director wants to view “custom date timeline” type, the director could only select a time that is an interval of 10 days apart.											

Table 25 - View report

2.3.6 <Admin> Overview use case

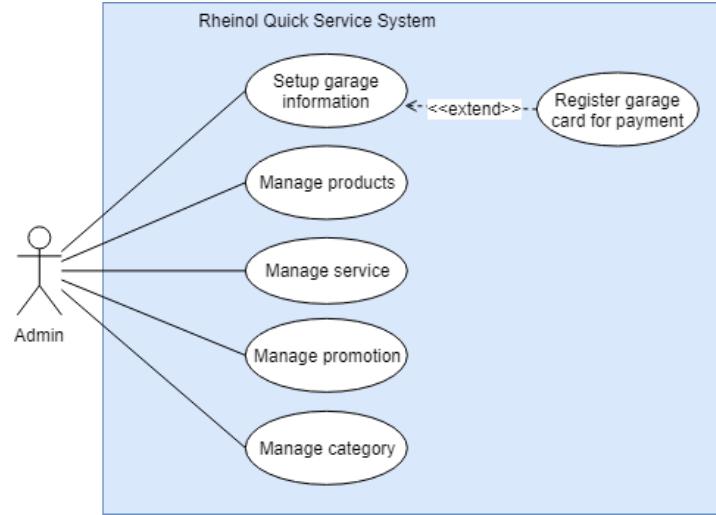


Figure 26 - <Admin> Over view use case

2.3.6.1 Add a product into the system

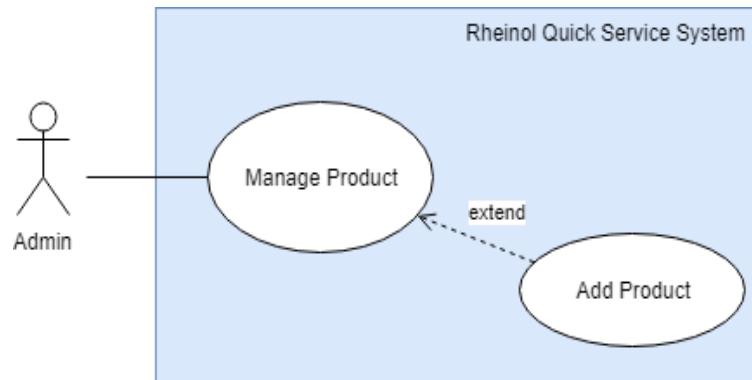


Figure 27 - Add product

USE CASE - RQS_UC_21			
Use Case No.	RQS_UC_21	Use Case Version	2.0
Use Case Name	Add Product		
Author	NamDHP		
Date	22/05/2019	Priority	High
Actor:			
Admin			
Summary:			
This use case allows admin could add a new product.			

Goal:

When admin added product successfully, this product was saved in the system.

Triggers:

There is a new product that needs to be added to the system.

Preconditions:

Guest must log in into the web application with role admin.

Postconditions:

Success: A product is saved in the system.

Fail: The product is not added to the system and shows a notification.

Main Success Scenario:

Step	Actor Action	System Response
1	Admin goes to new product view.	<p>The system requires information from Admin:</p> <p>Product's name: free input text, required, length 6-200.</p> <p>Product's price: free input text, required, number.</p> <p>Product's image: required, select at least one picture.</p> <p>Product's capacity: free input text, number.</p> <p>Product's description: free input text, length 3-250</p> <p>Category: required, select one category.</p>
2	Admin inputs information	
3.1	Admin sends a command to add a new product request.	<p>The system validates information, the system will display a message if the information is not matched.</p> <p>[Exception 1,2]</p>

3.2		The product is added to the database and system will show message successfully. All input text will reset.	
Exceptions:			
No	Actor Action	System Response	
1	Admin sends a command to add a new product request.	The system shows an error message to ask user input missing required fields or condition fields.	
2	The product name has existed in the system	The message will appear to notify admin that product's name has existed.	
Alternative:			
No	Actor Action	System Response	
1	Admin chooses import Excel.	The system shows device data that an admin can choose the excel file.	
2.1	Admin sends a command to add a new product request.	The system validates information, the system will display a message if the information in the excel file is not matched.	
2.2		The products are added to the database and system will show message successfully.	
Business rules:			
<p>The product belongs to only one category.</p> <p>The new product is added that must have the approval of the superior.</p> <p>When the new product is added to the system, the product's status is always enabled.</p> <p>Only admin can add a new product to the system.</p>			

Table 26 - Add product

2.3.6.2 Update existed product

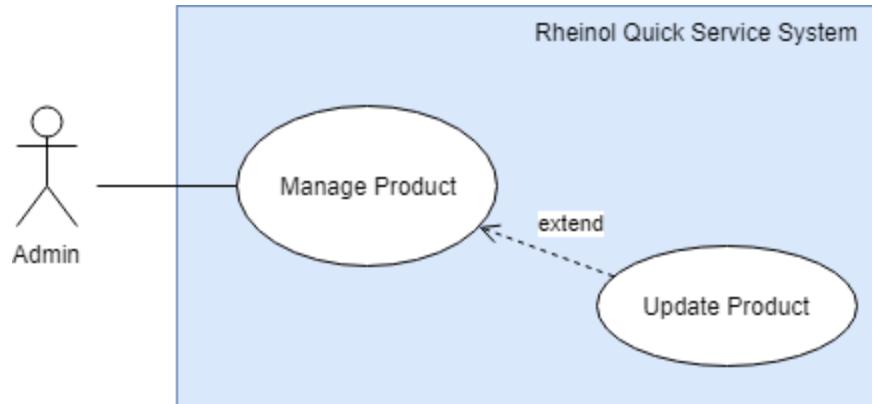


Figure 28 - Update product

USE CASE – RQS_UC_22			
Use Case No.	RQS_UC_22	Use Case Version	2.0
Use Case Name	Update Product		
Author	NamDHP		
Date	22/05/2019	Priority	Normal
Actor:			
Admin			
Summary:			
This use case allows admin could edit product's detail.			
Goal:			
When admin edited product's detail successfully, the new information was saved in the system.			
Triggers:			
There is the product's detail that needs to be edited in the system.			
Preconditions:			
Guest must log in into the web application with role admin.			
That product must exist in the system.			
Postconditions:			
Success: the new product's detail is saved in the system.			

Fail: the new product's detail is not edited in the system and shows a notification.

Main Success Scenario:

Step	Actor Action	System Response
1	Admin goes to update product view.	<p>The system shows the old product's detail:</p> <p>Product's name: required, length 3-50.</p> <p>Product's price: required, number.</p> <p>Product's image: required, select at least one picture.</p> <p>Product's capacity: free input text, number.</p> <p>Product's description: length 3- 250</p> <p>Category: required, select one category.</p>
2	Admin inputs new information	
3.1	Admin sends a command to edit new product's detail request.	<p>The system validates information, the system will display a message if the information is not matched.</p> <p>[Exception 1,2]</p>
3.2		<p>The product's detail is edited in the database and system will show message successfully. The system will return the list product view.</p>

Exceptions:

No	Actor Action	System Response
1	Admin sends a command to edit new product's detail request.	<p>The system shows an error message to ask user input missing required fields or condition fields.</p>

2	The product name has existed in the system	The message will appear to notify admin that product's name exists.
Business rules: The product is edited that must have the approval of the superior. When a product's status is enabled, that product cannot update. When a product's status is edited with enable, that product will be shown in the mobile application. When a product's status is edited with disabled, that product will not be shown in the mobile application. Only admin can edit product's detail to the system.		

Table 27 - Update product

2.3.6.3 Add a category into the system

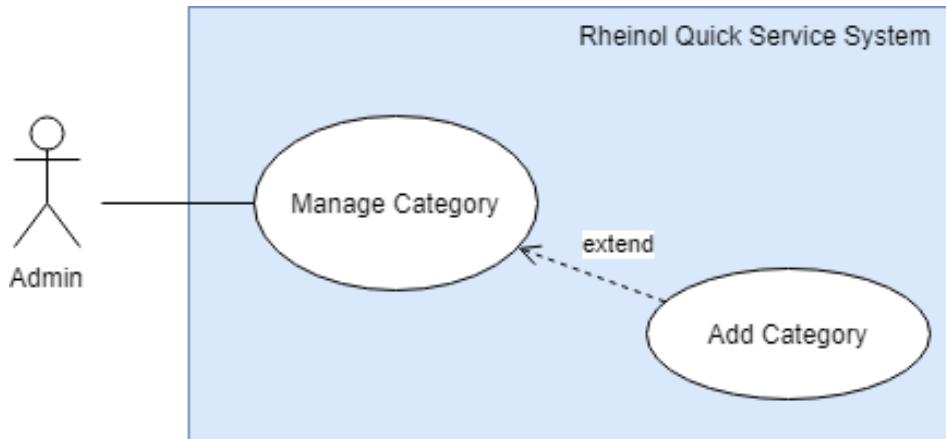


Figure 29 - Add category

USE CASE – RQS_UC_23			
Use Case No.	RQS_UC_23	Use Case Version	2.0
Use Case Name	Add Category		
Author	PhucPA		
Date	22/05/2019	Priority	Normal
Actor:			

Admin

Summary:

This use case allows Admin could add a new category.

Goal:

When admin added category successfully, this category was saved in System.

Triggers:

There is a new category that needs to be added into System.

Preconditions:

Guest must log in into the web application with role Admin.

Postconditions:

Success: category is saved in System.

Fail: category is not added to the System and show notification.

Main Success Scenario:

Step	Actor Action	System Response
1	Admin goes to add a new category view.	The system requires information from Admin: Category's name: free input text, required, length 3-50. Category's image: required, select one picture. Category's type: combo box with tree value (product, service, car).
2	Admin inputs information	
3.1	Admin sends a command to add a new category request.	The system validates information, the system will display a message if the information is not matched. [Exception 1,2]

3.2		The category is added to the database and system will show message successfully. All input text will reset.
Exceptions:		
No	Actor Action	System Response
1	Admin sends a command to add a new category request.	The system shows an error message to ask user input missing required fields or condition fields.
2	Category's name exists in the system	The message will appear to notify admin that category's name exists.

| **Business rules:** | | |
| The new category is added that must have the approval of the superior. When the new category is added to the System, the category's status is always enabled. Only admin can add a new category to the system. | | |

Table 28 - Add category

2.3.6.4 Update a category

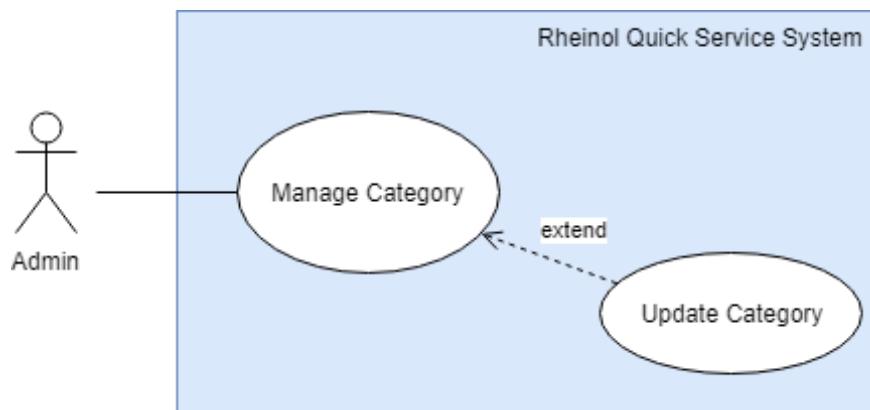


Figure 30 - Update category

USE CASE – RQS_UC_24						
Use Case No.	RQS_UC_24	Use Case Version	2.0			
Use Case Name	Update Category					
Author	PhucPA					
Date	22/05/2019	Priority	Normal			
Actor:						
Admin						
Summary:						
This use case allows admin could edit the category's detail.						
Goal:						
When admin edited category's detail successfully, the new information is saved in the system.						
Triggers:						
There is the category's detail that needs to be edited in the system.						
Preconditions:						
Guest must log in into the web application with role admin.						
That category must exist in the system.						
Postconditions:						
Success: The new category's detail is saved in the system.						
Fail: The new category's detail is not edited in the system and shows a notification.						
Main Success Scenario:						
Step	Actor Action	System Response				
1	Admin goes to update category view.	The system shows the old category's detail: Category's name: required, length 3-50. Category's type: combo box.				

		Category's image: select type, select one image, required. Category's status: checkbox.
2	Admin inputs new information	
3.1	Admin sends a command to edit category's detail request.	The system validates information, the system will display a message if the information is not matched. [Exception 1,2]
3.2		The category's detail is edited in the database and system will show message successfully. The system will return the list category view.
Exceptions:		
No	Actor Action	System Response
1	Admin sends a command to edit new category's detail request.	The system shows an error message to ask user input missing required fields or condition fields.
2	Category's name has existed in the system.	The message will appear to notify admin that category's name has existed.
Business rules:		
<p>The new category is added that must have the approval of the superior</p> <p>When the category's status is edited is enable, that category will be shown in the mobile application.</p> <p>When the category's status is edited is disabled, that category does not have any product which status is true, that category will not be shown in the mobile application.</p> <p>Only admin can edit the category's detail to the system.</p>		

Table 29 - Update category

2.3.6.5 Add a promotion into the system.

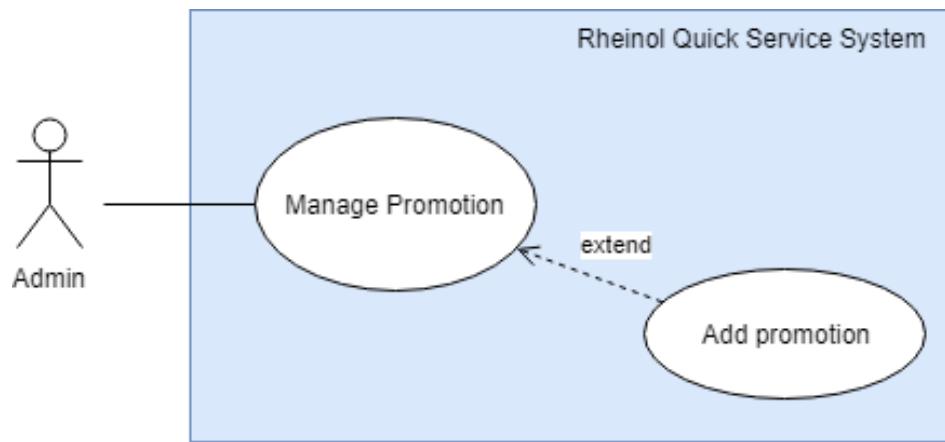


Figure 31 - Add promotion

USE CASE - RQS_UC_25			
Use Case No.	RQS_UC_25	Use Case Version	2.0
Use Case Name	Add Promotion		
Author	NamDHP		
Date	22/05/2019	Priority	High
Actor:			
Admin			
Summary:			
This use case allows Admin could add new promotion.			
Goal:			
When admin added promotion successfully, a new promotion is saved in System.			
Triggers:			
There is a new promotion that the enterprise wants to apply and add to the system.			
Preconditions:			
Guest must log in into the web application with role Admin.			
Postconditions:			
Success: a new promotion is saved in the system.			

Fail: promotion is not added to the System and shows a notification.

Main Success Scenario:

Step	Actor Action	System Response
1	Admin goes to new promotion view.	<p>The system requires information from Admin:</p> <p>Promotion's name: free input text, required, length 6-100.</p> <p>Time's promotion: DateTime input, required, there are 2 fields "From" and "To".</p> <p>Product: product's list.</p> <p>Service: service's list.</p> <p>Promotion's value: input text, required, number: from 1 to 100 (with value "%").</p> <p>Promotion's image: required, select one picture.</p> <p>Promotion's description: input text, length: 5 – 250.</p> <p>Promotion's members: select from list checked box.</p> <p>Promotion's type: required, select one type.</p>
2	Admin inputs information	<p>List product: the system will show popup modal and admin can select any products.</p> <p>List service: the system will show popup modal and admin can select many services.</p>
3.1	Admin sends a command to add a new promotion request.	<p>The system validates information, the system will display a message if the information is not matched.</p> <p>[Exception 1,2]</p>
3.2		<p>The promotion is added to the database and system will show message successfully. All input text will reset.</p>

Exceptions:

No	Actor Action	System Response
1	Admin sends a command to add a new product request.	The system shows an error message to ask user input missing required fields or condition fields.
2	Admin does not select a product or service.	The message will appear to notify admin that admin has to select at least one product or service to be used in promotion.

Business rules:

The new promotion is added that must have the approval of the superior with a complete document.

The promotion can apply with multiple products and services.

If Promotion's type is "combo", the products or services are selected together according to regulations, the promotion will apply.

If Promotion's type is "single", and the products or services are selected to follow the promotion, the promotion will apply.

The promotion will apply to the member's type that was selected.

When the new promotion is added to the System, the promotion's status is always disabled.

Table 30 - Add promotion

2.3.6.6 Delete a promotion.

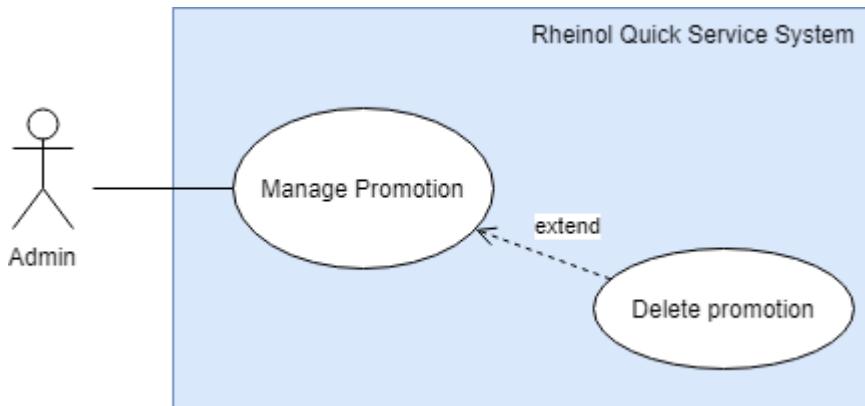


Figure 32 - Delete promotion

USE CASE – RQS_UC_26			
Use Case No.	RQS_UC_26	Use Case Version	2.0
Use Case Name	Delete Promotion		
Author	NamDHP		
Date	22/05/2019	Priority	Normal
Actor:			
Admin			
Summary:			
This use case allows admin could delete promotion.			
Goal:			
When admin delete promotion successfully, the System will remove promotion from the database.			
Triggers:			
There is the promotion that needs to be deleted in System.			
Preconditions:			
Guest must log in into the web application with role Admin.			
That promotion must exist in System.			
Postconditions:			
Success: System removes the promotion from the database.			
Fail: The promotion is not deleted and shows a notification.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Admin goes to list promotion view.	The system shows list promotion with “Delete” button with each promotion.	
2	Admin sends a command to delete promotion request through button	System shows confirm pop up	

3	Admin confirms to delete the promotion	The system will remove promotion and show list promotion.
Exceptions:		
No	Actor Action	System Response
1	Admin confirms to delete the promotion	If the promotion cannot delete, the system shows an error message.
Business rules:		
When promotion's status is enabled, that promotion cannot be removed.		
Only admin can delete promotion in the system.		
The promotion is deleted that must have the approval of the superior with a complete document.		

Table 31 - Delete promotion

2.3.6.7 Add a service.

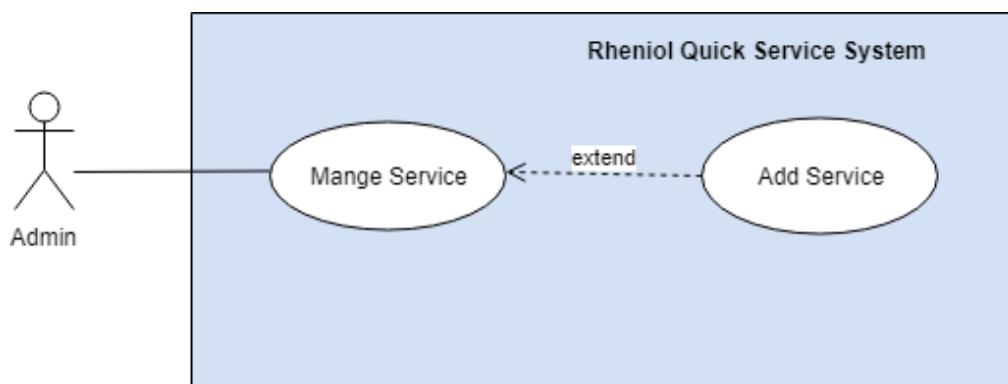


Figure 33 - Add service

USE CASE – RQS_UC_27			
Use Case No.	RQS_UC_27	Use Case Version	2.0
Use Case Name	Add Service		
Author	PhucPA		
Date	02/06/2019	Priority	High
<p>Actor: Admin</p> <p>Summary: This use case allows Admin could add new service.</p> <p>Goal: When admin added service successfully, service was saved in System.</p> <p>Triggers: There is a new service that needs to be added into System.</p> <p>Preconditions: Guest must log in into the web application with role Admin.</p> <p>Postconditions: Success: service is saved in System. Fail: service is not added to the System and show notification.</p> <p>Main Success Scenario:</p>			
Step	Actor Action	System Response	
1	Admin goes to new service view.	Service information screen display fields data for user input: Name: textbox, required, length more than 6 characters. Price: textbox, required, type integer. Duration: textbox, required, type double.	

		Description: textbox, length more than 6 characters. Image: select type, select one image. Category: combo box, required, select a category
2	Admin inputs information	
3.1	Admin sends a command to add a new service request.	The system validates information, the system will display a message if the information is not matched. [Exception 1,2]
3.2		The service is added to the database and system will show message successfully. All input text will reset.
Exceptions:		
No	Actor Action	System Response
1	Admin sends a command to add a new service request.	The system shows an error message to ask user input missing required fields or condition fields.
2	The name has existed in the system	The message will appear to notify admin that username has existed.
Business rules:		
<p>The service belongs to only one category.</p> <p>The new service is added that must have the approval of the superior</p> <p>When the new service is added to the system, the service's status is always enabled.</p> <p>Only admin can add a new service to the system.</p>		

Table 32 - Add service

2.3.6.8 Edit a service.

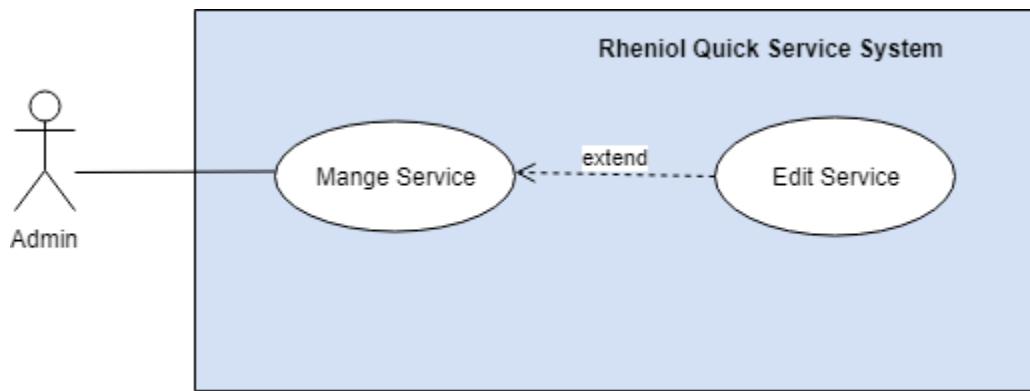


Figure 34 - Edit service

USE CASE – RQS_UC_28						
Use Case No.	RQS_UC_28	Use Case Version	2.0			
Use Case Name	Edit Service					
Author	PhucPA					
Date	02/06/2019	Priority	Normal			
Actor:						
Admin						
Summary:						
This use case allows Admin could edit service information.						
Goal:						
When admin edited service successfully, new service's information is updated in System.						
Triggers:						
There is a service that needs to be edited into System.						
Preconditions:						
Guest must log in into the web application with role Admin.						
Postconditions:						
Success: new service's information is updated in System.						
Fail: service is not edited to the System and shows a notification.						

Main Success Scenario:		
Step	Actor Action	System Response
1	Admin goes to edit service view.	<p>Edit service information screen display fields data which information for user edit:</p> <p>Name: textbox, required, length more than 6 characters.</p> <p>Price: textbox, required, type integer.</p> <p>Duration: textbox, required, type double.</p> <p>Description: textbox, length more than 6 characters.</p> <p>Image: select type, select one image.</p> <p>Category: combo box, required, select a category</p>
2	Admin edit information	
3.1	Admin sends command to edit service request.	<p>The system validates information, the system will display a message if the information is not matched.</p> <p>[Exception 1,2]</p>
3.2		<p>The service is edited to the database and system will show message successfully. All input text will reset.</p>
Exceptions:		
No	Actor Action	System Response
1	Admin sends a command to edit service request.	The system shows an error message to ask user input missing required fields or condition fields.

2	The name has existed in the system	The message will appear to notify admin that name has existed.
<p>Business rules:</p> <p>The service is edited that must have the approval of the superior.</p> <p>When a service's status is enabled, that service cannot update.</p> <p>When a service's status is edited with enable, that service will be shown in the mobile application.</p> <p>When service's status is edited with disabled, that service will not be shown in the mobile application.</p> <p>Only admin can edit the service's detail to the system.</p>		

Table 33 - Edit service

2.3.7 <System> Overview use case

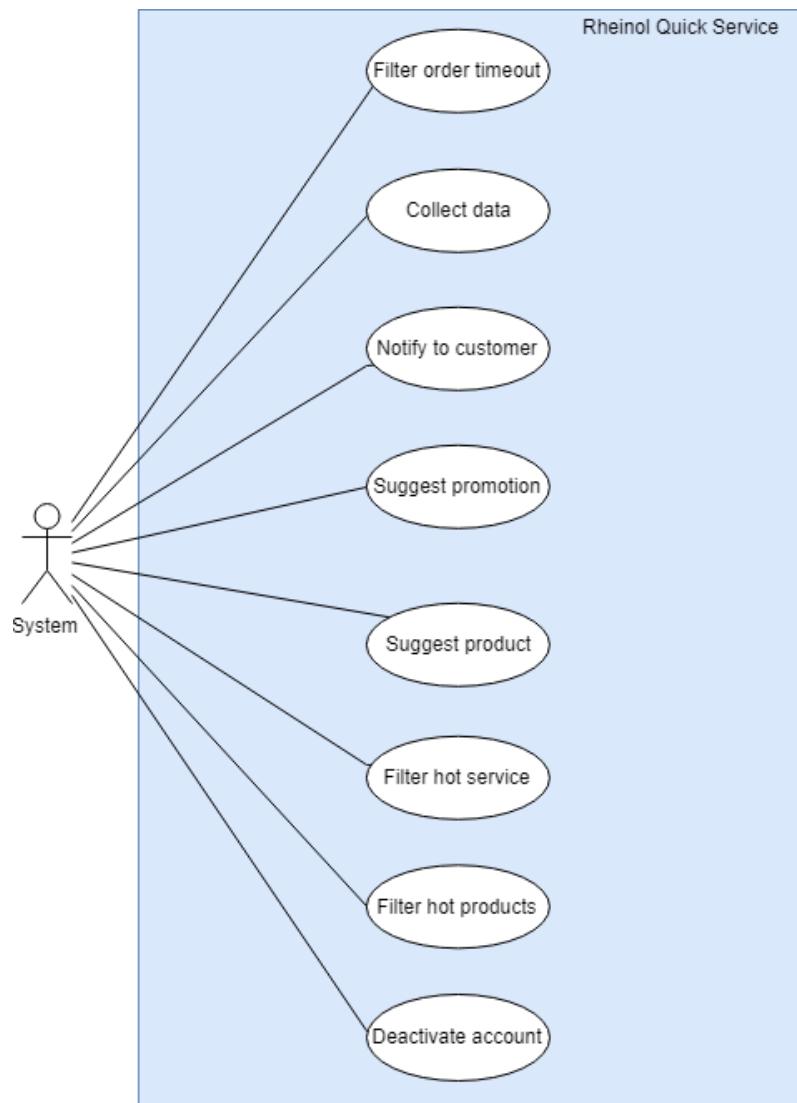


Figure 35 - <System> Over view

2.3.7.1 Notify to customer

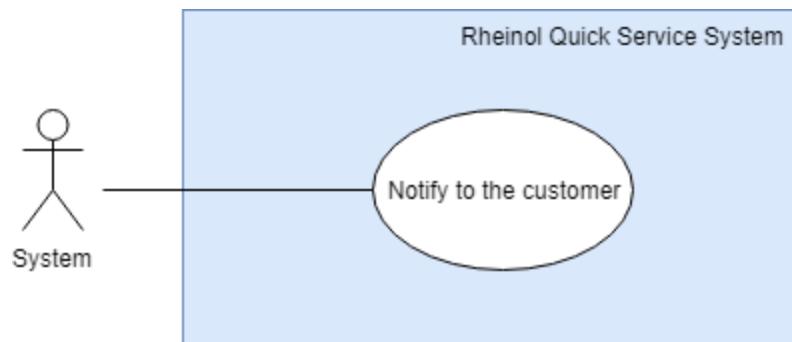


Figure 36 - Notify to customer

USE CASE - RQS_UC_29			
Use Case No.	RQS_UC_29	Use Case Version	2.0
Use Case Name	Notify to the customer		
Author	PhucPA		
Date	02/06/2019	Priority	High
Actor:			
System			
Summary:			
This use case allows the system could notify some information to the customer.			
Goal:			
The system could notify the promotion's information or warranty service.			
Triggers:			
There are promotions or warranty services that need to be notified to customers.			
Preconditions:			
N/A.			
Postconditions:			
Success: the system notifies information to the customer.			
Fail: the system cannot notify information to the customer.			
Main Success Scenario:			
Step	Actor Action	System Response	
1	The system notifies information to the customer.	The notification will be sent to the customer.	
Exceptions:			

No	Actor Action	System Response
1	The system notifies information to the customer.	The notification will not be sent to the customer.

Business rules:

The system will send the promotion's notification before 7 days when the promotion is applied.

The system will send the warranty's notification before 7 days when the service is due.

Table 34 - Notify to customer

2.3.7.2 Filter order timeout

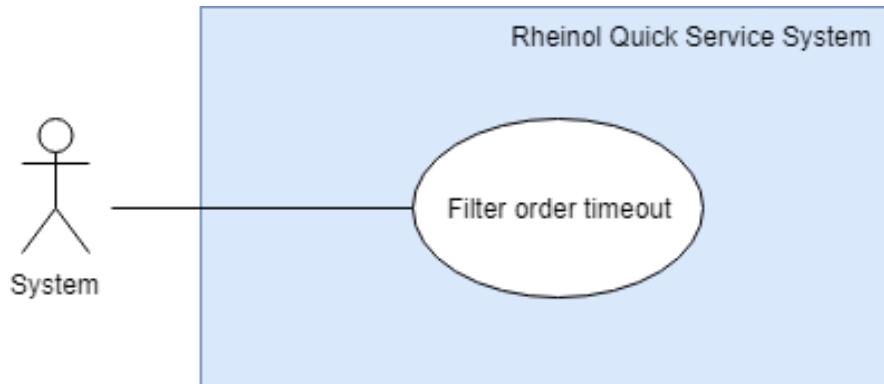


Figure 37 - Filter order timeout

USE CASE – RQS_UC_30			
Use Case No.	RQS_UC_30	Use Case Version	2.0
Use Case Name	Filter order timeout		
Author	PhucPA		
Date	02/06/2019	Priority	Normal
Actor: System			
Summary: This use case allows the system could change an order's status to cancel the order.			

Goal:

The system could change an order's status to cancel the order successfully, and the notification is sent to the customer.

Triggers:

The order's status is still new when the appointment time is over 15 minutes.

Preconditions:

The order's status is new.

Postconditions:

Success: the order is changed status, and the system sends the notification to the customer.

Fail: the order cannot be changed status.

Main Success Scenario:

Step	Actor Action	System Response
1	The system changes the order's status.	The order is changed status.
2	The system sends notifications to the customer.	The customer received notification that order changed status.

Exceptions:

No	Actor Action	System Response
1	The system changes the order's status	The order cannot be changed the status.

Business rules:

The order's status will change to cancel if the appointment time is over 15 minutes when the order's status is new.

Table 35 - Filter order timeout

2.3.7.3 Collect Data

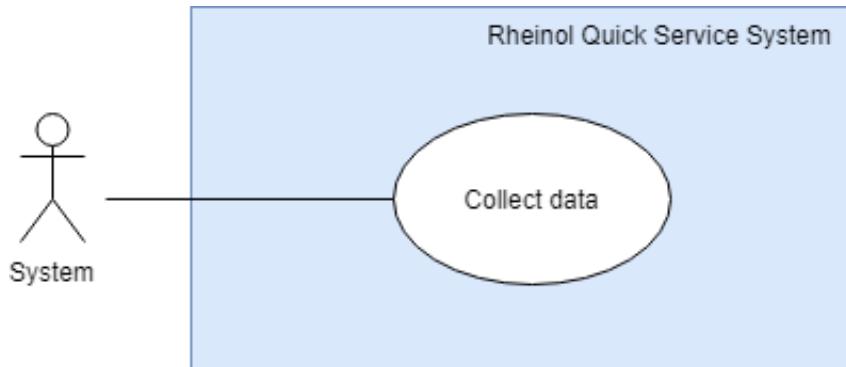


Figure 38 – Collect data

USE CASE – RQS_UC_31			
Use Case No.	RQS_UC_31	Use Case Version	2.0
Use Case Name	Collect data		
Author	VuLTD		
Date	02/06/2019	Priority	Normal
Actor:	System		
Summary:	This use case allows the system could collect data.		
Goal:	The system could collect data successfully.		
Triggers:	The director select timeline.		
Preconditions:	There is data corresponding to the time the employee chooses.		
Postconditions:	Success: the system collects data successfully.		

Fail: the system cannot collect data.

Main Success Scenario:

Step	Actor Action	System Response
1	The system collect data.	The report will be shown.

Exceptions:

No	Actor Action	System Response
1	The system collect data	The system cannot generate report

Business rules:

N/A

Table 36 - Generate report

2.3.7.4 Suggest promotion

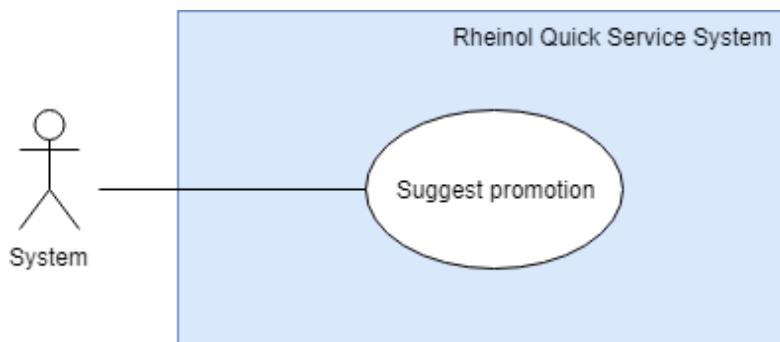


Figure 39 - Suggest promotion

USE CASE – RQS_UC_32			
Use Case No.	RQS_UC_32	Use Case Version	2.0
Use Case Name	Suggest promotion		
Author	VuLTD		
Date	02/06/2019	Priority	Normal

Actor:

System

Summary:

This use case allows the system could suggest promotion.

Goal:

The system could suggest promotion when the customer select product or service.

Triggers:

The customer selects product or service.

Preconditions:

N/A.

Postconditions:

Success: the system suggests the promotion that is corresponding with product or service.

Fail: the system cannot suggest the promotion.

Main Success Scenario:

Step	Actor Action	System Response
1	The system suggests the promotion to the customer.	The corresponding promotion will be shown.

Exceptions:

No	Actor Action	System Response
1	The system suggests the promotion to the customer.	The promotion will not be shown.

Business rules:

N/A.

Table 37 - Suggest promotion

2.3.7.5 Filter hot service

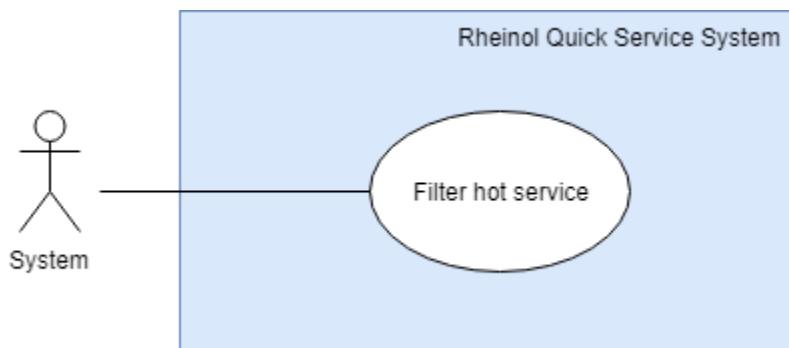


Figure 40 - Filter hot service

USE CASE – RQS_UC_33			
Use Case No.	RQS_UC_33	Use Case Version	2.0
Use Case Name	Filter hot service.		
Author	PhucPA		
Date	02/06/2019	Priority	Normal
Actor:			
System			
Summary:			
This use case allows the system could filter hot service through the number of times that service is booked.			
Goal:			
The system could filter hot service.			
Triggers:			
The services were selected by the customer before.			
Preconditions:			
There are data about the number of times that services are booked.			

Postconditions:

Success: the system filters hot service.

Fail: the system cannot filter hot service.

Main Success Scenario:

Step	Actor Action	System Response
1	The system filters hot service.	The system will show the hot service's list.

Exceptions:

No	Actor Action	System Response
1	The system filters hot service.	The system cannot show the hot service's list.

Business rules:

The system will calculate the number of times that service is booked to provide a hot list of services.

Table 38 - Filter hot service

2.3.7.6 Filter hot product

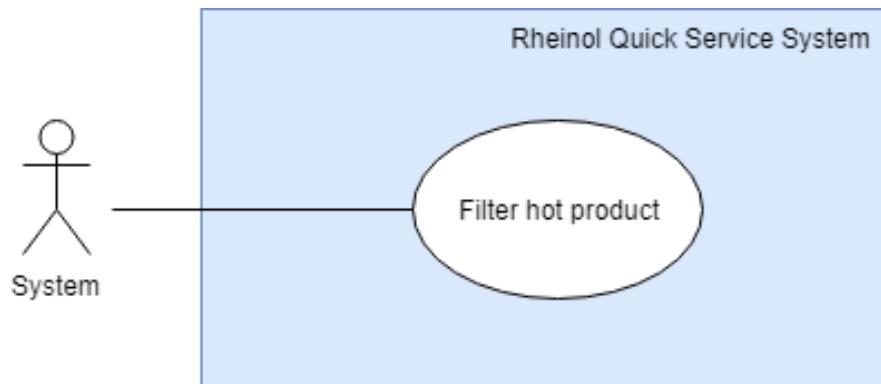


Figure 41 - Filter hot product

USE CASE – RQS_UC_34

Use Case No.	RQS_UC_34	Use Case Version	2.0												
Use Case Name	Filter hot product.														
Author	VuLTD														
Date	02/06/2019	Priority	Normal												
<p>Actor: System</p> <p>Summary: This use case allows the system could filter hot product through the number of times that the product is selected.</p> <p>Goal: The system could filter hot product.</p> <p>Triggers: The products were selected by the customer before.</p> <p>Preconditions: There are data about the number of times that products are selected.</p> <p>Postconditions: Success: the system filters the hot product. Fail: the system cannot filter the hot product.</p>															
<p>Main Success Scenario:</p> <table border="1"> <thead> <tr> <th>Step</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>The system filters hot product.</td><td>The system will show the hot product's list.</td></tr> </tbody> </table> <p>Exceptions:</p> <table border="1"> <thead> <tr> <th>No</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>The system filters hot product.</td><td>The system cannot show the hot product's list.</td></tr> </tbody> </table>				Step	Actor Action	System Response	1	The system filters hot product.	The system will show the hot product's list.	No	Actor Action	System Response	1	The system filters hot product.	The system cannot show the hot product's list.
Step	Actor Action	System Response													
1	The system filters hot product.	The system will show the hot product's list.													
No	Actor Action	System Response													
1	The system filters hot product.	The system cannot show the hot product's list.													

Business rules:

The system will calculate the number of times that the product is selected to provide a hot list of products.

Table 39 - Filter hot product

2.3.7.7 Remove an Account

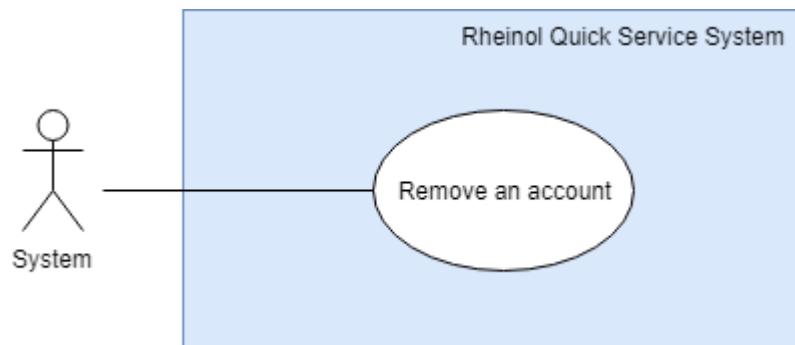


Figure 42 - Remove an account

USE CASE – RQS_UC_35

Use Case No.	RQS_UC_35	Use Case Version	2.0
Use Case Name	Remove an Account.		
Author	NamDHP		
Date	02/06/2019	Priority	Normal

Actor:

System

Summary:

This use case allows the system could remove an account that was disabled in 3 months.

Goal:

The system could remove an account in the database.

Triggers:

The account is disabled for 3 months.

Preconditions:						
N/A.						
Postconditions:						
Success: the system removes an account in the database.						
Fail: the system cannot remove an account.						
Main Success Scenario:						
<table border="1"> <thead> <tr> <th>Step</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>The system removes an account.</td><td>The account is removed in the database.</td></tr> </tbody> </table>	Step	Actor Action	System Response	1	The system removes an account.	The account is removed in the database.
Step	Actor Action	System Response				
1	The system removes an account.	The account is removed in the database.				
Exceptions:						
<table border="1"> <thead> <tr> <th>No</th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>The system removes an account.</td><td>The account is not removed in the database.</td></tr> </tbody> </table>	No	Actor Action	System Response	1	The system removes an account.	The account is not removed in the database.
No	Actor Action	System Response				
1	The system removes an account.	The account is not removed in the database.				
Business rules:						
If the account is disabled in 3 months, the system will remove the account in the database.						

Table 40 - Remove an account

3. Software System Attribute

3.1 Usability

- Easy for the user to become familiar with.
- Easy for the user to achieve their objectives.
- Easy to recall the user interface and how to use it.
- Font style: Roboto
- Font size: 12px – 20px.

3.2 Reliability

- User's card information for payment is not stored in the system.

- The information storing on the database is permanent.

3.3 Availability

- The system must be available at any time 24/7.

3.4 Security

- The system authorizes users by their token.

3.5 Maintainability

- The web application system was designed by following Dependency Inversion. Thus, easy to maintain.

3.6 Portability

- The web application can be run on Chrome browser version 42 or later.
- Mobile application can be run on Android version 9 or later.

3.7 Performance

- The system has successfully tested with the basic standard.

4. Conceptual Diagram

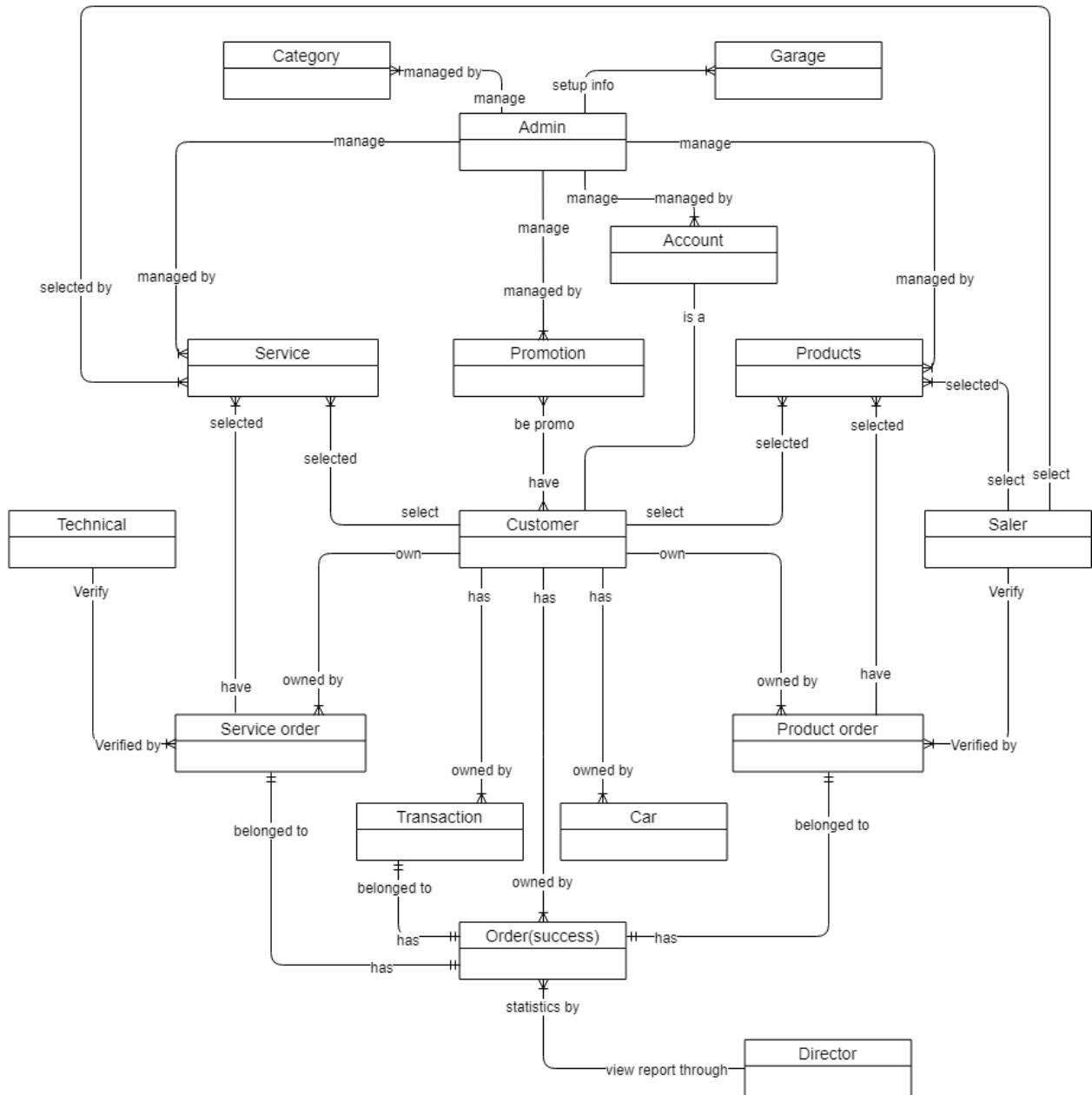


Figure 43 - Conceptual Diagram

Data Dictionary

Entity Data dictionary: describe all content of all entities	
Entity Name	Description
Category	Contain the category information.
Garage	Contain the garage information.
Admin	Contain the admin information.
Account	Abstract entity describes a user in the system.
Service	Contain the service information.
Promotion	Contain the promotion information.

Product	Contain the product information.
Technical	Contain the technical information.
Customer	Contain the customer information.
Sales clerk	Contain the sale information.
Service Order	Contain the orders that have services or both services and products.
Product Order	Contain the orders that have products.
Car	Contain the car information.
Order	Contain the order successfully information.
Director	Contain the director information.

Table 41 - Data dictionary

D. Software Design Description

1. Design Overview

- This document describes the technical and user interface design of Rheinol Quick Service. It includes the architectural design, the detailed design of common functions and business functions and the design of the database model.
- The architectural design describes the overall architecture of the system and the architecture of each main component and subsystem.
- The detailed design describes the static and dynamic structure for each component and functions. It includes class diagrams, class explanations and sequence diagrams for each use cases.
- The database design describes the relationships between entities and the details of each entity.
- Document Overview:
 - o Section 2: gives an overall description of the system architecture design.
 - o Section 3: gives component diagrams that describe the connection and integration of the system.
 - o Section 4: gives the detail design description which includes the class diagram, class explanation, and sequence diagram to details the application functions.
 - o Section 5: describe screens design.
 - o Section 6: describe a fully attributed ERD.
 - o Section 7: describe algorithms.

2. System Architectural Design

2.1 Web application architecture description

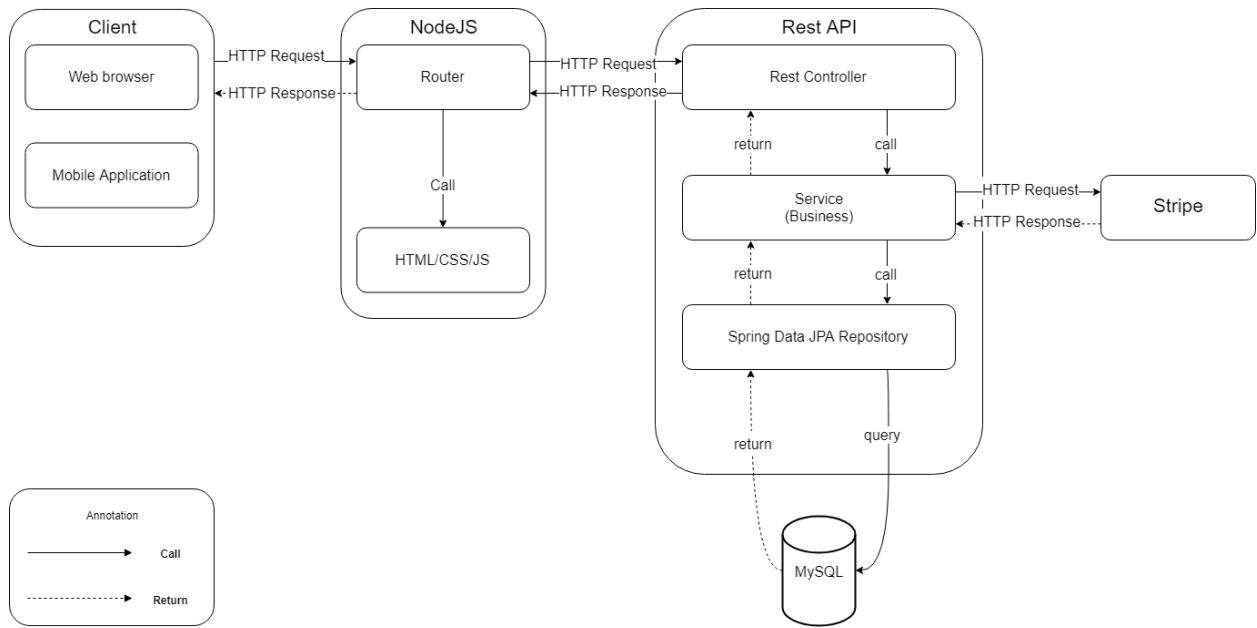


Figure 44 - Web application architecture description

3. Component Diagram

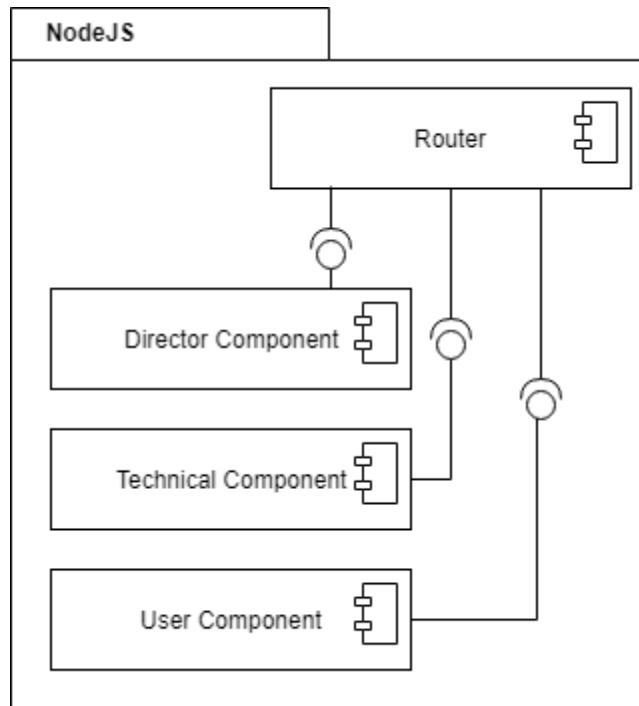


Figure 45 - Component Diagram

Component dictionary: Describes components	
Router	Handle request and response, accept input and convert it to commands for back-end.
Technical component	Handle technical activities in the system

User component	Handle user activities in the system
Director component	Handle director activities in the system

Table 42 - Describes components

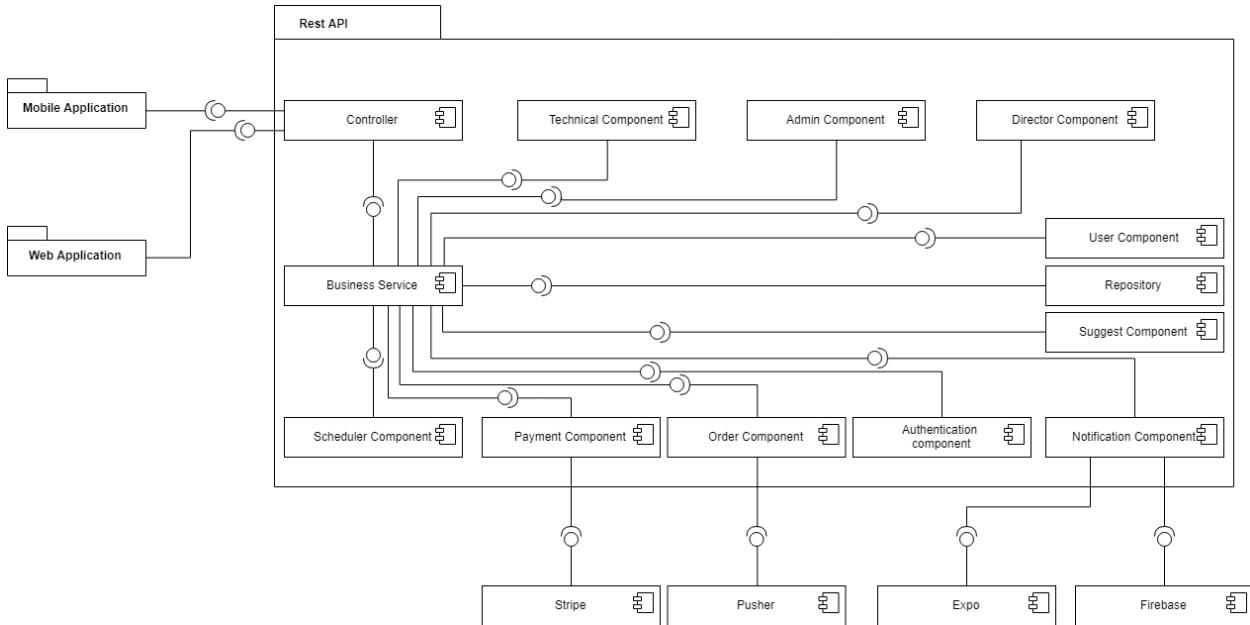


Figure 46 - Component Diagram

Component Dictionary: Describes components	
Web Application	Web application package
Mobile Application	Mobile application package
Stripe	Handle payment with Stripe API
Pusher	Handle real-time data with Pusher API
Expo	Handle notification with Expo API
Firebase	Handle notification with Firebase API
Controller	Handle requests and responses; accept input and convert it to commands for model and view.
Business Service	Handle system's business operators
Repository	Store data
Technical Component	Component to handle technical activities in the system
Admin Component	Component to handle admin activities in the system
Director Component	Component to handle director activities in the system
User Component	Component to handle user activities in the system
Suggest Component	Component to suggest product and promotion in the system
Notification Component	Component to handle notify in the system
Authentication Component	Component to handle authorize user in the system
Order Component	Component to handle orders in the system

Payment Component	Component to handle payment process
Scheduler Component	Component to handle process automatically from the system

Table 43 - Describes components

4. Detail Description

4.1 Class Diagram

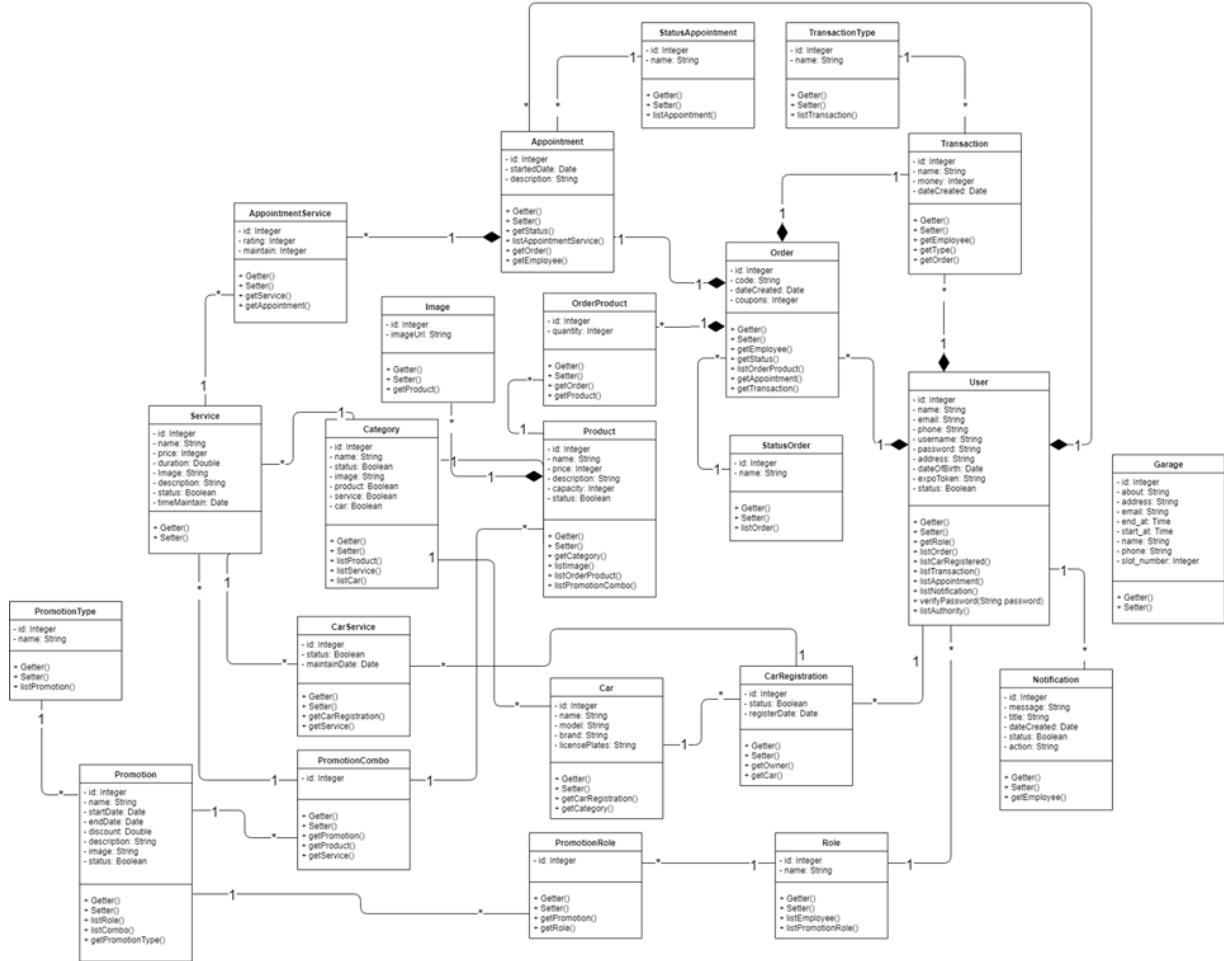


Figure 47 - Class diagram

4.2 Class Diagram Explanation

CLASS DICTIONARY: DESCRIBE CLASS		
Class Name	Mapping column with Conceptual diagram	Description
Appointment	Appointment	Not exist in the conceptual diagram. But needed in class.

AppointmentService	N/A	Not exist in the conceptual diagram. But needed in class.
Car	Car	Contain the car information.
Car Registration	N/A	Not exist in the conceptual diagram. But needed in class.
Category	Category	Contain the category information.
Garage	Garage	Contain the garage information.
Image	N/A	Not exist in the conceptual diagram. But needed in class.
Notification	N/A	Not exist in the conceptual diagram. But needed in class.
Order	Order	Contain the successfully order information.
Order Product	N/A	Contain the orders that have products.
Product	Product	Contain the product information.
Promotion	Promotion	Contain the promotion information.
PromotionCombo	N/A	Not exist in the conceptual diagram. But needed in class.
PromotionRole	N/A	Not exist in the conceptual diagram. But needed in class.
PromotionType	N/A	Not exist in the conceptual diagram. But needed in class.
Role	N/A	Not exist in the conceptual diagram. But needed in class.
Service	Service	Contain the service information.
StatusAppointment	N/A	Not exist in the conceptual diagram. But needed in class.
StatusOrder	N/A	Not exist in the conceptual diagram. But needed in class.
Transaction	Transaction	Contain the transaction information.
TransactionType	N/A	Not exist in the conceptual diagram. But needed in class.
Employee	Account	Contain the account information.

Table 44 - Class Diagram Explanation

4.2.1 Employee

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of an employee.
Name	String	Private	Store full name of an employee.
Email	String	Private	The email address of an employee.

Phone	String	Private	The telephone number of an employee.
Username	String	Private	The username of an employee that used to log in.
Password	String	Private	The password of an employee.
Address	String	Private	The address where an employee lives.
DateOfBirth	Date	Private	The date the employee was born.
ExpoToken	String	Private	The token generated when an employee uses a mobile app.
Status	Boolean	Private	The status of an employee (activate or deactivate)

Table 45 - <Class diagram attribute> Employee

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set value of attribute.
getRole	Role	Public	Get a role of an account.
listOrder	List	Public	Get list orders of an account.
listCarRegistered	List	Public	Get cars are registered by an account.
listTransaction	List	Public	Get list transactions of an account.
listAppointment	List	Public	Get list appointment of an account.
listNotifications	List	Public	Get list notifications of an account.
verifyPassword	Boolean	Public	Verify password of an account correct or not.
getAuthority	List	Public	Get list authorities of an account.

Table 46 - <Class diagram method> Employee

4.2.2 Appointment

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of an appointment.
StartedDate	Date	Private	The date starts to do the service.
Description	String	Private	Description of an appointment.

Table 47 - <Class diagram attribute> Appointment

Method	Redirect Type	Visibility	Description

Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set value of attribute.
getStatus	StatusAppointment	Public	Get status of an appointment.
listAppointmentService	List	Public	Get list appointment service of an appointment.
getOrder	Order	Public	Get order that an appointment belonged to.
getEmployee	Employee	Public	Get a technical that assigned to serve the appointment.

Table 48 - <Class diagram method> Appointment

4.2.3 AppointmentService

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of an appointment service.
Rating	Integer	Private	The date starts to do the service.

Table 49 - <Class diagram attribute> AppointmentService

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
Service	Service	Public	The services belonged to an appointment.
Appointment	Appointment	Public	The appointment that an appointment service belonged to.

Table 50 - <Class diagram method> AppointmentService

4.2.4 Car

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of a car.
Name	String	Private	Car's name.
licensePlates	String	Private	Car's license plates.
Model	String	Private	Model's car.

Brand	String	Private	Brand's car.
--------------	--------	---------	--------------

Table 51 - <Class diagram attribute> Car

Method	Redirect Type	Visibility	Description
Getter	Attribute Type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
listCarRegistered	List	Public	Get list cars registered.
getCategory	Category	Public	Get category a car belongs to.

Table 52 - <Class diagram method> Car

4.2.5 CarRegistration

Attribute	Type	Visibility	Description
Id	Int	Private	Unique identifier of car registered.
registeredDate	Date	Private	The date account's car registered.
Status	Boolean	Private	Status of a car.

Table 53 - <Class diagram attribute> CarRegistration

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
getEmployee	Employee	Public	Get the owner of a car.

Table 54 - <Class diagram method> CarRegistration

4.2.6 Category

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of a category.
Name	String	Private	Store name of category.
Car	Boolean	Private	Check a category belongs to the car or not.
Product	Boolean	Private	Verify a category belongs to the product or not.
Service	Boolean	Private	Check a category belongs to the service or not.
Image	String	Private	The path of the image stored in the Azure cloud.
Status	Boolean	Private	Status of a category.

Table 55 - <Class diagram attribute> Category

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.

getListProduct	List	Public	Get the list of products if the category is a kind of product.
getListService	List	Public	Get a list of services if the category is a kind of service.
getListCar	List	Public	Get a list of cars if the category is a kind of car.

Table 56 - <Class diagram method> Category

4.2.7 Garage

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of a garage.
Name	String	Private	Store the name of a garage.
Email	String	Private	The email address of a garage.
Phone	String	Private	The telephone number of a garage.
Address	String	Private	The address where a garage stays.
slotNumber	Integer	Private	The number a garage can handle at a time.
startAt	String	Private	The time when a garage starts to work.
endAt	String	Private	The time when a garage stops working.
About	String	Private	The description of a garage.

Table 57 - <Class diagram attribute> Garage

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.

Table 58 - <Class diagram method> Garage

4.2.8 Image

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of an image
imageUrl	String	Private	The path of an image that stored at Azure cloud.

Table 59 - <Class diagram attribute> Image

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.

Table 60 - <Class diagram method> Image

4.2.9 Notification

Attribute	Type	Visibility	Description

Id	Integer	Private	Unique identifier of a notification.
Message	String	Private	The content message of the notification.
Title	String	Private	The title of the notification.
dateCreated	Date	Private	The date that a notification was generated.
Status	Boolean	Private	The status of a notification. (read or unread)
Action	String	Private	The screen or page that navigated by the notification.

Table 61 - <Class diagram attribute> Notification

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
getEmployee	Employee	Public	The account has received the notification.

Table 62 - <Class diagram method> Notification

4.2.10 Order

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of an order.
Code	String	Private	Unique identifier of an order that shows with the user.
dateCreated	Date	Private	The date of the order was created.

Table 63 - <Class diagram attribute> Order

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
getEmployee	Employee	Public	Get an employee that was assigned for serving the order.
Status	StatusOrder	Public	Get status of an order.
listOrerProduct	List	Public	Get a list of products stored in order.
Appointment	Appointment	Public	Get an appointment that stored the date when starting the service.
Transaction	Transaction	Public	Set a transaction when the order was paid.

Table 64 - <Class diagram method> Order

4.2.11 OrderProduct

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of order that store product.
Quantity	Integer	Private	The quantity of a product.

Table 65 - <Class diagram attribute> OrderProduct

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
getProduct	Product	Public	Get a product that belongs to an order.
getOrder	Order	Public	Get order that contains this order of the product.

Table 66 - <Class diagram method> OrderProduct

4.2.12 Product

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of a product.
Name	String	Private	Store the name of a product.
Price	String	Private	The price of a product.
Description	String	Private	Describe how a product is.
Capacity	Integer	Private	The capacity of a product if products are oil.
Status	Boolean	Private	Status of a product. (activate or deactivated)

Table 67 - <Class diagram attribute> Product

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
getCategory	Category	Public	Get the category of a product.
getListImage	List	Public	Get list image that describes a product.
listOrderProduct	List	Public	Get a list of products that belongs to the.
listPromotionCombo	List	Public	Get a list of promotion with a status combo that the product belongs to.

Table 68 - <Class diagram method> Product

4.2.13 Promotion

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of a promotion.
Name	String	Private	The name of a promotion.
startDate	Date	Private	The date starts to run the promotions.
endDate	Date	Private	The date the promotion has been stopped.
Discount	Double	Private	The percent that order applies.
Description	String	Private	The description of a promotion.
Status	Boolean	Private	The status of a promotion (activate or deactivate).
Image	String	Private	The path of an image stored in the Azure cloud.

Table 69 - <Class diagram attribute> Promotion

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
getPromotionType	PromotionType	Public	Get type of promotion (combo or single)
listCombo	List	Public	Get a list of promotions that have typed is a combo.
listRole	List	Public	Get a list of promotions that have a different role.

Table 70 - <Class diagram method> Promotion

4.2.14 PromotionCombo

Attribute	Type	Visibility	Description
Id	Int	Private	Unique identifier of a promotion that belongs to a combo type.

Table 71 - <Class diagram attribute> Promotion Combo

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
getPromotion	Promotion	Public	Get a list of promotions that belong to a combo type.
getProduct	Product	Public	Get a product of promotions that belongs to a combo type.

getService	Service	Public	Get service of promotions that belongs to a combo type.
-------------------	---------	--------	---

Table 72 - <Class diagram method> PromotionCombo

4.2.15 PromotionRole

Attribute	Type	Visibility	Description
Id	Int	Private	Unique identifier of a promotion with a specific role.

Table 73 - <Class diagram attribute> Promotion role

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
getPromotion	Promotion	Public	Get a promotion with a specific role.
getRole	Role	Public	Get a role that uses the promotion.

Table 74 - <Class diagram method> Promotion role

4.2.16 PromotionType

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of the type of promotion.
Name	String	Private	The name of promotion type.

Table 75 - <Class diagram attribute> Promotion type

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
listPromotion	List	Public	Get list promotions that belong to a role.

Table 76 - <Class diagram method> Promotion type

4.2.17 Role

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of a role.
Name	String	Private	Store the name of a role.

Table 77 - <Class diagram attribute> Role

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.

Setter	Void	Public	Set attribute value.
listEmployee	List	Public	Get a list of employees that belongs to a specific role.
listPromotionRole	List	Public	Get a list of promotion with a specific role.

Table 78 - <Class diagram method> Role

4.2.18 Service

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of a service.
Name	String	Private	Store the name of a service.
Price	Integer	Private	The price of a service.
Duration	Double	Private	The duration that shows how long the service does.
Image	String	Private	The path of the image stored in the Azure cloud.
Description	String	Private	The description that describes how a service is.
Status	Boolean	Private	The status of a service (activate or deactivate).
maintainDate	Date	Private	The service warranty date.

Table 79 - <Class diagram attribute> Service

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
listPromotionCombo	List	Public	Get a list of promotions that belong to a combo type.
listAppointmentService	List	Public	Get a list of appointments that show info of an appointment.

Table 80 - <Class diagram method> Service

4.2.19 StatusAppointment

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of the status of an appointment.
Name	String	Private	The name of the status.

Table 81 - <Class diagram attribute> StatusAppointment

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
listAppointment	List	Public	Get a list of appointments that show info of an appointment.

Table 82 - <Class diagram method> StatusAppointment

4.2.20 StatusOrder

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of status of an order.
Name	String	Private	Store the name of the status.

Table 83 - <Class diagram attribute> StatusOrder

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
listOrder	List	Public	Get a list of orders that belong to a specific status.

Table 84 - <Class diagram method> StatusOrder

4.2.21 Transaction

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of a transaction.
Name	String	Private	Store the name of a transaction.
Money	Integer	Private	The money of a transaction.
dateCreated	Date	Private	The date when a transaction was created.

Table 85 - <Class diagram attribute> Transaction

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
getEmployee	Employee	Public	Get an account information that owns the transaction.
getType	TransactionType	Public	Get the type of transaction.
getOrder	Order	Public	Get an order that belongs to a transaction.

Table 86 - <Class diagram method> Transaction

4.2.22 TransactionType

Attribute	Type	Visibility	Description
Id	Integer	Private	Unique identifier of an appointment
Name	String	Private	Contains the name of the type of transaction.

Table 87 - <Class diagram attribute> TransactionType

Method	Redirect Type	Visibility	Description
Getter	Attribute type	Public	Get attribute value.
Setter	Void	Public	Set attribute value.
listTransaction	List	Public	Get a list of transactions that belong to a specific type.

Table 88 - <Class diagram method> TransactionType

4.3 Interaction Diagram

4.3.1 Login

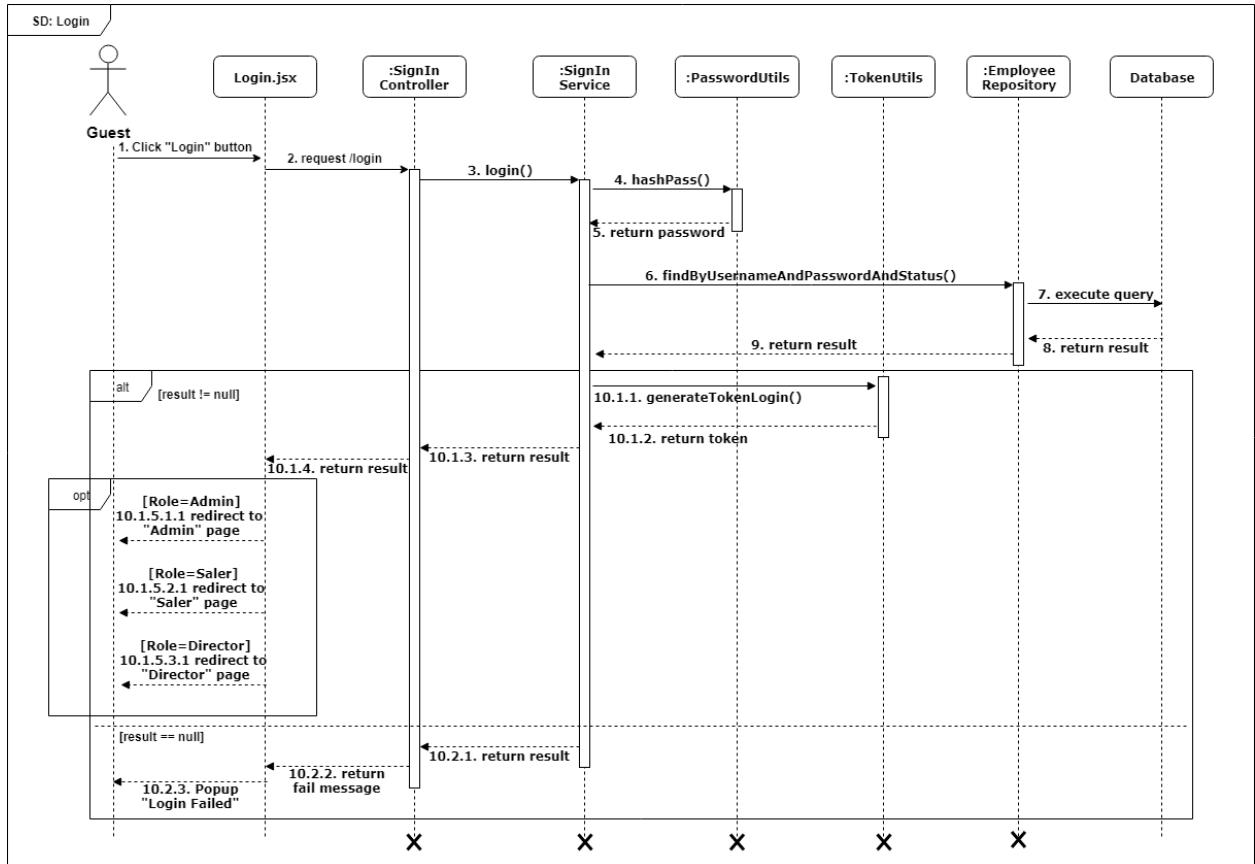


Figure 48 - <Sequence Diagram> Login

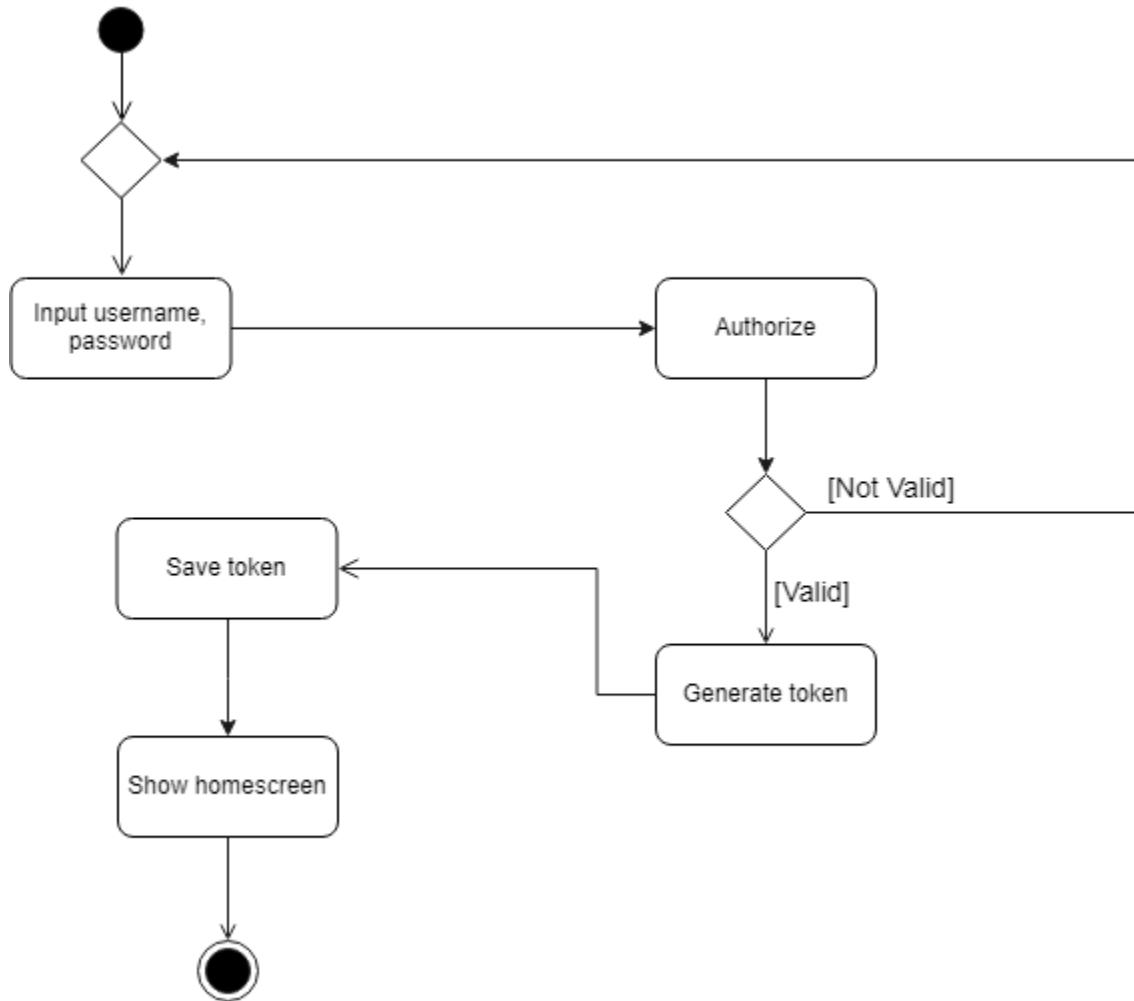


Figure 49 - <Activity Diagram> Login

4.3.2 Add new Order

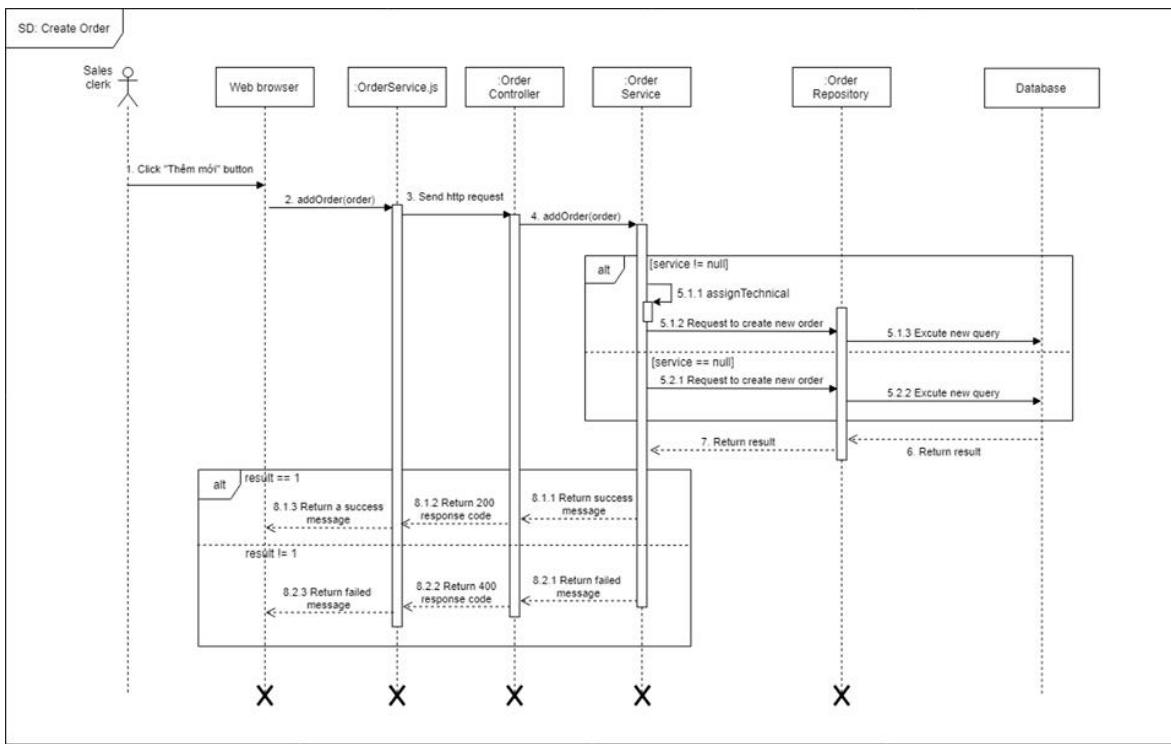


Figure 50 - <Sequence Diagram>Add new order

4.3.3 Update an Order

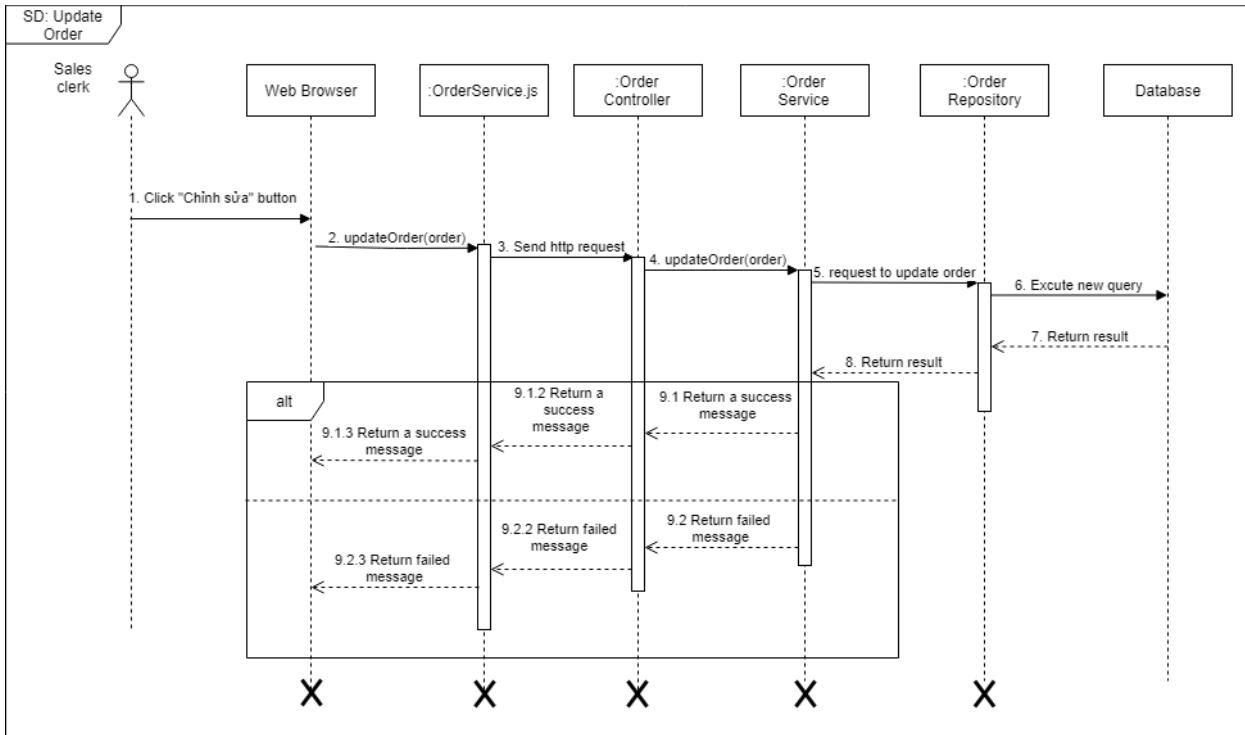


Figure 51 - <Sequence Diagram> Update order

4.3.4 Add new Category

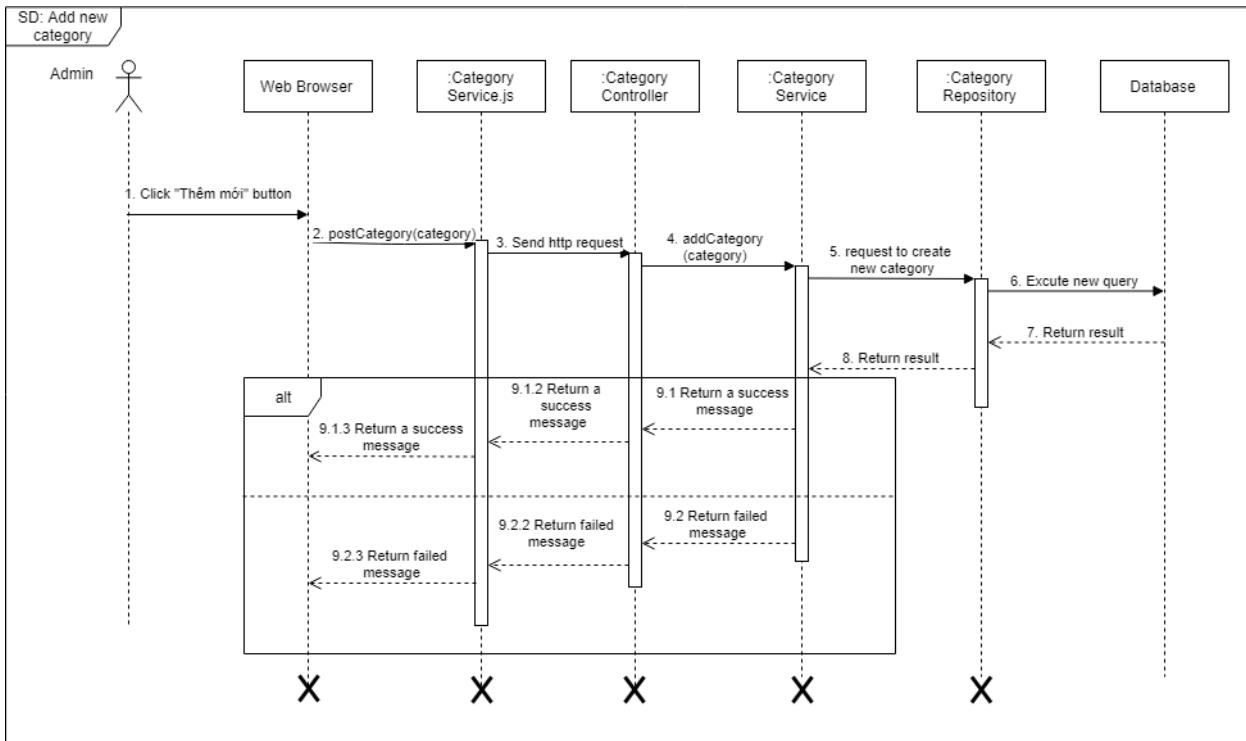


Figure 52 - <Sequence Diagram> Add new category

4.3.5 Update Category

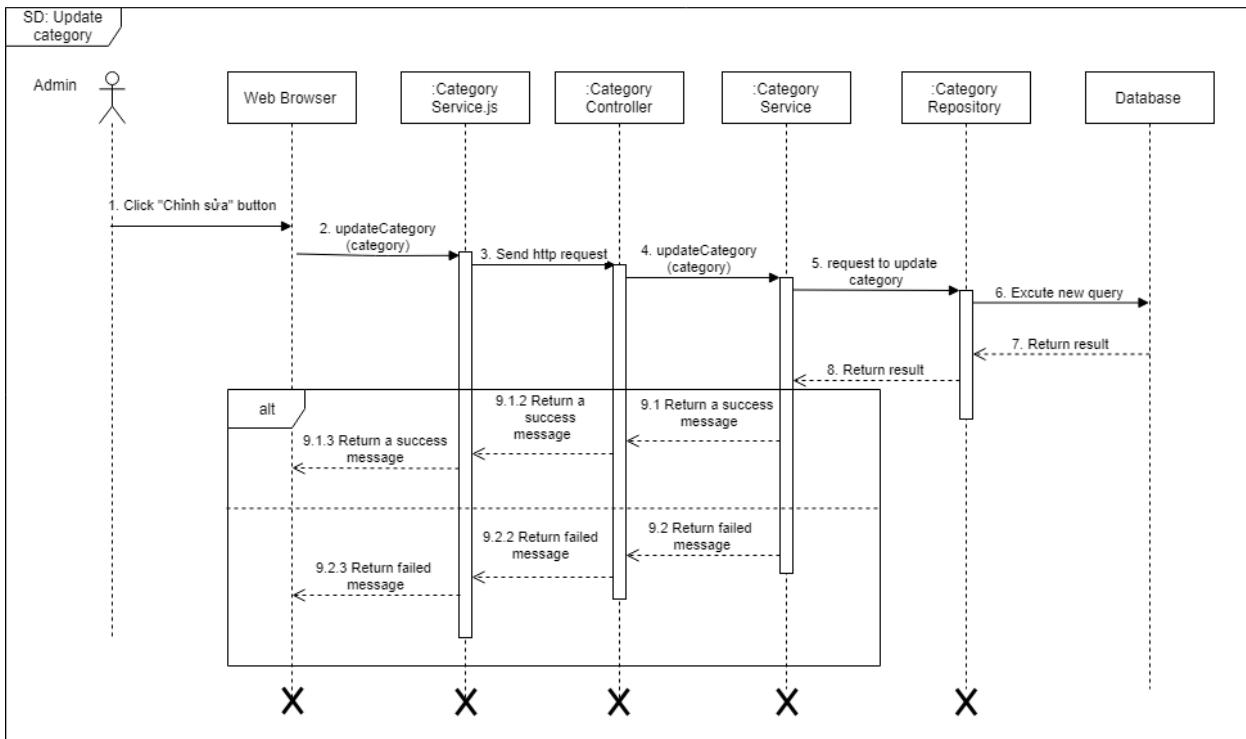


Figure 53 - <Sequence Diagram> Update category

4.3.6 Add new Service

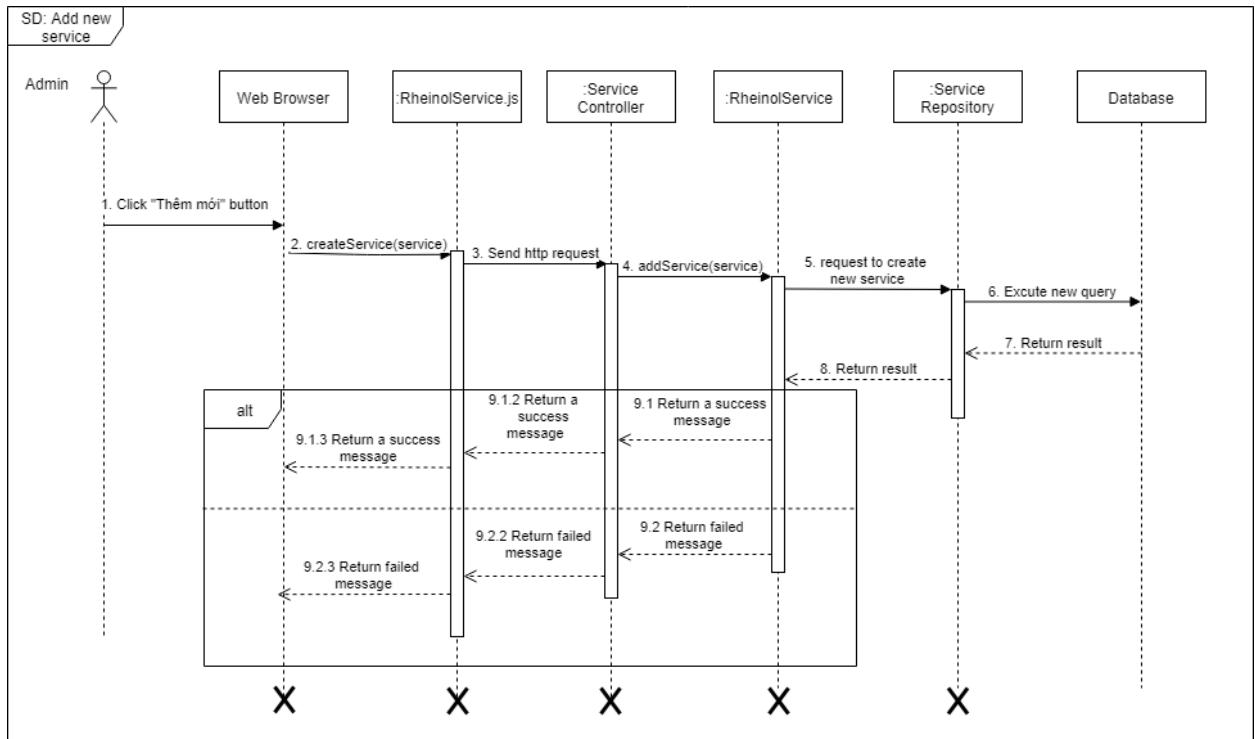


Figure 54 - <Sequence Diagram> Add new service

4.3.7 Update a Service

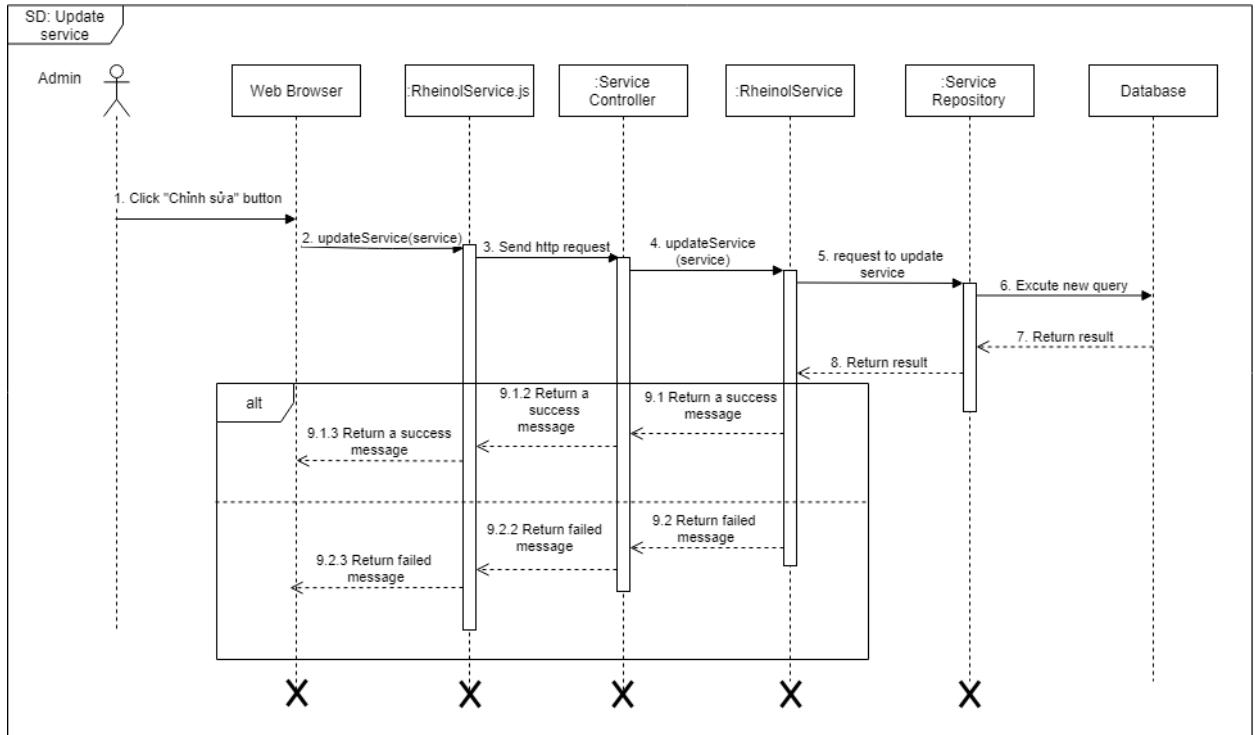


Figure 55 - <Sequence Diagram> Update service

4.3.8 Add new Product

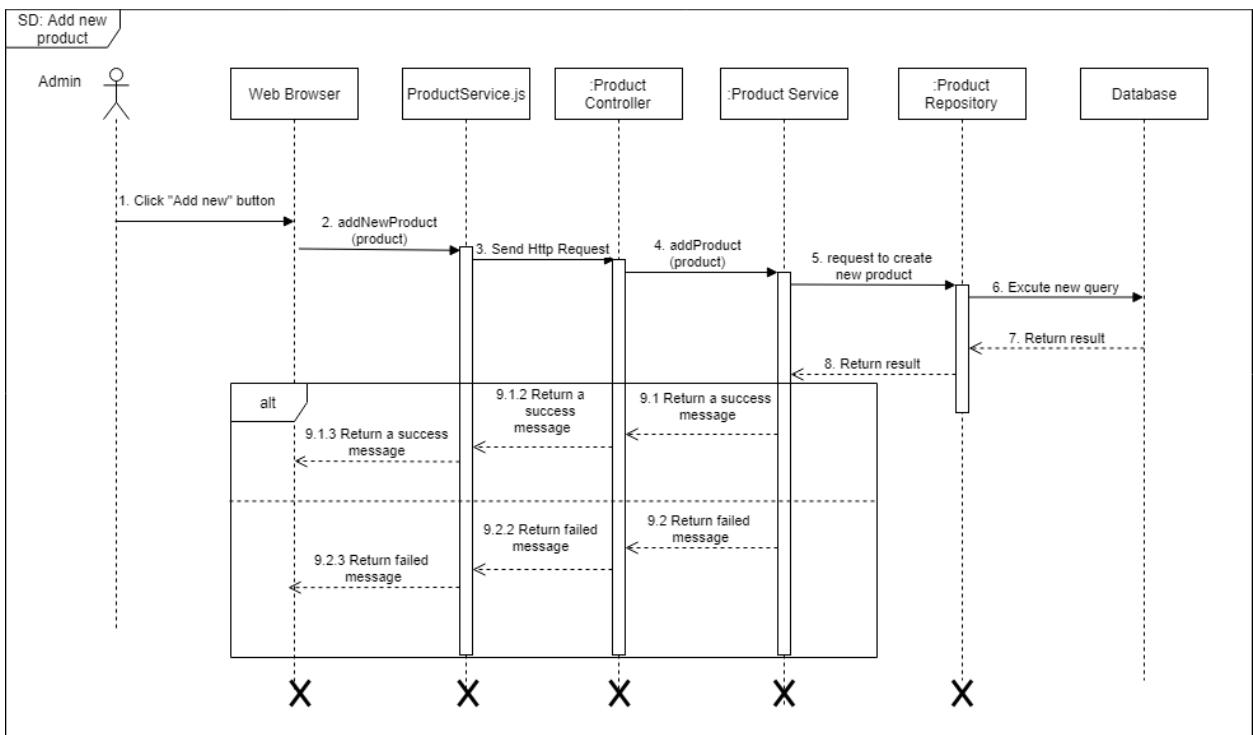


Figure 56 - <Sequence Diagram> Add new product

4.3.9 Update a product

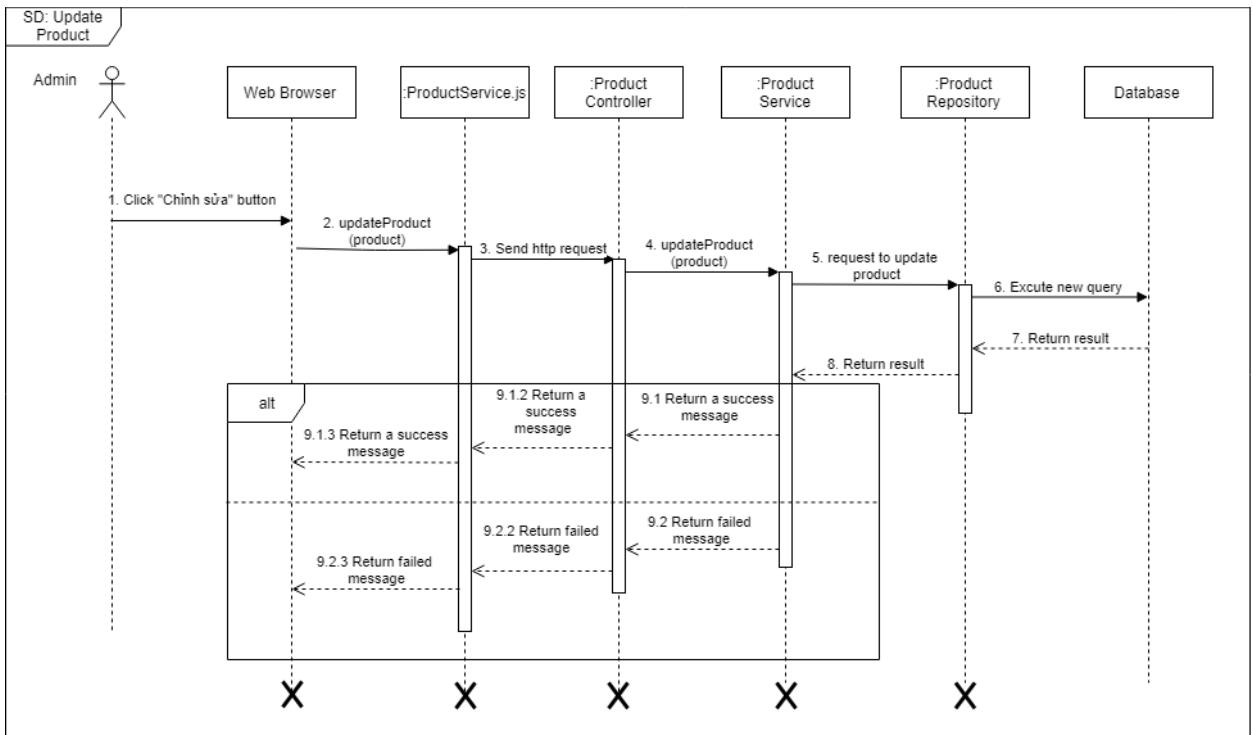


Figure 57 - <Sequence Diagram> Update product

4.3.10 Add new an Account

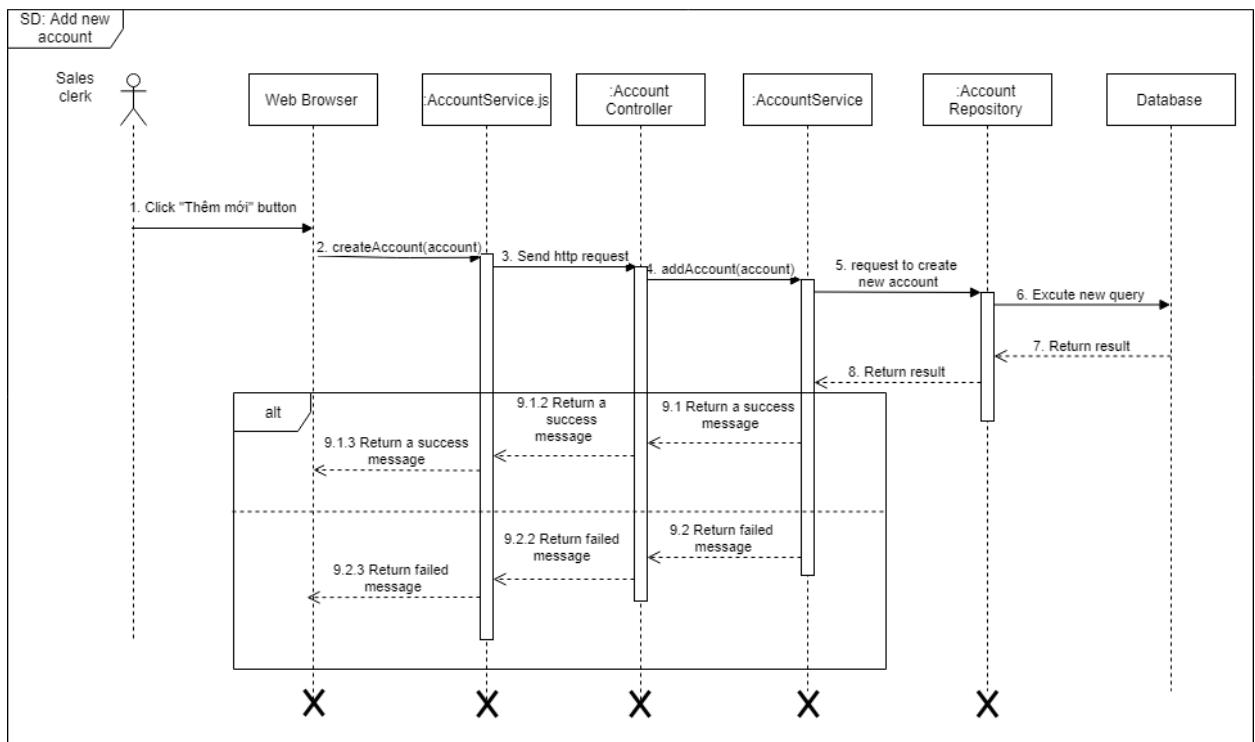


Figure 58 -- <Sequence Diagram> add new account

4.3.11 Add new a Promotion

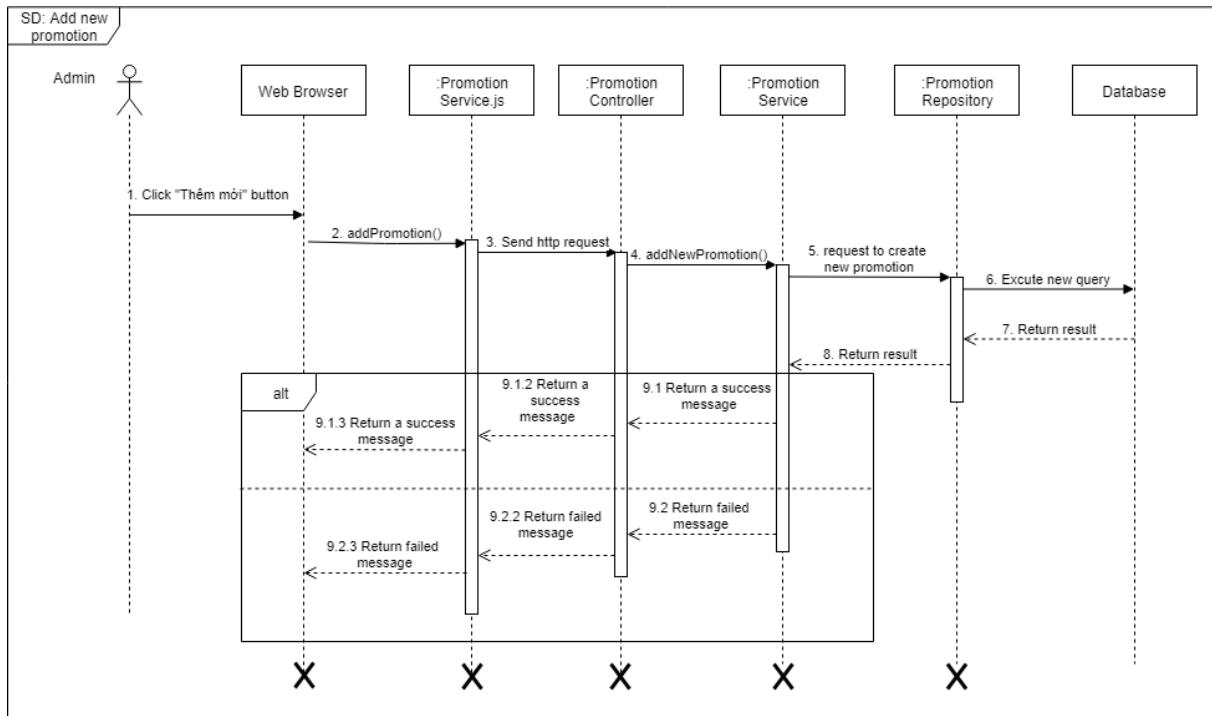


Figure 59 - <Sequence Diagram> Add new promotion

4.3.12 Filter hot product

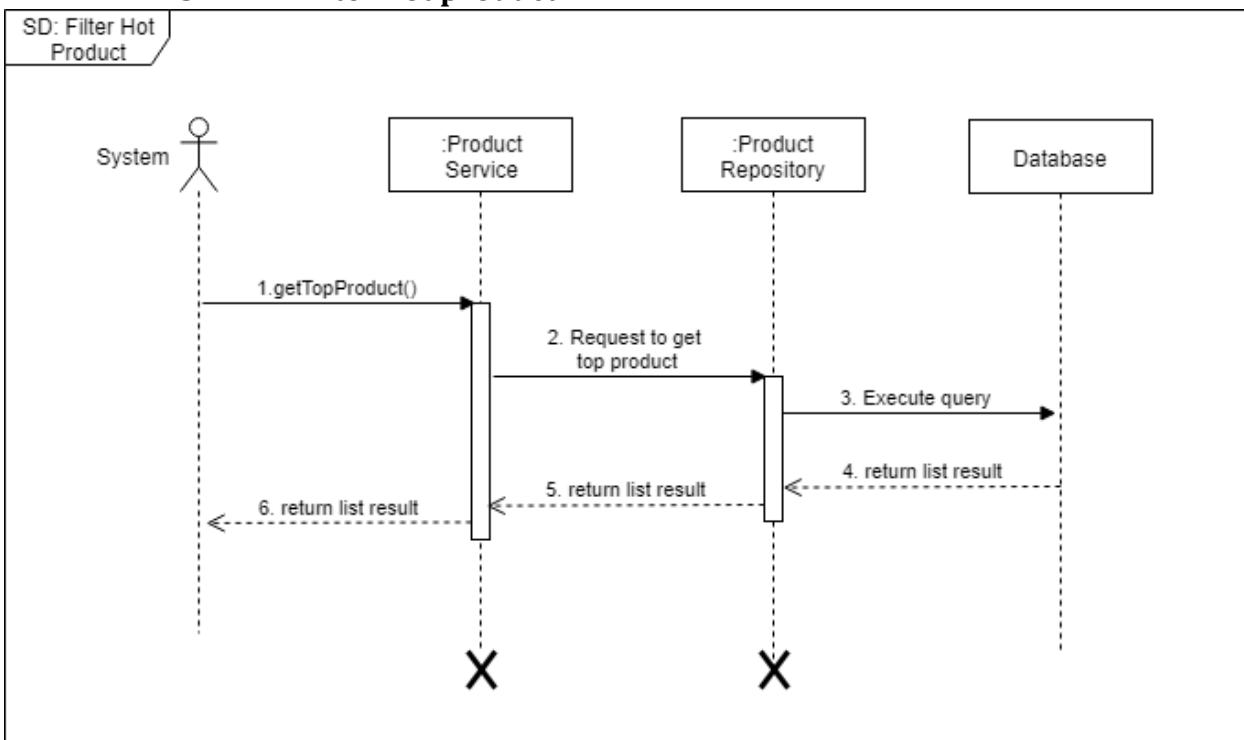


Figure 61 - <Sequence Diagram> Filter hot product

4.3.13 Filter hot service

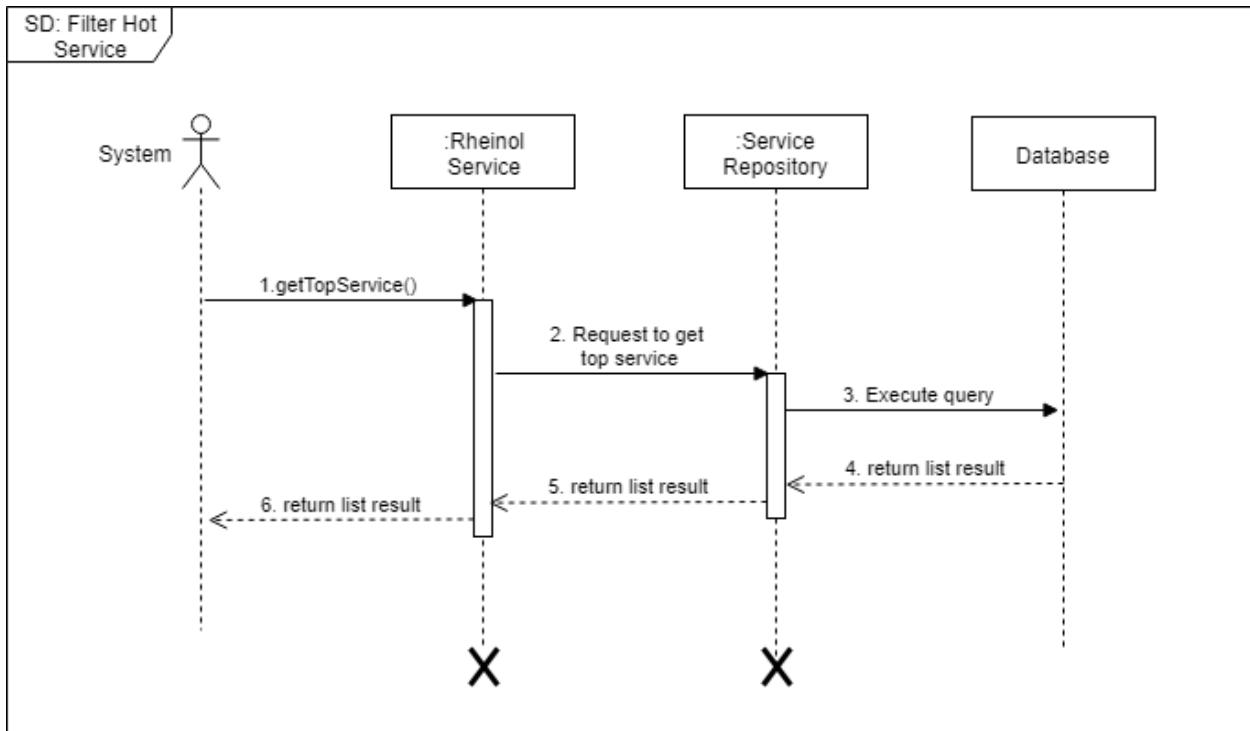


Figure 60 - <Sequence Diagram> Filter hot service

4.3.14 Notify customer

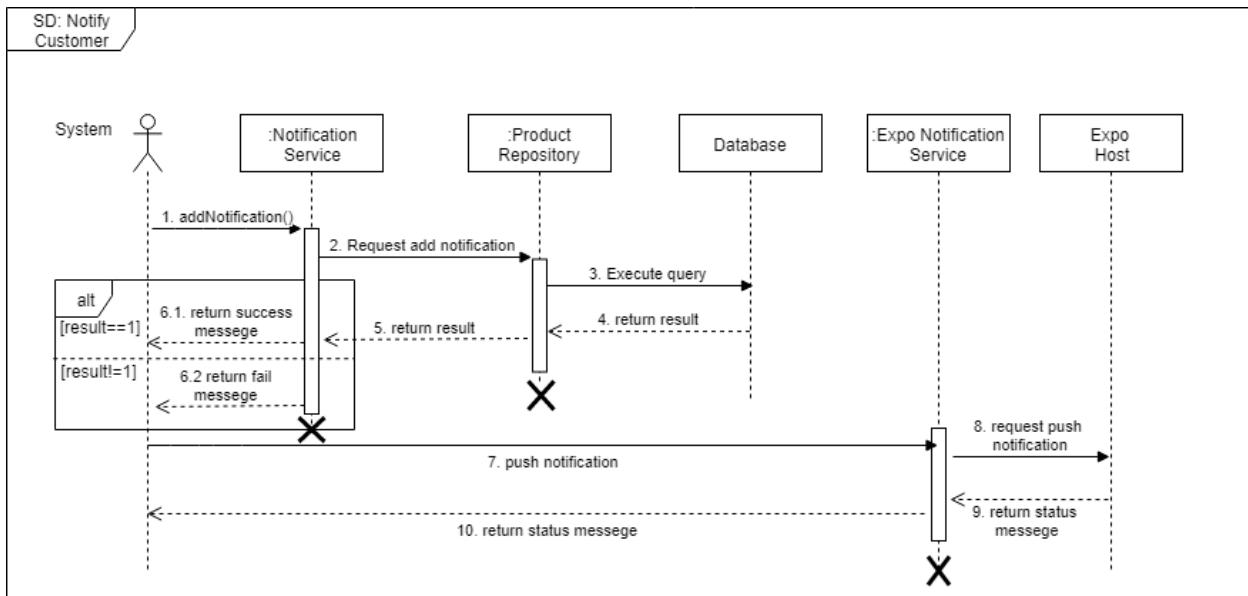


Figure 62 - <Sequence Diagram> Notify customer

4.3.15 Filter order timeout

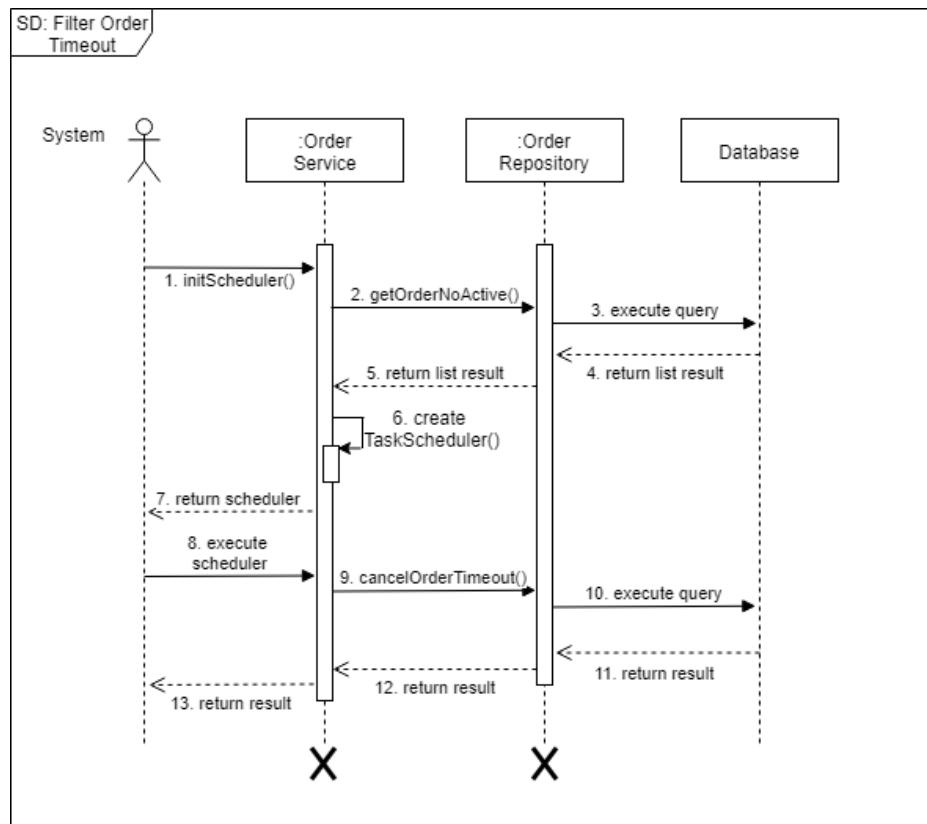


Figure 63 - <Sequence Diagram> Filter order timeout

4.3.16 Register card

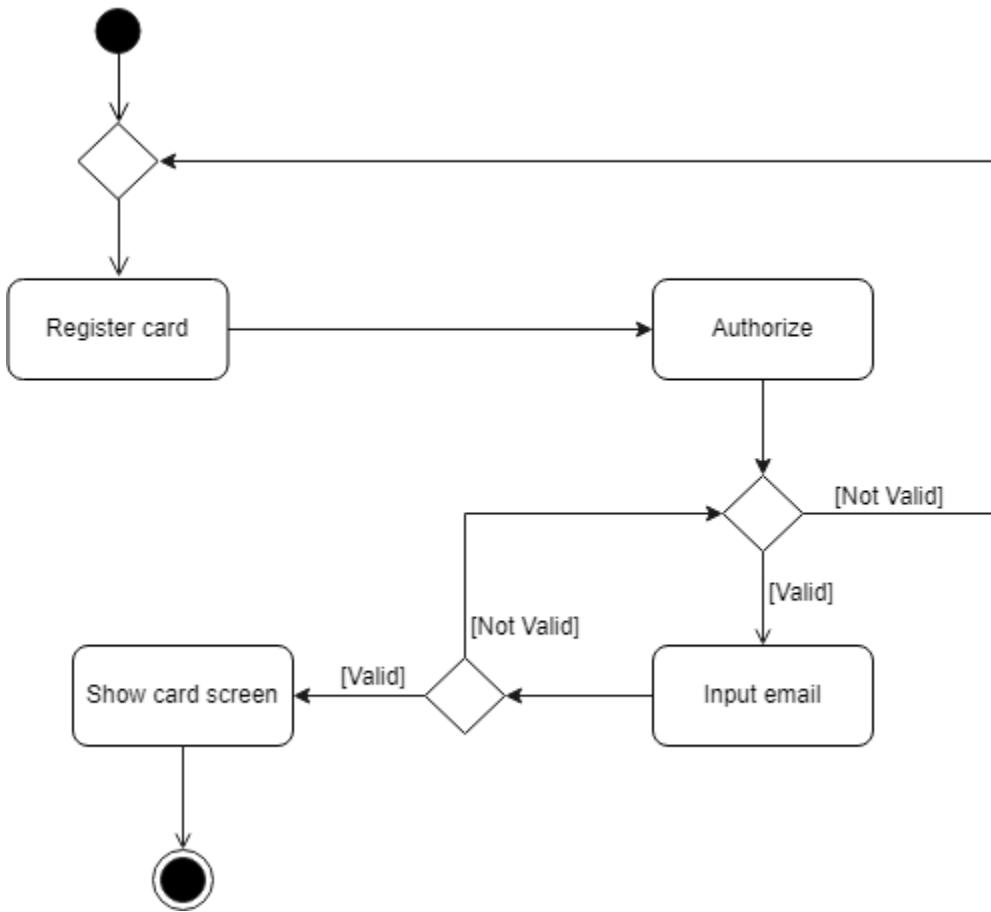


Figure 64 - <Activity Diagram> Register card

4.3.17 Payment order

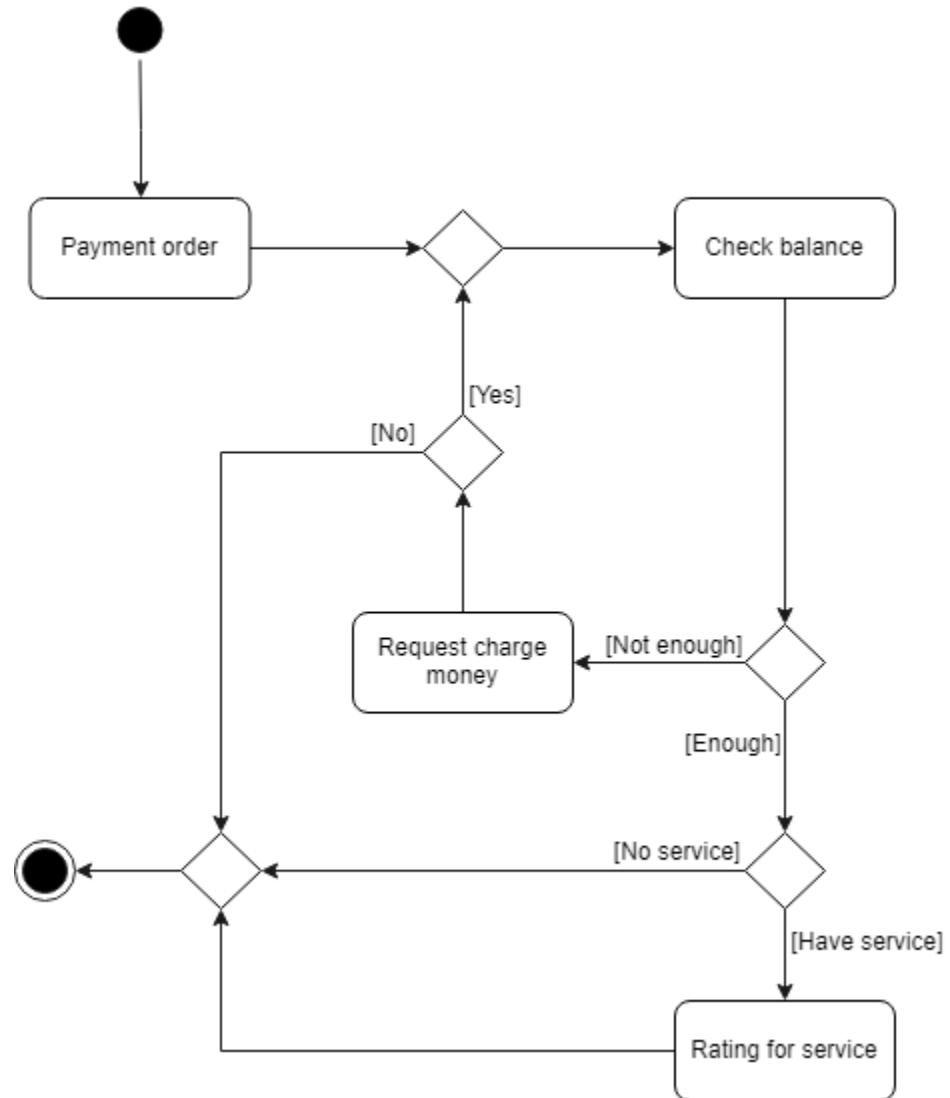


Figure 65 - <Activity Diagram> Payment order

4.3.18 Book service

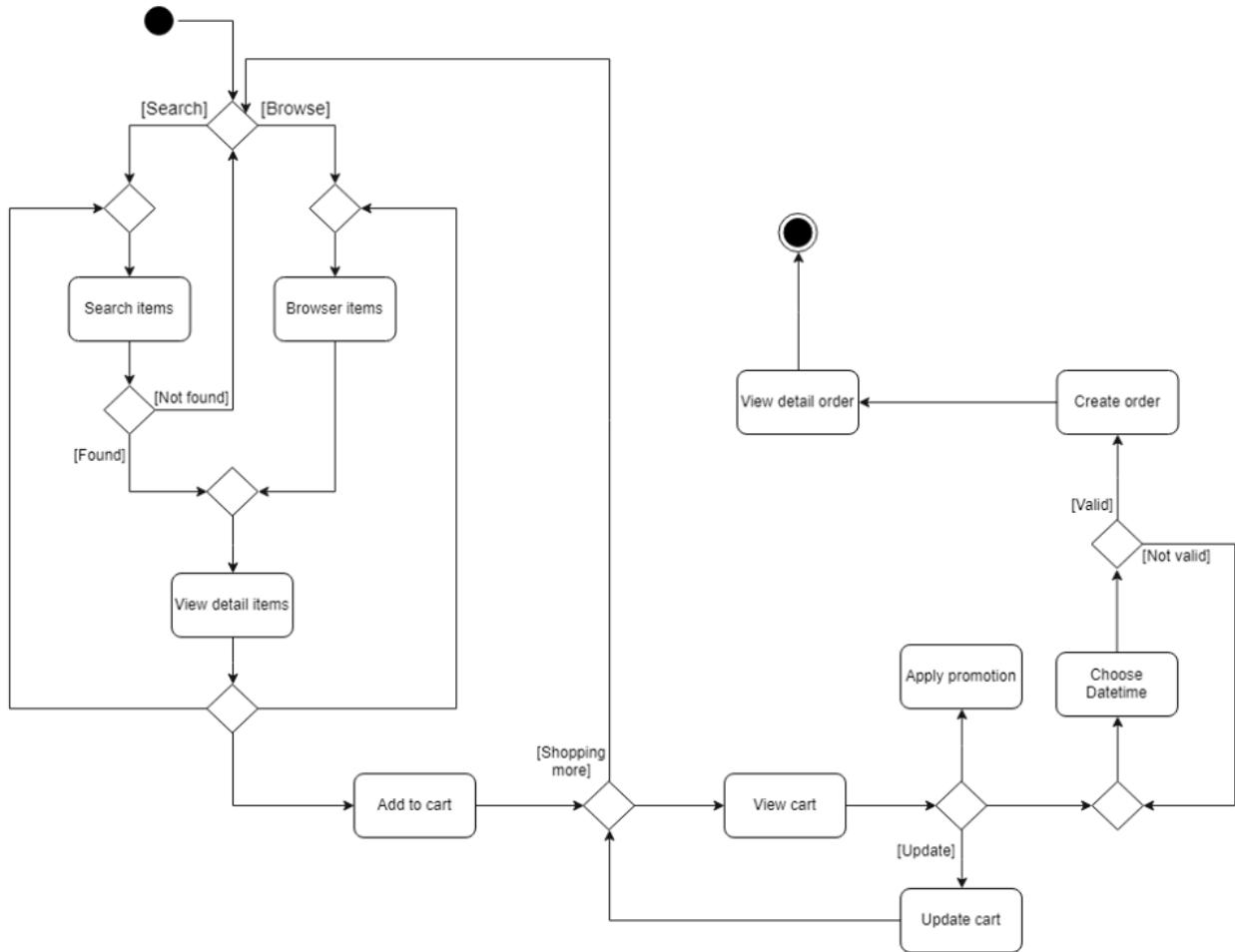


Figure 66 - <Activity Diagram> Book service

4.3.19 Buy product

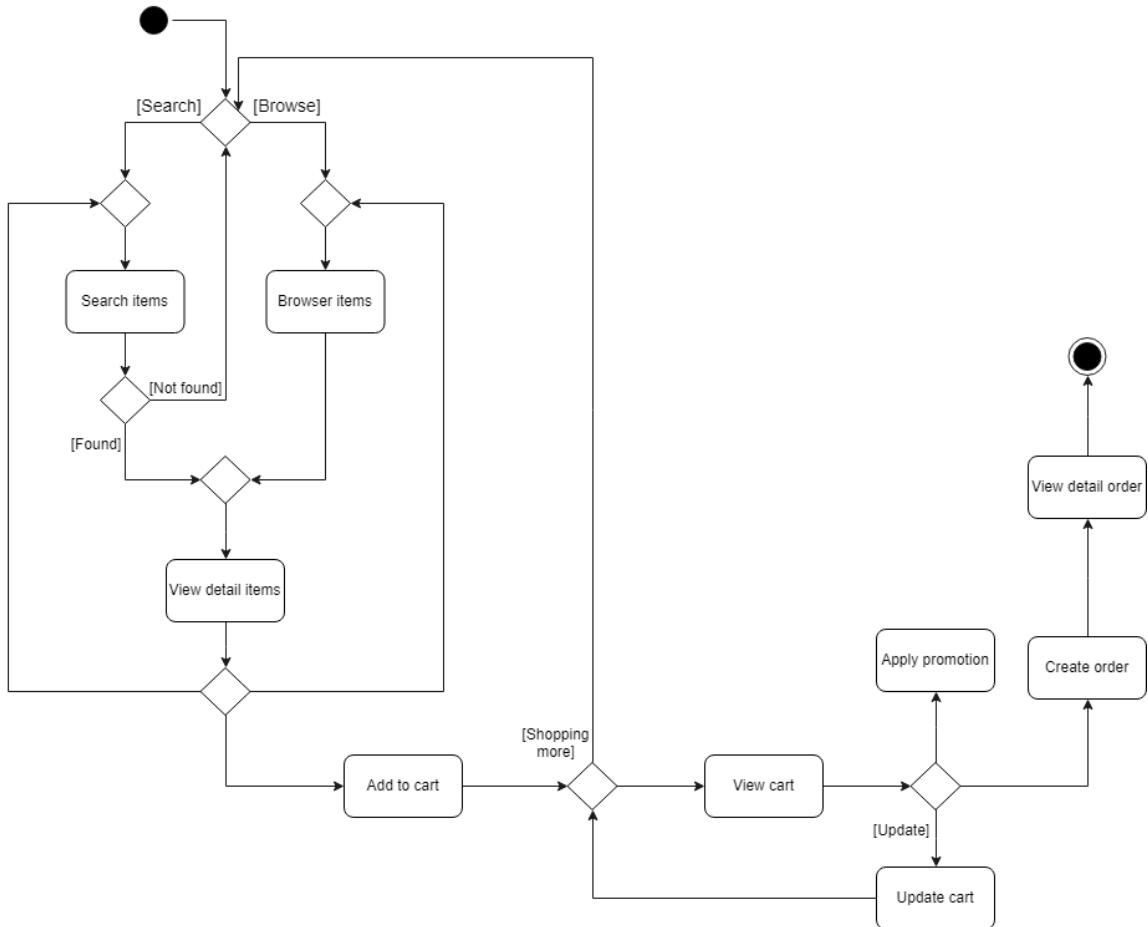


Figure 67 - <Activity diagram> Buy product

4.3.20 Register account

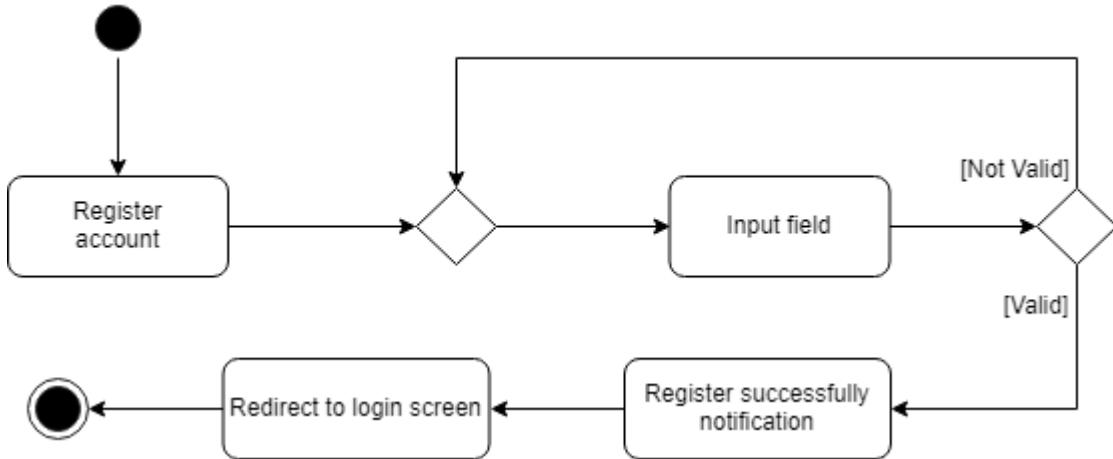


Figure 68 - <Activity Diagram> Register account

4.3.21 Edit account

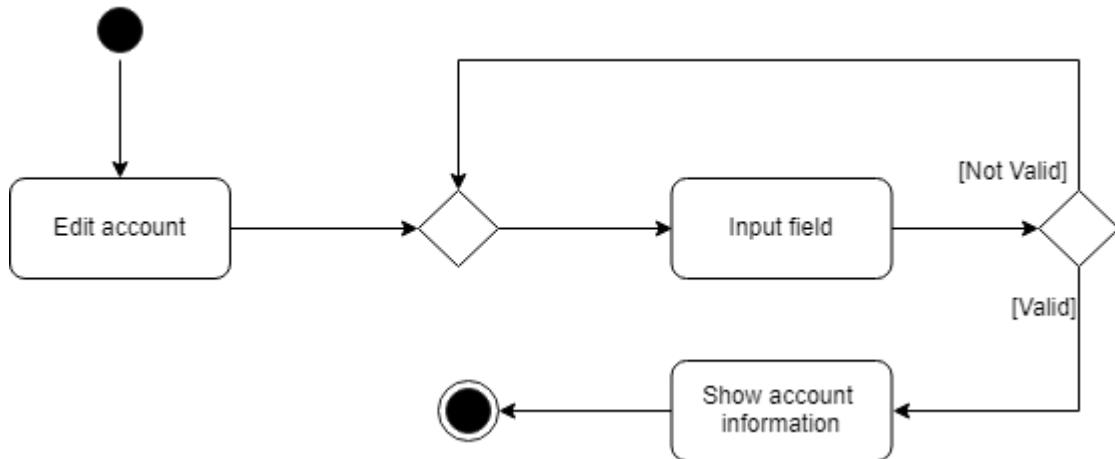


Figure 69 - <Activity Diagram> Edit account

5. Interface

5.1 User Interface Design

5.1.1 Web application Screen

5.1.1.1 Login

Đăng nhập

Tài khoản
1 nobino27

Mật khẩu
2
3 ĐĂNG NHẬP

Figure 70 - <Interface> Login

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
----	------------	-------------	-----------	-----------	--------------	-----------

1	Tài khoản	User name to login	No	Yes	Textbox	String
2	Mật khẩu	Password to login	No	Yes	Textbox	String

Table 89 - <Fields> Login

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
3	Đăng nhập	Login into the system	Required	Textbox

Table 90 - <Button> Login

5.1.1.2 Menu Admin

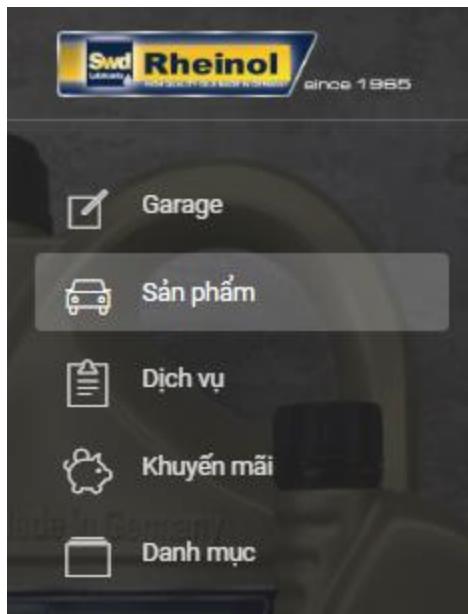


Figure 71 - <Interface> Menu admin

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
1	Garage	Admin can fill garage's information	User must had been authorized by Admin role	Redirect to garage page
2	Sản phẩm	Admin can manage products	User must had been authorized by Admin role	Redirect to product page
3	Dịch vụ	Admin can manage services	User must had been authorized by Admin role	Redirect to service page

4	Khuyến mãi	Admin can manage promotions	User must had been authorized by Admin role	Redirect to promotion page
5	Danh mục	Admin can manage categories	User must had been authorized by Admin role	Redirect to category page

Table 91 - <Button> Menu admin

5.1.1.3 Manage Product

The screenshot shows a web-based application for managing products. On the left, there's a sidebar with menu items: Garage (unchecked), Sản phẩm (checked), Dịch vụ, Khuyến mãi, and Danh mục. The main area has a header 'Sản phẩm' (1) with buttons '+Thêm mới' (2) and '+Import Excel'. Below is a search bar 'Tim kiem sản phẩm' (3) with a magnifying glass icon. A dropdown menu 'Danh sách sản phẩm' (4) is open, showing 'Dầu xe oto' as the selected category. The main content area displays a table of products (5) with columns: STT (6), TÊN SẢN PHẨM (7), GIÁ (VND) (8), LOẠI DỊCH VỤ (9), TRANG THÁI (10), and THAO TÁC (11). The table contains five rows of oil products. At the bottom, there are navigation buttons (12) and a page number indicator (13).

Figure 72 - <Interface> Manage product

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
3	Tìm kiếm	Admin can search all products in system	No	Yes	Textbox	String
4	Danh mục	Products are classified by category.	Yes	Yes	Combo Box	String
5	STT	Order number of products	Yes	Yes	Text	String

6	Tên sản phẩm	Product's name	Yes	Yes	Text	String
7	Giá	Product's price	Yes	Yes	Text	String
8	Loại sản phẩm	Product's type	Yes	Yes	Text	String
9	Trạng thái	Product's status	Yes	Yes	Text	boolean

Table 92 - <Fields> Manage product

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
1	Thêm mới	Admin can add new a product.	N/A	Navigate to add new page.
2	Import excel	List product is added by excel file	N/A	Select file pop up is shown.
11	Chi tiết	Admin can view product's detail	N/A	The modal is shown.
12	Sửa	Admin can edit the product	N/A	Navigate to edit page.
13	Trạng thái	Admin can change product's status	N/A	The modal is shown.

Table 93 - <Button> Manage product

5.1.1.4 Manage service

The screenshot shows a web-based application interface for managing services. On the left, there is a sidebar with navigation links: Garage, Sản phẩm, Dịch vụ (highlighted), Khuyến mãi, and Danh mục. The main area has a header 'Dịch vụ' and a search bar 'Tìm kiếm dịch vụ'. Below the header, there is a button '+ Thêm dịch vụ' (marked with red number 1). A dropdown menu 'Danh sách dịch vụ' is open, showing a list of services with columns: STT (marked with red number 4), TÊN DỊCH VỤ (marked with red number 5), GIÁ (VNĐ) (marked with red number 6), THỜI GIAN (marked with red number 7), LOẠI DỊCH VỤ (marked with red number 8), TRANG THÁI (marked with red number 9), and THAO TÁC (marked with red numbers 10, 11, 12). The services listed are: Phú bảo vệ động cơ (Hoạt động), Ốc xả nhớt (Hoạt động), and Lọc nhớt (Vô hiệu hóa). At the bottom, there are navigation buttons for the list: '<' (prev), '1' (current page), and '>' (next).

Figure 73 - <Interface> Manage service

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
2	Tìm kiếm	Admin can search all services in system	No	Yes	Textbox	String
3	Danh mục	Services are classified by category.	Yes	Yes	Combo Box	String
4	STT	Order number of services	Yes	Yes	Text	String
5	Tên dịch vụ	Service's name	Yes	Yes	Text	String
6	Giá	Service's price	Yes	Yes	Text	String
7	Thời gian	Time of service	Yes	Yes	Text	String
8	Loại dịch vụ	Service's type	Yes	Yes	Text	String
9	Trạng thái	Service's status	Yes	Yes	Text	boolean

Table 94 - <Fields> Manage service

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
1	Thêm mới	Admin can add new a service.	N/A	Navigate to add new page.
10	Chi tiết	Admin can view service's detail	N/A	The modal is shown.
11	Sửa	Admin can edit the service	N/A	Navigate to edit page.
12	Trạng thái	Admin can change service's status	N/A	The modal is shown.

Table 95 - <Button> Manage service

5.1.1.5 Manage promotion

The screenshot shows a software interface for managing promotions. On the left is a sidebar with icons for Garage, Sản phẩm (Products), Dịch vụ (Services), Khuyến mãi (Promotions), and Danh mục (Categories). The main area has a header 'Khuyến mãi' and a search bar 'Tìm kiếm khuyến mãi'. A blue button '+ Thêm mới' is highlighted with a red number '1'. Below is a table titled 'Danh sách khuyến mãi' with columns numbered 3 to 12:

STT	TÊN	GIẢM GIÁ	NGÀY BẮT ĐẦU	NGÀY KẾT THÚC	THẺ LOẠI	TRẠNG THÁI	THAO TÁC
1	Ngày thương binh liệt sĩ	10 %	18-07-2019	03-08-2019	Đơn lẻ	Hoạt động	Xem Nhắc Xoá
2	Mừng tết Trung thu	20 %	01-08-2019	31-08-2019	Đơn lẻ	Hoạt động	Xem Nhắc Xoá
3	Ngày thương binh liệt sĩ	20 %	18-07-2019	31-07-2019	Combo	Hoạt động	Xem Nhắc Xoá
4	Mừng ngày Phụ nữ Việt Nam 20/10	15 %	18-07-2019	25-10-2019	Đơn lẻ	Hoạt động	Xem Nhắc Xoá
5	Việt Nam Vô Địch	10 %	24-07-2019	24-08-2019	Combo	Hoạt động	Xem Nhắc Xoá

Below the table are navigation buttons: '<' (disabled), '1' (selected), '2', '3' (disabled), '>', and a row of red numbers 10, 11, 12.

Figure 74 - <Interface> Manage promotion

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
2	Tìm kiếm	Admin can search all services in system	No	Yes	Textbox	String
3	STT	Order number of promotions	Yes	Yes	Text	String
4	Tên	Promotion's name	Yes	Yes	Text	String
5	Giảm giá	The value of discount	Yes	Yes	Text	String
6	Ngày bắt đầu	The starting date of promotion	Yes	Yes	Text	date
7	Ngày kết thúc	The ending date of promotion	Yes	Yes	Text	date
8	Thẻ loại	Promotion's type	Yes	Yes	Text	String
9	Trạng thái	Promotion's status	Yes	Yes	Text	boolean

Table 96 - <Fields> Manage promotion

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
1	Thêm mới	Admin can add new a promotion.	N/A	Navigate to add new page.
10	Chi tiết	Admin can view promotion's detail	N/A	The modal is shown.
11	Thông báo	Admin can select the promotion to send the notification to customer	N/A	The pop up is shown
12	Xóa	Admin can delete promotion	N/A	The pop up is shown.

Table 97 - <Button> Manage promotion

5.1.1.6 Manage category

STT	HÌNH ẢNH	TÊN	TRẠNG THÁI	THAO TÁC
1		Dầu xe oto	Hoạt động	
2		Dầu xe moto - xe máy	Hoạt động	
3		Dầu hộp số tự động	Hoạt động	
4		Dầu máy nông nghiệp - xây dựng	Hoạt động	

Figure 75 - <Interface> Manage category

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
5	STT	Order number of categories	Yes	Yes	Text	String
6	Hình ảnh	Category's image	Yes	Yes	image	String

7	Tên	Category's name	Yes	Yes	Text	String
8	Trạng thái	Category's status	Yes	Yes	Text	boolean

Table 98 - <Fields> Manage category

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
1	Thêm mới	Admin can add new a category.	N/A	Navigate to add new page.
2	Sản phẩm	Admin can view all categories that belong a product type	N/A	Show list product category.
3	Dịch vụ	Admin can view all services that belong a product type	N/A	Show list service category.
4	Xe	Admin can view all cars that belong a product type	N/A	Show list car category.
10	Sửa	Admin can edit the category	N/A	Navigate to edit page.
11	Trạng thái	Admin can change category's status	N/A	The modal is shown.

Table 99 - <Button> Manage category

5.1.1.7 Manage order

The screenshot shows the 'Manage order' interface with the following details:

- Header:** Hóa đơn (Invoice) at the top left, with 'Xin chào: -' and 'Đăng xuất' (Logout) on the right.
- Navigation:** On the left, there are links for 'Tài khoản' (Account) and 'Hóa đơn' (Invoice). Below these are banners for 'Sản phẩm' (Products), 'Rheinol since 1985', and 'OW-20'.
- Table Headers:**

Mới	Chờ thực hiện	Đang thực hiện	Chờ thanh toán	Đã thanh toán	Hủy
-----	---------------	----------------	----------------	---------------	-----
- Table Data:**

STT	KHÁCH HÀNG	TỔNG TIỀN (VND)	NGÀY	MÃ ĐƠN	TRẠNG THÁI	THAO TÁC
1	Le Dinh Thien Vu	400.000	11-08-2019	FK5QUIQRN8	Mới	View Edit Delete
2	Le Dinh Thien Vu	400.000	11-08-2019	3MJ0LDJ8SZ	Mới	View Edit Delete
3	Le Dinh Thien Vu	400.000	11-08-2019	SPHWA4GQDM	Mới	View Edit Delete
4	Tấn Trung	510.000	09-08-2019	Q9NKD625LB	Mới	View Edit \$ Delete
5	Tấn Trung	510.000	09-08-2019	5PRFLEDIQ9	Mới	View Edit \$ Delete
- Actions:** Red numbers 1 through 18 are overlaid on specific elements: 1 (button), 2 (button), 3 (button), 4 (button), 5 (button), 6 (button), 7 (button), 8 (STT), 9 (KHÁCH HÀNG), 10 (TỔNG TIỀN (VND)), 11 (NGÀY), 12 (MÃ ĐƠN), 13 (TRẠNG THÁI), 14 (Search bar), 15 (View icon), 16 (Edit icon), 17 (\$ icon), and 18 (Delete icon).
- Pagination:** At the bottom left, there are navigation buttons for '<', '1', '2', '3', '4', and '>'.

Figure 76 - <Interface> Manage order

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
8	STT	Order number of orders	Yes	Yes	Text	String
9	Khách hàng	Customer's name	Yes	Yes	image	String
10	Tổng tiền	Order's money	Yes	Yes	Text	String
11	Ngày	Order creation date	Yes	Yes	Text	boolean
12	Mã đơn	Order's code	Yes	Yes	Text	String
13	Trạng thái	Order's status	Yes	Yes	Text	boolean
14	Tìm kiếm	Sales clerk can search order follow status	No	Yes	Textbox	String

Table 100 - <Fields> Manage order

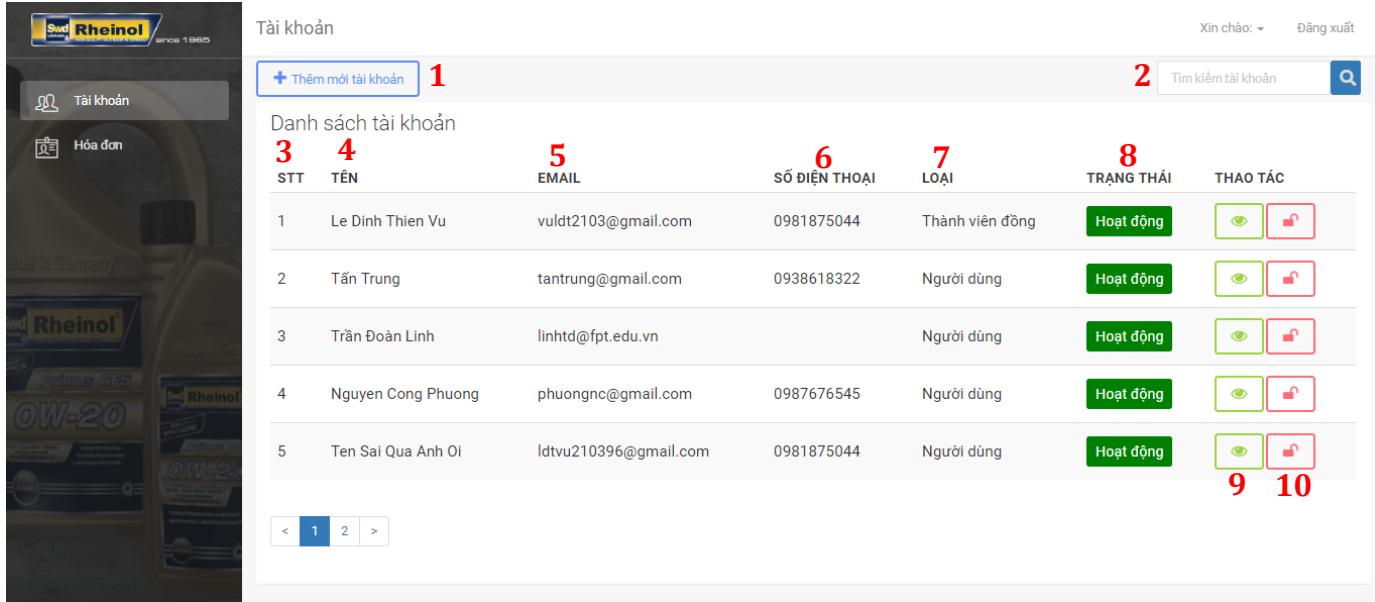
Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
1	Thêm mới	Sales clerk can add new an order.	N/A	Navigate to add new page.
2	Mới	Sales clerk can view all orders that belong a "new" status	N/A	Show list "new" order.
3	Chờ thực hiện	Sales clerk can view all orders that belong a "waiting" status	N/A	Show list "waiting" order.
4	Đang thực hiện	Sales clerk can view all orders that belong a "doing" status	N/A	Show list "doing" order.
5	Chờ thanh toán	Sales clerk can view all orders that belong a "wait for pay" status	N/A	Show list "wait for pay" order.
6	Đã thanh toán	Sales clerk can view all orders that belong a "payed" status	N/A	Show list "payed" order.

7	Hủy	Sales clerk can view all orders that belong a “cancel” status	N/A	Show list “cancel” order.
15	Chi tiết	Sales clerk can view order's detail	N/A	The modal is shown.
16	Sửa	Sales clerk can update order	N/A	Navigate to update page
17	Thanh toán	Sales clerk can change order's status to “payed” status	N/A	The modal is shown.
18	Hủy	Sales clerk can change order's status to “cancel” status	N/A	The modal is shown.

Table 101 - <Button> Manage order

5.1.1.8 Manage account



The screenshot shows a web-based application interface for managing accounts. On the left, there is a sidebar with icons for 'Tài khoản' (Account), 'Hóa đơn' (Bill), and 'Nhập liệu' (Enter data). The main area has a header 'Tài khoản' and a search bar 'Tim kiem tài khoản'. Below the header, there is a button '+ Thêm mới tài khoản' (Add new account) and a red number '1' indicating a new entry. The main content is a table titled 'Danh sách tài khoản' (List of accounts) with the following columns: STT (Number), TÊN (Name), EMAIL, SỐ ĐIỆN THOẠI (Phone number), LOẠI (Type), TRANG THÁI (Status), and THAO TÁC (Actions). The table contains five rows of data, each with a green eye icon and a red edit icon in the actions column. Red numbers '3', '4', '5', '6', '7', '8', '9', and '10' are overlaid on the table headers and data cells. At the bottom of the table, there are navigation buttons '<' and '>' with page numbers '1', '2', and '3'.

Figure 77 - <Interface> Manage account

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
2	Tìm kiếm	Sales clerk can search all customer	No	Yes	Textbox	String

		account in system				
3	STT	Order number of accounts	Yes	Yes	Text	String
4	Tên tài khoản	Account's name	Yes	Yes	Text	String
5	Email	Account's email	Yes	Yes	Text	String
6	Số điện thoại	Account's phone	Yes	Yes	Text	String
7	Loại	Account's type	Yes	Yes	Text	String
8	Trạng thái	Account's status	Yes	Yes	Text	boolean

Table 102 - <Fields> Manage account

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
1	Thêm mới	Sales clerk can add new an account.	N/A	Navigate to add new page.
9	Chi tiết	Sales clerk can view account's detail	N/A	The modal is shown.
10	Trạng thái	Sales clerk can change account's status	N/A	The modal is shown

Table 103 - <Button> Manage account

5.1.1.9 View revenue report

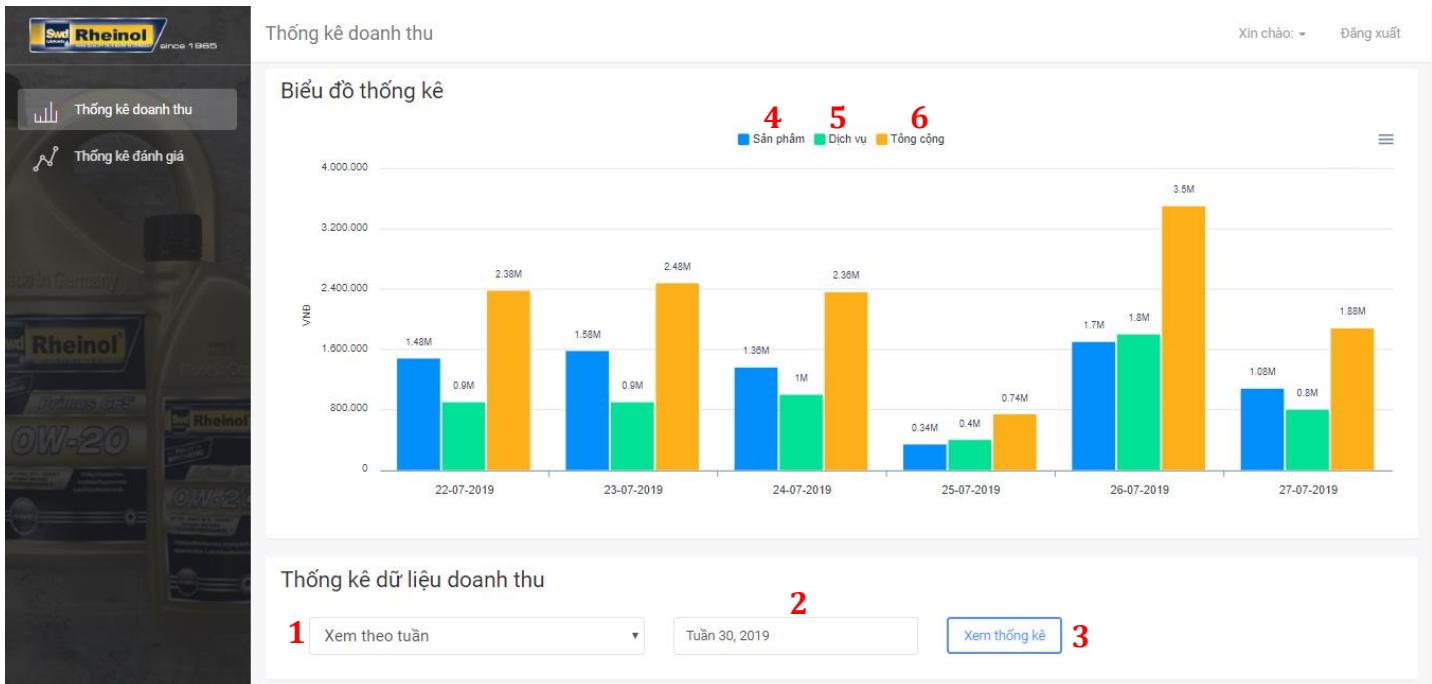


Figure 78 - <Interface> View report

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
4	Sản phẩm	Product's revenue	Yes	Yes	Text	String
5	Dịch vụ	Service's revenue	Yes	Yes	Text	String
6	Tổng cộng	Total revenue	Yes	Yes	Text	String
1	Thể loại	Director can choose type of time	No	Yes	Combo box	String
2	Thời gian	Director selects time line	No	Yes	Date	date

Table 104 - <Fields> View report

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
3	Xem thống kê	Director can view the revenue report	N/A	Report chart is shown

Table 105 - <Button> View report

5.1.1.10 View rating report



Figure 79 - <Interface> View rating report

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
4	Điểm đánh giá	The rating point	Yes	Yes	Text	String
5	Số lượt đánh giá	Number of reviews	Yes	Yes	Text	String
1	Thể loại	Director can choose type of time	No	Yes	Combo box	String
2	Thời gian	Director selects time line	No	Yes	Date	date

Table 106 - <Fields> View rating report

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome

3	Xem thống kê	Director can view the revenue report	N/A	Report chat is shown
---	--------------	--------------------------------------	-----	----------------------

Table 107 - <Button> View rating report

5.1.2 Mobile Screen

5.1.2.1 Login

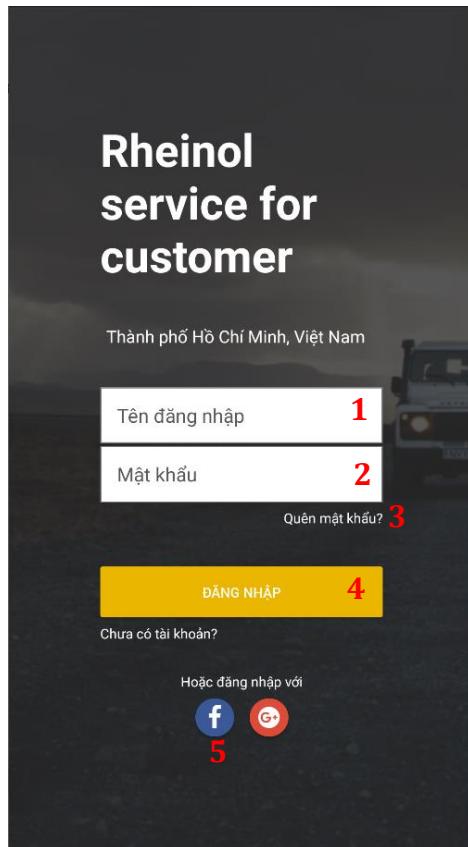


Figure 80 - <Mobile> Login

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
1	Tài khoản	User name to login	No	Yes	Textbox	String
2	Mật khẩu	Password to login	No	Yes	Textbox	String

Table 108 - <Fields> Login

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
3	Quên mật khẩu	User can update new password	N/A	Textboxes are shown

4	Đăng nhập	Guest can log in the app	Yes	Navigate to home screen
5	Facebook	Guest can log in by Facebook	Yes	Navigate to home screen

Table 109 - <Button> Login

5.1.2.2 Home Screen

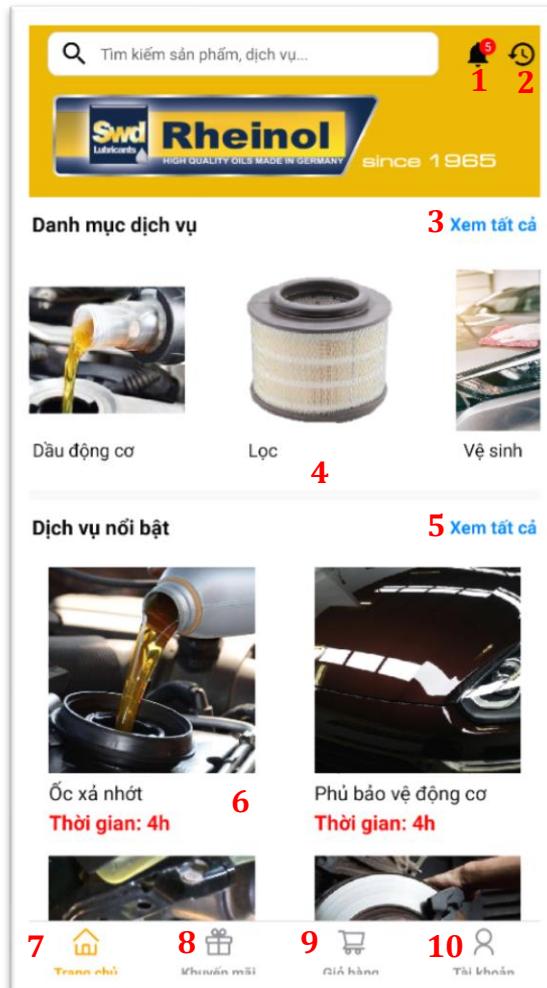


Figure 81 - <Mobile> Home screen

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
1	Thông báo	User can update new password	N/A	Navigate to notification screen
2	Đăng nhập	Guest can log in the app	Yes	Navigate to home screen
3	Xem tất cả	Customer can view all categories	Yes	Navigate to category screen
4	Danh mục dịch vụ	Customer can view list of product or service	N/A	Navigate to list product or service screen

		service belong to category		
5	Xem tất cả	Customer can view all services	N/A	Navigate to list service screen
6	Dịch vụ	Customer can view service's detail	N/A	Navigate to service screen
7	Trang chủ	Home screen	N/A	Navigate to home screen
8	Khuyến mãi	View all promotions	N/A	Navigate to promotion screen
9	Giỏ hàng	Cart screen	N/A	Navigate to cart screen
10	Tài khoản	User screen	N/A	Navigate to account screen

Table 110 - <Button> Home screen

5.1.2.3 List Categories of Service

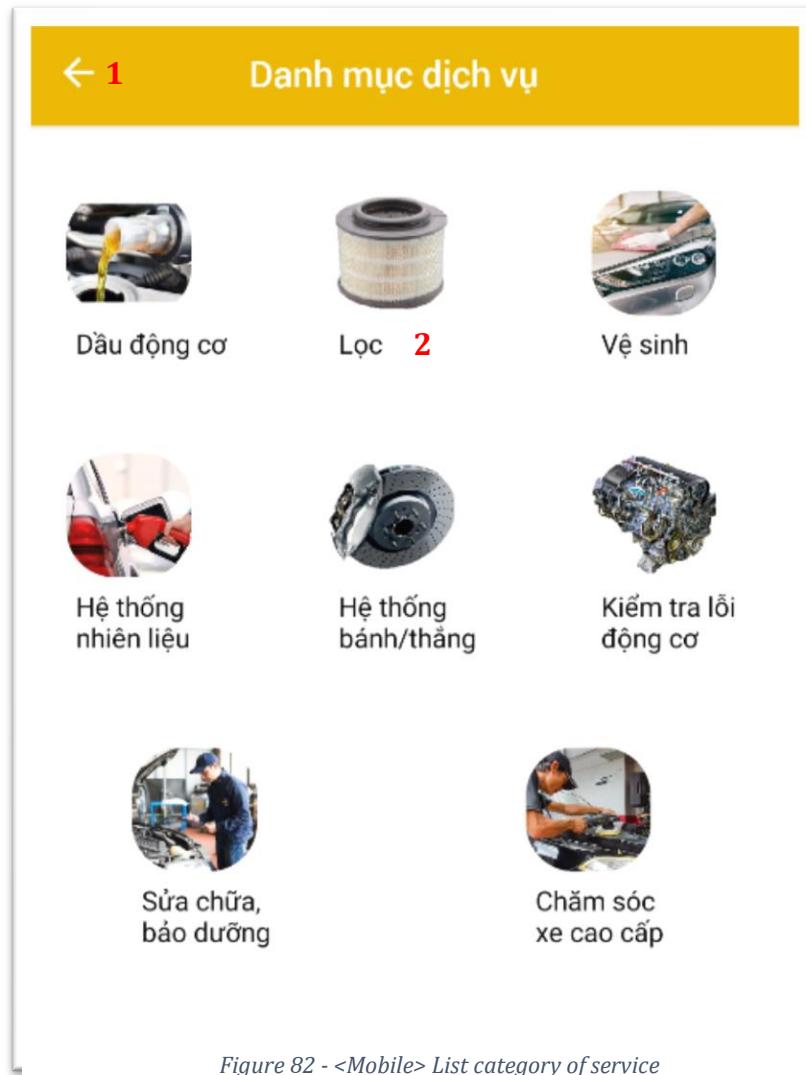


Figure 82 - <Mobile> List category of service

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
1	Quay lại	Back to home screen	N/A	Navigate to home screen
2	Danh mục	User can view all services belong to category	N/A	Navigate to service screen

Table 111 - <Button> List category

5.1.2.4 List of promotion



Việt Nam Vô Địch



Figure 83 - <Mobile> List of promotion

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
1	Hình ảnh	Promotion's image	No	Yes	image	String

2	Tên	Promotion's name	No	Yes	text	String
---	-----	------------------	----	-----	------	--------

Table 112 - <Fields> List of promotion

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
1,2	Chi tiết	Go to view promotion's detail	N/A	Navigate to promotion's detail screen

Table 113 - <Button> List of promotion

5.1.2.5 Promotion's detail



Figure 84 - <Mobile> Promotion detail

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
1	Tên	Promotion's name	Yes	Yes	text	String
2	Thời gian	Promotion's duration	Yes	Yes	text	String
3	Sản phẩm	List product of promotion	Yes	Yes	text	String
4	Dịch vụ	List service of promotion	Yes	Yes	text	String
5	Mô tả	Promotion's description	Yes	Yes	text	String

Table 114 - <Fields> Promotion detail

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
6	Thêm ngay	List product and service of promotion will be added to cart.	N/A	Navigate to cart screen

Table 115 - <Button> Promotion detail

5.1.2.6 Cart

Giỏ hàng (2)

Sản phẩm

	Primus ECO 5W20 2 Cung cấp bởi Rheinol 306.000 đ 3 340.000đ	1 ×
Dịch vụ		4 5
	Vệ sinh thắng 6 Cung cấp bởi Rheinol 100.000 đ	X

Khuyến mãi **7** **8**

Áp Dụng **Gợi ý**

Đã áp dụng khuyến mãi: Kỉ niệm ngày tựu trường

Thành tiền **10** **9 406.000 đ**
Đã bao gồm VAT

Đặt Lịch

Trang chủ Khuyến mãi Giỏ hàng Tài khoản

Figure 85 - <Mobile> Cart

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
2	Tên	Product's name	Yes	Yes	text	String
3	Giá	Product's price	Yes	Yes	text	String
6	Tên	Service's name	Yes	Yes	text	String
9	Tổng tiền	Order's total money	Yes	Yes	text	String

Table 116 - <Fields> Cart

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
1	Hủy	Product or service will be removed out the cart.	N/A	The item in car will be remove out the cart
4	Giảm	User can minus the product's quantity.	Yes	Product's quantity decreased.
5	Tăng	User can add the product's quantity.	Yes	Product's quantity increased.
7	Áp dụng	The promotion is suitable with order.	Yes	List of suitable promotion
8	Gợi ý	List product or service will be suggested to fit with promotion	Yes	List suggest promotion
10	Đặt lịch	User selects time to use service	Yes	Navigate to "select time" screen

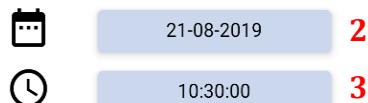
Table 117 - <Button> Cart

5.1.2.7 Book time



Chọn ngày hẹn

Tổng thời gian của dịch vụ là: 2 giờ **1**



Xếp lịch thành công

Quý khách vui lòng đến trước 10 phút so với giờ hẹn.

Chúng tôi xin phép hủy lịch của quý khách nếu quý khách đến trễ 15 phút so với giờ hẹn.

Thời gian đặt lịch sử dụng dịch vụ:

20-08-2019 23:30 4

Thời gian dự kiến hoàn thành:

21-08-2019 01:30 5

6

Đặt lịch hẹn

Figure 86 - <Mobile> Book time

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
1	Thời gian	Service's duration	Yes	Yes	text	String
2	Ngày	Select date to book	No	Yes	date	String
3	Giờ	Select time to book	No	Yes	time	String
4	Thời gian đặt	The time that customer selected	Yes	Yes	text	String
5	Thời gian kết thúc	The time will complete service	Yes	Yes	text	String

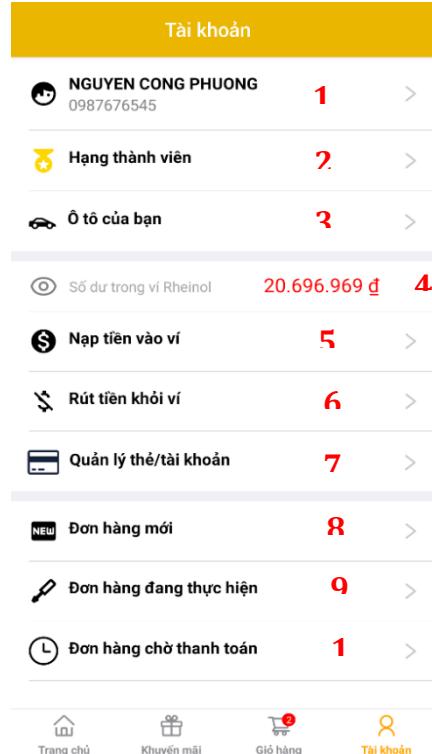
Table 118 - <Fields> Book time

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
6	Đặt lịch hẹn	The customer will create order.	Yes	Navigate to order detail screen

Table 119 - <Button> Book time

5.1.2.8 Customer



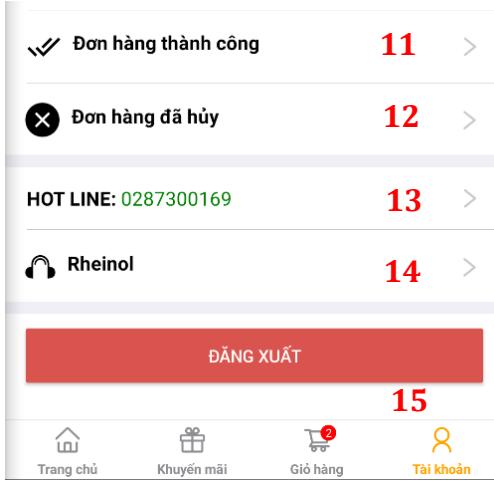


Figure 87 - <Mobile> Customer

Fields

No	Field Name	Description	Read only	Mandatory	Control type	Data type
4	Số dư	The wallet's balance	Yes	Yes	text	String

Table 120 - <Fields> Customer

Buttons/Hyperlinks

No	Function	Description	Validation	Outcome
1	Tài khoản	The customer can edit account	N/A	Navigate to "edit account" screen
2	Hang thành viên	The customer can view account's type.	N/A	Navigate to "account's type" screen.
3	Ô tô	The customer can add the new car.	N/A	Navigate to "add the new car" screen.
5	Nạp tiền	The customer can cash in wallet.	Yes	Navigate to "cash in" screen
6	Rút tiền	The customer can cash out wallet	Yes	Navigate to "cash out" screen
7	Quản lý thẻ	The customer can register the card	N/A	Navigate to "register the card" screen
8	Đơn hàng mới	The customer can see list new order	N/A	Navigate to "list new order" screen
9	Đơn hàng đang thực hiện	The customer can see list "doing" order	N/A	Navigate to list "doing" order screen

10	Đơn hàng chờ thanh toán	The customer can see list "waiting for pay" order	N/A	Navigate to list "waiting for pay" order screen
11	Đơn hàng thành công	The customer can see list success order	N/A	Navigate to "list success order" screen
12	Đơn hàng hủy	The customer can see list "cancel" order	N/A	Navigate to list "cancel" order screen
13	Hot line	The customer can call rheinol's phone	N/A	Navigate to phone screen
14	Thông tin	The customer can see Rheinol's information	N/A	Navigate to "information" screen
15	Đăng xuất	The customer can log out the system	N/A	Navigate to "Log in" screen

Table 121 - <Button> Customer

6. Database Design

6.1 Entity Relationship Diagram (ERD)

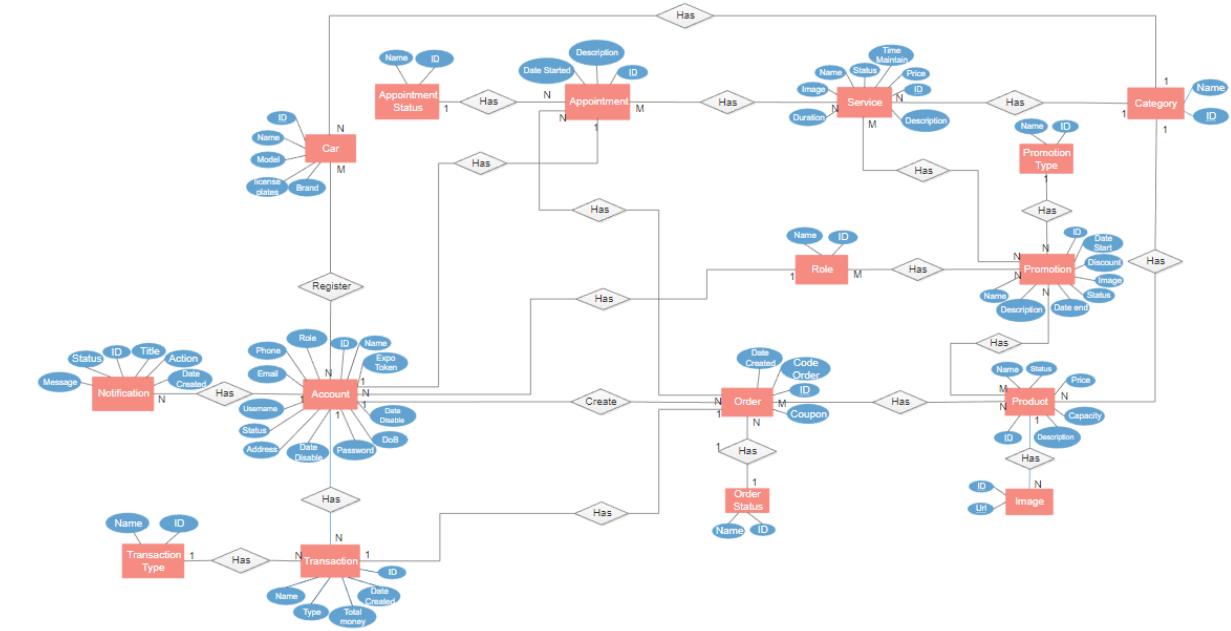


Figure 88 - ERD - Entity Relationship Diagram

6.2 Data Dictionary

Entity Data Dictionary: Describe the content of all entities

Entity Name	Description
Notification	Contain the notification information of an account.
Transaction Type	Store the type information of a transaction.
Transaction	Contain the information of a transaction that generated whenever user processes with money.
Account	Contain the account information.
Car	Contain the car information owned by the user.
Order	Contain information about an order created by the user.
Order Status	Contain the status of an order.
Appointment	Show the information of an appointment, which is created when the user sets a calendar.
Appointment status	Indicates the status of the appointment.
Service	Store information about the services.
Category	Contains type information of services, products, cars
Product	Store information on the products.
Image	Contain the image information.
Promotion	Contains information promotions for users.
Promotion Type	Contain type promotions of combo or single.
Role	Store the role information of an account.

Table 122 - <Entity> Dictionary

Entity Name	Attributes	Description	Domain	Null
Notification	Id	Id of notification	INT (11)	NO
	Status	Status of notification (read or unread)	BIT (1)	NO
	Message	Message of notification	VARCHAR (200)	NO
	Title	Title of notification	VARCHAR (255)	NO
	Action	A screen will forward to	VARCHAR (255)	NO
	Date Created	Date created of the notification.	DATETIME	NO

Transaction Type	Id	Id of transaction type	INT (11)	NO
	Name	Name of transaction type	VARCHAR (255)	NO
Transaction	Id	Id of transaction	INT (11)	NO
	Name	Name of transaction	VARCHAR (255)	NO
	Type	Type of an transaction	INT (11)	NO
	Total money	Total money of a transaction	INT (11)	NO
	Date created	Date created of a transaction	DATETIME	NO
Account	Id	Id of account	INT (11)	NO
	Username	Store username of account	VARCHAR (45)	NO
	Status	Status of an account	BIT (1)	NO
	Address	Indicate address of account	VARCHAR (255)	NO
	Date Disable	The last date that an account activates	DATETIME	NO
	DoB	Date of birth of an account	DATETIME	YES
	Password	Password of the account	VARCHAR (255)	NO
	Email	Store email of the account	VARCHAR (45)	NO
	Phone	Contain phone of the account	VARCHAR (45)	NO
	Role	Indicate role of the account	INT (11)	NO
	Name	Name of the account	VARCHAR (255)	YES
	Expo Token	Token of the account	VARCHAR (65)	NO
Car	Id	Id of the car	INT (11)	NO
	Name	Name of the car	VARCHAR (60)	NO
	Model	Contain model of a car	VARCHAR (45)	YES
	License plates	Contain license plates of the car	VARCHAR (45)	NO
	Brand	Contain the brand of the car	VARCHAR (45)	YES
Order	Id	Id of the order	INT (11)	NO

	Code Order	Store identity of the order	VARCHAR (255)	NO
	Date created	The date an order created	DATETIME	NO
	Coupons	Contain the identity of promotions	INT (11)	NO
Order Status	Id	Id of order status	INT (11)	NO
	Name	Contain name of order status	VARCHAR (255)	NO
Appointment	Id	Id of the appointment	INT (11)	NO
	Description	Store description of the appointment	VARCHAR (255)	NO
	Date Started	The date when the appointment occurs	DATETIME	NO
Appointment Status	Id	Id of appointment status	INT (11)	NO
	Name	Name of appointment status	VARCHAR (255)	NO
Service	Id	Id of the service	INT (11)	NO
	Price	Store price of the service	INT (11)	NO
	Description	Contain description of the service	VARCHAR (255)	NO
	Duration	Indicate how long a service	DOUBLE	NO
	Image	The path of an image that present for a service	VARCHAR (255)	NO
	Name	Name of the service	VARCHAR (255)	NO
	Status	Status of a service	BIT (1)	NO
Category	Id	Id of category	INT (11)	NO
	Name	Store name of the category	VARCHAR (255)	NO
Product	Id	Id of the product	INT (11)	NO
	Name	Contain the name of product	VARCHAR (255)	NO
	Status	Contain status of a product	BIT (1)	NO
	Price	Price of the product	INT (11)	NO
	Capacity	Capacity of oil product	INT (11)	YES
	Description	Description of the product	VARCHAR (255)	YES

Image	Id	Id of the image	INT (11)	NO
	Url	Store path of the image	VARCHAR (255)	NO
Promotion	Id	Id of the promotion	INT (11)	NO
	Name	Name of the promotion	VARCHAR (255)	NO
	Description	Store the description of the promotion.	VARCHAR (255)	NO
	Date start	The date a promotion starts	DATETIME	NO
	Status	Status of the promotion	BIT (1)	NO
	Image	Image of the promotion	VARCHAR (255)	NO
	Discount	Number discount of the promotion	INT (11)	NO
	Date end	The date a promotion stopped	DATETIME	NO
Promotion Type	Id	Id of promotion type	INT (11)	NO
	Name	Contain type of the promotion	VARCHAR (255)	NO
Role	Id	Id of the role	INT (11)	NO
	Name	Store name of the role	VARCHAR (255)	NO

Table 123 - <Entity> Description

E. System Implementation & Test

1. Introduction

1.1 Overview

- This section describes the approach and methodologies used by our team to plan, organizes and manage the testing process of RQS
- It provides all necessary information about implementation and testing procedure of the system including test plans, test cases, test result, and pass/fails criteria as well as testing flow to cover all possible cases.

1.2 Test Approach

- **Goal:** Test main features in the whole RQS based on core flow.

- **Method:** Black-box testing.

2. Data Relationship Diagram

2.1 Physical Diagram

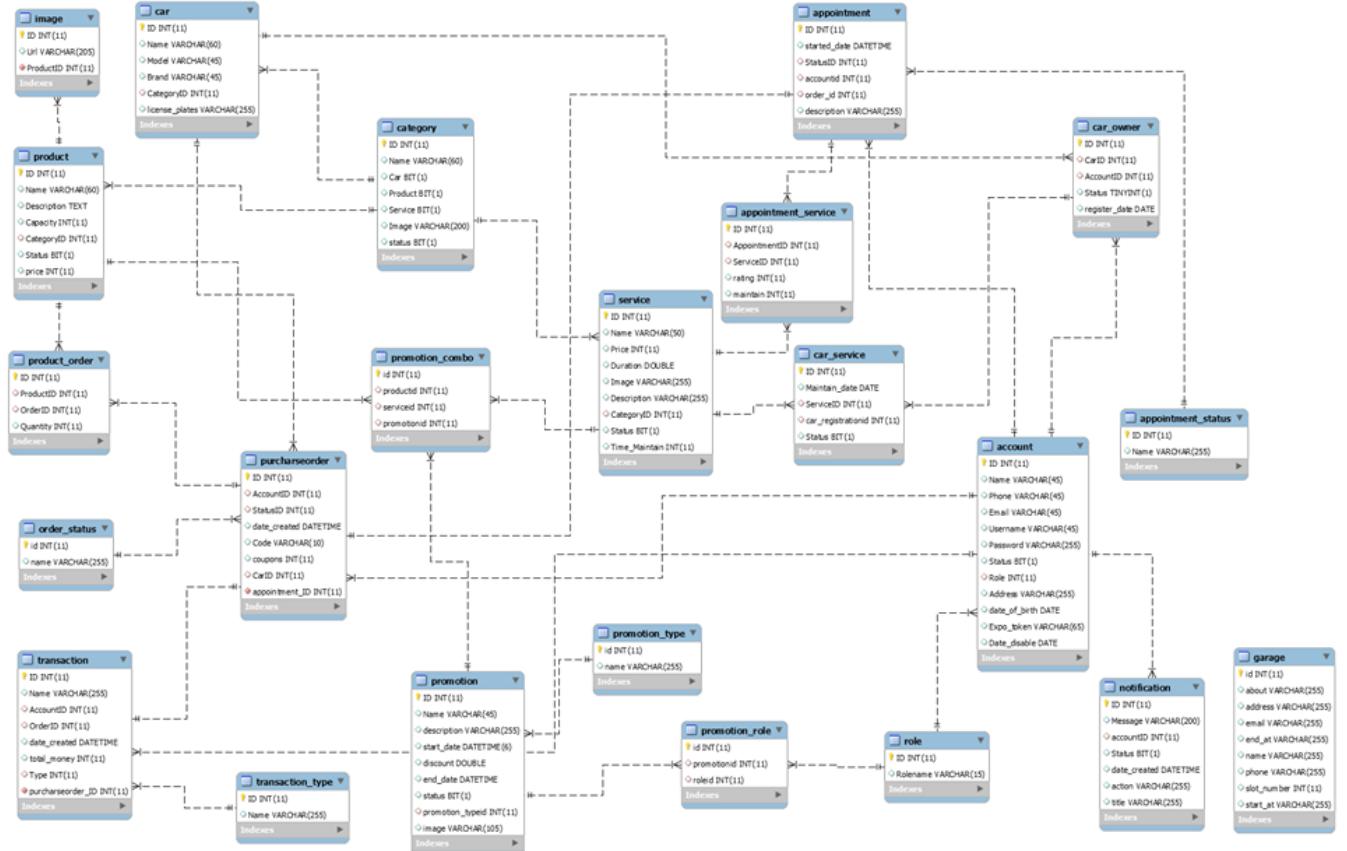


Figure 89 - Physical Diagram

2.2 Data Dictionary

Entity Data Description: Describe the content of all tables		
No.	Table Name	Description
1	garage	Contain garage information.
2	car	Contain car information.
3	car_owner	Contain information about the relationship between an account and a car.
4	service	Contain service information of the garage.
5	image	Contain image information.
6	product	Contain product information of the garage.
7	product_order	Contain information about the relationship between a product and an order.

8	account	Contain account information of the user.
9	promotion	Contain promotion information of the garage.
10	promotion_type	Contain information about the type of promotion.
11	promotion_role	Contain information about the relationship between a role and a promotion.
12	promotion_combo	Contain information about the relationship between product, service, and promotion.
13	purchaseorder	Contain order information that user creates.
14	order_status	Contain the status of an order.
15	transaction	Contain transaction information that generated when user processes with money.
16	transaction_type	Contain information about the type of a transaction.
17	appointment	Contain appointment information that generated when a user books the service.
18	appointment_status	Store the status of an appointment.
19	appointment_service	Contain information about the relationship between an appointment and a service.
20	notification	Contain notification information of the user.
21	category	The table contains categories of product, service, car.
22	role	Contain role information of an account.

Table 124 - <Physical> Data dictionary

3. Test Plan

The overall purpose of testing is to ensure RQS meets its entire technical, functional and business requirement. The aim of this document, in order to describe the test plan and strategy for testing the RQS. The following part will describe which features to be tested and which will not.

3.1 Features to be tested

Authorized User

- Buy product test
- Book service test
- Suggest promotion test
- Register visa/master card test
- Withdraw money test
- Recharge money test
- Scheduler maintains test
- Upgrade account test
- Notification test
- Register car test

Technical employees

- Recognize Voice test

3.2 Features not to be tested

Guest:

- Sign in
- Sign up
- Forget password

Admin

- Manage promotion
- Manage service
- Mange product
- Manage garage
- Manage account
- Manage category

Sales clerk

- Manage order

Director

- View report rating test
- View report transaction test

4. System Testing Test Case

4.1 Authorized User Test Case

4.1.1 Buy Product Test

ID	Test Case Description	Test Case Procedure	Expected Output	Inter Dependence -Test Case	Result	Test Date	Note
BPT_1	User buy product in the mobile app with enough balance	1. The user chooses a product and adds to cart 2. The user chooses “Tạo đơn hàng” in the cart screen 3. The user chooses “Thanh toán” button in “Chờ thanh toán” screen at “Tài khoản” bottom tab	The system displays success notification and balance lost money equal total money of the order.	N/A	Pass	01/08/2019	
BPT_2	User buy product in the mobile app with not enough balance	1. The user chooses the product and adds to cart 2. The user chooses “Tạo đơn hàng” in the cart screen	The system shows an error message and displays a toast notification	N/A	Pass	01/08/2019	

		3. The user chooses “Thanh toán” button in “Chờ thanh toán” screen at “Tài khoản” bottom tab					
--	--	--	--	--	--	--	--

Table 125 - <Test case> Buy product

4.1.2 Book Service Test

ID	Test Case Description	Test Case Procedure	Expected Output	Inter Dependence -Test Case	Result	Test Date	Note
BST_1	User book service in the mobile app on the available slot	1. The user chooses to service and add to cart 2. The user chooses “Đặt lịch” in the cart screen 3. The user chooses “Chọn ngày” button and system show a busy slot. 4. The user chooses time is	System confirm this date and time can book service	N/A	Pass	01/08/2019	

		outside of the busy slot					
BST_2	User book service in the mobile app on busy slot	1. The user chooses to service and add to cart 2. The user chooses “Đặt lịch” in the cart screen 3. The user chooses “Chọn ngày” button and system show a busy slot. 4. The user chooses time is inside of the busy slot	The system shows an error on this date and time and allows the user to choose other date or time or both	N/A	Pass	01/08/2019	

Table 126 - <Test case> Book Service

4.1.3 Suggest Promotion Test

ID	Test Case Description	Test Case Procedure	Expected Output	Inter Dependence -Test Case	Result	Test Date	Note
SPT_1	The user chooses a product or book	1. The user chooses to service or	System show corresponding promotion	N/A	Pass	01/08/2019	

	service that promotion contains it.	product and add to cart 2. The user goes to cart screen					
SPT_2	The user chooses a product or book service that there is no promotion contains it.	1. The user chooses to service or product and add to cart 2. The user goes to cart screen	System show “Không tìm thấy khuyến mãi” message	N/A	Pass	01/08/2019	

Table 127 - <Test case> Suggest promotion

4.1.4 Register Card Test

ID	Test Case Description	Test Case Procedure	Expected Output	Inter Dependence -Test Case	Result	Test Date	Note
RCT_1	The user registers a valid card	1. The user input valid information of the card	The system enables the “Subscribe” button when the card is valid	N/A	Pass	01/08/2019	
RCT_2	The user registers an invalid card	1. The user input invalid information of the card	The system disables the “Subscribe” button	N/A	Pass	01/08/2019	

Table 128 - <Test case> Register card

4.1.5 Withdraw Money Test

ID	Test Case Description	Test Case Procedure	Expected Output	Inter Dependence -Test Case	Result	Test Date	Note
WMT_1	The user withdraws money is equal or less balance in Rheinol wallet.	1. The user input amount of money to withdraw money	The system show success notification in screen	N/A	Pass	01/08/2019	
WMT_2	The user withdraws money is large than the balance in Rheinol wallet.	1. The user input amount of money to withdraw money	The system show error notification in screen and user can input again until the amount of money is equal or less than the balance in the wallet	N/A	Pass	01/08/2019	

Table 129 - <Test case> Withdraw Money

4.1.6 Recharge Money Test

ID	Test Case Description	Test Case Procedure	Expected Output	Inter Dependence -Test Case	Result	Test Date	Note
RMT_1	The user recharge	1. The user input	The system show success	N/A	Pass	01/08/2019	

	money is equal or less balance in their card.	amount of money to recharge money	notification in screen				
RMT_2	The user withdraws money is large than the balance in their card.	1. The user input amount of money to recharge money	The system show error notification in screen and user can input again until the amount of money is equal or less balanced in the card	N/A	Pass	01/08/2019	

Table 130 - <Test case> Recharge money

4.1.7 Scheduler Maintain Test

ID	Test Case Description	Test Case Procedure	Expected Output	Inter Dependence -Test Case	Result	Test Date	Note
SMT_1	The user uses a service have time maintain and system create a	1. The technical employee will check service that updates	The system scheduler a date, time in the future for this service	N/A	Pass	01/08/2019	

	scheduler for maintaining in the future	status to complete.					
SMT_2	The user uses a service don't have time to maintain and system create a scheduler for maintaining in the future	1. The technical employee will check service that updates status to complete.	The system doesn't schedule a date, time in the future for this service	N/A	Pass	01/08/2019	

Table 131 - <Test case> Schedule Maintain

4.1.8 Upgrade Account Test

ID	Test Case Description	Test Case Procedure	Expected Output	Inter Dependence -Test Case	Result	Test Date	Note
UAT_1	The user spent money enough for an upgrade to bronze member	1. User payment order by Rheinol wallet. Total spend money is more than bronze level	The account is upgraded to bronze account	N/A	Pass	01/08/2019	

UAT_2	The user spent money not enough for an upgrade to bronze member	1. User payment order by Rheinol wallet. Total spend money is less than the bronze level	The account is not upgraded to bronze account	N/A	Pass	01/08/2019	
UAT_3	The user spent money enough for an upgrade to silver member	1. User payment order by Rheinol wallet. Total spend money is more than the silver level	The account is upgraded to a silver account	N/A	Pass	01/08/2019	
UAT_4	The user spent money not enough for an upgrade to silver member	1. User payment order by Rheinol wallet. Total spend money is less than the silver level	The account is not upgraded to a silver account	N/A	Pass	01/08/2019	
UAT_5	The user spent money enough for an upgrade	1. User payment order by Rheinol wallet. Total	The account is upgraded to gold account	N/A	Pass	01/08/2019	

	to a gold member	spend money is more than the gold level					
UAT_6	The user spent money not enough for an upgrade to a gold member	1. User payment order by Rheinol wallet. Total spend money is less than the gold level	The account is not upgraded to gold account	N/A	Pass	01/08/2019	

Table 132 - <Test case> Upgrade account

4.1.9 Notification Test

ID	Test Case Description	Test Case Procedure	Expected Output	Inter Dependenc e -Test Case	Result	Test Date	Note
NT_1	The user payment an order successfully and have a notification to the user's device	1. The user goes to “Đơn hàng chờ thanh toán” screen and process payment an order	The system pushes notifications to the device and creates a notification for mobile app	N/A	Pass	01/08/2019	

NT_2	The user payment an order failed and has no notification to the user's device	1. The user goes to “Đơn hàng chờ thanh toán” screen and process payment an order	The system doesn't pusher notifications to the device and doesn't create a notification for mobile app	N/A	Paccountas	01/08/2019	
------	---	---	--	-----	------------	------------	--

Table 133 - <Test case> Notification

4.1.10 Register Car Test

ID	Test Case Description	Test Case Procedure	Expected Output	Inter Dependence -Test Case	Result	Test Date	Note
RCT_1	The user registers their car in the system	1. The user goes to “Tài khoản” screen and chooses “Ô tô của bạn” button 2. Input information of the car and choose “Đăng ký xe” button	The system show register success notification	N/A	Pass	01/08/2019	

RCT_2	The user registers their existing car in the system	<p>1. The user goes to “Tài khoản” screen and chooses “Ô tô của bạn” button</p> <p>2. Input information of the car and choose “Đăng ký xe” button</p>	The system show error when registering this car.	N/A	Pass	01/08/2019	
-------	---	---	--	-----	------	------------	--

Table 134 - <Test case> Register car

4.2 Technical Employees Test Case

4.2.1 Recognize Voice Test

ID	Test Case Description	Test Case Procedure	Expected Output	Inter Dependence -Test Case	Result	Test Date	Note
RVT_1	The technical employee check the car and confirm service by using recognize voice with good sound	<p>1. The technical employee check the car and go to detail service on the mobile app</p> <p>2. Choose “Giữ để ghi âm mô tả”</p>	System recognize their voice and convert it to text and display in input description	N/A	Pass	01/08/2019	

		button to record your voice					
RVT_2	The technical employee check the car and confirm service by using recognize voice with bad sound	1. The technical employee check the car and go to detail service on the mobile app 2. Choose “Giữ để ghi âm mô tả” button to record your voice	System recognize their voice and convert it to text and display in input description	N/A	Pass	01/08/2019	
RVT_3	The technical employee check the car and confirm service by using recognize voice with the English language	1. The technical employee check the car and go to detail service on the mobile app 2. Choose “Giữ để ghi âm mô tả” button to record your voice	System recognize their voice and convert it to text and display in input description	N/A	Pass	01/08/2019	

RVT_4	The technical employee check the car and confirm service by using recognize voice with the Vietnamese language	<p>1. The technical employee check the car and go to detail service on the mobile app</p> <p>2. Choose “Giữ đế ghi âm mô tả” button to record your voice</p>	System recognize their voice and convert it to text and display in input description	N/A	Pass	01/08/2019	
-------	--	--	--	-----	------	------------	--

Table 135 - <Test case> Recognize voice

F. Software User's Manual

1. Installation Guide

1.1 Setting up the environment at the server-side

1.1.1 Hardware requirements

Server	Minimum Requirements	Recommended
Internet Connection	Cable, Wi-Fi (8 Mbps)	Cable, Wi-Fi (50 Mbps or more)
Operating System	XP, Vista, 7, 10, Window Server 2008	10, Window Server 2008
Computer Processor	Intel® Xeon ® 1.4GHz	Intel® Xeon ® Quad Core (12M Cache, 2.50 GHz)
Computer Memory	4GB RAM	32GB or more
Storage space	1GB	5GB or more

Table 136 - Hardware requirement

1.1.2 Software requirements

Name	Name/Version	Description
Environment	Java EE 8.0,Node v10,npm v6	Specification for developing web application
Operating System	Window 10	Operating system and platform for development
Modeling tool	StarUML	Used to design diagram
IDE	IntelliJ IDEA 2018.1.5, Visual Studio Code 1.27.2	Programming tools
DBMS	MySQL 8.0	Used to create & manage the database for system
Web server	Apache Tomcat 8	Deployment environment
Web browser	Chrome 69 or above.	Testing browser

Table 137 - Software requirement

1.2 Deployment at server side

1.2.1 Prepare deployment package

Step 1: Download and Install Java 8

Link download: <http://www.oracle.com/technetwork/java/javase/downloads/java-archive-javase8-2177648.html>

Step 2: Download and setup Tomcat 8

Link download: <https://tomcat.apache.org/download-80.cgi>

Step 3: Download and install Node v10

Link download: <https://nodejs.org/en/download/>

Step 4: Download and install IntelliJ IDE

Link download: <https://www.jetbrains.com/idea/download/#section=windows>

Step 5: Download, install and configure MySQL 8.0

Link download: <https://dev.mysql.com/downloads/installer/>

1.2.1 Configure server before deploy

1.3 Setting up the environment at the client-side

The client devices should have one of the following browsers to access the website:

Google Chrome

Link download: <https://www.google.com/chrome/>

Firefox

Link download: <https://www.mozilla.org/en/firefox/channel/desktop/>

Besides the browser, the client mobile devices could use the application with android version 9

Android v9: <https://www.android.com/versions/pie-9-0/>

2. User Guide

2.1 Admin

2.1.1 Product

2.1.1.1 Add product

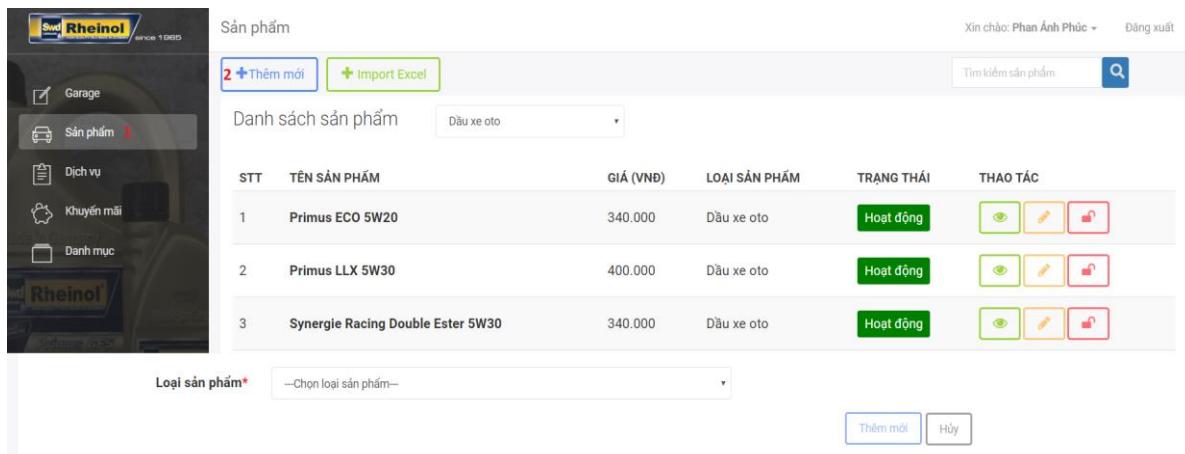


Figure 90 - Add product

Figure 91 - Add product detail

Step	Description
Pre-Condition	Login as Admin
1	Click “Sản phẩm” on menu bar
2	Click “Thêm mới” button
3	Fill the field “Tên”
4	Fill the field “Giá cả”
5	Fill the field “Dung tích”

6	Choose “Hình ảnh” for product
7	Choose “Loại sản phẩm”
8	Click “Thêm mới” button

Table 138 - Add product steps

2.1.1.2 Edit product

STT	TÊN SẢN PHẨM	GIÁ (VND)	LOẠI SẢN PHẨM	TRẠNG THÁI	THAO TÁC
11	Synergie Racing Double Ester 5W30	510.000	Dầu xe oto	Vô hiệu hóa	
12	Primus DX 5W30	340.000	Dầu xe oto	Vô hiệu hóa	
13	Primus GF5 SAE 0W-20	340.000	Dầu xe oto	Vô hiệu hóa	

Figure 92 - Edit product

Chỉnh sửa sản phẩm

Tên *	Primus DX 5W30 3
Giá cả *	340000 4
Dung tích	1 5
Mô tả	là dầu tiết kiệm nhiên liệu hiệu suất cao được làm từ dầu gốc tổng hợp. Sản phẩm được dự kiến sử dụng tất cả các mùa cho các dòng ô tô với động cơ xăng và diesel hiện đại 6
Hình ảnh*	7
Loại sản phẩm*	Dầu xe oto 8
<input type="button" value="9 Chính sửa"/> <input type="button" value="Hủy"/>	

Figure 93 - Edit product detail

Step	Description
Pre-Condition	Login as Admin The product must be disable
1	Click “Sản phẩm” on the menu bar
2	Click edit icon button
3	Fill the field “Tên”
4	Fill the field “Giá cả”
5	Fill the field “Dung tích”
6	Fill the field “Mô tả”
7	Choose “Hình ảnh” for product
8	Choose “Loại sản phẩm”
9	Click “Chỉnh sửa” button

Table 139 - Edit product steps

2.1.1.3 Search product

Figure 94 - Search product detail

Step	Description
Pre-Condition	Login as Admin
1	Click “Sản phẩm” on the menu bar
2	Fill the field search with “Tìm kiếm sản phẩm” placeholder
3	Click the search icon button

Table 140 - Search product steps

2.1.1.4 Disable product

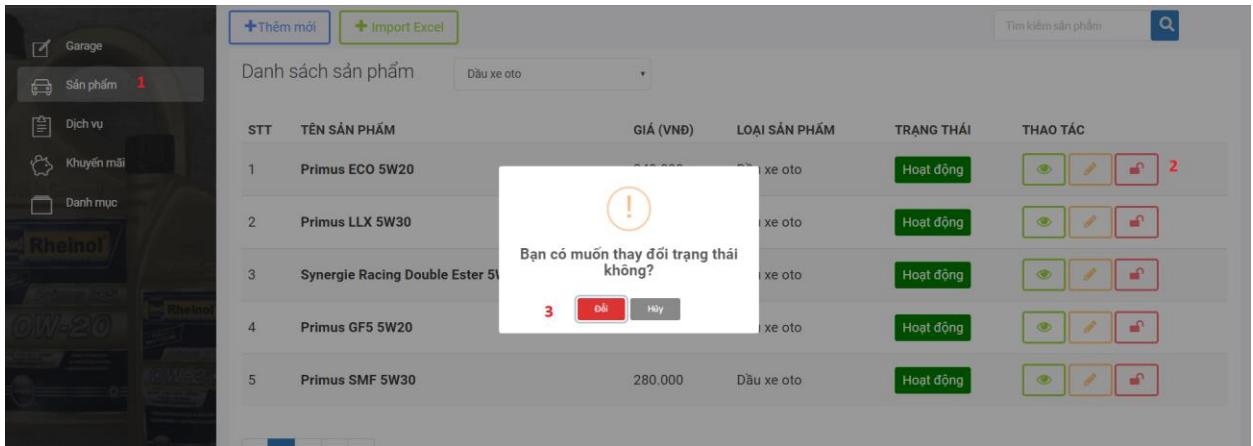


Figure 95 - Disable product

Step	Description
Pre-Condition	Login as Admin Product has status enable
1	Click “Sản phẩm” on the menu bar
2	Click the “lock” icon button
3	Choose “Đổi” button

Table 141 - Disable product steps

2.1.1.5 Add the product by import excel

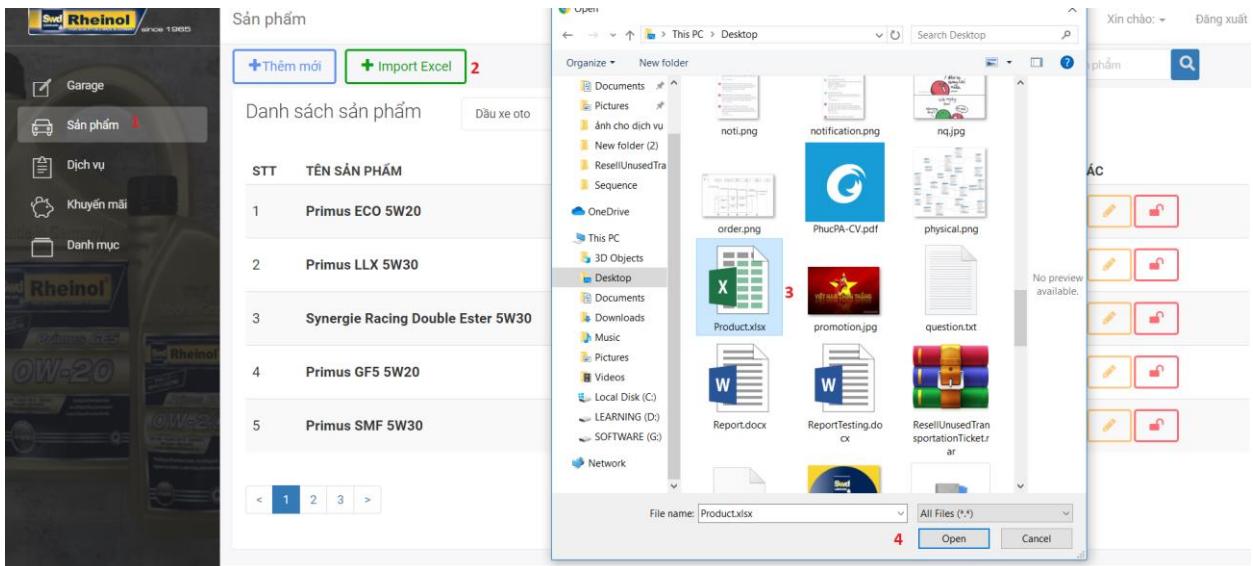


Figure 96 - Add the product by excel

Step	Description
Pre-Condition	Login as Admin Excel must be in the right form
1	Click “Sản phẩm” on the menu bar
2	Click the “Import excel” button
3	Choose excel file
4	Choose an open button

Table 142 - Add the product by excel steps

2.1.2 Service

2.1.2.1 Add service

Figure 97 - Add service

Thêm mới dịch vụ

Tên dịch vụ *	3
Giá cả *	0 4
Thời gian(h)*	0 5
Thời gian bảo hành/ngày)	0 6
Mô tả	7
Hình ảnh *	Chọn hình ảnh 8
Loại dịch vụ *	—Chọn loại dịch vụ— 9
10 Thêm Hủy	

Figure 98 - Add service detail

Step	Description
Pre-Condition	Login as Admin
1	Click “Dịch vụ” on the menu bar
2	Click “Thêm dịch vụ” button
3	Fill the field “Tên dịch vụ”
4	Fill the field “Giá cả”
5	Fill the field “Thời gian”
6	Fill the field “Thời gian bảo hành”
7	Choose “Mô tả” for service
8	Choose “Hình ảnh” for service
9	Choose “Loại dịch vụ”
10	Click “Thêm” button

Table 143 - Add service steps

2.1.2.2 Edit service

STT	TÊN DỊCH VỤ	GIÁ (VNĐ)	THỜI GIAN	LOẠI DỊCH VỤ	TRANG THÁI	THAO TÁC
1	Lọc nhót	400,000đ	4h	Đầu động cơ	Hoạt động	
2	Phù bảo vệ động cơ	500,000đ	4h	Đầu động cơ	Hoạt động	
3	Ốc xà nhót	400,000đ	4h	Đầu động cơ	Vô hiệu hóa	

Figure 99 - Edit service

Tên dịch vụ *	Ốc xá nhớt 3
Giá cả *	400000 4
Thời gian(h)*	4 5
Thời gian bảo hành(ngày)	120 6
Mô tả	Ốc xá nhớt 7



Hình ảnh * **8**

Chọn hình ảnh

Loại dịch vụ *

Dầu động cơ **9**

10 Chỉnh sửa Hủy

Figure 100 - Edit service detail

Step	Description
Pre-Condition	Login as Admin Service must be disable
1	Click “Dịch vụ” on the menu bar
2	Click edit icon button
3	Fill the field “Tên dịch vụ”
4	Fill the field “Giá cả”
5	Fill the field “Thời gian”
6	Fill the field “Thời gian bảo hành”
7	Choose “Mô tả” for service
8	Choose “Hình ảnh” for service
9	Choose “Loại dịch vụ”
10	Click “Chỉnh sửa” button

Table 144 - Edit service steps

2.1.2.3 Search service

STT	TÊN DỊCH VỤ	GIÁ (VNĐ)	THỜI GIAN	LOẠI DỊCH VỤ	TRẠNG THÁI	THAO TÁC
1	Lọc nhớt	400,000đ	4h	Dầu động cơ	Hoạt động	

Figure 101 - Search service

Step	Description
Pre-Condition	Login as Admin
1	Click “Dịch vụ” on the menu bar
2	Input search field with “Tìm kiếm dịch vụ” placeholder
3	Choose the search icon button

Table 145 - Search service steps

2.1.3 Promotion

2.1.3.1 Add promotion

STT	TÊN	GIẢM GIÁ	NGÀY BẮT ĐẦU	NGÀY KẾT THÚC	THỂ LOẠI	TRẠNG THÁI	THAO TÁC
1	Ngày thương binh liệt sĩ	10 %	18-07-2019	03-08-2019		Hoạt động	
2	Mừng tết Trung thu	20 %	01-08-2019	31-08-2019		Hoạt động	
3	Ngày thương binh liệt sĩ	20 %	18-07-2019	31-07-2019		Hoạt động	
4	Mừng ngày Phụ nữ Việt Nam 20/10	15 %	18-07-2019	25-10-2019		Hoạt động	

Figure 102 - Add promotion

Tên khuyến mãi *

Giá trị giảm giá *

Từ ngày *

Đến ngày *

Sản phẩm

Dịch vụ

Hình ảnh *

Loại khuyến mãi *

Đối tượng khuyến mãi * Thành viên thường Thành viên vàng Thành viên bạc Thành viên đồng 11

Mô tả

13

Figure 103 - Add promotion detail

Step	Description
Pre-Condition	Login as Admin
1	Click “Khuyến mãi” on the menu bar
2	Click “Thêm mới” button
3	Fill the field “Tên khuyến mãi”
4	Fill the field “Giá trị giảm giá”
5	Fill the field “Từ ngày”
6	Fill the field “Đến ngày”
7	Choose add product in “Danh sách sản phẩm”
8	Choose add service in “Danh sách dịch vụ”
9	Choose add “Hình ảnh” button
10	Choose “Loại khuyến mãi” combo box
11	Choose “Đối tượng khuyến mãi” checkbox

12	Input “Mô tả” field
13	Click “Thêm mới” button

Table 146 - Add promotion steps

2.1.3.2 Notify promotion

Figure 104 - Notify promotion

Step	Description
Pre-Condition	Login as Admin Promotion is active
1	Click “Khuyến mãi” on the menu bar
2	Click bell icon button

Table 147 - Notify promotion steps

2.1.3.3 Delete promotion

Figure 105 - Delete promotion

Step	Description
Pre-Condition	Login as Admin

	Promotion is disable and startTime of promotion < time now
1	Click “Khuyến mãi” on the menu bar
2	Click cancel icon button
3	Click “Hủy” button

Table 148 - Delete promotion steps

2.1.4 Category

2.1.4.1 Add category

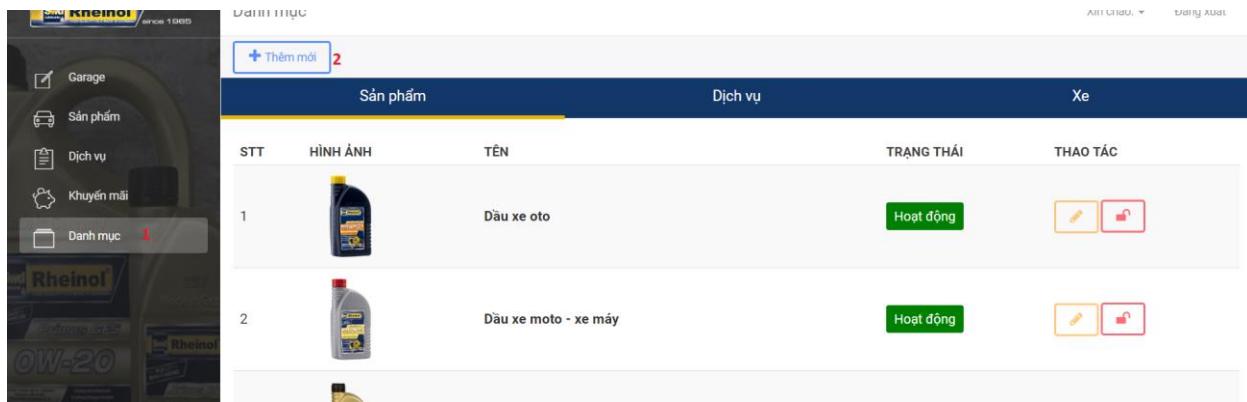


Figure 106 - Add category

Thêm mới thẻ loại

Tên*	<input type="text" value="3"/> 3
Hình ảnh *	<input type="button" value="Chọn hình ảnh"/> 4
Thể loại	<input type="button" value="-- Chọn thể loại --"/> 5
6 <input type="button" value="Thêm mới"/> <input type="button" value="Hủy"/>	

Figure 107 - Add category detail

Step	Description
Pre-Condition	Login as Admin
1	Click “Danh mục” on the menu bar
2	Click “Thêm mới” button

3	Fill the field “Tên”
4	Choose “Hình ảnh” of category
5	Choose “Thể loại” combo box
6	Click “Thêm mới” button

Table 149 - Add category steps

2.1.4.2 Disable category

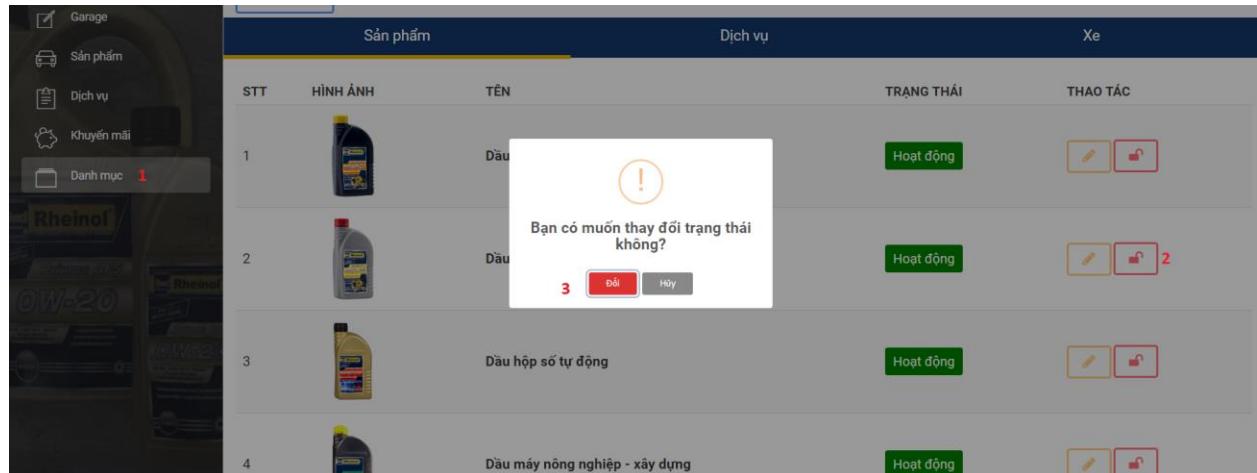


Figure 108 - Disable category

Step	Description
Pre-Condition	Login as Admin Category contains 0 items
1	Click “Danh mục” on the menu bar
2	Click lock icon button
3	Click “Đổi” button

Table 150 - Disable category steps

2.1.5 Garage

2.1.5.1 Edit information garage

The screenshot shows a user interface for editing garage information. On the left is a sidebar with icons for Garage (selected), Sản phẩm, Dịch vụ, Khuyến mãi, and Danh mục. The main area is titled 'Thông Tin Garage'. It contains fields for: 'Tên garage' (Rheinol), 'Giờ bắt đầu làm việc' (08:30 AM), 'Giờ nghỉ làm việc' (05:00 PM), 'Email' (support@rheinolvietnam.com), 'Sức chứa' (4), 'Số điện thoại' (0287300169), 'Địa chỉ' (1 Tô Ký, Quận 12, Thành phố Hồ Chí Minh), and 'Mô tả về garage' (Rheinol kính chào quý khách). A blue 'Chỉnh sửa' button is at the bottom right.

Figure 109 - Edit garage

Step	Description
Pre-Condition	Login as Admin
1	Fill the field “Giờ bắt đầu làm việc”
2	Fill the field “Giờ nghỉ làm việc”
3	Fill the field “Email”
4	Fill the field “Sức chứa”
5	Fill the field “Số điện thoại”
6	Fill the field “Địa chỉ”
7	Fill the field “Mô tả về garage”
8	Choose add service in “Danh sách dịch vụ”
9	Click “Chỉnh sửa” button

Table 151 - Edit garage steps

2.2 Sales clerk

2.2.1 Account

2.2.1.1 Add account

STT	TÊN	EMAIL	SỐ ĐIỆN THOẠI	LOẠI	TRẠNG THÁI	THAO TÁC
1	Kháng vãng lai	rheinol@gmail.com	0993695269	Người dùng	Hoạt động	
2	Le Dinh Thien Vu	vuldt2103@gmail.com	0981875044	Thành viên đồng	Hoạt động	
3	Tấn Trung	tantrung@gmail.com	0938618322	Thành viên đồng	Hoạt động	

Figure 110 - Add account

Thêm mới tài khoản

Tên *	3
Email *	4
Số điện thoại *	5
Địa chỉ	6
Ngày sinh	dd----yyyy 7
Tên đăng nhập *	8
Mật khẩu *	9
Nhập lại mật khẩu	10
11 <input type="button" value="Thêm mới"/> <input type="button" value="Hủy"/>	

Figure 111 - Add account detail

Step	Description
Pre-Condition	Login as Saler
1	Click “Tài khoản” on the menu bar
2	Click “Thêm mới tài khoản” button
3	Fill the field “Tên”
4	Fill the field “Email”
5	Fill the field “Số điện thoại”
6	Fill the field “Địa chỉ”

7	Fill the field “Ngày sinh”
8	Fill the field “Tên đăng nhập”
9	Fill the field “Mật khẩu”
10	Fill the field “Nhập lại mật khẩu”
11	Click “Thêm mới” button

Table 152 - Add account steps

2.2.1.2 Disable account

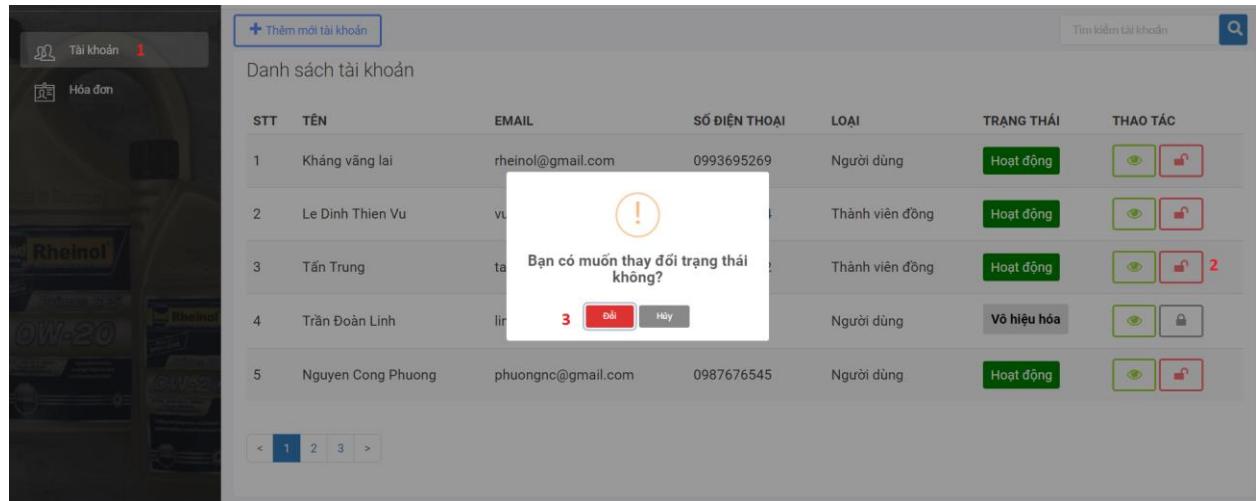


Figure 112 - Disable account

Step	Description
Pre-Condition	Login as Saler
1	Click “Tài khoản” on the menu bar
2	Click lock icon button
3	Click “Đổi” button

Table 153 - Disable account steps

2.2.2 Order

2.2.2.1 Add order

Hóa đơn							Xin chào: -	Đang xuất
<input type="button" value="Thêm đơn hàng"/> 2								
	Mới	Chờ thực hiện	Đang thực hiện	Chờ thanh toán	Đã thanh toán	Hủy		
STT	KHÁCH HÀNG	TỔNG TIỀN (VND)	NGÀY	MÃ ĐƠN	TRẠNG THÁI	THAO TÁC		
1		400.000	18-08-2019	W4F855C5Y1	Mới	<input type="button" value="Xem"/> <input type="button" value="Sửa"/> <input type="button" value="Xoá"/>		
2		400.000	18-08-2019	WLNDW9ZMHS	Mới	<input type="button" value="Xem"/> <input type="button" value="Sửa"/> <input type="button" value="Xoá"/>		
3		400.000	18-08-2019	MNN6PUBHDS	Mới	<input type="button" value="Xem"/> <input type="button" value="Sửa"/> <input type="button" value="Xoá"/>		

Figure 113 - Add order

Thêm mới đơn hàng

Tên Khách hàng: *	<input type="text"/> 3 Nhập tên theo tài khoản khách hàng
Sản phẩm:	<input type="button" value="Danh sách sản phẩm"/> 4
Dịch vụ:	<input type="button" value="Danh sách dịch vụ"/> 5
Tổng số tiền:	0 vnđ
<input type="button" value="Thêm mới"/> 6 <input type="button" value="Hủy"/>	

Figure 114 - Add order detail

Step	Description
Pre-Condition	Login as Saler
1	Click “Hóa đơn” on the menu bar
2	Click “Thêm đơn hàng” button
3	Fill the field “Tên khách hàng”
4	Choose “sản phẩm” for order
5	Choose “dịch vụ” for order
6	Click “Thêm mới” button

Table 154 - Add order steps

2.2.2.2 Edit order

Hóa đơn						
Thêm đơn hàng						
	Mới	Chờ thực hiện	Đang thực hiện	Chờ thanh toán	Đã thanh toán	Hủy
STT	KHÁCH HÀNG	TỔNG TIỀN (VND)	NGÀY	MÃ ĐƠN	TRẠNG THÁI	THAO TÁC
1		400.000	18-08-2019	W4F855C5Y1	Mới	
2		400.000	18-08-2019	WLNDW9ZMHS	Mới	
3		400.000	18-08-2019	MNN6PUBHDS	Mới	
4	Tấn Trung	1.100.000	17-08-2019	WISJFHJ26Q	Mới	

Figure 115 - Edit order

Chỉnh sửa đơn hàng

Tên Khách hàng: * 3

Sản phẩm: 4

Dịch vụ:

Lọc nhót - 400.000 vnđ - 4h ×

Ngày đặt:

Thời gian đặt: 5

Tổng số tiền:

Figure 116 - Edit order detail

Step	Description
Pre-Condition	Login as Saler
1	Click “Hóa đơn” on the menu bar
2	Click “edit” icon button
3	Fill the field “Tên khách hàng”
4	Choose “sản phẩm” for order
5	Click “Chỉnh sửa” button

Table 155 - Edit order steps

2.2.2.3 Cancel order

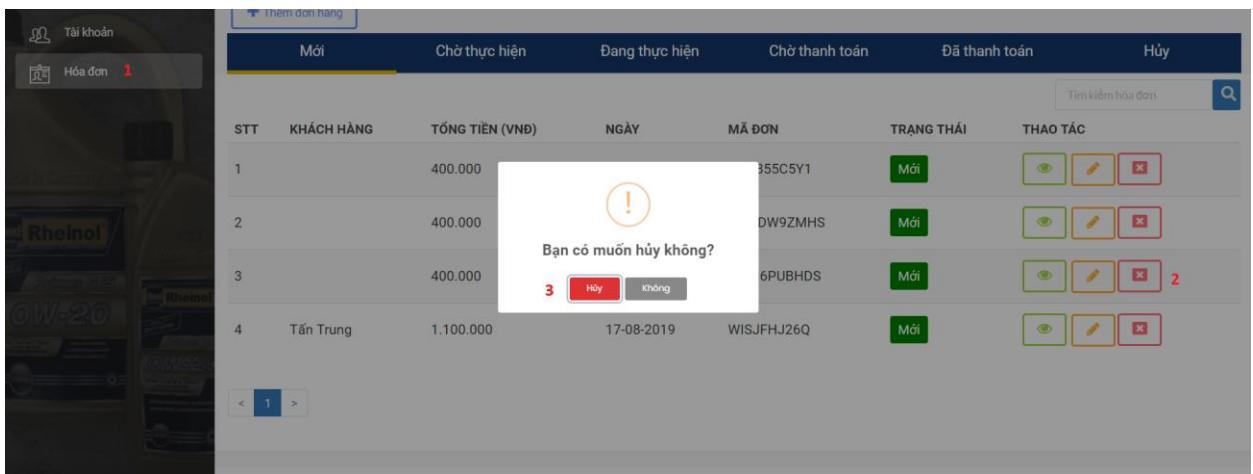


Figure 117 - Cancel order

Step	Description
Pre-Condition	Login as Saler
1	Click “Hóa đơn” on the menu bar
2	Click “Cancel” icon button
3	Click “Hủy” button

Table 156 - Cancel order steps

2.3 Customer

2.3.1 Create order have service

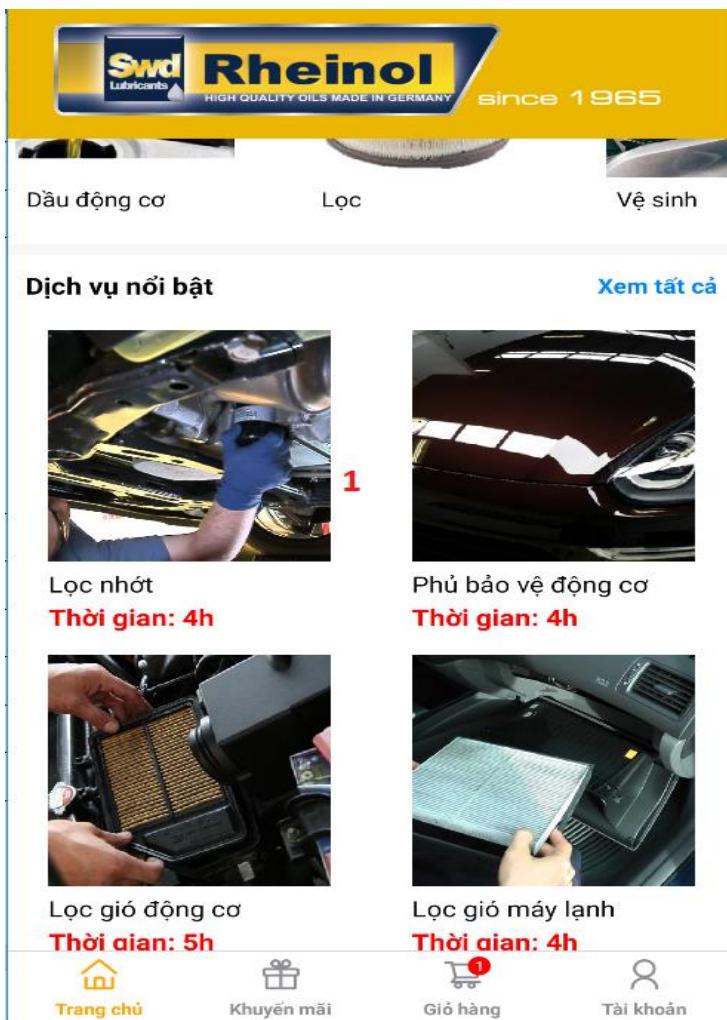


Figure 118 - Create order have service



Figure 119 - Create order have service

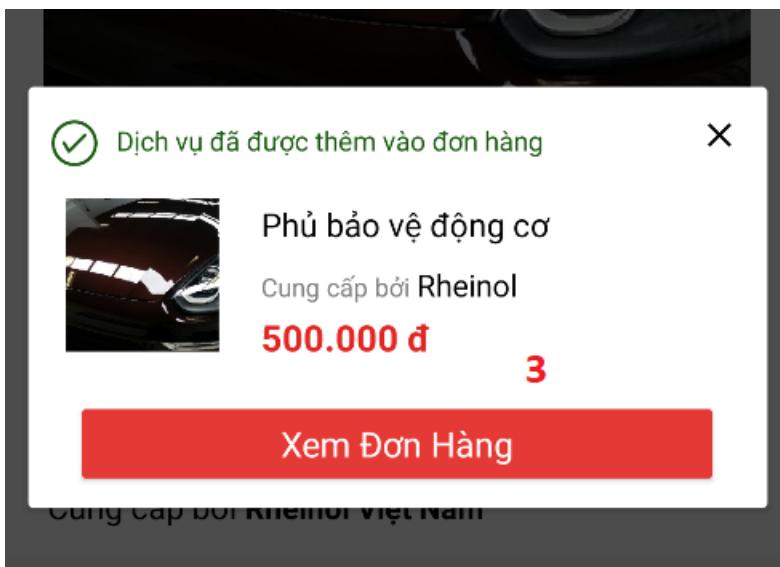


Figure 120 - Create order have service

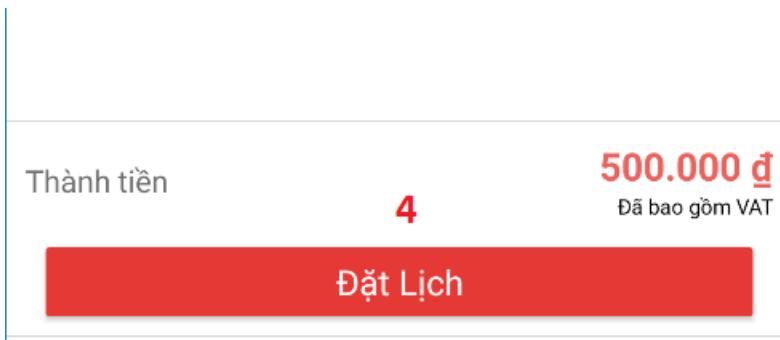


Figure 121 - Create order have service



Figure 122 - Create order have service

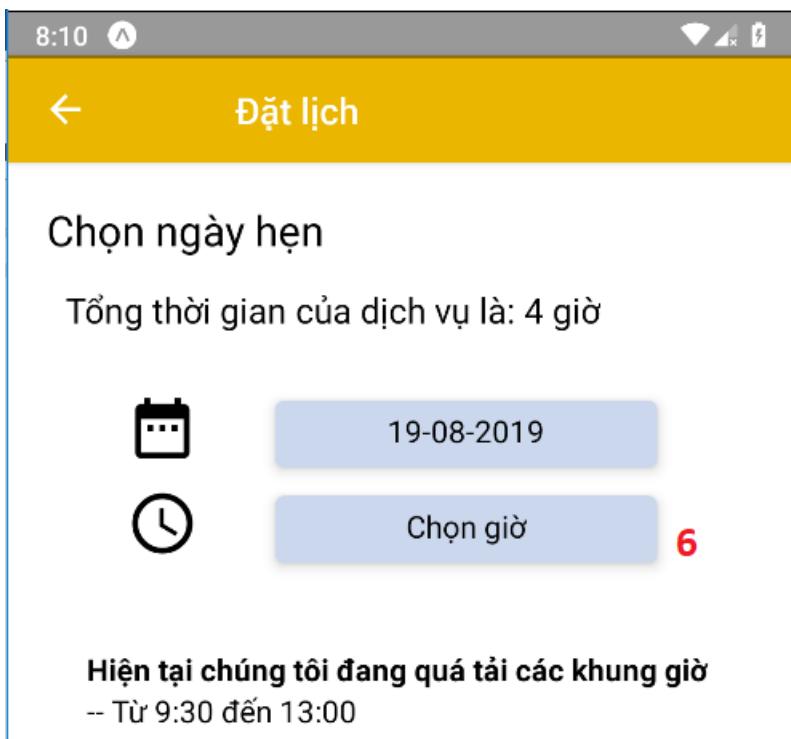


Figure 123 - Create order have service

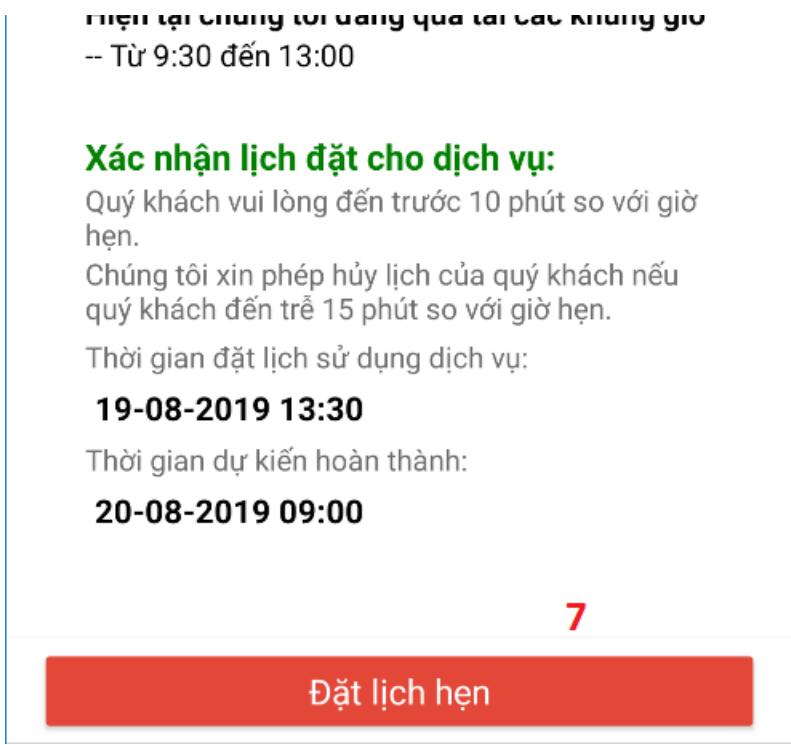


Figure 124 - Create order have service

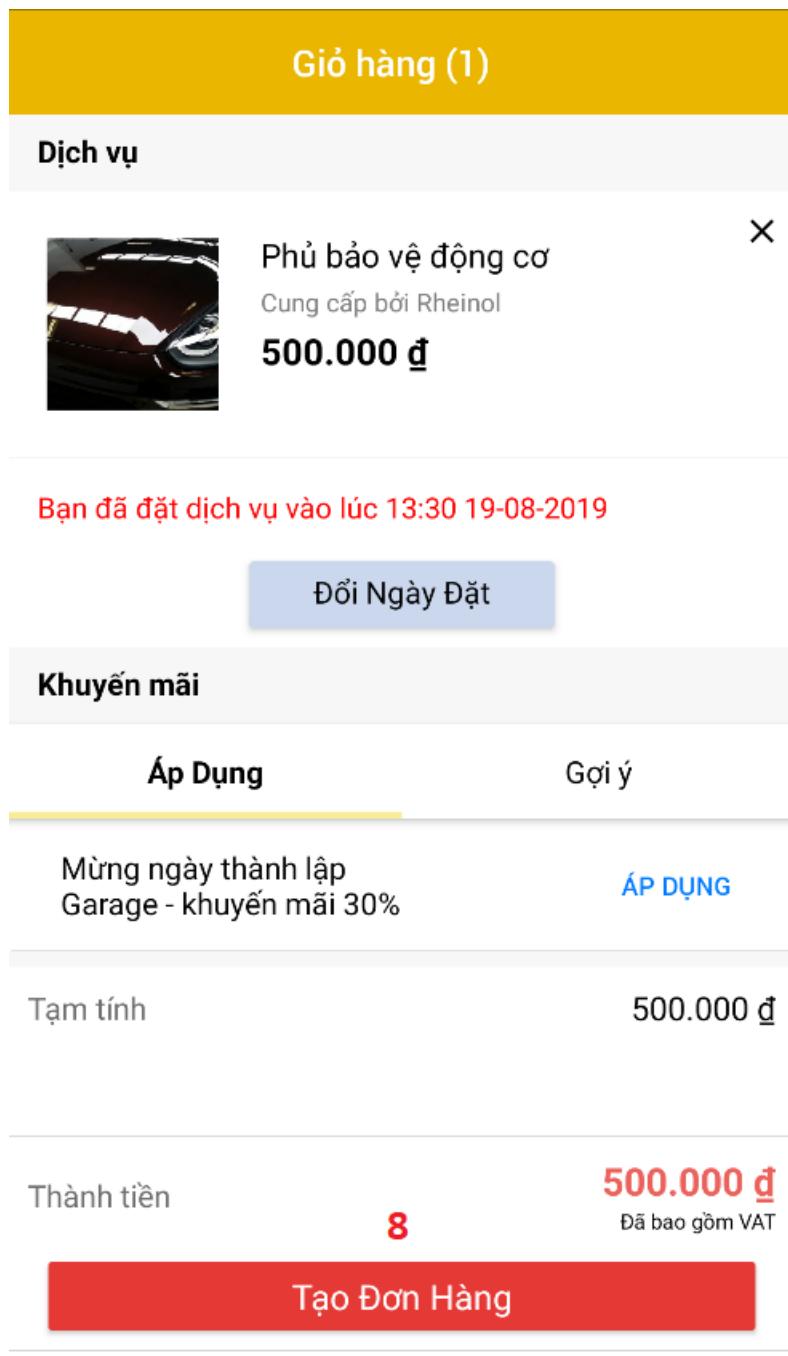


Figure 125 - Create order have service

Step	Description
Pre-Condition	Login as customer
1	Choose service that want book on screen
2	Press “Đặt lịch dịch vụ” button

3	Press “Xem đơn hàng” button
4	Press “Đặt lịch” button
5	Press “Chọn ngày” button
6	Press “Chọn giờ” button
7	Press “Đặt lịch hẹn” button
8	Press “Tạo đơn hàng” button

Table 157 - Create order steps for customer steps

2.3.2 Buy product



Figure 126 - Buy product

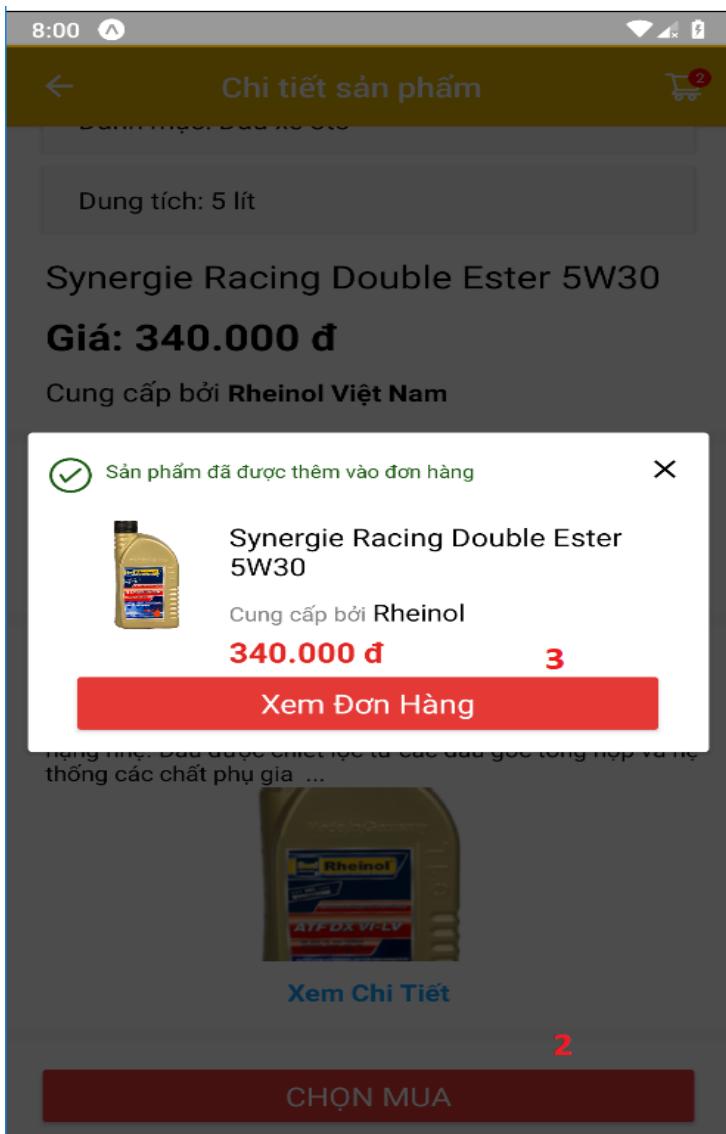


Figure 127 - Buy product

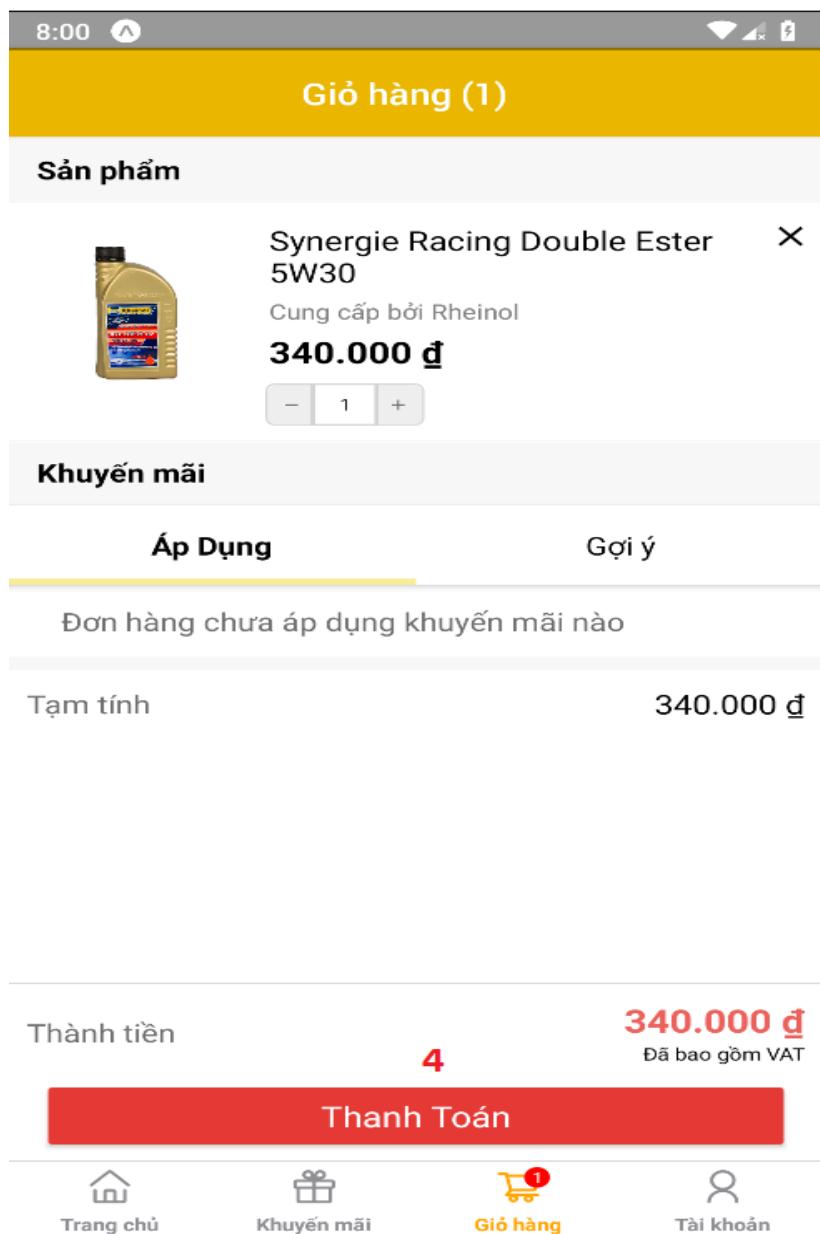


Figure 128 - Buy product

Step	Description
Pre-Condition	Login as customer
1	Choose product that want buy
2	Press “Chọn mua” button
3	Press “Xem đơn hàng” button
4	Press “Thanh toán” button

Table 158 - Buy product steps

2.4 Technical

2.4.1 Confirm order

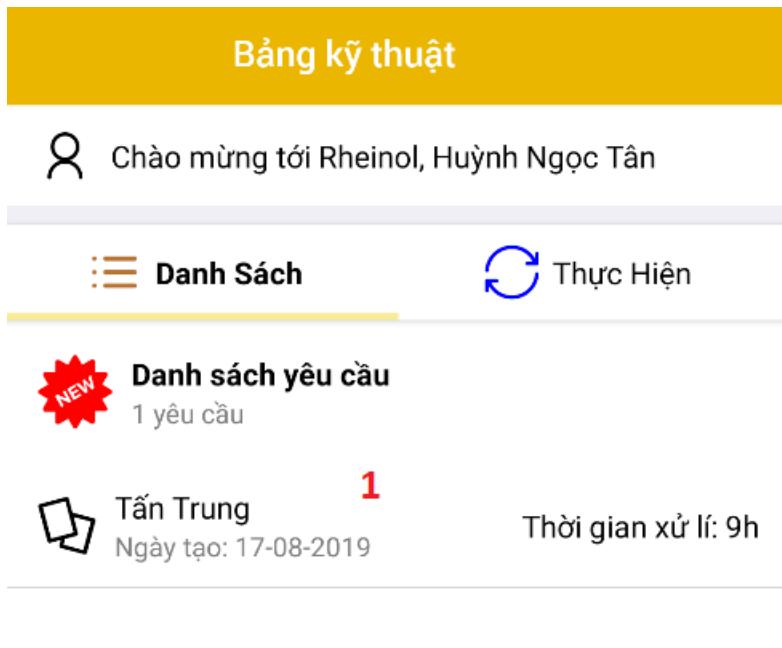


Figure 129 - Confirm service

This screenshot shows a service confirmation screen with a summary table and a note input field. The table lists two items: "Lọc gió máy lạnh" (Air filter machine) at 400.000đ with a note "Thời gian thực hiện: 9h" (Execution time: 9h) and "Lịch hẹn: 19-08-2019" (Scheduled: 19-08-2019); and "Lọc gió động cơ" (Motor air filter) at 700.000đ with a note "Thời gian thực hiện: 9h" (Execution time: 9h) and "Lịch hẹn: 19-08-2019" (Scheduled: 19-08-2019). The total amount "Tổng Tiền" (Total Amount) is highlighted in red as "1.100.000đ". Below the table is a note input field with the placeholder "Nhập mô tả" (Enter description) and a red number "2" indicating the count. A red button at the bottom right contains the text "GỬI MÔ TẢ" (Send description) with a red number "3" indicating the count. The overall layout is organized and user-friendly.

Figure 130 - Confirm service

Step	Description
Pre-Condition	Login as technical
1	Choose service that is assign
2	Fill the fields “Nhập mô tả”
3	Press “Gửi mô tả” button

Table 159 - Confirm service steps