



**MINISTRY OF EDUCATION AND
TRAINING**

Capstone Project Document

Maintenance and Repair Services

MRS Group	
Group Members	Đặng Dương Hùng - 60540 Đương Thị Diễm Kiều – 60471 Trần Minh Thành – 60101
Supervisor	Mr. Nguyễn Trọng Tài
Capstone Project code	MRS

Ho Chi Minh City, 13th May, 2013

Record of Changes

Date	Changed Item	Description	Actor	Version
May-13-2013	Introduction	Create the document	KieuDTD	0.1
May-18-2013	Background Our proposal Problem abstract The current system	Update requirement	KieuDTD	0.2
May-19-2013	Software Project Management Plan (SPMP)	Update requirement	HungDD	0.3
May-21-2013	User Interfaces	Add screen names	HungDD	0.4
May-24-2013	Task sheet	Add task sheet	HungDD	0.5
May-25-2013	Our proposal	Update requirement Format document	KieuDTD	0.6
May-28-2013	System Features (Use case Diagram)	Collect use case diagram	KieuDTD	0.7
July-21-2013	Sequence	Collect sequence diagram	KieuDTD	0.8
Aug-2-2013	Test case	Collect testcase	KieuDTD	0.9
Aug-16-2013	All	Review and update document	HungDD	1.0

Table of Contents

I. Introduction	14
1. Project Information	14
2. Purposes	14
3. The People.....	14
4. Background	14
5. Review of Existing Systems.....	15
6. Our Proposal	15
7. Products.....	16
II. Software Project Management Plan (SPMP)	16
1. Problem Definition.....	16
1.1. Name of this Capstone Project.....	16
1.2. Problem Abstract.....	16
1.3. Project Overview.....	17
2. Project Organization.....	18
2.1. Software Process Model.....	18
2.2. Roles and Responsibilities	20
2.3. Tools and Techniques	20
3. Project Management Plan	21
3.1. Tasks	21
3.2. Task sheet.....	24
3.3. All Meeting Minutes	28
4. Coding convention	29
III. Software Requirements Specifications (SRS).....	29
1. User Requirement Specification	29
1.1. Guest Requirements	29
1.2. Registered User Requirements.....	29
1.3. Customer Requirements	29
1.4. Receptionist Requirements.....	30

1.5.	Staff Requirements.....	30
1.6.	Technical Manager Requirements.....	30
2.	System Requirement Specification (Specific Requirements)	31
2.1.	External Interface Requirement	31
2.2.	System Features	31
2.3.	Software System Attributes.....	188
3.	Entity Relationship Diagram.....	190
IV.	Software Design Description	192
1.	Design Overview.....	192
2.	System Architectural Design.....	193
2.1.	Choice of System Architecture	193
2.2.	Discustion of Alternative Designs.....	194
2.3.	Description of System Interface.....	195
3.	Component Diagram	196
4.	Details Description of Component.....	196
4.1.	Customer Service	196
4.2.	Guest Service	203
4.3.	Staff Service	205
4.4.	Manager Service.....	206
5.	Sequence Diagram	209
5.1.	<Guest> Register.....	209
5.2.	<Guest> View FAQ	211
5.3.	<Guest> [View Equipment] View Equipment by Category	212
5.4.	<Guest> [View Equipment] View Equipment Detail	213
5.5.	<Guest> [View Equipment] Search Equipment.....	214
5.6.	<Guest> Tracking request process	215
5.7.	<Guest> Login	216
5.8.	<Registered User> [Update Profile] Update Basic Information	217
5.9.	<Registered User > [Update Profile] Change Password	218
5.10.	<Registered User> Logout	219
5.11.	<Customer> [Tracking Request] View Request List	219
5.12.	< Customer > [Tracking Request] View Request Detail.....	220

5.13.	< Customer > [Tracking Request] Update Request	221
5.14.	< Staff > [Tracking Task] View All Tasks List	222
5.15.	< Staff > [Tracking Task] View New Tasks List	224
5.16.	< Staff > [Tracking Task] View In-Progress Tasks List	225
5.17.	< Staff > [Tracking Task] View Finished Tasks List.....	226
5.18.	< Staff > [Tracking Task] Filter Task	227
5.19.	< Staff > [Tracking Task] View Task Detail.....	228
5.20.	< Staff > [Tracking Task] Update Task	229
5.21.	< Technical Manager > [Manage FAQ] Create FAQ	230
5.22.	< Technical Manager > [Manage FAQ] View all FAQ list	231
5.23.	< Technical Manager > [Manage FAQ] View FAQ detail	233
5.24.	< Technical Manager > [Manage FAQ] Update FAQ	234
5.25.	< Technical Manager > [Manage repair/maintenance order] Create Request.....	235
5.26.	< Technical Manager > [Manage repair/maintenance order] View All Request List ..	236
5.27.	< Technical Manager > [Manage repair/maintenance order] Filter Request	238
5.28.	< Technical Manager > [Manage repair/maintenance order] View Request Detail....	239
5.29.	< Technical Manager > [Manage repair/maintenance order] Update Request.....	240
5.30.	< Technical Manager > [Manage Task] Create Task	241
5.31.	< Technical Manager > [Manage Task] View All Tasks List	242
5.32.	< Technical Manager > [Manage Task] Filter Task.....	244
5.33.	< Technical Manager > [Manage Task] View Task Detail.....	245
5.34.	< Technical Manager > [Manage Task] Update Task.....	246
5.35.	< Technical Manager > [Manage Symptom] Create Symptom	247
5.36.	< Technical Manager > [Manage Symptom] View All Symptom List	248
5.37.	< Technical Manager > [Manage Symptom] View Symptom Detail	249
5.38.	< Technical Manager > [Manage Symptom] Update Symptom	250
5.39.	< Technical Manager > [Manage Equipment] Add Equipment	251
5.40.	< Technical Manager > [Manage Equipment] View All Equipment List	252
5.41.	< Technical Manager > [Manage Equipment] View Equipment Detail	253
5.42.	< Technical Manager > [Manage Equipment] Update Equipment	254
5.43.	< Technical Manager > [Statistics] Statistics by Request Type.....	255
5.44.	< Technical Manager > [Statistics] Statistics Top Fixed Model	255

5.45.	< Technical Manager > [Statistics] Statistics by Symptom	256
5.46.	< Technical Manager > [Manage User] Add new Account	257
5.47.	< Technical Manager> [Manage User] View List of Accounts	258
5.48.	<Technical Manager> [Manage User] Filter Account	259
5.49.	< Technical Manager> [Manage User] Update Account Information	260
5.50.	< Technical Manager> [Manage Product] Add Product Category.....	262
5.51.	< Technical Manager> [Manage Product] View All Product Category List.....	263
5.52.	< Technical Manager> [Manage Product] Update Product Category	264
5.53.	< Technical Manager> [Manage Product] Add Product Prototype.....	266
5.54.	< Technical Manager> [Manage Product] View All Product Prototype List.....	267
5.55.	< Technical Manager> [Manage Product] Update Product Prototype	268
5.56.	< Technical Manager> [Manage Product] Add Product Model.....	270
5.57.	< Technical Manager> [Manage Product] View All Product Model List.....	271
5.58.	< Technical Manager> [Manage Product] Update Product Model	272
5.59.	< Technical Manager> [Manage Invoice] View Finished Request.....	274
5.60.	< Technical Manager> [Manage Invoice] View Invoice	275
5.61.	< Technical Manager> [Manage Invoice] Update Invoice	277
5.62.	< Technical Manager> [Manage Promotion] View All Promotion List	278
5.63.	< Technical Manager> [Manage Promotion] Create Promotion.....	280
5.64.	< Technical Manager> [Manage Promotion] View Promotion Detail.....	281
5.65.	< Technical Manager> [Manage Promotion] Update Promotion.....	282
5.66.	< System> Auto Notification	282
5.67.	< System> Auto Check Missed Deadline Request / Task.....	283
6.	User Interface Design.....	283
6.1.	Page Flow Diagram.....	283
6.2.	Detail Design of the User Interface.....	283
7.	Database Design.....	284
7.1.	Database Diagram	284
7.2.	Relations Schema	285
7.3.	Relations Schema Details.....	286
7.4.	Database Implementation	295
V.	Software Test Documentation	295

1.	Introduction.....	295
2.	Test Approach.....	295
3.	Test Plan.....	296
3.1.	Features to be tested	296
3.2.	Features not to be tested.....	296
3.3.	Test Tool and Environment.....	296
4.	Test Cases	297
4.1.	Customer - [Tracking Request] View Request List	297
4.2.	Customer - [Tracking Request] Update Request.....	298
4.3.	Customer - [Tracking Request] Push/Receive Notification	300
4.4.	Staff - [Tracking Task] View All Tasks List.....	300
4.5.	Staff - [Tracking Task] Filter Task	300
4.6.	Staff - [Tracking Task] View Task Detail.....	301
4.7.	Staff - [Tracking Task] Update Task.....	303
4.8.	Staff - [Tracking Task] Push/Receive Notification.....	304
4.9.	Technical Manager - [Manage FAQ] Create FAQ.....	305
4.10.	Technical Manager - [Manage FAQ] View all FAQ list.....	305
4.11.	Technical Manager - [Manage FAQ] View FAQ detail.....	305
4.12.	Technical Manager - [Manage FAQ] Update FAQ.....	305
4.13.	Technical Manager - [Manage repair/maintenance order] Create Request.....	305
4.14.	Technical Manager - [Manage repair/maintenance order] View All Request List	307
4.15.	Technical Manager - [Manage repair/maintenance order] Filter Request.....	308
4.16.	Technical Manager - [Manage repair/maintenance order] View Request Detail	309
4.17.	Technical Manager - [Manage repair/maintenance order] Update Request.....	310
4.18.	Technical Manager - [Manage repair/maintenance order] Push/Receive Notification	311
4.19.	Technical Manager - [Manage Task] Create Task	311
4.20.	Technical Manager - [Manage Task] View All Tasks List.....	313
4.21.	Technical Manager - [Manage Task] Filter Task	314
4.22.	Technical Manager - [Manage Task] View Task Detail	315
4.23.	Technical Manager - [Manage Task] Update Task	316
4.24.	Technical Manager - [Manage Task] Push/Receive Notification	317
4.25.	Technical Manager - [Manage Symptom] Create Symptom.....	317

4.26.	Technical Manager - [Manage Symptom] View All Symptom List	317
4.27.	Technical Manager - [Manage Symptom] View Symptom Detail.....	317
4.28.	Technical Manager - [Manage Symptom] Update Symptom	317
4.29.	Technical Manager - [Manage Equipment] Add Equipment	317
4.30.	Technical Manager - [Manage Equipment] View All Equipment List	317
4.31.	Technical Manager - [Manage Equipment] View Equipment Detail.....	317
4.32.	Technical Manager - [Manage Equipment] Update Equipment.....	317
4.33.	Technical Manager - [Statistics] Statistics by request type.....	317
4.34.	Technical Manager - [Statistics] Statistics by top fixed model.....	317
4.35.	Technical Manager - [Statistics] Statistics by symptom	317
4.36.	Technical Manager - [Manage User] Add new Account	318
4.37.	Technical Manager - [Manage User] View List of Accounts	320
4.38.	Technical Manager - [Manage User] Update Account Information	321
5.	Checklists	321
5.1.	Checklist of Validation	321
5.2.	Submission Checklist.....	322
6.	Other material	322

VI. Installation Guide 322

1.	Setting up the environment at server side	322
1.1.	Hardware requirements	322
1.2.	Software requirments	322
2.	Deployment at server side.....	322
2.1.	Prepare deployment package.....	322
2.2.	Deploy database	322
2.3.	Deploy web application.....	323
2.4.	Change configurations	324
3.	Setting up the environment at client side	324
4.	User's Guide	325
4.1.	Guest's Guide.....	325
4.2.	Registered User's Guide	331
4.3.	Customer's Guide.....	335
4.4.	Staff's Guide	337

4.5. Technical manager's Guide.....	340
-------------------------------------	-----

Tables

Table 1-Member of Project	20
Table 2-Task Plan	24
Table 3-Tasks sheet.....	28
Table 4-Meeting minutes	29
Table 5- User Interfaces	31
Table 6- Actors.....	40
Table 7-Entity diagram description.....	190
Table 8-BaseAction class diagram explanation	198
Table 9-CustomerAction class diagram explanation	198
Table 10-DAOFactory class diagram explanation	199
Table 11-HibernateDAOFactory class diagram explanation	200
Table 12-IGenericDAO class diagram explanation	201
Table 13-GenericDAO class diagram explanation.....	202
Table 14-HibernateUtil class diagram explanation.....	202
Table 15-IProductrequestDAO class diagram explanation	202
Table 16-ProductrequestDAO class diagram explanation	203
Table 17-GuestAction class diagram explanation.....	204
Table 18-IProductprototypeDAO class diagram explanation	204
Table 19-StaffAction class diagram explanation	206
Table 20-ITaskDAO class diagram explanation	206
Table 21-TaskDAO class diagram explanation	206
Table 22-IUsedequipmentDAO	206
Table 23-UsedequipmentDAO.....	206
Table 24-MangerAction class diagram explanation.....	209
Table 25-Relationship Schema description.....	285
Table 26-Table Account data dictionary	286
Table 27-Table Profile data dictionary	286
Table 28-Table Notification data dictionary	287
Table 29-Table Roles data dictionary	287
Table 30-Table ProductRequest data dictionary	288
Table 31-Table Invoice data dictionary	289
Table 32-Table ProductPrototype data dictionary	289
Table 33-Table ProductCategory data dictionary	289
Table 34-Table Equipment data dictionary	290
Table 35-Table Image data dictionary	290
Table 36-Table EquipmentCategory data dictionary	290

Table 37-Table Symptom data dictionary	291
Table 38-Table FAQ data dictionary	291
Table 39-Table Task data dictionary.....	291
Table 40-Table TaskStaff data dictionary.....	292
Table 41-Table PromotionEvent data dictionary	292
Table 42-Table EquipmentPromotion data dictionary	292
Table 43-Table UsedEquipment data dictionary.....	293
Table 44-Table EstimateEquipment data dictionary	293
Table 45-Table ProductRequestEquipmentTag data dictionary	293
Table 46-Table ProductRequestSymptom data dictionary.....	294
Table 47-Table StaffSkill data dictionary	294

List of Figures

Figure 1-Software Process Model.....	19
Figure 2-Overall Use case diagram.....	32
Figure 3-Guest's use case diagram	33
Figure 4-Registered User's use case diagram	34
Figure 5-Customer's use case diagram	35
Figure 6-Staff's use case diagram	36
Figure 7-Technical manager's use case diagram	38
Figure 8 - Guest Register usecase diagram	40
Figure 9 – Guest View FAQ usecase diagram	43
Figure 10-View Equipment use case diagram.....	44
Figure 11- View Equipment By Category	45
Figure 12- View Equipment Detail usecase diagram.....	46
Figure 13 - Search Equipment usecase diagram	47
Figure 14 - Tracking Request Process usecase diagram	49
Figure 15 – Login usecase diagram	50
Figure 16-Update Profile use case diagram	53
Figure 17 - Update Basic Information usecase diagram	53
Figure 18 - Change Password usecase diagram	55
Figure 19 – Logout usecase diagram	57
Figure 20 - Customer View Request List usecase diagram	58
Figure 21 - Customer View Request Detail usecase diagram	60
Figure 22 - Customer Update Request usecase diagram.....	62
Figure 23 - Staff View All Task List usecase diagram	64
Figure 24 - Staff Filter Task usecase diagram.....	70
Figure 25 - Staff View Task Detail usecase diagram.....	72
Figure 26 - Staff Update Task usecase diagram.....	74
Figure 27 - Manager Mange FAQ usecase diagram	77
Figure 28-Manager Create FAQ usecase diagram	78

Figure 29 - Manager View All FAQ List usecase diagram.....	80
Figure 30 - Manager View FAQ Detail usecase diagram	81
Figure 31 - Manager Update FAQ usecase diagram.....	83
Figure 32- Manger Create Request use case diagram	85
Figure 33-Manager Create Request usecase diagram	85
Figure 34-Manager Create Request List usecase diagram	88
Figure 35-Manager Filter Request usecase diagram	90
Figure 36-Manger View Request Detail usecase diagram	91
Figure 37-Manger Update Request usecase diagram.....	94
Figure 38-Manager Manage Task use case diagram.....	97
Figure 39-Manger Create Task usecase diagram.....	97
Figure 40-Manger View All Task List usecase diagram	100
Figure 41-Manger Filter Task usecase diagram.....	101
Figure 42-Manger View Task Detail usecase digram.....	103
Figure 43-Manger Update Task usecase dgram.....	105
Figure 44-Manager Manage Symptom use case diagram	108
Figure 45-Manger Create Symptom usecase diagram	108
Figure 46-Manger View All Symptom List usecase dgram.....	110
Figure 47-Manger View Symptom Detail usecase diagram	112
Figure 48-Manger Update Symptom usecase dgram	114
Figure 49-Manger Manage Equipment use case diagram	117
Figure 50-Manger Add Equipment usecase diagram.....	117
Figure 51-Manger View All Request List usecase diagram	120
Figure 52-Manager View Equipment Detail usecase dgram	122
Figure 53-Manger Update Equipment usecase diagram	124
Figure 54-Statistics use case diagram	127
Figure 55-Statistic by request type usecase dgram	127
Figure 56-Statistic Top fix model usecase diagram.....	129
Figure 57-Statistic by symptom usecase dgram.....	131
Figure 58-Manger Manage User use case diagram.....	133
Figure 59-Manger Add New Account usecase diagram	133
Figure 60-Manger View List Of Account usecase diagram	136
Figure 61-Mager Filter Account usecase dgram	138
Figure 62-Manger Update Account usecase diagram	139
Figure 63-Manage Product Category use case diagram.....	141
Figure 64-Manger View All Product Category usecase diagram	144
Figure 65-Manger Update Product Prototype usecase dgram.....	145
Figure 66-Manage Product Prototype use case diagram	148
Figure 67-Manger Add Product Prototype usecase dgram	148
Figure 68-Manager View All Product Prototype usecase dgram.....	150
Figure 69-Manger Update Product Prototype usecase dgram.....	152
Figure 70-Manage Product Model use case diagram	154
Figure 71-Manger Add Product Model usecase diagram	155
Figure 72-Manger View All Product Model.....	157

Figure 73-Manger Update Product Model	159
Figure 74-Manage Invoice use case diagram.....	161
Figure 75-Manger View Finish Request List usecase digram	161
Figure 76-Manger View Invoice usecase digram	163
Figure 77-Update Invoice usecase digram.....	165
Figure 78-Manger Mange Promotion usecase diagram	168
Figure 79-Manger View All Promotion List.....	168
Figure 80-Manager Create Promotion use case diagram	170
Figure 81-Manger Update Promotion usecase diagram.....	173
Figure 82-Manger View Promotion Detail usecase diagram	176
Figure 83-System Auto Send Mail usecase diagram	179
Figure 84-System Auto Check Task/ Request Missed Deadline	181
Figure 85-System Recommend Staff For Create Task usecase diagram	182
Figure 86-System Auto Notification usecase diagram.....	184
Figure 87-System Auto Send SMS usecase diagram	186
Figure 88-System Estimate range of repair cost / hour	187
Figure 89-Entity Relationship diagram.....	191
Figure 90 - System Architecture	193
Figure 91 – MVP Model	194
Figure 92 – MVVM Model	195
Figure 93-Component Diagram	196
Figure 94-Customer Service Class Diagram.....	197
Figure 95-Guest Service Class Diagram	203
Figure 96-Staff Service Class Diagram.....	205
Figure 97-Manger Service Class Diagram.....	207
Figure 98-Register sequence diagram.....	210
Figure 99-Guest View FAQ sequence diagram	211
Figure 100-Guest View Equipment By Category sequence diagram.....	212
Figure 101- View Equipment Detail sequence diagram	213
Figure 102-Guest Search Equipment	214
Figure 103-Guest Tracking Request Process sequence diagram.....	215
Figure 104-Login sequence diagram.....	216
Figure 105-Load Customer CP Page sequence diagram	217
Figure 106-Update Basic Information sequence diagram.....	217
Figure 107-Registered User Change Password sequence diagram	218
Figure 108-Registered User Logout sequence diagram	219
Figure 109-Customer View Request List sequence diagram	219
Figure 110-Customer View Request List sequence diagram	220
Figure 111-Customer Update Request sequence diagram	221
Figure 112-Customer View All Task List sequence diagram	222
Figure 113-Customer Load Task Equipment sequence diagram	223
Figure 114-Staff View New Tasks List sequence diagram.....	224
Figure 115-Staff View In-progress Tasks List sequence diagram	225
Figure 116-Staff View Finished Tasks List sequence diagram.....	226

Figure 117-Staff Filter Task sequence diagram	227
Figure 118-Staff View Task Detail sequence diagram	228
Figure 119-Staff Update Task sequence diagram	229
Figure 120-Technical Manager Create FAQ sequence diagram	230
Figure 121-Technical Manager View All FAQs List sequence diagram	231
Figure 122-Technical Manager Load FAQ Form sequence diagram.....	232
Figure 123-Technical Manager View FAQ Detail sequence diagram	233
Figure 124-Technical Manager Update FAQ sequence diagram.....	234
Figure 125-Technical Manager Create Request sequence diagram	235
Figure 126-Technical Manager View All Request List sequence diagram.....	236
Figure 127-Technical Manager Load Request Form sequence diagram.....	237
Figure 128-Technical Manager Filter Request sequence diagram.....	238
Figure 129-Technical Manager View Request Detail sequence diagram	239
Figure 130-Technical Manager Update Request sequence diagram	240
Figure 131-Technical Manager Create Task sequence diagram	241
Figure 132-Technical Manager View All Task List sequence diagram.....	242
Figure 133-Technical Manager Load Task Form sequence diagram.....	243
Figure 134-Technical Manager Filter Task sequence diagram.....	244
Figure 135-Technical Manager View Task Detail sequence diagram	245
Figure 136-Technical Manager Update Task sequence diagram	246
Figure 137-Technical Manager Create Symptom sequence diagram	247
Figure 138-Technical Manager View All Symptom List sequence diagram.....	248
Figure 139-Technical Manager Load Symptom Form sequence diagram	249
Figure 140-Technical Manager View Symptom Detail sequence diagram	249
Figure 141-Technical Manager Update Symptom sequence diagram	250
Figure 142-Technical Manager Add Equipment sequence diagram	251
Figure 143-Technical Manager View All Equipment List sequence diagram	252
Figure 144-Technical Manager View Equipment Detail sequence diagram	253
Figure 145-Technical Manager Update Equipment sequence diagram	254
Figure 146-Technical Manager Statistics Maintenance of Repair sequence diagram	255
Figure 147-Technical Manager Statistics Top Product Request sequence diagram	255
Figure 148-Technical Manager Add New Account sequence diagram	257
Figure 149-Technical Manager View List of Accounts sequence diagram	258
Figure 150-Technical Manager Filter Account sequence diagram	259
Figure 151-Technical Manager View Account Detail sequence diagram	260
Figure 152-Technical Manager Update Account Information sequence diagram	261
Figure 153-Technical Manager Add Product Category sequence diagram	262
Figure 154-Technical Manager View All Product Category List sequence diagram	263
Figure 155-Technical Manager View Product Category Detail sequence diagram	264
Figure 156-Technical Manager Update Product Category sequence diagram.....	265
Figure 157-Technical Manager Add Product Prototype sequence diagram.....	266
Figure 158-Technical Manager View All Product Prototype List sequence diagram.....	267
Figure 159-Technical Manager View Product Prototype Detail sequence diagram	268
Figure 160-Technical Manager Update Product Prototype sequence diagram	269

Figure 161-Technical Manager Add Product Model sequence diagram.....	270
Figure 162-Technical Manager View All Product Model List sequence diagram.....	271
Figure 163-Technical Manager View Product Model Detail sequence diagram	272
Figure 164-Technical Manager Update Product Model sequence diagram	273
Figure 165-Technical Manager View All Promotion List sequence diagram.....	278
Figure 166-Technical Manager Load Promotion Form sequence diagram.....	279
Figure 167-Technical Manager Filter Promotion sequence diagram Error! Bookmark not defined.	
Figure 168-Technical Manager Create Promotion sequence diagram	280
Figure 169-Technical Manager View Promotion Detail sequence diagram	281
Figure 170-Technical Manager Update Promotion sequence diagram ... Error! Bookmark not defined.	
Figure 171 Logical Diagram	285
Figure 172 - Physical Diagram	295
Figure 173 - Prepare Deployment Package.....	322
Figure 174-Deploy Database (1).....	323
Figure 175-Deploy Database (2).....	323
Figure 176 - Change database connection configuration (1)	324
Figure 177 - Change database connection configuration (2)	324
Figure 178 - Click Register button.....	325
Figure 179 - Register Page.....	326
Figure 180 - Click FAQ tab	327
Figure 181 - FAQ Page	328
Figure 182 - Click Equipment tab.....	328
Figure 183 - Equipment Page.....	329
Figure 184 - Equipment Detail.....	330
Figure 185 - Search Equipment.....	331
Figure 186 - Login at Home Page	331
Figure 187 - Login at Register Page	332
Figure 188 - Login success	332
Figure 189 - Click Profile	333
Figure 190 - Update Profile.....	333
Figure 191 - Link Change Password.....	334
Figure 192 - Change Password	334
Figure 193- Click Logout link.....	335
Figure 194 - Redirect to Home Page.....	335
Figure 195 - Click Request box	335
Figure 196 - Customer Tracking Request Page	336
Figure 197 - Request Detail	336
Figure 198-Manager Request List.....	341
Figure 199-Manager Add Request	342

I. Introduction

1. Project Information

- Project name: **Maintenance and Repair Services**
- Project code: **MRS**
- Product type: **After-sale supporting website**
- Timeline: **From 8th May 2012 to 25th Aug 2012**

2. Purposes

This project is registered and implemented as the capstone project for the group number 2. The first purpose is to fulfill the requirements from FPT University studying program. The second purpose is to create a complete product for going live.

3. The People

Supervisor:

Full name	Phone	E-Mail	Title
Nguyễn Trọng Tài	0903346184	taint@fpt.edu.vn	Lecturer

Team members:

	Full name	Student code	Phone	E-mail	Role in Group
Student 1	Đặng Dương Hùng	60540	0964987678	hungdd60540@fpt.edu.vn	Leader
Student 2	Dương Thị Diễm Kiều	60471	0975252206	kieudtd60471@fpt.edu.vn	Member
Student 3	Trần Minh Thành	60101	0935024987	thanhtm60101@fpt.edu.vn	Member

4. Background

In Vietnam, customers buy any products from sale department and always have maintenance and repairing needs. They directly bring products to maintain/repair and receive the appointment letter. In some center, this usual way has some problems:

- Customer cannot know the maintenance and repair process. There are some cases that products are not repaired completely at the appointment date.
- In case the product is damaged seriously, customer is not reported estimate the cost of repairing at the beginning.

- History of maintenance/repairing is not stored to support the next time.
- Most of current centers using handwritten receipt or invoice. It is hard for statistics or reporting later.
- Managers in center hard to decide which staff is the best matched with customer request (staff skills, current task, experience...)
- Managers in center hard to know which staffs missed task deadline, manages time of task of staffs is not efficient.
- Managers in center are not notified which tasks are nearly deadline or missed deadline.

These problems are not overcome because the after-sale supporting system is not interested in developing. Nowadays, it is the solution in any sales operation. It makes services reliable and professional. MRS system is built to meet the need of above:

- Customers bring products to maintain or repair. They are provided an account (or created by themselves), which is used to log in the website for checking the repairing process. Therefore, tracking progress becomes more explicit.
- Customer can cancel the order of maintenance or repairing whenever the cost of damage estimating is not worthy.
- Damage symptoms and repairing experiences are collected and shared to the customer. It helps customers get more experiences and estimate damage by themselves.
- Task assignment is recommended for managers of service center. This function is very useful for right people, right work. This function makes work active and effective.
- Beside of manage CRM, it also help internal use in service center such as managers can manage service request of customer, assign and manage task of staffs, get or make notification when have some changes to related people, report and statistic, export invoice for customer...

5. Review of Existing Systems

According to research in Vietnam, the website focus on supporting after-sale for user is not common. They are almost e-commerce websites that just support sale department.

Existing systems that are used to maintain/repair are not focused in the website system. It still works with manual procedure.

6. Our Proposal

Our website will provide those functions to specific objects:

- Customer:
 - Tracking the maintenance/repairing process.
 - View maintenance/repairing history of product.
 - View and search equipment items
 - View FAQ about common damage symptoms and preparing experiences.

- Automatically notify to customer via website or mail whenever request status is updated
- Internal user of service center:
 - Manage service requests from customer.
 - Assign tasks for staff.
 - Staff can tracking task.
 - Manage equipment items.
 - Update common damage symptom and preparing experiences
 - Automatically notify to staff when assigning task for them.
 - Automatically notify to technical manager when maintenance/repairing process is updated or tasks of staff is updated.
 - Statistic and report
 - Export invoice or receipt for customer.
 - Manage user of system.

7. Products

The main product of this project is a website as mentioned above. The website supports after-sales.

II. Software Project Management Plan (SPMP)

1. Problem Definition

1.1. Name of this Capstone Project

English: Maintenance and repair services

Vietnamese: Quản lý dịch vụ bảo hành và sửa chữa

Abbreviation: MRS Website

1.2. Problem Abstract

In Vietnam, website supporting after-sale operations is not common. There are almost e-commerce websites that just support sale operations. Existing systems that are used to maintain/repair are almost work with manual procedure and do not build in the website system.

MRS system for after-sale operations supports not only Technical manager to manage maintenance/repairing process and tasks of staff but also Customer to track product's status.

1.3. Project Overview

1.3.1. The Current System

In the usual way, the customers directly bring products to maintain/repair and receive the appointment letter. This way has some problems:

- Customer cannot know the maintenance and repair process. There are some cases that products are not repaired completely at the appointment date.
- In case the product is damaged seriously, customer is not reported estimate the cost of repairing at the beginning.
- History of maintenance/repairing is not stored to support the next time.
- Most of current centers using handwritten receipt or invoice. It is hard for statistics or reporting later.
- Managers in center hard to decide which staff is the best matched with customer request (staff skills, current task, experience...)
- Managers in center hard to know which staffs missed task deadline, manages time of task of staffs is not efficient.
- Managers in center are not notified which tasks are nearly deadline or missed deadline

1.3.2. The Proposed System

- The web-based application is built for product supporting after-sales. It should support:
 - The customer can track their request for repair or maintenance.
 - The staffs of service center can receive the requests, estimate repair costs, or dispatch the repaired product to the dealer to change.
 - The service center can manage the equipment item.
- This Web-based application should have the following features:
 - Users management
 - Tracking service request
 - Manage tasks of staffs in service center
 - Equipment management
 - Searching, statistic
 - Automatically notify when have some changes

1.3.3. Boundaries of the System

- English user-interface
- Based on web application
- Allocate suitable roles for each kind of users

1.3.4. Development Environment

Hardware:

- Personal computers for developing with the minimum configuration: CPU Core 2 Duo 2.0GHz, 2GB of RAM, 120GB of hard disk, and internet.
- Browsers support HTML5, enable javascript.

Software:

- Microsoft Windows 7: operating system and platform for development
- Netbeans 7.3: IDE used to code system modules
- MySQL Workbench 5.2: used to create and manage the database for system
- Google Code and TortoiseSVN: used to control source code and documents of project
- Apache Tomcat: web server
- Software Ideas Modeler and Visio: used to create models and diagrams
- Microsoft Project 2010: used to manage process and work schedules.
- Skype: used for communication and meeting.

2. Project Organization

2.1. Software Process Model

- With the schedule of weekly reports for every stage, the software will be developed by using waterfall model which is very simple and require minimal resource for implementation. In the waterfall approach, the whole process of software development is divided into separate phases. These phases in the model are:
 - Requirement specifications phase
 - Software design
 - Implementation
 - Testing and maintenance

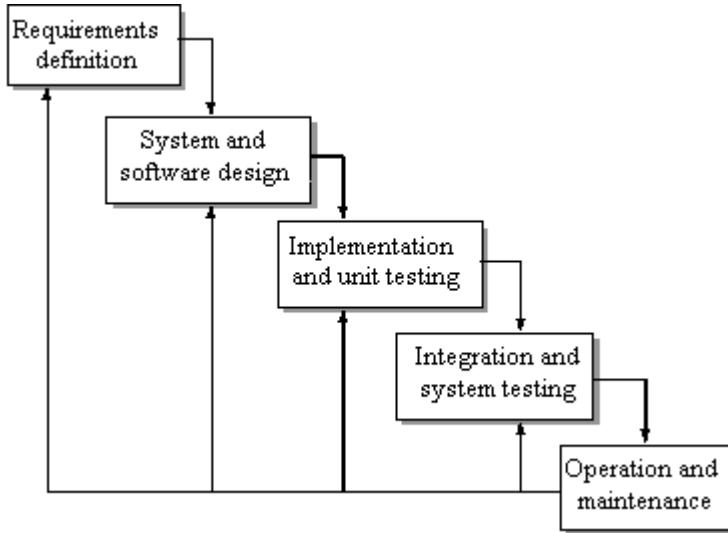


Figure 1-Software Process Model

- Stages of Waterfall Model explained:

- Requirement Analysis and Definition: All possible requirements of the system to be developed are captured in this phase. The requirements are gathered from the end user at the start of the software development phase. These requirements are analyzed for their validity, and the possibility of incorporating the requirements in the system to be developed is also studied. Finally, a requirement specification document is created which serves the purpose of guideline for the next phase of the model.
- System and Software Design: The requirement specifications from the first phase are studied in this phase and a system design is prepared. System design helps in specifying hardware and system requirements and also helps in defining the overall system architecture. The system design specifications serve as an input for the next phase of the model.
- Implementation and Unit Testing: On receiving system design documents, the work is divided in modules/units and actual coding is started. Each unit is developed and tested for its functionality; this is referred to as unit testing. Unit testing mainly verifies if the modules/units meet their specifications.
- Integration and System Testing: Units from above phase are integrated into a complete system during integration phase and tested to check if all modules/units coordinate with each other and the system as a whole behaves as per the specifications. After successfully testing the software, it is delivered to the customer.
- Operations & Maintenance: Generally, problems with the system developed (which are not found during the development life cycle)

come up after its practical use starts, so the issues related to the system are solved after deployment of the system. Not all the problems come into picture directly but they arise from time to time and need to be solved; hence, this process is referred to as maintenance.

2.2. Roles and Responsibilities

Name	Role	Responsibilities
Mr. Nguyen Trong Tai	Project Manager	<ul style="list-style-type: none"> - Supervise, give project guidance, and clear up team's queries. - Review and approve project documents
Dang Duong Hung	Team Leader	<ul style="list-style-type: none"> - Manage team by set up plan. - Control process - Support team members with technical problems. - Solve problems in communicating among team member. - Research - Developing - Testing
Duong Thi Diem Kieu	Team Member	<ul style="list-style-type: none"> - Research - Developing - Testing - Following project plan given by team leader
Tran Minh Thanh	Team Member	<ul style="list-style-type: none"> - Research - Developing - Testing - Following project plan given by team leader
Tran Quoc Dat	Team Member	<ul style="list-style-type: none"> - Research - Developing - Testing - Following project plan given by team leader.

Table 1-Member of Project

2.3. Tools and Techniques

- Tools: Netbeans 7.3, MySQL 5, Notepad++
- Front-end technologies: JSP, jQuery, AJAX, JSON, Knockoutjs
- Programming languages: Java, JavaScript
- Architecture: MVC
- Object-Relational Mapper: Hibernate

3. Project Management Plan

3.1. Tasks

<u>3.1.1 Researching:</u>	
Description	Do many kind of researching. Include: <ul style="list-style-type: none"> - Traditional system research and user's need research - Technology research: research some technologies will use in this project, such as Struts2, Hibernate...
Deliverables	Project Introduction Report
Resources Needed	All team members, 7 days
Dependencies and Constrains	Depend on: <ul style="list-style-type: none"> - Information from traditional system research. - User's need research. - Project Introduction report template.
Risks	<ul style="list-style-type: none"> - Some users may not agree to spend time for making survey. - Lack of technologies or chosen technologies are not feasible.
<u>3.1.2 Planning:</u>	
Description	Divide modules, create work breakdown structure prepare the project plan with task list (duration, predecessors, resources, etc.)
Deliverables	Software Project Management Plan report
Resources Needed	HungDD, 4 days
Dependencies and Constrains	<ul style="list-style-type: none"> - Depend on the Project Introduction report. - Project Management Plan report template. - The result of researching. - Task schedule must be in 15 weeks.
Risks	<ul style="list-style-type: none"> - Dividing modules and planning tasks maybe unbalance, and unreasonable. - In fact, workflow of system may change when implementation. - Maybe need to update functions list that lead to change schedule. - Knowledge about technologies and working in group of team is not enough which may lead to fail in keeping schedule.
<u>3.1.3 Create Software Requirement Specification:</u>	
Description	<ul style="list-style-type: none"> - Gather requirement from customer. - Analyze requirements carefully. - Create all the documents that necessary to describe the system.
Deliverables	Software Requirement Specification (SRS) document.
Resources Needed	All team members, 8 days

Dependencies and Constrains	<ul style="list-style-type: none"> - Planning process has to be finished. - Customer requirements - Follow the Software Requirement Specification template.
Risks	<ul style="list-style-type: none"> - All requirements come from team members, so conflicts may happen regularly. - Misunderstand between user requirements and receiving information from designer may make designing and developing phrases be failed. - Some customers just describe their requirement in simple way.

3.1.4 Designing the Database:

Description	<ul style="list-style-type: none"> - Create 3 steps in database design: <ul style="list-style-type: none"> ▪ Conceptual. ▪ Logical. ▪ Physical.
Deliverables	<ul style="list-style-type: none"> - Database design. - Database script. - This part is delivered with Software Design Description (SDD).
Resources Needed	HungDD, 4 days
Dependencies and Constrains	Finish the requirement and specification phrase.
Risks	<p>SRS is not good. It does not describe sufficient all business rules. Therefore, database design:</p> <ul style="list-style-type: none"> - Is not appropriate - Maybe changed much in the future.

3.1.5 Designing User Interface:

Description	<ul style="list-style-type: none"> - Design layouts for each page in the system. - Design style for each page in the system. - Design connection between all pages in the system. - Define some basic JavaScript for each file.
Deliverables	Prototype which includes: <ul style="list-style-type: none"> - HTML pages. - CSS files. - JavaScript files.
Resources Needed	ThanhTM, 6 days
Dependencies and Constrains	<ul style="list-style-type: none"> - Finish the requirement specification task. - User Interface must be friendly, easy to see and suitable with all functions in the system and the topic of this project.
Risks	User interface may not completely satisfy all dependencies and constrains above.

3.1.6 Creating Software Design Description:

Description	Do architecture design and detailed design.
-------------	---

Deliverables	Software Design Description report
Resources Needed	All team members, 6 days
Dependencies and Constraints	<ul style="list-style-type: none"> - Finish Software Requirement Specification report. - Follow Software Design Description (SDD) report template.
Risks	<ul style="list-style-type: none"> - SRS may not be detailed enough in other to capture business logic. - Designing work is not good which lead to occur problem when implement, such as spending much effort for coding, maintaining hard...

3.1.7 Create Coding Framework:

Description	<ul style="list-style-type: none"> - Mapping the architecture design into source code. - Create project solution files with: <ul style="list-style-type: none"> ▪ Common classes ▪ Common functions
Deliverables	Java web project and solution files.
Resources Needed	HungDD, 4 days
Dependencies and Constraints	<ul style="list-style-type: none"> - Finish Software Design Description (SDD) report. - Follow coding convention.
Risks	<ul style="list-style-type: none"> - Having bugs when coding framework. - Team members do not understand thoroughly framework. - Problem about coding skill which is not follow coding convention. - Actual time may be more than estimated time.

3.1.8 Implementation:

Description	Implement the website based on all requirements and designs.
Deliverables	The whole website of Student Club Management System project.
Resources Needed	All team members, 23 days
Dependencies and Constraints	<ul style="list-style-type: none"> - Depend on the completion of SRS, SDD and Database design. - Follow coding convention.
Risks	<ul style="list-style-type: none"> - Team members do not understand thoroughly framework. - Problem about coding skill which is not follow coding convention. - Behind schedule. - Unit test completion depend on team member's skill may require more effort in later testing phrase.

3.1.9 Performing System Test:

Description	Do systems test, include: <ul style="list-style-type: none"> - Function test
-------------	---

	<ul style="list-style-type: none"> - Performance test
Deliverables	Software Test Documentation (STD) report.
Resources Needed	All team members, 2 days
Dependencies and Constrains	<ul style="list-style-type: none"> - Finish coding development. - Follow Software Test Documentation (STD) report template.
Risks	Team members are both developer and tester could lead to compromise and leak.

3.1.10 Deployment and Input Initial Data:

Description	<ul style="list-style-type: none"> - Deploy the system - Input initial data for the system
Deliverables	Complete MRS project source code, up-to-date all documentations.
Resources Needed	All team members, 8 days
Dependencies and Constrains	Finish System testing.
Risks	<ul style="list-style-type: none"> - Update all documentations may need more time than schedule. - Team members may not familiar with input consistent data for all tables in database. - The program is not running properly in real environment. - Cannot release before the deadline.

3.1.11 Creating Software User's Manual:

Description	Create User's manual.
Deliverables	Software User's Manual (SUM) report.
Resources Needed	All team members, 8 days
Dependencies and Constrains	<ul style="list-style-type: none"> - Finish Deployment process. - Follow Software User's Manual (SUM) report template.
Risks	<ul style="list-style-type: none"> - User's manual may be not easy to understand with the customer. - Team members may be not familiar with crating user's manual which leads to behind schedule.

Table 2-Task Plan

3.2. Task sheet

Name	Duration	Start	Finish	Resource Name
Initiating	1 day?	5/6/13	5/6/13	
Identify key stakeholders	4 hours	5/6/13	5/6/13	HungDD
Hold project kick-off meeting	4 hours	5/6/13	5/6/13	All members
Research	7 days	5/7/13	5/15/13	
Current Systems Research	2 days	5/7/13	5/8/13	All members
User's Needs Research	3 days	5/9/13	5/13/13	All members

Technology Research	2 days	5/14/13	5/15/13	All members
<i>Project Introduction Report</i>	0 days	5/15/13	5/15/13	All members
Planning	4 days	5/16/13	5/21/13	
Create Work Breakdown Structure (WBS)	1 day	5/16/13	5/16/13	HungDD
Create project management plan	1 day	5/17/13	5/17/13	HungDD
<i>Software Project Management Plan</i>	0 days	5/17/13	5/17/13	HungDD
Executing	59.5 days?	5/20/13	8/9/13	
Getting requirement	6 days	5/20/13	5/27/13	
Define user requirements	1 day	5/20/13	5/20/13	All members
Define system requirements	4 days	5/21/13	5/24/13	All members
Define non-functional requirements	4 hours	5/27/13	5/27/13	All members
Complete Software Requirement Specification	4 hours	5/27/13	5/27/13	All members
<i>Software Requirement Specification</i>	0 days	5/27/13	5/27/13	All members
Write system test cases	1 day	5/28/13	5/28/13	All members
<i>System Test Cases Document</i>	0 days	5/28/13	5/28/13	All members
Designing	14 days	5/28/13	6/14/13	
Perform architecture design	2 days	5/28/13	5/29/13	HungDD
<u>Design Database</u>	4 days	5/30/13	6/4/13	HungDD
Create conceptual database design	1 day	5/30/13	5/30/13	HungDD
Create logical database design	1.5 days	5/31/13	6/3/13	HungDD
Create physical database design	1.5 days	6/3/13	6/4/13	HungDD
<i>Database design and database script</i>	0 days	6/4/13	6/4/13	HungDD
<u>Design User Interface</u>	4 days	5/30/13	6/4/13	ThanhTM
Design global master page	4 hours	5/30/13	5/30/13	ThanhTM
Design admin master page	4 hours	5/30/13	5/30/13	ThanhTM
Design detailed pages	2.5 days	5/31/13	6/4/13	ThanhTM
Define page flows	4 hours	6/4/13	6/4/13	ThanhTM
<i>All pages' user interface and page flow document</i>	0 days	6/4/13	6/4/13	ThanhTM
Perform detailed design	6 days	6/5/13	6/12/13	All members
Write unit test cases	1 day	6/13/13	6/13/13	All members
<i>Unit Test Cases Documents</i>	0 days	6/13/13	6/13/13	All members
Create Software Design Description (SDD) document	1 day	6/14/13	6/14/13	All members
<i>Software Design Description Document</i>	0 days	6/14/13	6/14/13	All members
Development	22.5 days?	6/17/13	7/17/13	
Create coding framework	4 days	6/17/13	6/20/13	HungDD
<i>MRS project</i>	0 days	6/20/13	6/20/13	HungDD

<u>Develop Authentication module</u>	2 days?	6/21/13	6/24/13	DatTQ
Code register function	1 day?	6/21/13	6/21/13	DatTQ
Code login function	6 hours	6/24/13	6/24/13	DatTQ
Code logout function	2 hours	6/24/13	6/24/13	DatTQ
<u>Develop Customer module</u>	3.5 days	6/20/13	6/25/13	KieuDTD
Update profile	4 hours	6/20/13	6/20/13	KieuDTD
Code View Equipment functions	4 hours	6/20/13	6/20/13	KieuDTD
Code view FAQ functions	4 hours	6/21/13	6/21/13	KieuDTD
Code Tracking Request functions	2 days	6/21/13	6/25/13	KieuDTD
<u>Develop Staff module</u>	2 days	6/20/13	6/21/13	ThanhTM
Code View all task list functions	4 hours	6/20/13	6/20/13	ThanhTM
Code View task detail function	4 hours	6/20/13	6/20/13	ThanhTM
Code Update task function	4 hours	6/21/13	6/21/13	ThanhTM
Code Filter task function	4 hours	6/21/13	6/21/13	ThanhTM
<u>Develop Technical Manager Module</u>	9.5 days?	6/20/13	7/3/13	All members
<i>Code Manage FAQ functions</i>	1.5 days	6/25/13	6/26/13	KieuDTD
Code List all FAQ function	4 hours	6/25/13	6/25/13	KieuDTD
Code Create new FAQ function	4 hours	6/26/13	6/26/13	KieuDTD
Code Update FAQ function	4 hours	6/26/13	6/26/13	KieuDTD
<i>Code Manage repair/maintenance order functions</i>	2.5 days	6/20/13	6/24/13	HungDD
Code List all request function	4 hours	6/20/13	6/20/13	HungDD
Code Filter request function	4 hours	6/20/13	6/20/13	HungDD
Code Create new request function	4 hours	6/21/13	6/21/13	HungDD
Code Request detail function	4 hours	6/21/13	6/21/13	HungDD
Code Update request function	4 hours	6/24/13	6/24/13	HungDD
<i>Code Manage Task functions</i>	2.5 days	6/24/13	6/26/13	HungDD
Code List all Tasks function	4 hours	6/24/13	6/24/13	HungDD
Code Filter Tasks function	4 hours	6/25/13	6/25/13	HungDD
Code Create new Task function	4 hours	6/25/13	6/25/13	HungDD
Code Task detail function	4 hours	6/26/13	6/26/13	HungDD
Code Update Task function	4 hours	6/26/13	6/26/13	HungDD
<i>Code Manage Symptom/Solution functions</i>	2 days	6/24/13	6/25/13	ThanhTM
Code List all symptoms function	4 hours	6/24/13	6/24/13	ThanhTM
Code Create new symptom function	4 hours	6/24/13	6/24/13	ThanhTM
Code Symptom detail function	4 hours	6/25/13	6/25/13	ThanhTM
Code Update symptom function	4 hours	6/25/13	6/25/13	ThanhTM
<i>Code Manage Equipment</i>	1.5 days	6/27/13	6/28/13	HungDD

<i>functions</i>				
Code List all equipment function	4 hours	6/27/13	6/27/13	HungDD
Code Create new equipment function	4 hours	6/27/13	6/27/13	HungDD
Code Update equipment function	4 hours	6/28/13	6/28/13	HungDD
<i>Code Statistics functions</i>	3 days?	6/28/13	7/3/13	HungDD
Code Statistics by Symptom function	1 day?	6/28/13	7/1/13	HungDD
Code Statistics by Equipment function	1 day	7/1/13	7/2/13	HungDD
Code Statistics by Cost function	1 day	7/2/13	7/3/13	HungDD
<u>Develop Admin module</u>	1.5 days	6/25/13	6/26/13	DatTQ
Code List all accounts function	4 hours	6/25/13	6/25/13	DatTQ
Code Create new account function	4 hours	6/25/13	6/25/13	DatTQ
Code Update account function	4 hours	6/26/13	6/26/13	DatTQ
<u>Develop Push/Receive Notification module</u>	6 days	7/3/13	7/11/13	HungDD
Code push notification function	3 days	7/3/13	7/8/13	HungDD
Code receive notification function	3 days	7/8/13	7/11/13	HungDD
<u>Develop Authorization Module</u>	4 days	7/11/13	7/17/13	HungDD
Integrate Authorization into functions	4 days	7/11/13	7/17/13	HungDD
<i>Testing</i>	2 days	7/17/13	7/19/13	
Perform system test	2 days	7/17/13	7/19/13	All members
<i>Software Test Documentation</i>	0 days	7/19/13	7/19/13	All members
<u>Deployment & Input Initial Data</u>	7 days	7/19/13	7/30/13	
Deploy system to the Internet	1 day	7/19/13	7/22/13	HungDD
Input initial data	6 days	7/22/13	7/30/13	All members
<u>Create User's Manual</u>	8 days	7/30/13	8/9/13	
Create user's manual	8 days	7/30/13	8/9/13	All members
<i>Software User's Manual</i>	0 days	8/9/13	8/9/13	All members
<u>Monitoring and Controlling</u>	70 days	5/6/13	8/9/13	
Monitor and control project work	70 days	5/6/13	8/9/13	HungDD
Perform scope control	70 days	5/6/13	8/9/13	HungDD
Perform schedule control	70 days	5/6/13	8/9/13	HungDD
Perform change control	70 days	5/6/13	8/9/13	HungDD
Perform quality control	70 days	5/6/13	8/9/13	HungDD
<u>Closing</u>	6 days?	8/9/13	8/19/13	
Prepare final project presentation	5 days	8/9/13	8/16/13	All members
Deliver final report and presentation	1 day?	8/16/13	8/19/13	All members
<i>Capstone Project Completed</i>	0 days	8/19/13	8/19/13	All members

Table 3-Tasks sheet**3.3. All Meeting Minutes****3.3.1. Meeting May 23th 2013**

Subject	Define user requirements	Date	May-23-2013
Facilitator	Library of FPT University	Time	17:30 – 18:45
Location	FPT University, District 12, HCMC	Scribe	KieuDTD
Attendees	HungDD(leader), KieuDTD, ThanhTM, DatTQ		

Key Points Discussed		
No.	Topic	Highlights
1	User requirements	Discuss user requirements of this project
2	ER diagram	Discuss design of ER diagram

Action Plan			
No.	Action Item(s)	Owner	Target Date
1.1	Technical managers would only view and notify their tasks/requests.	All	May-23-2013
1.2	In case of repair at home, customer cannot create the request with specific form. They must login and access Contact item to send request (with capcha). Then, center service would confirm with customer and assign staff to repair. When finish task, technical manager save the information in table [ProductRequest]	All	May-23-2013
1.3	Add to Request textarea for customer's confirm.	All	May-23-2013
1.4	We can estimate cost by 2 ways: - Based on the previously repaired (the same model) to calculate the average of cost - Based on symptoms, the inspector checks into the list text box of replacement item. The sum of all those items is the cost.	All	May-23-2013
1.5	When technical manager create a symptom, suggest name of existed symptoms.	All	May-23-2013
2.1	Save the components expected to replace the table [SolutionDescription]. After replacing specific product, we update to table [UsedEquipment] and re-update the attribute SolutionDescription in table [ProductRequest]	HungDD	May-23-2013

Action Plan			
No.	Action Item(s)	Owner	Target Date
2.2	Add the minimum price when maintain/repair product in table [ProductCategory]	HungDD	May-23-2013
2.3	Add attribute IsWaitingConfirm in table [ProductRequest] to customer's confirm: We have all power to repair product without their confirmation or not.	HungDD	May-23-2013
2.4	Combine table [SkillTag] and [EquipmentCategory]	HungDD	May-23-2013
2.5	Delete table [Solution], add attribute SolutionContent into table [Symptom]	HungDD	May-23-2013
2.6	ER diagram has no foreign key	HungDD	May-23-2013

Table 4-Meeting minutes

4. Coding convention

Following Java coding conventions at

<http://www.oracle.com/technetwork/java/javase/documentation/codeconvtoc-136057.html>

III. Software Requirements Specifications (SRS)

1. User Requirement Specification

1.1. Guest Requirements

- All people who visit the MRS website seem to be “Guest” role.
- Guest can view equipment information, FAQ.
- Guest can register to be user of the website. Username, password, first name, last name, gender and email address are required for registration.

1.2. Registered User Requirements

- Registered User is person who registered an account at MRS website.
- Registered User must use their username and password to log into the website.
- Registered User can log out the website.
- Registered User can view and edit his/her account information: Name, Password, Email, Phone number, etc.
- Registered User has all function of [Guest] role, except registering.

1.3. Customer Requirements

- Customer is client who has product to repair or maintain.
- Customer is user of the website after registering an account with “Customer” role.

- Customer can create, update and tracking progress of product which being maintained or repaired at service center.
- Customer can view history request.
- Customer can get notification when have new updates belong to his (her) request.
- Member has all function of “Registered User” role.

1.4. Receptionist Requirements

- Receptionist is person in service center who meet customer and record request.
- Receptionist is user of the website when Technical manager creates for that receptionist an account with “Receptionist” role.
- Receptionist can create request for customer, export receipt or invoice.
- Member has all function of “Registered User” role.

1.5. Staff Requirements

- Staff is person in service center who directly repairs or maintains products of customer.
- Staff is user of the website when Technical Manager creates for that staff an account with “Staff” role.
- Staff can track, update progress of his tasks that is assigned by Technical Manager.
- Staff can view history tasks belong to him (her).
- Staff can get notification when have new tasks or updated tasks belong to him (her).
- Member has all function of “Registered User” role.

1.6. Technical Manager Requirements

- Technical Manager is person in service center who checks general symptom of customer’s products and assign tasks to staff who repairs or maintains product.
- Technical Manager is user of the website after registering an account with “Technical Manager” role.
- Technical Manager can manage FAQ such as views, creates and updates FAQ.
- Technical Manager can manage repair/maintain order from customer such as views, creates, updates repair/maintain request, filter request...
- Technical Manager can manage tasks of staff such as views, creates and updates tasks.
- Technical Manager can manage symptom such as add new symptom belong to which product model, update existed symptom.
- Technical Manager can manage equipment such as views, creates and updates equipment (such as RAM, Mainboard...)
- Technical Manager can get notification when customer confirms in repairing/maintenance request or when staff updates task.
- Technical Manager can manage user account such as create new user account with corresponding role, update role or deactivate user account.

- Technical Manager has all function of “Registered User” role.

2. System Requirement Specification (Specific Requirements)

2.1. External Interface Requirement

2.1.1. User Interfaces

Screen Name	Function
Register Page	Allow actor (Guest) to register
Home Page	Allow all actors view Home Page
FAQ Page	Allow all actors view FAQ Page
Equipment Page	Allow all actors view Equipment Page
Equipment Detail Page	Allow all actors view Equipment Detail Page
Profile Page	Allow registered actors view and update registered information in Profile Page
Customer Request Page	Allow actor (Customer) views, creates and updates request about repairing product
Staff Tracking Task Page	Allow actor (Staff) tracks his (her) tasks and updates tasks
Manager CP Page	Page with display all management functions of Technical Manager
Manage FAQ	Allow actor (Technical Manager) views, creates and updates FAQ
Manage Repair Maintenance Page	Allow actor (Technical Manager) views, creates and updates requests about repairing or maintenance products of customers.
Manage Task Page	Allow actor (Technical Manager) views, creates, and updates tasks for staffs that repair products of customers.
Manage Symptom Page	Allow actor (Technical Manager) views, creates and updates symptoms of products and solutions for it.
Manage Equipment Page	Allow actor (Technical Manager) views, creates and updates equipment in inventory
Statistics Page	Allow actor (Technical Manager) views statistics about common symptoms, replaced equipment...
Manage Account Page	Allow actor (Technical Manager) to manage account
Error Page	Display error messages when website has errors.

Table 5- User Interfaces

2.2. System Features

❖ Overall Use Case Diagram:

**Figure 2-Overall Use case diagram**

Note: Actor Customer, Staff, Receptionist, Technical manager inherit all use case of actor Registered User. Registered User is Guest have an account to access the website and inherit all use case of Guest.

❖ Guest's Use Case Diagram:

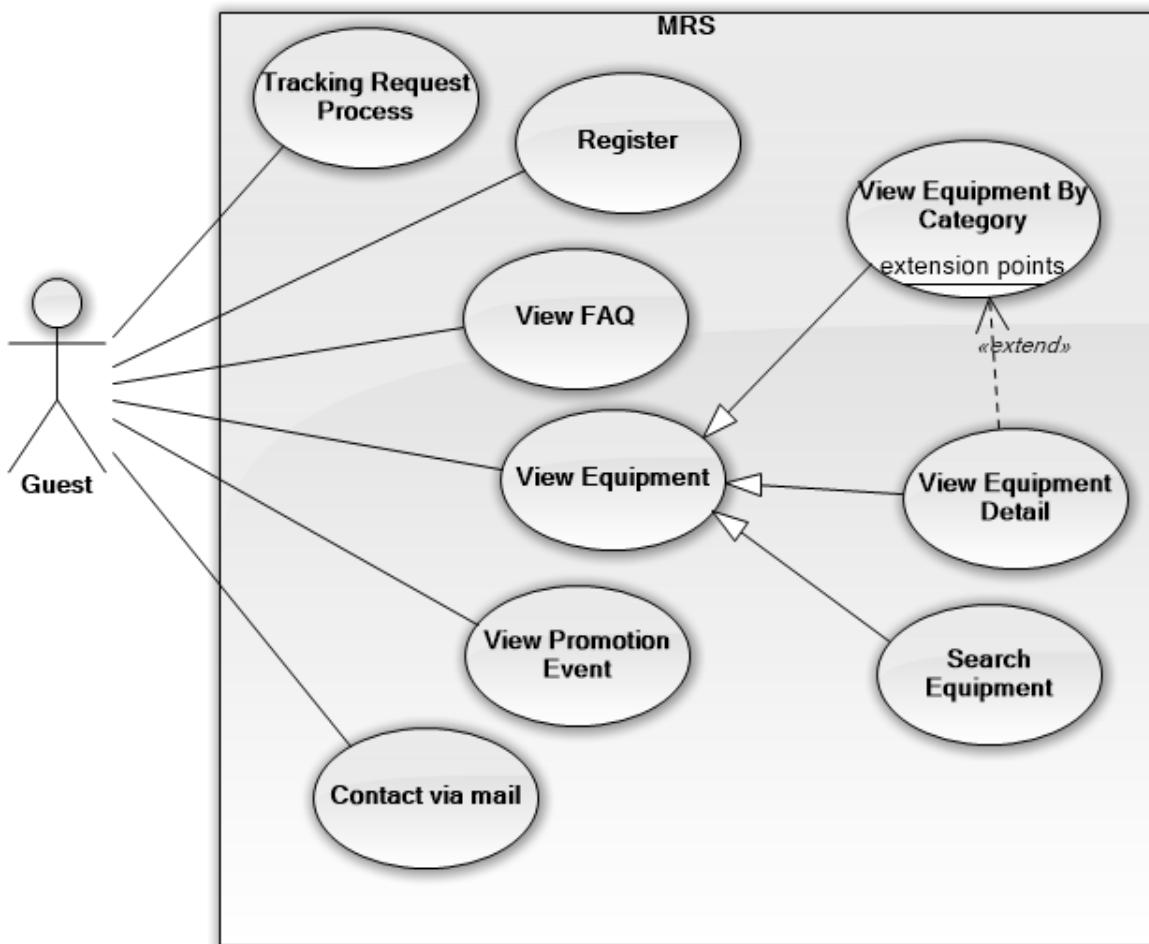


Figure 3-Guest's use case diagram

❖ Registered User's Use Case Diagram:

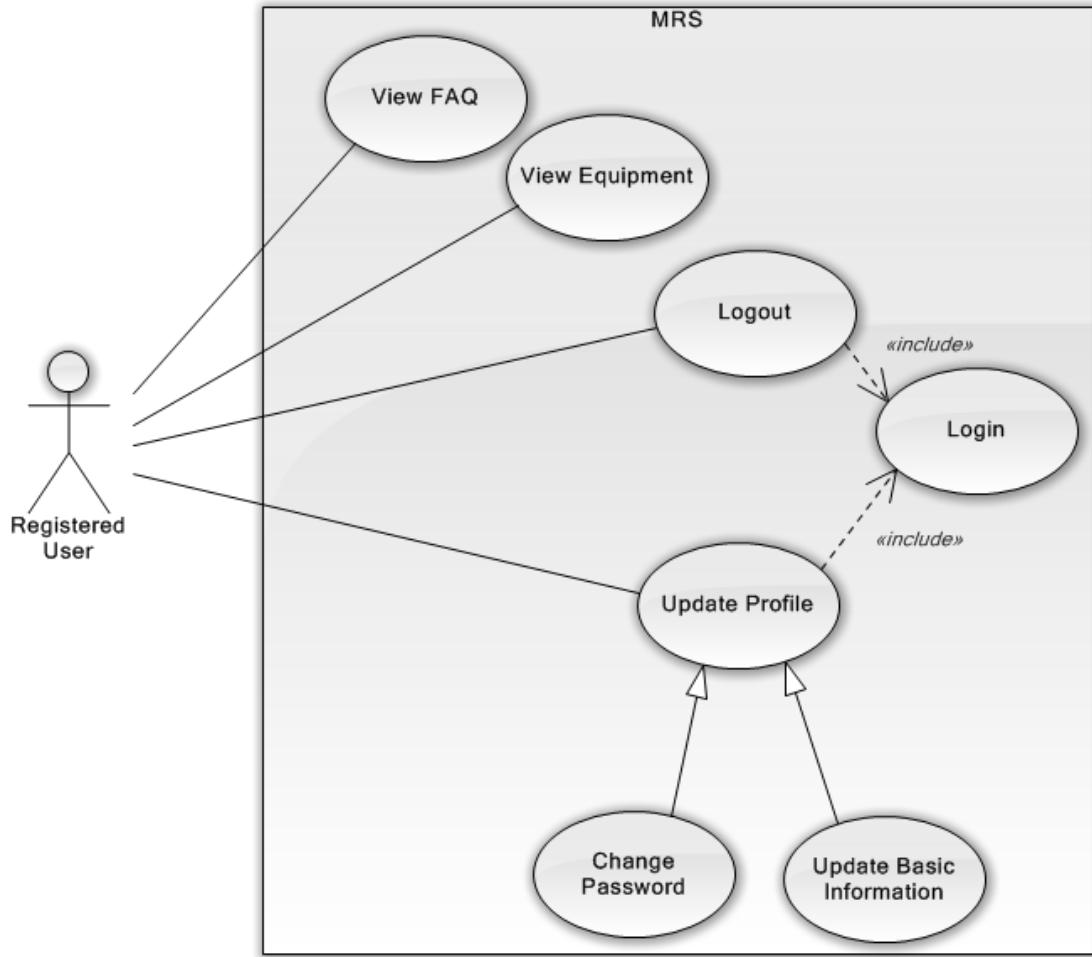


Figure 4-Registered User's use case diagram

Note: Actor Registered User is actor Guest who uses [Register] function to create an account.

❖ Customer's Use Case Diagram:

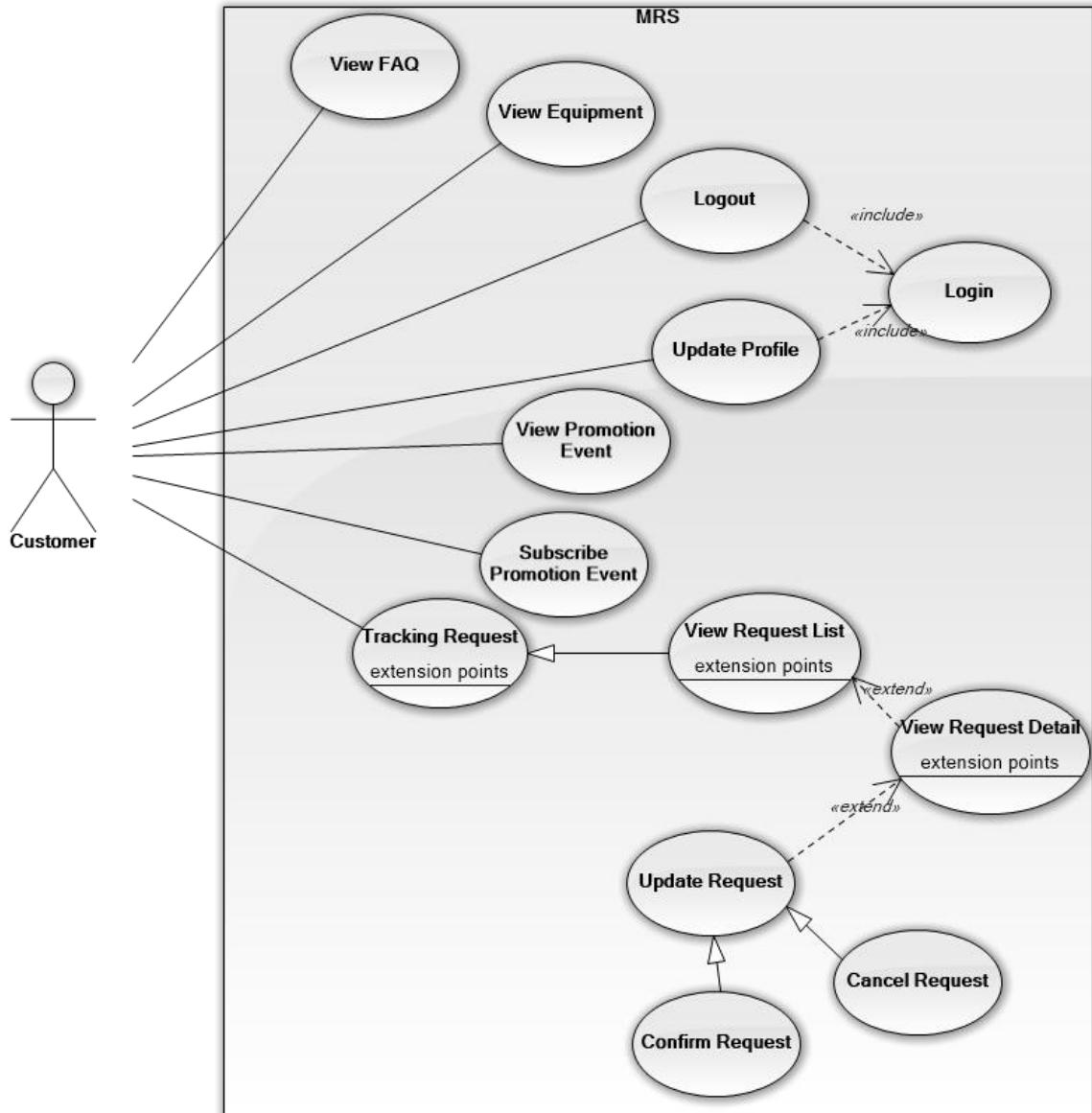


Figure 5-Customer's use case diagram

❖ Staff's Use Case Diagram:

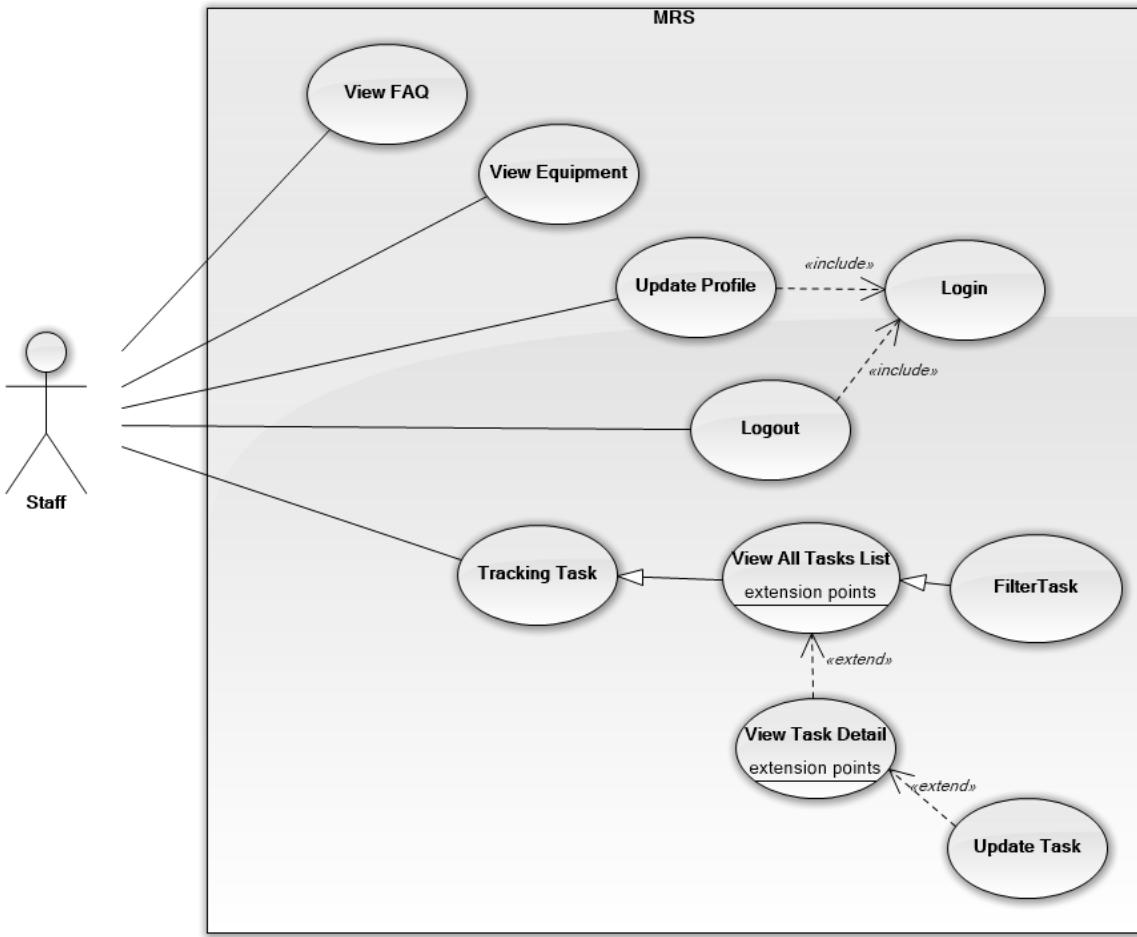


Figure 6-Staff's use case diagram

❖ Receptionist's Use Case Diagram:

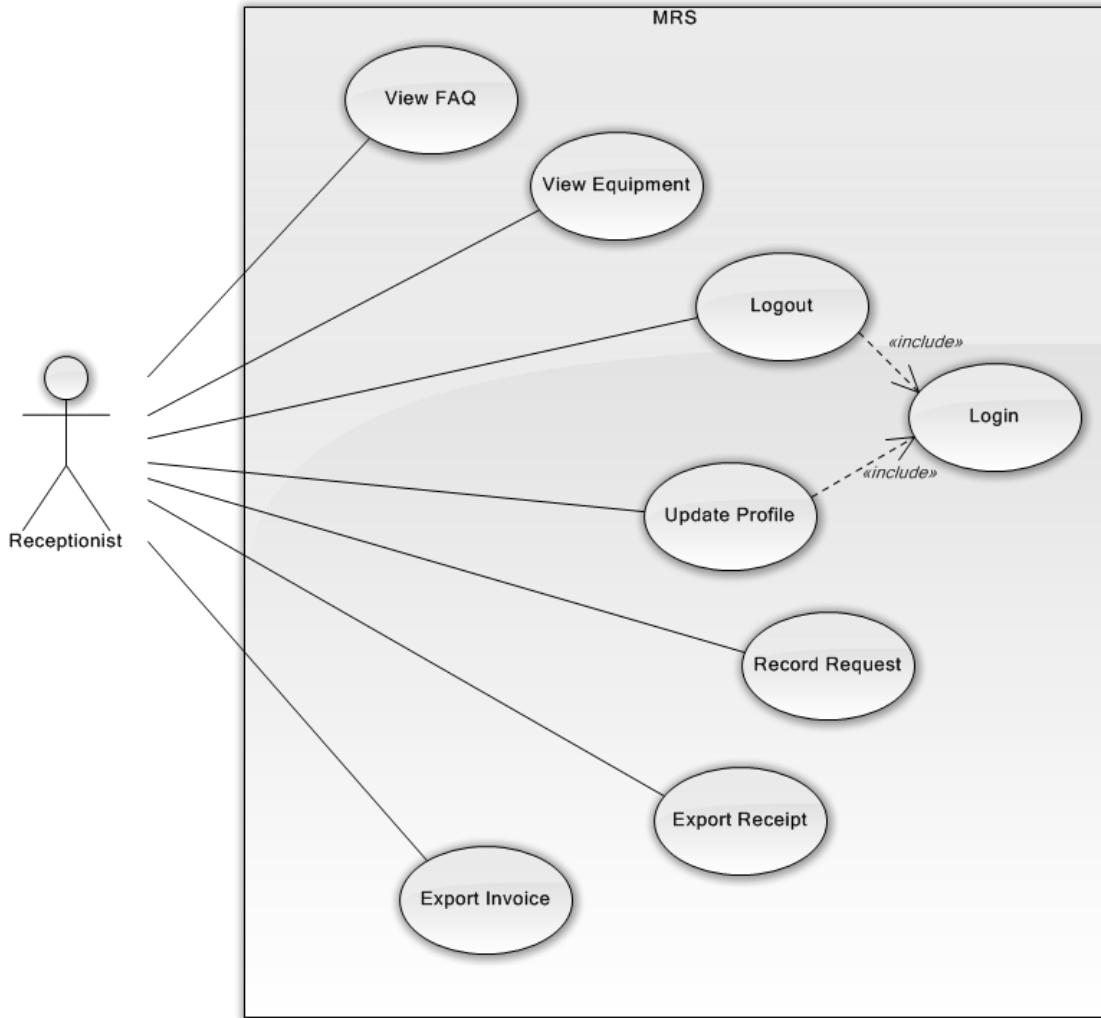


Figure 7 – Receptionist use case diagram

❖ **Technical manager's Use Case Diagram:**

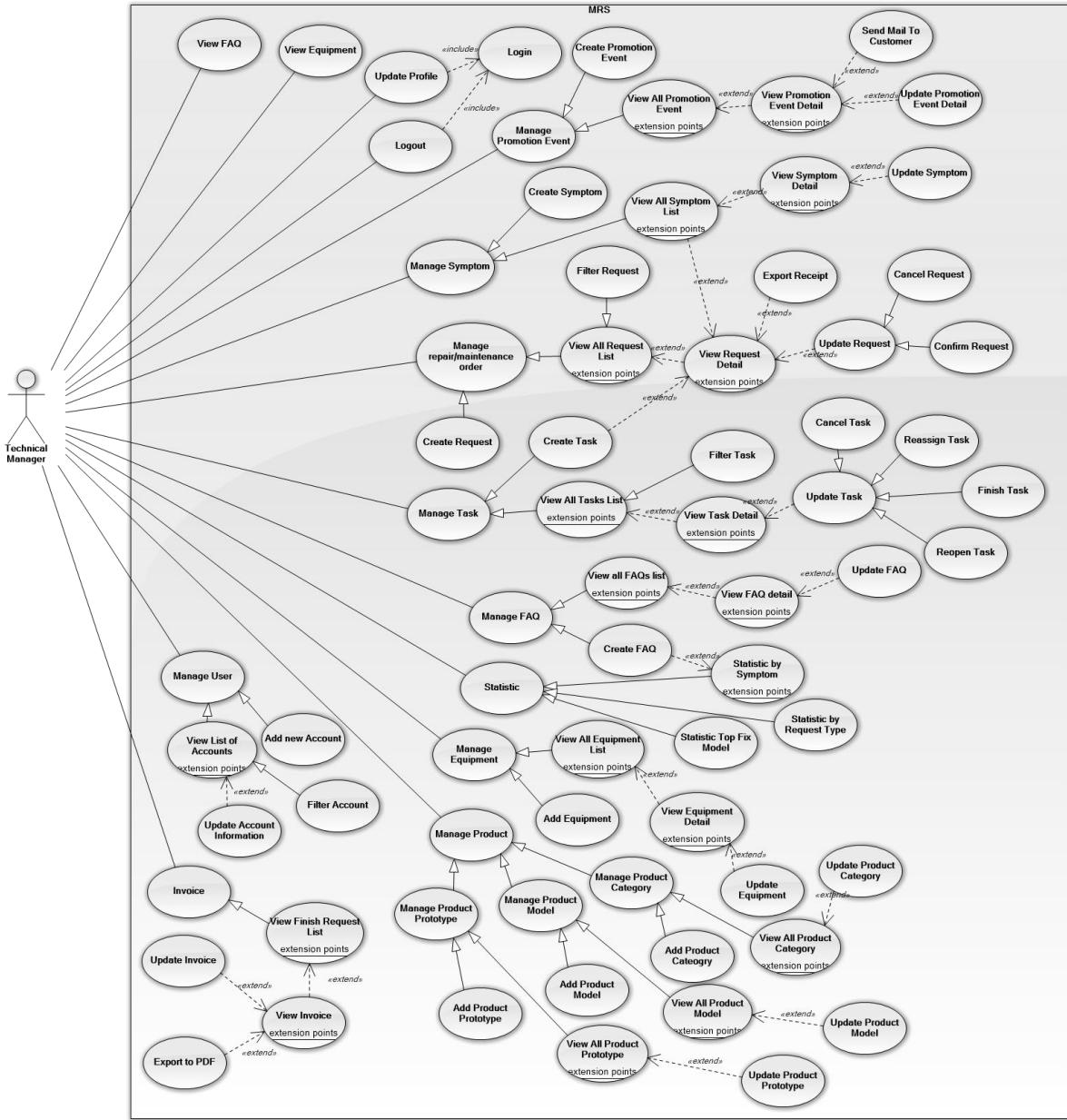


Figure 8-Technical manager's use case diagram

❖ **System's Use Case Diagram:**

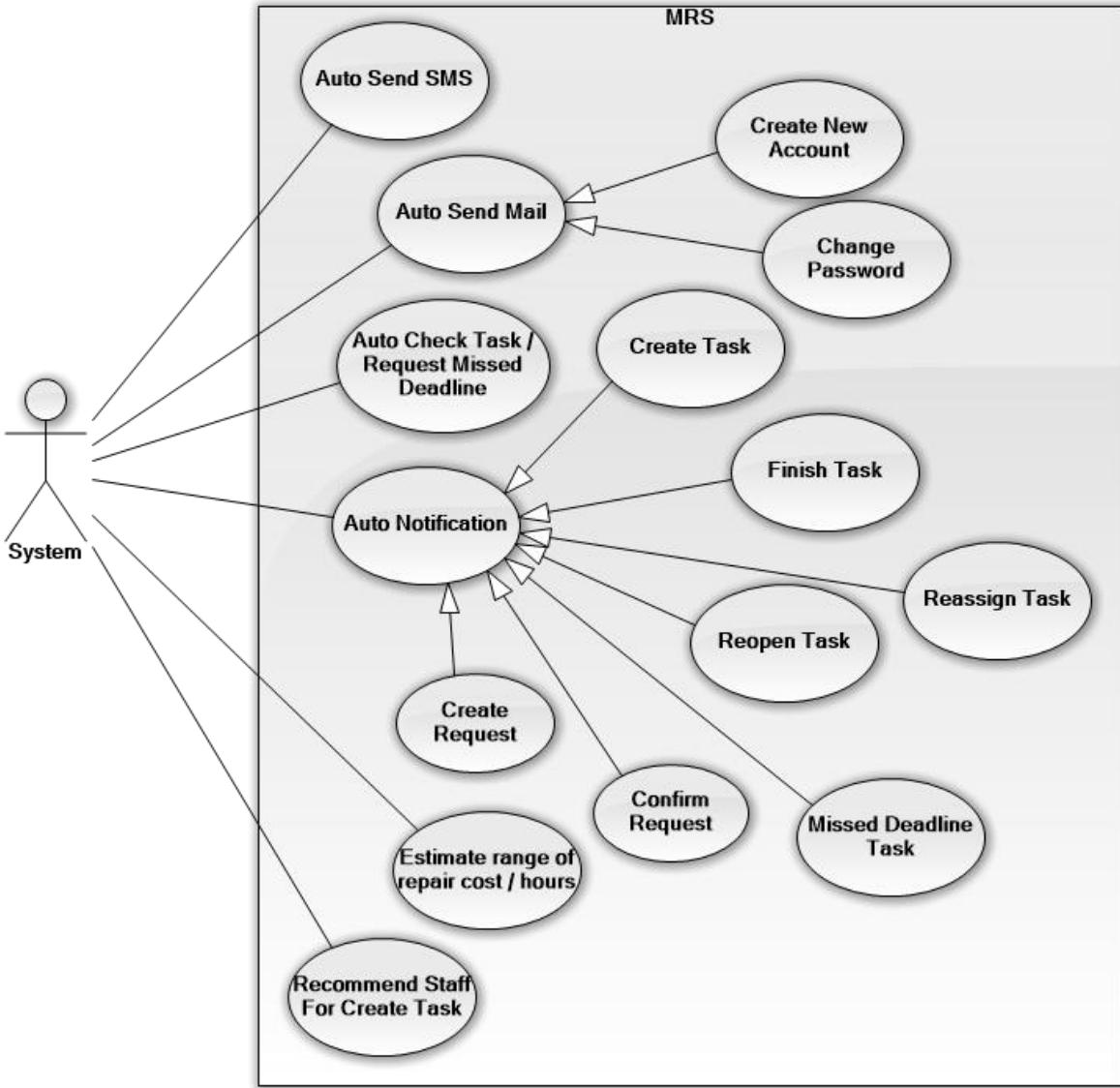
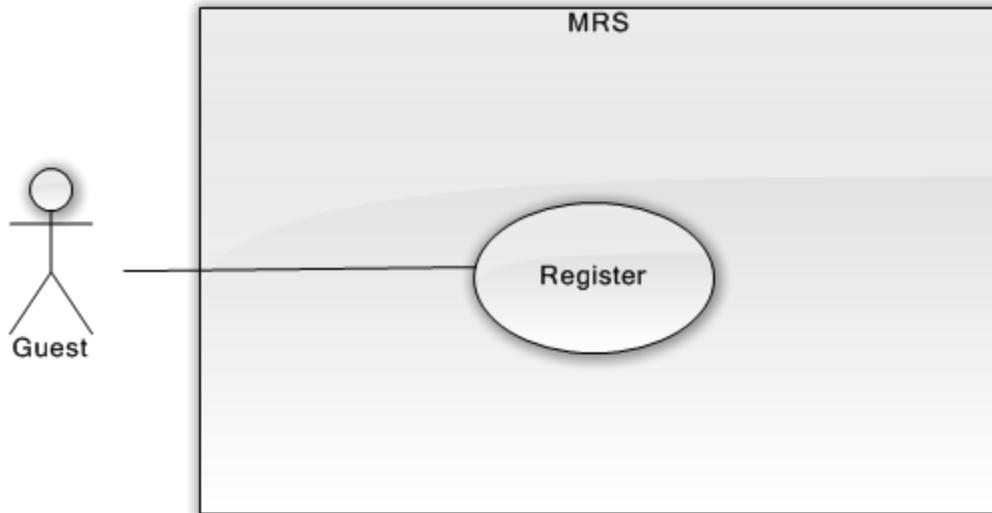


Figure 9 - System's Use Case Diagram

❖ **Actors:**

Actor	Description
Guest	All people who visit the MRS website seem to be Guest
Registered user	Registered is a person who registered an account at MRS website
Customer	Customer is a person who will send a request to maintain/repair
Staff	Staff is a person who have direct responsible for maintain/repair.
Receptionist	Receptionist is a person who meet customer and record request of customer

Technical manager	Technical manager is person who will manage a request and assign task for staff. Ensure the fulfillment of maintain/repair process.
--------------------------	---

Table 6- Actors**2.2.1. <Guest> Register****2.2.1.1. Use Case Diagram****Figure 10 - Guest Register usecase diagram****2.2.1.2. Use Case Specification**

USE CASE – Register SPECIFICATION			
Use case No.	UC001	Use case version	2.0
Use case name	Register		
Author	DatTQ		
Date	15/05/2013	Priority	Normal
Actor: Guest			
Summary: Guest wants to be the customer of the website. With this use case, guest can create an account. They must create an account with some information: username, password, email, address, phone number, etc...			
Goal: Allow guest to create a new account to use main functions of the website.			

Triggers:

On the [Home Page], guest clicks on [Register] button and [Register Page] will be shown then input the information, and clicks on [CREATE ACCOUNT] button to finish.

Preconditions:

- Guest has not had an account of the website yet.
- The username and email use to register that do not exist before

Post Conditions:

Success: Guest creates an account successfully. The website shows pop up to confirm that the action was finish successfully at current page.

Fail: The website will transfer to [Register Page] and display the error message.

Main Success Scenario:

Actor Action	System Response
1. Click on [Register] button in the [Home Page].	<p>2. The website will transfer [Register Page] with [Register form] includes:</p> <ul style="list-style-type: none"> - FIRSTNAME: Textbox (min length: 6, max length: 20) - LASTNAME: Textbox (min length:6, max length:20) - USERNAME: Textbox (min length: 6, max length: 20) - PASSWORD: Password Field (min length: 6, max length: 20) - TYPE PASSWORD: Password Field (matching with password) - EMAIL: Textbox - RETYPE EMAIL: Textbox - PHONE NUMBER: Number - ADDRESS: Text Area - GENDER: Checkbox - CAPCHA CODE: Textbox - RESET: Button [Allternative1] - CREATE ACCOUNT: Button
3. Enter information into text fields.	4. [Exception 1,3,5,7,9,11,13,15]
5. Click on [CREATE ACCOUNT] button.	6. Display pop up: " You have registered successfully".

Alternative Scenario:

Actor Action	System Response

1. Click on [RESET] button.	2. The website will reset all information that guest was fill.
-----------------------------	--

Exceptions:

Actor Action	System Response
1. First name, Last name, User name, Password, Retype Password, Email, Retype Email, Phone Number, Address, Capcha Code is blank.	2. Display error message: "Please fill out this field." on the field which is blank.
3. Username, Password is less than 6 characters or greater than 20 characters.	4. Display error message: "Please match the requested format. 6 to 20 characters [a-zA-z0-9]"
5. Username, Email is registered before.	6. Display error message: "Not available in use."
7. Retype Password is different with password.	8. Display error message: "Confirm password does not match."
9. Retype Email is different with email.	10. Display error message: "Confirm email does not match."
11. Phone Number is not a number.	12. Display error message: "Phone must be a number."
13. The guest enters the phone that is less than 9 numbers or greater than 11 numbers.	14. Display error message: "Please match the requested format. 9 to 11 characters [0-9]."
15. Capcha Code does not match	16. Display error message: "The characters you entered didn't match the word verification."

Relationships:

N/A

Business Rules:

- New account with all information can be created and saved on database.
- Guest can use this account to login into the website.
- Guest creates new account successfully, the system also push an email to notice finish action.

2.2.2. <Guest> View FAQ

2.2.2.1. Use Case Diagram



Figure 11 – Guest View FAQ usecase diagram

2.2.2.2. Use Case Specification

USE CASE – View FAQ SPECIFICATION			
Use case No.	UC002	Use case version	2.0
Use case name	View FAQ		
Author	DatTQ		
Date	18/05/2013	Priority	Normal
Actor: Guest			
Summary: With View FAQ function, the website will show all FAQs. When Guest visits the website, they can view all FAQs on the [FAQ Page] includes:			
+ Right side: display all symptoms of the system. When guest clicks on any symptoms, specific solution will expand below.			
+ Left side: display all products model that have damage symptom. When guest selects a product model, all symptoms belong to that product will show on the right side.			
Goal: Guest can view all FAQs which exist in the website. FAQ includes symptom and solution.			
Triggers: On the [Home Page], guest clicks on [FAQ] tab. The website will transfer to [FAQ Page].			
Preconditions: The website must have at least one symptom to view.			
Post Conditions:			

Success: When guest clicks on the symptom, specific solution will expand below of that symptom.

Fail: N/A

Main Success Scenario:

Actor Action	System Response
1. In the [Home Page] page, guest clicks on [FAQ] tab.	2. The website will response [FAQ Page].
3. Guest clicks on the symptom.	4. Specific solution will expand below.

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

N/A

Business Rules:

When guest clicks on the symptom, the solution of that symptom will be expanded below that symptom.

2.2.3. <Guest> [View Equipment] View Equipment by Category

2.2.3.1. Use Case Diagram

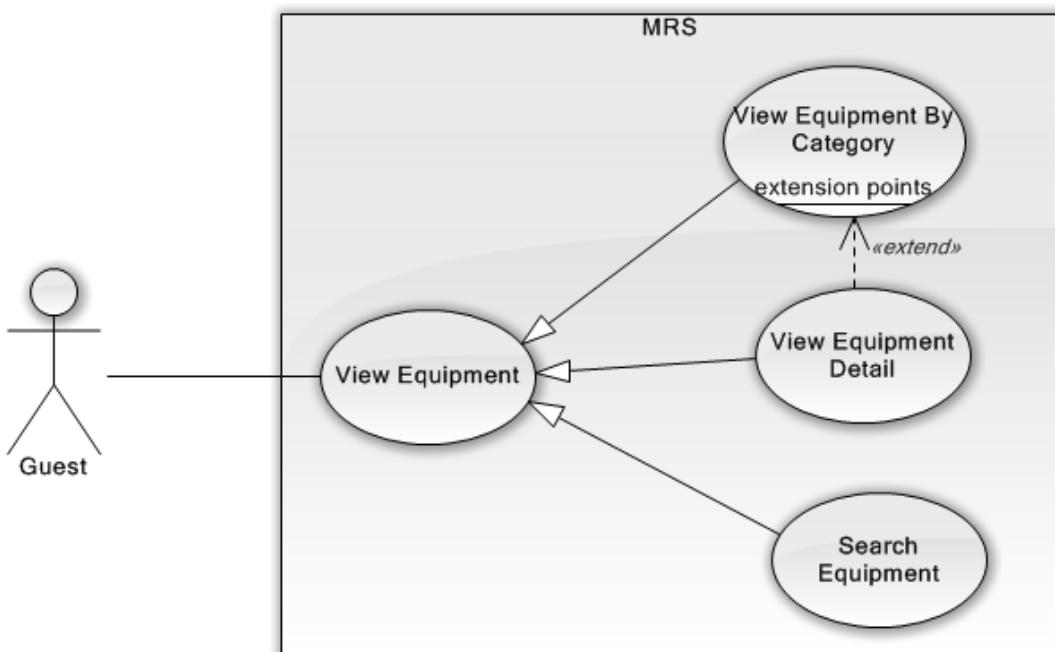


Figure 12-View Equipment use case diagram

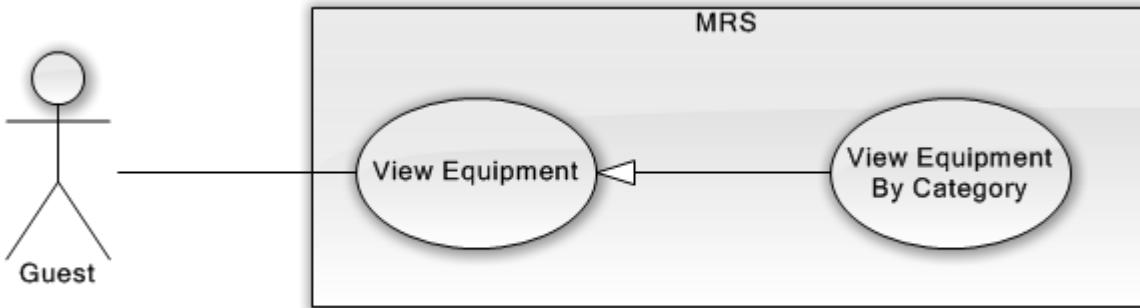


Figure 13- View Equipment By Category

2.2.3.2. Use Case Specification

USE CASE – View Equipment By Category SPECIFICATION						
Use case No.	UC003	Use case version	2.0			
Use case name	View Equipment By Category					
Author	DatTQ					
Date	21/05/2013	Priority	Normal			
Actor: Guest						
Summary: With View Equipment By Category function, the website will show all Equipment by Category. When Guest visits the website, they can view all Equipment by Category to select suitable equipment with them.						
Goal: Guest can view all equipment by category which exists in the website.						
Triggers: On the [Home Page], guest clicks on [EQUIPMENT] tab, the website will transfer [Equipment Page]. Then, guest clicks on categories tab, the website will show list equipment belong to that category.						
Preconditions: N/A						
Post Conditions: Success: Guest can view all equipment in the system or according to each category. Fail: N/A						
Main Success Scenario:						
<table border="1"> <thead> <tr> <th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1. On the [Home Page], guest clicks on [EQUIPMENT] tab.</td><td>2. The website will transfer [Equipment Page] and load all equipment.</td></tr> </tbody> </table>			Actor Action	System Response	1. On the [Home Page], guest clicks on [EQUIPMENT] tab.	2. The website will transfer [Equipment Page] and load all equipment.
Actor Action	System Response					
1. On the [Home Page], guest clicks on [EQUIPMENT] tab.	2. The website will transfer [Equipment Page] and load all equipment.					

2. Guest clicks on category tabs.	4. The website will show all the equipment of that category.
Alternative Scenario: N/A	
Exceptions: N/A	
Relationships: View Equipment use case.	
Business Rules: When [Equipment Page] is loaded, default value category tab is [All Product]. Guest can view all equipment belong to each category or all equipment in the system.	

2.2.4. <Guest> [View Equipment] View Equipment Detail

2.2.4.1. Use Case Diagram

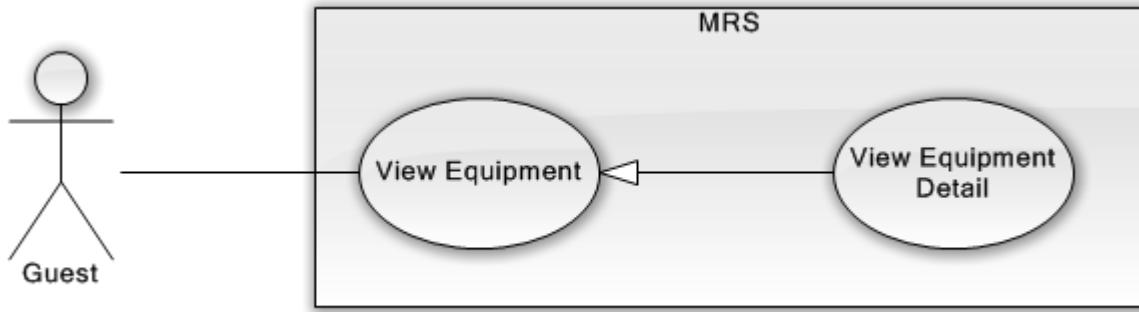


Figure 14- View Equipment Detail usecase diagram

2.2.4.2. Use Case Specification

USE CASE – View Equipment Detail SPECIFICATION			
Use case No.	UC004	Use case version	2.0
Use case name	View Equipment Detail		
Author	DatTQ		
Date	21/05/2013	Priority	Normal
Actor: Guest			
Summary: With View Equipment Detail function, Guest clicks on specific equipment and can view all details of that equipment easily.			
Goal:			

Guest can view all details of equipment.

Triggers:

On the [Home Page], guest clicks on [EQUIPMENT] tab. The website will transfer [Equipment Page]. Then, guest clicks on the equipment, the website will show the detail of that equipment.

Preconditions:

N/A

Post Conditions:

Success: Display detail of the equipment when guest clicks on that equipment.

Fail: N/A

Main Success Scenario:

Actor Action	System Response
1. On the [Home Page], guest clicks on [EQUIPMENT] tab.	2. The website will transfer [Equipment Page] and load all equipment.
3. Click on any the equipment.	4. The website will show detail of the equipment.

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

View equipment use case, View Equipment by Category use case.

Business Rules:

When guest clicks on the equipment, the detail of that equipment will be shown.

2.2.5. <Guest> [View Equipment] Search Equipment

2.2.5.1. Use Case Diagram

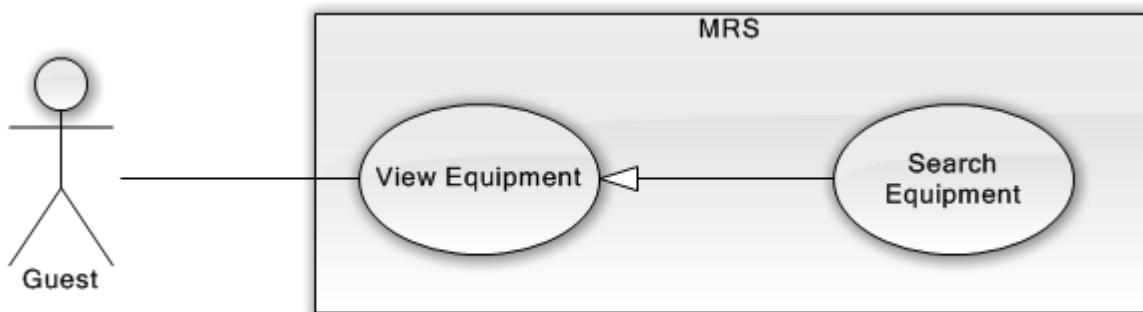


Figure 15 - Search Equipment usecase diagram

2.2.5.2. Use Case Specification

USE CASE – Search Equipment SPECIFICATION								
Use case No.	UC005	Use case version	2.0					
Use case name	Search Equipment							
Author	KieuDTD							
Date	8/8/2013	Priority	Normal					
Actor: Guest								
Summary: With Search Equipment function, Guest can search quickly equipment that they attention. The system searches by Name of equipment and show result below.								
Goal: Guest can search equipment by inputting equipment's name.								
Triggers: In [Equipment Page], Guest inputs equipment's name into text box and click [Search] button to finish.								
Preconditions: N/A								
Post Conditions: Success: The result of searching would be shown with equipment list format. Fail: N/A								
Main Success Scenario:								
<table border="1"> <thead> <tr> <th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1. On the [Home Page], guest clicks on [EQUIPMENT] tab.</td><td>2. The website will transfer [Equipment Page] and load all equipment.</td></tr> <tr> <td>3. Input equipment's name into text box and click [Search] button.</td><td>4. The website will show search result of the equipment list format.</td></tr> </tbody> </table>			Actor Action	System Response	1. On the [Home Page], guest clicks on [EQUIPMENT] tab.	2. The website will transfer [Equipment Page] and load all equipment.	3. Input equipment's name into text box and click [Search] button.	4. The website will show search result of the equipment list format.
Actor Action	System Response							
1. On the [Home Page], guest clicks on [EQUIPMENT] tab.	2. The website will transfer [Equipment Page] and load all equipment.							
3. Input equipment's name into text box and click [Search] button.	4. The website will show search result of the equipment list format.							
Alternative Scenario: N/A								
Exceptions: N/A								
Relationships: View equipment use case.								
Business Rules: When guest searches equipment, the result will be shown below. If there is no equipment was found, display message "No result found".								

2.2.6. <Guest> Tracking Request Process

2.2.6.1. Use Case Diagram



Figure 16 - Tracking Request Process usecase diagram

2.2.6.2. Use Case Specification

USE CASE – Tracking Request Process SPECIFICATION			
Use case No.	UC006	Use case version	2.0
Use case name	Tracking Request Process		
Author	KieuDTD		
Date	14/08/2013	Priority	Normal
Actor: Guest			
Summary: When Customer has a request to maintain/repair, they will be provided an account to login for tracking this request status. However, some case that customer do not want to login only for tracking status and do nothing other. With using this function, customer do not need to login but can track the request status.			
Goal: Customer can track their request without login.			
Triggers: In [Customer Service Page], Guest input search value into [Request Status] and click Search button to finish.			
Preconditions: Guest must be a customer of the system, who has product to maintain/ repair			
Post Conditions: Success: Current request will be shown on table format with request's information: No, Request name, Model, Status, and Request Date Fail: N/A			
Main Success Scenario:			

Actor Action	System Response
1. Guest clicks on [Service] box on [Home] page.	2. Display [Customer Service Page] with [Request Status] form
3. Input search value: request code, email or phone number and clicks on [Search] button.	<p>4. The result of current request will be shown on [Request List] table with columns:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Request name: Name of request - Request Code: code of request - Status: Current status of that request - Date: The date that request started. - Created By: Name of created person

Alternative Scenario:
N/A

Exceptions:
N/A

Relationships:
N/A

Business Rules:
If there cannot find request, show message “No Results Found.”

2.2.7. <Registered User> Login

2.2.7.1. Use Case Diagram

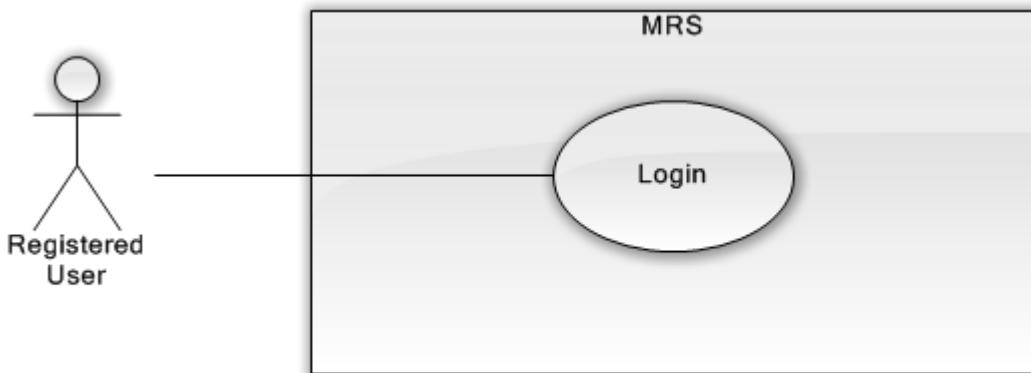


Figure 17 – Login usecase diagram

2.2.7.2. Use Case Specification

USE CASE – Login SPECIFICATION			
Use case No.	UC007	Use case version	2.0
Use case name	Login		
Author	DatTQ		
Date	15/05/2013	Priority	Normal

Actor:

Registered User: Customer, Staff, and Technical Manager

Summary:

-Customer, Staff, Technical Manager are the Registered User who have the account to access the website. They can login to use the correlative functions with their role.

-With the login function, Registered User becomes the customer, staff or technical manager to access the website.

Goal:

Allow user access to the website and use the correlative functions with their role.

This use case also helps the website authenticate user accounts that belong to website.

Triggers:

-On the [Home Page], Registered User inputs username and password, then click [Login] button to finish.

-On the [Register Page], Registered User can input username and password in [Login] form, then click [Login] button to finish.

Preconditions:

Registered User must have an account which can login to the website.

Post Conditions:

Success: Registered User logsins successfully in the website

Fail:

- When Registered User logsins with wrong username or password, display error message
- When Registered User logsins with wrong role, redirect to [Error Page]

Main Success Scenario:

Actor Action	System Response
1. Registered User accesses the [Home Page] of the website.	<p>2. The website shows [Home Page] with header includes login form:</p> <ul style="list-style-type: none"> - Username: Textbox (min length: 6, max length: 20) - Password: Textbox (min length: 6, max length: 20, type: password field) - Login: Button

	- Register: Button
3. Enter information into text fields.	4. [Exception 1,3,5]
5. Click on [Login] button to finish.	5. The website will transfer the current page.

In case of Login at Register Page, references Main Success Scenario above.

Alternative Scenario:

N/A

Exceptions:

Actor Action	System Response
1. Username or password is blank.	2. Display error message: “Please fill out this field.6 to 20 characters [a-zA-Z0-9]”.
3. Username or password is less than 6 characters or greater than 20 characters.	4. Display error message: “Please match the requested format.6 to 20 characters [a-zA-Z0-9]”.
5. Username or password is incorrect.	6. Display error message: “Username or password is incorrect.”

Relationships:

Register use case, [Manage User] Add New Account use case.

Business Rules:

With the registered account, Registered User can login to the website and will be authorized with role of account.

After login success, region of login is hidden and appear “Hello [username]”, profile image , icon notification and [Logout] button on the right

2.2.8. <Registered User> [Update Profile] Update Basic Information

2.2.8.1. Use Case Diagram

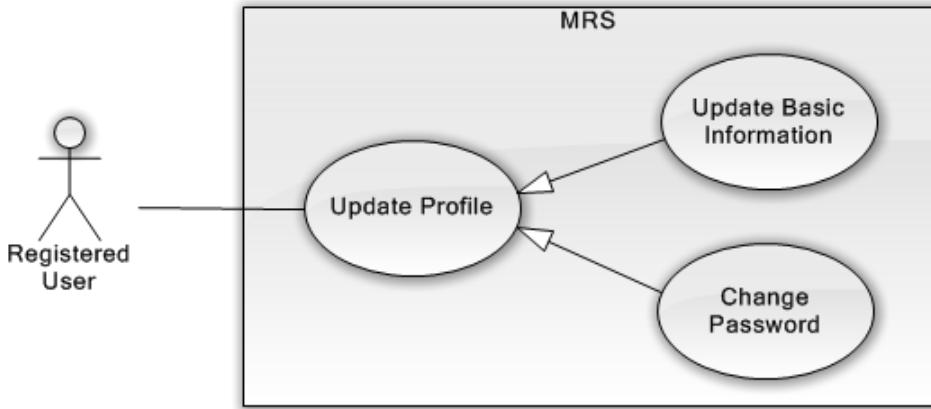


Figure 18-Update Profile use case diagram

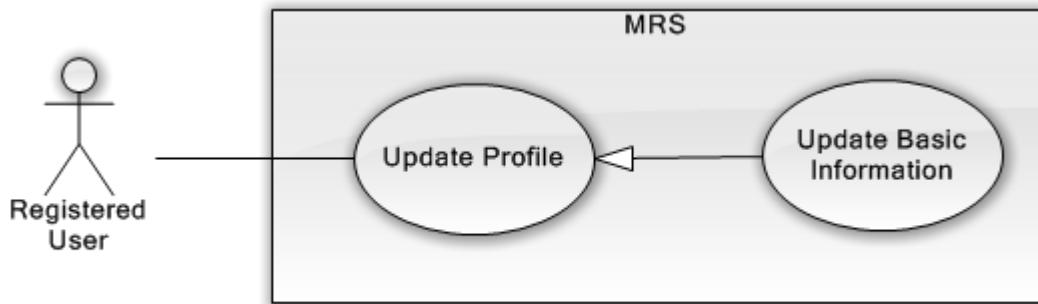


Figure 19 - Update Basic Information usecase diagram

2.2.8.2. Use Case Specification

USE CASE – Update Basic Information SPECIFICATION			
Use case No.	UC008	Use case version	2.0
Use case name	Update Basic Information		
Author	DatTQ		
Date	21/05/2013	Priority	Normal
Actor: Registered User: Customer, Staff and Technical Manager			
Summary: When Registered User was login to the website successfully, become Logged User. They want to change the information in their account. With Update Basic Information function, they			

can update the information easily.

Goal:

Logged User can update all information in their account in which already existed.

Triggers:

In [Profile Page], Logged User changes the information in [Basic Information] form and clicks on [UPDATE] button.

Preconditions:

-Registered User has logged to the website.

Post Conditions:

Success: The account's information of Logged User will be updated.

Fail: The account's information does not update to database and display error message.

Main Success Scenario:

Actor Action	System Response
1. In [Home Page], Logged User clicks on Username Link.	<p>2. The website will transfer [Profile Page] and show [Basic Information] form includes:</p> <ul style="list-style-type: none"> - First Name: Textbox (min length: 6, max length: 20) - Last Name: Textbox (min length:6, max length:20) - Email: Textbox - Gender: Combo box - Phone Number: Number - Address: Text area - UPDATE: Submit Button
5. Logged User changes the information in [Basic Information] form.	6. [Exception 1,3,5]
7. Click on [UPDATE] button to finish.	8. Display pop up: "The account's information was updated successfully."

Alternative Scenario:

N/A

Exceptions:

Actor Action	System Response
1. Phone Number, Address is blank.	2. Display error message: "Please fill out this field." on the field which is blank.
3. Phone Number is not a number.	4. Display error message: "Phone must be

	a number.”
5. The Logged User changes the phone that is less than 9 numbers or greater than 11 numbers.	6. Display error message: “Please match the requested format. 9 to 11 characters [0-9].”

Relationships:
Login use case, Update Profile use case.
Business Rules:
When Logged User click Update button, profile information will be updated to database.

2.2.9. <Registered User > [Update Profile] Change Password

2.2.9.1. Use Case Diagram

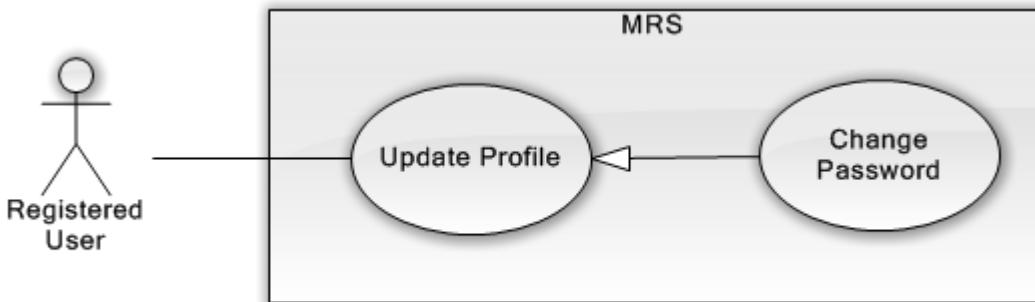


Figure 20 - Change Password usecase diagram

2.2.9.2. Use Case Specification

USE CASE – Change Password SPECIFICATION			
Use case No.	UC009	Use case version	2.0
Use case name	Change Password		
Author	DatTQ		
Date	21/05/2013	Priority	Normal
Actor: Registered User: Customer, Staff and Technical Manager			
Summary: When Registered User was login to the website successfully, become Logged User. They want to change the password in their account. With Change Password function, they can change the password easily.			
Goal: Logged User can change the password in their account in which already existed.			
Triggers:			

In [Change Password Page], Logged User changes the password and clicks on [CHANGE] button.

Preconditions:

-Registered User has logged successfully.

Post Conditions:

Success: The account's password will be changed successfully.

Fail: The account's password does not change to database. Display error message.

Main Success Scenario:

Actor Action	System Response
1. In [Home Page], Logged User clicks on Username Link.	2. The website will transfer [Profile Page].
3. Click on [Change Password] link.	4. The website will transfer [Change Password Page] and show [Change Password] form includes: <ul style="list-style-type: none"> - Old Password: Password Field (min length: 6, max length: 20) - New Password: Password Field (min length:6, max length:20) - Confirm Password: Password Field. - CHANGE: Button
5. Logged User changes the password in [Change Password] form.	6. [Exception 1,3,5]
7. Click on [CHANGE] button to finish.	8. Display message:" The account's password was changed successfully. "

Alternative Scenario:

N/A

Exceptions:

Actor Action	System Response
1. Old Password, New Password, Confirm Password is blank.	2. Display error message: "Please fill out this field." on the field which is blank.
3. Enter Old Password is different with the password when Logged User creates new account.	4. Display error message: "Password is wrong."
5. Confirm Password is different with	6. Display error message: "Confirm

New Password.	Password does not match."
Relationships:	
Login use case, Update Profile use case.	
Business Rules:	
When Logged User changes the password successfully, the website will redirect to [Profile Page].	

2.2.10. <Registered User> Logout

2.2.10.1. Use Case Diagram

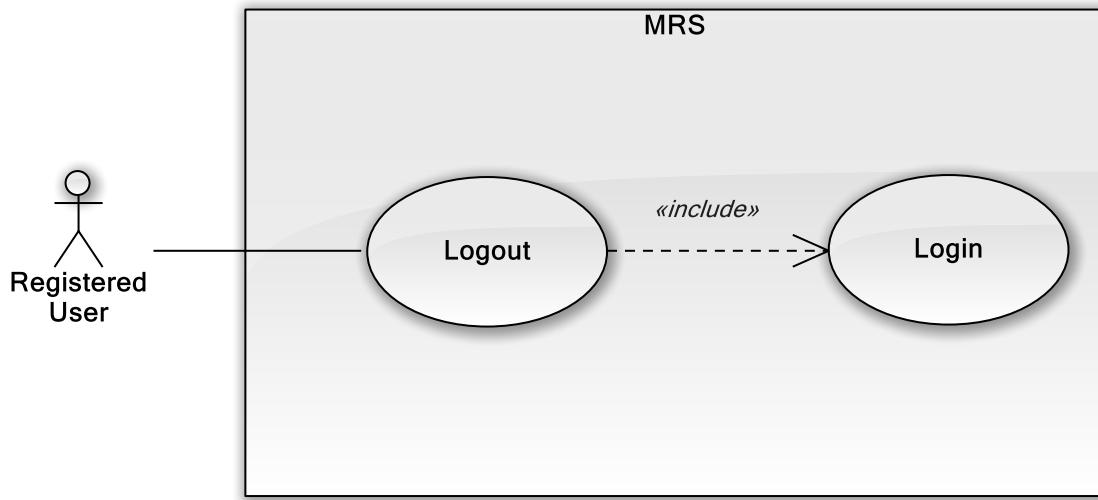


Figure 21 – Logout usecase diagram

2.2.10.2. Use Case Specification

USE CASE – Logout SPECIFICATION			
Use case No.	UC010	Use case version	2.0
Use case name	Logout		
Author	DatTQ		
Date	15/05/2013	Priority	Normal
Actor:	Registered User: Customer, Staff, and Technical Manager		
Summary:	Registered User was login to the website and become Logged User. With logout function, Logged User becomes “Guest” role.		
Goal:	Logged User can log out the website and become “Guest” role.		

Triggers:

Logged User clicks on [Logout] button and the website will redirect to [Home Page].

Preconditions:

Registered User must login into the website.

Post Conditions:

Success: Logged User moves to [Home Page].

Fail: N/A

Main Success Scenario:

Actor Action	System Response
1. Logged User clicks on [Logout] button.	2. The website will redirect to [Home Page].

Alternative Scenario:

Actor Action	System Response
1. Don't use the website about 30 minutes.	2. The account of Logged user will be automatically logged out.

Exceptions:

N/A

Relationships:

Login use case.

Business Rules:

-When logged user wants not to use the website for a moment, they will logout of the website.

-When logged user does not use the website about 30 minutes, their account will be automatically logged out

2.2.11. <Customer> [Tracking Request] View Request List

2.2.11.1. Use Case Diagram

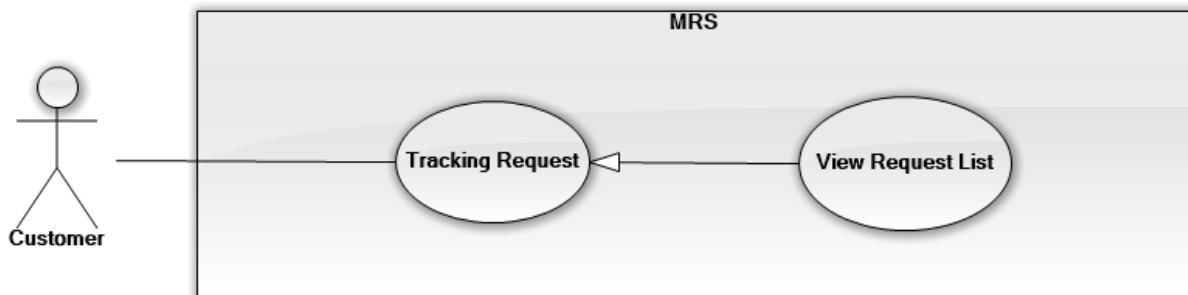


Figure 22 - Customer View Request List usecase diagram

2.2.11.2. Use Case Specification

USE CASE – View Requests List SPECIFICATION

Use case No.	UC011	Use case version	2.0
Use case name	View Requests List		
Author	ThanhTM		
Date	21/05/2013	Priority	Normal

Actor:

Customer.

Summary:

Customer uses this case to review the repair/maintenance requests list related to their product. By using this use case, customer can control all repair/maintenance requests, which problem their product is, and the status of request.

Goal:

Customers want to track the repair/maintenance requests of their product, so this use case help them to get in the tracking request and view requests list of their product

Triggers:

Customer can track the repair/maintenance history by click on [Request] box at [Home] page, then the website will display [Customer Request] page.

Preconditions:

User logged in successfully with role [Customer]

Post Conditions:

- **Success:** Current requests of customer will be displayed as a table with request's information: No, Request name, Model, Status, and Request Date.
- **Fail:** N/A.

Main Success Scenario:

Actor Action	System Response
1. Customer clicks on [Request] box on home page.	<p>2. Redirect to [Customer Request] page.</p> <p>[Customer Request] page by default displays current requests' information of customer in [Request List] table with columns:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Request name: Name of request - Status: Current status of that request - Model: Model of product - Request Date: date that request was created.

Alternative Scenario:

N/A

Exceptions:

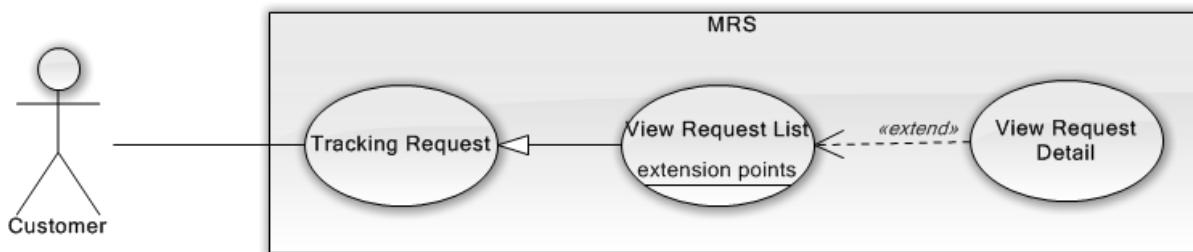
N/A

Relationships:

Tracking request

Business Rules:

- Customers are repairing/maintaining their product, and they want to track the progress, status of the current requests, they use this use case.
- By default, the [Customer Request] page displays current requests list relate to customer in table.
- All current requests will display in a list with order from latest to oldest updated date.

2.2.12. < Customer > [Tracking Request] View Request Detail**2.2.12.1. Use Case Diagram****Figure 23 - Customer View Request Detail usecase diagram****2.2.12.2. Use Case Specification**

USE CASE – View Request Detail SPECIFICATION			
Use case No.	UC012	Use case version	2.0
Use case name	View Request Detail		
Author	KieuDTD		
Date	8/8/2013	Priority	Normal
Actor: Customer			
Summary: Customer uses this case to view detail of request. They could track not only process of maintenance repairing but also products damage, equipment used.			
Goal: Customer can have clearly detail about their request.			
Triggers: In [Customer Request], Customer has to click on the request that he/she want to view detail to			

display [Request Detail] form.

Preconditions:

User logged in successfully with role [Customer]

Post Conditions:

- **Success:** Detail of selected request will be shown on [Request Detail] form on pop up.
- **Fail:** N/A

Main Success Scenario:

Actor Action	System Response
1. Customer clicks on [Request] box on home page.	2. Redirect to [Customer Request] page [Customer Request] page by default displays current requests' information of customer in [Request List] table
5. Customer clicks on a row in body of [Request List] table which status is "Pending"	6. Display request detail form with: <ul style="list-style-type: none"> - Request name - Model: model of product request - Status - Equipment: equipment use for maintain/repair - Confirm content - CLOSE: button - CANCEL: button - CONFIRM: button

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Tracking request, View Request List

Business Rules:

- In case of status of request is "Pending", system will display [CANCEL] and [CONFIRM] button.
- In case of status of request is "Finished", system will display [CLOSE] button.

2.2.13. <Customer> [Tracking Request] Update Request

2.2.13.1. Use Case Diagram

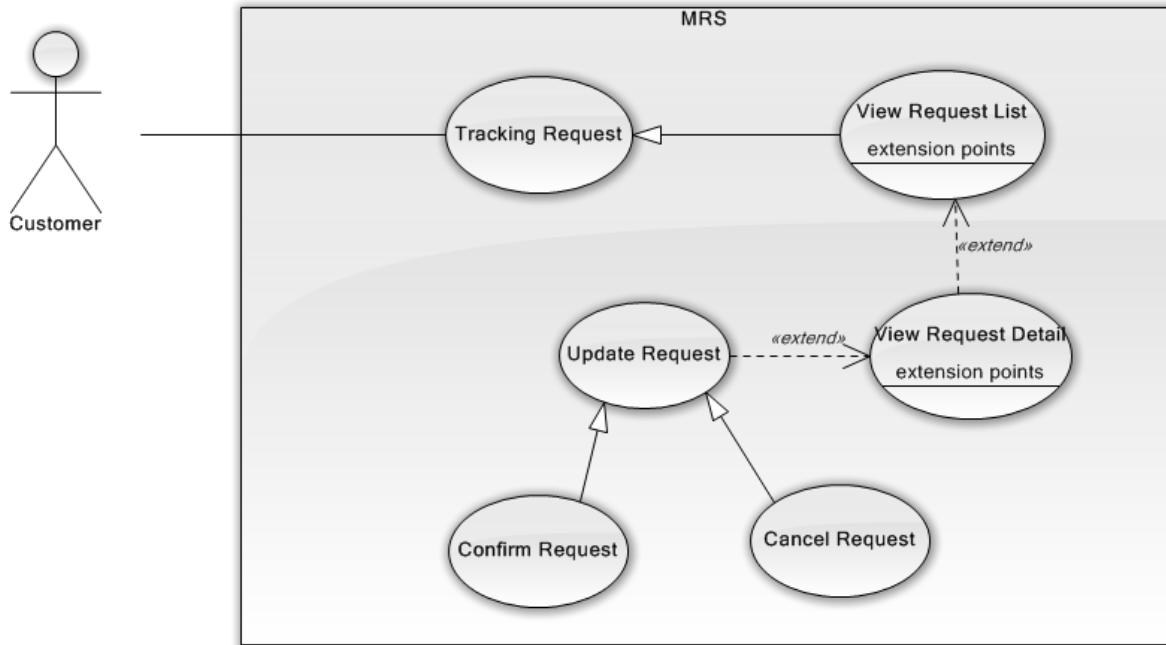


Figure 24 - Customer Update Request usecase diagram

2.2.13.2. Use Case Specification

USE CASE – Update Request SPECIFICATION			
Use case No.	UC013	Use case version	2.0
Use case name	Update Request		
Author	KieuDTD		
Date	21/05/2013	Priority	Normal
Actor: Customer			
Summary:	<ul style="list-style-type: none"> - Customer uses this case to confirm or cancel the repair/maintenance requests which status is “Pending”. When doing confirmation, customer can enter some comment about repairing their product, such as:” Just replace the Display, don’t replace the Keyboard”. - By using this case, customer can control the information of the request and reply to Technical Manager their decision on repairing/maintaining request. When customer updates the request, the system will automatic push a notification about the request to the Technical Manager. 		
Goal:			

When the [Technical Manager] creates a new request, the Customer may be there or not to confirm the request. In case the Customer is not there, the [Technical Manager] will call them to notice about the request. And the Customer will enter their [Tracking Request] page, check for the information of the request and do update the request.

Triggers:

In [Customer Request] page, Customer has to click on the request that he/she want to update to display [Request Detail] form of that request.

Preconditions:

User logged in successfully with role [Customer]

Post Conditions:

- **Success:** The request's status will be updated successful and the system will automatic push a notification about the request to the Technical Manager.
- **Fail:** Redirect to [Error Page] or still at current page with error messages.

Main Success Scenario:

Actor Action	System Response
1. Customer clicks on [Request] box on home page.	2. Redirect to [Customer Request] page [Customer Request] page by default displays current requests' information of customer in [Request List] table
5. Customer clicks on a row in body of [Request List] table	6. Display the request detail form with: <ul style="list-style-type: none"> - Request name - Model: model of product request - Status - Equipment: equipment use for maintain/repair - Confirm content - CLOSE: button - CANCEL: button - CONFIRM: button
7. Customer clicks on [CANCEL] or [CONFIRM] button.	8. The request's status will be updated successful and the system will automatic push a notification about the request to the Technical Manager.

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Tracking request

Business Rules:

- Customers were not at the Service Center to confirm the request when the Technical Manager finish checking the product and create the request.
- Be noticed by the Technical Manager, Customer get in the Control Panel to view the request detail and confirm or cancel it.
- When customers click [CANCEL] or [CONFIRM], the request's status will be updated successful and the system will automatic push a notification about the request to the Technical Manager.
- Customers can enter into the [Customer Confirmation] text area or not.
- When status of Request is not "Pending", system accept for view request detail and display [CLOSE] button.

2.2.14. < Staff > [Tracking Task] View All Tasks List

2.2.14.1. Use Case Diagram

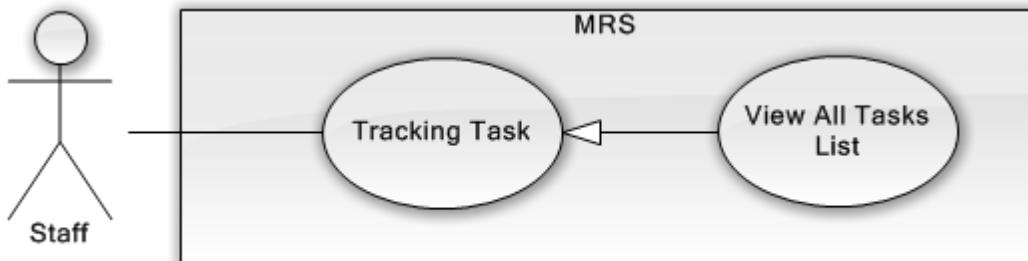


Figure 25 - Staff View All Task List usecase digram

2.2.14.2. Use Case Specification

USE CASE – View All Tasks List SPECIFICATION			
Use case No.	UC014	Use case version	2.0
Use case name	View All Tasks list		
Author	KieuDTD		
Date	23/05/2013	Priority	Normal
Actor: Staff			
Summary: Each staff log in to the system could track all tasks that he/she was assigned by "View All Tasks List" function. This will help staffs to manage all their tasks since the past.			
Goal: Staff can view all tasks that they were assigned in list with table format.			
Triggers: In [Home] Page, staff clicks on [TASK] box, in [Staff Tracking Task] Page, staff clicks on			

[All Task] tab, [All Task] page will display with table [All Task List].

Preconditions:

User logged successfully with role [Staff]

Post Conditions:

- Success: Display [Task List] table contains all tasks that they was assigned
- Fail: N/A

Main Success Scenario:

Actor Action	System Response
1. Staff clicks on [TASK] box on [Home] Page.	2. Redirect to [Staff Tracking Task] page
3. Staff clicks on [All Task] tab	<p>4. Open [Staff Tracking Task] page with [All Task List] table includes:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Task name: Name of task - Status: Current status of that task - Start date: The date which task started. - Finish date: The date which task finished. <p>Activation: show that that task in effect.</p>

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Login use case, Tracking Task use case

Business Rules:

- Staff can choose the maximum item to display on list by 10, 15, 25, etc.
- If the number of task is less than the number of item you want to display on a page, the list will not show paging.
- If there are no task assigned, show message "There are no task is assigned."
- Background color of row of [All Task List] table would change to describe for the task have missed deadline notice:
 - o The background color is orange if the task is going to be missed deadline.
 - o The background color is red if the task is missed deadline.

2.2.15. < Staff > [Tracking Task] View New Tasks List

2.2.15.1. Use Case Diagram

2.2.15.2. Use Case Specification

USE CASE – View New Tasks List SPECIFICATION							
Use case No.	UC015	Use case version	2.0				
Use case name	View New Tasks list						
Author	KieuDTD						
Date	23/05/2013	Priority	Normal				
Actor: Staff							
Summary:	Each staff will have many tasks with many kind of status to maintain/repair. So that the task management will be difficult if all tasks are displayed in a table list. This function allows the staff to view their new tasks.						
Goal:	Staff can view all their new tasks that they were assigned in list with table format.						
Triggers:	In [Home] Page, staff clicks on [TASK] box, [Staff Tracking Task] Page will be displayed with table [New Task List].						
Preconditions:	User logged in successfully with role [Staff]						
Post Conditions:	<ul style="list-style-type: none"> - Success: Display [Task List] table contains all new tasks that they was assigned - Fail: N/A 						
Main Success Scenario:	<table border="1"> <thead> <tr> <th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1. Staff clicks on [TASK] box on [Home] Page.</td><td> <p>2. Open [Staff Tracking Task] page with [Task List] table includes:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Task name: Name of task - Status: Current status of that task - Start date: The date which task started. - Finish date: The date which task finished. - Activation: show that that task in </td></tr> </tbody> </table>			Actor Action	System Response	1. Staff clicks on [TASK] box on [Home] Page.	<p>2. Open [Staff Tracking Task] page with [Task List] table includes:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Task name: Name of task - Status: Current status of that task - Start date: The date which task started. - Finish date: The date which task finished. - Activation: show that that task in
Actor Action	System Response						
1. Staff clicks on [TASK] box on [Home] Page.	<p>2. Open [Staff Tracking Task] page with [Task List] table includes:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Task name: Name of task - Status: Current status of that task - Start date: The date which task started. - Finish date: The date which task finished. - Activation: show that that task in 						

	effect.
--	---------

Alternative Scenario:
N/A

Exceptions:
N/A

Relationships:
Login use case, Tracking Task use case

Business Rules:

- Staff can choose the maximum item to display on list by 10, 15, 25, etc.
- If the number of task is less than the number of item you want to display on a page, the list will not show paging.
- If there are no task assigned, show message “There are no task is assigned.”
- Background color of row of [New Task List] table would change to describe for the task have missed deadline notice:
 - o The background color is orange if the task is going to be missed deadline.
 - o The background color is red if the task is missed deadline.

2.2.16. <Staff> [Tracking Task] View In-Progress Tasks List

2.2.16.1. Use Case Diagram

2.2.16.2. Use Case Specification

USE CASE – View In-Progress Tasks List SPECIFICATION			
Use case No.	UC016	Use case version	2.0
Use case name	View In-Progress Tasks list		
Author	KieuDTD		
Date	23/05/2013	Priority	Normal
Actor: Staff			
Summary: Each staff will have many tasks with many kind of status to maintain/repair. So that the task management will be difficult if all tasks are displayed in a table list. This function allows the staff to view their working tasks.			
Goal: Staff can view all their tasks that they were doing in list with table format.			
Triggers: In [Home] Page, staff clicks on [TASK] box, in [Staff Tracking Task] Page, staff clicks on [In-progress Task] tab, [In-progress Task] page will display with table [In-progress Task List].			
Preconditions: User logged in successfully with role [Staff]			

There must be at least 1 task was doing.

[Staff Tracking Task] Page is displayed without error.

Post Conditions:

- Success: Display [Task List] table contains all in-progress tasks that they was assigned
- Fail: N/A

Main Success Scenario:

Actor Action	System Response
1. Staff clicks on [TASK] box on [Home] Page.	2. Redirect to [Staff Tracking Task] page
3. Staff clicks on [In-progress Task] tab	4. Open [Staff Tracking Task] page with [In-progress Task List] table includes: <ul style="list-style-type: none"> - No.: Ordinal number - Task name: Name of task - Status: Current status of that task - Start date: The date which task started. - Finish date: The date which task finished. - Activation: show that that task in effect.

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Login use case, Tracking Task use case

Business Rules:

- Staff can choose the maximum item to display on list by 10, 15, 25, etc.
- If the number of task is less than the number of item you want to display on a page, the list will not show paging.
- If there is no task that the staff is doing, show message "There is no task in progress."
- Background color of row of [Task List] table would change to describe for the task have missed deadline notice:
 - o The background color is orange if the task is going to be missed deadline.
 - o The background color is red if the task is missed deadline.

2.2.17. < Staff > [Tracking Task] View Finished Tasks List

2.2.17.1. Use Case Diagram

2.2.17.2. Use Case Specification

USE CASE – View Finished Tasks List SPECIFICATION					
Use case No.	UC017	Use case version	2.0		
Use case name	View Finished Tasks list				
Author	KieuDTD				
Date	23/05/2013	Priority	Normal		
Actor: Staff					
Summary: Each staff will have many tasks with many kind of status to maintain/repair. So that the task management will be difficult if all tasks are displayed in a table list. This function allows the staff to view their finished tasks.					
Goal: Staff can view all their tasks that they were finished in list with table format.					
Triggers: In [Home] Page, staff clicks on [TASK] box, in [Staff Tracking Task] Page, staff clicks on [Finished Task] tab, [Finished Task] page will display with table [Finished Task List].					
Preconditions: User logged in successfully with role [Staff] There must be at least 1 task was finished. [Staff Tracking Task] Page is displayed without error.					
Post Conditions: - Success: Display [Task List] table contains all finished tasks that they was assigned - Fail: N/A					
Main Success Scenario:					
	Actor Action	System Response			
1. Staff clicks on [TASK] box on [Home] Page.	2. Redirect to [Staff Tracking Task] page				
3. Staff clicks on [Finished Task] tab	4. Open [Staff Tracking Task] page with [Finished Task List] table includes: - No.: Ordinal number - Task name: Name of task - Status: Current status of that task - Start date: The date which task				

	<p>started.</p> <ul style="list-style-type: none"> - Finish date: The date which task finished. - Activation: show that that task in effect. 	
--	--	--

Alternative Scenario:
N/A

Exceptions:
N/A

Relationships:
Login use case, Tracking Task use case

Business Rules:

- Staff can choose the maximum item to display on list by 10, 15, 25, etc.
- If the number of task is less than the number of item you want to display on a page, the list will not show paging.
- If there is no finished task, show message “There is no finished task.”
- Background color of row of [Task List] table would change to describe for the task have missed deadline notice:
 - o The background color is orange if the task is going to be missed deadline.
 - o The background color is red if the task is missed deadline.

2.2.18. < Staff > [Tracking Task] Filter Task

2.2.18.1. Use Case Diagram

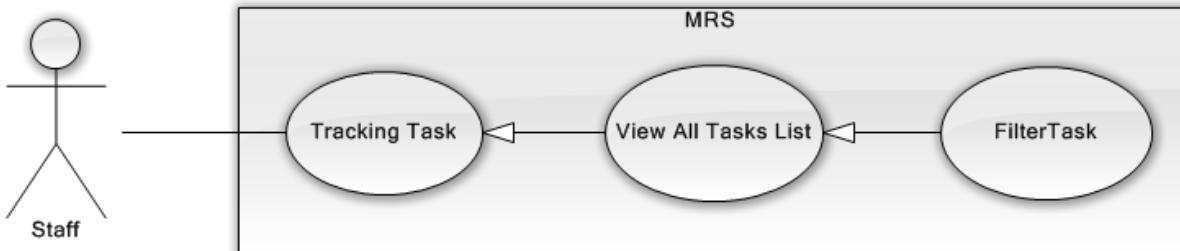


Figure 26 - Staff Filter Task usecase diagram

2.2.18.2. Use Case Specification

USE CASE – Filter Tasks by Status SPECIFICATION			
Use case No.	UC018	Use case version	2.0
Use case name	Filter Task By Status		
Author	KieuDTD		

Date	23/05/2013	Priority	Normal			
Actor: Staff						
Summary: Each staff will have many tasks. So that the task management will be difficult if all tasks are displayed in a table list. Therefore, this function can help staffs to filter tasks. There have two approaches: <ul style="list-style-type: none">+ Filter tasks by Time: display list of tasks which were assigned or finished in the selected time.						
Goal: The tasks list was filtered would be shown in the [Task List] table.						
Triggers: Staff selects the time that they want to filter. Then click on [Filter] button. The result of filtered is shown in [Task List] below.						
Preconditions: User logged in successfully with role [Staff] [Staff Tracking Task] page is displayed without error.						
Post Conditions: <ul style="list-style-type: none">- Success: The result of filtered is shown in [Task List] table- Fail: N/A						
Main Success Scenario:						
Actor Action	System Response					
1. Staff clicks on [TASK] box on [Home] Page.	2. Open [Staff Tracking Task] page with [Task List] table and [Filter area] includes: <ul style="list-style-type: none">- From: Date picker- To: Date picker- Filter: Button					
3. Select the time. Click on [Filter] button	4. The result of filtered is shown in [Task List] table.					
Alternative Scenario: N/A						
Exceptions: N/A						
Relationships: Login use case, Tracking Task use case, View all Tasks List use case, View New Task use case, View In-progress use case, View Finished use case.						
Business Rules: If there are no tasks were found, show message "There are no task found."						

2.2.19. < Staff > [Tracking Task] View Task Detail

2.2.19.1. Use Case Diagram

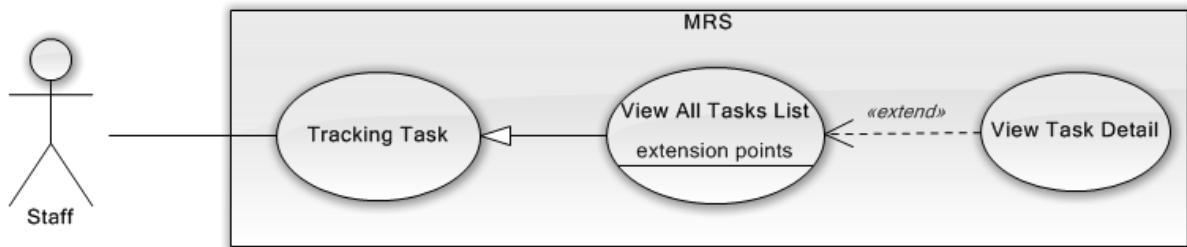


Figure 27 - Staff View Task Detail usecase diagram

2.2.19.2. Use Case Specification

USE CASE – View Task Detail SPECIFICATION					
Use case No.	UC019	Use case version	2.0		
Use case name	View Task Detail				
Author	KieuDTD				
Date	23/05/2013	Priority	Normal		
Actor: Staff					
Summary:	Staff wants to view detail information about one task that they were assigned, They just selects a task in [Task List] table to views more detail about that task.				
Goal:	Task's detail is shown in [Task Detail] form on pop up whenever they click on a row of task list. Staff has a clear detail about task that they want to focus.				
Triggers:	In the [Task List] table, click on a row of task, detail information is shown in [Task Detail] form on pop up.				
Preconditions:	User logged successfully with role [Staff] There must be at least 1 task to view detail. [Staff Tracking Task] page is displayed without error.				
Post Conditions:	<ul style="list-style-type: none"> - Success: Detail information of selected task will be shown in [Task Detail] form on pop up. - Fail: N/A 				
Main Success Scenario:	<table border="1"> <thead> <tr> <th>Actor Action</th><th>System Response</th></tr> </thead> </table>			Actor Action	System Response
Actor Action	System Response				

1. Staff clicks on [Task] box on [Home] Page.	2. Open [Staff Tracking Task] page with [Task List] table.	
3. Click on a row of a new task in [Task List] table	<p>4. Display form [Task Detail] have specify form includes:</p> <ul style="list-style-type: none"> - Request relate: Label - Request information: Label - Task Name: textbox (max length: 150) - Current Status: Label(Status of this task) - Started date: Calendar (The date which task started) - Due date: Calendar (The date which task finished) - Task describe: text area - Equipment Area: Combo box(show equipment for this task) - CLOSE: button (Alternative 1) - START: button - All fields are disabled. 	

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Login use case, Tracking Task use case ,View All Tasks List use case

Business Rules:

- When staff selects task to view detail, display specific [Task Detail] form on pop up
- In case of status of current task is New, display [CLOSE], [START] button, hidden some information not use and [FINISH] button.
- In case of status of current task is In Progress, display [CLOSE], [FINISH] button, hidden [START] button.
- In case of status of current task is Finish, only display [CLOSE] button.

2.2.20. < Staff > [Tracking Task] Update Task

2.2.20.1. Use Case Diagram

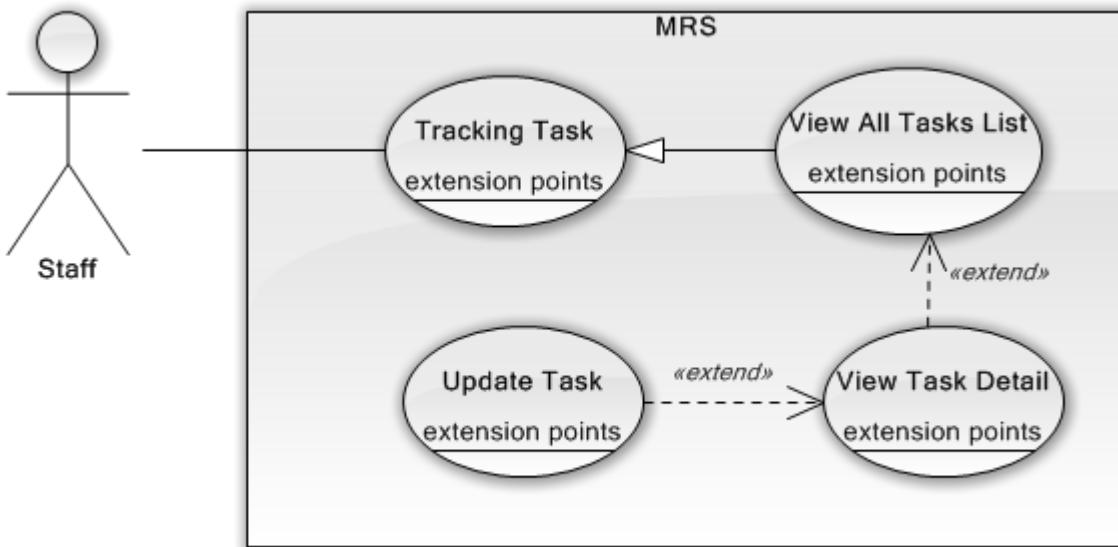


Figure 28 - Staff Update Task usecase diagram

2.2.20.2. Use Case Specification

USE CASE – Update task SPECIFICATION			
Use case No.	UC020	Use case version	2.0
Use case name	Update Task		
Author	KieuDTD		
Date	23/05/2013	Priority	Normal
Actor: Staff			
Summary:	With Update Task function, when staff starts or finishes repairing/maintaining the product of customer, they can update the detail of task: status, more issues, etc. At this time, the system pushes a notification to technical manager about the update, and check for tasks status to change the status of relate request.		
Goal:	Detail of updated task is added to the [Task List] table and saved to database The system notifies to technical manager a change. The system checks for synchronizing the status of tasks and request.		
Triggers:	Staff clicks on a row of [Task List] table, detail of selected task will be shown below with fields that can be edited. If the task is New, staff clicks on [Start] button to start the task. If the		

task is in progress, staff fills in details of task and clicks on [FINISH] button to finish task.

Preconditions:

User logged in successfully with role [Staff]

There must be at least 1 task to update details.

[Staff Tracking Task] Page is displayed without error.

Detail of task is loaded successfully

Post Conditions:

- Success: The status of tasks and relate request are changed correctly. And detail of updated task is added to the [Task List] table and saved to database

- Fail: The status of tasks and relate request are not changed correctly. Display [Error] page.

Detail of updated task is not added to the [Task List] table and not saved to database. Display error message in [Task Detail] form. Entered information is still in form.

Main Success Scenario:

Actor Action	System Response
1. Staff clicks on [Task] box on [Home] Page.	2. Open [Staff Tracking Task] page with [Task List] table.
3. Click on a row of a new task in [Task List] table	<p>4. Display form [Task Detail] have specify form includes:</p> <ul style="list-style-type: none"> - Request relate: Label - Request information: Label - Task Name: textbox (max length: 150) - Current Status: Label(Status of this task) - Started date: Calendar (The date which task started) - Due date: Calendar (The date which task finished) - Task describe: text area - Equipment Area: Combo box(show equipment for this task) - CLOSE: button (Alternative 1) - START: button - All fields are disabled.
5. Click on [START] button to start	6. Status of task and relate request are changed to “In-progress”

7. Click on the row of that task in [Task List] table again.	<p>8. Display form [Task Detail] have specify form includes:</p> <ul style="list-style-type: none"> - Request relate: Label - Request information: Label - Task Name: textbox (max length: 150) - Current Status: Label(Status of this task) - Started date: Calendar (The date which task started) - Due date: Calendar (The date which task finished) - Task describe: text area - Equipment Area: Combo box(show equipment for this task) - More issues: text area (max length: 1000) - Spend hours: Text box - CLOSE: button (Alternative 1) <p>FINISH: button</p>	
9. Fill data in the item which want to update	10. [Exception 1,3]	
11. Click on [FINISH] button to finish.	<p>12. Status of task is changed to “Finished”. System check for all tasks of this request, if all tasks are finished, system will change the status of relate request to finished.</p> <ul style="list-style-type: none"> - Update data to the database 	

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish.	2. Data was filled is cleared.

Exceptions:

Actor Action	System Response

1. More issues are blank	2. Display error message: "Please fill in more issue".	
3. More issues are more than 1000 characters.	4. Display error message: "Please match the requested format. This field must less than 1000 characters [a-zA-Z0-9]"	

Relationships:

Login use case, Tracking Task use case, View all Tasks List use case, View New Task use case, View In-progress use case, View Finished use case, View Task Detail use case

Business Rules:

- Updated Task is saved to the database.
- In case of status of current task is New, display [CLOSE], [START] button, hidden some information not use and [FINISH] button. When staff clicks on [START] button, system will change status of task and relate request to in-progress.
- In case of status of current task is In Progress, display [CLOSE], [FINISH] button, hidden [START] button. When staff clicks on [FINISH] button, system will change status of task to "finished", and check for all tasks status to update the status of relate request.
- In case of status of current task is Finish, only display [CLOSE] button.

2.2.21. < Technical Manager > [Manage FAQ] Create FAQ

2.2.21.1. Use Case Diagram

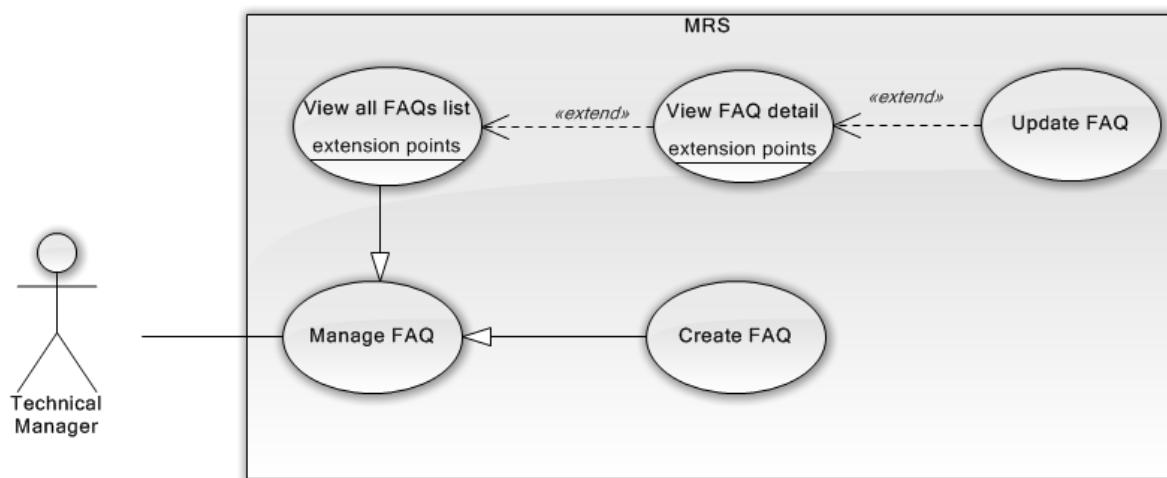


Figure 29 - Manager Mange FAQ usecase diagram

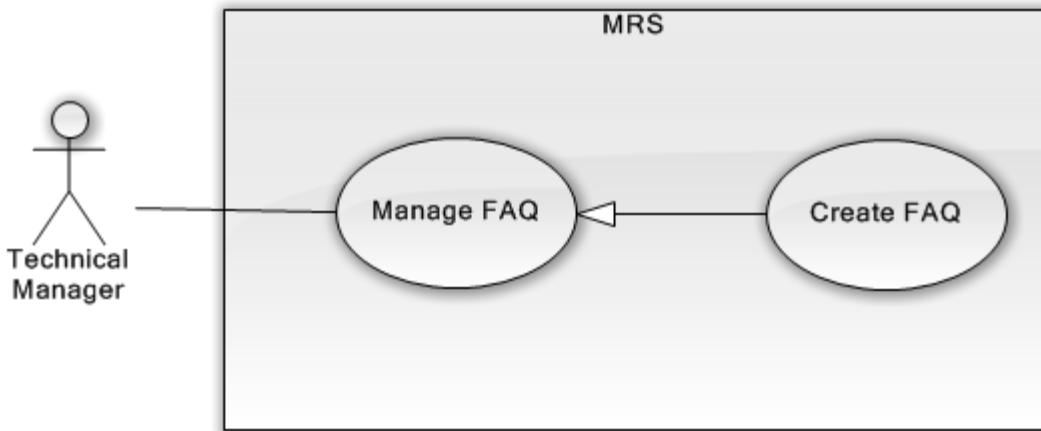


Figure 30-Manager Create FAQ usecase diagram

2.2.21.2. Use Case Specification

USE CASE – Create FAQ SPECIFICATION			
Use case No.	UC021	Use case version	2.0
Use case name	Create FAQ		
Author	DatTQ		
Date	20/05/2013	Priority	Normal
Actor: Technical Manager			
Summary: When Symptom of damage become popular, technical manager decided to add new FAQ into the website. This function helps Technical Manager to create new FAQ easily and this FAQ will be saved in the [FAQ List] table.			
Goal: Technical Manager can create new FAQ in the website.			
Triggers: In [Manage Faq Page], clicks on [ADD NEW] button to appear [Add Question-Answer] form on pop up. Fill the specify form and click on [ADD] button to finish.			
Preconditions: User logged successfully with [Technical Manager] role.			
Post Conditions: Success: New FAQ is added to FAQ list and saved in the database. Fail: New FAQ is not added to FAQ list. Display error message.			
Main Success Scenario:			
Actor Action	System Response		
1. Click [ADD NEW] button in	2. Display [Add Question-Answer] form		

[Manage Faq Page].	has specify form includes: <ul style="list-style-type: none"> - FAQ Subject: Textbox - Brand: Dropdown List - Model: Dropdown List - Content: Text Area - CLOSE: Button [Alternative 1] - ADD: Button 	
3. Fill data in specify form.	4. [Exception 1,3]	
5. Click on [ADD] button to finish.	6. Save new FAQ to [FAQ List] table.	

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button.	2. The website will reset all information that technical manager fills.

Exceptions:

Actor Action	System Response
1. FAQ Subject is blank.	2. Display error message: "Please fill out this field." on the Subject field.
3. Content is blank.	4. Display error message: "Please fill in FAQ content"

Relationships:

Login use case, Manage FAQ use case, Statistic by Symptom use case.

Business Rules:

When technical manager clicks on [ADD] button, the new FAQ will be saved in the [FAQ List] table and [Add Question-Answer] form on pop up will be closed.

2.2.22. < Technical Manager > [Manage FAQ] View all FAQ list

2.2.22.1. Use Case Diagram

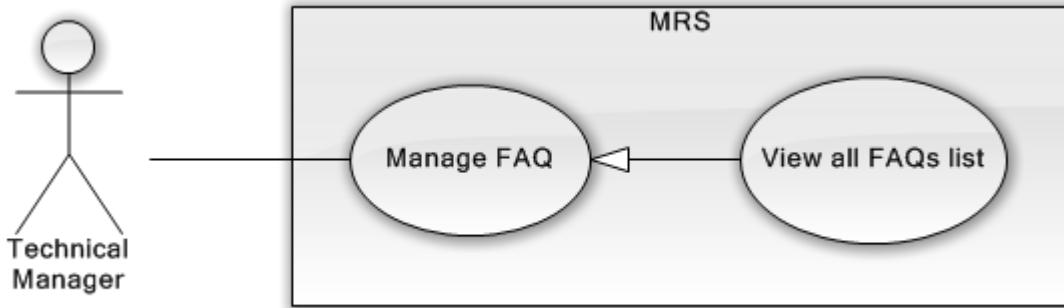


Figure 31 - Manager View All FAQ List usecase diagram

2.2.22.2. Use Case Specification

USE CASE – View all FAQ List SPECIFICATION			
Use case No.	UC022	Use case version	2.0
Use case name	View all FAQ List		
Author	DatTQ		
Date	20/05/2013	Priority	Normal
Actor: Technical Manager			
Summary: Technical Manager had to manage FAQ list. This function helps Technical Manager to manage FAQ easily by display all FAQs in the [FAQ List] table.			
Goal: Technical Manager can view and manage all FAQs which already existed in the website.			
Triggers: In [Manage Faq Page], all FAQs are showed in the [FAQ List] table.			
Preconditions: -User logged successfully with [Technical Manager] role. -The website must have at least one Faq to view.			
Post Conditions: Success: Technical Manager can view all FAQs which are displayed in the [FAQ List] table.			
Fail: N/A			
Main Success Scenario:			
Actor Action	System Response		
1. Technical Manager clicks on	2. Redirect to [Manage Repair]		

[Request] box in [Home] page.	Maintenance] Page
3. Click on [FAQ] tab on tab header.	<p>4. Display all FAQs in the [FAQ List] table includes:</p> <ul style="list-style-type: none"> - Table header: No, Subject, Product Prototype, Active

Alternative Scenario:
N/A

Exceptions:
N/A

Relationships:
Login use case, Manage FAQ use case.

Business Rules:
When Technical Manager selects View all FAQ List function, the website will show all FAQs in the [FAQ List] table.

2.2.23. < Technical Manager > [Manage FAQ] View FAQ detail

2.2.23.1. Use Case Diagram

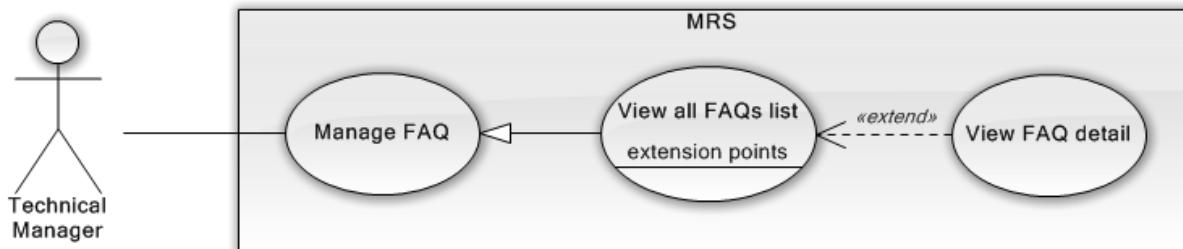


Figure 32 - Manager View FAQ Detail usecase diagram

2.2.23.2. Use Case Specification

USE CASE – View FAQ detail SPECIFICATION			
Use case No.	UC023	Use case version	2.0
Use case name	View FAQ detail		
Author	DatTQ		
Date	20/05/2013	Priority	Normal
Actor:	Technical Manager		
Summary:			

This function helps Technical Manager to view FAQ detail and the FAQ detail will show in the [FAQ Detail] form when clicks a row in body of [FAQ List] table.

Goal:

Technical Manager can view FAQ detail in the website.

Triggers:

In [Manage Faq Page], technical manager clicks on a row in body of [FAQ List] table and the detail of FAQ will be show [FAQ Detail] form on pop up.

Preconditions:

- User logged successfully with role [Technical Manager]
- The website must have at least one Faq to view.

Post Conditions:

Success: Technical Manager can view the detail of all FAQs which are displayed in the [FAQ List] table.

Fail: N/A

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box in [Home] page.	2. Redirect to [Manage Repair Maintenance] Page
3. Click on [FAQ] tab on tab header.	4. Display all FAQs in the [FAQ List] table
5. Click on a row in body of [FAQ List].	6. Display the detail of FAQ in the [FAQ Detail] form includes: -FAQ Subject: Text -Model: Dropdown List -Content: Text Area -CLOSE: Button -UPDATE: Button

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Login use case, Manage FAQ use case, View all FAQ List use case.

Business Rules:

When Technical Manager selects a FAQ to view the detail, the website shows detail information about this selected FAQ in the [FAQ Detail] form.

2.2.24. < Technical Manager > [Manage FAQ] Update FAQ

2.2.24.1. Use Case Diagram

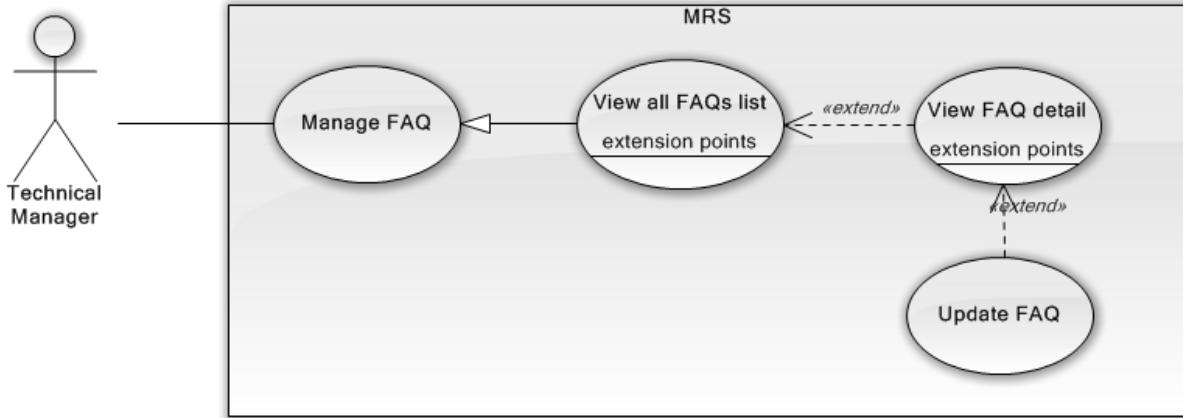


Figure 33 - Manager Update FAQ usecase diagram

2.2.24.2. Use Case Specification

USE CASE – Update FAQ SPECIFICATION			
Use case No.	UC024	Use case version	2.0
Use case name	Update FAQ		
Author	DatTQ		
Date	20/05/2013	Priority	Normal
Actor: Technical Manager			
Summary: When Technical Manager creates a new FAQ successfully. After that, Technical Manager wants to update some information. This updated FAQ function helps Technical Manager to update easily.			
Goal: Technical Manager can update all information of all FAQs which already existed in the website.			
Triggers: In [Manage Faq Page], technical manager clicks on a row in body of [FAQ List] table and the detail of FAQ will be show [FAQ Detail] form on pop up. Technical manager changes the information in the detail of FAQ, and clicks on [UPDATE] button.			
Preconditions: - User logged successfully with role [Technical Manager]. - The website must have at least one symptom to update.			
Post Conditions: Success: The information of FAQ is updated in the [FAQ Detail] form and changed in the			

[FAQ List] table.

Fail: The information of FAQ does not update to database and change in the [FAQ List] table. Display error message.

Main Success Scenario:

Actor Action	System Response
1. Click on a row in body of [FAQ List] table.	4. Display the detail of FAQ in the [FAQ Detail] form includes: -FAQ Subject: Text -Model: Dropdown List - -Content: Text Area -CLOSE: Button (Alternative 1) -UPDATE: Button
2. Change data in the item which is allowed: FAQ Subject, Type, Brand, Active, Question, Answer.	3. [Exception 1]
4. Click on [UPDATE] button to finish.	5. Update the information which was edited to the database.

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish.	2. The information will be cleared.

Exceptions:

Actor Action	System Response
1. Content is blank.	2. Display error message: "Please fill in FAQ Content."

Relationships:

Login use case, Manage FAQ use case, View all FAQ List use case, View FAQ detail use case.

Business Rules:

When technical manager changes the information in [FAQ Detail] form and clicks on [UPDATE] button, FAQ will be updated in [FAQ List] table.

2.2.25. < Technical Manager > [Manage repair/maintenance order] Create Request

2.2.25.1. Use Case Diagram

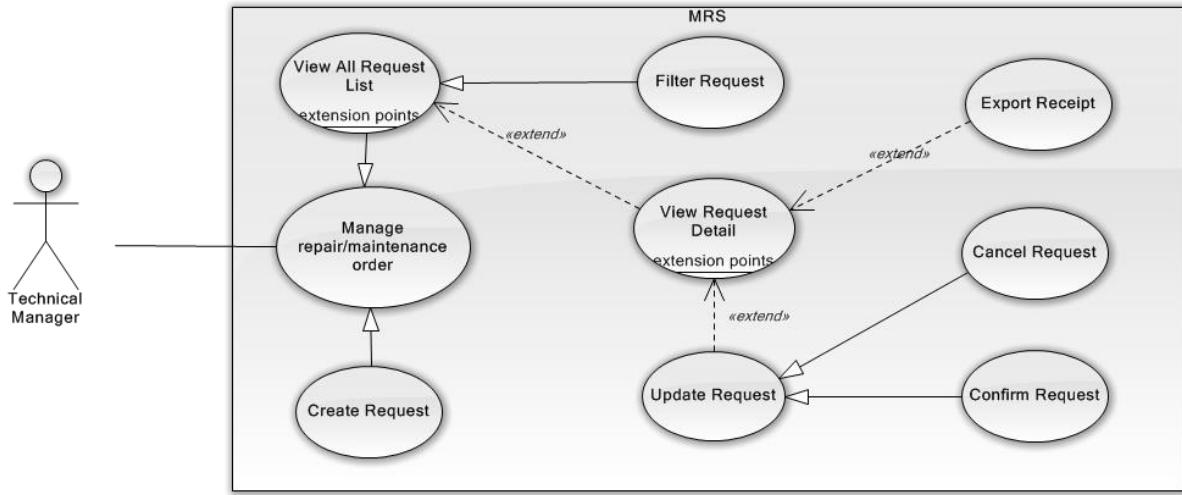


Figure 34- Manger Create Request use case diagram

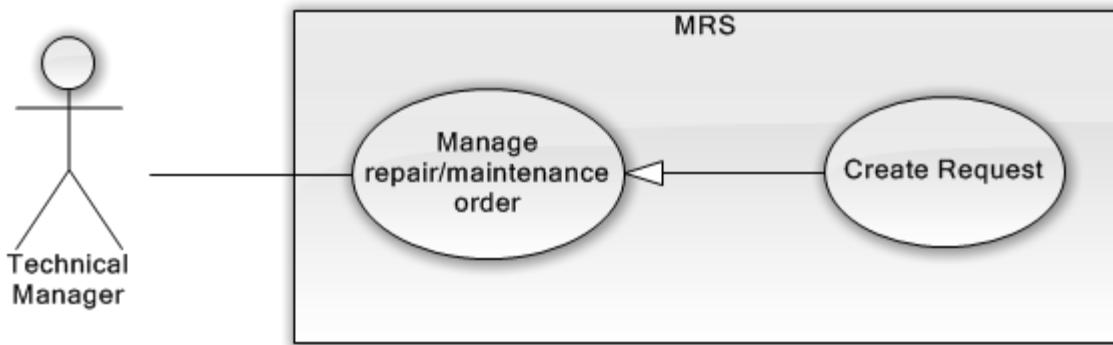


Figure 35-Manager Create Request usecase diagram

2.2.25.2. Use Case Specification

USE CASE – Create Request SPECIFICATION

Use case No.	UC025	Use case version	2.0
Use case name	Create request		
Author	HungDD		
Date	20/06/2013	Priority	Normal
Actor: Technical Manager Summary:			

With Create Request function, technical manager can create a new request from customer by click on [ADD REQUEST] button on [Request List] table. At this time, the system also pushes a notification to customer.

Goal:

New request from customer is added to the [Requests List] and saved to the database
The system notifies about creation success to customer

Triggers:

In [Manage Repair Maintenance] page, click [ADD REQUEST] button to display [Add Request- Product Problem] form. Fill the specify form and click [ADD] button to finish.

Preconditions:

User logged successfully with role [Technical Manager]
[Manage Repair Maintenance] page is displayed without error.

Post Conditions:

Success: New request is added to the [Requests List], saved to the database and notify customer.

Fail: New request is not added to the [Requests List] and not saved to the database. Display error message on [Add Request- Product Problem] form. Entered information is still in form.

Main Success Scenario:

Actor Action	System Response		
1. Technical manager clicks on [REQUEST] box on [Home] page.	2. Display [Manage Repair Maintenance] page with [Request List] table		
3. Click [ADD REQUEST] button on [Request List] table	4. Display form [Add Request- Product Problem] have specify form includes:		
	Basic information	- Request Name: textbox (max length: 100) - Request type: dropdown list - Estimate Finish: Calendar	
	Customer Related	- Customer relate: dropdown list(customer request to maintain-repair)	
	Product information	- Category: dropdown list - Prototype: dropdown list - Model: dropdown list	
	Estimate replace	- Add equipment button. Click on this	

	equipment	<ul style="list-style-type: none"> - button then appear: - Category: dropdown list - Equipment: dropdown list - [quantity]: dropdown list 	
	Request information	<ul style="list-style-type: none"> - Symptom: select symptom of request - Problem description: text area - Customer confirmation: text area - Request tags: request from customer to decided tags appropriate for assign task. 	
	Button	<ul style="list-style-type: none"> - CLOSE: button [Alternative 1] - ADD: button 	
5. Fill data in specify form		6. [Exception 1,3,5]	
7. Click on [ADD] button to finish.		8. Insert new request to the request list and save to database.	

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish.	2. Data in specify form is cleared.

Exceptions:

Actor Action	System Response
1. Request Name is blank	2. Display error message: "Please fill out this field".
3. Problem description is blank	4. Display error message: "Please fill in request problem."
5. Symptom is blank	6. Display error message: "Please select symptom."

Relationships:

Login use case, Manage Repair/Maintenance Order use case.

Business Rules:

Technical manager can add new request to [Request List].

When technical manager clicks on [ADD REQUEST] button, [Add Request- Product Problem] form will be shown on pop up.

After technical manager clicks on [ADD] button to finish, [Add Request- Product Problem] form will be disappeared. New request is added and display on [Request List] table.

2.2.26. < Technical Manager > [Manage repair/maintenance order] View All Request List

2.2.26.1. Use Case Diagram

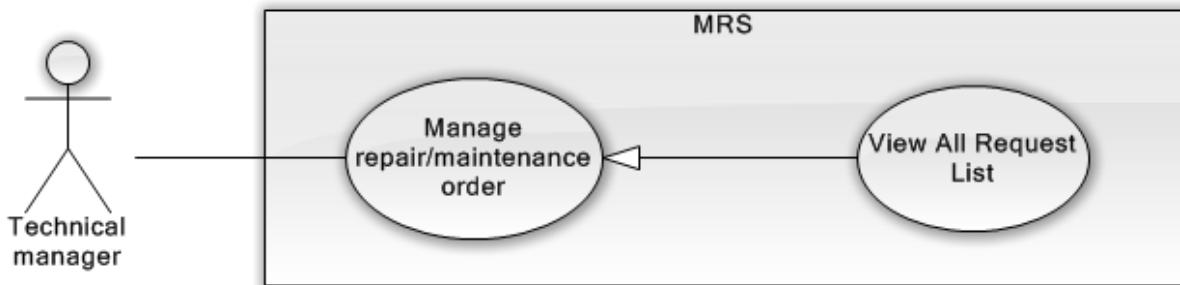


Figure 36-Manager Create Request List usecase diagram

2.2.26.2. Use Case Specification

USE CASE – View All Request List SPECIFICATION			
Use case No.	UC026	Use case version	2.0
Use case name	View all request list		
Author	HungDD		
Date	20/06/2013	Priority	Normal
Actor:	Technical Manager		
Summary:	Technical manager creates a request for customer to maintain/repair a product. With this function, Technical manager can view all requests that created by themselves. They can track a process of request and some details.		
Goal:	Technical manager would only view the requests that they create. The system shows all that requests on [Request List] table and sort in descending by time.		

Triggers:

Technical manager clicks [REQUEST] box on [Home] page, [Manage Repair Maintenance] Page will be displayed with all requests are shown on [Request List] table

Preconditions:

User logged successfully with role [Technical Manager]

There must be at least 1 request from customer

[Manage Repair Maintenance] page is displayed without error.

Post Conditions:

- Success: Display [Request List] table contains all requests by which logged Technical manager creates.
- Fail: N/A

Main Success Scenario:

Actor Action	System Response
1. Technical manager clicks on [Request] box on [Home] page.	<p>2. Display [Manage Repair Maintenance] page with [Request List] table with columns:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Code: Request code - Request Name: Name of request - Status: Current status of that request - Date: The date that request started. - Created Date: Date of request was created - Estimate Finish Date: Date of estimate request will finish. - Active

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Login use case, Manage Repair/Maintenance Order use case.

Business Rules:

Technical Manager can choose the maximum item on list paging is 10, 15, 25, etc.

If the number of requests is less than 10 records, the list will not show paging.

If there are no request created, show message "There are no request from customer is created."

Technical manager would only view the requests that they create

2.2.27. < Technical Manager > [Manage repair/maintenance order] Filter Request

2.2.27.1. Use Case Diagram

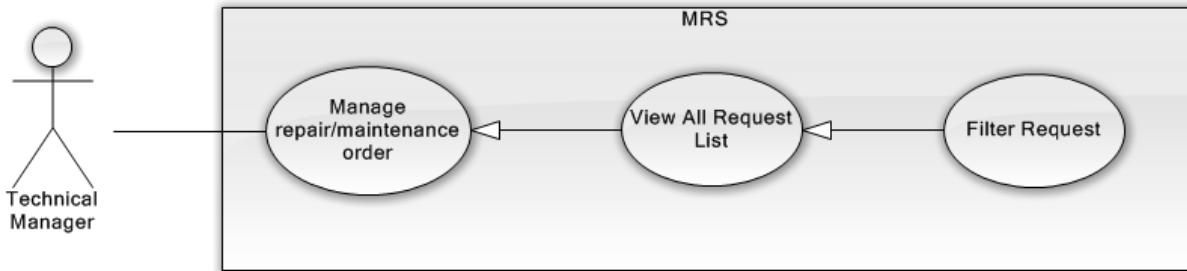


Figure 37-Manager Filter Request usecase diagram

2.2.27.2. Use Case Specification

USE CASE – Filter Request by Status SPECIFICATION			
Use case No.	UC027	Use case version	2.0
Use case name	Filter Request By Status		
Author	HungDD		
Date	20/06/2013	Priority	Normal
Actor:	Technical Manager		
Summary:	Technical manager can have more requests to manage process. Therefore, this function can help them to manage by filter request. There have two approaches:		
	<ul style="list-style-type: none"> + Filter by Name: filter according request name. + Filter by Status: filter all requests belong to status. + Filter by Code: filter by code of request. + Filter by Type: filter by model that this request belongs. 		
Goal:	The result of filtered would be shown in the [Request List] table.		
Triggers:	Technical manager selects type to search. Then input value that they want to filter. The result of filtered is shown in [Request List] below.		
Preconditions:	User logged successfully with role [Technical Manager] There must be at least 1 request from customer [Manage Repair Maintenance] page is displayed without error.		
Post Conditions:	- Success: The result of filtered would be shown in the [Request List] table.		

- Fail: N/A

Main Success Scenario:

Actor Action	System Response
1. Technical manager clicks on [Request] box on [Home] page.	2. Display [Manage Repair Maintenance] page with [Request List] table
Select type of filter on [Request List] table	Display all requests have filtered in [Request List] table

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Login use case, Manage Repair/Maintenance Order use case, View all Request List use case.

Business Rules:

When [Manage Repair Maintenance] page is loaded, default of filter is “Name”, corresponding to Filter by Name request.

If there is no a request was found, show message “There are no request found.”

2.2.28. < Technical Manager > [Manage repair/maintenance order] View Request Detail

2.2.28.1. Use Case Diagram

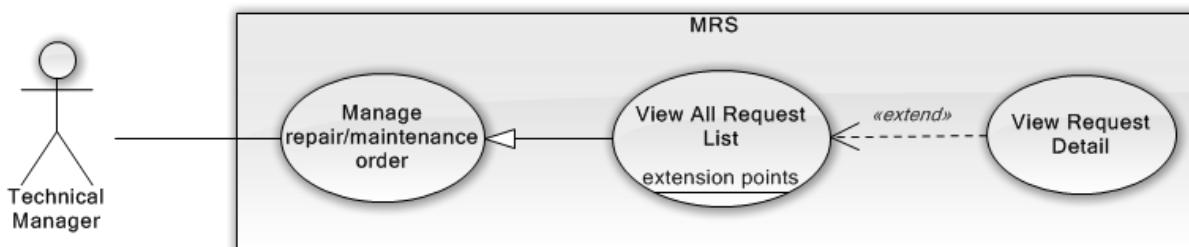


Figure 38-Manger View Request Detail usecase diagram

2.2.28.2. Use Case Specification

USE CASE – View Request Detail SPECIFICATION

Use case No.	UC028	Use case version	2.0
Use case name	View Request Detail		

Author	HungDD		
Date	20/06/2013	Priority	Normal

Actor:

Technical Manager

Summary:

With View Request Detail function, Technical manager wants to view detail information about one request for management. They just select and clicks on a request in [Request List] table, the screen shows detail of specific request on pop up.

Goal:

Request's detail of customer is shown in [Request Detail] form whenever technical manager click on a row of table [Request List].

Triggers:

In the [Request List] table, click on a row of request, detail information would be shown in [Request Detail] form on pop up.

Preconditions:

User logged successfully with role [Technical Manager]

There must be at least 1 request from customer to view detail.

[Manage Repair Maintenance] page is displayed without error.

Post Conditions:

- Success: Detail information of selected request will be shown in [Task Detail] form on pop up
- Fail: N/A

Main Success Scenario:

Actor Action	System Response							
1. Technical manager clicks on [Request] box on [Home] page.	2. Display [Manage Repair Maintenance] page with [Request List] table							
3. Click on a row of request in [Request List] table	<p>4. Display form [Request Detail] have specify form includes:</p> <table border="1"> <tr> <td>Basic information</td> <td> <ul style="list-style-type: none"> - Request Name: textbox (max length: 100) - Request type: dropdown list - Estimate Finish: Calendar </td> </tr> <tr> <td>Customer Related</td> <td> <ul style="list-style-type: none"> - Customer relate: dropdown list(customer request to maintain-repair) </td> </tr> <tr> <td>Product information</td> <td> <ul style="list-style-type: none"> - Prototype: dropdown list </td> </tr> </table>	Basic information	<ul style="list-style-type: none"> - Request Name: textbox (max length: 100) - Request type: dropdown list - Estimate Finish: Calendar 	Customer Related	<ul style="list-style-type: none"> - Customer relate: dropdown list(customer request to maintain-repair) 	Product information	<ul style="list-style-type: none"> - Prototype: dropdown list 	
Basic information	<ul style="list-style-type: none"> - Request Name: textbox (max length: 100) - Request type: dropdown list - Estimate Finish: Calendar 							
Customer Related	<ul style="list-style-type: none"> - Customer relate: dropdown list(customer request to maintain-repair) 							
Product information	<ul style="list-style-type: none"> - Prototype: dropdown list 							

		<ul style="list-style-type: none"> - Model: dropdown list 	
	Estimate replace equipment	<ul style="list-style-type: none"> - Add equipment button. Click on this button then appear: - Category: dropdown list - Equipment: dropdown list - [quantity]: dropdown list 	
	Request information	<ul style="list-style-type: none"> - Symptom: select symptom of request - Problem description: text area - Customer confirmation: text area - Request tags: request from customer to decided tags appropriate for assign task. 	
	Button	<ul style="list-style-type: none"> - CLOSE: button - CONFIRM: button - CANCEL REQUEST: button - CREATE TASK: button - UPDATE SOLUTION: button 	

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Login use case, Manage Repair/Maintenance Order use case

Business Rules:

Technical manager selects request to view detail, display specific [Request Detail] form.

In case of status of request is “Pending”, system display [CLOSE], [CONFIRM], [CANCEL REQUEST] button.

In case of status of request is “Confirmed”, system display [CLOSE], [CANCEL REQUEST], [CREATE TASK] button

In case of status of request is “In Progress”, system display [CLOSE], [CANCEL REQUEST] button.

In case of status of request is “Finished”, system display [CLOSE], [UPDATE SOLUTION] button.

2.2.29. < Technical Manager > [Manage repair/maintenance order] Update Request

2.2.29.1. Use Case Diagram

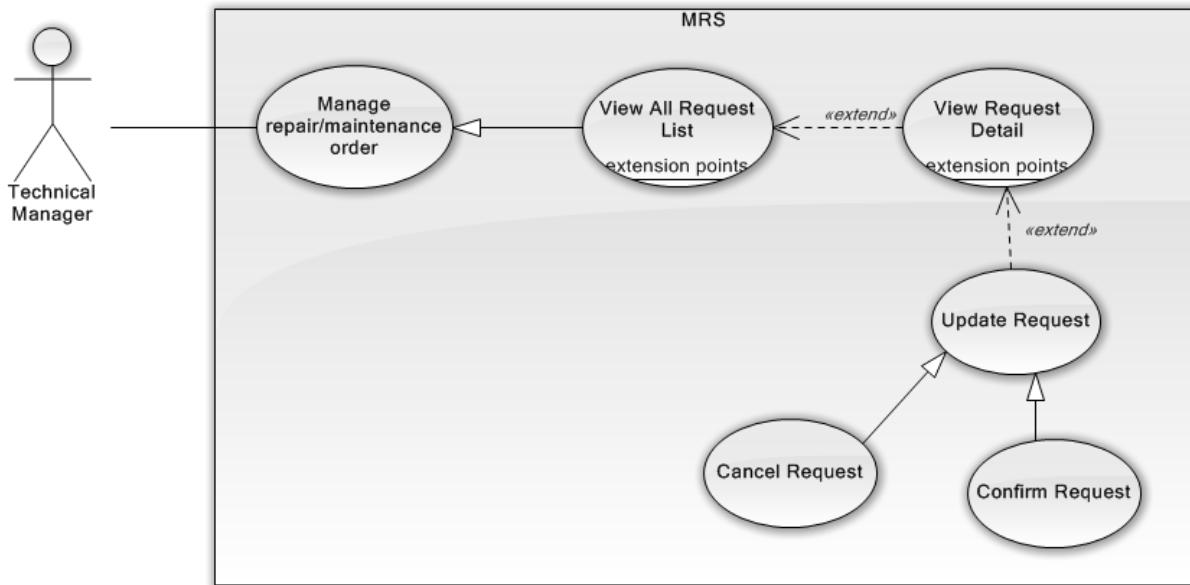


Figure 39-Manger Update Request usecase diagram

2.2.29.2. Use Case Specification

USE CASE – Update Request SPECIFICATION			
Use case No.	UC029	Use case version	2.0
Use case name	Update Request		
Author	HungDD		
Date	20/06/2013	Priority	Normal
Actor:	Technical Manager		
Summary:	With Update Request function, technical manager can update the detail of request. This function is the positive support for technical manager to manage the process of maintenance/repairing. Technical manager can update customer confirm or cancel request when customer need. At this time, the system also pushes a notification to customer.		
Goal:	Detail of updated request is added to [Request List] table and saved to database The system notifies to customer a change.		

Triggers:

Technical manager clicks on a row of [Request List] table, detail of selected request will be shown with fields that can be edited. They fill details that want to update and choice to click on button to finish.

Preconditions:

User logged successfully with role [Technical Manager]

There must be at least 1 request from customer to update details.

[Manage Repair Maintenance] page is displayed without error.

Detail of request is loaded successfully.

Post Conditions:

- Success: Detail of updated request is added to [Request List] table, saved to database and notify customer.
- Fail: Detail of updated request is not added to the [Request List] and not saved to the database. Display error message on [Request Detail] form.

Main Success Scenario:

Actor Action	System Response	
1. Technical manager clicks on [Request] box on [Home] page.	2. Display [Manage Repair Maintenance] page with [Request List] table	
3. Click on a row of request in [Request List] table	4. Display form [Request Detail] have specify form includes:	
	Basic information	<ul style="list-style-type: none"> - Request Name: textbox (max length: 100) - Request type: dropdown list - Estimate Finish: Calendar
	Customer Related	<ul style="list-style-type: none"> - Customer relate: dropdown list(customer request to maintain-repair)
	Product information	<ul style="list-style-type: none"> - Prototype: dropdown list - Model: dropdown list
	Estimate replace equipment	<ul style="list-style-type: none"> - Add equipment button. Click on this button then appear: - Category: dropdown list - Equipment: dropdown list

		<ul style="list-style-type: none"> - [quantity]: dropdown list 	
	Request information	<ul style="list-style-type: none"> - Symptom: select symptom of request - Problem description: text area - Customer confirmation: text area - Request tags: request from customer to decided tags appropriate for assign task. 	
	Button	<ul style="list-style-type: none"> - CLOSE: button - CONFIRM: button - CANCEL REQUEST: button - CREATE TASK: button - UPDATE SOLUTION: button 	
7. Fill data in the item which want to update and choice to click on button to finish		8. Update data to the database.	

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Login use case, Manage Repair/Maintenance Order use case, View Request Detail use case.

Business Rules:

Detail of updated request is added to [Request List] table and saved to database

In case of status of request is “Pending”, system display [CLOSE], [CONFIRM] and [CANCEL REQUEST] button. In this case, click on [CONFIRM] button to change status to “Confirmed” or [CANCEL REQUEST] button to cancel this request.

In case of status of request is “Confirmed”, system display [CLOSE], [CANCEL REQUEST] and [CREATE TASK] button. In this case, click on [CREATE TASK] button to assign task for staff or [CANCEL REQUEST] button to cancel this request.

In case of status of request is “In Progress”, system display [CLOSE] and [CANCEL REQUEST] button. In this case, click on [CANCEL REQUEST] button to cancel this request.

In case of status of request is “Finished”, system display [CLOSE] and [UPDATE SOLUTION] button. In this case, click on [UPDATE SOLUTION] button to update solution

for damage symptom of request.

2.2.30. < Technical Manager > [Manage Task] Create Task

2.2.30.1. Use Case Diagram

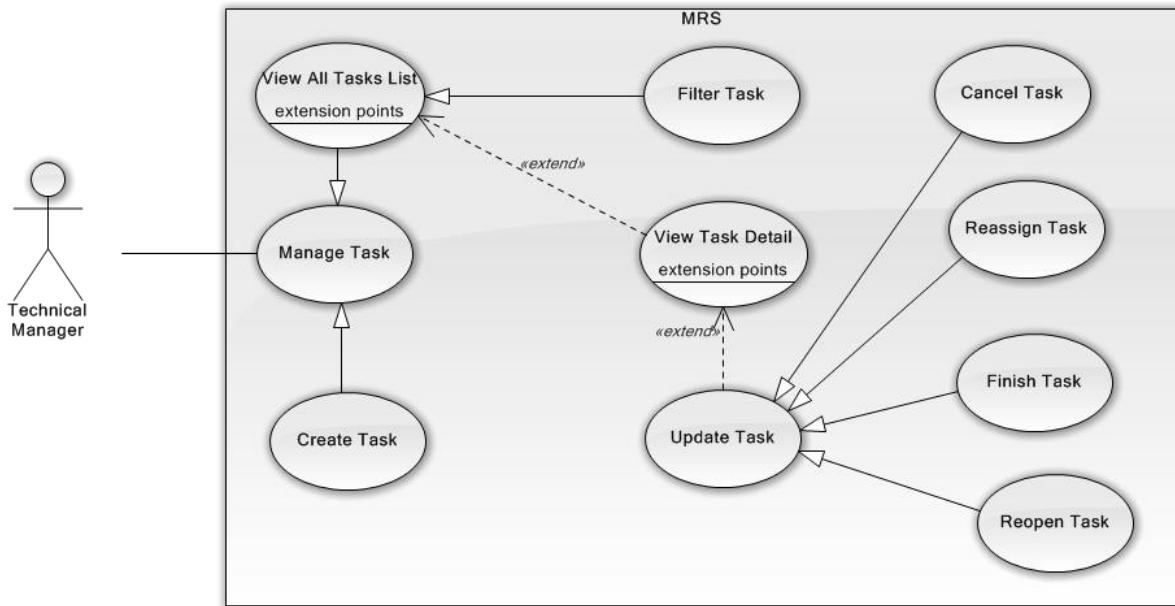


Figure 40-Manager Manage Task use case diagram

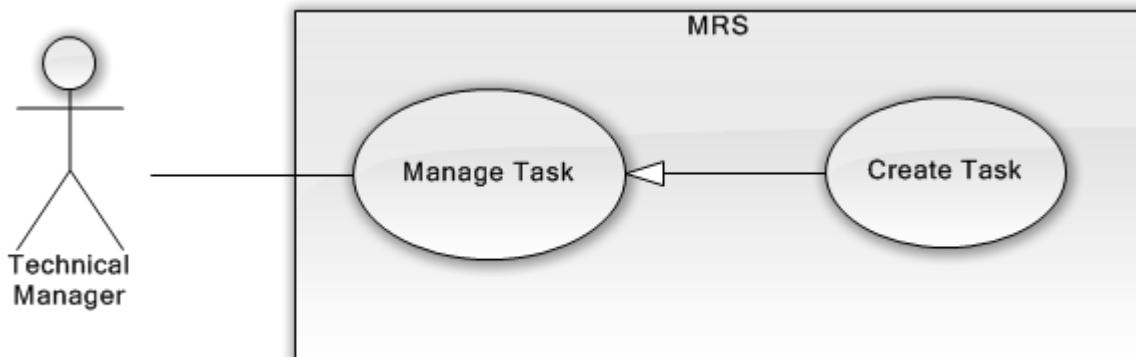


Figure 41-Manger Create Task usecase diagram

2.2.30.2. Use Case Specification

USE CASE – Create Task SPECIFICATION

Use case No.	UC030	Use case version	2.0
Use case name	Create Task		

Author	HungDD		
Date	21/06/2013	Priority	Normal

Actor:

Technical Manager

Summary:

When Technical manager create a request for customer to maintain/repair. Technical manager want to break down that requests to small tasks and assign to appropriate staffs. With Create Task function, technical manager can create a new task for staff as they expect. At this time, the system also pushes a notification to staff.

Goal:

New task of related request is added to [Task List] and saved to the database

The system notifies about created task to appropriate staff

Triggers:

In [Manage Task] Page, click [ADD TASK] button to appear [Add New Task] form. Fill the specify form and click [ADD] button to finish.

Preconditions:

User logged successfully with role [Technical Manager]

[Manage Task] Page is displayed without error.

Post Conditions:

Success: New task of related request is added to task list.

Fail: New task of related request is not added to task list. Display error message on [Add New Task] form. Entered information is still in form.

Main Success Scenario:

Actor Action	System Response							
1. Technical manager clicks on [Request] box on [Home] page.	2. Display [Manage Repair Maintenance] Page							
3. Clicks on [TASK] tab on [Manage Repair Maintenance] page.	4. Display [Manage Task] page with [Task List] table							
5. Click [ADD TASK] button on [Task List] table	6. Display form [Add New Task] have specify form includes:							
	<table border="1"> <tr> <td>Request Related</td> <td> <ul style="list-style-type: none"> - Request: Dropdown list - Request tags - Problem: Text area </td> </tr> <tr> <td>Staff Related</td> <td> <ul style="list-style-type: none"> - Staff Related: Dropdown list - Recommend Staff: button </td> </tr> <tr> <td>Task</td> <td> <ul style="list-style-type: none"> - Name: text box </td> </tr> </table>	Request Related	<ul style="list-style-type: none"> - Request: Dropdown list - Request tags - Problem: Text area 	Staff Related	<ul style="list-style-type: none"> - Staff Related: Dropdown list - Recommend Staff: button 	Task	<ul style="list-style-type: none"> - Name: text box 	
Request Related	<ul style="list-style-type: none"> - Request: Dropdown list - Request tags - Problem: Text area 							
Staff Related	<ul style="list-style-type: none"> - Staff Related: Dropdown list - Recommend Staff: button 							
Task	<ul style="list-style-type: none"> - Name: text box 							

	information	- Start: Calendar - Finish: Calendar - Description: Text area	
	Button	- CLOSE: button [Alternative] - ADD: button	
7. Fill data in specify form	8. [Exception 1, 3]		
9. Click on [ADD] button to finish.	10. Save all data and add new task to the task list.		

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish.	2. Data in specify form is cleared.

Exceptions:

Actor Action	System Response
1. Task Name is blank	2. Display error message: "Please fill out this field".
3. Description is blank	4. Display error message: "Please fill out task description"

Relationships:

Login use case, Manage Task use case, View Request Detail.

Business Rules:

Technical manager can add new task of a related request to [Task List].

When technical manager clicks on [ADD TASK] button, [Add New Task] form will be show on pop up.

2.2.31. < Technical Manager > [Manage Task] View All Tasks List

2.2.31.1. Use Case Diagram

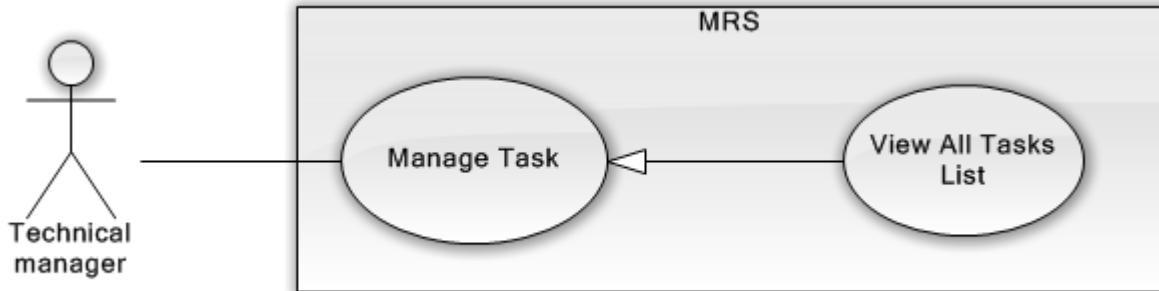


Figure 42-Manger View All Task List usecase diagram

2.2.31.2. Use Case Specification

USE CASE – View All Tasks List SPECIFICATION			
Use case No.	UC031	Use case version	2.0
Use case name	View all Tasks list		
Author	HungDD		
Date	21/06/2013	Priority	Normal
Actor: Technical Manager			
Summary: When Technical Manager creates a request for customer to maintain/repair. They want to break down that requests to small tasks and assign to appropriate staffs. With this function, Technical manager could track all that tasks on [Task List] table. It can help them to manage process of maintenance/repairing.			
Goal: The system shows all tasks of staff on list that are sorted in descending by time.			
Triggers: In [Manage Task] page will be displayed with all task are shown on [Task List] table.			
Preconditions: User logged successfully with role [Technical Manager] There must be at least 1 task of staff. [Manage Task] page is displayed without error.			
Post Conditions: <ul style="list-style-type: none">- Success: Display [Task List] table contains all tasks with task information about: task name, status, Start Date, Finish Date, Assigned to.- Fail: N/A			
Main Success Scenario:			

Actor Action	System Response
1. Technical manager clicks on [Request] box on [Home] page.	2. Display [Manage Repair Maintenance] Page
3. Clicks on [TASK] tab on [Manage Repair Maintenance] page.	4. Display [Manage Task] page with [Task List] table with columns: <ul style="list-style-type: none"> - No.: Ordinal number - Task name: Name of task - Status: Current status of that task - Start date: The date which task started. - Finish date: The date which task finished. - Assigned to: Person in charge

Alternative Scenario:
N/A

Exceptions:
N/A

Relationships:
Login use case, Manage Task use case.

Business Rules:
Technical manager can view all tasks list belong to [Request List] that they manage.
Technical manager can choose the maximum item on list paging is 10, 15, 25, etc.
If the number of task is less than 10 records, the page will not show paging.
If there are no task created, show message “There are no task is created.”

2.2.32. < Technical Manager > [Manage Task] Filter Task

2.2.32.1. Use Case Diagram

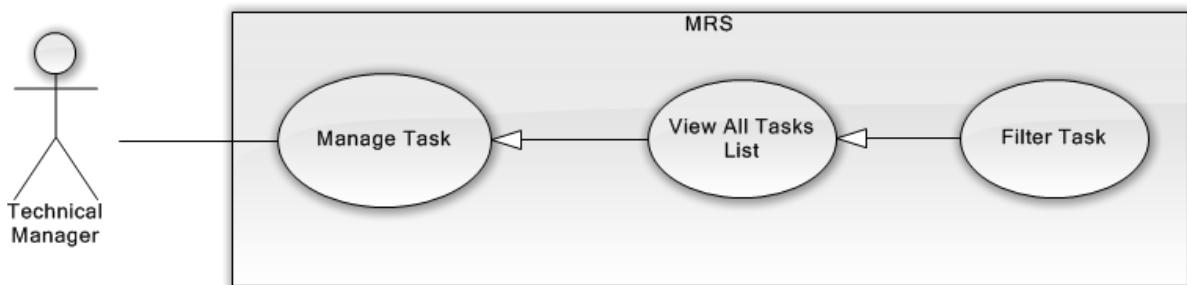


Figure 43-Manger Filter Task usecase diagram

2.2.32.2. Use Case Specification

USE CASE – Filter Task by Status SPECIFICATION										
Use case No.	UC032	Use case version	2.0							
Use case name	Filter Task By Status									
Author	HungDD									
Date	21/06/2013	Priority	Normal							
Actor: Technical Manager										
Summary: Technical manager can manage more tasks of staff. With this function, they can manage more easily by filter task. There have 3 approaches: + Filter by Name: filter according task name. + Filter by Status: filter all tasks belong to status. + Filter by Request: filter by Request name to show task belongs										
Goal: The result of filtered would be shown in the [Task List] table.										
Triggers: Technical manager selects filter type. Then input value that want to filter. The result is shown in [Task List] below.										
Preconditions: User logged successfully with role [Technical Manager] There must be at least 1 task of related request. [Manage Task] page is displayed without error.										
Post Conditions: - Success: The result of selected status is shown in [Task List]. - Fail: N/A										
Main Success Scenario:										
<table border="1"> <thead> <tr> <th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1. Technical manager clicks on [Request] box on [Home] page.</td><td>2. Display [Manage Repair Maintenance] Page</td></tr> <tr> <td>3. Clicks on [TASK] tab on [Manage Repair Maintenance] page.</td><td>4. Display [Manage Task] page with [Task List] table</td></tr> <tr> <td>5. Select filter type, input value and click on [Filter] button.</td><td>6. Display result filtered in [Task List] table.</td></tr> </tbody> </table>			Actor Action	System Response	1. Technical manager clicks on [Request] box on [Home] page.	2. Display [Manage Repair Maintenance] Page	3. Clicks on [TASK] tab on [Manage Repair Maintenance] page.	4. Display [Manage Task] page with [Task List] table	5. Select filter type, input value and click on [Filter] button.	6. Display result filtered in [Task List] table.
Actor Action	System Response									
1. Technical manager clicks on [Request] box on [Home] page.	2. Display [Manage Repair Maintenance] Page									
3. Clicks on [TASK] tab on [Manage Repair Maintenance] page.	4. Display [Manage Task] page with [Task List] table									
5. Select filter type, input value and click on [Filter] button.	6. Display result filtered in [Task List] table.									
Alternative Scenario:										

N/A
Exceptions:
N/A
Relationships:
Login use case, Manage Task use case, View all Task List use case.
Business Rules:
When [Manage Task] page is loaded, default type of filter is “Name”, corresponding to Filter by Task name.
If there is no task was found, show message “There are no task found.”

2.2.33. < Technical Manager > [Manage Task] View Task Detail

2.2.33.1. Use Case Diagram

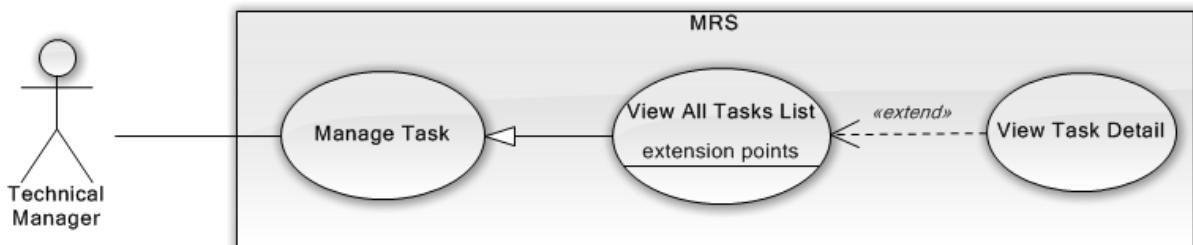


Figure 44-Manger View Task Detail usecase diagram

2.2.33.2. Use Case Specification

USE CASE – View task detail SPECIFICATION			
Use case No.	UC033	Use case version	2.0
Use case name	View Task Detail		
Author	HungDD		
Date	21/06/2013	Priority	Normal
Actor:	Technical Manager		
Summary:	With View Task Detail function, Technical manager wants to view detail information about one task for management. They just selects and clicks on a task in [Task List] table, the screen shows detail of specific task on pop up.		
Goal:	Task's detail of selected task is shown in [Task Detail] form whenever technical manager click on a row of [Task List] table.		
Triggers:	In the [Task List] table, click on a row of task, detail information is shown in [Task Detail] form on pop up.		

Preconditions:

User logged successfully with role [Technical Manager]
 There must be at least 1 task of related request to view detail.
 [Manage Task] page is displayed without error.

Post Conditions:

- Success: Detail information of selected task will be shown in [Task Detail] form on pop up
- Fail: N/A

Main Success Scenario:

Actor Action	System Response	
1. Technical manager clicks on [Request] box on [Home] page.	2. Display [Manage Repair Maintenance] Page	
3. Clicks on [TASK] tab on [Manage Repair Maintenance] page.	4. Display [Manage Task] page with [Task List] table	
5. Click on a row of task in [Task List] table	6. Display form [Task Detail] have specify form includes:	
	Request Related	<ul style="list-style-type: none"> - Request: Label - Problem: Label
	Staff Related	<ul style="list-style-type: none"> - Staff: Dropdown list - Recommend Staff: button
	Task information	<ul style="list-style-type: none"> - Name: text box - Start: Calendar - Finish: Calendar - Status: Label - Spend hours: Dropdown list - Description: Text area - More issues: Text area
	Button	<ul style="list-style-type: none"> - CLOSE: button - UPDATE: button - FINISH: button - RE-OPEN: button

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Login use case, Manage Task use case

Business Rules:

Technical manager selects task to view detail, display specific [Task Detail] form.

In case of status of task is “New” or “In Progress”, system will display [CLOSE], [UPDATE] and [FINISH] button.

In case of status of task is “Finished”, system will display [CLOSE] and [RE-OPEN] button.

Background color of row of table belongs to status of task. It make easy to cover process of task.

2.2.34. < Technical Manager > [Manage Task] Update Task

2.2.34.1. Use Case Diagram

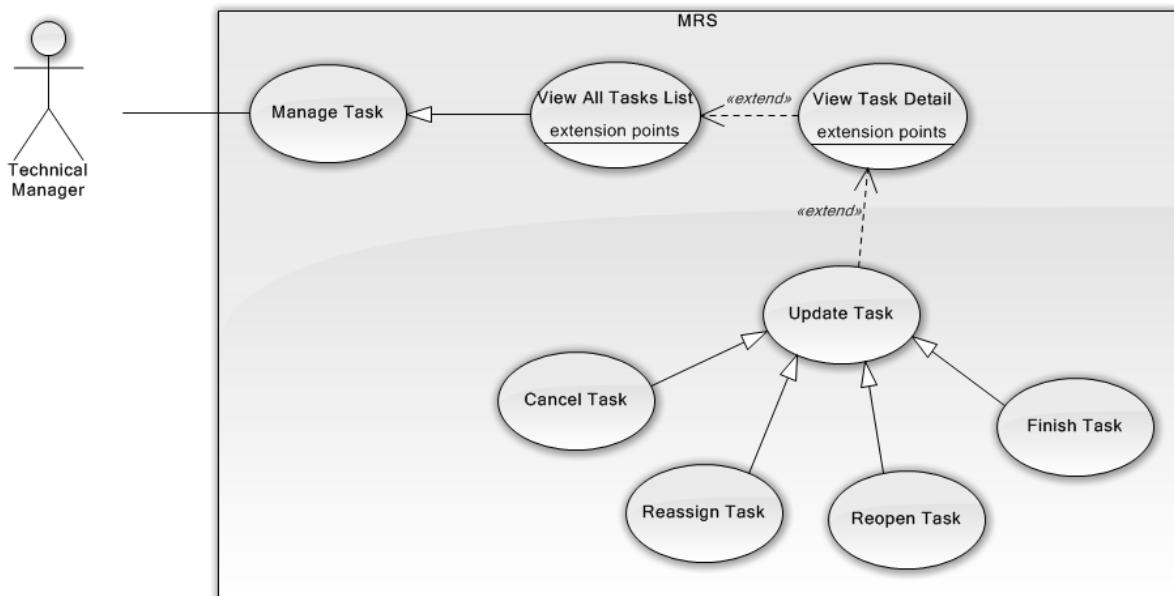


Figure 45-Manger Update Task usecase diagram

2.2.34.2. Use Case Specification

USE CASE – Update Task SPECIFICATION

Use case No.	UC034	Use case version	2.0
Use case name	Update Task		
Author	HungDD		
Date	21/06/2013	Priority	Normal
Actor:	Technical Manager		

Summary:

With Update Task function, technical manager can update the detail of task. In this process, there is could have more issues. This function helps technical manager record them. At this time, the system pushes a notification to staff about their task.

Goal:

Detail of updated task is saved to database.

The system notifies to staff a change.

Triggers:

Technical manager clicks on a row of [Task List] table and detail of selected task will be shown with fields that can be edited. They fill detail and choice to click on button to finish.

Preconditions:

User logged successfully with role [Technical Manager]

There must be at least 1 task of related request to update details.

[Manage Task] page is displayed without error.

Detail of task is loaded successfully.

Post Conditions:

- Success: Detail of updated task is saved to database.

- Fail: Detail of updated task is not saved to database. Display error message.

Main Success Scenario:

Actor Action	System Response							
1. Technical manager clicks on [Request] box on [Home] page.	2. Display [Manage Repair Maintenance] Page							
3. Clicks on [TASK] tab on [Manage Repair Maintenance] page.	4. Display [Manage Task] page with [Task List] table							
5. Click on a row of task in [Task List] table	<p>6. Display form [Task Detail] have specify form includes:</p> <table border="1"> <tr> <td>Request Related</td><td> <ul style="list-style-type: none"> - Request: Label - Problem: Label </td></tr> <tr> <td>Staff Related</td><td> <ul style="list-style-type: none"> - Staff: Dropdown list - Recommend Staff: button </td></tr> <tr> <td>Task information</td><td> <ul style="list-style-type: none"> - Name: text box - Start: Calendar - Finish: Calendar - Status: Label - Spend hours: Dropdown list - Description: Text area - More issues: Text area </td></tr> </table>	Request Related	<ul style="list-style-type: none"> - Request: Label - Problem: Label 	Staff Related	<ul style="list-style-type: none"> - Staff: Dropdown list - Recommend Staff: button 	Task information	<ul style="list-style-type: none"> - Name: text box - Start: Calendar - Finish: Calendar - Status: Label - Spend hours: Dropdown list - Description: Text area - More issues: Text area 	
Request Related	<ul style="list-style-type: none"> - Request: Label - Problem: Label 							
Staff Related	<ul style="list-style-type: none"> - Staff: Dropdown list - Recommend Staff: button 							
Task information	<ul style="list-style-type: none"> - Name: text box - Start: Calendar - Finish: Calendar - Status: Label - Spend hours: Dropdown list - Description: Text area - More issues: Text area 							

	Button	<ul style="list-style-type: none"> - CLOSE: button - UPDATE: button - FINISH: button - RE-OPEN: button 	
7. Fill data in the item which want to update	8. [Exception 1]		
9. Click on button to finish.	10. Update data to the database.		

Alternative Scenario:

N/A

Exceptions:

Actor Action	System Response
1. More issues is blank	2. Display error message: “Please fill out this field”.

Relationships:

Manage Task use case.

Business Rules:

Detail of task is updated to the database.

In case of status of task is “New” or “In Progress”, system will display [CLOSE], [UPDATE] and [FINISH] button. In this case, technical manager could click on [UPDATE] button to update information about this task or [FINISH] to cancel task and change status of task to “Finished”.

In case of status of task is “Finished”, system will display [CLOSE] and [RE-OPEN] button. In this case, technical manager could click on [RE-OPEN] to open again due date for this task.

2.2.35. < Technical Manager > [Manage Symptom] Create Symptom

2.2.35.1. Use Case Diagram

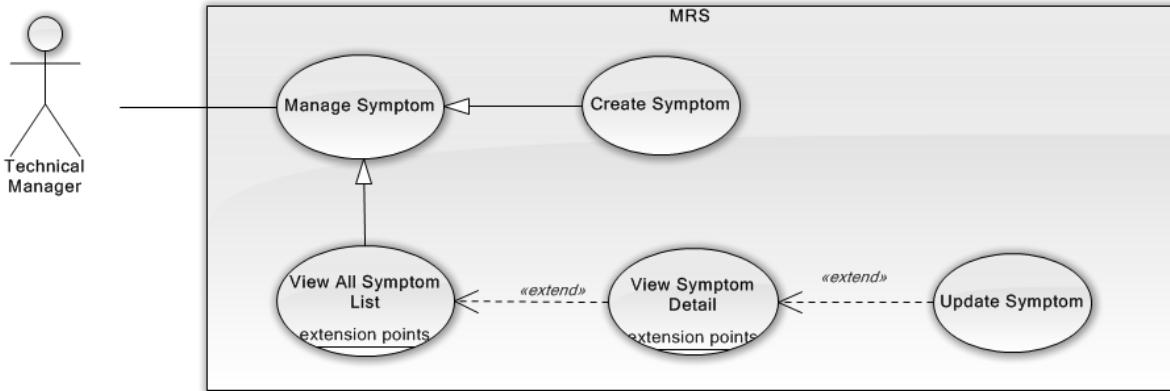


Figure 46-Manager Manage Symptom use case diagram

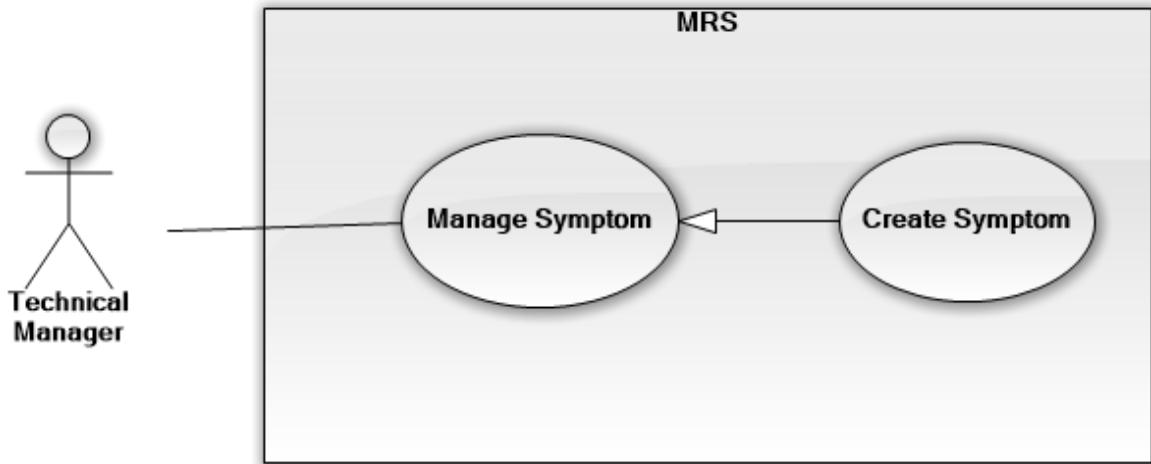


Figure 47-Manger Create Symptom usecase digram

2.2.35.2. Use Case Specification

USE CASE – Create Symptom SPECIFICATION			
Use case No.	UC035	Use case version	2.0
Use case name	Create Symptom		
Author	HungDD		
Date	21/5/2013	Priority	Normal
Actor: Technical Manager			
Summary: Technical manager uses this use case to create a new symptom for repaired of maintained products. When create a new symptom successfully, it will store in database and Technical			

Manager can view symptoms later.

Goal:

Store a new symptom of repaired or maintained products of customers to database for later use.

Triggers:

In [Manage Symptom Page], Technical manager can create new symptom of product by clicks on [ADD SYMPTOM] button, then [Add Symptom] form with necessary information will be shown on pop up.

Preconditions:

User logged in successfully with role of Technical Manager.

Post Conditions:

- **Success:** New symptom will be stored in database and new row of symptom will be inserted to [Symptom List] table.
- **Fail:** Redirect to [Error Page] or still at current page with error messages.

Main Success Scenario:

Actor Action	System Response
1. Technical manager clicks on [Request] box on [Home] page.	2. Display [Manage Repair Maintenance] Page
3. Technical Manager clicks on [SYMPTOM] tab in [Manage Repair Maintenance] Page	4. Redirect to [Manage Symptom Page] with table [Symptom List]
5. Technical Manager clicks on [Add Symptom] button.	<p>6. Show [Add Symptom] form includes:</p> <ul style="list-style-type: none"> - Symptom Name: textbox (required, max length: 150) - Product Category: dropdown list - Brand: dropdown list - Model: dropdown list (All prototype name of products) - Solution: text area - ADD: button - CLOSE: button [Alternative 1] - SUBMIT: button
7. Enter information into text fields.	8. [Exception 1]
9. Click on [ADD] button to finish this use case.	10. New symptom belongs to selected model and solutions are stored in database and new row is inserted to [Symptom List] table with added symptom information.

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish this use case.	2. All information in [Add Symptom] form is cleared.

Exceptions:

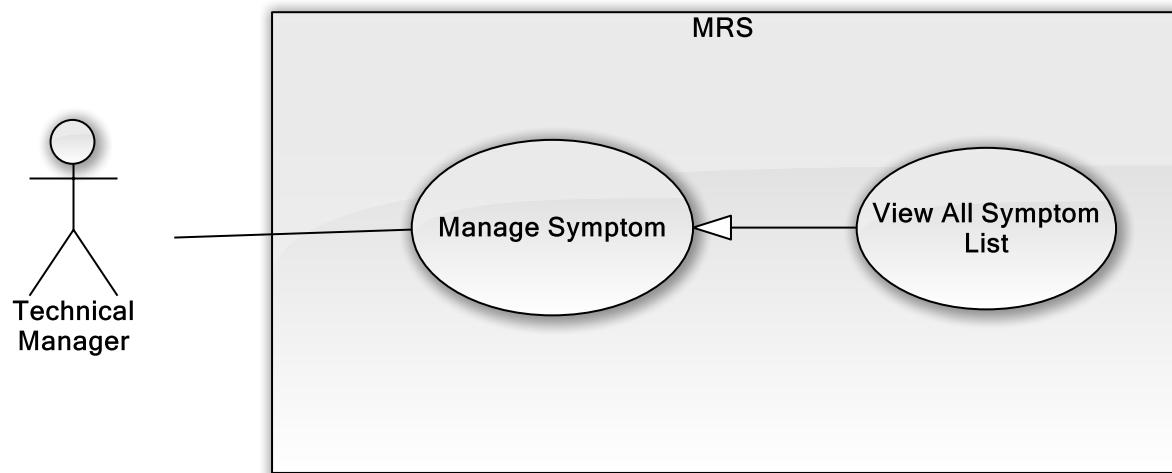
Actor Action	System Response
1. [Symptom Name] textbox, [Solution] text area is blank	2. Display message: "Please fill out this field." on the fields that is blank.

Relationships:

Manage Symptom

Business Rules:

- Technical Manager wants to store a new symptom of repaired or maintained products of customers to database for later use.
- Technical Manager fills necessary information about new symptom into [Add Symptom] form and submits.
- New symptom is stored in database and new row is inserted to [Symptom List] table.

2.2.36. < Technical Manager > [Manage Symptom] View All Symptom List**2.2.36.1. Use Case Diagram****Figure 48-Manger View All Symptom List usecase diagram****2.2.36.2. Use Case Specification****USE CASE – View All Symptom List SPECIFICATION**

Use case No.	UC036	Use case version	2.0
Use case name	View All Symptom List		
Author	ThanhTM		
Date	21/5/2013	Priority	Normal

Actor:

Technical Manager

Summary:

Technical manager uses this case to tracking all symptoms of products in the past. These symptoms are stored in database when staff repaired or maintained products of customers and then informed to technical manager logged these symptoms for future purposes.

Goal:

Technical manager can view all symptoms in list with table format, which symptoms belong to which products and corresponding solutions.

Triggers:

Technical manager can view all symptoms list by clicks on [SYMPTOM] tab in [Manage Repair Maintenance Page]. Then the system display [Manage Symptom Page] with [Symptom List] table

Preconditions:

User logged in successfully with role of Technical Manager.

Post Conditions:

- **Success:** List all symptoms of products which had been repaired or maintained in the past.
- **Fail:** N/A

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [SYMPTOM] tab [Manage Repair Maintenance Page].	<p>4. Redirect to [Manage Symptom Page].</p> <p>List all symptoms of products which had been repaired or maintained in the past in [Symptom List] table with columns:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Symptom Name: Name of symptom - Product Prototype: Model which symptom belongs to - Activation: Disable or enable that

	symptom
<p>Alternative Scenario: N/A</p> <p>Exceptions: N/A</p> <p>Relationships: Manage Symptom.</p> <p>Business Rules:</p> <ul style="list-style-type: none"> - Technical manager wants to view all symptoms of products in the past. - Technical manager can choose the maximum item on list paging is 10, 15, 25, etc. - All symptoms will display in a list with order from latest to oldest updated date. - Background color of row of [Symptom List] table would change to describe for The symptom have solution or not. 	

2.2.37. < Technical Manager > [Manage Symptom] View Symptom Detail

2.2.37.1. Use Case Diagram

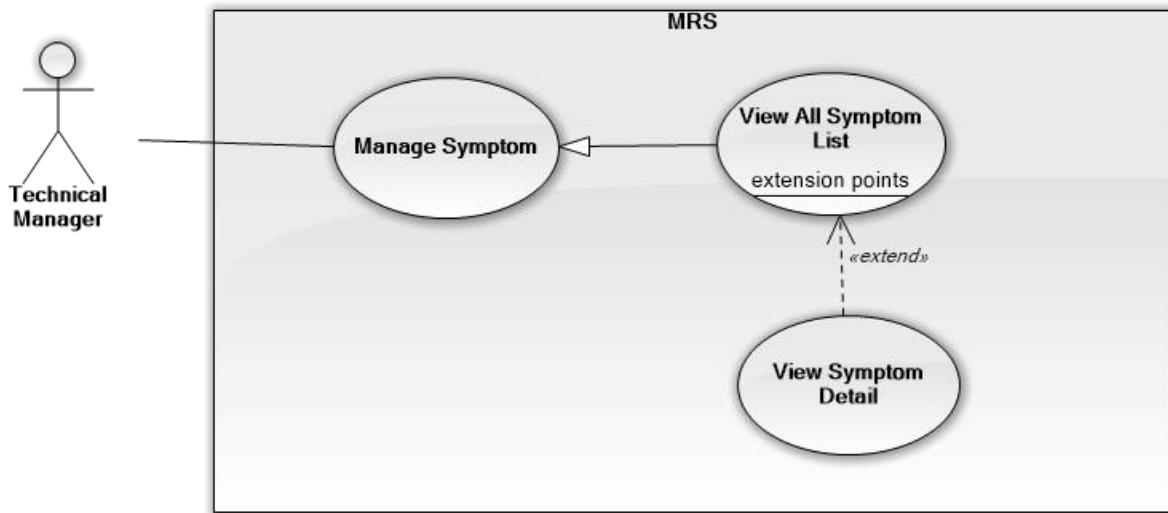


Figure 49-Manger View Symptom Detail usecase diagram

2.2.37.2. Use Case Specification

USE CASE – View Symptom Detail SPECIFICATION			
Use case No.	UC037	Use case version	2.0
Use case name	View Symptom Detail		
Author	ThanhTM		

Date	21/5/2013	Priority	Normal
Actor:			
Technical Manager			
Summary:			
Technical Manager wants to view detail information about one symptom, he (she) just selects a symptom in [Symptom List] table and views more detail about that symptom.			
Goal:			
Technical Manager has a clear detail about a symptom that they want to focus. Additional, this use case can extend to help Technical Manager update a symptom easily which will be described more detail in [Update Symptom] use case below.			
Triggers:			
In [Manage Symptom Page], Technical manager can view detail of a symptom by clicks on that symptom in [Symptom List] table.			
Preconditions:			
User logged in successfully with role of Technical Manager and Technical Manager clicks on a row of symptom in the body of [Symptom List] table (except table header row).			
Post Conditions:			
<ul style="list-style-type: none"> - Success: Detail information of clicked symptom will be shown in [Symptom Detail] form on pop up. - Fail: N/A 			
Main Success Scenario:			
Actor Action	System Response		
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].		
3. Technical Manager clicks on [SYMPTOM] tab in [Manage Repair Maintenance Page].	4. Redirect to [Symptom Management Page] with [Symptom List] table		
5. Technical Manager clicks on a row in body of [Symptom List] table to finish this use case.	6. Show [Symptom Detail] form with detail information of clicked symptom includes: <ul style="list-style-type: none"> - Symptom Name: textbox (max length: 150) - Product Relate: dropdown list (All prototype name of products) - Active: check box - Solution: text area - CLOSE: button - UPDATE: button 		

Alternative Scenario:

N/A

Exceptions:

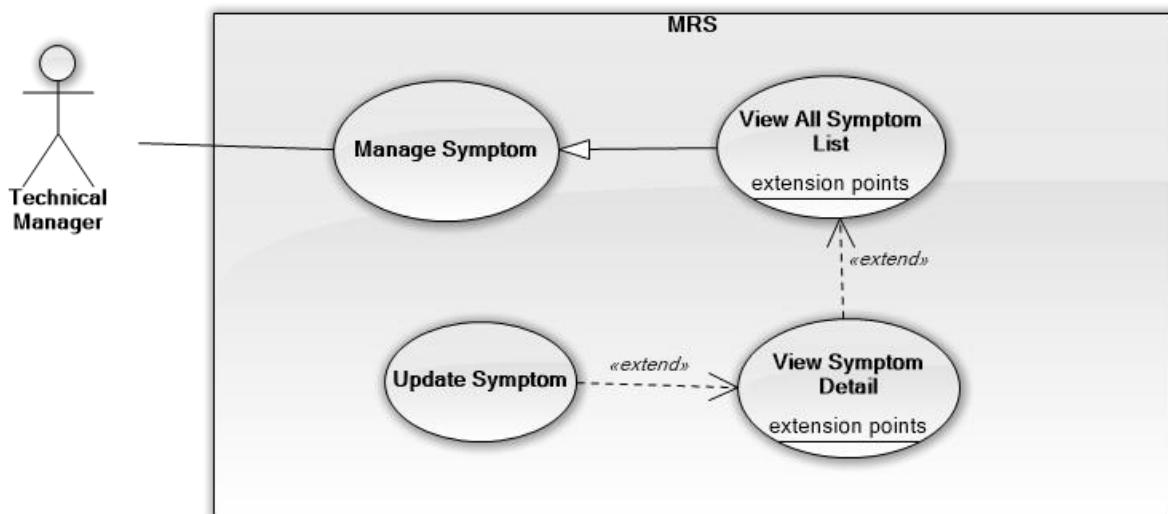
N/A

Relationships:

Manage Symptom

Business Rules:

- Technical Manager wants to view detail information of one symptom.
- Technical Manager clicks on a symptom row in [Symptom List] table and detail information of that symptom will be showed in [Symptom Detail] form on pop up.

2.2.38. < Technical Manager > [Manage Symptom] Update Symptom**2.2.38.1. Use Case Diagram****Figure 50-Manger Update Symptom usecase diagram****2.2.38.2. Use Case Specification****USE CASE – Update Symptom SPECIFICATION**

Use case No.	UC038	Use case version	2.0
Use case name	Update Symptom		
Author	ThanhTM		
Date	21/5/2013	Priority	Normal
Actor: Technical Manager Summary:			

For a reason, Technical Manager wants to update content of a symptom that existed; this use case can help him (her) to do it easily.

Goal:

Update existed symptom information in database such as solutions

Triggers:

In [Manage Symptom Page], Technical Manager can update a symptom by clicks on that symptom in [Symptom List] table. [Symptom Detail] form will be shown with detail information of that symptom and from now, Technical Manager can edit information for his purpose and stored edited information to database.

Preconditions:

User logged successfully with role of Technical Manager and Technical Manager clicks on a row of symptom in the body of [Symptom List] table that he (she) wants to update (except table header row).

Post Conditions:

- **Success:** Edited symptom information is updated to database and changes of that edited symptom are update in [Symptom List] table.
- **Fail:** Redirect to [Error Page] or still at current page with error messages.

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [SYMPTOM] tab in [Manage Repair Maintenance Page].	4. Redirect to [Symptom Management Page] with [Symptom List] table
5. Technical Manager clicks on a row in body of [Symptom List] table	<p>6. Show [Symptom Detail] form with detail information of clicked symptom includes:</p> <ul style="list-style-type: none"> - Symptom Name: textbox (max length: 150) - Product Relate: dropdown list (All prototype name of products) - Active: check box - Solution: text area - CLOSE: button - UPDATE: button
7. Technical Manager makes some changes for his purpose by edit text fields	8. [Exception 1]
9. Click on [UPDATE] button to finish	10. Edited symptom information is

this use case.	updated to database and changes of that edited symptom are update in [Symptom List] table.	
----------------	--	--

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish this use case.	2. Nothing changes and [Symptom Detail] form is hidden.

Exceptions:

Actor Action	System Response
1. [Solution] text area is blank.	2. Display message: "Please fill out this field." on the fields that is blank.

Relationships:

Manage Symptom

Business Rules:

- Technical Manager wants to update information of existed symptom.
- Technical Manager clicks on a symptom row in [Symptom List] table to update.
- Edited symptom information is updated to database and changes are update in [Symptom List] table.

2.2.39. < Technical Manager > [Manage Equipment] Add Equipment

2.2.39.1. Use Case Diagram

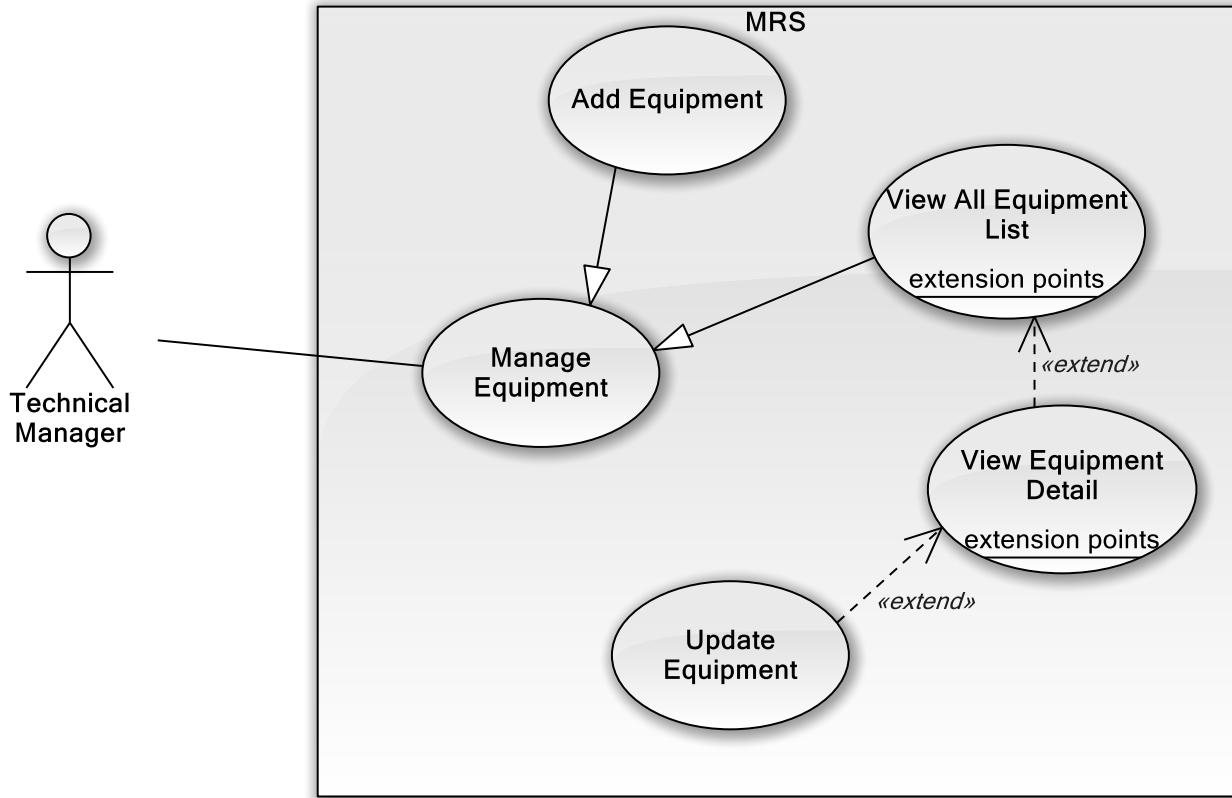


Figure 51-Manger Manage Equipment use case diagram

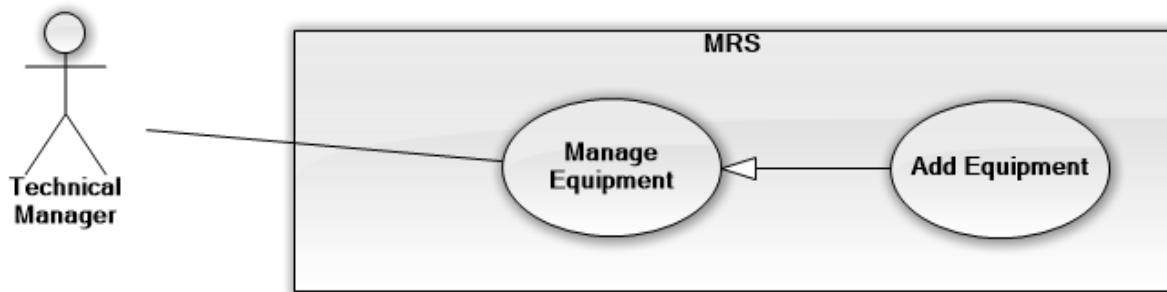


Figure 52-Manger Add Equipment usecase diagram

2.2.39.2. Use Case Specification

USE CASE – Add Equipment SPECIFICATION

Use case No.	UC039	Use case version	2.0
Use case name	Add Equipment		

Author	KieuDTD		
Date	21/6/2013	Priority	Normal

Actor:

Technical Manager

Summary:

Technical manager uses this use case to create new equipment when have new equipment that have not existed in inventory yet. When create new equipment successfully, it will store in database and Technical Manager and customer can view this new equipment.

Goal:

Store new equipment information to database for later management.

Triggers:

In [Manage Equipment Page], Technical Manager can create new equipment by clicks on [ADD EQUIPMENT] button; [Add Equipment] form will be shown with necessary information of equipment to fill.

Preconditions:

User logged in successfully with role of Technical Manager.

Post Conditions:

- **Success:** New equipment information will be stored in database and new row of equipment will be inserted to [Equipment List] table.
- **Fail:** Redirect to [Error Page] or still at current page with error messages.

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [EQUIPMENT] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Equipment Page] with [Equipment List] table
5. Technical Manager clicks on [Add Equipment] button.	<p>6. Expand an [Add Equipment] form includes:</p> <ul style="list-style-type: none"> - Equipment Name: textbox (required, max length: 100) - Equipment Category: dropdown list (Equipment belongs to which category) - Price: Text box (required) - Add more images : click on this button will display: Image: file (choose file) and Remove button - Description: text area

	<ul style="list-style-type: none"> - CLOSE: button [Alternative 1] - ADD: button 	
7. Enter information into text fields.	8. [Exception 1, 3]	
9. Click on [ADD] button to finish this use case.	10. New equipment information will be stored in database and new row of equipment will be inserted to [Equipment List] table.	

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish this use case.	2. All information in [Add Equipment] form is cleared.

Exceptions:

Actor Action	System Response
1. [Equipment Name] textbox, [Price] textbox, [Description] text area is blank.	2. Display message: “Please fill out this field.” on the fields that is blank.
3. Enter invalid value to [Price] textbox	4. Display message: “Please enter a number”.

Relationships:

Manage Equipment

Business Rules:

- Technical Manager wants to store new equipment information to database for later use.
- Technical Manager fills necessary information about new equipment into [Add Equipment] form and click [ADD] button to finish.
- New equipment is stored in database and new row is inserted to [Equipment List] table.
- Maximum image number for equipment is 3 images.

2.2.40. < Technical Manager > [Manage Equipment] View All Equipment List

2.2.40.1. Use Case Diagram

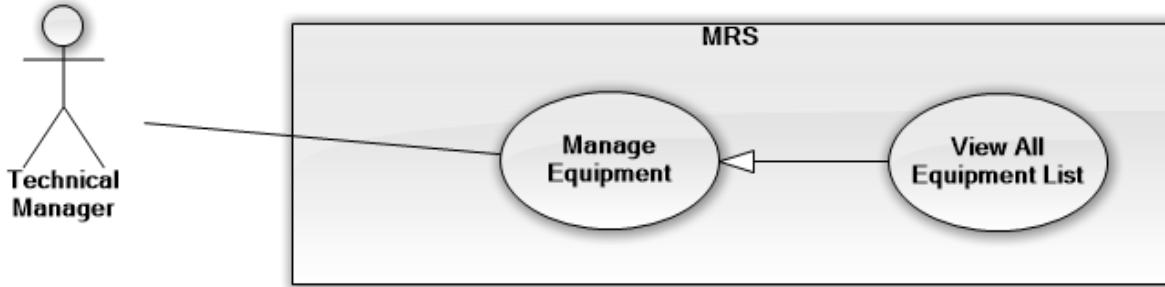


Figure 53-Manger View All Request List usecase diagram

2.2.40.2. Use Case Specification

USE CASE – View All Equipment List SPECIFICATION							
Use case No.	UC040	Use case version	2.0				
Use case name	View All Equipment List						
Author	KieuDTD						
Date	21/6/2013	Priority	Normal				
Actor:	Technical Manager						
Summary:	Technical Manager uses this case to tracking all equipment in inventory. These equipment use for maintenance or repairing of staff.						
Goal:	Technical manager can view all equipment in list with table format, which have some information such as equipment name, price, category...						
Triggers:	Technical manager can view all list equipment by clicks on [EQUIPMENT] tab in [Manage Repair Maintenance Page]. [Manage Equipment Page] will be shown with [Equipment List] table.						
Preconditions:	User logged in successfully with role of Technical Manager.						
Post Conditions:	<ul style="list-style-type: none"> - Success: List all equipment in inventory. - Fail: N/A 						
Main Success Scenario:	<table border="1"> <thead> <tr> <th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1. Technical Manager clicks on</td><td>2. Redirect to [Manage Repair</td></tr> </tbody> </table>			Actor Action	System Response	1. Technical Manager clicks on	2. Redirect to [Manage Repair
Actor Action	System Response						
1. Technical Manager clicks on	2. Redirect to [Manage Repair						

[Request] box on [Home page].	Maintenance Page].	
3. Technical Manager clicks on [EQUIPMENT] tab in [Manage Repair Maintenance Page].	<p>4. Redirect to [Manage Equipment Page]</p> <p>List all equipment in inventory in [Equipment List] table with columns:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Equipment Name: Name of equipment - Category: Equipment belongs to which category - Price: Price of equipment - Active: Disable or enable equipment 	

Alternative Scenario:
N/A

Exceptions:
N/A

Relationships:
Manage Equipment

Business Rules:

- Technical Manager wants to view all equipment in inventory.
- All equipment will display with order from latest to oldest updated date.
- Technical manager can choose the maximum item on list paging is 10, 15, 25, etc.

2.2.41. < Technical Manager > [Manage Equipment] View Equipment Detail

2.2.41.1. Use Case Diagram

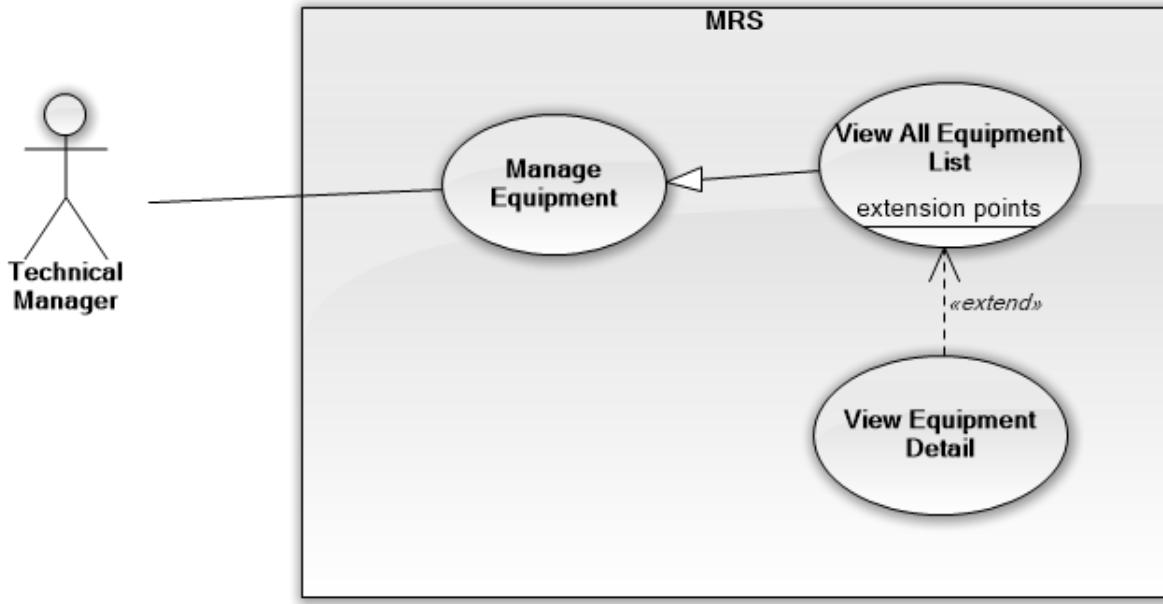


Figure 54-Manager View Equipment Detail usecase diagram

2.2.41.2. Use Case Specification

USE CASE – View Equipment Detail SPECIFICATION			
Use case No.	UC041	Use case version	2.0
Use case name	View Equipment Detail		
Author	KieuDTD		
Date	21/6/2013	Priority	Normal
Actor: Technical Manager			
Summary: Technical Manager want to view detail about equipment, they just select equipment in [Equipment List] table and view more detail about that equipment.			
Goal: Technical Manager has a clear detail about equipment that they want to focus. Additional, this use case can extend to help Technical Manager update equipment easily which will be described more detail in [Update Equipment] use case below.			
Triggers: In [Manage Equipment Page], Technical Manager can view detail of equipment by clicks on that equipment in [Equipment List] table.			
Preconditions:			

User logged in successfully with role of Technical Manager and Technical Manager click on a row of equipment in the body of [Equipment List] table (except table header row).

Post Conditions:

- **Success:** Detail information of clicked equipment will be shown in [Equipment Detail] form on pop up.
- **Fail:** N/A

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [EQUIPMENT] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Equipment Page] with [Equipment List] table
5. Technical Manager clicks on a row in body of [Equipment List] table to finish this use case.	<p>6. Show [Equipment Detail] form with detail information of clicked equipment includes:</p> <ul style="list-style-type: none"> - Equipment Name: textbox (max length: 100) - Equipment Category: dropdown list (Equipment belongs to which category) - Price: textbox - Add more images : click on this button will display: Image: file (choose file) and Remove button - Description: text area - Active: checkbox - CLOSE: button - UPDATE: button

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Manage Equipment

Business Rules:

- Technical Manager wants to view detail information of equipment in [Equipment List] table.

- Technical Manager clicks on equipment row in [Equipment List] table and detail information of that equipment will be showed detail in [Equipment Detail] form.

2.2.42. < Technical Manager > [Manage Equipment] Update Equipment

2.2.42.1. Use Case Diagram

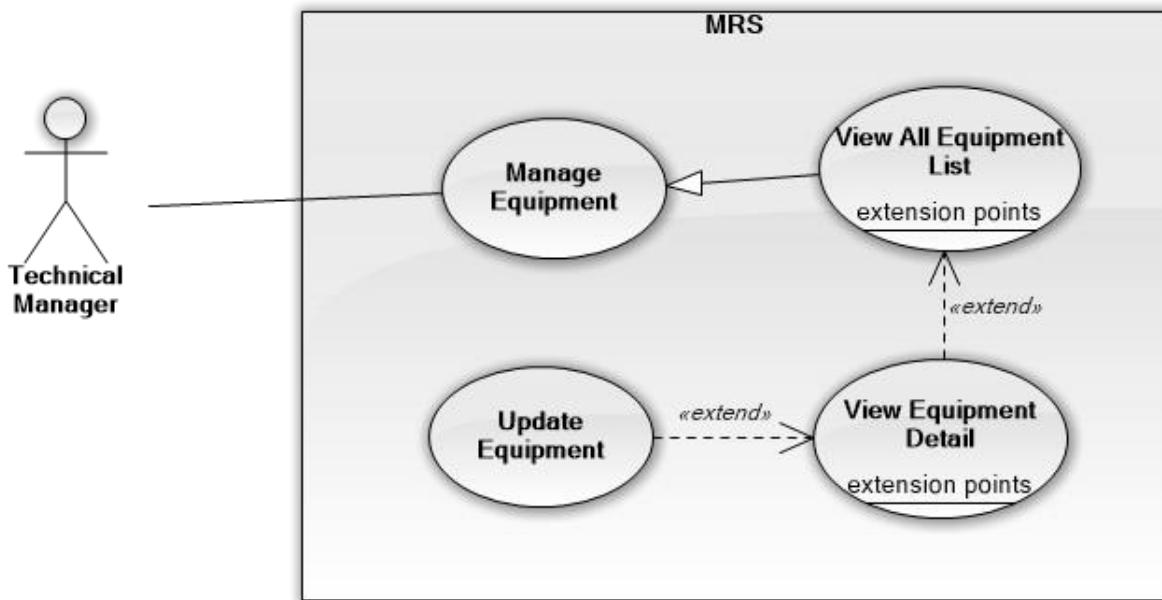


Figure 55-Manger Update Equipment usecase digram

2.2.42.2. Use Case Specification

USE CASE – Update Equipment SPECIFICATION			
Use case No.	UC042	Use case version	2.0
Use case name	Update Equipment		
Author	KieuDTD		
Date	21/6/2013	Priority	Normal
Actor: Technical Manager			
Summary: For a reason, Technical Manager want to update content of equipment that existed, this use case can help him (her) to do it easily.			
Goal: Update existed equipment information in database such as equipment name, equipment belong to another category or image...			
Triggers:			

In [Manage Equipment Page], Technical Manager can update equipment by clicks on that equipment in [Equipment List] table. [Equipment Detail] form will be shown with detail information of that equipment and from now, Technical Manager can edit information for his purpose and stored edited information to database.

Preconditions:

User logged in successfully with role of Technical Manager and Technical Manager click on a row of equipment in the body of [Equipment List] table that he (she) wants to update (except table header row).

Post Conditions:

- **Success:** Edited equipment information is updated to database and changes of that edited equipment are update in [Equipment List] table.
- **Fail:** Redirect to [Error Page] or still at current page with error messages.

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [EQUIPMENT] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Equipment Page] with [Equipment List] table
5. Technical Manager clicks on a row in body of [Equipment List] table	<p>6. Show [Equipment Detail] form with detail information of clicked equipment includes:</p> <ul style="list-style-type: none"> - Equipment Name: textbox (max length: 100) - Equipment Category: dropdown list (Equipment belongs to which category) - Price: textbox - Add more images : click on this button will display: Image: file (choose file) and Remove button - Description: text area - Active: checkbox - CLOSE: button [Alternative 1] - UPDATE: button
7. Technical Manager make some changes for his purpose by edit text fields	8. [Exception 1, 3]

9. Click on [UPDATE] button to finish this use case.	10. Edited equipment information is updated to database and changes of that edited.
--	---

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish this use case.	2. Nothing changes and [Equipment Detail] form is hidden.

Exceptions:

Actor Action	System Response
1. [Equipment Name] textbox, [Price] textbox, [Description] text area are blank.	2. Display message: “Please fill out this field.” on the fields that is blank.
3. Enter invalid value to [Price] textbox	4. Display message: “Please enter a number”.

Relationships:

Manage Equipment

Business Rules:

- Technical Manager wants to update information of existed equipment.
- Technical Manager clicks on equipment row in [Equipment List] table to update.
- Edited equipment information is updated to database and changes are update in [Equipment List] table.
- Maximum image number for equipment is 3 images. So, updating image cannot overhead this number

2.2.43. < Technical Manager > [Statistics] Statistics by Request Type

2.2.43.1. Use Case Diagram

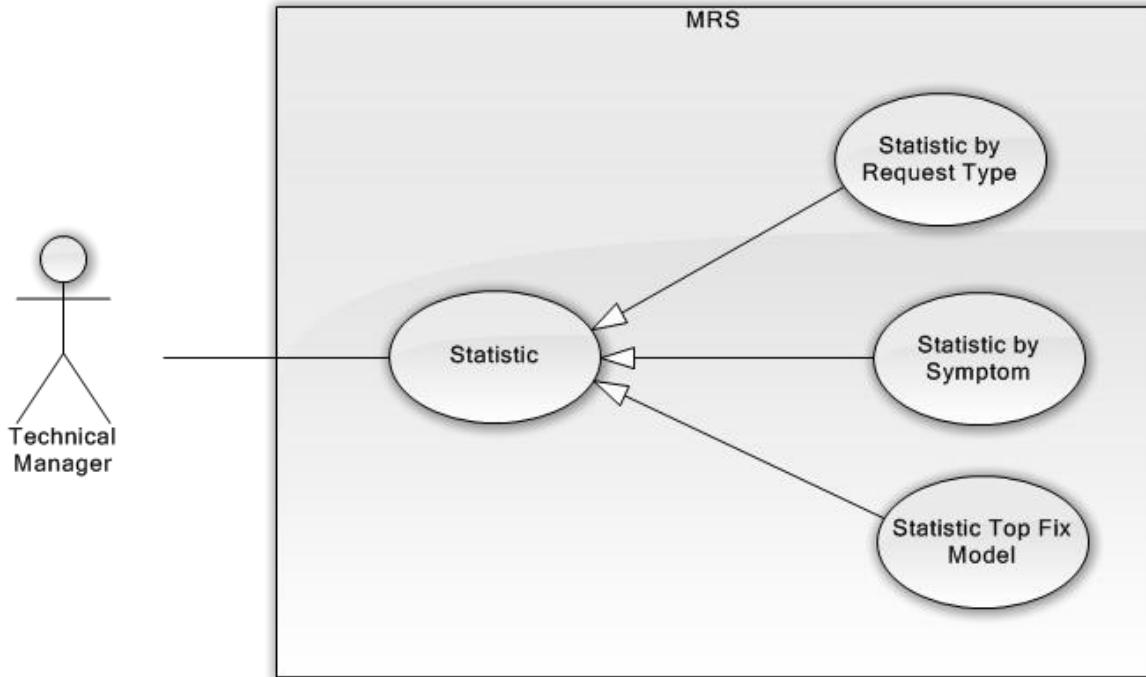


Figure 56-Statistics use case diagram

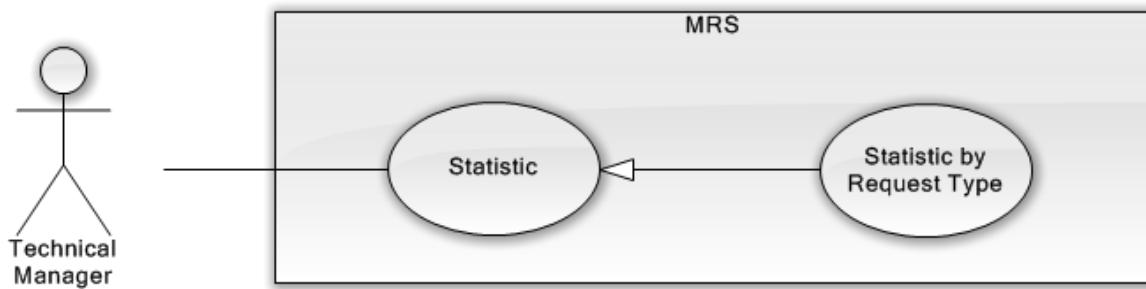


Figure 57-Statistic by request type usecase diagram

2.2.43.2. Use Case Specification

USE CASE – Statistics Maintenance or Repair SPECIFICATION			
Use case No.	UC043	Use case version	2.0
Use case name	Statistics by Request Type		
Author	HungDD		

Date	22/6/2013	Priority	Normal								
Actor: Technical Manager											
Summary: Technical Manager wants to track which request type of customer is popular (maintenance or repair), so this use case help Technical Manager to compare about that.											
Goal: Show total maintenance request compare with repair request in every month and every year, display in data in table and bar chart also.											
Triggers: In [Statistics Page], Technical Manager clicks on [Maintenance/Repair] link on the left side bar. Technical Manager can select statistic by year in [Year] combobox.											
Preconditions: User logged in successfully with role of Technical Manager and had some more requests in the past.											
Post Conditions: <ul style="list-style-type: none"> - Success: Display data of maintenance and repair in table format and in bar chart. - Fail: Redirect to [Error Page] or still at current page with error messages. 											
Main Success Scenario:											
<table border="1"> <thead> <tr> <th>Actor Action</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1. Technical Manager clicks on [Request] box on [Home page].</td> <td>2. Redirect to [Manage Repair Maintenance Page].</td> </tr> <tr> <td>3. Technical Manager clicks on [STATISTIC] tab on [Manage Repair Maintenance Page].</td> <td> 4. Redirect to [Statistics Page] include: <ul style="list-style-type: none"> - Year : combobox (display year value) - Data table : table (show data in 12 months of selected year) - Data chart: chart (line chart) </td> </tr> <tr> <td>5. Technical Manager selects another year in [Year] combobox</td> <td>6. Data in table is changed corresponding to selected year and chart is updated also.</td> </tr> </tbody> </table>				Actor Action	System Response	1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].	3. Technical Manager clicks on [STATISTIC] tab on [Manage Repair Maintenance Page].	4. Redirect to [Statistics Page] include: <ul style="list-style-type: none"> - Year : combobox (display year value) - Data table : table (show data in 12 months of selected year) - Data chart: chart (line chart) 	5. Technical Manager selects another year in [Year] combobox	6. Data in table is changed corresponding to selected year and chart is updated also.
Actor Action	System Response										
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].										
3. Technical Manager clicks on [STATISTIC] tab on [Manage Repair Maintenance Page].	4. Redirect to [Statistics Page] include: <ul style="list-style-type: none"> - Year : combobox (display year value) - Data table : table (show data in 12 months of selected year) - Data chart: chart (line chart) 										
5. Technical Manager selects another year in [Year] combobox	6. Data in table is changed corresponding to selected year and chart is updated also.										
Alternative Scenario: N/A											
Exceptions: N/A											
Relationships: Statistics											
Business Rules: <ul style="list-style-type: none"> - Display current year in [Select year] combobox by default. - Display data of 12 months of selected year in table. 											

- Display in bar chart also.

2.2.44. < Technical Manager > [Statistics] Statistic Top Fixed Model

2.2.44.1. Use Case Diagram

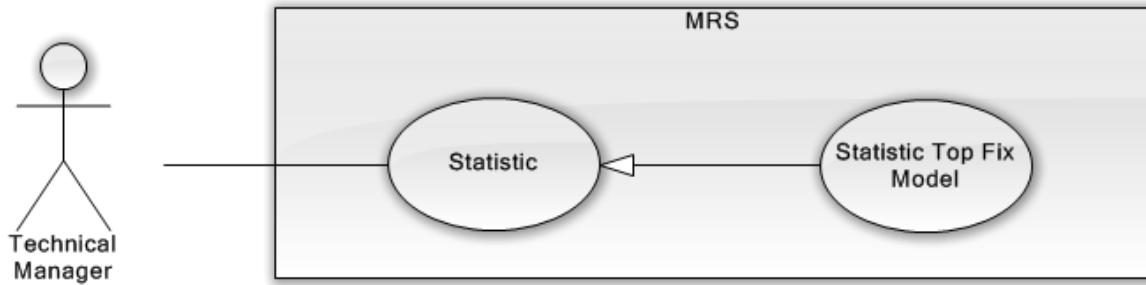


Figure 58-Statistic Top fixed model use case diagram

2.2.44.2. Use Case Specification

USE CASE – Statistics Top Fixed Model SPECIFICATION			
Use case No.	UC044	Use case version	2.0
Use case name	Statistic Top Fixed Model		
Author	ThanhTM		
Date	22/6/2013	Priority	Normal
Actor: Technical Manager			
Summary: Technical Manager wants to track which product categories, product models are repaired or maintained in top 5 or 10			
Goal: Show top products that repaired or maintenance most. Then view the relate request to that product. Base on the information, the manager can realize which equipment should the center import more or collect. Or should the center have to manage the staff again to suitable for the work, add more staff who have skill suitable to that kind of product.			
Triggers: In [Statistics Page], Technical Manager clicks on [TOP FIXED MODEL] link on the left side bar. Technical Manager can select the number of result in [Amount] combo box.			
Preconditions: User logged in successfully with role of Technical Manager and had some more requests in the past.			
Post Conditions: - Success: Display data of top fixed model in table format and in line chart.			

- **Fail:** Redirect to [Error Page] or still at current page with error messages.

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [STATISTIC] tab navigation.	4. Redirect to [Statistics Page]
5. Technical Manager clicks on [TOP FIXED MODEL] on left menu navigation.	6. Redirect to [Top Fixed Model Statistics Page] include: <ul style="list-style-type: none"> - Amount : combo box (Choose number of result) - Data table : table (show top fixed product categories) - Data chart: chart (pie chart)
7. Technical Manager selects a row of category in data table.	8. Page is scrolled down automation and display the data table of top fixed products model related to selected product category. And a pie chart relates to that data is drawn.
9. Technical Manager selects a row of product model in data table.	10. Page is scrolled down automation and display the table of requests related to selected product model.

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Statistics

Business Rules:

- [Amount] combo box is set to 5 by default.
- Display data of top 5 fixed products categories by default.
- Display pie chart also.
- When click on a row of category, display top fixed product models relate to that category, and a pie chart also.
- When click on a row of product model, display top request of that product models.
- When click on a row of request, a popup of request detail will display which information about that request.

2.2.45. < Technical Manager > [Statistics] Statistic by Symptom

2.2.45.1. Use Case Diagram

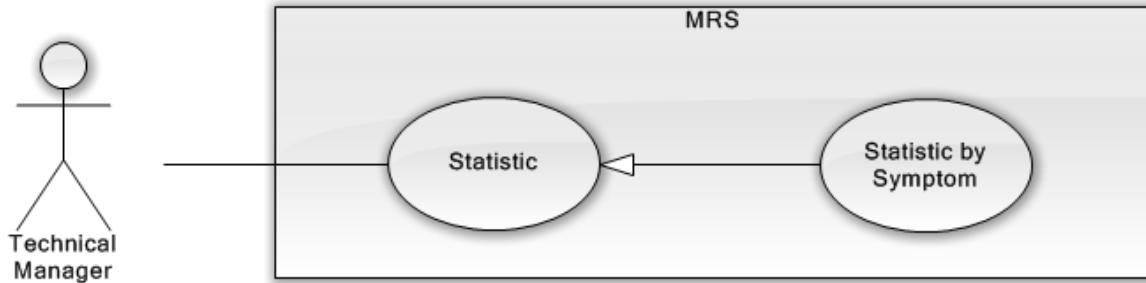


Figure 59-Statistic by symptom usecase diagram

2.2.45.2. Use Case Specification

USE CASE – Statistics by Symptom SPECIFICATION			
Use case No.	UC045	Use case version	2.0
Use case name	Statistic by Symptom		
Author	ThanhTM		
Date	22/6/2013	Priority	Normal
Actor: Technical Manager			
Summary: Technical Manager wants to add a new FAQ, and they don't know what symptom should they use to create a new FAQ.			
Goal: Base on top fixed product models, the manager can see a list of symptom of that product, then with those symptoms, the manager can give out a summary of solution to create a FAQ for customer			
Triggers: In [Statistics Page], Technical Manager clicks on [SYMPTOM STATISTIC] link on the left side bar. Technical Manager can select the number of result will display in data table in [Result] combo box, then select the category, the product model to view symptom list.			
Preconditions: User logged in successfully with role of Technical Manager and had some more requests in the past			
Post Conditions: <ul style="list-style-type: none"> - Success: Display symptom of selected product model in table format. - Fail: Redirect to [Error Page] or still at current page with error messages 			
Main Success Scenario:			

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [STATISTIC] tab navigation.	4. Redirect to [Statistics Page]
5. Technical Manager clicks on [TOP FIXED SYMPTOM] on left menu navigation	<p>6. Redirect to [Top Fixed Symptom Statistics Page] include:</p> <ul style="list-style-type: none"> - Result : combo box (number of result will display) - Top fixed category :combo box (list of top fixed category) <p>Top fixed model: combo box (number of top fixed model relate to category)</p>

Alternative Scenario:
N/A

Exceptions:
N/A

Relationships:
Statistics

Business Rules:

- [Result] combo box is set to 5 by default.
- [Product Categories] combo box is set to first category in list by default.
- [Product Models] combo box is set to first model in list by default.
- Display a symptom list in data table.

2.2.46. < Technical Manager > [Manage User] Add new Account

2.2.46.1. Use Case Diagram

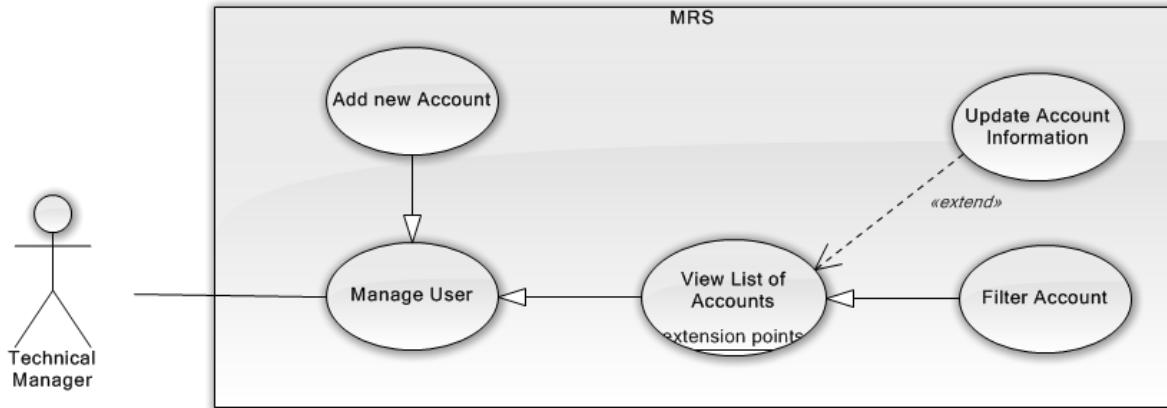


Figure 60-Manger Manage User use case diagram

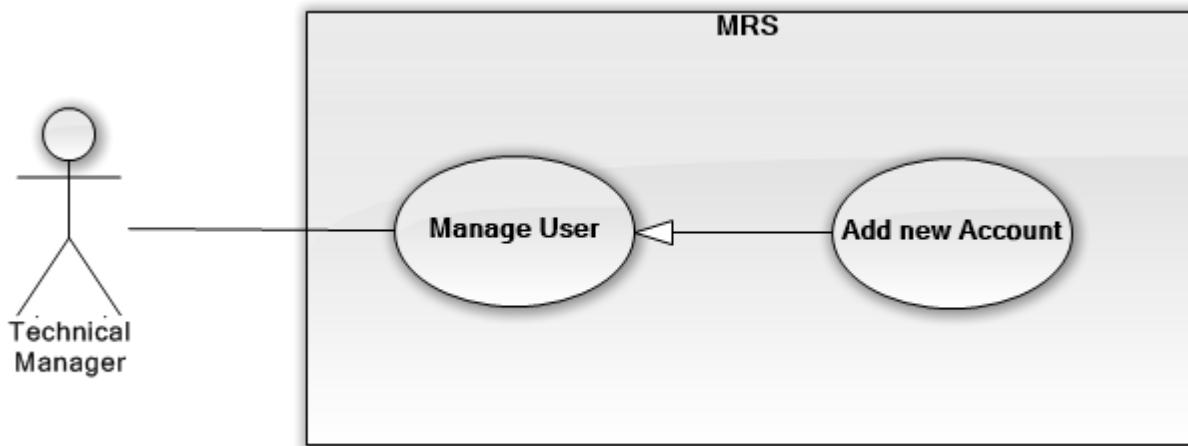


Figure 61-Manger Add New Account usecase digram

2.2.46.2. Use Case Specification

USE CASE – Add New Account SPECIFICATION			
Use case No.	UC046	Use case version	2.0
Use case name	Add New Account		
Author	ThanhTM		
Date	22/05/2013	Priority	Normal
Actor:	Technical Manager.		

Summary:

Technical Manager uses this case to create a new account. When create a new account successfully, it will store in database and user can use this account to login to system, and Technical Manager can view the account detail later.

Goal:

Create a new account for user.

Triggers:

In [Account Management] page, Technical Manager clicks on [Add Account] button to display [Add New Account] form.

Preconditions:

User logged in successfully with role [Technical Manager]

Post Conditions:

- **Success:** New account will be added and new row of account will be inserted to [Account List] table.
- **Fail:** Redirect to [Error Page] or still at current page with error messages

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page]
3. Technical Manager clicks on [Account] tab on [Manage Repair Maintenance Page].	4. Display [Account Management] page
5. Technical Manager clicks on [Add Account] button.	<p>6. Show [Add New Account] form includes:</p> <ul style="list-style-type: none"> - Role: dropdown list (Required, Role list) - First name : Text box (required) - Last name : Text box (required) - User Name: textbox (required, max length: 20) - Avatar: file (choose file) - Gender: dropdown list (required, Male, Female) - Email: textbox (required) - Gender: Dropdown list - Phone number: textbox (required) - Address: textbox (required) - Active: checkbox (not check) - CLOSE: button [Alternative 1]

	- ADD: button
7. Enter information into fields.	8. [Exception 1, 3, 5]
9. Clicks on [ADD] button to finish this use case	10. New account will be added and new row of account will be inserted to [Account List] table with added account information.

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish this use case.	2. Cancel the adding account and close the [Add New Account] form.

Exceptions:

Actor Action	System Response
1. [First Name], [Last Name], [User Name], [Email], [Phone number], [Address] are blank.	2. Display message: "Please fill out this field." on the fields that is blank.
3. Phone Number is not a number.	4. Display error message: "Phone must be a number."
5. Phone number is less than 9 numbers or greater than 11 numbers.	6. Display error message: "Please match the requested format. 9 to 11 characters [0-9]."

Relationships:

Manage User

Business Rules:

- Technical Manager fills necessary information about account onto [Add New Account] form and submits.
- If there are any problem on validate form, an error bar will appear with error message.
- New account is added and new row of account is inserted to [Account List] table.
- When account is created, the system also pushes an email to send password for customer.
- With role is "Customer", [Username] field is enabled to edit; When role is not "Customer", [Username] is disabled and will be auto generate when [Technical Manager] submits add account

2.2.47. < Technical Manager> [Manage User] View List of Accounts

2.2.47.1. Use Case Diagram

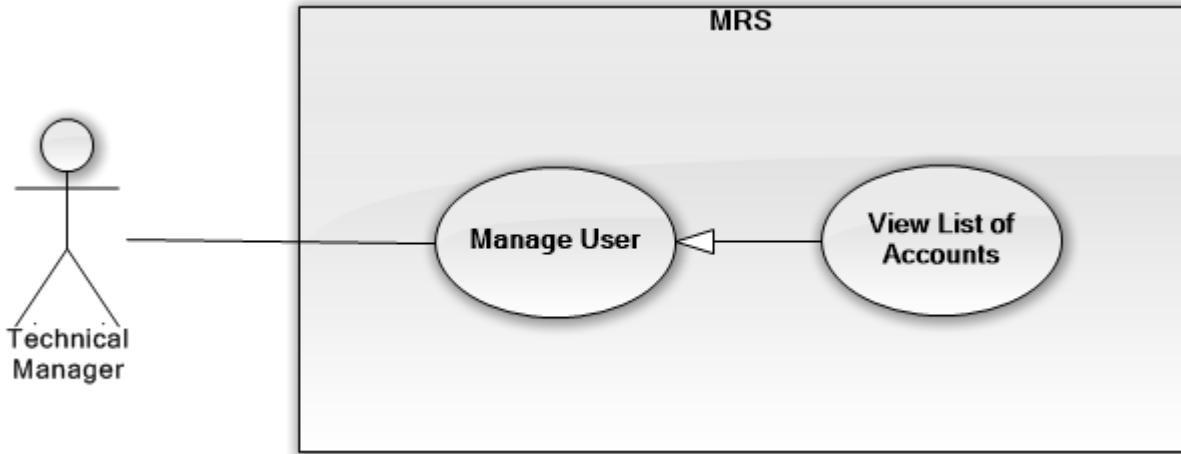


Figure 62-Manger View List Of Account usecase diagram

2.2.47.2. Use Case Specification

USE CASE – View List Of Accounts SPECIFICATION			
Use case No.	UC047	Use case version	2.0
Use case name	View List of Accounts		
Author	ThanhTM		
Date	22/05/2013	Priority	Normal
Actor: Technical Manager.			
Summary: Technical Manager uses this case to manage all user accounts of the system. This case allow Technical Manager to view the list of all accounts with detail information.			
Goal: Technical Manager can view all accounts by list with table format. Technical Manager can manage information about account such as: username, full name, gender, email, phone number, address, role, etc.			
Triggers: Technical Manager can view list of accounts by clicks on [ACCOUNT] tab on [Manage Repair Maintenance Page] to display [Account Management] page.			
Preconditions: User logged in successfully with role [Technical Manager].			
Post Conditions: - Success: List of accounts will be displayed as a table with account's information: No,			

profile picture, username, full name, gender, email, phone number, address, role, activation, etc.

- **Fail:** N/A.

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [Account] tab on [Manage Repair Maintenance Page].	<p>4. Display [Account Management] page</p> <p>List of all accounts information in [Account List] table with columns:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Profile Picture: image - User Name: login id of user - Full name: Full name of user - Gender - Email - Phone Number - Address - Role: Role of user (customer, staff, technical manager) - Activation: account is enabled or disabled

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Manage User

Business Rules:

- Technical Manager wants to view all account of system.
- All accounts will display in a list with order from latest to oldest updated date.
- Technical manager can choose the maximum item on list paging is 10, 15, 25, etc.

2.2.48. < Technical Manager> [Manage User] Filter Account

2.2.48.1. Use Case Diagram

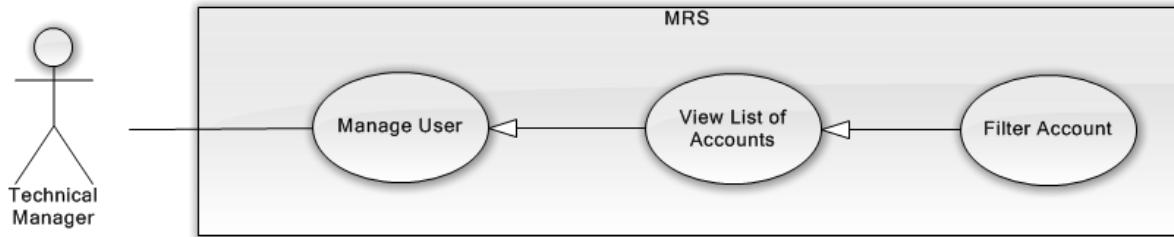


Figure 63-Mager Filter Account usecase diagram

2.2.48.2. Use Case Specification

USE CASE – Filter Account SPECIFICATION								
Use case No.	UC048	Use case version	2.0					
Use case name	Filter Account							
Author	ThanhTM							
Date	22/05/2013	Priority	Normal					
Actor: Technical Manager.								
Summary: Technical Manager manage all user accounts of the system. By using this function, they can filter account list according email, phone number or role of account.								
Goal: The result of filtered would be shown in the [Task List] table.								
Triggers: In [Account Management] page, Technical Manager can choose type to filter. Then input value into text box and click [Filter] button to finish.								
Preconditions: User logged in successfully with role [Technical Manager].								
Post Conditions: <ul style="list-style-type: none">- Success: The result of filtered would be shown in the [Task List] table.- Fail: N/A.								
Main Success Scenario:								
<table border="1"> <thead> <tr> <th>Actor Action</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1. Technical Manager clicks on [Request] box on [Home page].</td> <td>2. Redirect to [Manage Repair Maintenance Page].</td> </tr> <tr> <td>3. Technical Manager clicks on</td> <td>4. Display [Account Management] page</td> </tr> </tbody> </table>			Actor Action	System Response	1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].	3. Technical Manager clicks on	4. Display [Account Management] page
Actor Action	System Response							
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].							
3. Technical Manager clicks on	4. Display [Account Management] page							

[Account] tab on [Manage Repair Maintenance Page].	with [Account List] table	
5. Select type to filter, input value into text box and click on [Filter] button	6. The result of filtered would be shown in the [Task List] table.	

Alternative Scenario:
N/A

Exceptions:
N/A

Relationships:
Manage User use case, View List of Account use case.

Business Rules:

- Technical Manager wants to filter all account of system.

2.2.49. < Technical Manager> [Manage User] Update Account Information

2.2.49.1. Use Case Diagram

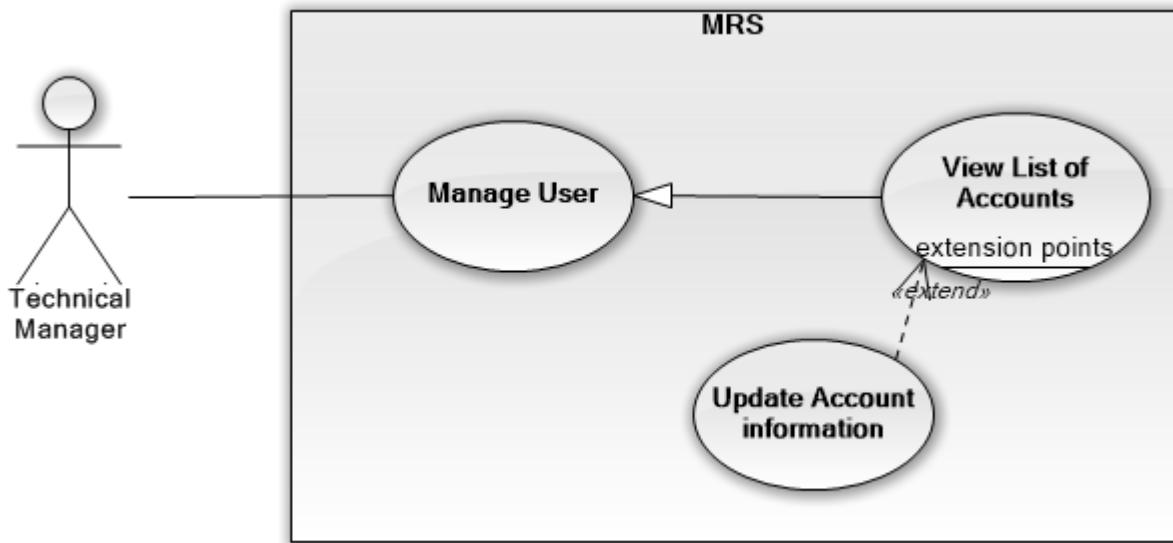


Figure 64-Manger Update Account usecase diagram

2.2.49.2. Use Case Specification

USE CASE – Update Account Information SPECIFICATION			
Use case No.	UC049	Use case version	2.0

Use case name	Update Account Information		
Author	ThanhTM		
Date	22/05/2013	Priority	Normal

Actor:

Technical Manager.

Summary:

For a reason, Technical Manager wants to change information of an account such as: role, skill list, activation, and avatar. This case helps him to do that.

Goal:

Technical Manager uses this case to update the information of accounts.

Triggers:

In [Account Management] page, Technical Manager clicks on an account in [Account list] to display the [Account Detail - Update] form relate to that account.

Preconditions:

User logged in successfully with role [Technical Manager]

Post Conditions:

- **Success:** Updated account's information will be updated in database and in [Account List] table.
- **Fail:** Redirect to [Error Page] or still at current page with error messages

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [Account] tab on [Manage Repair Maintenance Page]	4. Display [Account Management] page with [Account List] table
5. Technical Manager clicks on a row in body of [Account List] table	<p>6. Display an [Account Detail- Update] form includes:</p> <ul style="list-style-type: none"> - User Name: textbox (disabled) - Full name: textbox (disabled) - Current Avatar- New Avatar: File (choose file) - Active: checkbox (current status of that account) - Role: dropdown list (Current role of that account) - Staff skill: text field – dropdown list (Skill list of staff, display when

	<p>selected role is “Staff”</p> <ul style="list-style-type: none"> - CLOSE: button [Alternative 1] - UPDATE: button 	
7. Change the information as needed and Technical Manager clicks on [UPDATE] button	8. Updated account’s will be updated as changed information in database and in [Account List] table	

Alternative Scenario:

Actor Action	System Response
1. Click on [CANCEL] button to finish this use case.	2. Cancel the updating account and close the [Account Detail- Update] form.

Exceptions:
N/A

Relationships:
Manage user use case, View List of Account use case.

Business Rules:

- Technical Manager wants to update account information.
- Technical Manager changes account information as needed into [Account Detail- Update] form and submits.
- New information of account will be updated in database and in [Account List] table.

2.2.50. < Technical Manager > [Manage Product] Add Product Category

2.2.50.1. Use Case Diagram

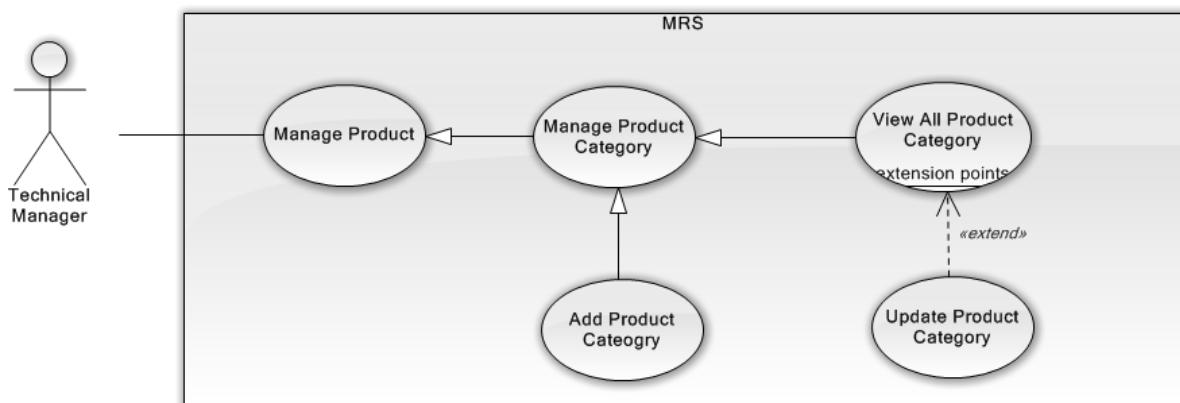
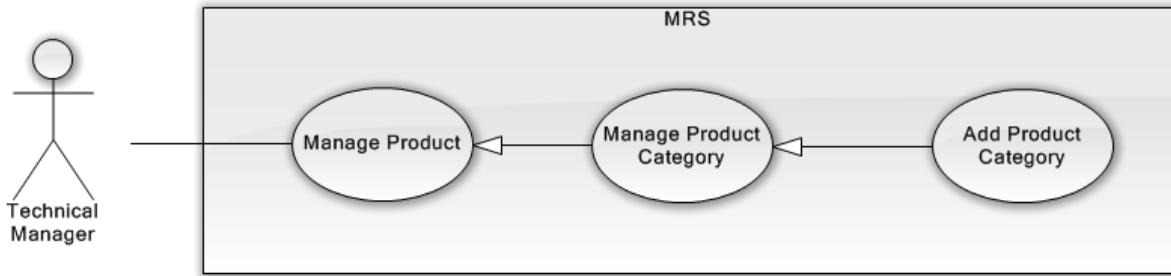


Figure 65-Manage Product Category use case diagram



2.2.50.2. Use Case Specification

USE CASE – Add Product Category SPECIFICATION			
Use case No.	UC050	Use case version	2.0
Use case name	Add Product Category		
Author	KieuDTD		
Date	7/8/2013	Priority	Normal

Actor:

Technical Manager

Summary:

Technical manager uses this use case to create new product category to the system. When create new product category successfully, it will store in database and Technical Manager can view to manage later.

Goal:

Store new product category to database for later management

Triggers:

In [Manage Product Category Page], Technical Manager can create new product category by clicks on [ADD CATEGORY] button; [Add Category] form will be shown with necessary information of category to fill.

Preconditions:

User logged in successfully with role of Technical Manager.

Post Conditions:

- **Success:** New product category will be stored in database and new row of category will be inserted to [Category List] table.
- **Fail:** Redirect to [Error Page] or still at current page with error messages

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [PRODUCT] tab in [Manage Repair	4. Redirect to [Manage Product Category]

Maintenance Page].	Page] with [Category List] table	
5. Technical Manager clicks on [ADD CATEGORY] button.	6. Show [Add Category] form includes: <ul style="list-style-type: none">- Category Name: textbox (required, max length: 100)- Default cost: text box- CLOSE: button [Alternative 1]- ADD: button	
7. Enter information into text fields.	8. [Exception 1]	
9. Click on [ADD] button to finish this use case.	10. New category will be stored in database and new row of category will be inserted to [Category List] table.	

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish this use case.	2. All information in [Add Category] form is cleared.

Exceptions:

Actor Action	System Response
1. Enter invalid value to [Default Cost] textbox	2. Display message: “Please enter a number”.

Relationships:

Manage Product use case, Manage Product Category use case.

Business Rules:

- Technical Manager wants to store new product category to database for later use.
- Technical Manager fills necessary information about new category into [Add Category] form and click [ADD] button to finish.
- New category is stored in database and new row is inserted to [Category List] table.
- Technical manager can choose the maximum item on list paging is 10, 15, 25, etc.

2.2.51. < Technical Manager > [Manage Product] View All Product Category List

2.2.51.1. Use Case Diagram

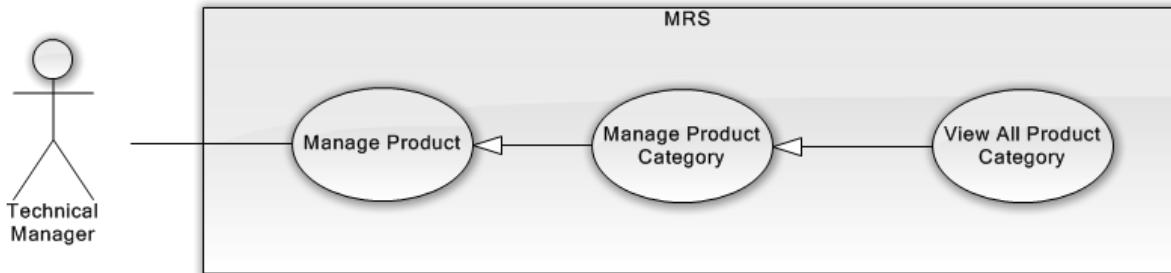


Figure 66-Manger View All Product Category usecase diagram

2.2.51.2. Use Case Specification

USE CASE – View All Product Category List SPECIFICATION			
Use case No.	UC051	Use case version	2.0
Use case name	View All Product Category List		
Author	KieuDTD		
Date	7/8/2013	Priority	Normal
Actor: Technical Manager			
Summary: Technical Manager uses this case to tracking all product categories in the system.			
Goal: Technical manager can view all product categories in list with table format, which have some information such as category name, default cost...			
Triggers: Technical manager can view all list product categories by clicks on [PRODUCT] tab in [Manage Repair Maintenance Page]. [Manage Product Category Page] will be shown with [Category List] table.			
Preconditions: User logged in successfully with role of Technical Manager.			
Post Conditions: <ul style="list-style-type: none"> - Success: List all product categories in the system. - Fail: N/A 			
Main Success Scenario:			
	Actor Action	System Response	
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].		

<p>3. Technical Manager clicks on [PRODUCT] tab in [Manage Repair Maintenance Page].</p>	<p>4. Redirect to [Manage Product Category Page]</p> <p>List all product category in [Category List] table with columns:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Category Name: Name of category - Default cost: Price of equipment - Activation: Disable or enable product category 	
--	---	--

Alternative Scenario:

N/A

Exceptions:

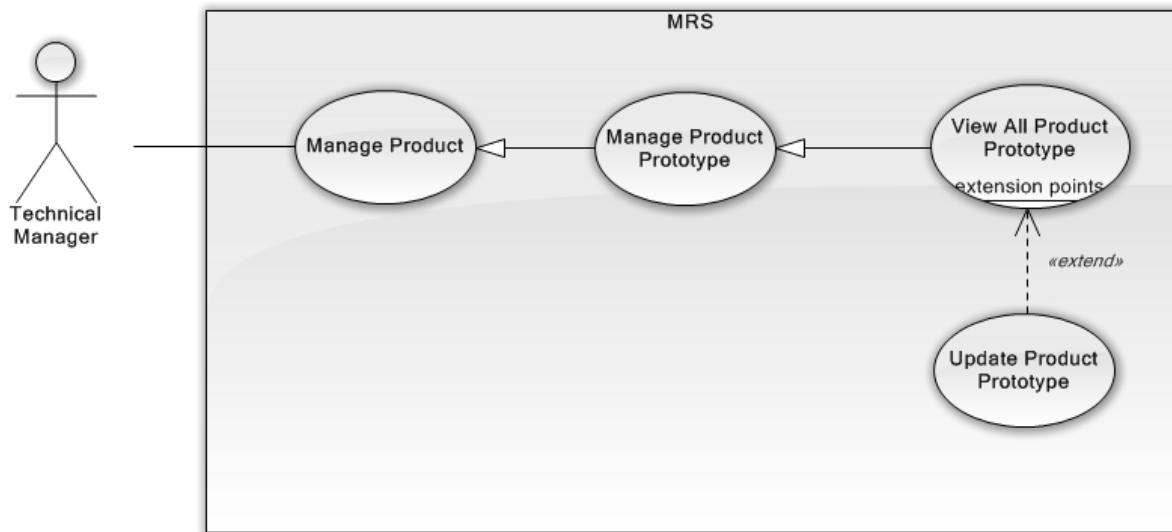
N/A

Relationships:

Manage Product use case, Manage Product Category use case.

Business Rules:

- Technical Manager wants to view all product categories.
- All categories will display with order from latest to oldest updated date.
- Technical manager can choose the maximum item on list paging is 10, 15, 25, etc.

2.2.52. < Technical Manager > [Manage Product] Update Product Category**2.2.52.1. Use Case Diagram****Figure 67-Manger Update Product Prototype usecase diagram**

2.2.52.2. Use Case Specification

USE CASE – Update Product Category SPECIFICATION			
Use case No.	UC052	Use case version	2.0
Use case name	Update Product Category		
Author	KieuDTD		
Date	7/8/2013	Priority	Normal
Actor: Technical Manager			
Summary: For a reason, Technical Manager want to update content of product category that existed, this use case can help them to do it easily.			
Goal: Update existed category information in database such as default cost, active category or not			
Triggers: In [Manage Product Category Page], Technical Manager can update category by clicks on that category in [Category List] table. [Category Detail] form will be shown with detail information of that product category and from now, Technical Manager can edit information and update to database.			
Preconditions: User logged in successfully with role of Technical Manager and Technical Manager click on a row of category in the body of [Category List] table that they wants to update (except table header row).			
Post Conditions: <ul style="list-style-type: none"> - Success: Edited information is updated to database and changes of that edited category are update in [Category List] table. - Fail: Redirect to [Error Page] or still at current page with error messages. 			
Main Success Scenario:			
Actor Action	System Response		
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].		
3. Technical Manager clicks on [PRODUCT] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Product Category Page] with [Category List] table		
5. Technical Manager clicks on a row in body of [Category List] table	6. Show [Category Detail] form with detail information of clicked category includes: <ul style="list-style-type: none"> - Category Name: textbox - Default cost: textbox 		

	<ul style="list-style-type: none"> - Active - CLOSE: button [Alternative 1] - UPDATE: button 	
7. Technical Manager make some changes for his purpose by edit text fields	8. [Exception 1, 3]	
9. Click on [UPDATE] button to finish this use case.	10. Edited information is updated to database and changes of that edited.	

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish this use case.	2. Nothing changes and [Category Detail] form is hidden.

Exceptions:

Actor Action	System Response
3. Enter invalid value to [Default cost] textbox	4. Display message: “Please enter a number”.

Relationships:

Manage Product use case, Manage Product Category use case, View All Product Category list use case.

Business Rules:

- Technical Manager wants to update information of existed category.
- Technical Manager clicks on category row in [Category List] table to update.
- Edited information is updated to database and changes are update in [Category List] table.

2.2.53. < Technical Manager > [Manage Product] Add Product Prototype

2.2.53.1. Use Case Diagram

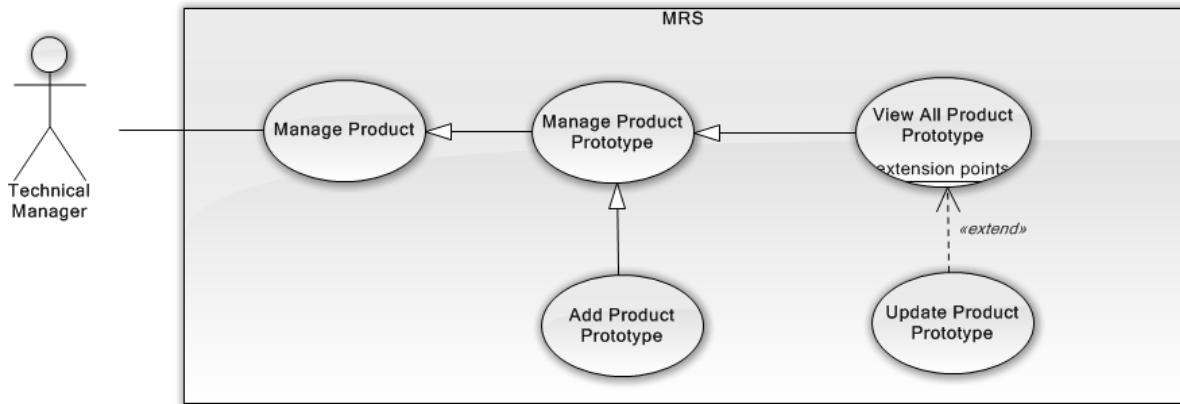


Figure 68-Manage Product Prototype use case diagram

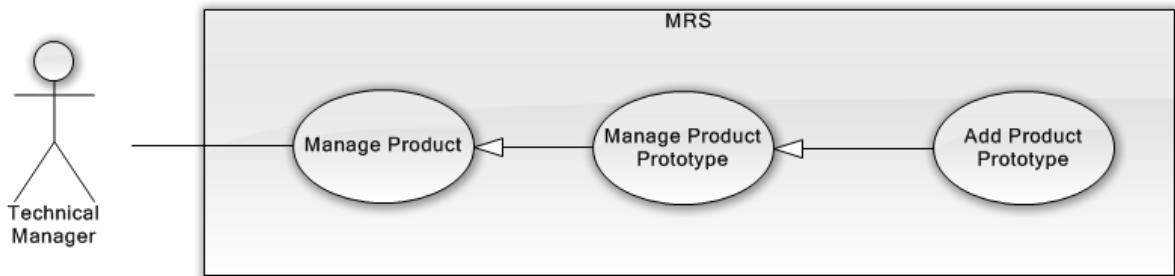


Figure 69-Manger Add Product Prototype usecase digram

2.2.53.2. Use Case Specification

USE CASE – Add Product Prototype SPECIFICATION			
Use case No.	UC053	Use case version	2.0
Use case name	Add Product Prototype		
Author	KieuDTD		
Date	8/8/2013	Priority	Normal
Actor: Technical Manager			
Summary: Technical manager uses this use case to create new product prototype to the system. When create new product prototype successfully, it will store in database and Technical Manager can view to manage later.			
Goal:			

Store new product prototype to database for later management

Triggers:

In [Manage Product Prototype Page], Technical Manager can create new product prototype by clicks on [ADD PROTOTYPE] button; [Add Prototype] form will be shown with necessary information of product prototype to fill.

Preconditions:

User logged in successfully with role of Technical Manager.

Post Conditions:

- **Success:** New product prototype will be stored in database and new row of product prototype will be inserted to [Prototype List] table.
- **Fail:** Redirect to [Error Page] or still at current page with error messages

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [PRODUCT] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Product Category Page]
5. Technical Manager clicks on [Product Prototype] link in [Manage Product Category Page].	6. Redirect to [Manage Product Prototype Page] with [Prototype List] table
7. Technical Manager clicks on [ADD PROTOTYPE] button.	8. Show [Add Prototype] form includes: <ul style="list-style-type: none"> - Prototype Name: textbox (required, max length: 100) - Product Category : Dropdown List - Description: Text area - CLOSE: button [Alternative 1] - ADD: button
9. Enter information into text fields.	10. [Exception 1]
11. Click on [ADD] button to finish this use case.	12. New Product prototype will be stored in database and new row will be inserted to [Prototype List] table.

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish	2. All information in [Add Prototype] form

this use case.	is cleared.
----------------	-------------

Exceptions:

Actor Action	System Response
1. [Product Prototype], [Description] are blank	2. Display message: "Please fill out this field." on the fields that is blank.

Relationships:

Manage Product use case, Manage Product Prototype use case.

Business Rules:

- Technical Manager wants to store new product prototype to database for later use.
- Technical Manager fills necessary information about new prototype into [Add Prototype] form and click [ADD] button to finish.
- New product prototype is stored in database and new row is inserted to [Prototype List] table.
- Technical manager can choose the maximum item on list paging is 10, 15, 25, etc.

2.2.54. < Technical Manager > [Manage Product] View All Product Prototype List

2.2.54.1. Use Case Diagram

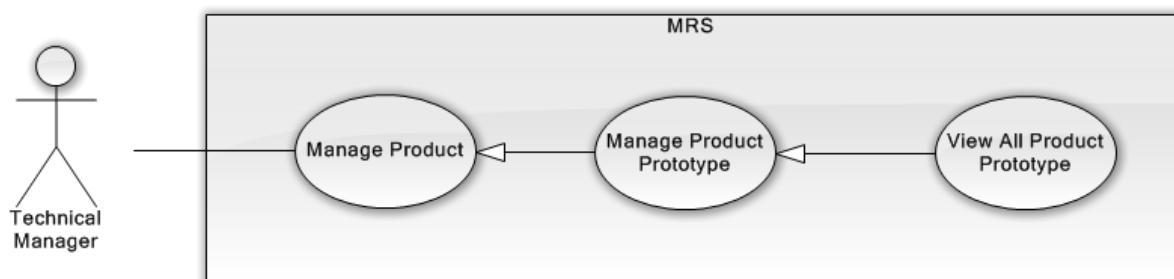


Figure 70-Manager View All Product Prototype usecase diagram

2.2.54.2. Use Case Specification

USE CASE – View All Product Prototype List SPECIFICATION			
Use case No.	UC054	Use case version	2.0
Use case name	View All Product Prototype List		
Author	KieuDTD		
Date	8/8/2013	Priority	Normal
Actor:			

Technical Manager

Summary:

Technical Manager uses this case to tracking all product prototypes in the system.

Goal:

Technical manager can view all product prototypes in list with table format, which have some information such as prototype name, category, avg repaired cost....

Triggers:

Technical manager can view all list product prototypes by clicks on [Product Prototype] link in [Manage Product Category Page]. [Manage Product Prototype Page] will be shown with [Prototype List] table.

Preconditions:

User logged in successfully with role of Technical Manager.

Post Conditions:

- **Success:** List all product prototypes in the system.
- **Fail:** N/A

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [PRODUCT] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Product Category Page]
5. Technical Manager clicks on [Product Prototype] link in [Manage Product Category Page].	<p>6. Redirect to [Manage Product Prototype Page] with [Prototype List] table</p> <p>List all product prototype in [Prototype List] table with columns:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Prototype Name: Name of prototype - Description - Activation: Disable or enable product prototype

Alternative Scenario:

N/A

Exceptions:

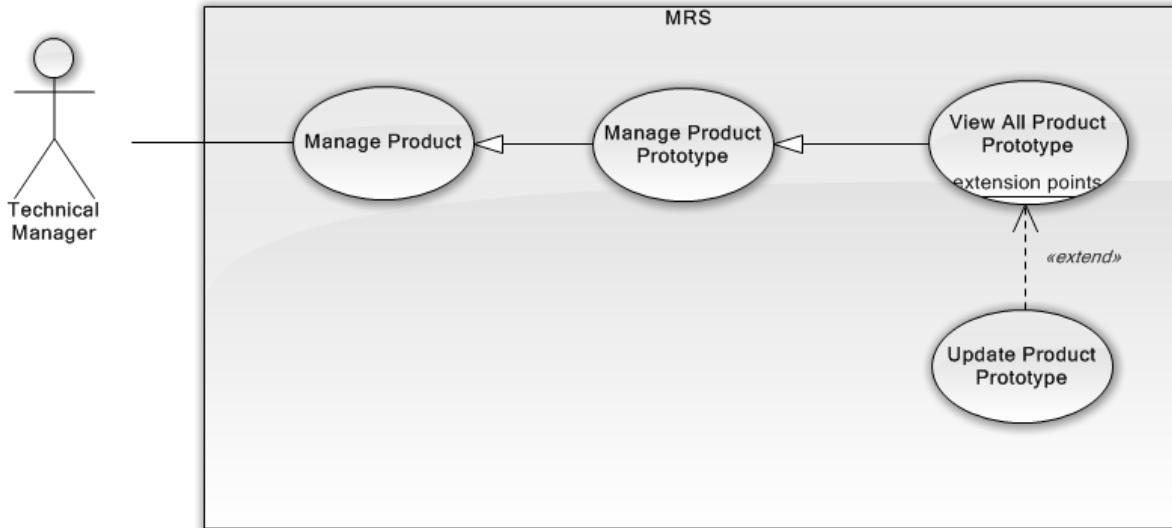
N/A

Relationships:

Manage Product use case, Manage Product prototype use case.

Business Rules:

- Technical Manager wants to view all product prototypes.
- All prototypes will display with order from latest to oldest updated date.
- Technical manager can choose the maximum item on list paging is 10, 15, 25, etc.

2.2.55. < Technical Manager > [Manage Product] Update Product Prototype**2.2.55.1. Use Case Diagram****Figure 71-Manger Update Product Prototype usecase diagram****2.2.55.2. Use Case Specification**

USE CASE – Update Product Prototype SPECIFICATION			
Use case No.	UC055	Use case version	2.0
Use case name	Update Product Prototype		
Author	KieuDTD		
Date	8/8/2013	Priority	Normal
Actor: Technical Manager			
Summary: For a reason, Technical Manager want to update content of product prototype that existed, this use case can help them to do it easily.			
Goal: Update existed prototype information in database such as average repaired cost, active prototype or not			
Triggers:			

In [Manage Product Prototype Page], Technical Manager can update prototype by clicks on that prototype in [Prototype List] table. [Prototype Detail] form will be shown with detail information of that product prototype and from now, Technical Manager can edit information and update to database.

Preconditions:

User logged in successfully with role of Technical Manager and Technical Manager click on a row of prototype in the body of [Prototype List] table that they wants to update (except table header row).

Post Conditions:

- **Success:** Edited information is updated to database and changes of that edited prototype are update in [Prototype List] table.
- **Fail:** Redirect to [Error Page] or still at current page with error messages

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [PRODUCT] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Product Category Page]
5. Technical Manager clicks on [Product Prototype] link in [Manage Product Category Page].	6. Redirect to [Manage Product Prototype Page] with [Prototype List] table
7. Technical Manager clicks on a row in body of [Prototype List] table	8. Show [Prototype Detail] form with detail information of clicked prototype includes: <ul style="list-style-type: none"> - Prototype Name: textbox - Category: Dropdown list - Description - Active - CLOSE: button [Alternative 1] - UPDATE: button
9. Technical Manager make some changes for his purpose by edit text fields	10. [Exception 1]

Alternative Scenario:

Actor Action	System Response

1. Click on [CLOSE] button to finish this use case.	2. Nothing changes and [Prototype Detail] form is hidden.
---	---

Exceptions:

Actor Action	System Response
1. [Description] are blank	2. Display message: "Please fill out this field." on the fields that is blank.

Relationships:

Manage Product use case, Manage Product Prototype use case, View All Product Prototype list use case.

Business Rules:

- Technical Manager wants to update information of existed prototype.
- Technical Manager clicks on prototype row in [Prototype List] table to update.
- Edited information is updated to database and changes are update in [Prototype List] table.

2.2.56. < Technical Manager > [Manage Product] Add Product Model

2.2.56.1. Use Case Diagram

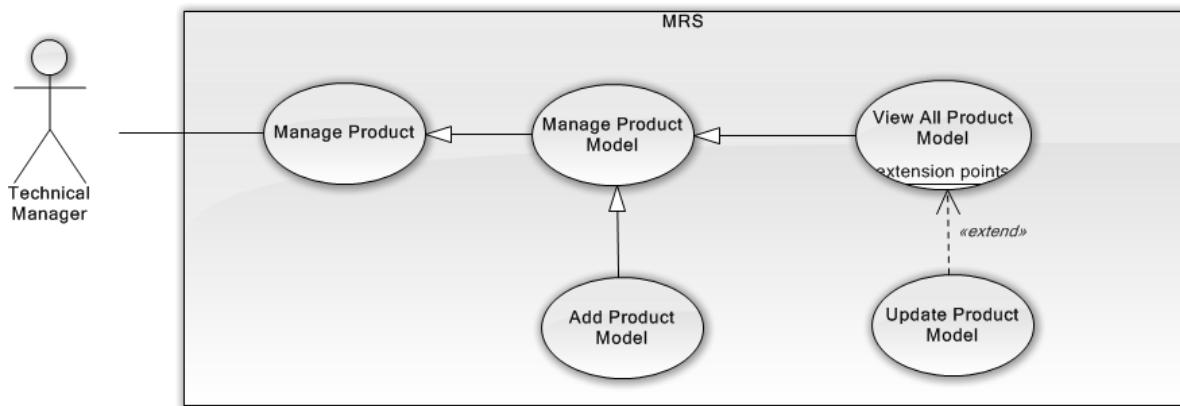


Figure 72-Manage Product Model use case diagram

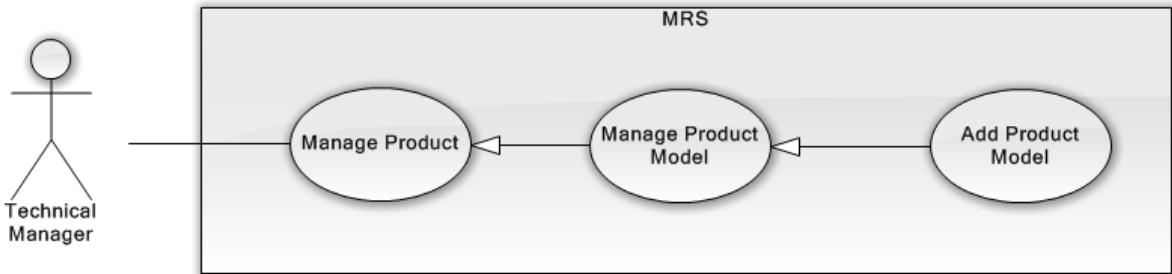


Figure 73-Manger Add Product Model usecase diagram

2.2.56.2. Use Case Specification

USE CASE – Add Product Model SPECIFICATION						
Use case No.	UC056	Use case version	2.0			
Use case name	Add Product Model					
Author	KieuDTD					
Date	8/8/2013	Priority	Normal			
Actor: Technical Manager						
Summary: Technical manager uses this use case to create new product model to the system. When create new product model successfully, it will store in database and Technical Manager can view to manage later.						
Goal: Store new product model to database for later management						
Triggers: In [Manage Product Model Page], Technical Manager can create new product model by clicks on [ADD MODEL] button; [Add Model] form will be shown with necessary information of product model to fill.						
Preconditions: User logged in successfully with role of Technical Manager.						
Post Conditions: <ul style="list-style-type: none"> - Success: New product model will be stored in database and new row of product model will be inserted to [Model List] table. - Fail: Redirect to [Error Page] or still at current page with error messages 						
Main Success Scenario:						
<table border="1"> <thead> <tr> <th>Actor Action</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1. Technical Manager clicks on [Request] box on [Home page].</td> <td>2. Redirect to [Manage Repair Maintenance Page].</td> </tr> </tbody> </table>			Actor Action	System Response	1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
Actor Action	System Response					
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].					

3. Technical Manager clicks on [PRODUCT] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Product Category Page]	
5. Technical Manager clicks on [Product Model] link in [Manage Product Category Page].	6. Redirect to [Manage Product Model Page] with [Model List] table	
7. Technical Manager clicks on [ADD MODEL] button.	8. Show [Add Model] form includes: <ul style="list-style-type: none">- Model Name: textbox (required, max length: 100)- Product Category : Dropdown List- Product Prototype: Dropdown list- Description: Text area- CLOSE: button [Alternative 1]- ADD: button	
9. Enter information into text fields.	10. [Exception 1, 3]	
11. Click on [ADD] button to finish this use case.	12. New Product model will be stored in database and new row will be inserted to [Model List] table.	

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish this use case.	2. All information in [Add Model] form is cleared.

Exceptions:

Actor Action	System Response
1. [Product Model], [Description] are blank	2. Display message: "Please fill out this field." on the fields that is blank.

Relationships:

Manage Product use case, Manage Product Model use case.

Business Rules:

- Technical Manager wants to store new product model to database for later use.
- Technical Manager fills necessary information about new model into [Add Model] form and click [ADD] button to finish.

- New product model is stored in database and new row is inserted to [Model List] table.
- Technical manager can choose the maximum item on list paging is 10, 15, 25, etc.

2.2.57. < Technical Manager > [Manage Product] View All Product Model List

2.2.57.1. Use Case Diagram

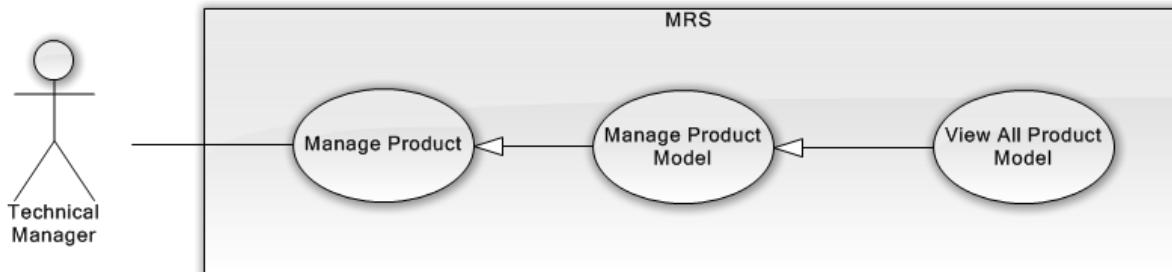


Figure 74-Manger View All Product Model

2.2.57.2. Use Case Specification

USE CASE – View All Product Model List SPECIFICATION			
Use case No.	UC057	Use case version	2.0
Use case name	View All Product Model List		
Author	KieuDTD		
Date	8/8/2013	Priority	Normal
Actor: Technical Manager			
Summary: Technical Manager uses this case to tracking all product models in the system.			
Goal: Technical manager can view all product models in list with table format, which have some information such as model name, category, product brand....			
Triggers: Technical manager can view all list product models by clicks on [Product Model] link in [Manage Product Category Page]. [Manage Product Model Page] will be shown with [Model List] table.			
Preconditions: User logged in successfully with role of Technical Manager.			
Post Conditions: <ul style="list-style-type: none"> - Success: List all product models in the system. - Fail: N/A 			
Main Success Scenario:			

Actor Action	System Response	
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page]	
3. Technical Manager clicks on [PRODUCT] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Product Category Page]	
5. Technical Manager clicks on [Product Model] link in [Manage Product Category Page].	<p>6. Redirect to [Manage Product Model Page] with [Model List] table</p> <p>List all product model in [Model List] table with columns:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Model Name: Name of model - Category - Product Brand - Activation: Disable or enable product model 	

Alternative Scenario:
N/A

Exceptions:
N/A

Relationships:
Manage Product use case, Manage Product Model use case.

Business Rules:

- Technical Manager wants to view all product models.
- All models will display with order from latest to oldest updated date.
- Technical manager can choose the maximum item on list paging is 10, 15, 25, etc.

2.2.58. < Technical Manager > [Manage Product] Update Product Model

2.2.58.1. Use Case Diagram

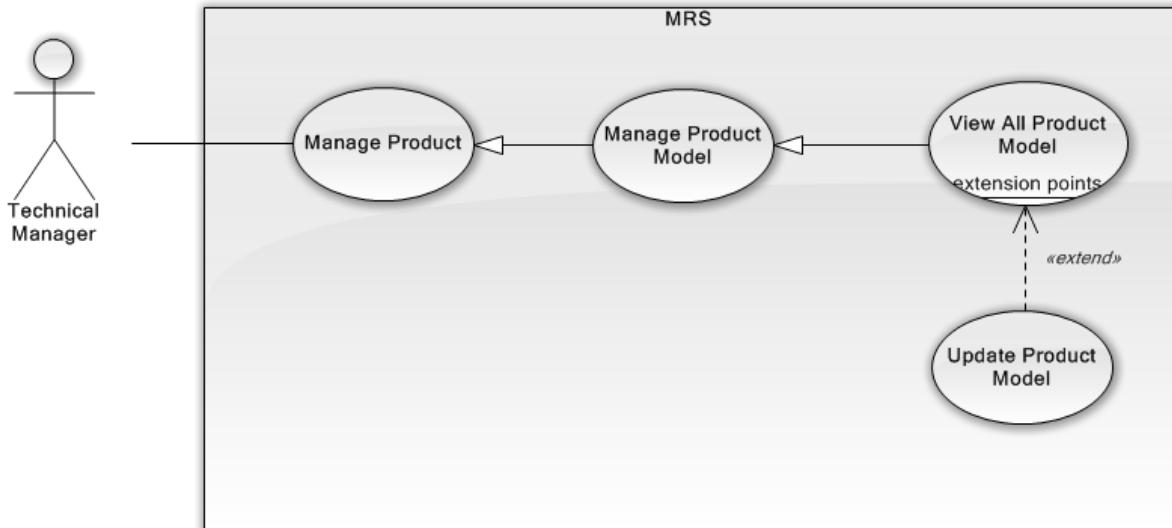


Figure 75-Manger Update Product Model

2.2.58.2. Use Case Specification

USE CASE – Update Product Model SPECIFICATION			
Use case No.	UC058	Use case version	2.0
Use case name	Update Product Model		
Author	KieuDTD		
Date	8/8/2013	Priority	Normal
Actor: Technical Manager			
Summary: For a reason, Technical Manager want to update content of product model that existed, this use case can help them to do it easily.			
Goal: Update existed model information in database such as description, active model or not			
Triggers: In [Manage Product Model Page], Technical Manager can update model by clicks on that model in [Model List] table. [Model Detail] form will be shown with detail information of that product model and from now, Technical Manager can edit information and update to database.			
Preconditions: User logged in successfully with role of Technical Manager and Technical Manager click on a row of model in the body of [Model List] table that they wants to update (except table header row).			

Post Conditions:

- **Success:** Edited information is updated to database and changes of that edited model are update in [Model List] table.
- **Fail:** Redirect to [Error Page] or still at current page with error messages

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [PRODUCT] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Product Category Page]
5. Technical Manager clicks on [Product Model] link in [Manage Product Category Page].	6. Redirect to [Manage Product Model Page] with [Model List] table
7. Technical Manager clicks on a row in body of [Model List] table	8. Show [Model Detail] form with detail information of clicked model includes: <ul style="list-style-type: none"> - Model Name: textbox - Product Category: Dropdown list - Product Prototype: Dropdown list - Description - Active - CLOSE: button [Alternative 1] - UPDATE: button
9. Technical Manager make some changes for his purpose by edit text fields	10. [Exception 1, 3]
11. Click on [UPDATE] button to finish this use case.	12. Edited information is updated to database and changes of that edited.

Alternative Scenario:

Actor Action	System Response
1. Click on [CLOSE] button to finish this use case.	2. Nothing changes and [Model Detail] form is hidden.

Exceptions:

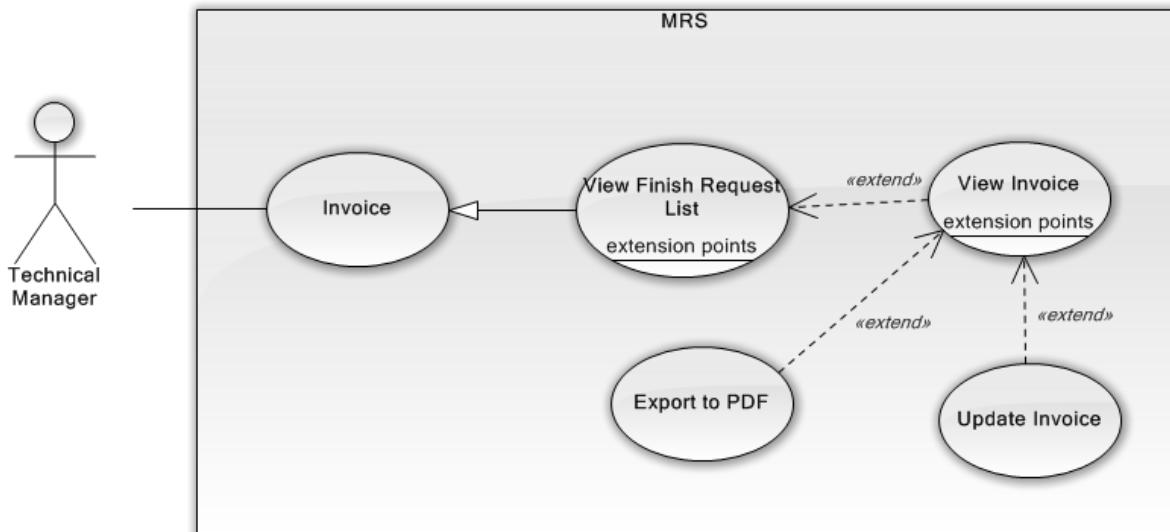
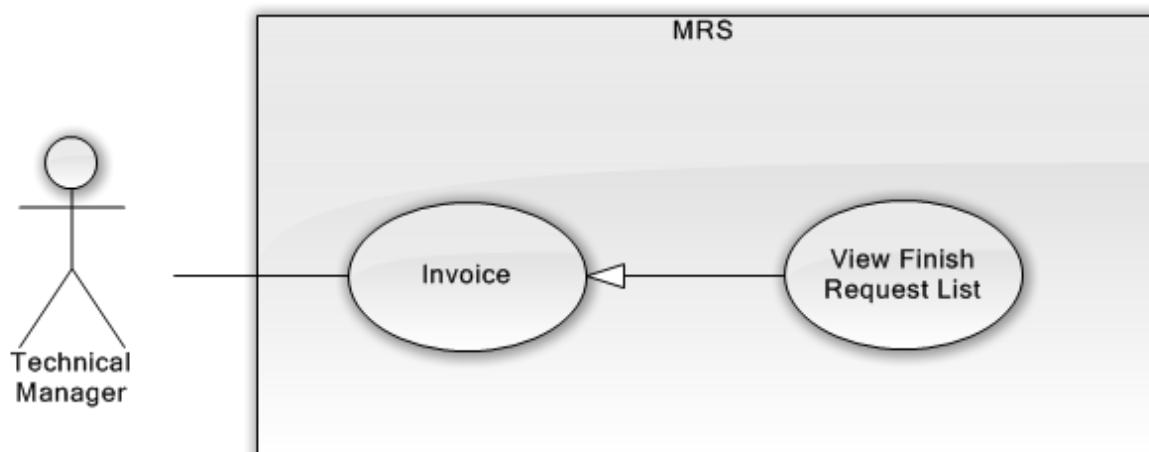
N/A

Relationships:

Manage Product use case, Manage Product Model use case, View All Product Model list use case.

Business Rules:

- Technical Manager wants to update information of existed model.
- Technical Manager clicks on model row in [Model List] table to update.
- Edited information is updated to database and changes are update in [Model List] table.

2.2.59. < Technical Manager > [Manage Invoice] View Finish Request List**2.2.59.1. Use Case Diagram****Figure 76-Manage Invoice use case diagram****Figure 77-Manger View Finish Request List usecase diagram**

2.2.59.2. *Use Case Specification*

USE CASE – View Finish Request List SPECIFICATION								
Use case No.	UC059	Use case version	2.0					
Use case name	View Finish Request List							
Author	KieuDTD							
Date	8/8/2013	Priority	Normal					
Actor: Technical Manager								
Summary: When the process of maintenance/repairing finish, one of these final task is export invoice. Export invoice support only finished or cancelled request. With this function, technical manager can view all finished and cancelled request.								
Goal: Technical manager can view all requests that finished and cancelled.								
Triggers: Technical manager can view all list of finished request by clicks on [INVOICE] tab in [Manage Repair Maintenance Page]. [Manage Invoice Page] will be shown with [Request List] table.								
Preconditions: User logged in successfully with role of Technical Manager.								
Post Conditions: <ul style="list-style-type: none">- Success: List all finished and cancelled request in [Request List] table.- Fail: N/A								
Main Success Scenario:								
<table border="1"> <thead> <tr> <th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1. Technical Manager clicks on [Request] box on [Home page].</td><td>2. Redirect to [Manage Repair Maintenance Page]</td></tr> <tr> <td>3. Technical Manager clicks on [INVOICE] tab in [Manage Repair Maintenance Page].</td><td>4. Redirect to [Manage Invoice Page] with [Request List] table includes:<ul style="list-style-type: none">- No.: Ordinal number- Code: Request code- Name: Name of request- Status : Status of request- Create Date- Finish Date</td></tr> </tbody> </table>			Actor Action	System Response	1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page]	3. Technical Manager clicks on [INVOICE] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Invoice Page] with [Request List] table includes: <ul style="list-style-type: none">- No.: Ordinal number- Code: Request code- Name: Name of request- Status : Status of request- Create Date- Finish Date
Actor Action	System Response							
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page]							
3. Technical Manager clicks on [INVOICE] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Invoice Page] with [Request List] table includes: <ul style="list-style-type: none">- No.: Ordinal number- Code: Request code- Name: Name of request- Status : Status of request- Create Date- Finish Date							
Alternative Scenario:								

N/A

Exceptions:

N/A

Relationships:

Manage Invoice use case.

Business Rules:

- Technical Manager wants to view all finished and cancelled request to export invoice.
- All requests will display with order from latest to oldest finish date.
- Technical manager can choose the maximum item on list paging is 10, 15, 25, etc.

2.2.60. < Technical Manager > [Manage Invoice] View Invoice

2.2.60.1. Use Case Diagram

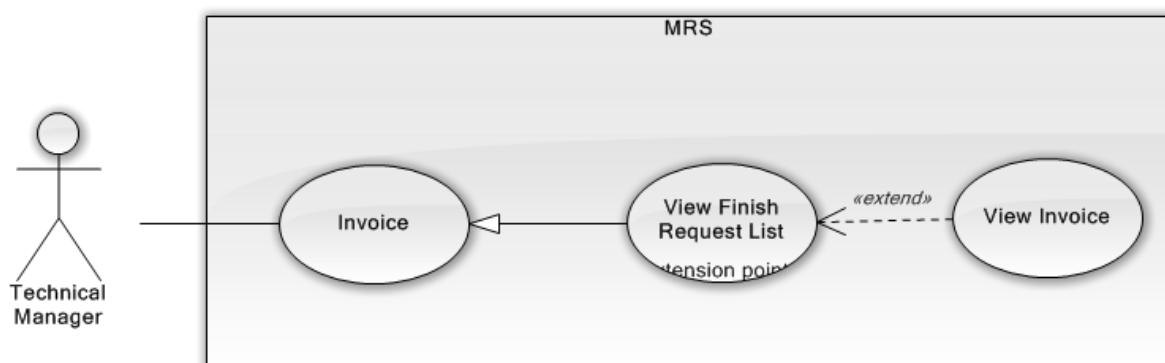


Figure 78-Manger View Invoice usecase diagram

2.2.60.2. Use Case Specification

USE CASE – View Invoice SPECIFICATION			
Use case No.	UC060	Use case version	2.0
Use case name	View Invoice		
Author	KieuDTD		
Date	8/8/2013	Priority	Normal
Actor:			
Technical Manager			
Summary:			
View invoice is view detail of finished or cancelled request to check data before export invoice. By using this function, technical manager could view request in detail.			
Goal:			
Technical manager can view all details of invoice.			
Triggers:			
In [Manage Invoice Page], Technical manager clicks on a row of [Request List] table,			

[Request Detail] form will be shown on pop up.

Preconditions:

User logged in successfully with role of Technical Manager.

Post Conditions:

- **Success:** Detail of selected request for invoice will be shown in [Request Detail] form on pop up.
- **Fail:** N/A

Main Success Scenario:

Actor Action	System Response										
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page]										
3. Technical Manager clicks on [INVOICE] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Invoice Page] with [Request List] table										
4. Technical manager click on a row of [Request List] table	<p>5. Detail of selected request for invoice will be shown in [Request Detail] form includes:</p> <table border="1"> <tr> <td>Basic information</td><td> <ul style="list-style-type: none"> - Request Name: textbox (max length: 100) - Request type: dropdown list - Estimate Finish: Calendar </td></tr> <tr> <td>Customer Related</td><td> <ul style="list-style-type: none"> - Customer relate: dropdown list(customer request to maintain-repair) </td></tr> <tr> <td>Product information</td><td> <ul style="list-style-type: none"> - Prototype: dropdown list - Model: dropdown list </td></tr> <tr> <td>Estimate replace equipment</td><td> <ul style="list-style-type: none"> - Add equipment button. Click on this button then appear: - Category: dropdown list - Equipment: dropdown list - [quantity]: dropdown list </td></tr> <tr> <td>Request</td><td> <ul style="list-style-type: none"> - Symptom: select </td></tr> </table>	Basic information	<ul style="list-style-type: none"> - Request Name: textbox (max length: 100) - Request type: dropdown list - Estimate Finish: Calendar 	Customer Related	<ul style="list-style-type: none"> - Customer relate: dropdown list(customer request to maintain-repair) 	Product information	<ul style="list-style-type: none"> - Prototype: dropdown list - Model: dropdown list 	Estimate replace equipment	<ul style="list-style-type: none"> - Add equipment button. Click on this button then appear: - Category: dropdown list - Equipment: dropdown list - [quantity]: dropdown list 	Request	<ul style="list-style-type: none"> - Symptom: select
Basic information	<ul style="list-style-type: none"> - Request Name: textbox (max length: 100) - Request type: dropdown list - Estimate Finish: Calendar 										
Customer Related	<ul style="list-style-type: none"> - Customer relate: dropdown list(customer request to maintain-repair) 										
Product information	<ul style="list-style-type: none"> - Prototype: dropdown list - Model: dropdown list 										
Estimate replace equipment	<ul style="list-style-type: none"> - Add equipment button. Click on this button then appear: - Category: dropdown list - Equipment: dropdown list - [quantity]: dropdown list 										
Request	<ul style="list-style-type: none"> - Symptom: select 										

	information	<ul style="list-style-type: none"> - symptom of request - Problem description: text area - Customer confirmation: text area - Request tags: request from customer to decided tags appropriate for assign task. 	
	Button	-	

Alternative Scenario:
N/A

Exceptions:
N/A

Relationships:
Manage Invoice use case, View Finish Request List.

Business Rules:

- Technical Manager uses this function to view request detail to export invoice.

2.2.61. < Technical Manager > [Manage Invoice] Update Invoice

2.2.61.1. Use Case Diagram

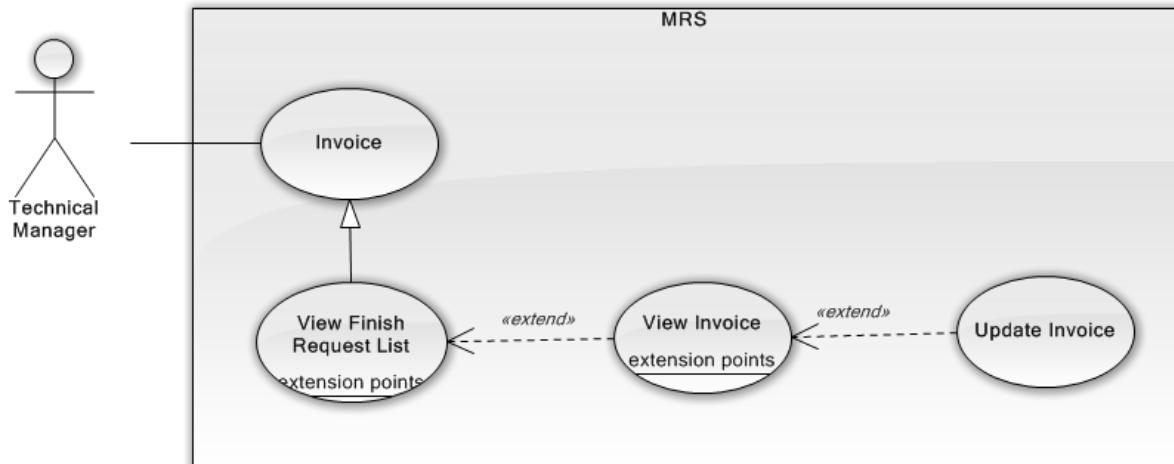


Figure 79-Update Invoice usecase diagram

2.2.61.2. *Use Case Specification***USE CASE – Update Invoice SPECIFICATION**

Use case No.	UC061	Use case version	2.0
Use case name	Update Invoice		
Author	HungDD		
Date	12/8/2013	Priority	Normal

Actor:

Technical Manager

Summary:

Update invoice information such as labor fee, discount amount...

Goal:

Update invoice information before export to customer.

Triggers:

Manager click [Save] button to update invoice

Preconditions:

- User logged in successfully with role of Technical Manager.
- Request is Finished or Cancelled

Post Conditions:

- **Success:** Detail of selected request for invoice will be shown in [Request Detail] form on pop up.
- **Fail:** N/A

Main Success Scenario:

Actor Action	System Response		
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page]		
3. Technical Manager clicks on [INVOICE] tab in [Manage Repair Maintenance Page].	4. Redirect to [Manage Invoice Page] with [Request List] table		
4. Technical manager click on a row of [Request List] table	<p>5. Detail of selected request for invoice will be shown in [Request Detail] form includes:</p> <table border="1"> <tr> <td>Basic information</td> <td> <ul style="list-style-type: none"> - Request Name: textbox (max length: 100) - Request type: dropdown list - Estimate Finish: </td> </tr> </table>	Basic information	<ul style="list-style-type: none"> - Request Name: textbox (max length: 100) - Request type: dropdown list - Estimate Finish:
Basic information	<ul style="list-style-type: none"> - Request Name: textbox (max length: 100) - Request type: dropdown list - Estimate Finish: 		

		Calendar	
	Customer Related	- Customer relate: dropdown list(customer request to maintain-repair)	
	Product information	- Prototype: dropdown list - Model: dropdown list	
	Estimate replace equipment	- Add equipment button. Click on this button then appear: - Category: dropdown list - Equipment: dropdown list - [quantity]: dropdown list	
	Request information	- Symptom: select symptom of request - Problem description: text area - Customer confirmation: text area - Request tags: request from customer to decided tags appropriate for assign task.	
	Button	-	

Alternative Scenario:
N/A

Exceptions:
N/A

Relationships:
Manage Invoice use case, View Finish Request List.

Business Rules:

- Technical Manager uses this function to view request detail to export invoice.

2.2.62. < Technical Manager > [Manage Promotion] View All Promotion List

2.2.62.1. Use Case Diagram

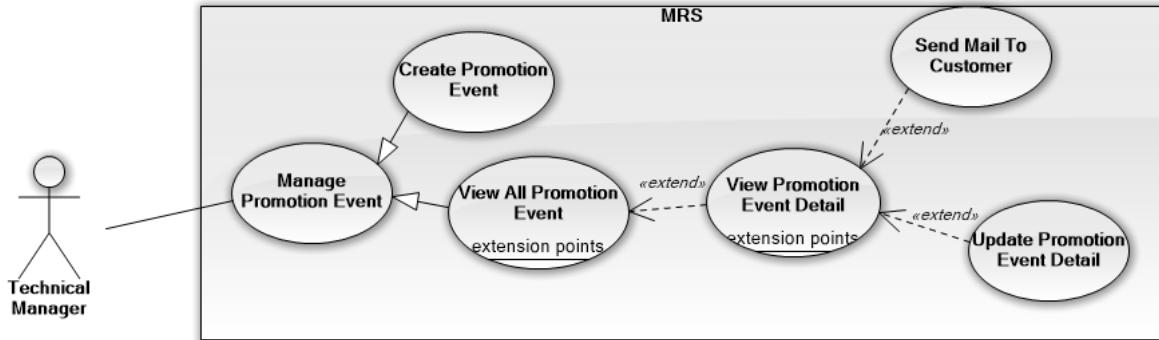


Figure 80-Manger Mange Promotion usecase diagram

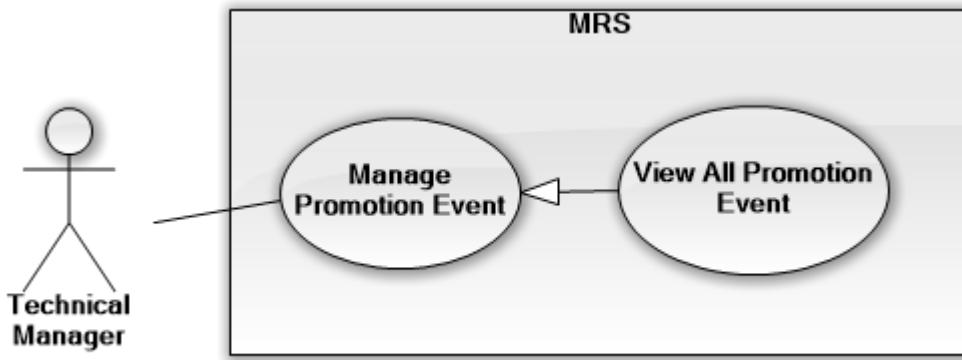


Figure 81-Manger View All Promotion List

2.2.62.2. Use Case Specification

USE CASE – View All Promotion List SPECIFICATION			
Use case No.	UC062	Use case version	1.0
Use case name	View All Promotion List		
Author	ThanhTM		
Date	10/8/2013	Priority	Normal
Actor: Technical Manager			
Summary: Technical manager uses this case to tracking all promotions of MRS center. These promotions are stored in database when MRS has new promotion program and added by Technical Manager.			

Goal:

Technical manager can view all promotions in list with table format, which promotion belongs to which equipment or the time the promotion event start/expire.

Triggers:

Technical manager can view all promotions list by clicks on [PROMOTION] tab in [Manage Repair Maintenance Page]. Then the system display [Manage Promotion Page] with [Promotion List] table

Preconditions:

User logged in successfully with role of Technical Manager.

Post Conditions:

- **Success:** List all promotions of center which had been created.
- **Fail:** N/A

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [PROMOTION] tab [Manage Repair Maintenance Page].	<p>4. Redirect to [Manage Promotion Page].</p> <p>List all promotions which had been created in [Promotion List] table with columns:</p> <ul style="list-style-type: none"> - No.: Ordinal number - Promotion Name: Name of promotion - Content: The content of the promotion - StartDate: The date which the promotion program starts. - ExpireDate: The date which the promotion program ends. - Activation: Disable or enable that promotion.

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Manage Promotion.

Business Rules:

- Technical manager wants to view all program of promotions of center.
- Technical manager can choose the maximum item on list paging is 10, 15, 25, etc.

- All promotions will display in a list with order from latest to oldest created date.

2.2.63. < Technical Manager > [Manage Promotion] Create Promotion

2.2.63.1. Use Case Diagram

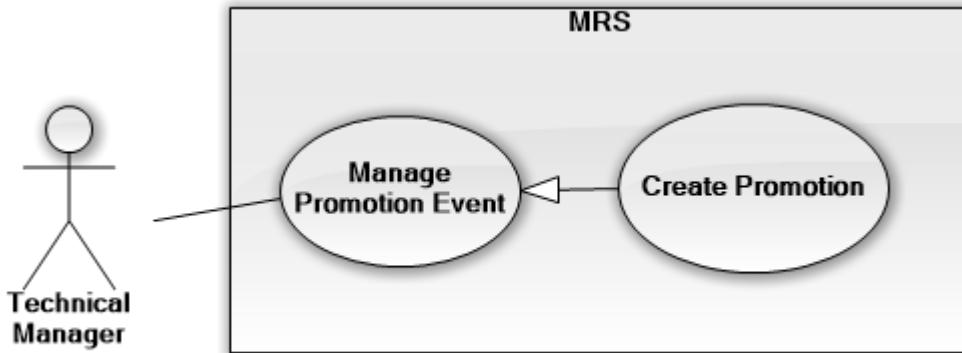


Figure 82-Manager Create Promotion use case diagram

2.2.63.2. Use Case Specification

USE CASE – Create Promotion SPECIFICATION			
Use case No.	UC063	Use case version	1.0
Use case name	Create Promotion		
Author	ThanhTM		
Date	10/08/2013	Priority	Normal
Actor: Technical Manager			
Summary: Technical manager uses this use case to create a new promotion event. When create a new promotion event successfully, it will store in database and Technical Manager can view promotions later.			
Goal: Store a new promotion event to database.			
Triggers: In [Manage Promotion Page], Technical manager can create new promotion event by clicks on [ADD PROMOTION] button, then [Add Promotion] form with necessary information will be shown on pop up.			
Preconditions: User logged in successfully with role of Technical Manager.			
Post Conditions:			

- **Success:** New promotion will be stored in database and new row of promotion will be inserted to [Promotion List] table.
- **Fail:** Redirect to [Error Page] or still at current page with error messages.

Main Success Scenario:

Actor Action	System Response
1. Technical manager clicks on [Request] box on [Home] page.	2. Display [Manage Repair Maintenance] Page
3. Technical Manager clicks on [PROMOTION] tab in [Manage Repair Maintenance] Page	4. Redirect to [Manage Promotion Page] with table [Promotion List]
5. Technical Manager clicks on [Add Promotion] button.	<p>6. Show [Add Promotion] form includes:</p> <ul style="list-style-type: none"> - Promotion Name: textbox (required, max length: 150) - Thumbnail image: image for promotion event. - Start date: date (The date the event starts). - Finish date: date (The date the event ends). - Content: Text area (Content of the promotion event) - All Equipment: checkbox (Select all equipment to be affected by this promotion) - SELECT EQUIPMENT: button (Add equipment relate to this promotion event) [Alternative 1, 3, 5] - ADD: button - CLOSE: button [Alternative 7]
7. Enter information into text fields.	8. [Exception 1, 3, 5, 7]
9. Click on [ADD] button to finish this use case.	10. New promotion event are stored in database and new row is inserted to [Promotion List] table with added promotion information.

Alternative Scenario:

Actor Action	System Response
1. Click on [SELECT EQUIPMENT CATEGORY] button.	2. Display [Equipment] popup include information below: <ul style="list-style-type: none"> - Equipment category: Dropdown list (The equipment category relate to this promotion event) - Select all: checkbox (Select all equipment of selected Equipment Category) - Equipment: text field with dropdown list (The equipment relate to this promotion event) - Discount percent: number (The discount percent of this promotion event, relate to equipment category or each equipment) - SAVE: button [Alternative 3] - CLOSE: button [Alternative 5]
3. Click on [SAVE] button.	4. Temporary save selected equipment and close [Equipment] popup.
5. Click on [CLOSE] button.	6. Close [Equipment] popup.
7. Click on [CLOSE] button to finish adding promotion.	8. All information in [Add Promotion] form is cleared

Exceptions:

Actor Action	System Response
1. [Promotion Name] textbox, [Start Date] and [Expire Date] date time field are blank	2. Display message: “Please fill out this field.” on the fields that is blank.
3. [Content] text area is blank	4. Display error bar with message: “Please fill out the content of promotion.”
5. [Start Date], [Expire Date] are earlier than today	6. Display message: “Start date can’t be earlier than today”
7. [Expire Date] is earlier than [Start	8. Display message: “Expire date must be

Date]	greater than Start date".
-------	---------------------------

Relationships:
Manage Promotion

Business Rules:

- Technical Manager wants to store a new promotion event to database.
- Technical Manager fills necessary information about new promotion into [Add Promotion] form and submits.
- New promotion is stored in database and new row is inserted to [Promotion List] table.

2.2.64. < Technical Manager > [Manage Promotion] Update Promotion

2.2.64.1. Use Case Diagram

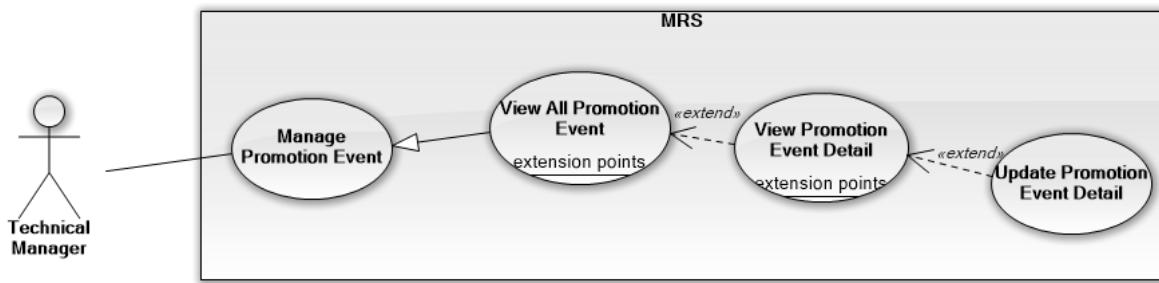


Figure 83-Manger Update Promotion usecase diagram

2.2.64.2. Use Case Specification

USE CASE – Update Promotion SPECIFICATION			
Use case No.	UC064	Use case version	1.0
Use case name	Update Promotion		
Author	ThanhTM		
Date	10/08/2013	Priority	Normal
Actor: Technical Manager			
Summary: For a reason, Technical Manager wants to update content of a promotion that existed; this use case can help him (her) to do it easily.			
Goal: Update existed promotion information in database such as content, equipment relate, start date, finish date, thumbnail image.			
Triggers: In [Manage Promotion Page], Technical Manager can update a promotion by clicks on that			

promotion in [Promotion List] table. [Promotion Detail] form will be shown with detail information of that promotion and from now, Technical Manager can edit information for his purpose and stored edited information to database.

Preconditions:

User logged successfully with role of Technical Manager and Technical Manager clicks on a row of promotion in the body of [Promotion List] table that he (she) wants to update (except table header row).

Post Conditions:

- **Success:** Edited promotion information is updated to database and changes of that edited promotion are updated in [Promotion List] table.
- **Fail:** Redirect to [Error Page] or still at current page with error messages.

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [PROMOTION] tab in [Manage Repair Maintenance Page].	4. Redirect to [Promotion Management Page] with [Promotion List] table
5. Technical Manager clicks on a row in body of [Promotion List] table	<p>6. Show [Promotion Detail] form with detail information of clicked promotion includes:</p> <ul style="list-style-type: none"> - Promotion Name: textbox (required, max length: 150) - Current Thumbnail image: old image of promotion event. - New Thumbnail image: new image for promotion event. - Start date: date (The date the event starts). - Finish date: date (The date the event ends). - Content: Text area (Content of the promotion event) - All Equipment: checkbox (Select all equipment to be affected by this promotion) - SELECT EQUIPMENT: button (Add equipment relate to this promotion event) [Alternative 1, 3,

	<p>5]</p> <ul style="list-style-type: none"> - ADD: button - UPDATE: button [Alternative 7] 	
7. Technical Manager makes some changes for his purpose by edit text fields	8. [Exception 1, 3, 5, 7, 9]	
9. Click on [UPDATE] button to finish this use case.	10. Edited promotion information is updated to database and changes of that edited promotion are update in [Promotion List] table.	

Alternative Scenario:

Actor Action	System Response
1. Click on [SELECT EQUIPMENT CATEGORY] button.	<p>2. Display [Equipment] popup include information below:</p> <ul style="list-style-type: none"> - Equipment category: Dropdown list (The equipment category relate to this promotion event) - Select all: checkbox (Select all equipment of selected Equipment Category) - Equipment: text field with dropdown list (The equipment relate to this promotion event) - Discount percent: number (The discount percent of this promotion event, relate to equipment category or each equipment) - SAVE: button [Alternative 3] - CLOSE: button [Alternative 5]
3. Click on [SAVE] button.	4. Temporary save selected equipment and close [Equipment] popup.
5. Click on [CLOSE] button.	6. Close [Equipment] popup.
7. Click on [CLOSE] button.	8. All information in [Update Promotion] form is cleared. Close [Update Promotion]

	form.
--	-------

Exceptions:

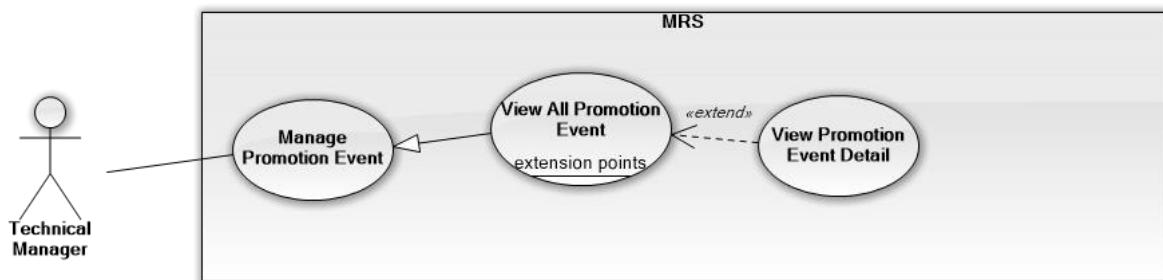
Actor Action	System Response
1. [Start Date] and [Expire Date] date time field are blank	2. Display message: "Please fill out this field." on the fields that is blank.
3. [Content] text area is blank	4. Display error bar with message: "Please fill out the content of promotion."
5. [Start Date], [Expire Date] are earlier than today	6. Display message: "Start date can't be earlier than today"
7. [Expire Date] is earlier than [Start Date]	8. Display message: "Expire date must be greater than Start date".

Relationships:

Manage Promotion

Business Rules:

- Technical Manager wants to update information of existed promotion.
- Technical Manager clicks on a promotion row in [Promotion List] table to update.
- Edited promotion information is updated to database and changes are update in [Promotion List] table.

2.2.65. < Technical Manager > [Manage Promotion] View Promotion Detail**2.2.65.1. Use Case Diagram****Figure 84-Manger View Promotion Detail usecase diagram****2.2.65.2. Use Case Specification****USE CASE – View Promotion Detail SPECIFICATION**

Use case No.	UC065	Use case version	1.0
Use case name	View Promotion Detail		
Author	ThanhTM		
Date	10/08/2013	Priority	Normal

Actor:

Technical Manager

Summary:

Technical Manager wants to view detail information about one promotion, he (she) just selects a promotion in [Promotion List] table and views more detail about that promotion.

Goal:

Technical Manager has a clear detail about a promotion that they want to focus. Additional, this use case can extend to help Technical Manager update a promotion easily which will be described more detail in [Update Promotion] use case below.

Triggers:

In [Manage Promotion Page], Technical manager can view detail of a promotion by clicks on that promotion in [Promotion List] table.

Preconditions:

User logged in successfully with role of Technical Manager and Technical Manager clicks on a row of promotion in the body of [Promotion List] table (except table header row).

Post Conditions:

- **Success:** Detail information of clicked promotion will be shown in [Promotion Detail] form on pop up.
- **Fail:** N/A

Main Success Scenario:

Actor Action	System Response
1. Technical Manager clicks on [Request] box on [Home page].	2. Redirect to [Manage Repair Maintenance Page].
3. Technical Manager clicks on [PROMOTION] tab in [Manage Repair Maintenance Page].	4. Redirect to [Promotion Management Page] with [Promotion List] table
5. Technical Manager clicks on a row in body of [Promotion List] table to finish this use case.	6. Show [Promotion Detail] form with detail information of clicked promotion includes: <ul style="list-style-type: none"> - Promotion Name: textbox (required, max length: 150) - Current Thumbnail image: current thumbnail image for promotion

	<p>event.</p> <ul style="list-style-type: none"> - Start date: date (The date the event starts). - Finish date: date (The date the event ends). - Content: Text area (Content of the promotion event) - Equipment category: Dropdown list (The equipment category relate to this promotion event) - Equipment: Dropdown list (The equipment relate to this promotion event) - Discount percent: number (The discount percent of this promotion event, relate to equipment category or each equipment) - ADD EQUIPMENT CATEGORY: button (Add equipment category relate to this promotion event) - ADD EQUIPMENT: button (Add equipment relate to this promotion event) - CLOSE: button - UPDATE: button 	
--	--	--

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Manage Promotion

Business Rules:

- Technical Manager wants to view detail information of one promotion.
- Technical Manager clicks on a promotion row in [Promotion List] table and detail information of that promotion will be showed in [Promotion Detail] form on pop up.

2.2.66. < System > Auto Send Mail

2.2.66.1. Use Case Diagram

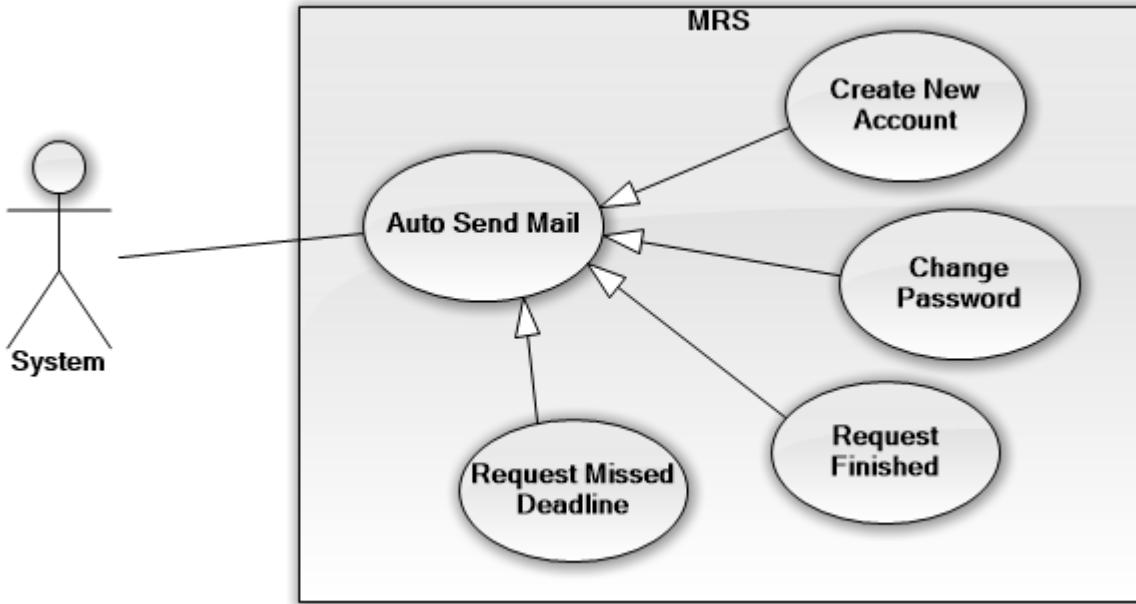


Figure 85-System Auto Send Mail usecase diagram

2.2.66.2. Use Case Specification

USE CASE –Auto Send Mail SPECIFICATION			
Use case No.	UC066	Use case version	2.0
Use case name	Auto Send Mail		
Author	HungDD		
Date	7/7/2013	Priority	Normal
Actor: System			
Summary: To support for security problem and convenient for customer, the website also implement function auto send email. There are have some cases the system would auto send email to customer:			
<ul style="list-style-type: none"> - Technical manager create account for customer to access the website - Customer change their password - When request of customer is finished or missed deadline 			
Goal: The system autos send email to customer with corresponding information.			
Triggers:			

Reference [Manage User] Add new Account use case: On [Add New Account] form in [Account Management] Page, Technical manager fill information and click [ADD] button to finish create account.

Reference [Update Profile] Change Password use case: In [Change Password Page], customer change password and click [CHANGE] button to finish.

System auto check request finished or missed deadline to send email.

Preconditions:

Post Conditions:

- **Success:** The system auto send an email for customer to inform corresponding information.
- **Fail:** N/A

Main Success Scenario:

Reference [Manage User] Add new Account:

Actor Action	System Response
1. In [Account Management Page] Technical Manager fill data about customer information to create new account and click [ADD] button	2. System auto send an email to customer. Email confirms about grant access and password od account.

Reference [Update Profile] Change Password:

Actor Action	System Response
1. In [Change Password Page] Customer change password and click [CHANGE] button	2. System auto send an email to customer. Email confirms about change password success.

Actor Action	System Response
1. System auto check request is finished or missed deadline	2. System auto send an email to customer to inform missed deadline or finish.

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

[Manage User] Add new Account use case, [Update Profile] Change Password use case.

Business Rules:

- Sending an email is automated immediately.

2.2.67. < System > Auto Check Task Missed Deadline

2.2.67.1. Use Case Diagram

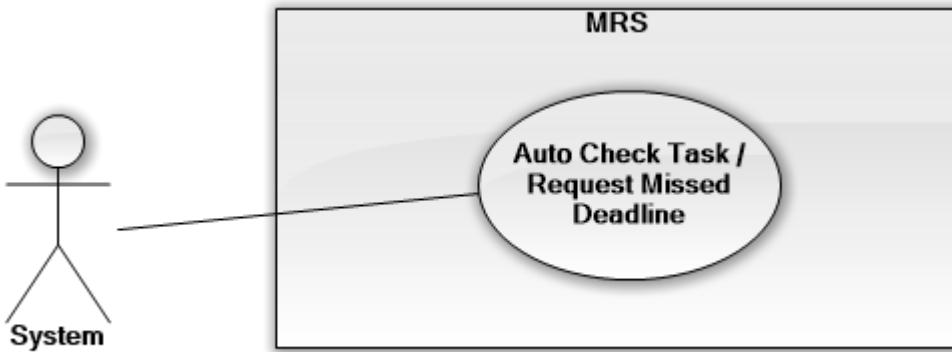


Figure 86-System Auto Check Task/ Request Missed Deadline

2.2.67.2. Use Case Specification

USE CASE –Auto Check Task Missed Deadline SPECIFICATION			
Use case No.	UC067	Use case version	2.0
Use case name	Auto Check Task Missed Deadline		
Author	HungDD		
Date	7/7/2013	Priority	Normal
Actor: System			
Summary: When having more complex request, breaking down task for each request at this time become difficult to control. Technical manager cannot track all of task-missed deadline or not.			
Goal: The system auto checking a task miss deadline or not and highlight this task in [Task List] table.			
Triggers: Checking a task miss deadline or not is automatic.			
Preconditions: N/A			
Post Conditions: <ul style="list-style-type: none"> - Success: The task that miss deadline will be highlight in [Task List] table. - Fail: N/A 			
Main Success Scenario: The system will auto access database to check [Finish Date] value of task.			
Alternative Scenario: N/A			

Exceptions:

N/A

Relationships:

N/A

Business Rules:

- Checking task miss deadline or not is automated by the system in period.

2.2.68. < System > Recommend Staff for Create Task**2.2.68.1. Use Case Diagram****Figure 87-System Recommend Staff For Create Task usecase diagram****2.2.68.2. Use Case Specification****USE CASE –Recommend Staff for Create Task SPECIFICATION**

Use case No.	UC068	Use case version	2.0
Use case name	Recommend Staff for Create Task		
Author	HungDD		
Date	7/7/2013	Priority	Normal

Actor:

System

Summary:

When request of customer was created, Technical manager will assign task for staff. This assignment is difficult with assign for right task. Therefore, this function is a solution to support for assignment task by Recommend staff.

Goal:

The system will recommend staffs that have most matching skills and current task is at least.

Triggers:

At [Add Task] form in [Manage Task] Page, Technical manager click on [Recommend Staff] button, system will display pop up about staff list is sorted according appropriate level.

Preconditions:

N/A

Post Conditions:

- **Success:** Display Staff list is sorted according appropriate level.
- **Fail:** N/A

Main Success Scenario:

Actor Action	System Response
1. At [Add Task] form in [Manage Task] Page, Technical manager click on [Recommend Staff] button	2. Display Staff list is sorted on pop up.

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Create Task use case

Business Rules:

- System automatically selects the best matching staff and displays in [Staff Relate] text box.
- Best staffed are sorted in priority: Most matched skill, least current task

2.2.69. < System > Auto Notification

2.2.69.1. Use Case Diagram

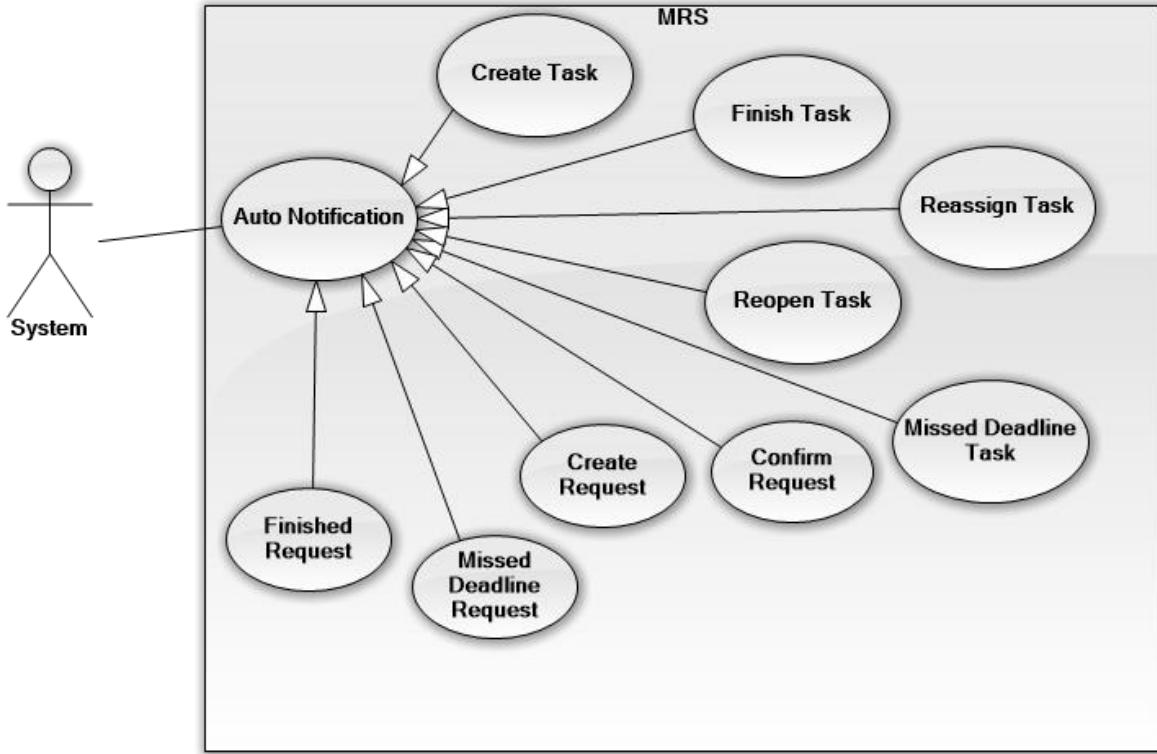


Figure 88-System Auto Notification usecase diagram

2.2.69.2. Use Case Specification

USE CASE –Auto Notification SPECIFICATION			
Use case No.	UC069	Use case version	2.0
Use case name	Auto Notification		
Author	HungDD		
Date	7/7/2013	Priority	Normal
Actor: System			
Summary: When manager create a task or task is missed deadline, finished... then system auto notify for corresponding person.			
Goal: The system will auto notification to corresponding person.			
Triggers: When have some action: Create Task, Finish Task, Reassign Task, Reopen Task, Missed			

Deadline Task, Confirm Request, Create Request, Missed Deadline Request, Finished Request

Preconditions:

N/A

Post Conditions:

- **Success:** New notification is display on header page
- **Fail:** N/A

Main Success Scenario:

Actor Action	System Response
1. When have some action: Create Task, Finish Task, Reassign Task, Reopen Task, Missed Deadline Task, Confirm Request, Create Request, Missed Deadline Request, Finished Request	2. Display notification on the header page of related person.

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

Auto notification usecase

Business Rules:

- System automatically notify to related person.

2.2.70. < System > Auto Send SMS

2.2.70.1. Use Case Diagram

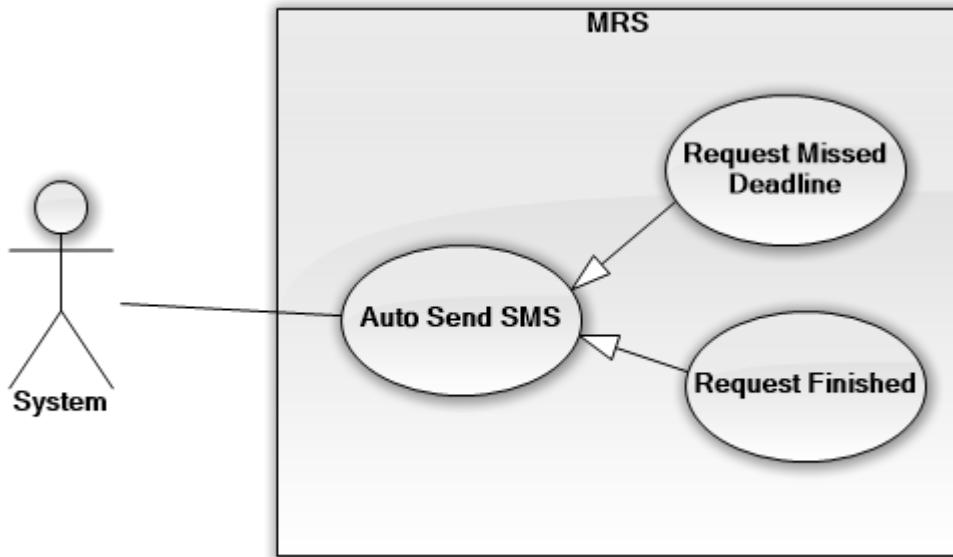


Figure 89-System Auto Send SMS usecase diagram

2.2.70.2. Use Case Specification

USE CASE –Auto Send SMS SPECIFICATION			
Use case No.	UC070	Use case version	2.0
Use case name	Auto Send SMS		
Author	HungDD		
Date	7/7/2013	Priority	Normal
Actor: System			
Summary: When request of customer is finished or missed deadline, system will auto send SMS message to customer.			
Goal: The system will auto send SMS to inform customer.			
Triggers: When request is finished or missed deadline.			
Preconditions: N/A			
Post Conditions: - Success: New SMS is sent to customer. - Fail: N/A			

Main Success Scenario:

Actor Action	System Response
1. When request is finished or missed deadline	2. New SMS message is sent to customer registered phone number.

Alternative Scenario:

N/A

Exceptions:

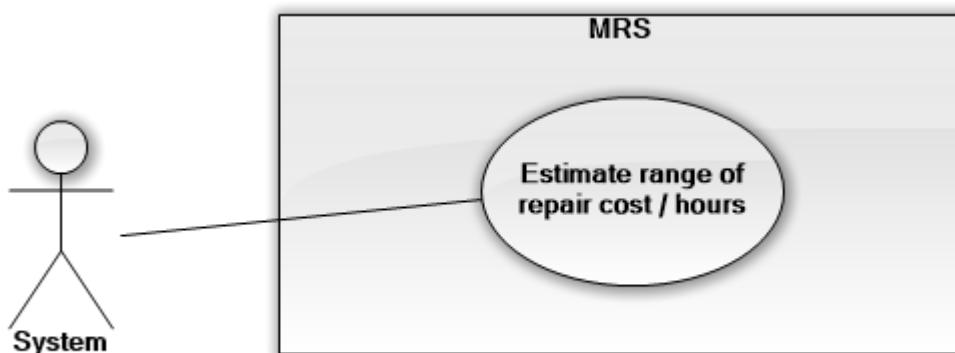
N/A

Relationships:

Auto Send SMS

Business Rules:

- System automatically send SMS to customer

2.2.71. < System > Estimate Range of repaired Cost / Hour**2.2.71.1. Use Case Diagram****Figure 90-System Estimate range of repair cost / hour****2.2.71.2. Use Case Specification**

USE CASE –Estimate range of repaired Cost / Hour SPECIFICATION			
Use case No.	UC071	Use case version	2.0
Use case name	Estimate range of repaired Cost / Hour		
Author	HungDD		
Date	7/7/2013	Priority	Normal
Actor: System Summary: When customer bring product to center to repair / maintain, system can recommend repaired			

cost and hour

Goal:

System recommends repaired cost or hours.

Triggers:

When selected a product of request

Preconditions:

N/A

Post Conditions:

- **Success:** Average repaired cost and repaired hour is displayed in create new request form.
- **Fail:** N/A

Main Success Scenario:

Actor Action	System Response
1. Select a product related to request	2. Average repaired cost and hour is displayed in create new request form

Alternative Scenario:

N/A

Exceptions:

N/A

Relationships:

N/A

Business Rules:

- System will calculate data from history repaired / maintenance of all request related to that product.

2.3. Software System Attributes

2.3.1. Usability

▪ **Ease of Use**

The system should be easy to use by casual users for becoming good at all particular operation. Every users, includes guests, customer, staff or technical manager will be able to use all MRS's functions after a total of 60 minutes' reading user's manual.

Users can use some functionalities of MRS without logging into the system: View FAQ, View Equipment.

▪ **Fast**

With a 6,144 Kbps download, 640 Kbps upload speed or faster Internet connection, every task is finished under 5 seconds.

▪ **User's Manual**

User's manual must support both Vietnamese and English

User's manual must be uploaded to the MRS website for user to be able to read and download whenever the system is running.

2.3.2. Reliability

- **Accuracy**

Defect rate: bugs per thousand lines of code (bugs/KLOC) are 0.4.

2.3.3. Availability

Time server shall be working at least **23.75 hours a day**. When the system goes in under-maintenance, system administrators have maximum **60 minutes** to finish their job.

Do not need to control cases of power failure, operating system error, disconnected network/Internet connection...

2.3.4. Security

Only registered user with authorization can access corresponding functions.

2.3.5. Maintainability

All the errors should be logged, supporting for bug fixing and maintenance.

The system must provide an email address to receive customer feedback.

2.3.6. Portability

MRS can be accessed by any machine with browser and Internet connection.

The system can be accessible via Internet for maintenance.

2.3.7. Performance

- **Respond Time**

With a 6,144 Kbps download, 640 Kbps upload speed or faster Internet connection, the average respond time is **2 seconds**; and the maximum respond time is **5 seconds**.

- **Resource utilization**

- Disk: 800MB
- Bandwidth: 15GB/month

- **Mailing**

Mail server must send emails within 2 minutes after a task that requires email sending has just finished.

- **Compatibility**

MRS's user interface must display without any error for nowadays common screen resolutions: 1024*768 and 1280*800.

MRS's user interface must display without any error on nowadays browsers or newer ones: Mozilla Firefox 12, and Google Chrome 19.

3. Entity Relationship Diagram

Entity	Description
Account	Stores accounts in the system
Profile	Stores basic information of an account
Notification	Stores actions using for notification
Roles	Stores some roles using in the system
ProductRequest	Stores some basic information of requests in the system
Invoice	Stores invoice information of request
ProductPrototype	Stores some information of product prototypes exist in the system
ProductCategory	Stores some information of categories of product prototype
Equipment	Stores some information of equipments in the system
Image	Stores images of equipment
EquipmentCategory	Stores some information of categories of equipment
FAQ	Stores some information of faqs provide for customer
Symptom	Stores some symptoms information after repair or maintenance
Task	Stores some information of tasks of staff
PromotionEvent	Store promotion information

Table 7-Entity diagram description

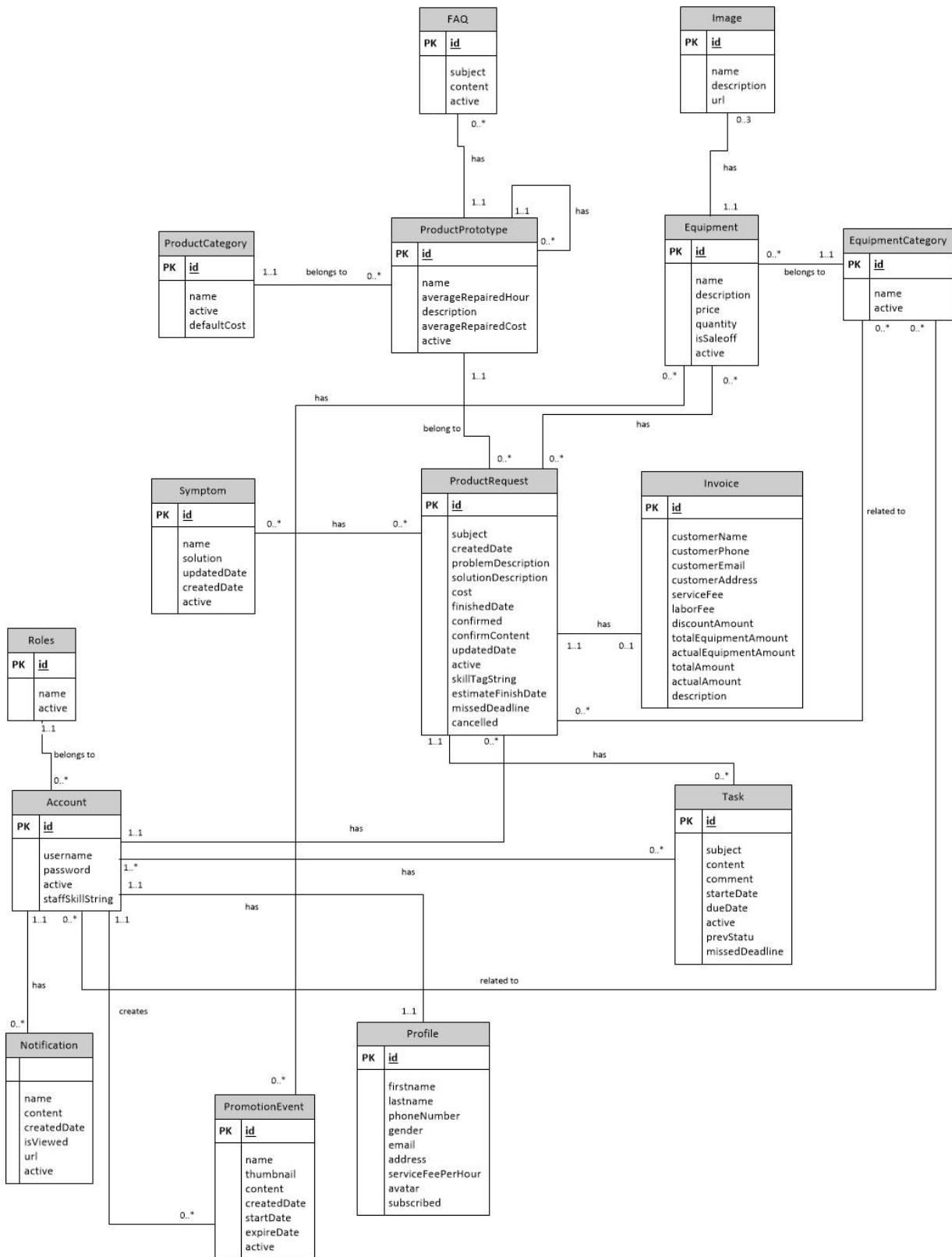


Figure 91-Entity Relationship diagram

IV. Software Design Description

1. Design Overview

- This document describes the technical and user interface design of the MRS system. It includes the system architectural design, the detailed description of components, and detailed design of classes. It also includes the sequence diagrams, database design, and design of user interface.
- The architectural design describes the overall architecture of the system, and the architecture of each main component. It will describe the role of each component. The component design describes more detail about main components. It will describe how sub-components communicate together, and their responsibilities.
- The detailed design describes static and dynamic structure for each component. It includes class diagrams, class explanations, algorithms of some important methods, and sequence diagrams of all use cases.
- The user interface design describes the layout of the system. It also includes description of links between webpages, by a page flow diagram.
- The database design describes entities, relationship between entities, and their details.

2. System Architectural Design

2.1. Choice of System Architecture

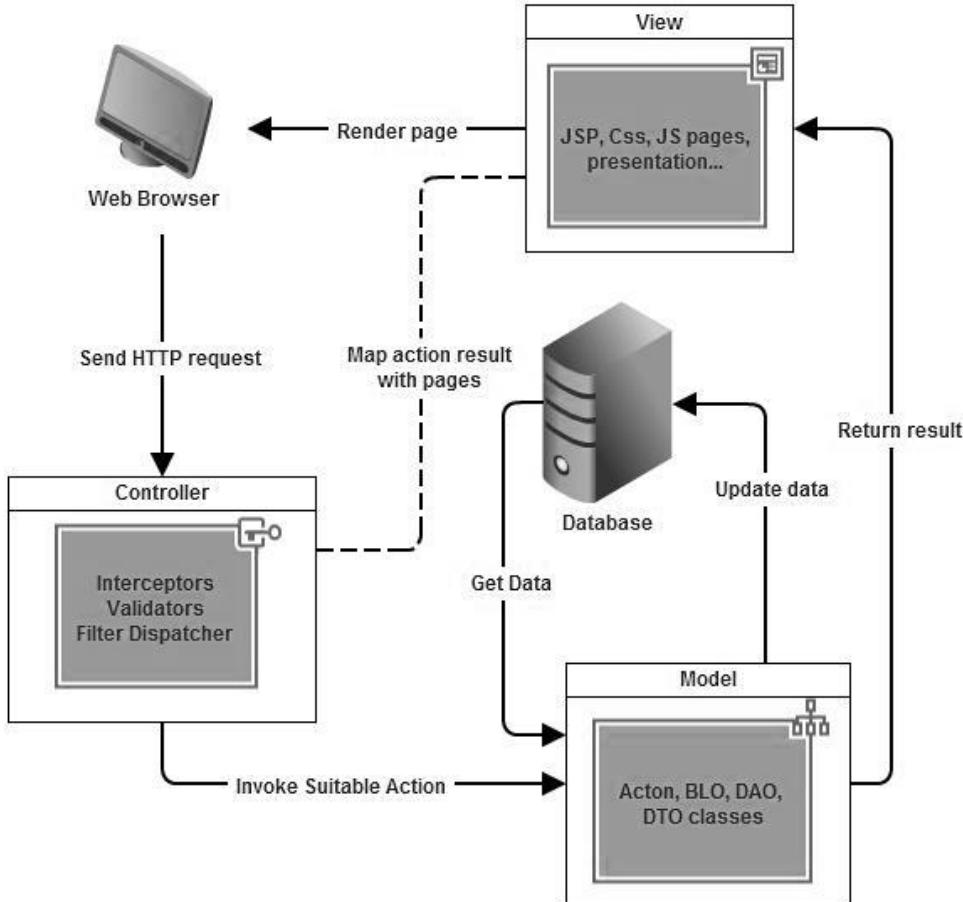


Figure 92 - System Architecture

MRS system is designed based on MVC (Model-View-Controller) model with struts 2 framework for java web application which contains three discrete layers. Each layer has their own responsibility which help stakeholder to implement, and monitor system easier:

- **Controller:** receive HTTP request from web browser and map to appropriate action by Filter Dispatcher (same as Controller) which setting by XML tags in struts.xml file (Controller layer may also contain Valuators, Interceptor to check, validate data before invoke the action)
- **Model:** contains data and business logic (Action), DAO (data access object) and DTO (data transfer object) classes). This layer interacts directly with database to get or update data, calls methods for business logic execution, and returns result to View
- **View:** respond for presenting the data with designed interface to web browser

2.2. Discussion of Alternative Designs

2.2.1. MVP (Model-View-Presenter)

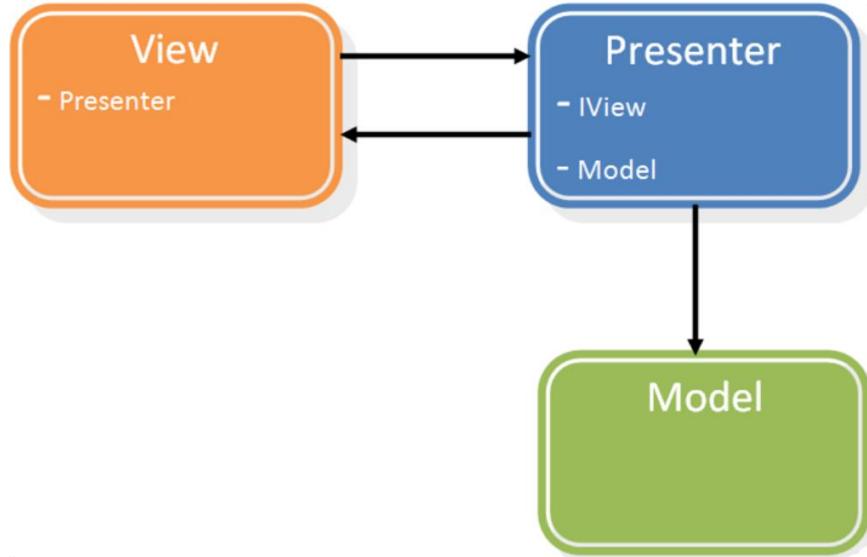


Figure 93 – MVP Model

- Model-view-presenter (MVP) is a derivative of the model-view-controller (MVC) software pattern, also used mostly for building user interfaces.
- In MVP the presenter assumes the functionality of the "middle-man" (played by the controller in MVC). In MVP, all presentation logic is pushed to the presenter. Eventually, the model becomes strictly a domain model.

2.2.2. MVVM (Model-View-View-Model)

The Model-View-View-Model (MVVM) pattern is an application pattern that isolates the user interface from the underlying business logic. MVVM belongs to a class of patterns called Separated Presentation. These patterns provide a clean separation between the UI and the rest of the application. This improves the testability of the application and allows the application and its UI to evolve more easily and independently.

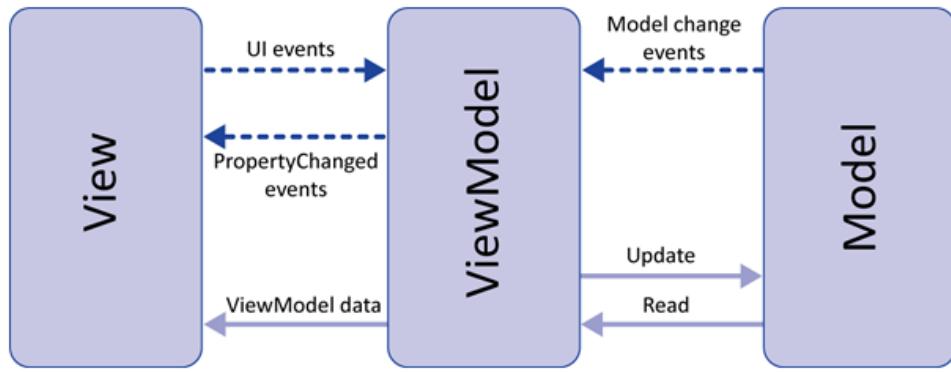


Figure 94 – MVVM Model

2.3. Description of System Interface

N/A

3. Component Diagram

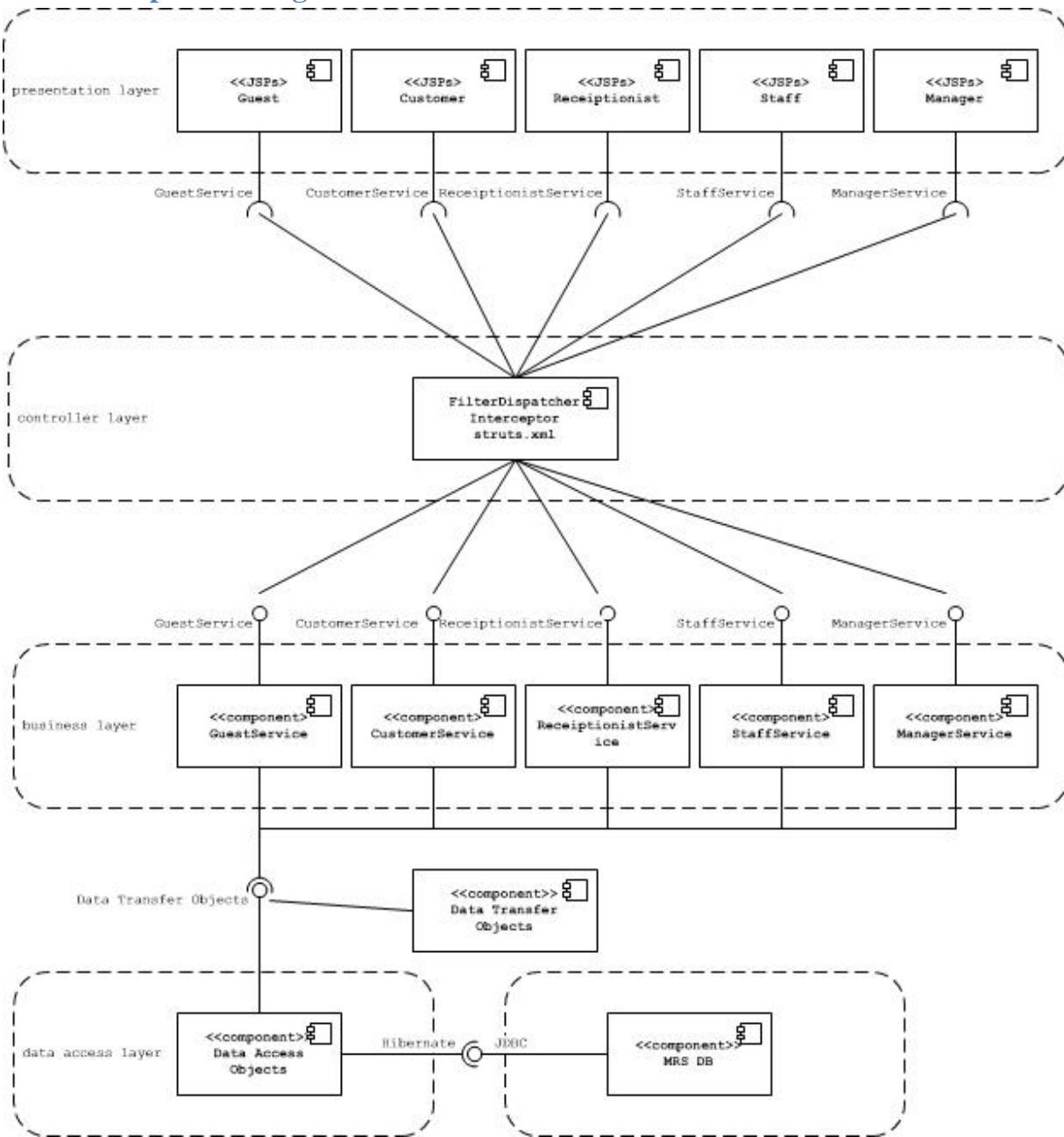


Figure 95-Component Diagram

4. Details Description of Component

4.1. Customer Service

4.1.1. Class Diagram

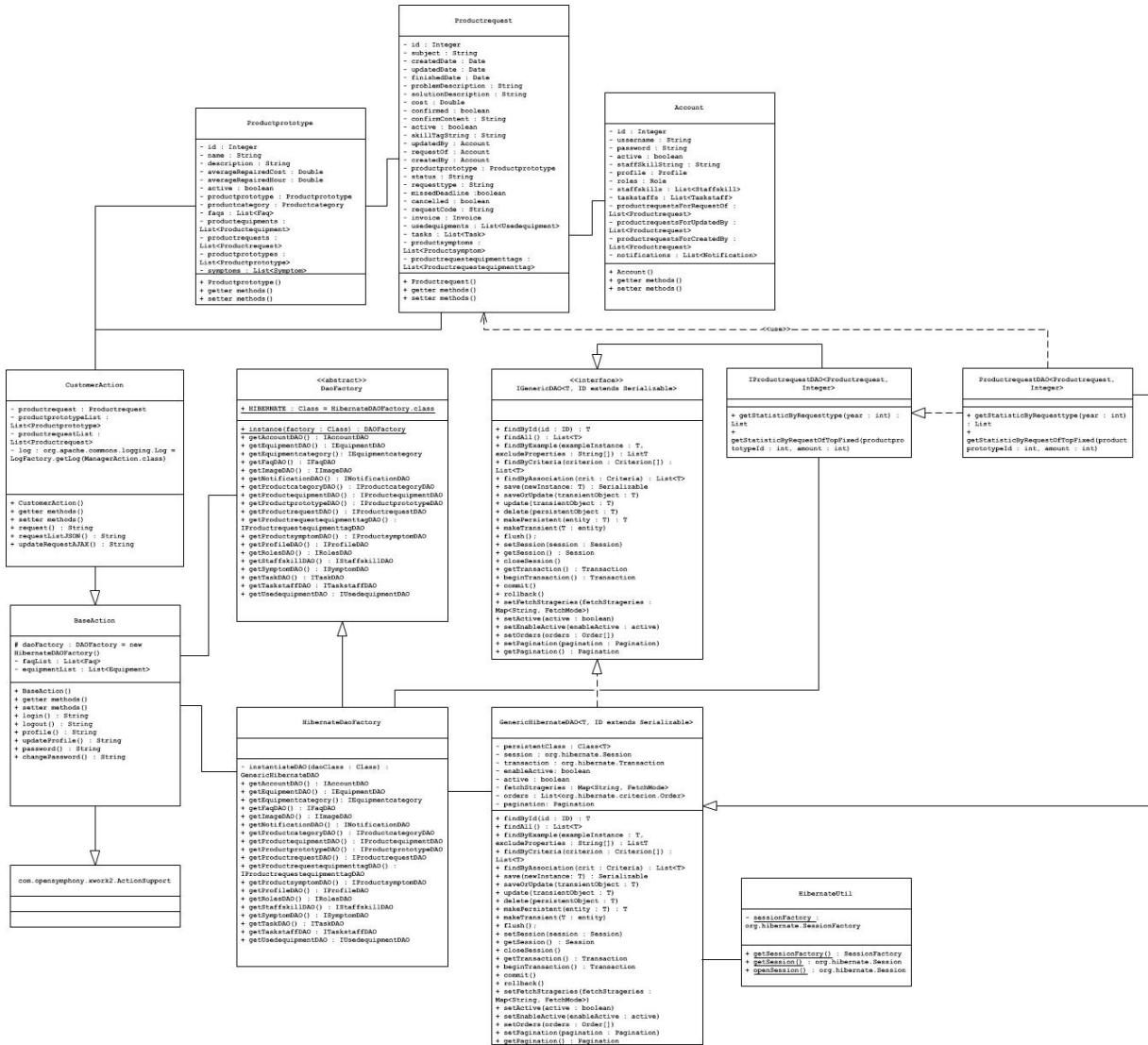


Figure 96-Customer Service Class Diagram

4.1.2. Class Diagram Explanation

BaseAction	
Properties	
protected DAOFactory daoFactory	DAOFactory object
private List<Faq> faqList	List of Faq object
private List<Equipment> equipmentList	List of Equipment object
Methods	
public BaseAction()	Constructor
public String login()	Process login action
public String logout()	Process logout action
public String profile()	Process view profile action
public String updateProfile()	Process update profile action
public String password()	Process redirect to change password action

public String changePassword()	Process change password action
Getter methods	Get value of variables
Setter methods	Set value for variables

Table 8-BaseAction class diagram explanation

CustomerAction	
Properties	
private Productrequest productrequest	Productrequest object
private List<Productprototype> productprototypeList	List of Productprototype object
private List<Productrequest> productrequestList	List of Productrequest object
private Log log	Apache Log object
Methods	
public CustomerAction()	Constructor
public String request()	Process redirect to view request page action
public String requestListJSON()	Process load requests action
public String updateRequestAJAX()	Process update request action
Getter methods	Get value of variables
Setter methods	Set value for variables

Table 9-CustomerAction class diagram explanation

DAOFactory	
Properties	
public static final Class HIBERNATE	HibernateDAOFactory class
Methods	
public static DAOFactory instance(Class factory)	Factory method
public abstract IAccountDAO getAccountDAO();	Abstract method get IAccountDAO interface
public abstract IEquipmentDAO getEquipmentDAO()	Abstract method get IEquipmentDAO interface
public abstract IEquipmentcategoryDAO getEquipmentcategoryDAO()	Abstract method get IEquipmentcategoryDAO interface
public abstract IEstimateequipmentDAO getEstimateequipmentDAO()	Abstract method get IEstimateequipmentDAO interface
public abstract IFaqDAO getFaqDAO()	Abstract method get IFaqDAO interface
public abstract IIImageDAO getImageDAO()	Abstract method get IIImageDAO interface
public abstract INotificationDAO getNotificationDAO()	Abstract method get INotificationDAO interface

public abstract IProductcategoryDAO getProductcategoryDAO()	Abstract method get IProductcategoryDAO interface
public abstract IProductprototypeDAO getProductprototypeDAO()	Abstract method get IProductprototypeDAO interface
public abstract IProductrequestDAO getProductrequestDAO()	Abstract method get IProductrequestDAO interface
public abstract IProductrequestequipmenttagDAO getProductrequestequipmenttagDAO()	Abstract method get IProductrequestequipmenttagsDAO interface
public abstract IProfileDAO getProfileDAO()	Abstract method get IProfileDAO interface
public abstract IRolesDAO getRoleDAO()	Abstract method get IRolesDAO interface
public abstract IStaffskillDAO getStaffskillDAO()	Abstract method get IStaffskillDAO interface
public abstract ISymptomDAO getSymptomDAO()	Abstract method get ISymptomDAO interface
public abstract ITaskDAO getTaskDAO()	Abstract method get ITaskDAO interface
public abstract ITaskstaffDAO getTaskstaffDAO()	Abstract method get ITaskstaffDAO interface
public abstract IUUsedequipmentDAO getUsedequipmentDAO()	Abstract method get IUUsedequipmentDAO interface

Table 10-DAOFactory class diagram explanation

HibernateDAOFactory	
Methods	
public static GenericHibernateDAO instantiateDAO(Class daoClass)	Instantiate GenericHibernateDAO object
public abstract IAccountDAO getAccountDAO();	Override method get IAccountDAO
public abstract IEquipmentDAO getEquipmentDAO()	Override method get IEquipmentDAO
public abstract IEquipmentcategoryDAO getEquipmentcategoryDAO()	Override method get IEquipmentcategoryDAO
public abstract IEstimateequipmentDAO getEstimateequipmentDAO()	Override method get IEstimateequipmentDAO
public abstract IFaqDAO getFaqDAO()	Override method get IFaqDAO
public abstract IImageDAO getImageDAO()	Override method get IImageDAO
public abstract INotificationDAO getNotificationDAO()	Override method get INotificationDAO
public abstract IProductcategoryDAO getProductcategoryDAO()	Override method get IProductcategoryDAO
public abstract IProductprototypeDAO	Override method get IProductprototypeDAO

getProductprototypeDAO()	
public abstract IProductrequestDAO getProductrequestDAO()	Override method get IProductrequestDAO
public abstract IProductrequestequipmenttagDAO getProductrequestequipmenttagDAO()	Override method get IProductrequestequipmenttagsDAO
public abstract IProfileDAO getProfileDAO()	Override method get IProfileDAO
public abstract IRolesDAO getRoleDAO()	Override method get IRolesDAO
public abstract IStaffskillDAO getStaffskillDAO()	Override method get IStaffskillDAO
public abstract ISymptomDAO getSymptomDAO()	Override method get ISymptomDAO
public abstract ITaskDAO getTaskDAO()	Override method get ITaskDAO
public abstract ITaskstaffDAO getTaskstaffDAO()	Override method get ITaskstaffDAO
public abstract IUUsedequipmentDAO getUsedequipmentDAO()	Override method get IUUsedequipmentDAO

Table 11-HibernateDAOFactory class diagram explanation

IGenericDAO<T, ID extends Serializable>	
Methods	
public T findById(ID id)	Get entity T by id
public List<T> findAll()	Get list of T
public List<T> findByExample(T exampleInstance, String[] excludeProperties)	Get list of T depend of some properties
public List<T> findByCriteria(Criterion[] criterion)	Get list of T on some criteria
public List<T> findByAssociation(Criteria crit)	Get list of T using criteria
public Serializable save(T newInstance)	Save instance of T to database
public void saveOrUpdate(T transientObject)	Save or update instance of T to database
public void update(T transientObject)	Update instance of T to database
public void delete(T persistentObject)	Delete instance of T in database
public void makePersistent(T entity)	Make an instance of T persistent with database
public void makeTransient(T entity)	Make an instance of T transient
public void flush()	Flush data to database
public void setSession(Session session)	Set session for hibernate session
public Session getSession()	Get hibernate session object
public void closeSession()	Close hibernate session
public Transaction getTransaction()	Get hibernate transaction

public Transaction beginTransaction()	Begin hibernate transaction
public void commit()	Commit data in session to database
public void rollback()	Rollback transaction
public void setFetchStrategies(Map<String, FetchMode> fetchStrategies)	Set FetchMode
public void setActive(boolean active)	Set active value
public void setEnableActive(boolean active)	Set enable active value
public void setOrders(Order[] orders)	Set order by which properties
public void setPagination(Pagination pagination)	Set pagination
public Pagination getPagition()	Get Pagination object

Table 12-IGenericDAO class diagram explanation

GenericDAO<T, ID extends Serializable>	
Properties	
private Class<T> persistentClass	Class type of T object
private Session session	Hibernate session object
private Transaction transaction	Hibernate transaction object
private boolean enableActive	Flag variable
private boolean active	Value of active
private Map<String, FetchMode> fetchStrategies	Collection of Hibernate FetchMode object
private List<Order> orders	Collection of Hibernate Order object
private Pagination pagination	Pagination object
Methods	
public T findById(ID id)	Get entity T by id
public List<T> findAll()	Get list of T
public List<T> findByExample(T exampleInstance, String[] excludeProperties)	Get list of T depend of some properties
public List<T> findByCriteria(Criterion[] criterion)	Get list of T on some criteria
public List<T> findByAssociation(Criteria crit)	Get list of T using criteria
public Serializable save(T newInstance)	Save instance of T to database
public void saveOrUpdate(T transientObject)	Save or update instance of T to database
public void update(T transientObject)	Update instance of T to database
public void delete(T persistentObject)	Delete instance of T in database
public void makePersistent(T entity)	Make an instance of T persistent with database
public void makeTransient(T entity)	Make an instance of T transient

public void flush()	Flush data to database
public void setSession(Session session)	Set session for hibernate session
public Session getSession()	Get hibernate session object
public void closeSession()	Close hibernate session
public Transaction getTransaction()	Get hibernate transaction
public Transaction beginTransaction()	Begin hibernate transaction
public void commit()	Commit data in session to database
public void rollback()	Rollback transaction
public void setFetchStrategies(Map<String, FetchMode> fetchStrategies)	Set FetchMode
public void setActive(boolean active)	Set active value
public void setEnableActive(boolean active)	Set enable active value
public void setOrders(Order[] orders)	Set order by which properties
public void setPagination(Pagination pagination)	Set pagination
public Pagination getPagition()	Get Pagination object

Table 13-GenericDAO class diagram explanation

HibernateUtil	
Properties	
private SessionFactory sessionFactory	Hibernate SessionFactory object
Methods	
public static SessionFactory getSessionFactory()	Get Hibernate SessionFactory object
public static Session getSession()	Get Hibernate Session object
public static Session openSession()	Open Hibernate Session

Table 14-HibernateUtil class diagram explanation

IProductrequestDAO<Productrequest, Integer>	
Methods	
public List getStatisticByRequesttype(int year)	Statistic by request type depend on year
public List getStatisticByRequestOfTopFixed(int productprototypeId, int amount)	Statistic by product prototype depend on amount

Table 15-IProductrequestDAO class diagram explanation**ProductrequestDAO<Productrequest, Integer>**

Methods	
public List getStatisticByRequesttype(int year)	Statistic by request type depend on year
public List getStatisticByRequestOfTopFixed(int productprototypeId, int amount)	Statistic by product prototype depend on amount

Table 16-ProductrequestDAO class diagram explanation

4.2. Guest Service

4.2.1. Class Diagram

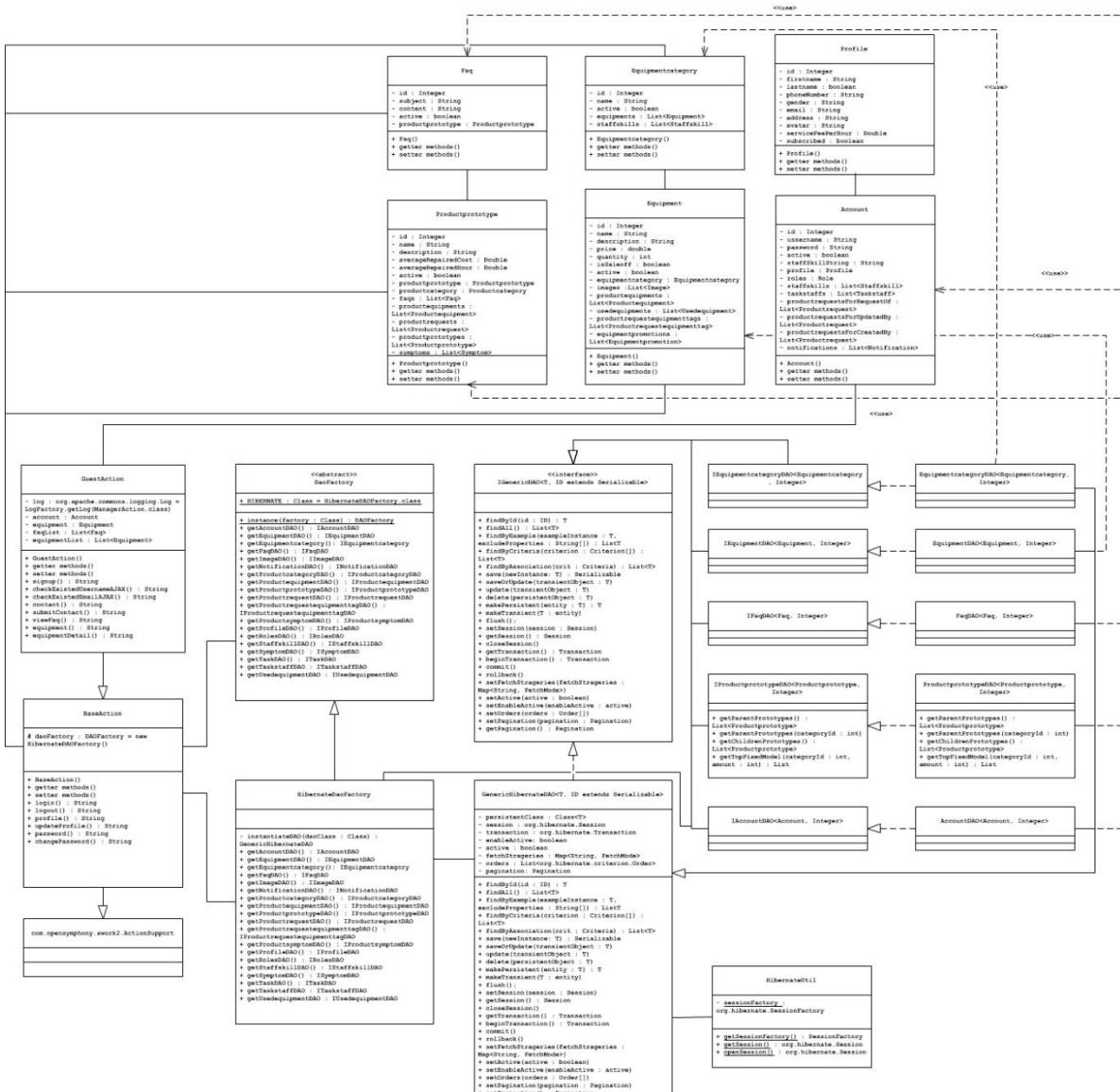


Figure 97-Guest Service Class Diagram

4.2.2. Class Diagram Explanation

GuestAction	
Properties	
private Account account	Account object
private Equipment equipment	Equipment object
private List<Faq> faqList	List of Faq object
private List<Equipment> equipmentList	List of Equipment object
private Log log	Appache Log object
Methods	
public GuestAction()	Constructor
public String signup()	Process sign up action
public String checkExistedUsernameAJAX()	Process check existed username action
public String checkExistedEmailAJAX()	Process check existed email action
public String contact()	Process redirect to contact page action
public String submitContact()	Process submit contact action
public String viewFaq()	Process view faq action
public String equipment()	Process redirect to equipment page action
public String equipmentDetail()	Process view equipment detail action
Getter methods	Get value of variables
Setter methods	Set value for variables

Table 17-GuestAction class diagram explanation

IProductprototypeDAO<Productprototype>	
Methods	
public List<Productprototype> getParentProrotype()	List of Productprototype object that have not children prorotype
public List<Productprototype> getParentProrotype(int categoryId)	Get Productprototype object that have not children prorotype depend on Productcategory id
public List<Productprototype> getChildrenPrototypes()	List of Productprototype object that are children of prototype
public List getTopFixedModel(int categoryId, int amount)	List of top repaired/maintained prototype depend on category

Table 18-IProductprototypeDAO class diagram explanation

4.3. Staff Service

4.3.1. Class Diagram

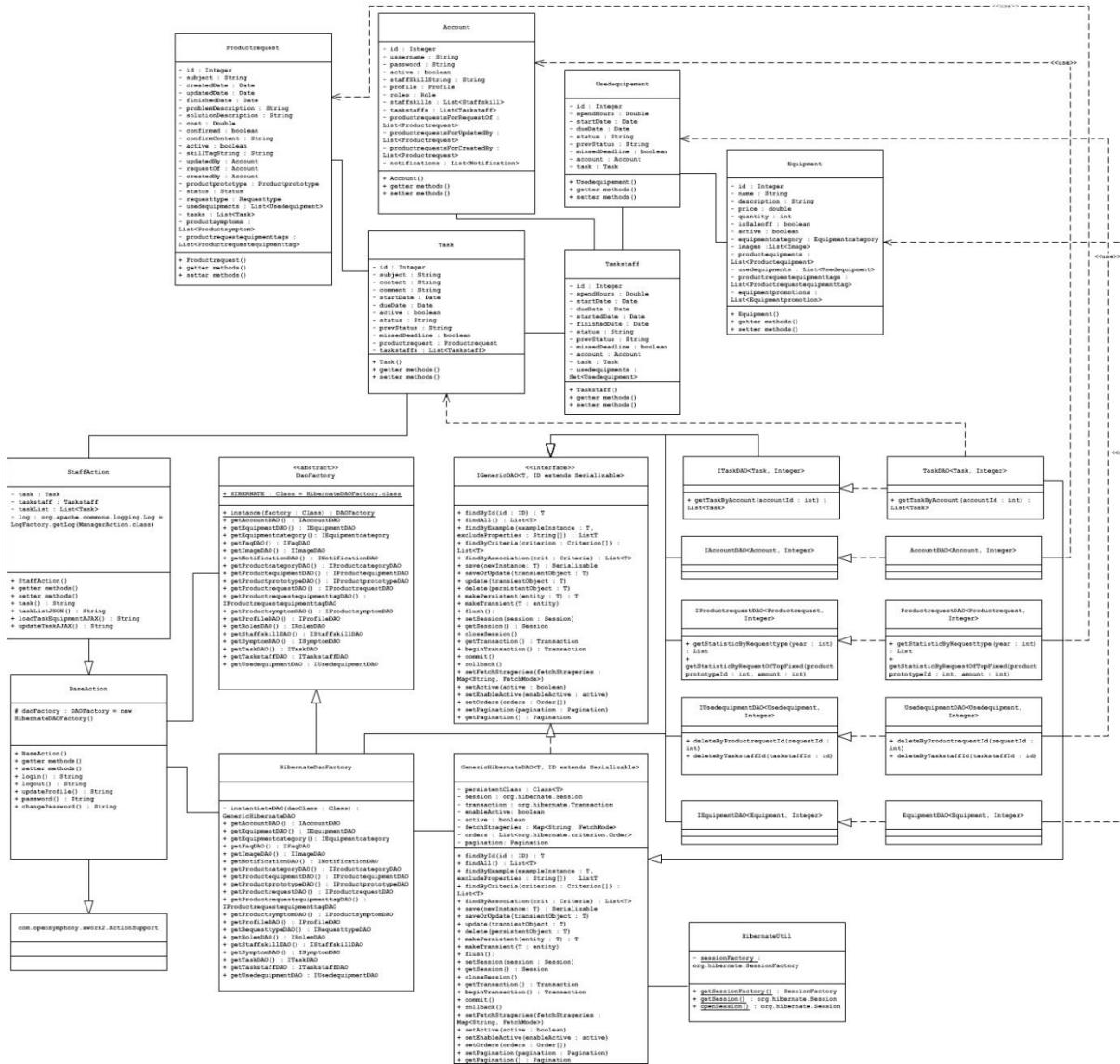


Figure 98-Staff Service Class Diagram

4.3.2. Class Diagram Explanation

StaffAction	
Properties	
private Task task	Task object
private Taskstaff taskstaff	Taskstaff object
private List<Task> taskList	List of Task object
private Log log	Appache Log object
Methods	
public StaffAction()	Constructor

Getter methods	Get value of variables
Setter methods	Set value for variables

Table 19-StaffAction class diagram explanation

ITaskDAO<Task, Integer>	
Methods	
public void getTaskByAccount(int accountId)	List of Task object belongs to account

Table 20-ITaskDAO class diagram explanation

TaskDAO<Task, Integer>	
Methods	
public List<Task> getTaskByAccount(int accountId)	List of Task object belongs to account

Table 21-TaskDAO class diagram explanation

IUsedEquipmentDAO<UsedEquipment, Integer>	
Methods	
public void deleteByProductrequestId(int requestId)	Delete UsedEquipment from database depend on request id
public void deleteByTaskstaffId(int taskstaffId)	Delete UsedEquipment from database depend on taskstaff id

Table 22-IUsedEquipmentDAO

UsedEquipmentDAO<UsedEquipment, Integer>	
Methods	
public void deleteByProductrequestId(int requestId)	Delete UsedEquipment from database depend on request id
public void deleteByTaskstaffId(int taskstaffId)	Delete UsedEquipment from database depend on taskstaff id

Table 23-UsedEquipmentDAO

4.4. Manager Service

4.4.1. Class Diagram

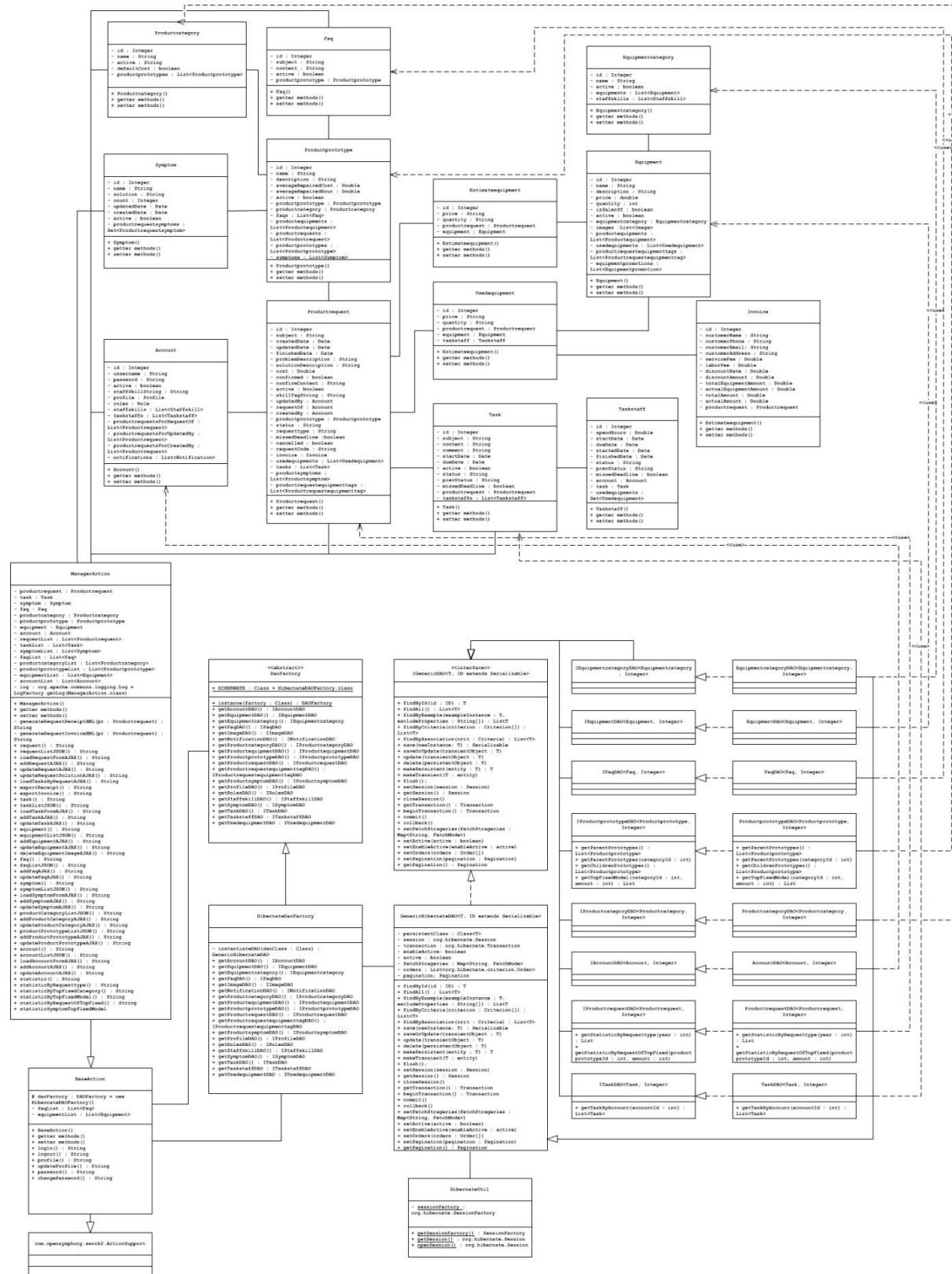


Figure 99-Manger Service Class Diagram

4.4.2. Class Diagram Explanation

ManagerAction	
Properties	
private Productrequest productrequest	Productrequest object
private Task task	Task object
private Symptom symptom	Symptom object
private Faq faq	Faq object
private Productcategory productcategory	Productcategory object
private Productprototype productprototype	Productprototype object
private Equipment equipment	Equipment object
private Account account	Account object
private List<Productrequest> requestList	List of Productrequest object
private List<Task> taskList	List of Task object
private List<Symptom> symptomList	List of Symptom object
private List<Faq> faqList	List of Faq object
private List<Productcategory> productcategoryList	List of Productcategory object
private List<Productprototype> productprototypeList	List of Productprototype object
private List<Equipment> equipmentList	List of Equipment object
private List<Account> accountList	List of Account object
private Log log	Appache Log object
Methods	
public ManagerAction()	Constructor
private String generateRequestReceiptXML(Productrequest pr)	Generate some information of request to XML string
private String generateRequestInvoiceXML(Productrequest pr)	Generate some information of request to XML string
public String request()	Process redirect to request page action
public String requestListJSON()	Process load list of request action
public String loadRequestFormAJAX()	Process load information for request form action
public String addRequestAJAX()	Process add new request action
public String updateRequestAJAX()	Process update new request action
public String updateRequestSolutionAJAX()	Process update request solution action
public String loadTaskByRequestAJAX()	Process load list of task belong to request action
public String exportReceipt()	Process export receipt to pdf action
public String exportInvoice()	Process export invoice to pdf action
public String task()	Process redirect to task page action
public String taskListJSON()	Process load list of task action
public String loadTaskFormAJAX()	Process load information for task form action

public String addTaskAJAX()	Process add new task action
public String updateTaskAJAX()	Process update task action
public String equipment()	Process redirect to equipment page action
public String addEquipmentAJAX()	Process add new equipment action
public String updateEquipmentAJAX()	Process update equipment action
public String deleteEquipmentImageAJAX()	Process delete image of equipment action
public String faq()	Process redirect to faq page action
public String faqListJSON()	Process load list of faq action
public String addFaqAJAX()	Process add new faq action
public String updateFaqAJAX()	Process update faq action
public String productPrototypeListJSON()	Process load list of productprototype action
public String addProductPrototypeAJAX()	Process add new product prototype action
public String updateProductPrototypeAJAX()	Process update product prototype action
public String account()	Process redirect to account page action
public String accountListJSON()	Process load list of account action
public String loadAccountFormAJAX()	Process load information of account form action
public String addAccountAJAX()	Process add new account action
public String updateAccountAJAX()	Process update account action
public String statistic()	Process redirect to statistic page action
public String statisticByRequesttype()	Process statistic by request type action
public String statisticByTopFixedCategory()	Process statistic by top fixed category action
public String statisticByTopFixedModel()	Process statistic by top fixed model action
public String statisticByRequestOfTopFixed()	Process statistic by request of top fixed action
public String statisticSymptomTopFixedModel()	Process statistic symptom top fixed model action
Getter methods	Get value of variables
Setter methods	Set value for variables

Table 24-MangerAction class diagram explanation

5. Sequence Diagram

5.1. <Guest> Register

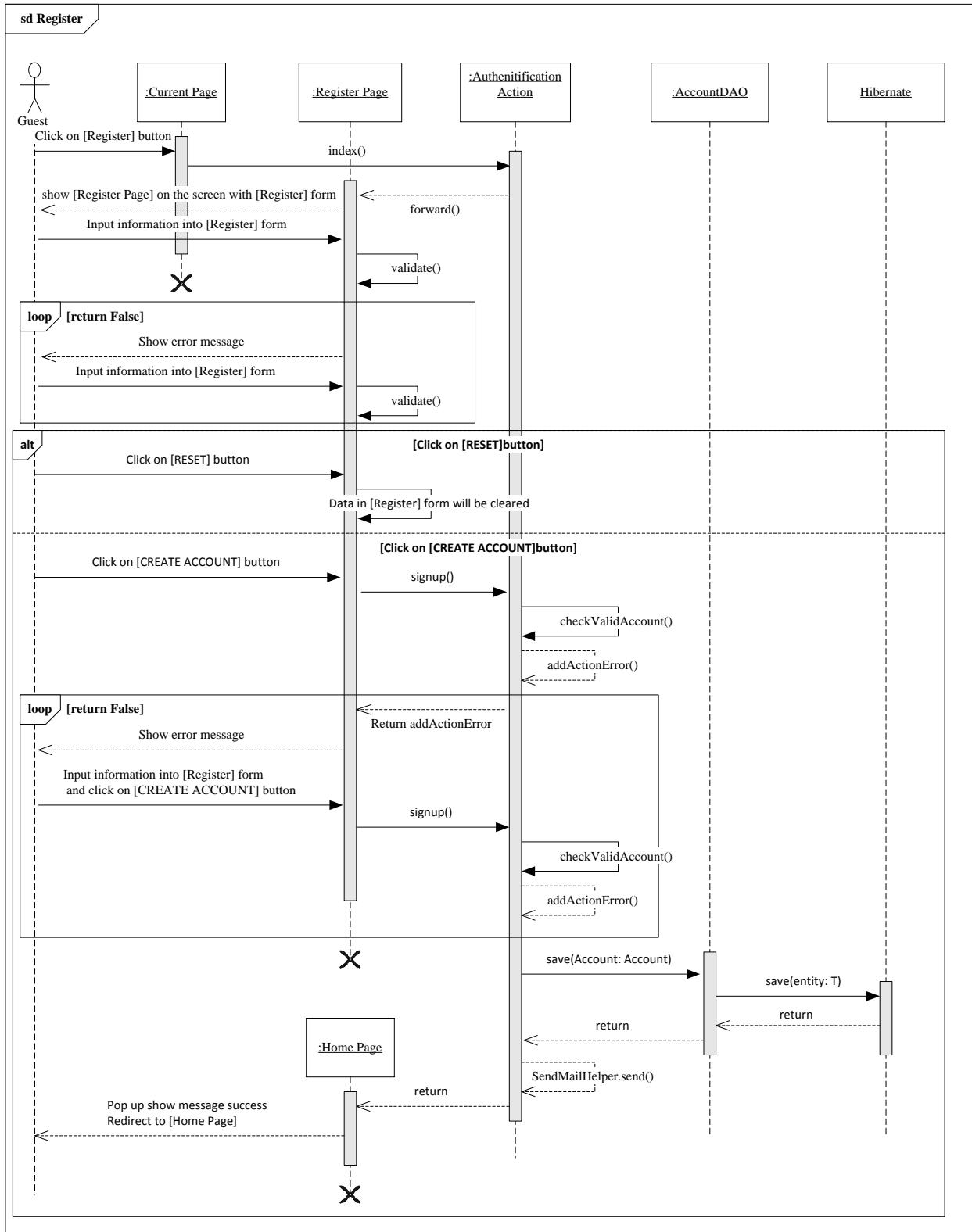


Figure 100-Register sequence diagram

5.2. <Guest> View FAQ

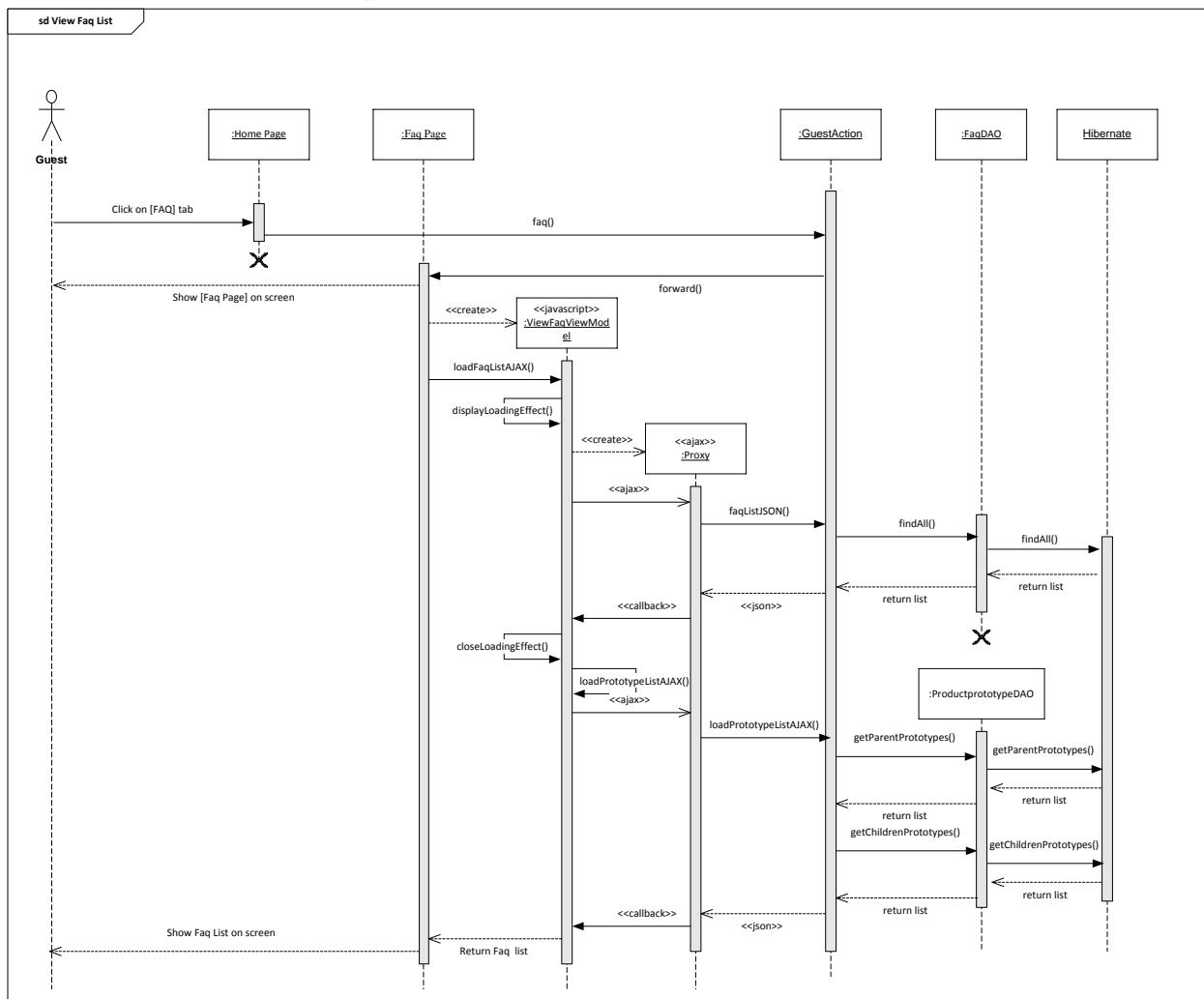


Figure 101-Guest View FAQ sequence diagram

5.3. <Guest> [View Equipment] View Equipment by Category

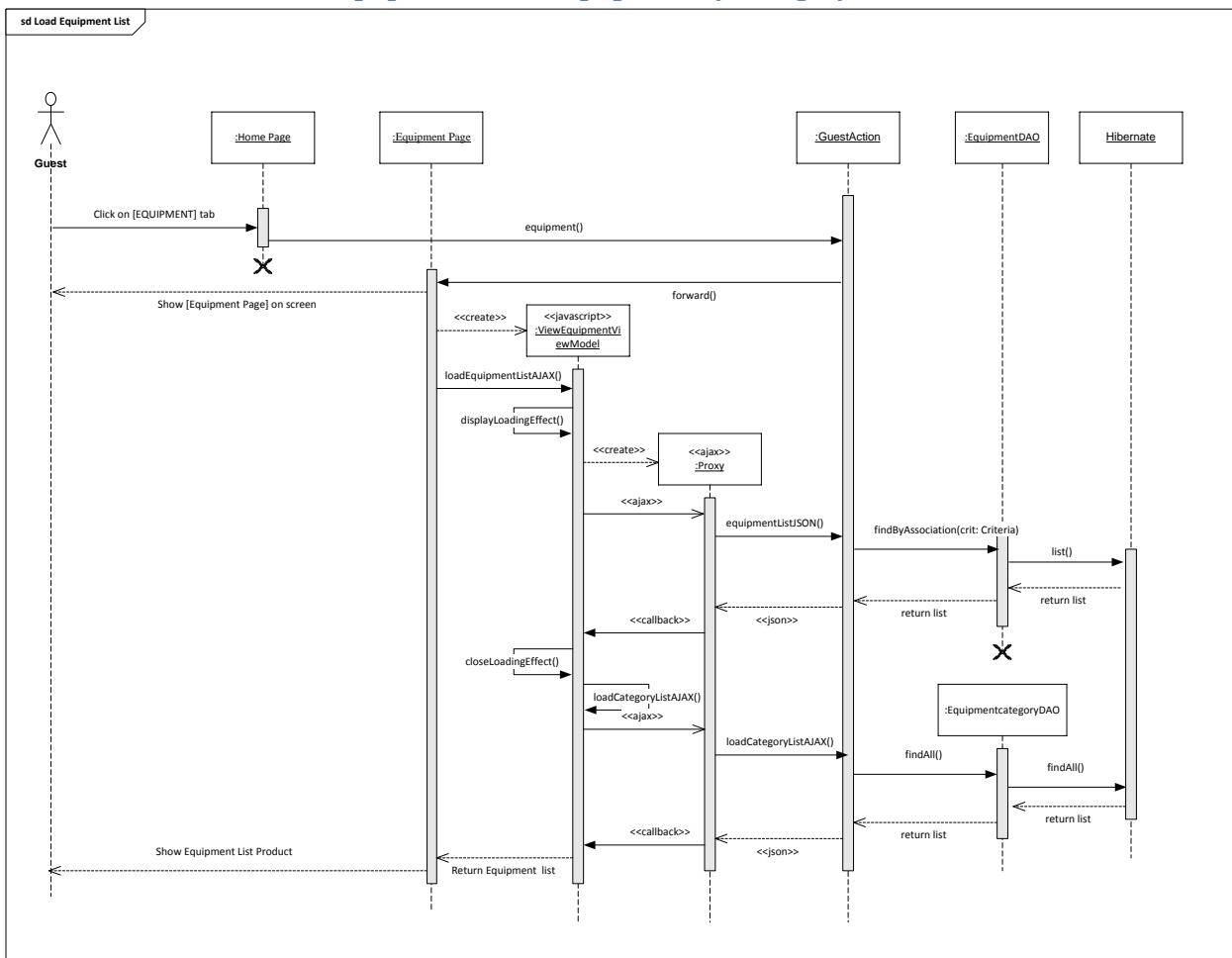
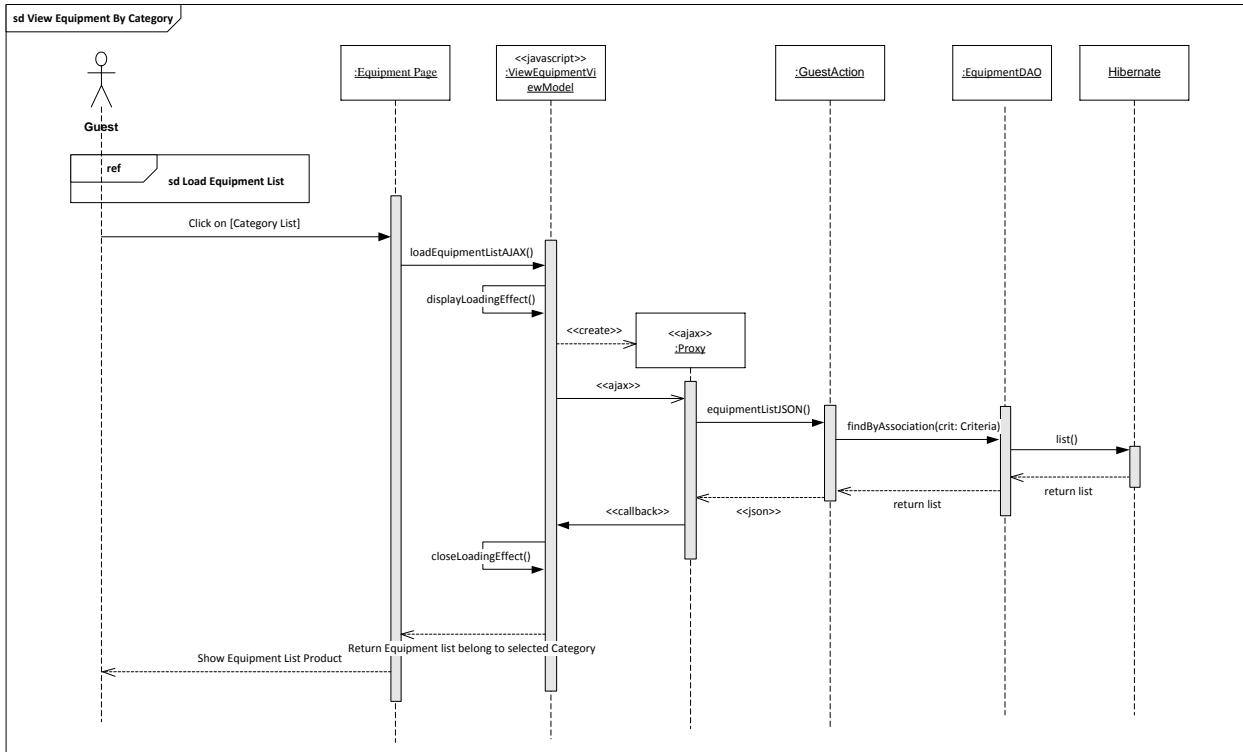


Figure 102-Guest View Equipment By Category sequence diagram



5.4. <Guest> [View Equipment] View Equipment Detail

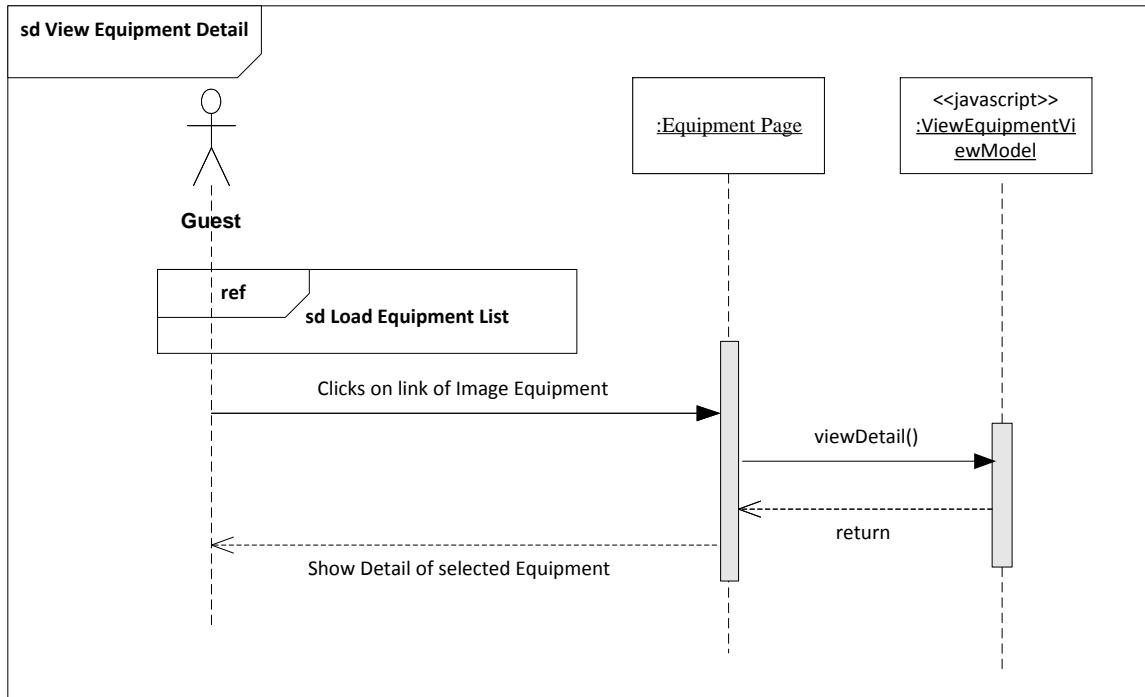


Figure 103- View Equipment Detail sequence diagram

5.5. <Guest> [View Equipment] Search Equipment

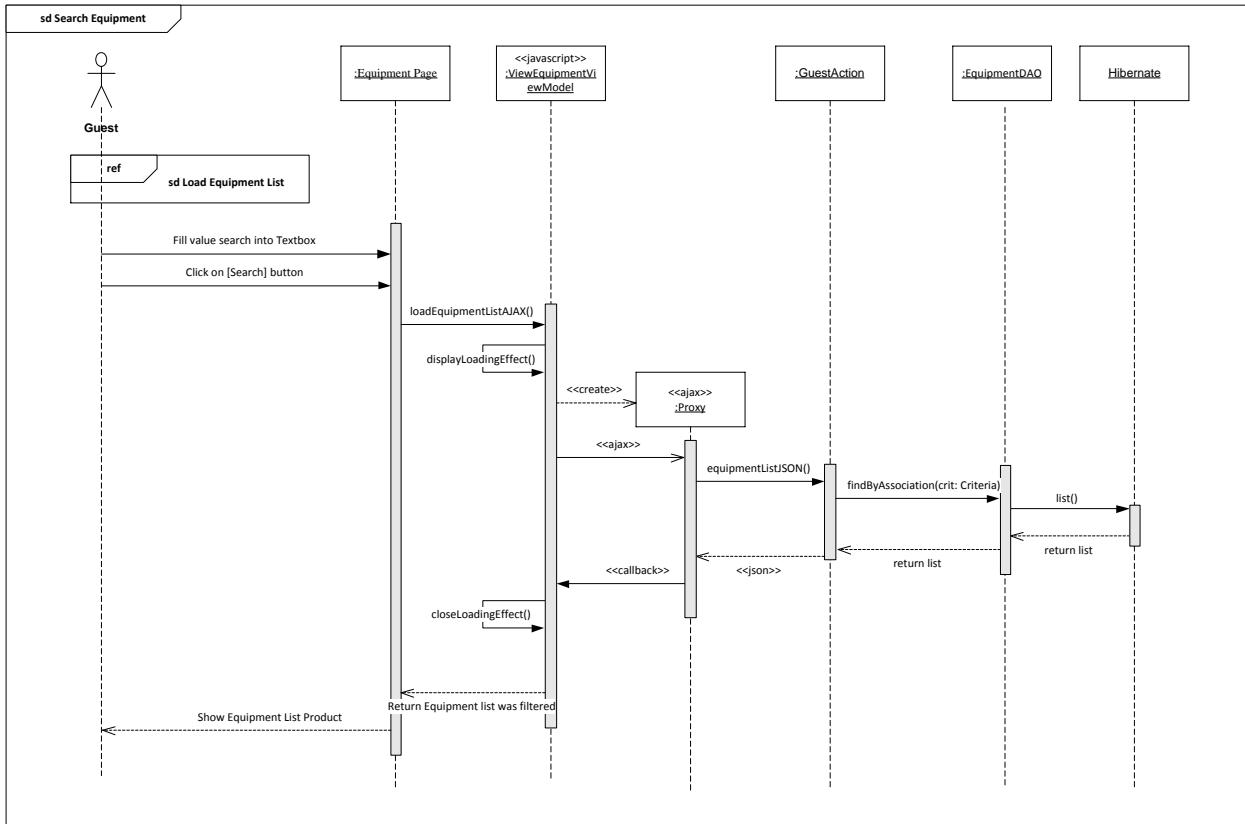


Figure 104-Guest Search Equipment

5.6. <Guest> Tracking request process

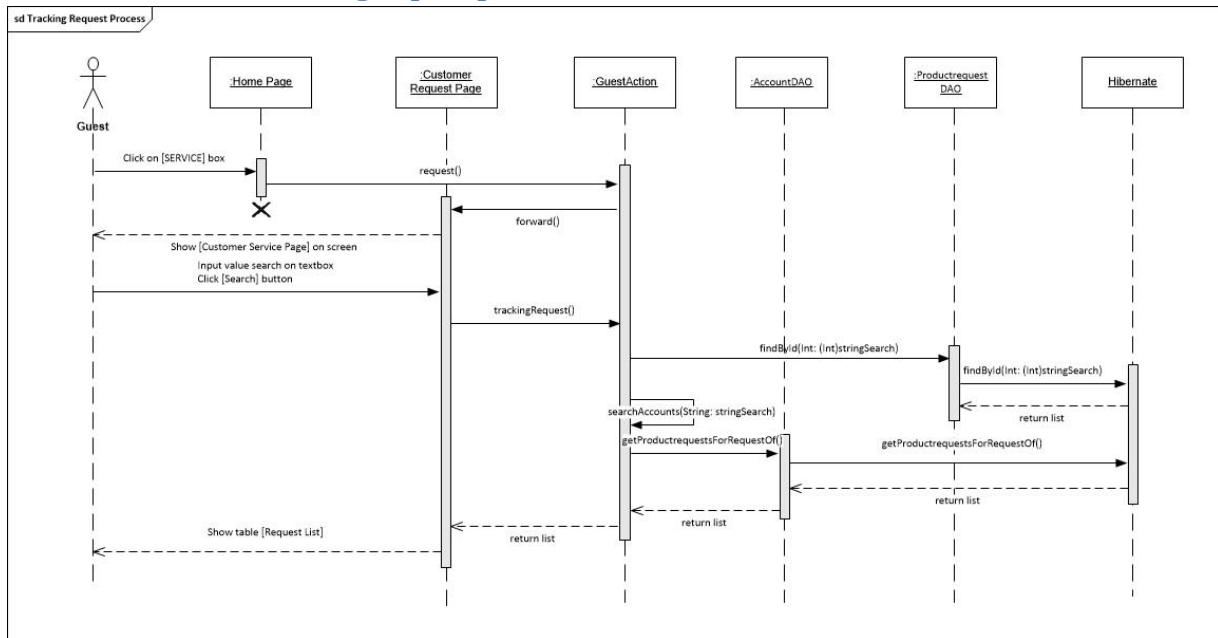


Figure 105-Guest Tracking Request Process sequence diagram

5.7. <Guest> Login

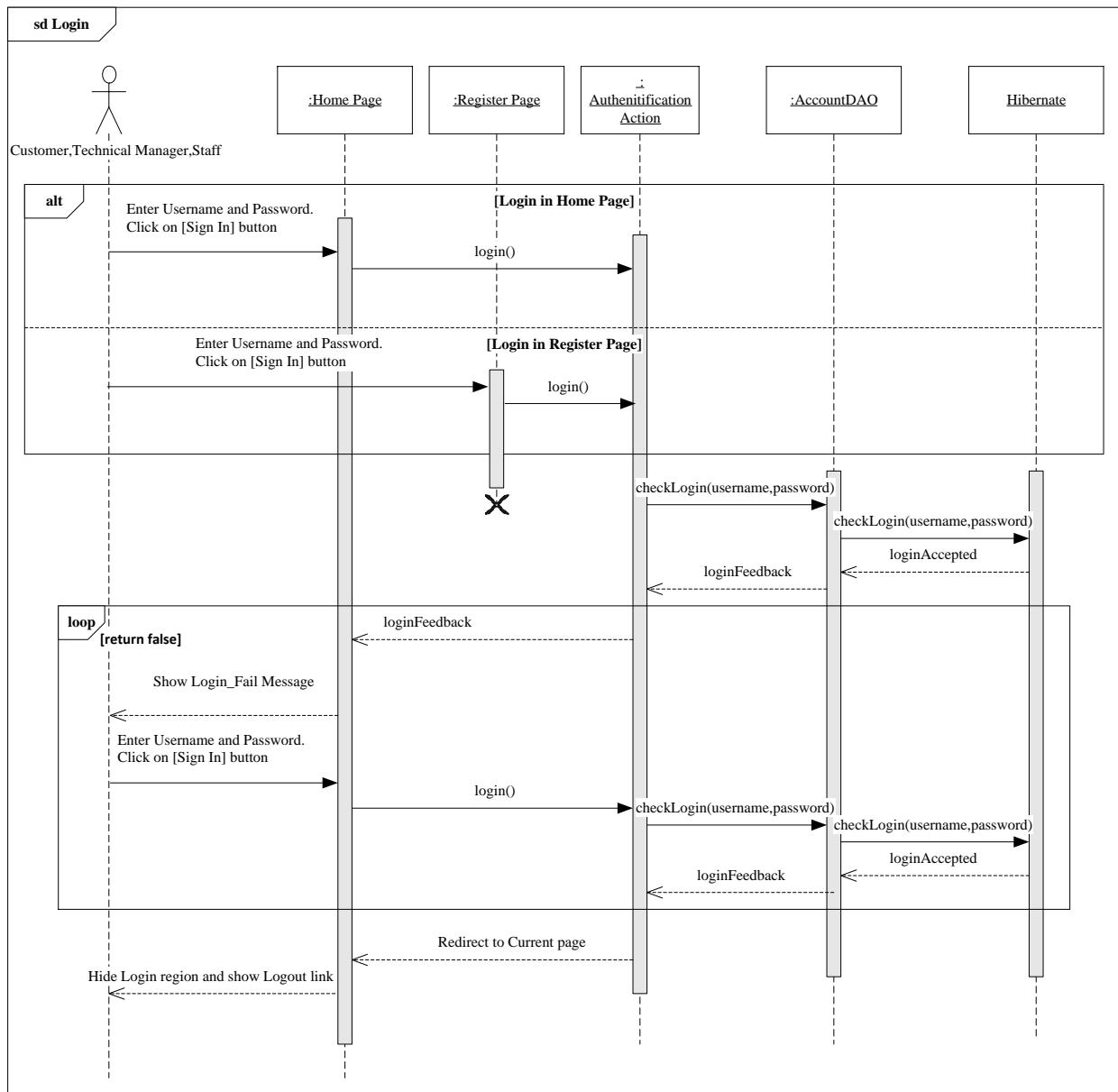


Figure 106-Login sequence diagram

5.8. <Registered User> [Update Profile] Update Basic Information

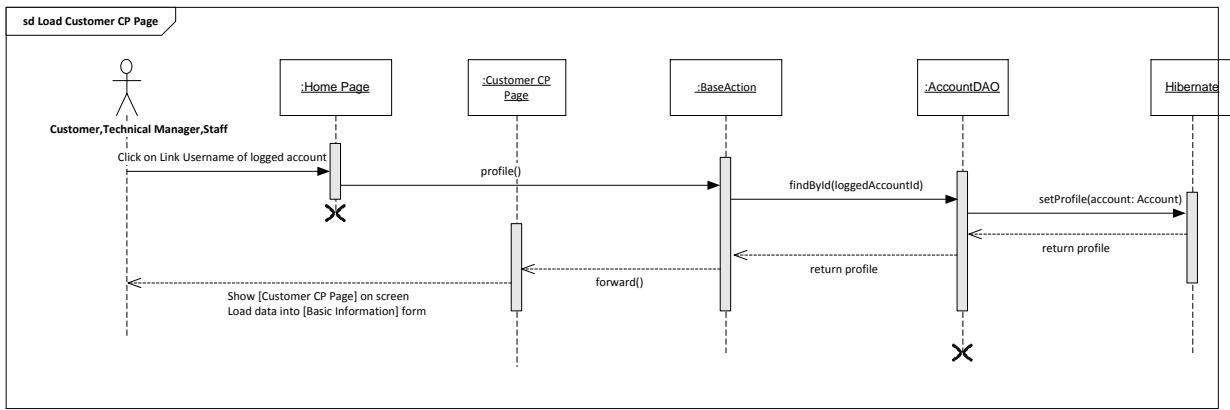


Figure 107-Load Customer CP Page sequence diagram

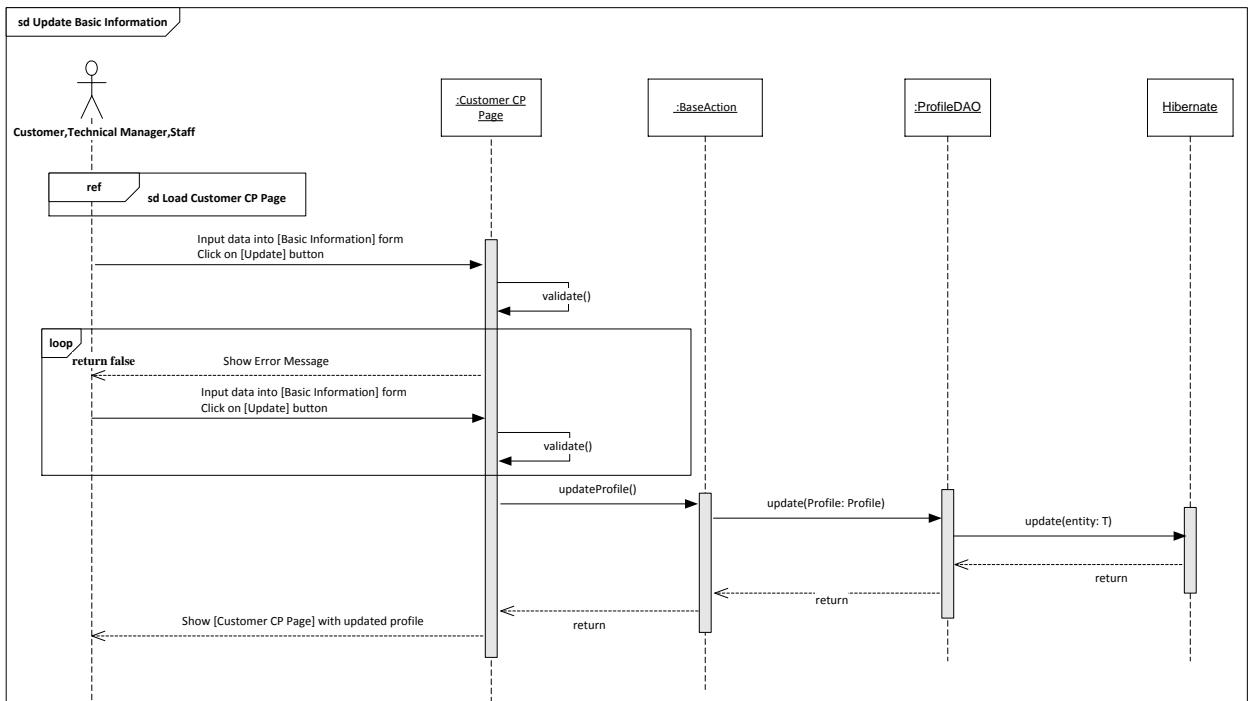


Figure 108-Update Basic Information sequence diagram

5.9. <Registered User > [Update Profile] Change Password

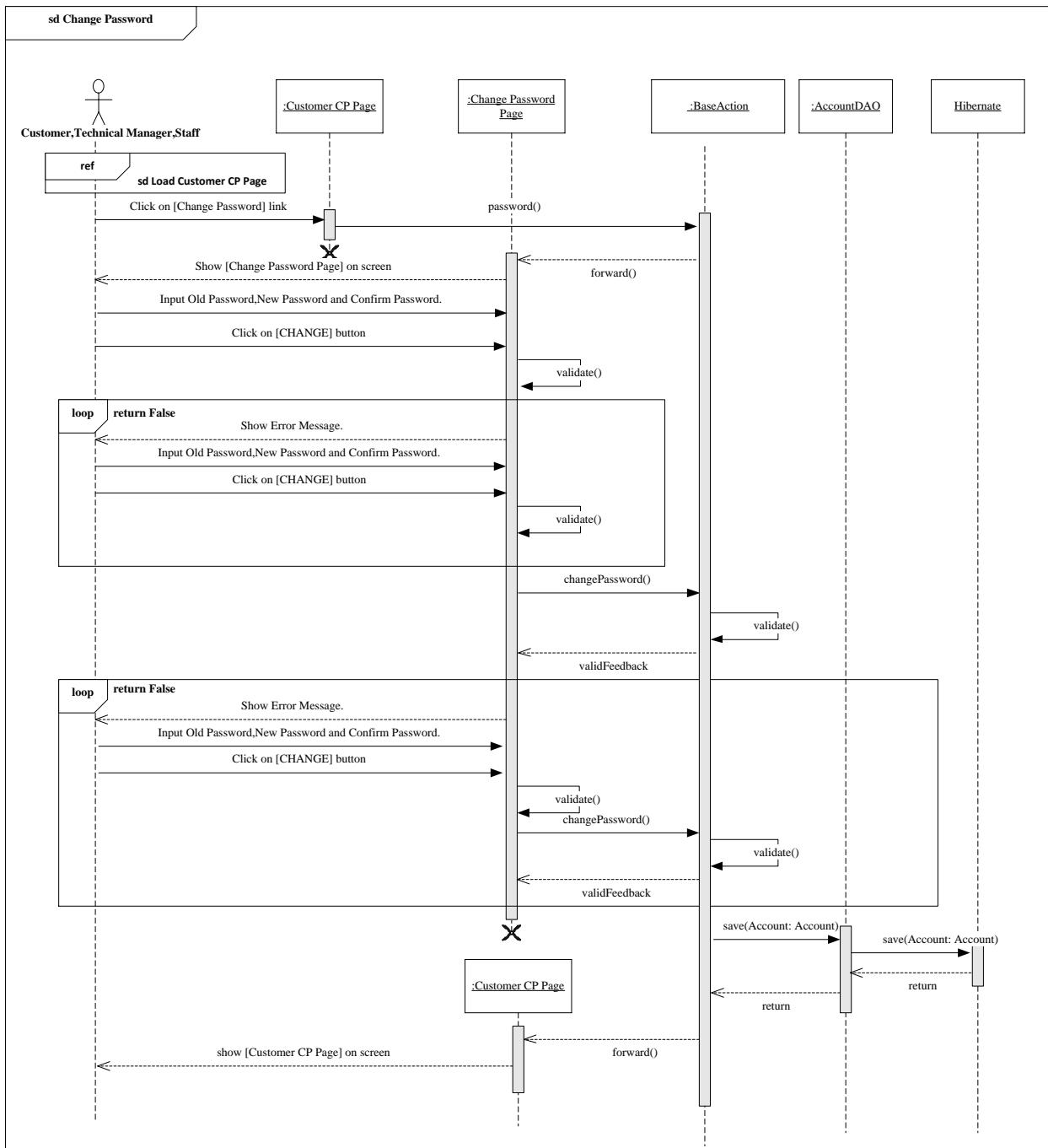


Figure 109-Registered User Change Password sequence diagram

5.10. <Registered User> Logout

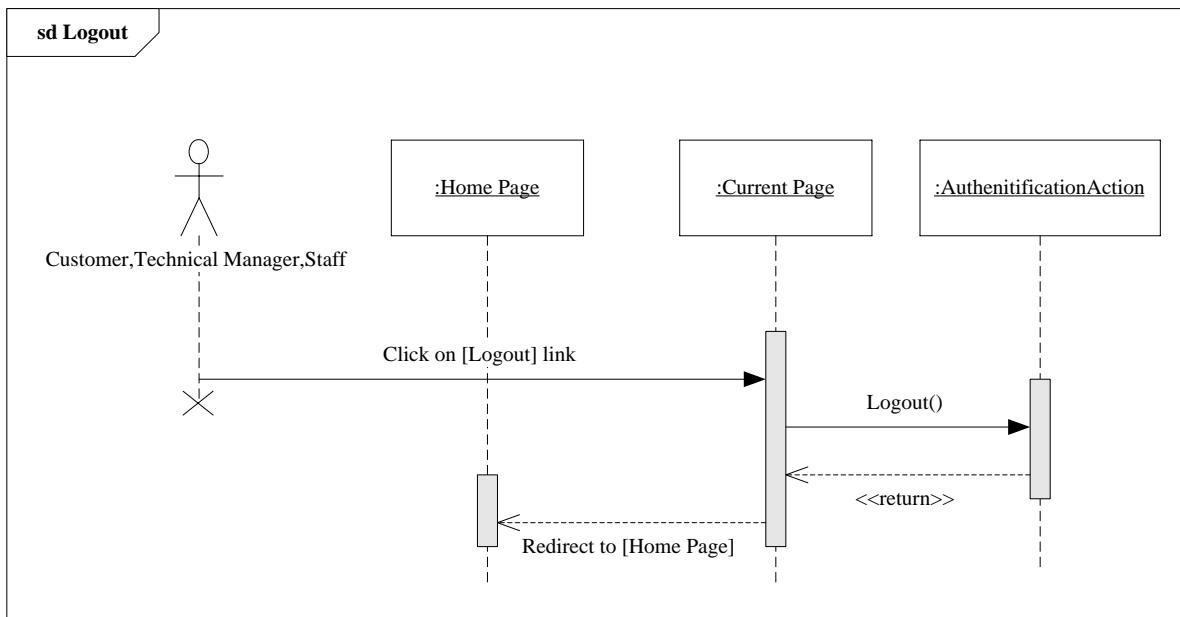


Figure 110-Registered User Logout sequence diagram

5.11. <Customer> [Tracking Request] View Request List

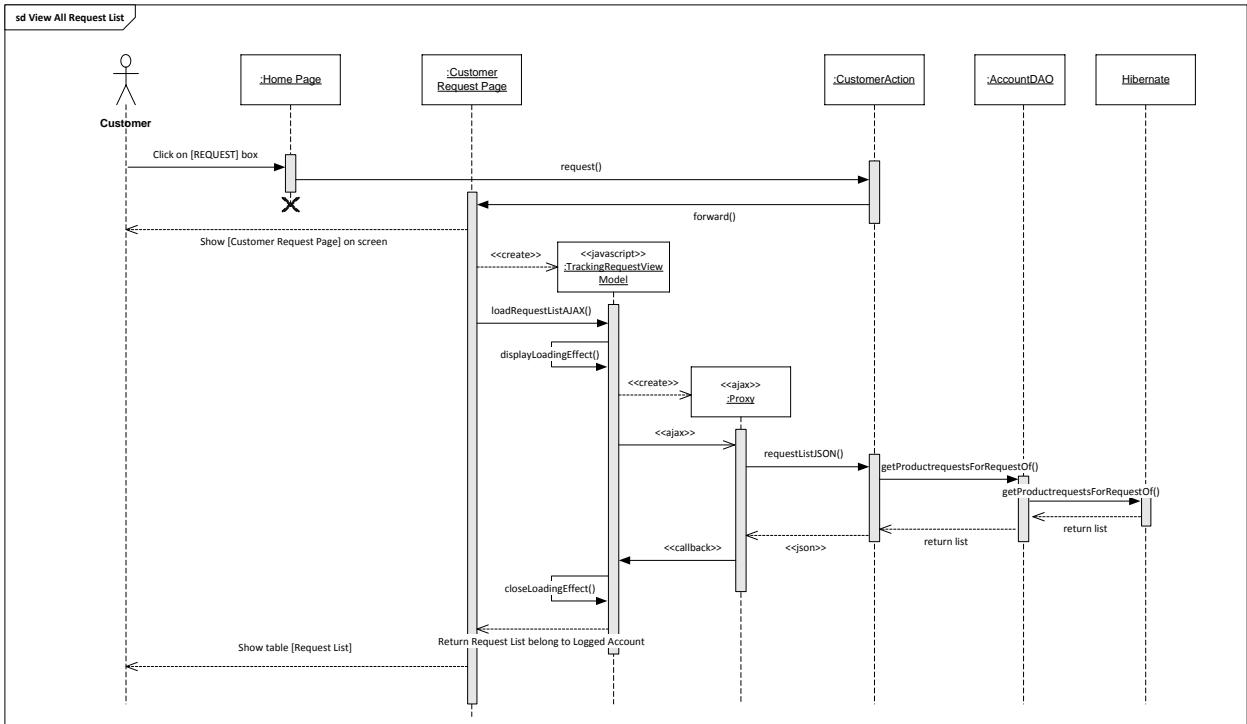


Figure 111-Customer View Request List sequence diagram

5.12. < Customer > [Tracking Request] View Request Detail

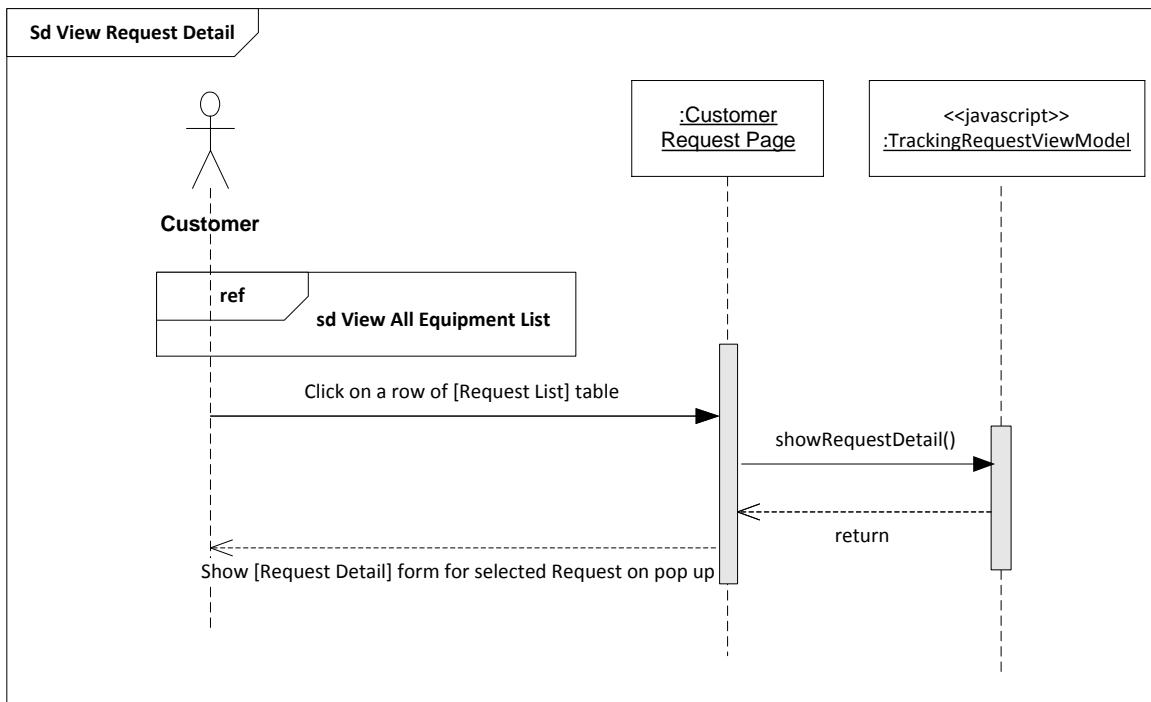


Figure 112-Customer View Request List sequence diagram

5.13. < Customer > [Tracking Request] Update Request

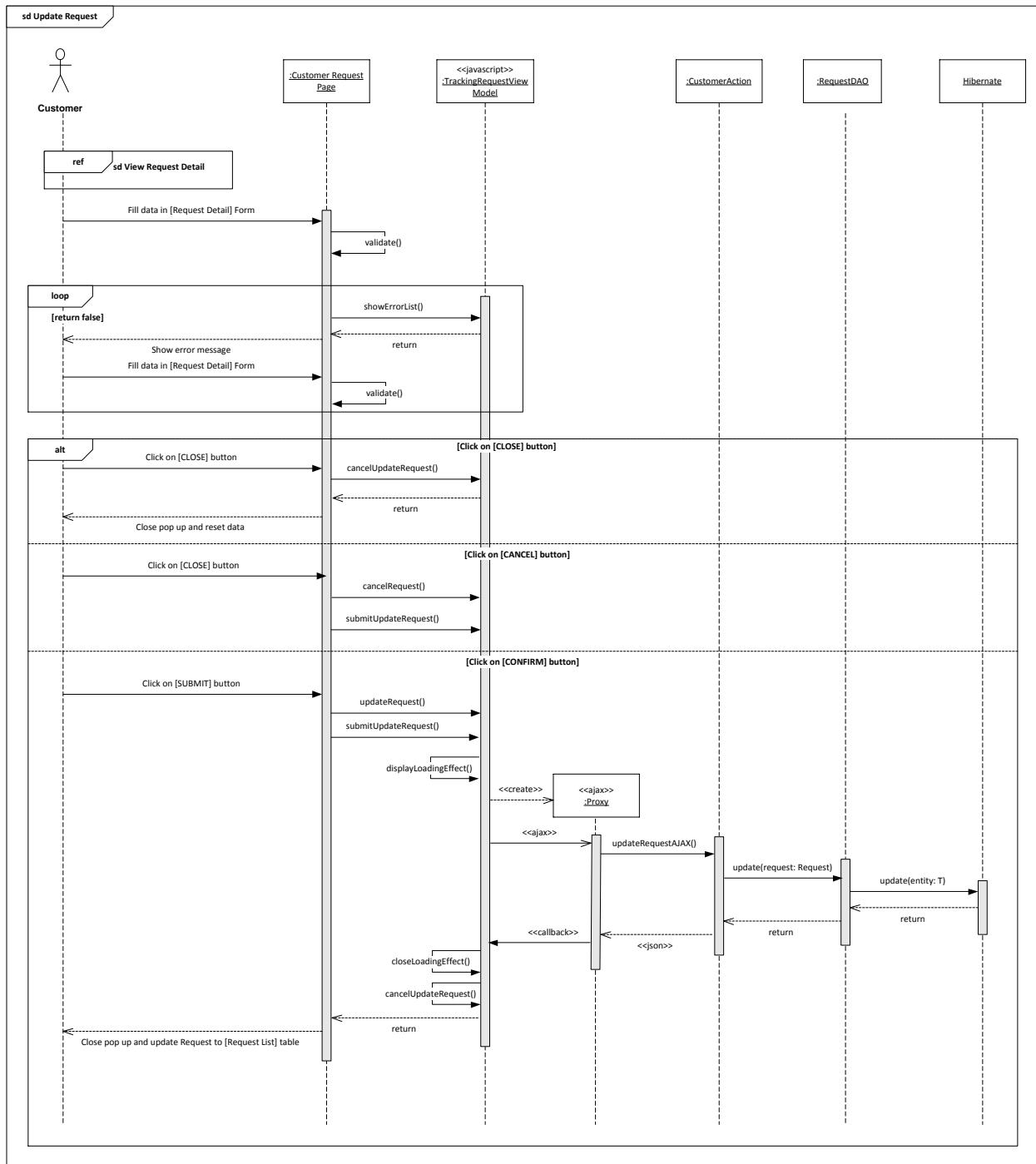


Figure 113-Customer Update Request sequence diagram

5.14. < Staff > [Tracking Task] View All Tasks List

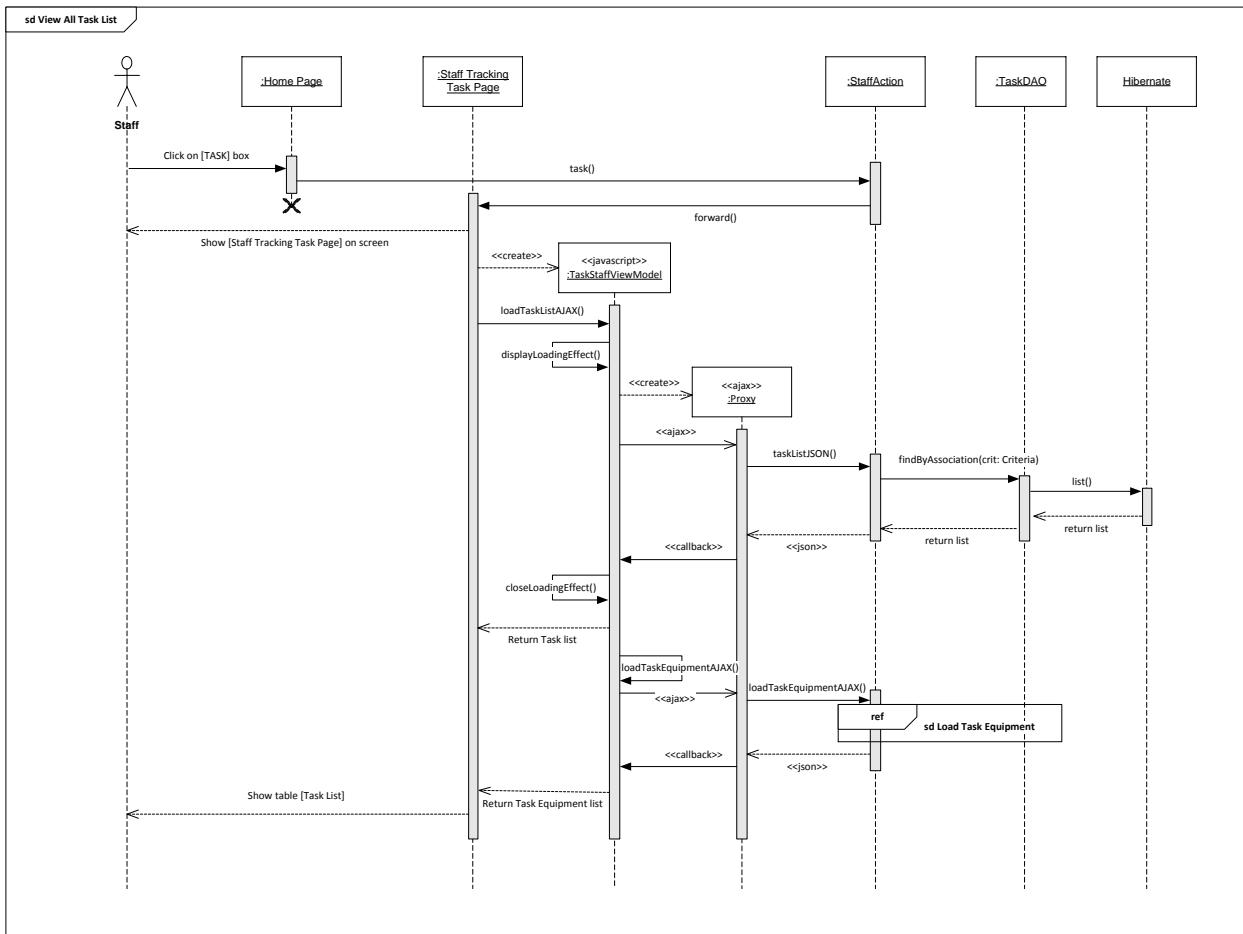


Figure 114-Customer View All Task List sequence diagram

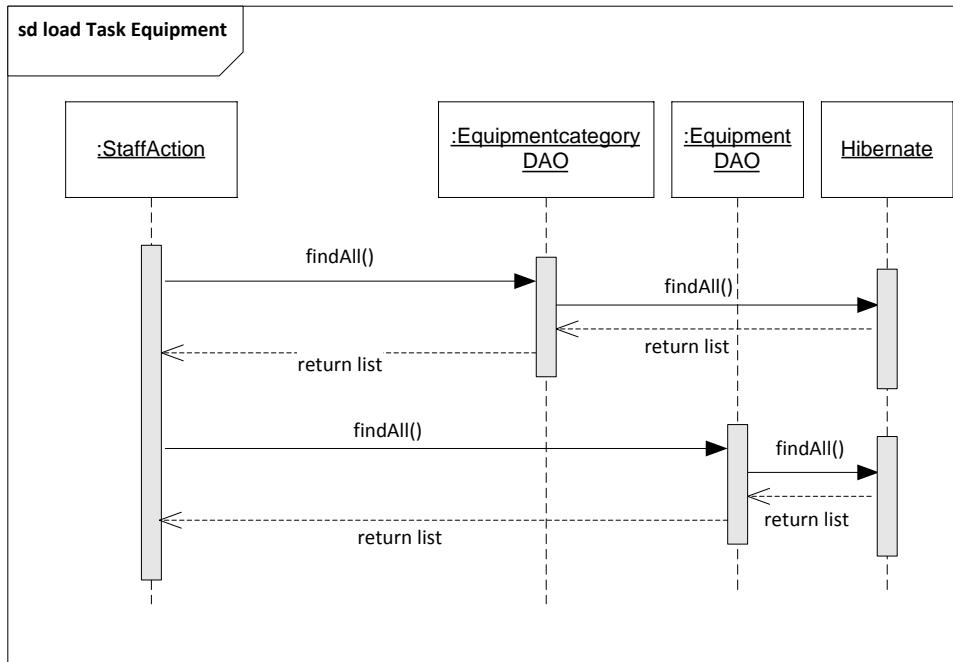


Figure 115-Customer Load Task Equipment sequence diagram

5.15. < Staff > [Tracking Task] View New Tasks List

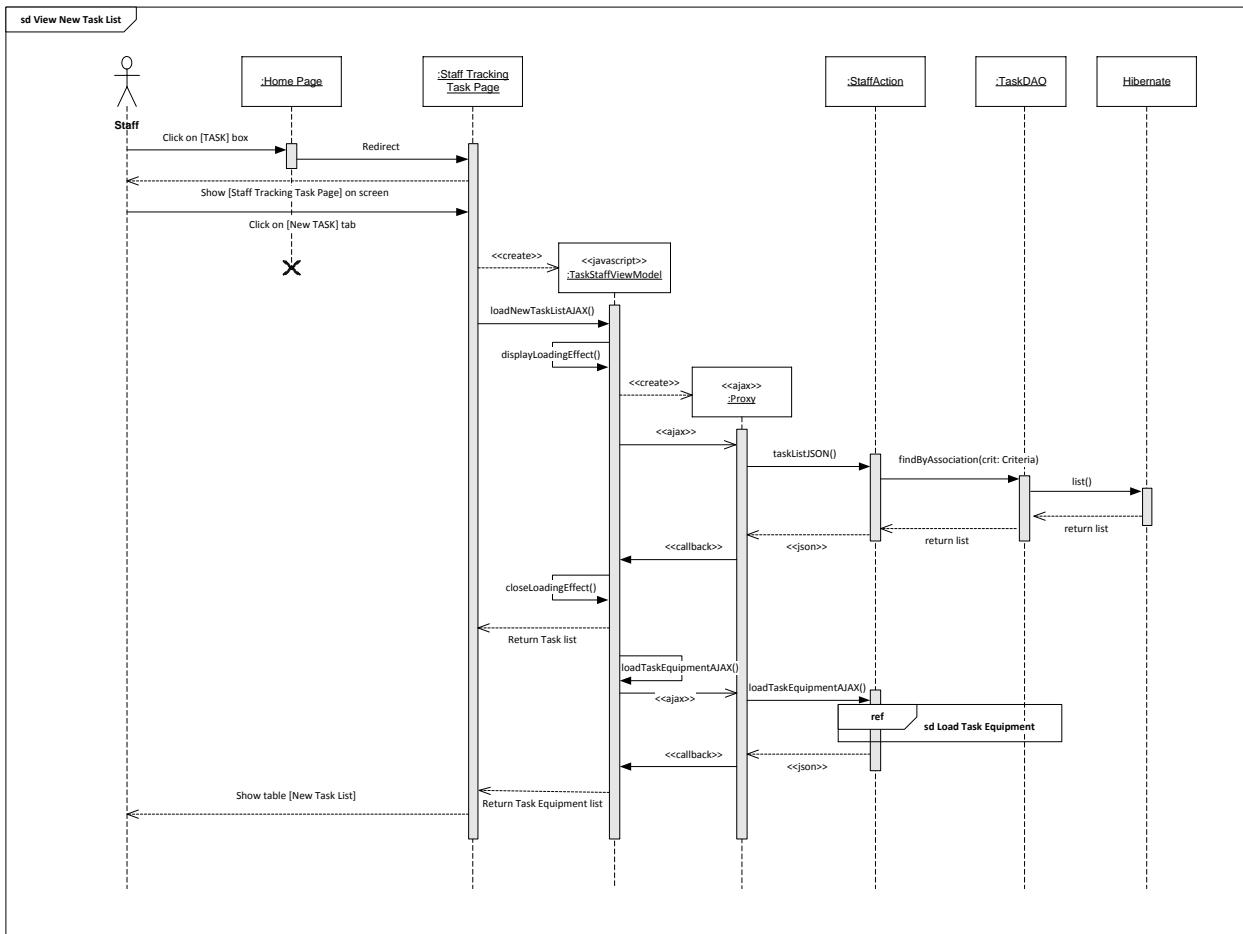


Figure 116-Staff View New Tasks List sequence diagram

5.16. < Staff > [Tracking Task] View In-Progress Tasks List

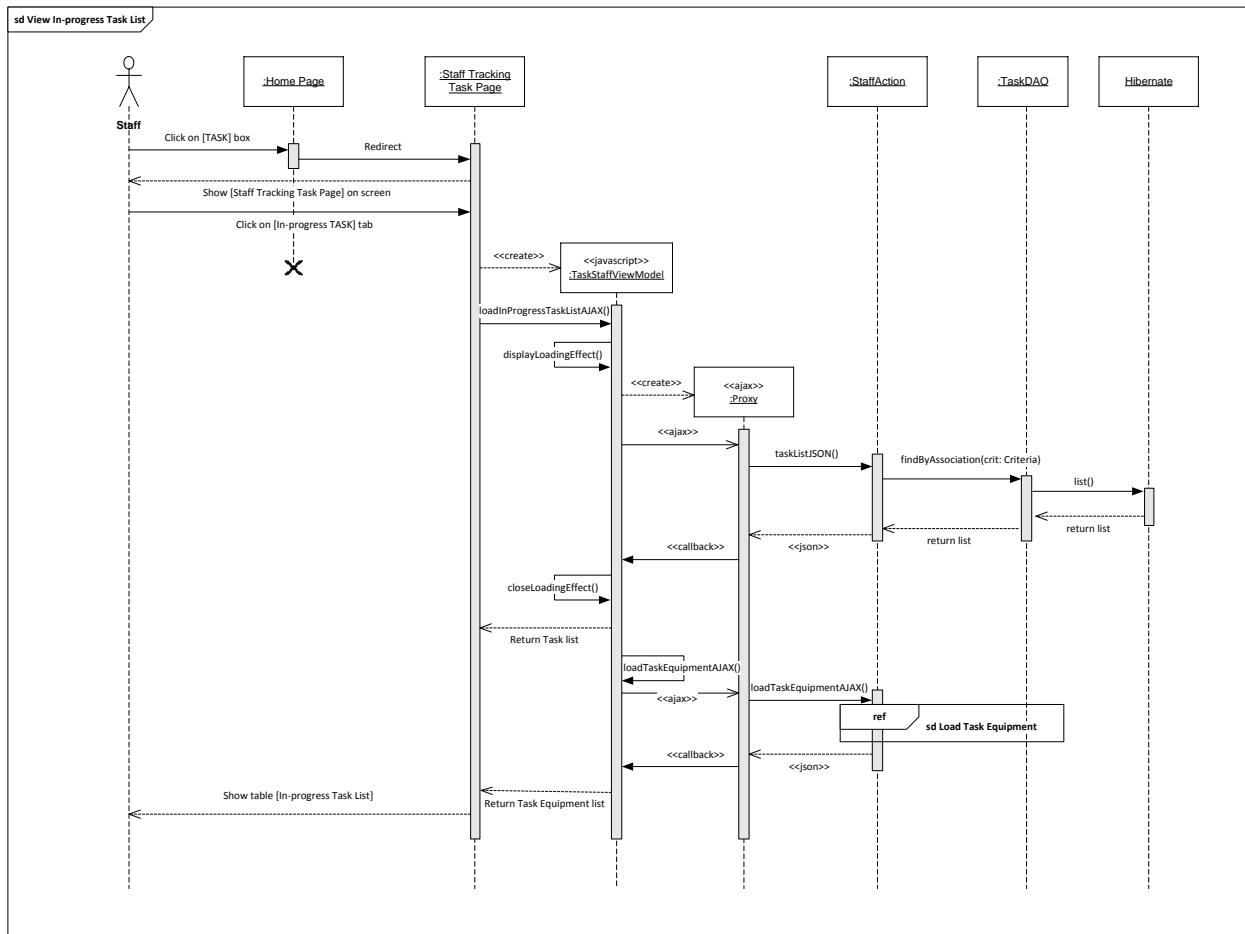


Figure 117-Staff View In-progress Tasks List sequence diagram

5.17. < Staff > [Tracking Task] View Finished Tasks List

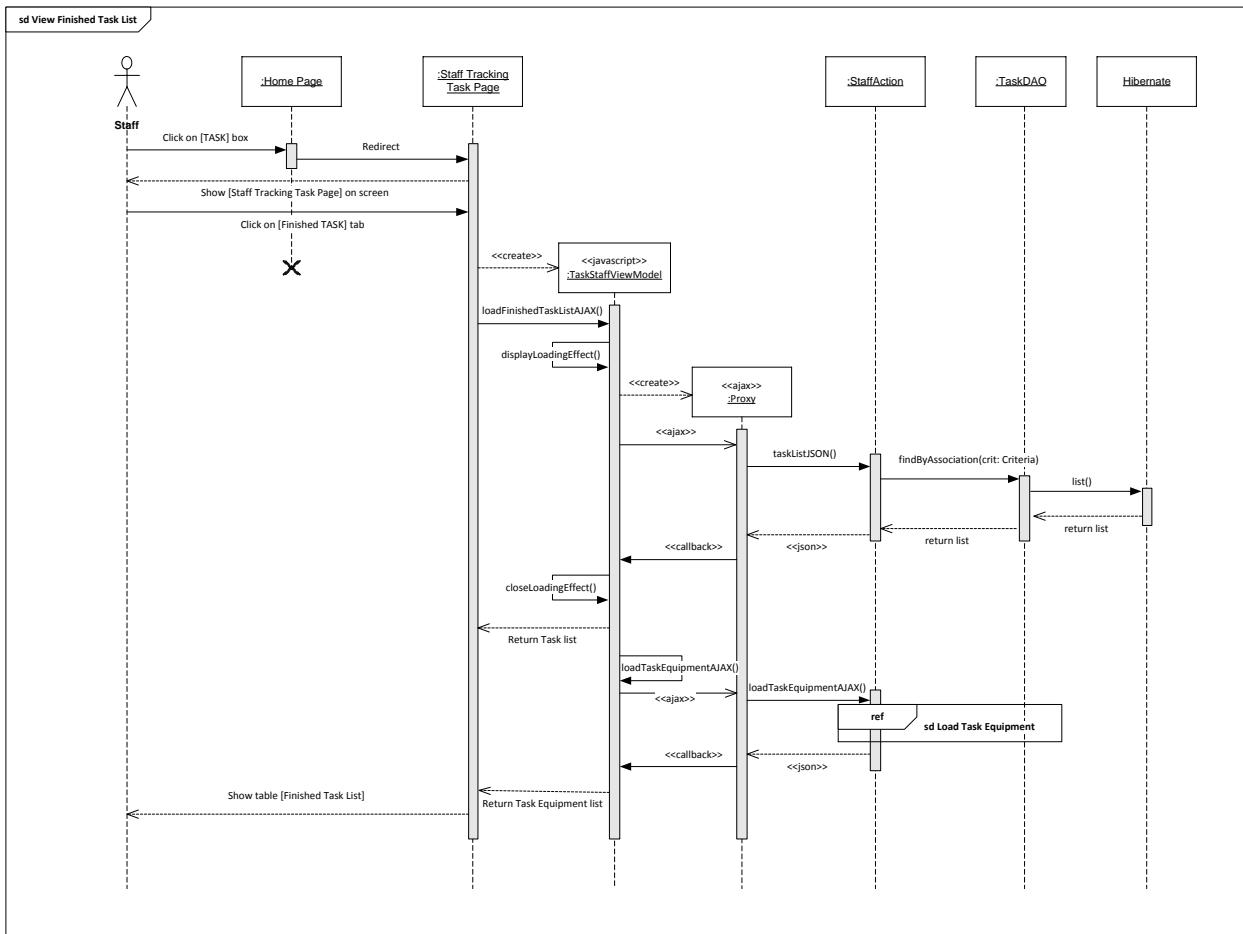


Figure 118-Staff View Finished Tasks List sequence diagram

5.18. < Staff > [Tracking Task] Filter Task

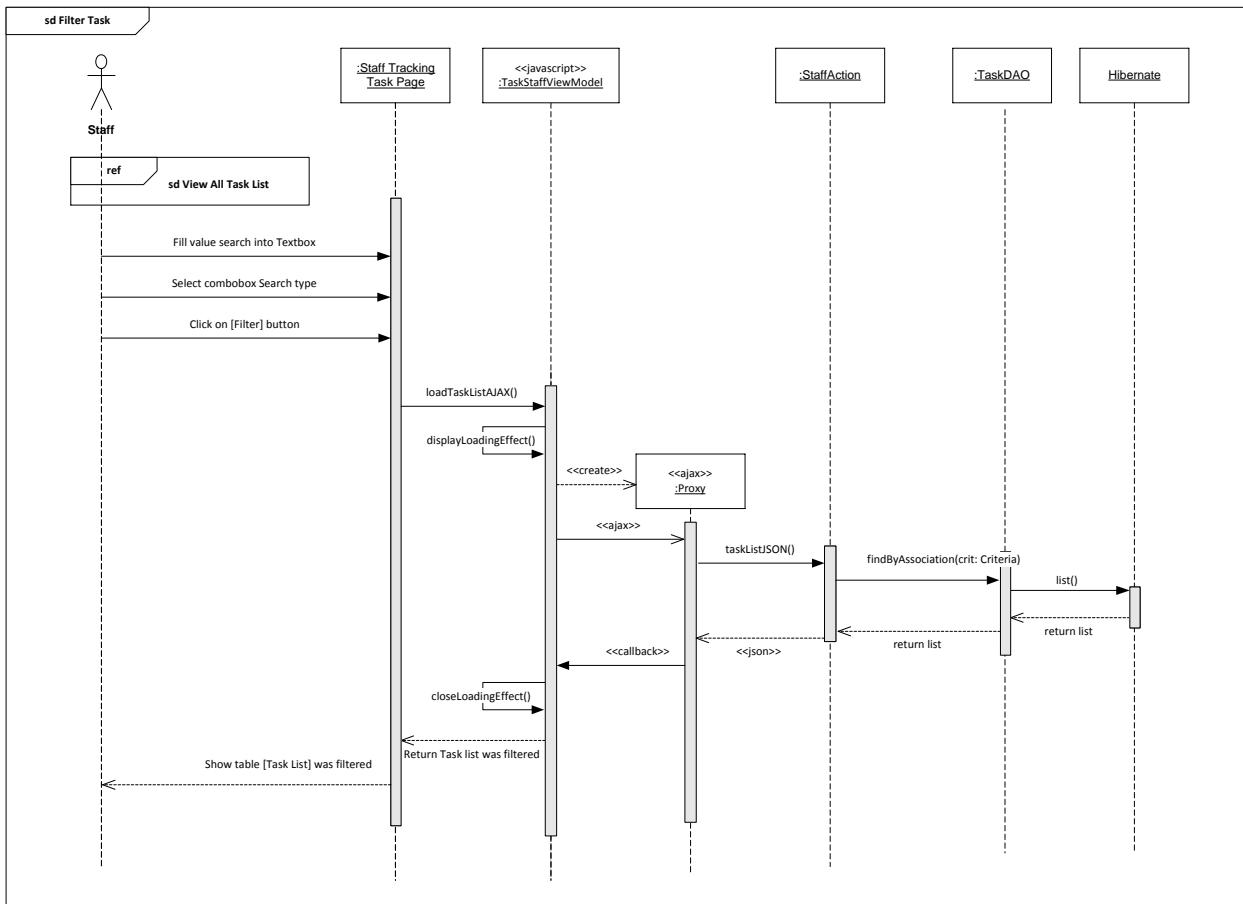
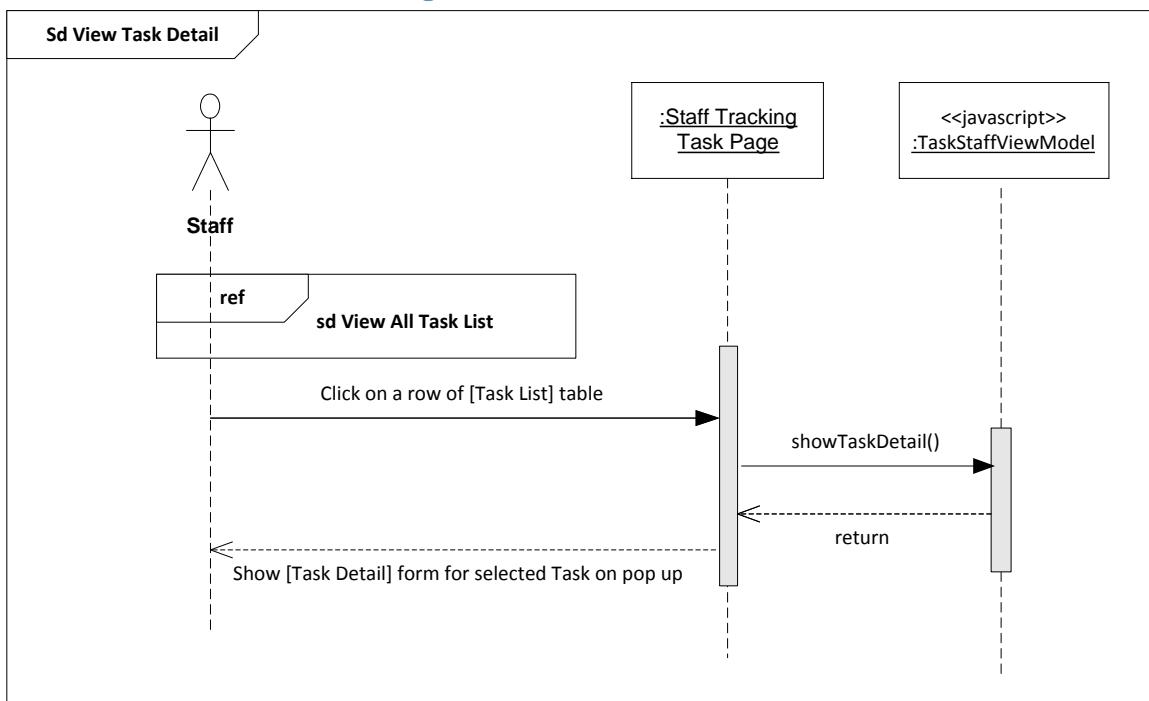


Figure 119-Staff Filter Task sequence diagram

5.19. < Staff > [Tracking Task] View Task Detail**Figure 120-Staff View Task Detail sequence diagram**

5.20. < Staff > [Tracking Task] Update Task

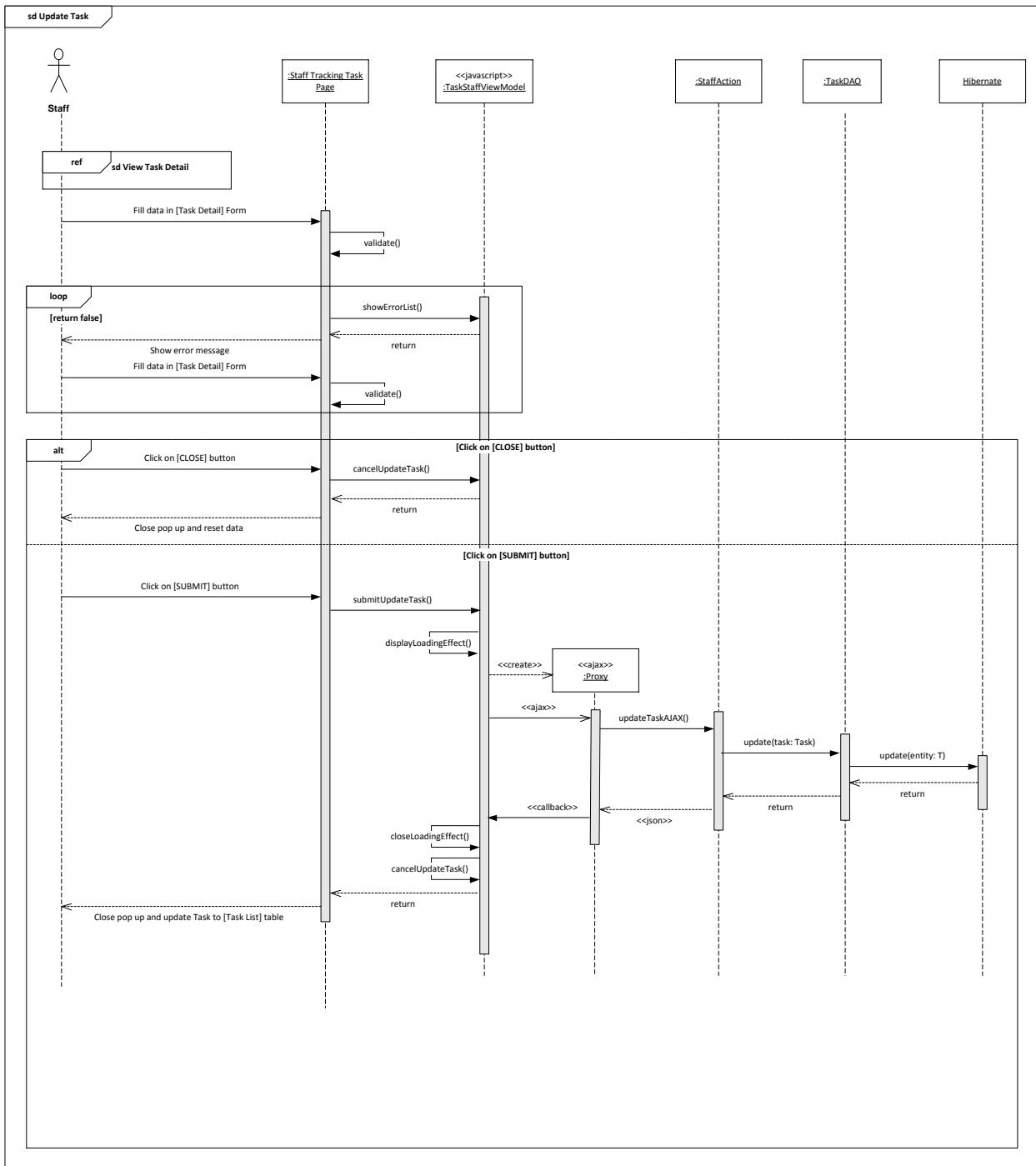


Figure 121-Staff Update Task sequence diagram

5.21. < Technical Manager > [Manage FAQ] Create FAQ

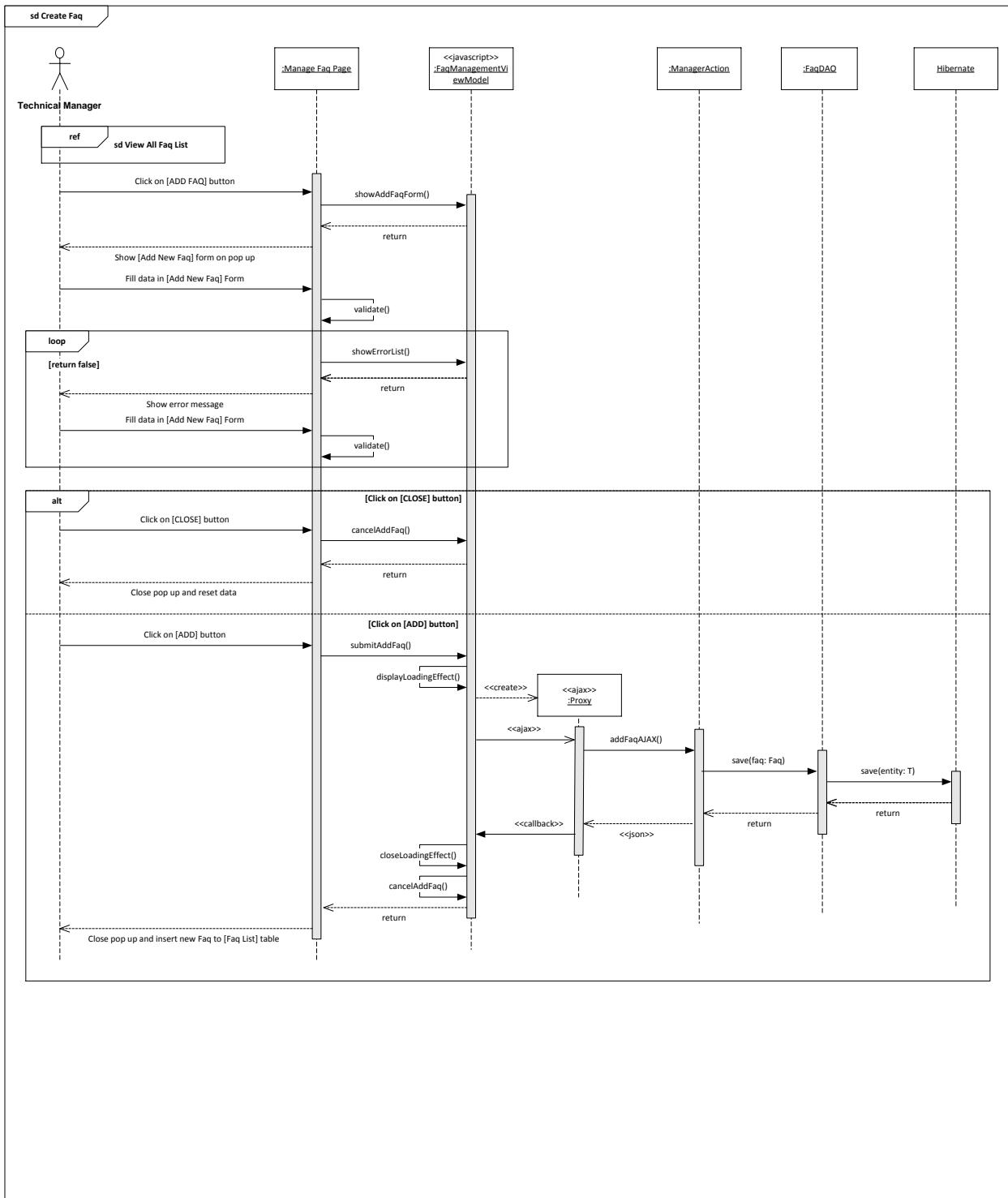


Figure 122-Technical Manager Create FAQ sequence diagram

5.22. < Technical Manager > [Manage FAQ] View all FAQ list

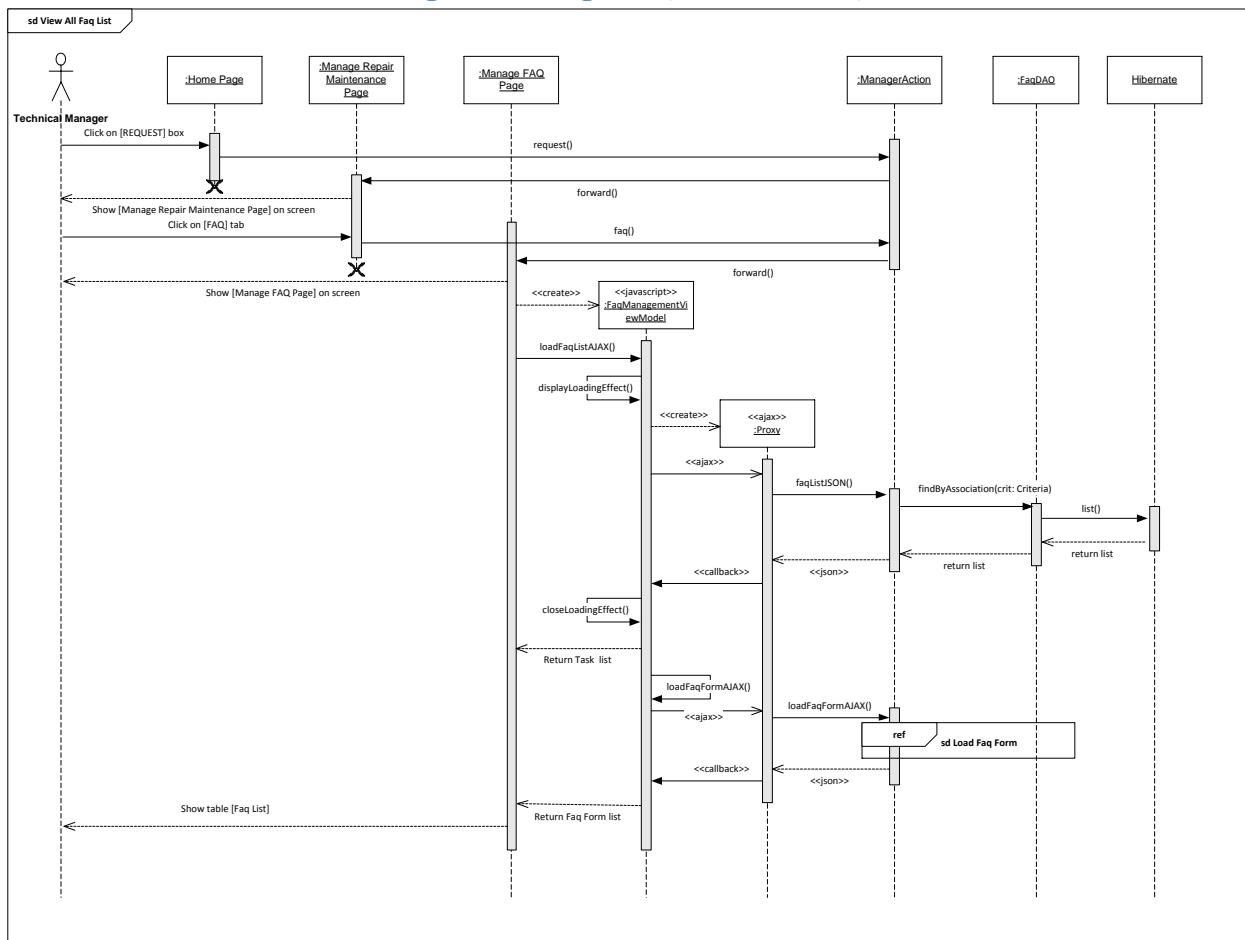


Figure 123-Techical Manager View All FAQs List sequence diagram

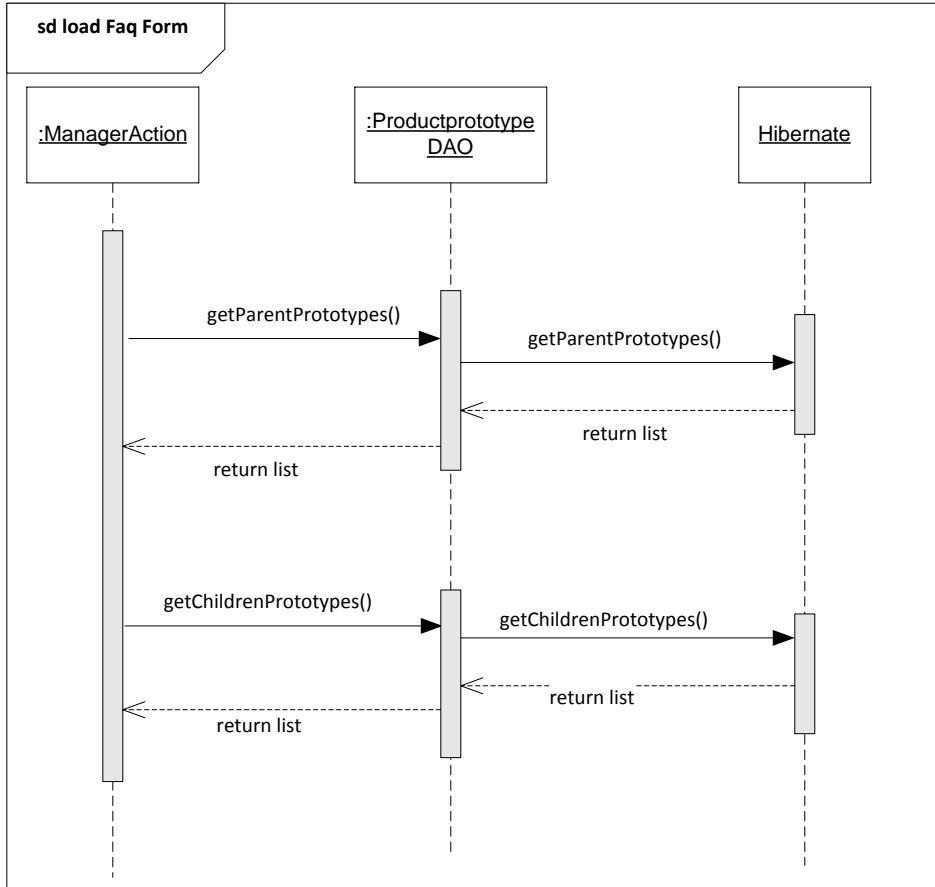


Figure 124-Techical Manager Load FAQ Form sequence diagram

5.23. < Technical Manager > [Manage FAQ] View FAQ detail

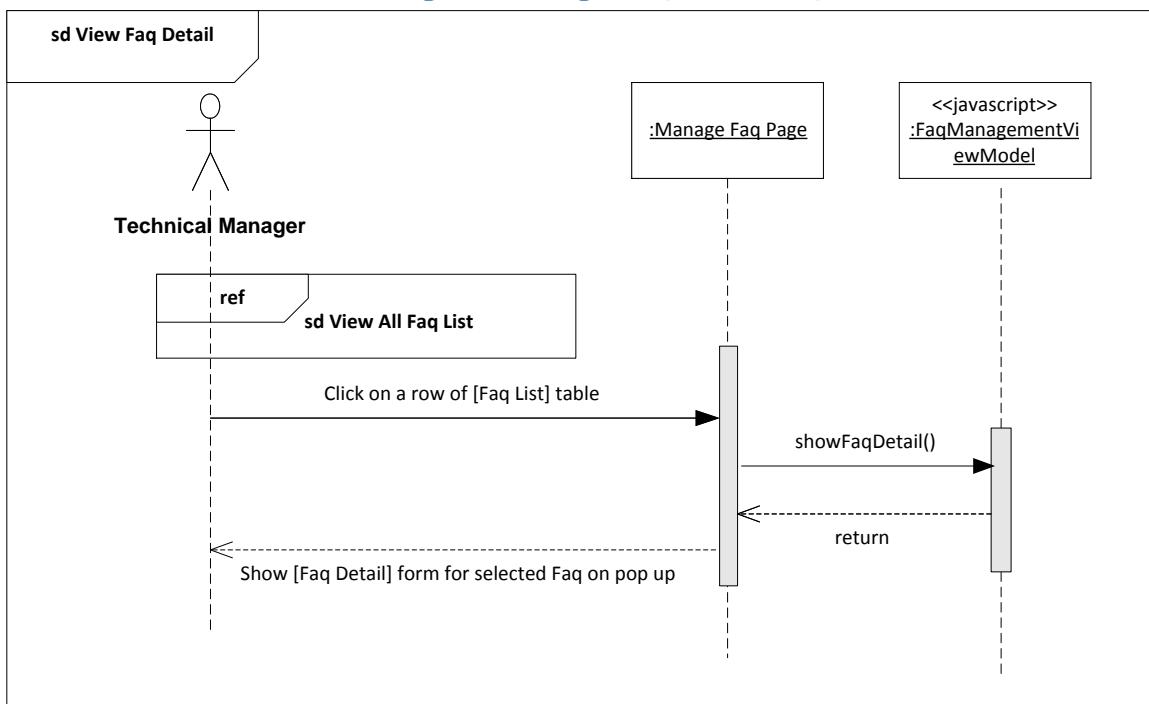


Figure 125-Technical Manager View FAQ Detail sequence diagram

5.24. < Technical Manager > [Manage FAQ] Update FAQ

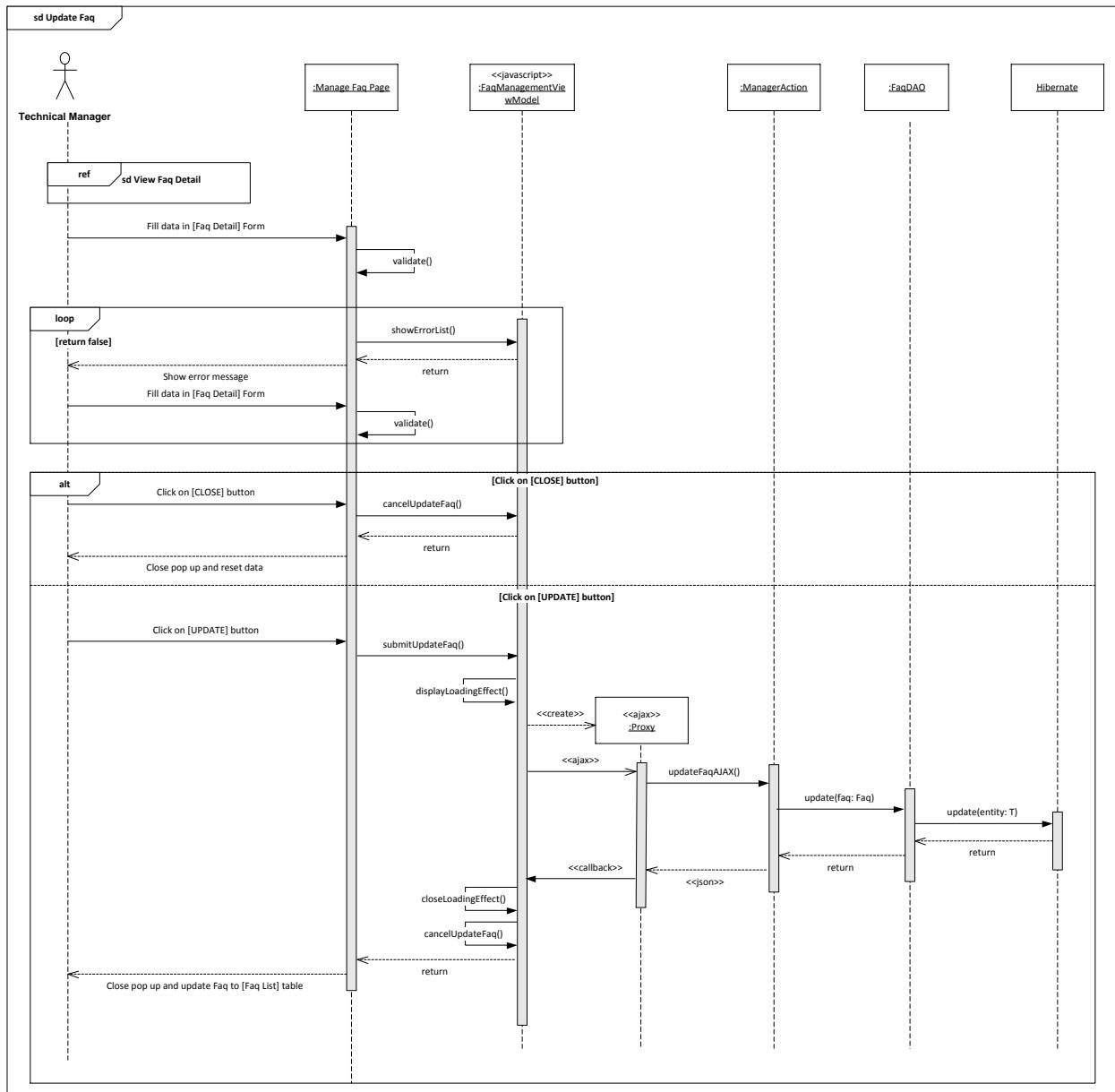


Figure 126-Technical Manager Update FAQ sequence diagram

5.25. < Technical Manager > [Manage repair/maintenance order] Create Request

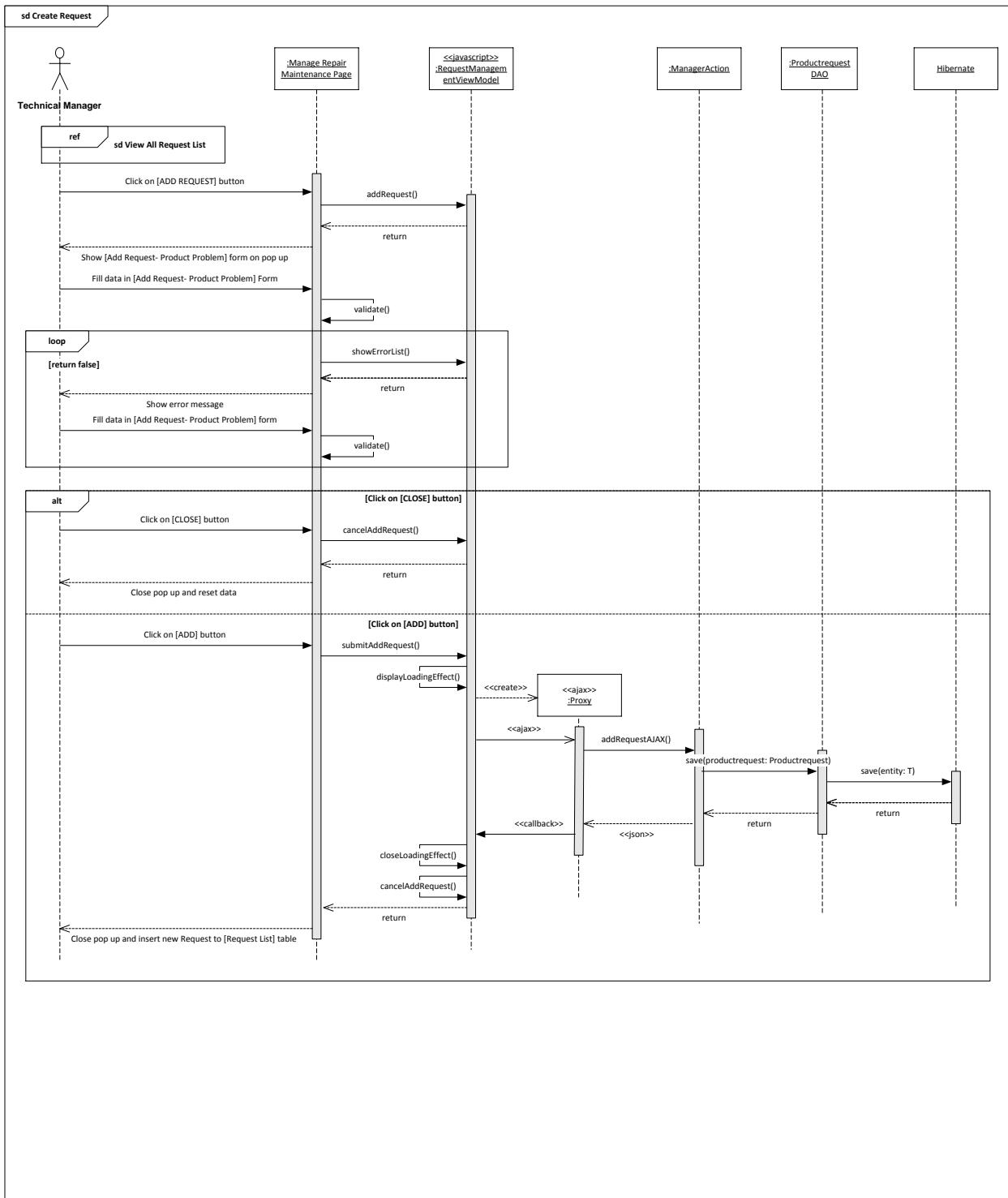


Figure 127-Technical Manager Create Request sequence diagram

5.26. < Technical Manager > [Manage repair/maintenance order] View All Request List

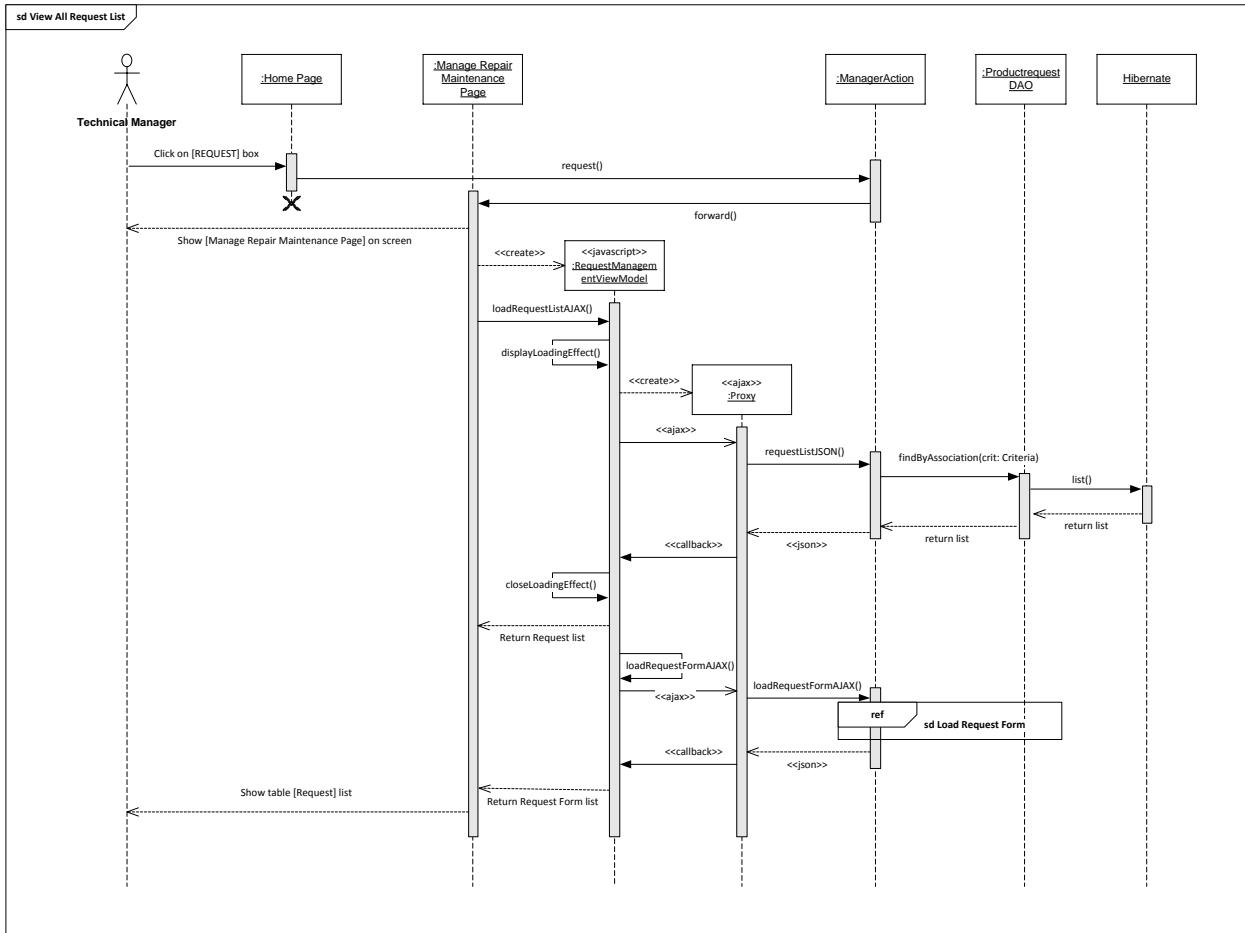


Figure 128-Technical Manager View All Request List sequence diagram

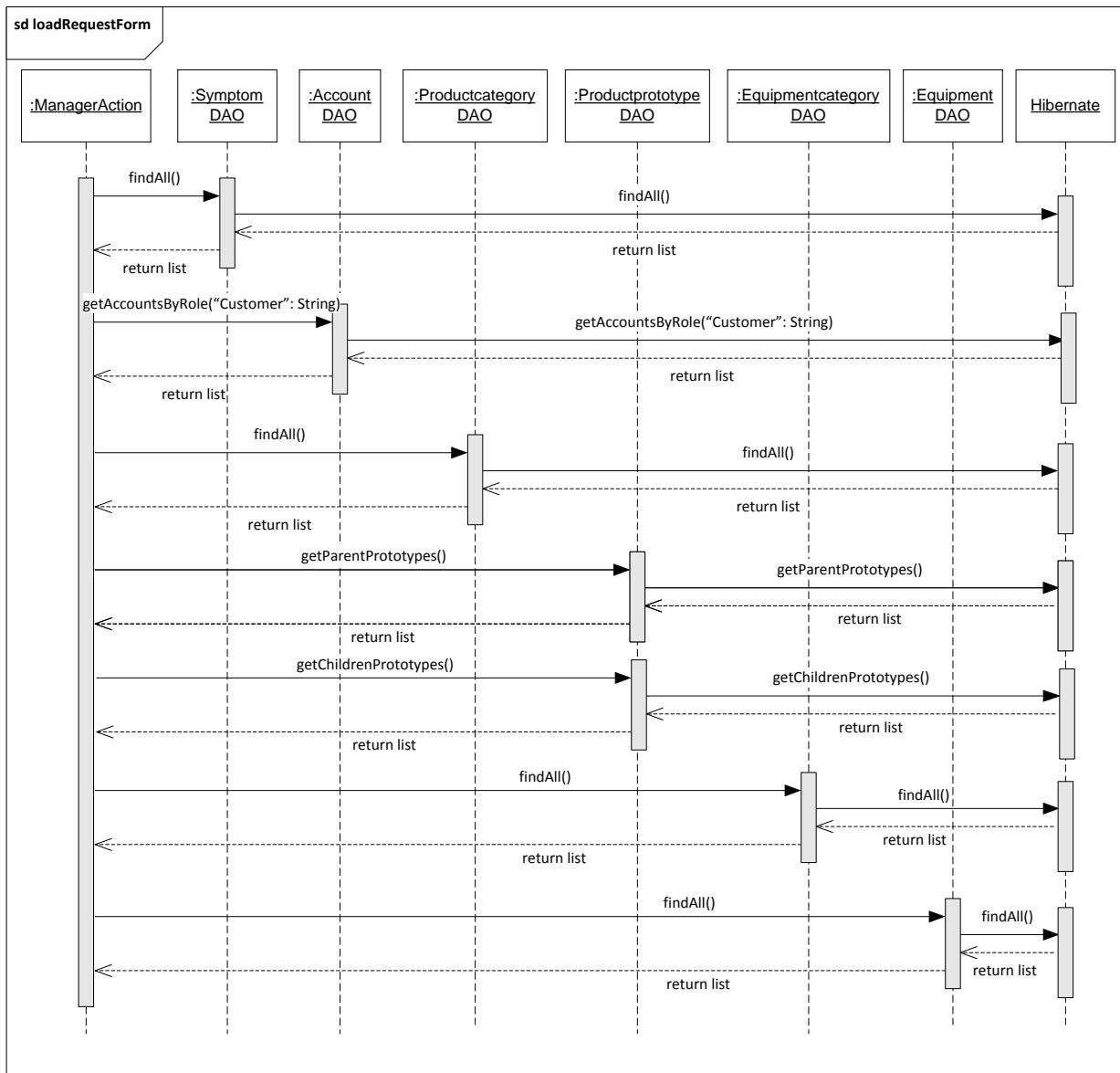


Figure 129-Technical Manager Load Request Form sequence diagram

5.27. < Technical Manager > [Manage repair/maintenance order] Filter Request

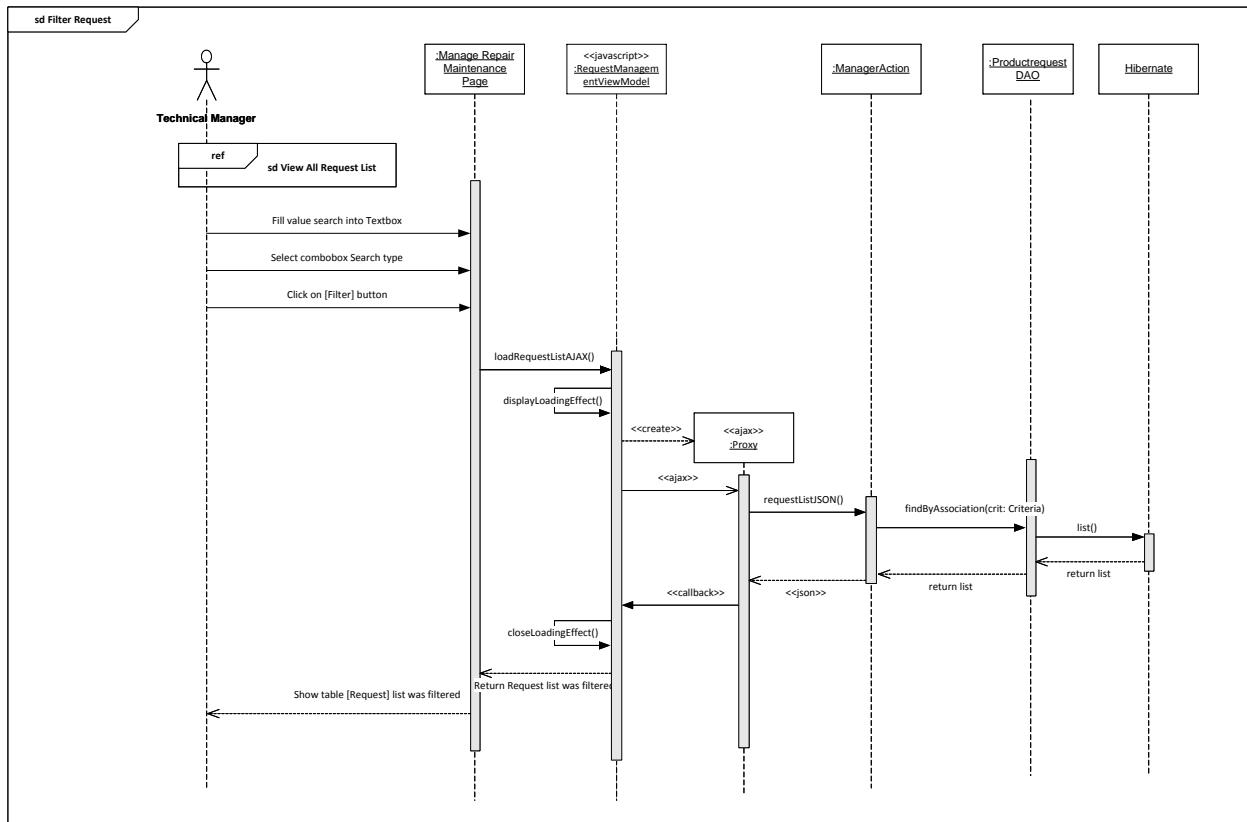


Figure 130-Technical Manager Filter Request sequence diagram

5.28. < Technical Manager > [Manage repair/maintenance order] View Request Detail

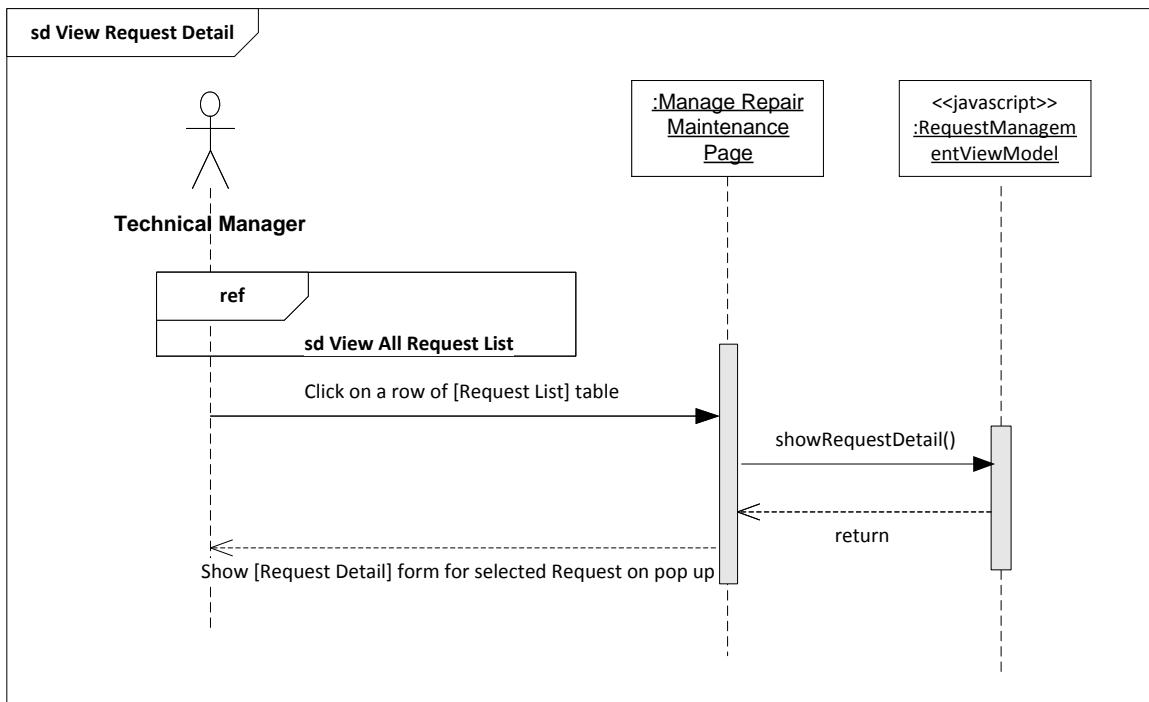


Figure 131-Technical Manager View Request Detail sequence diagram

5.29. < Technical Manager > [Manage repair/maintenance order] Update Request

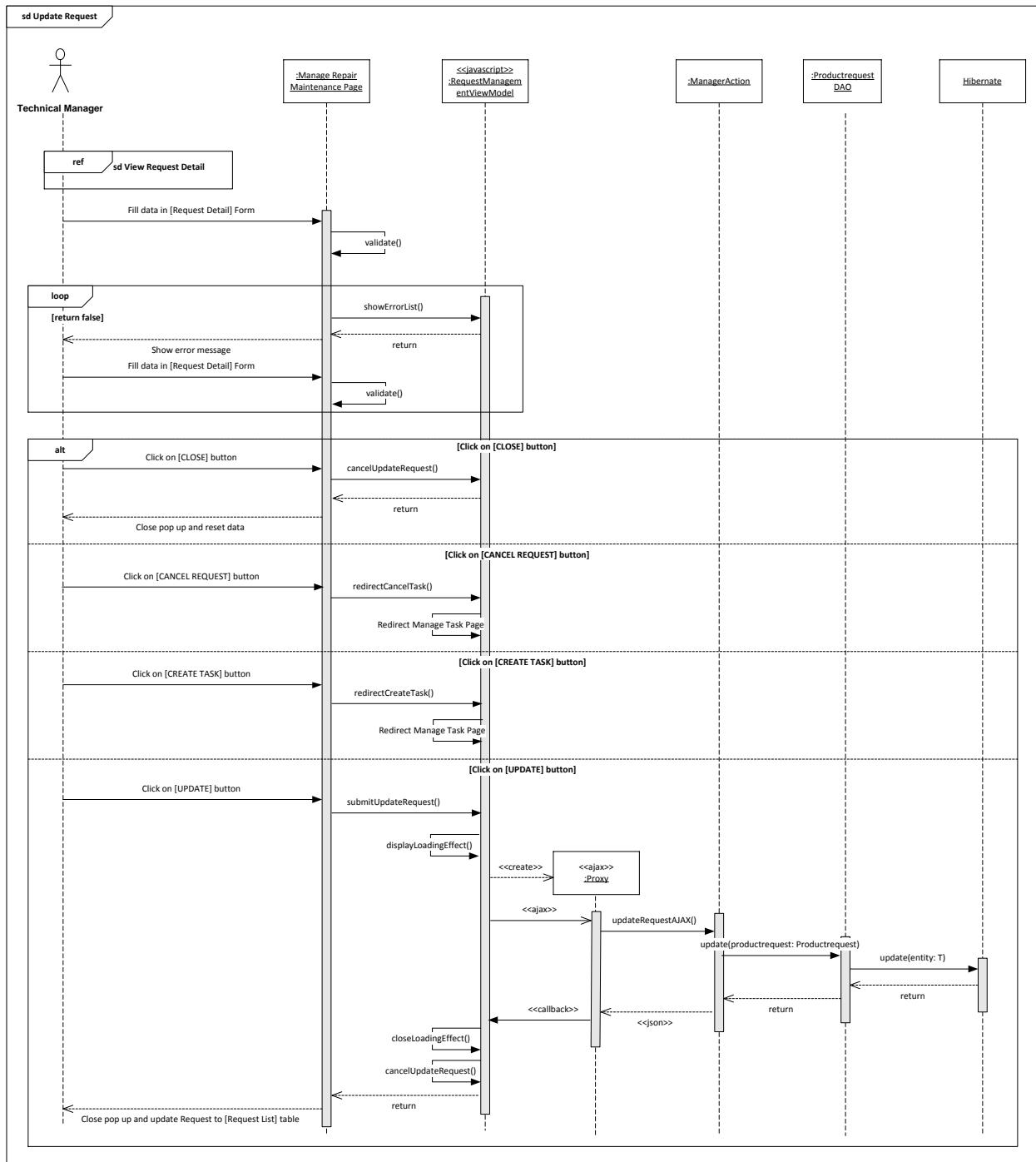


Figure 132-Technical Manager Update Request sequence diagram

5.30. < Technical Manager > [Manage Task] Create Task

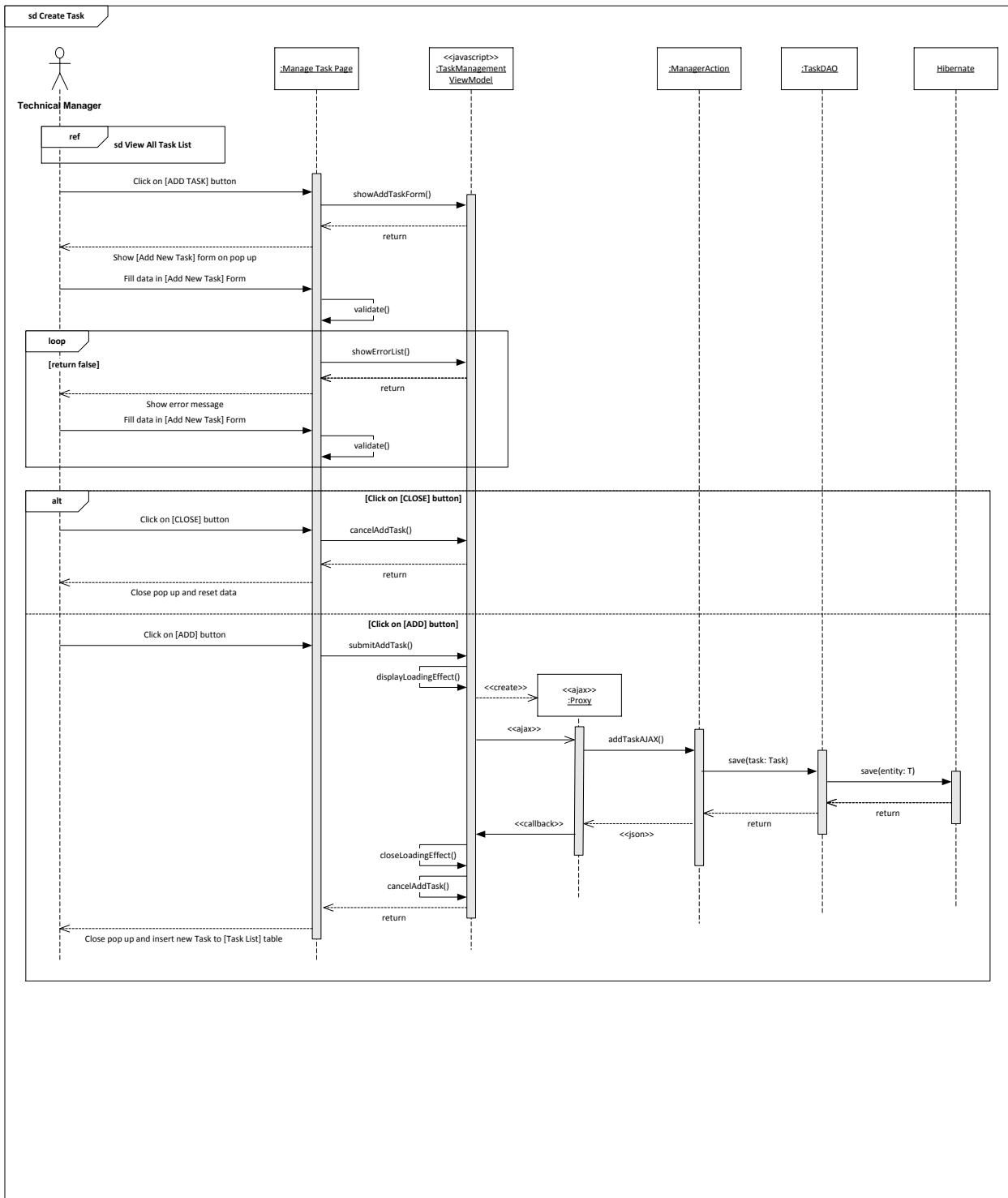


Figure 133-Technical Manager Create Task sequence diagram

5.31. < Technical Manager > [Manage Task] View All Tasks List

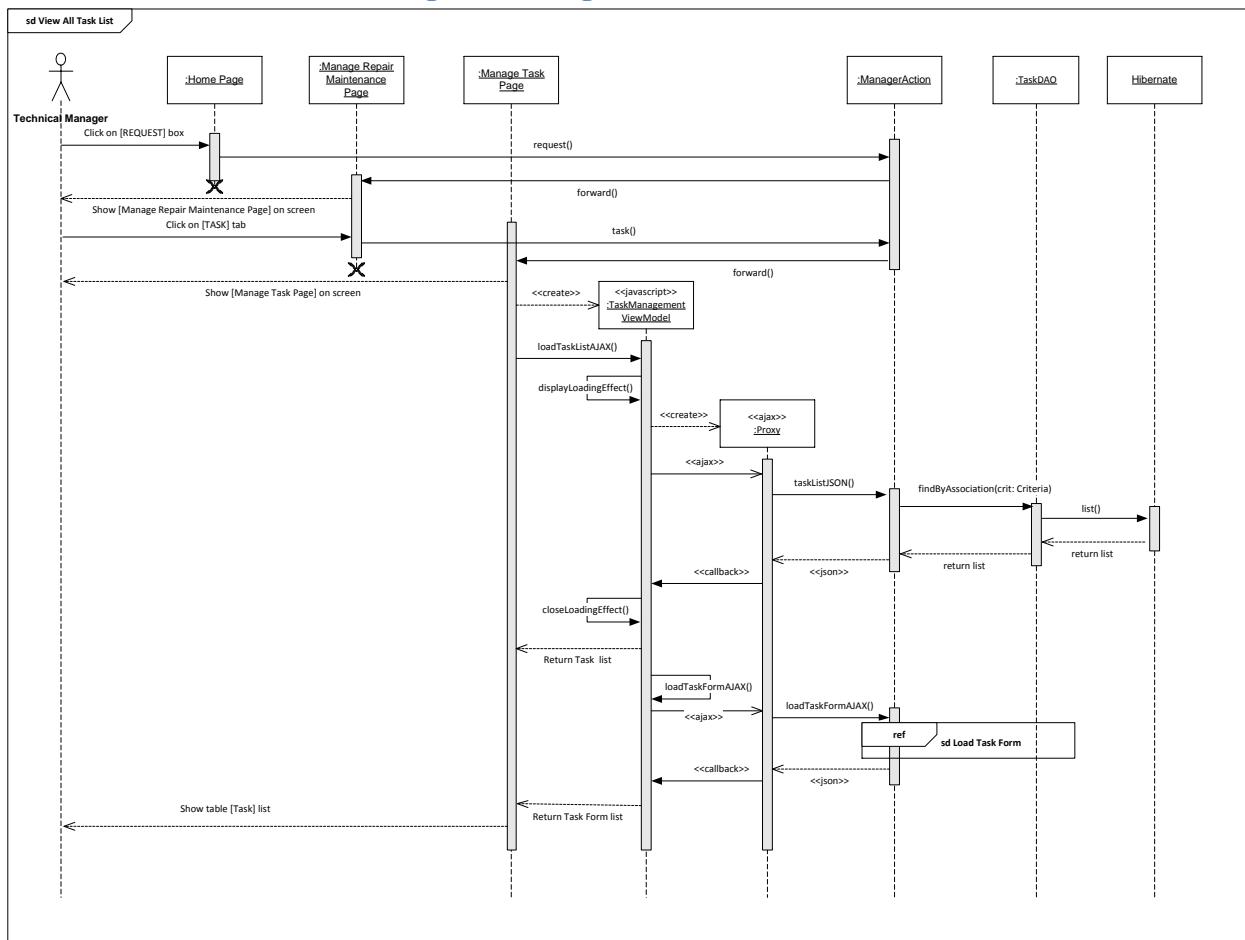


Figure 134-Technical Manager View All Task List sequence diagram

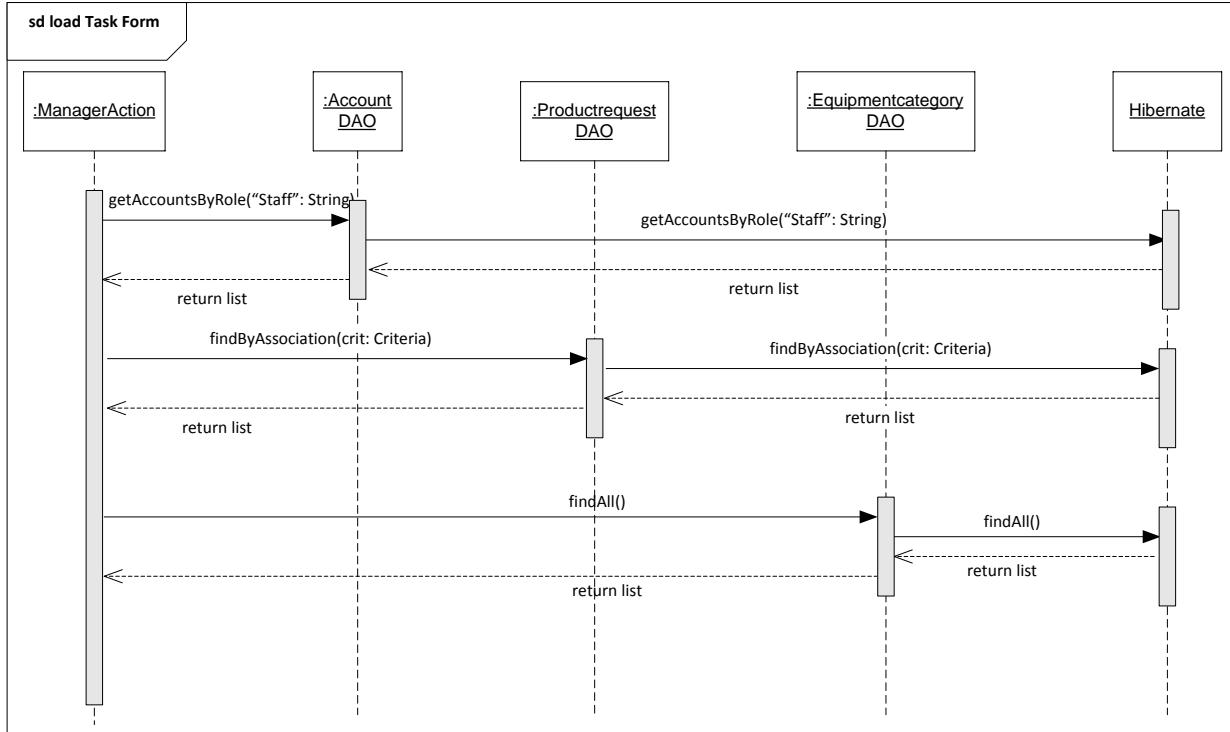


Figure 135-Techical Manager Load Task Form sequence diagram

5.32. < Technical Manager > [Manage Task] Filter Task

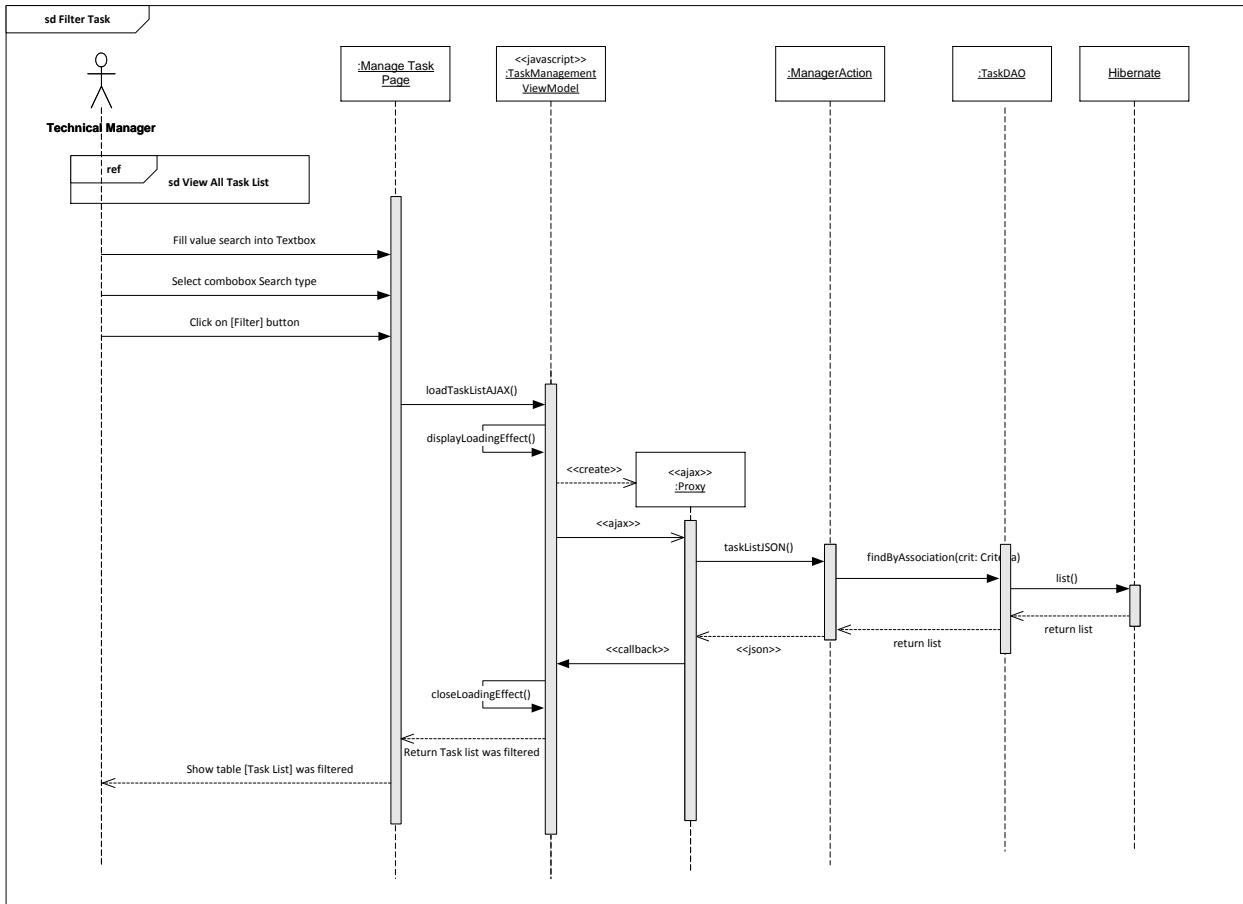


Figure 136-Technical Manager Filter Task sequence diagram

5.33. < Technical Manager > [Manage Task] View Task Detail

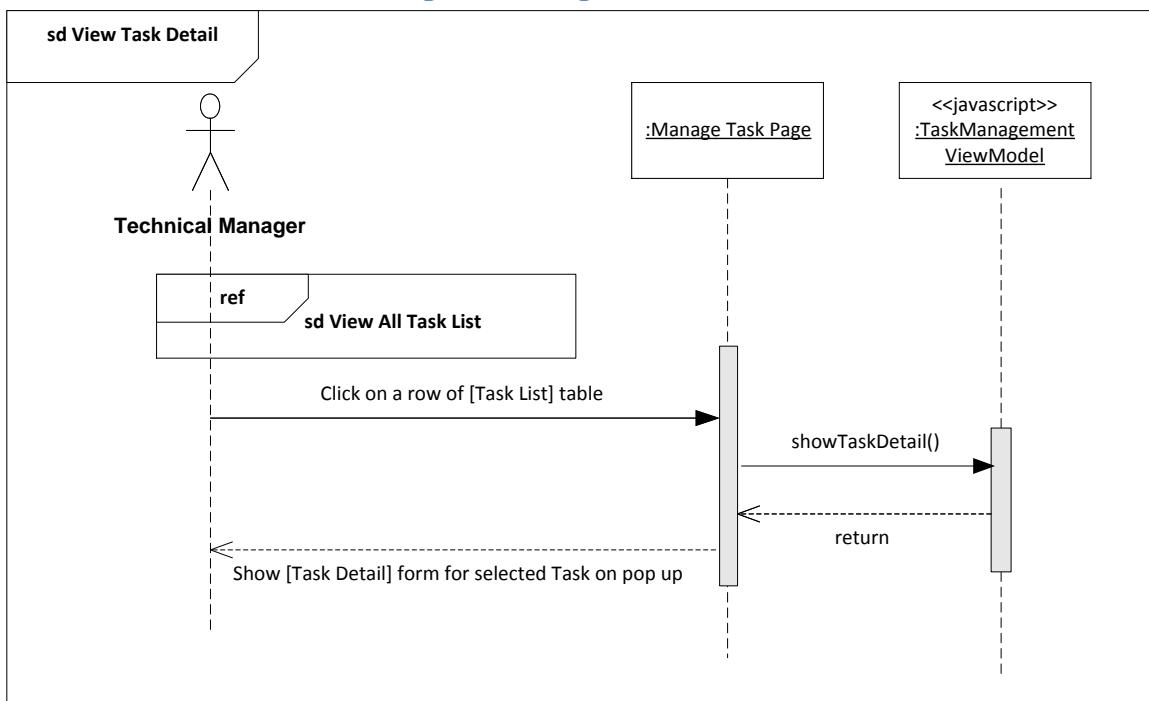


Figure 137-Techical Manager View Task Detail sequence diagram

5.34. < Technical Manager > [Manage Task] Update Task

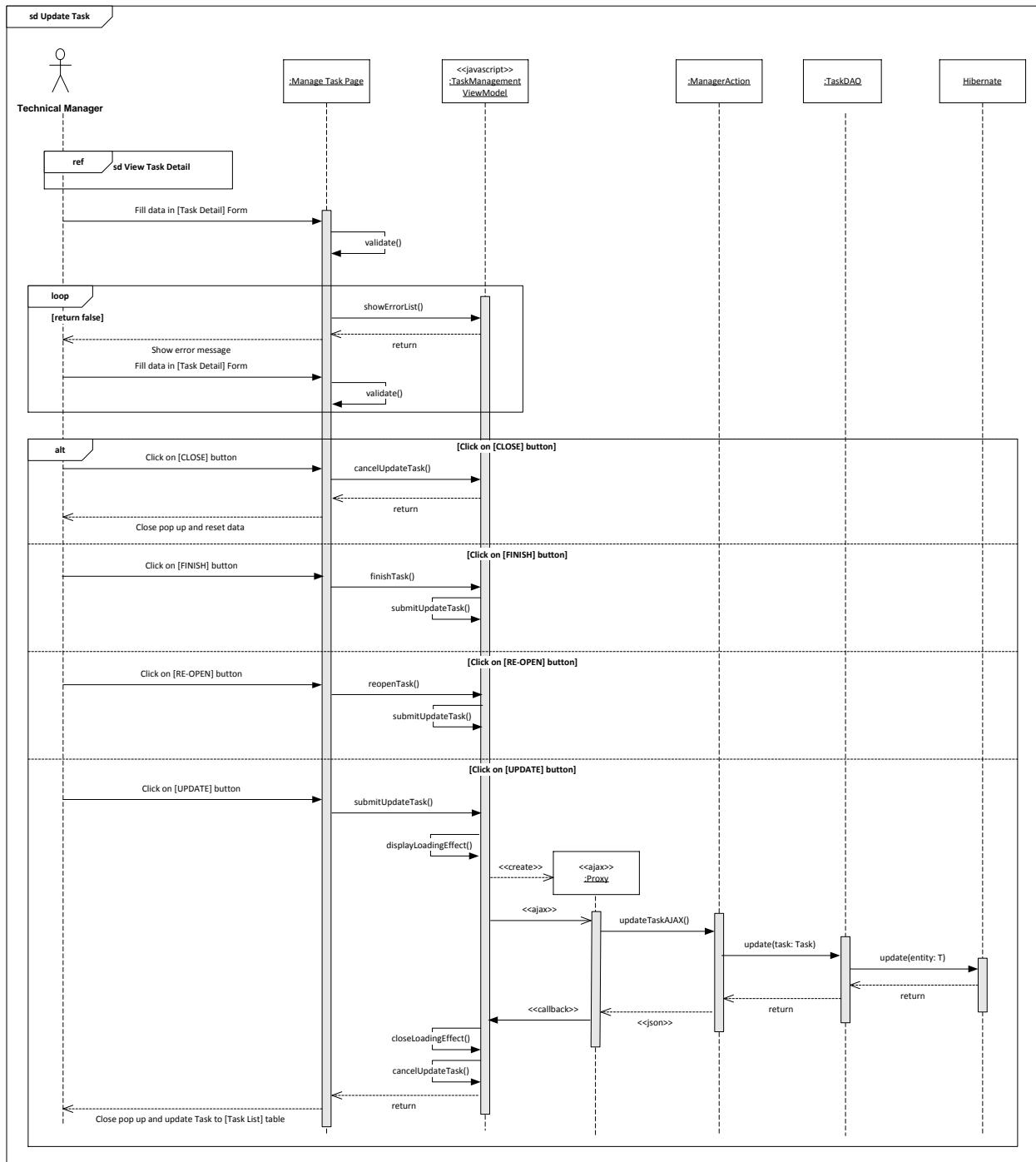


Figure 138-Technical Manager Update Task sequence diagram

5.35. < Technical Manager > [Manage Symptom] Create Symptom

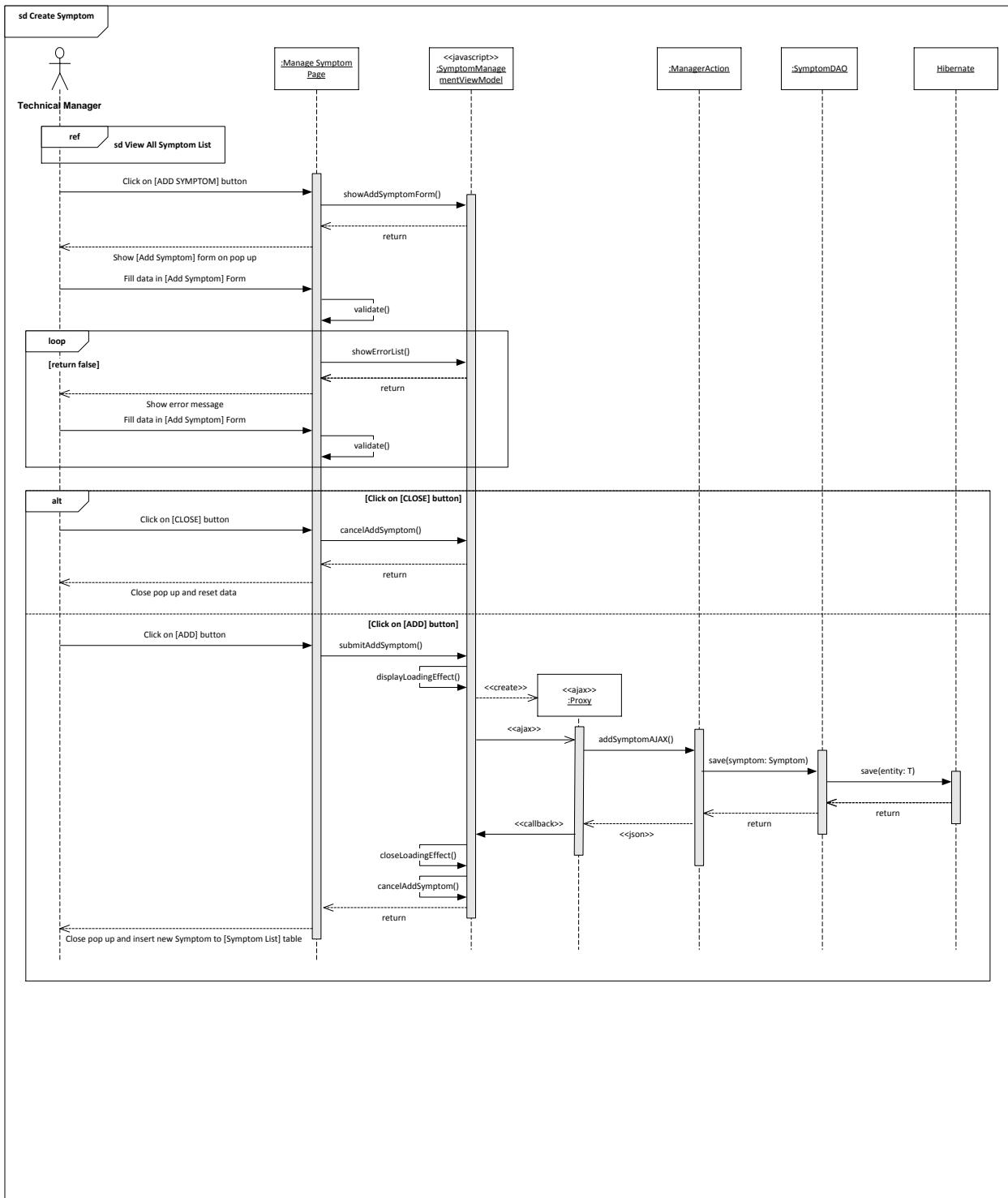


Figure 139-Technical Manager Create Symptom sequence diagram

5.36. < Technical Manager > [Manage Symptom] View All Symptom List

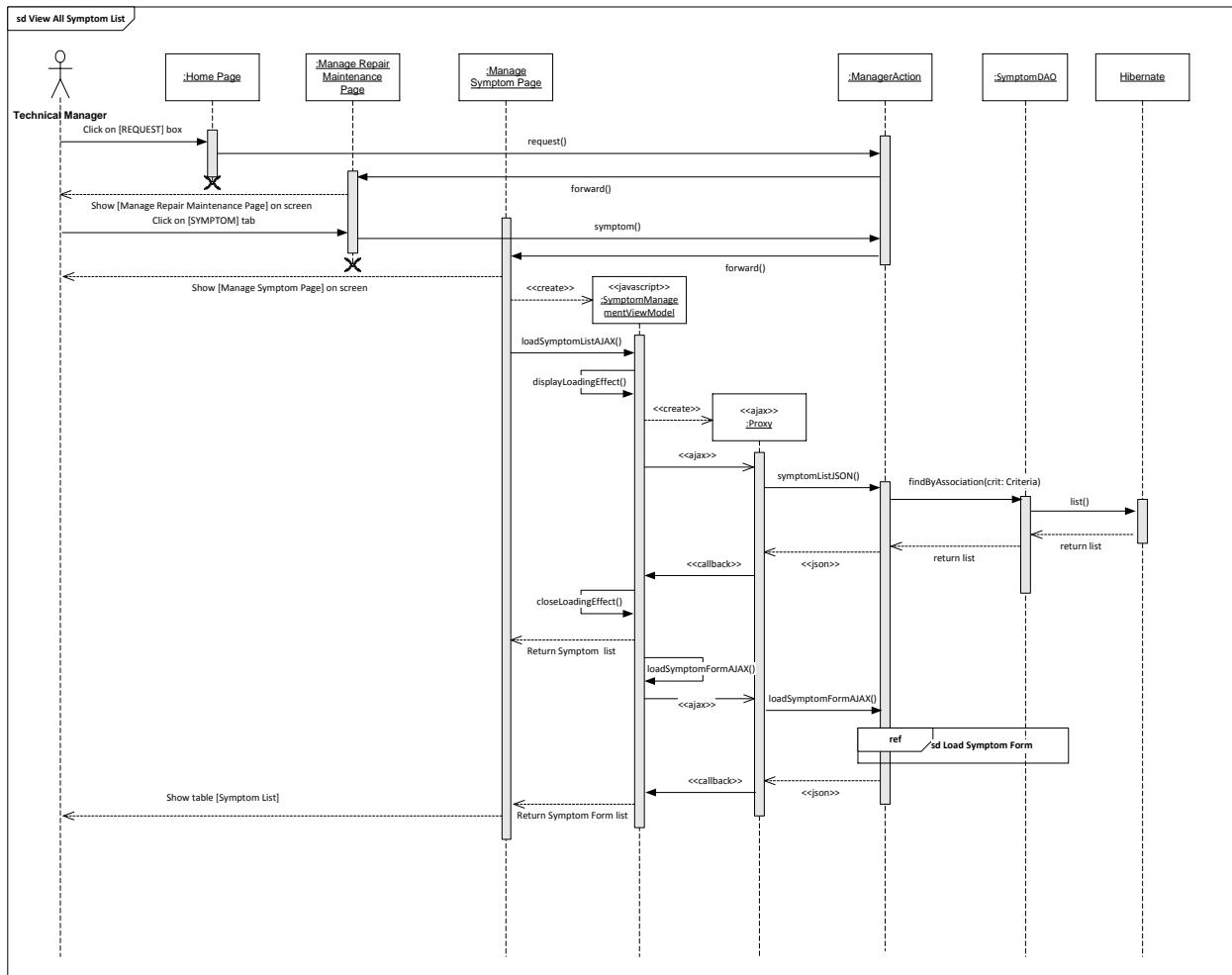


Figure 140-Technical Manager View All Symptom List sequence diagram

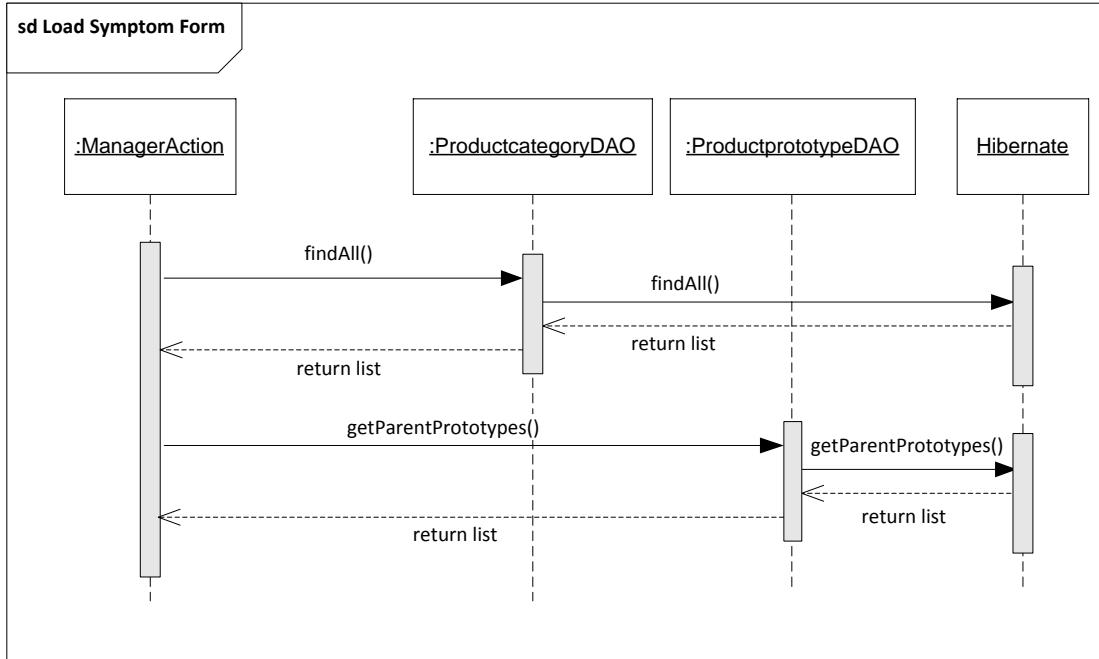


Figure 141-Techical Manager Load Symptom Form sequence diagram

5.37. < Technical Manager > [Manage Symptom] View Symptom Detail

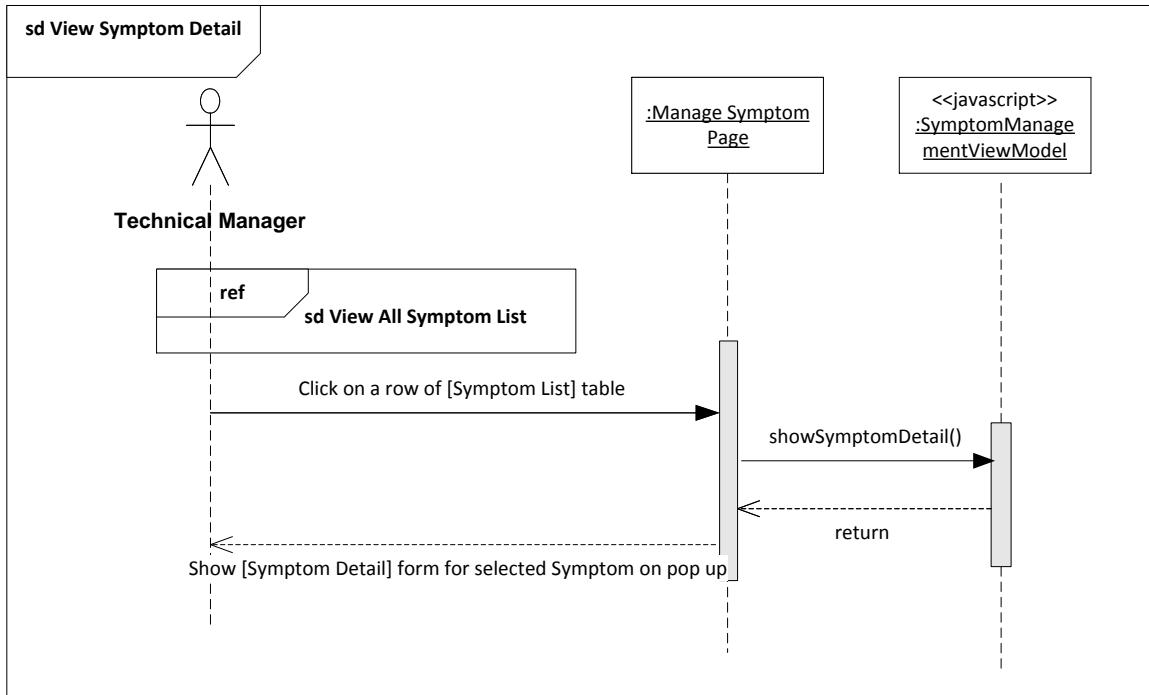


Figure 142-Techical Manager View Symptom Detail sequence diagram

5.38. < Technical Manager > [Manage Symptom] Update Symptom

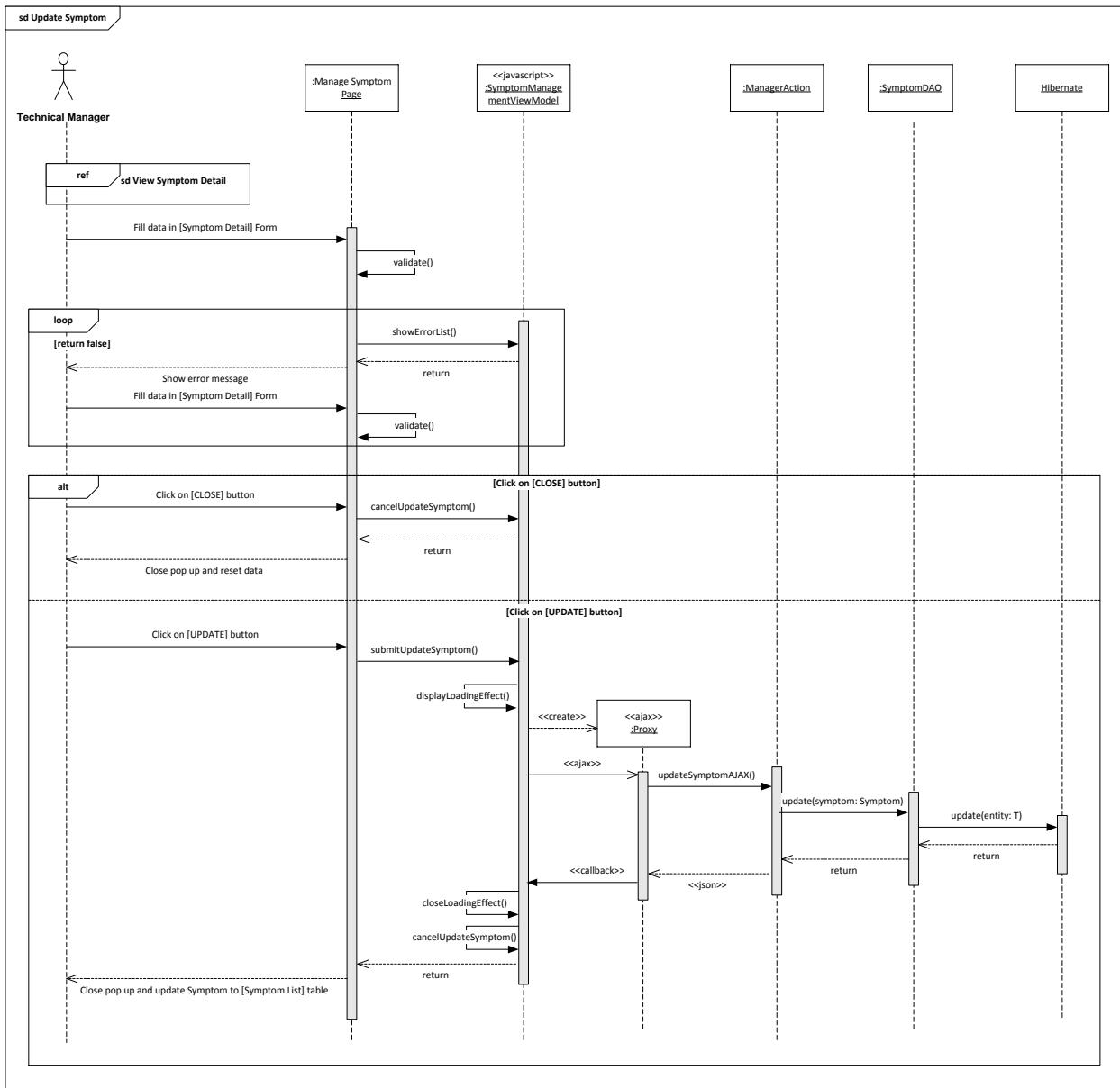


Figure 143-Technical Manager Update Symptom sequence diagram

5.39. < Technical Manager > [Manage Equipment] Add Equipment

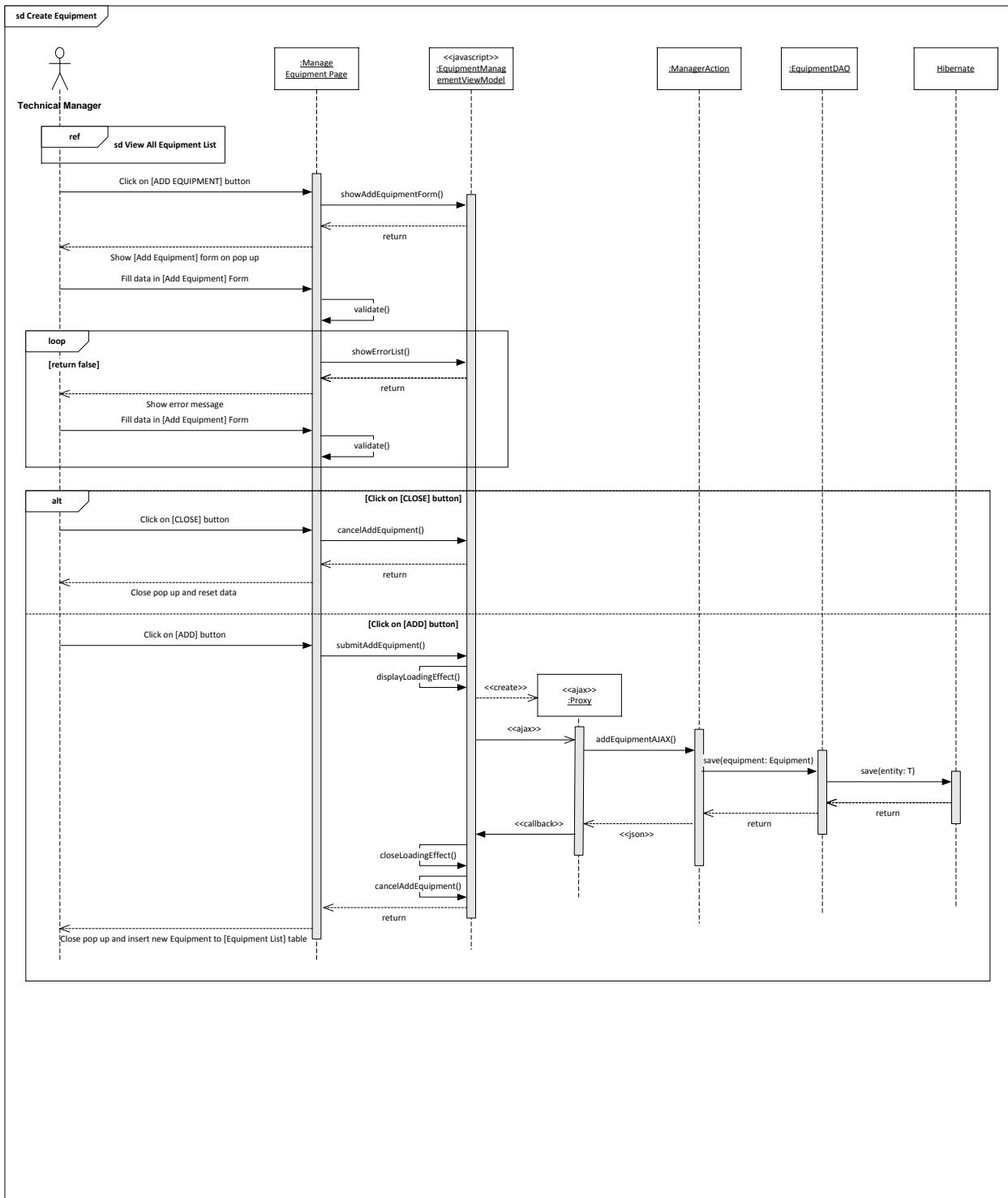


Figure 144-Technical Manager Add Equipment sequence diagram

5.40. < Technical Manager > [Manage Equipment] View All Equipment List

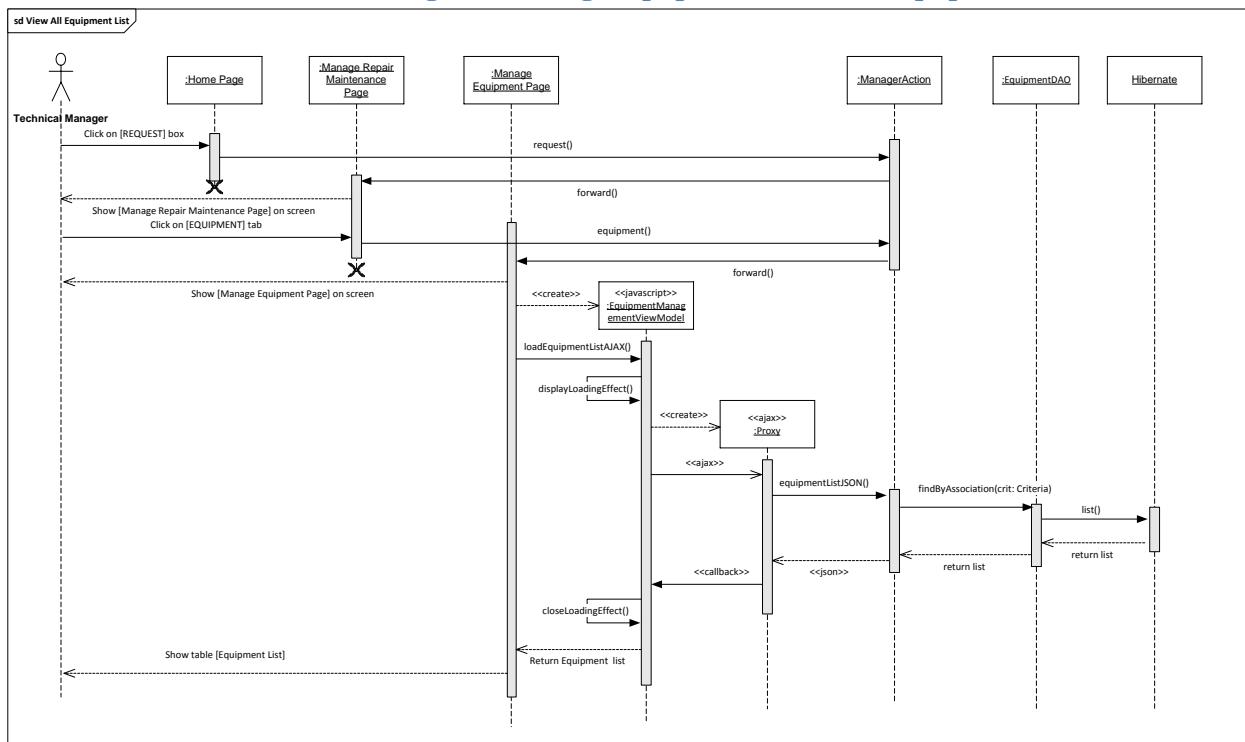


Figure 145-Techical Manager View All Equipment List sequence diagram

5.41. < Technical Manager > [Manage Equipment] View Equipment Detail

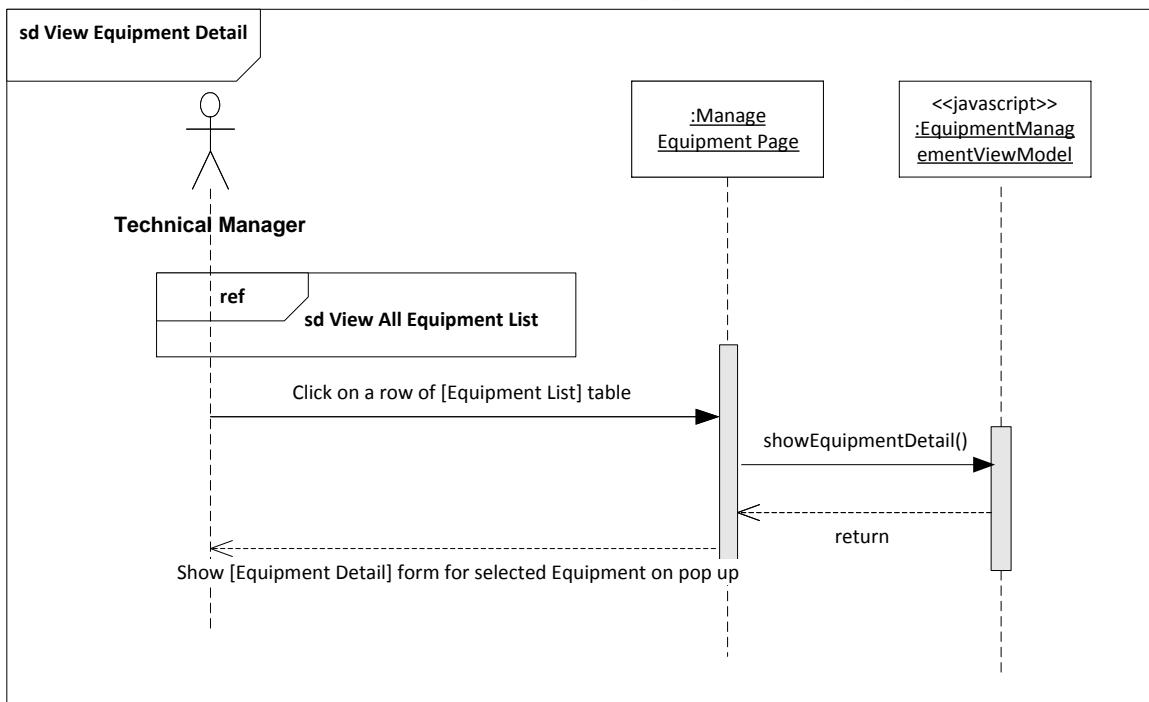


Figure 146-Technical Manager View Equipment Detail sequence diagram

5.42. < Technical Manager > [Manage Equipment] Update Equipment

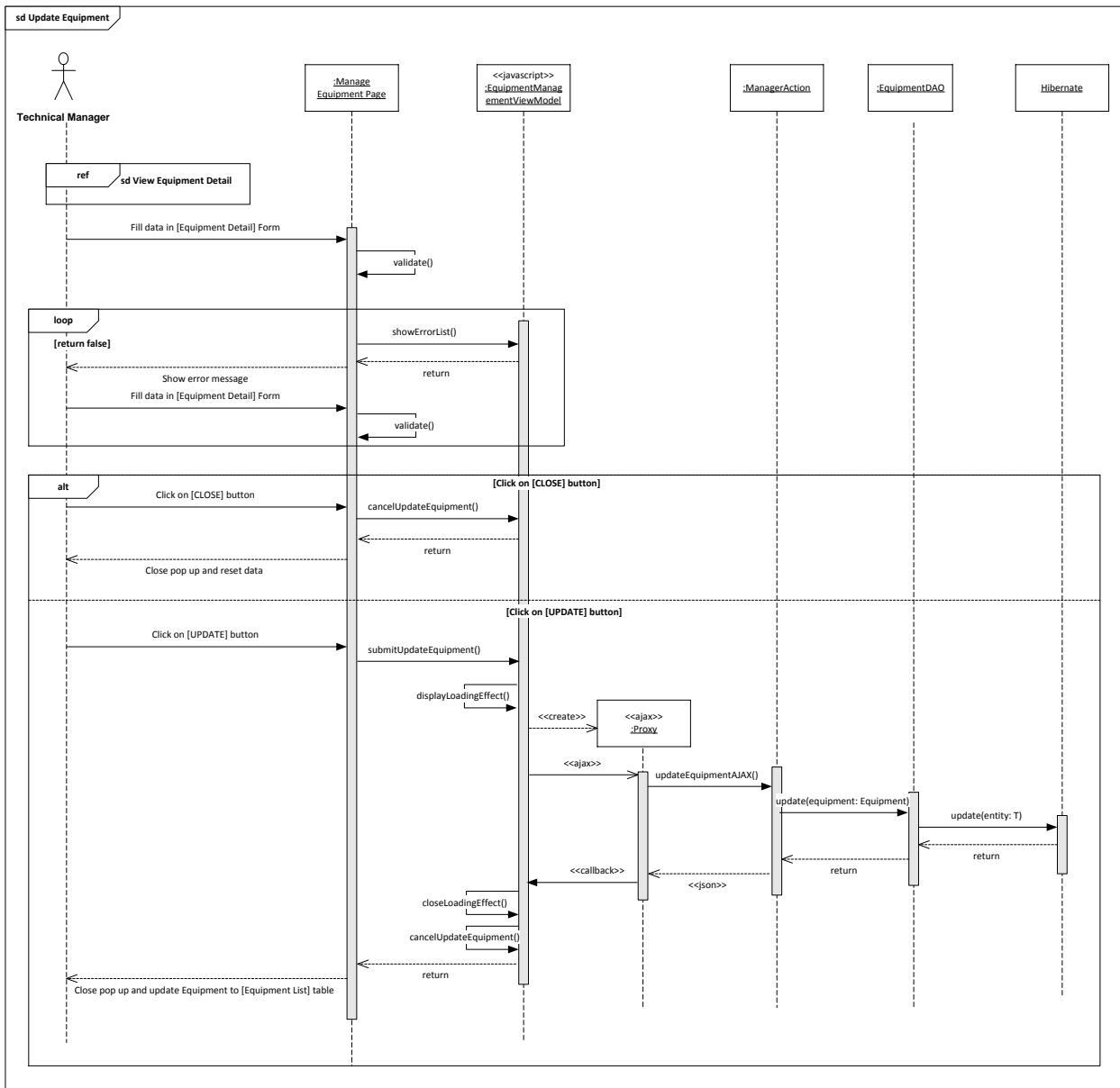


Figure 147-Technical Manager Update Equipment sequence diagram

5.43. < Technical Manager > [Statistics] Statistics by Request Type

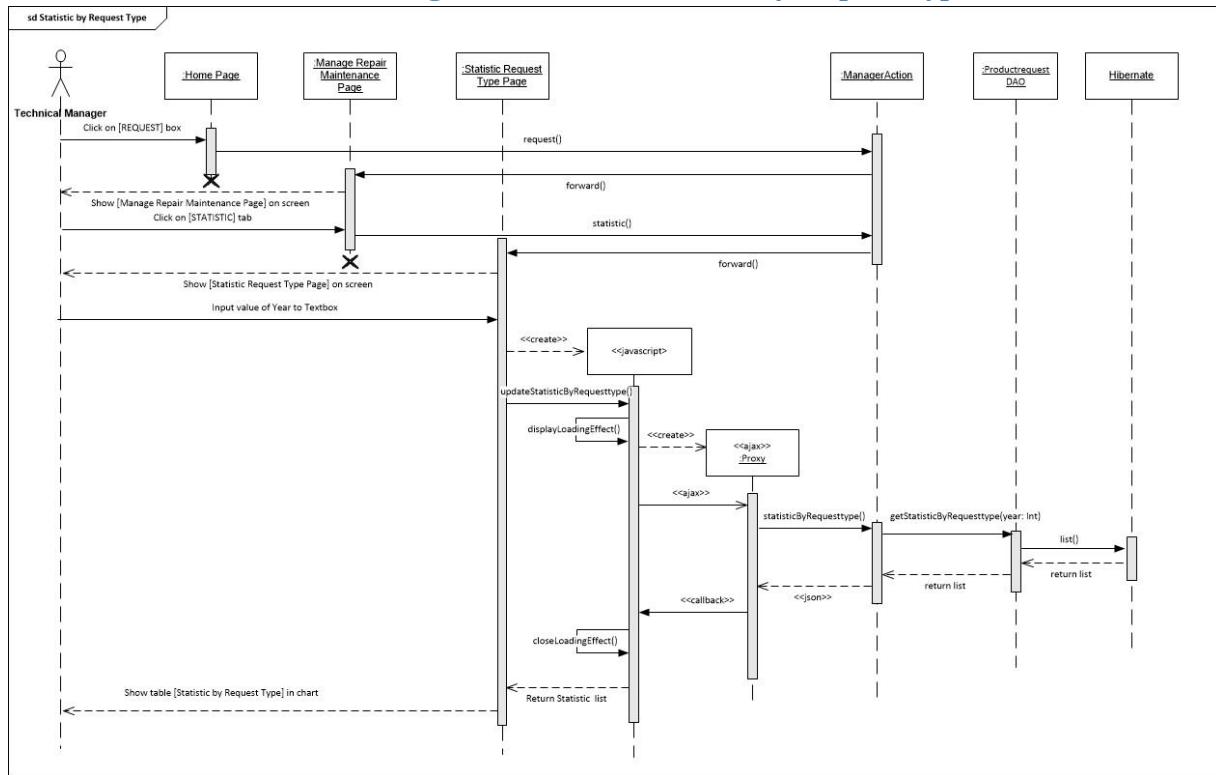


Figure 148-Techical Manager Statistics Maintenance of Repair sequence diagram

5.44. < Technical Manager > [Statistics] Statistics Top Fixed Model

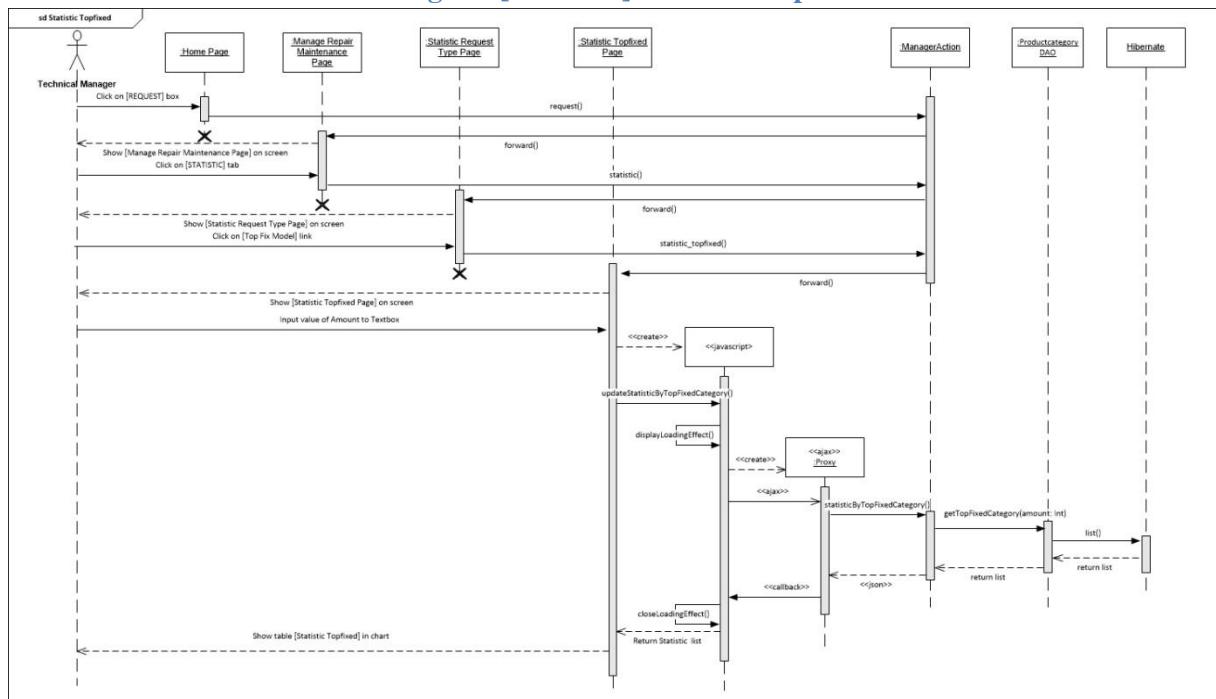


Figure 149-Techical Manager Statistics Top Product Request sequence diagram

5.45. < Technical Manager > [Statistics] Statistics by Symptom

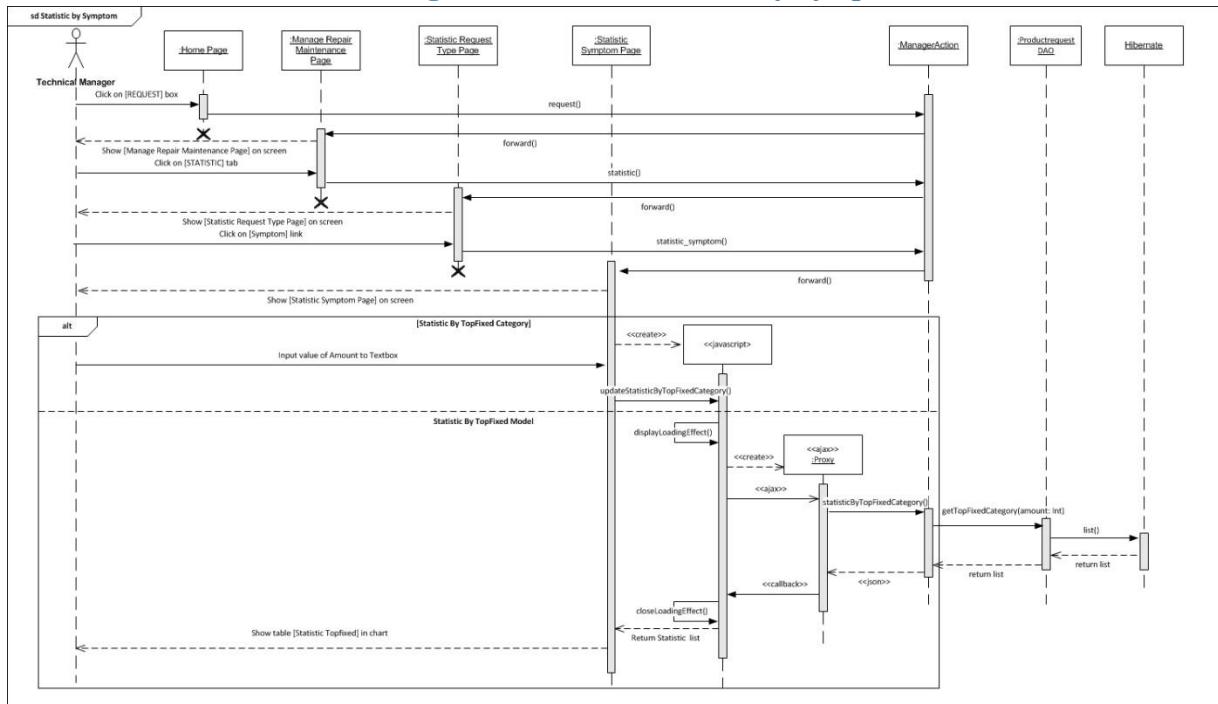


Figure 150-Technical Manager Statistics by Symptom sequence diagram

5.46. < Technical Manager > [Manage User] Add new Account

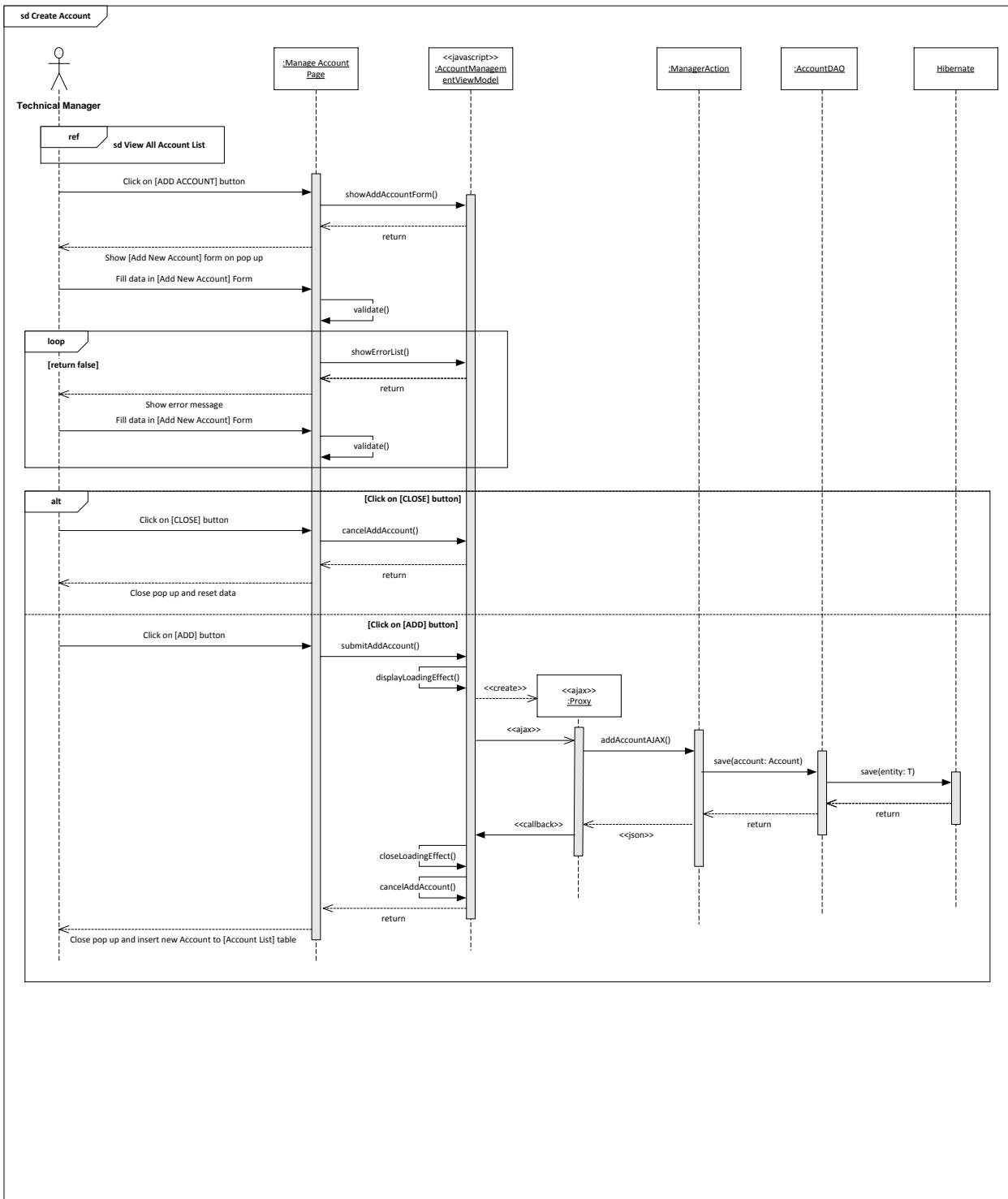


Figure 151-Technical Manager Add New Account sequence diagram

5.47. < Technical Manager> [Manage User] View List of Accounts

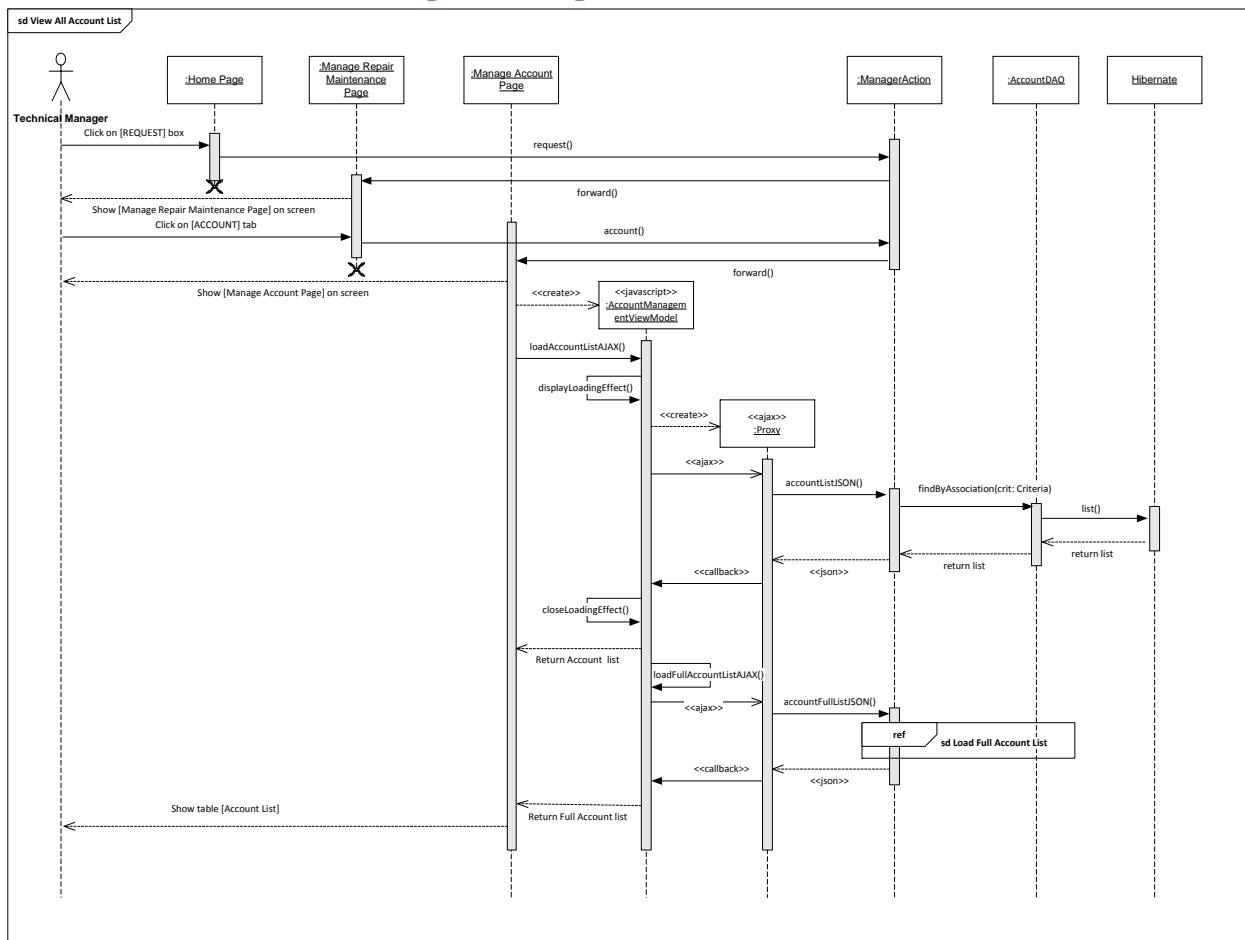


Figure 152-Technical Manager View List of Accounts sequence diagram

5.48. <Technical Manager> [Manage User] Filter Account

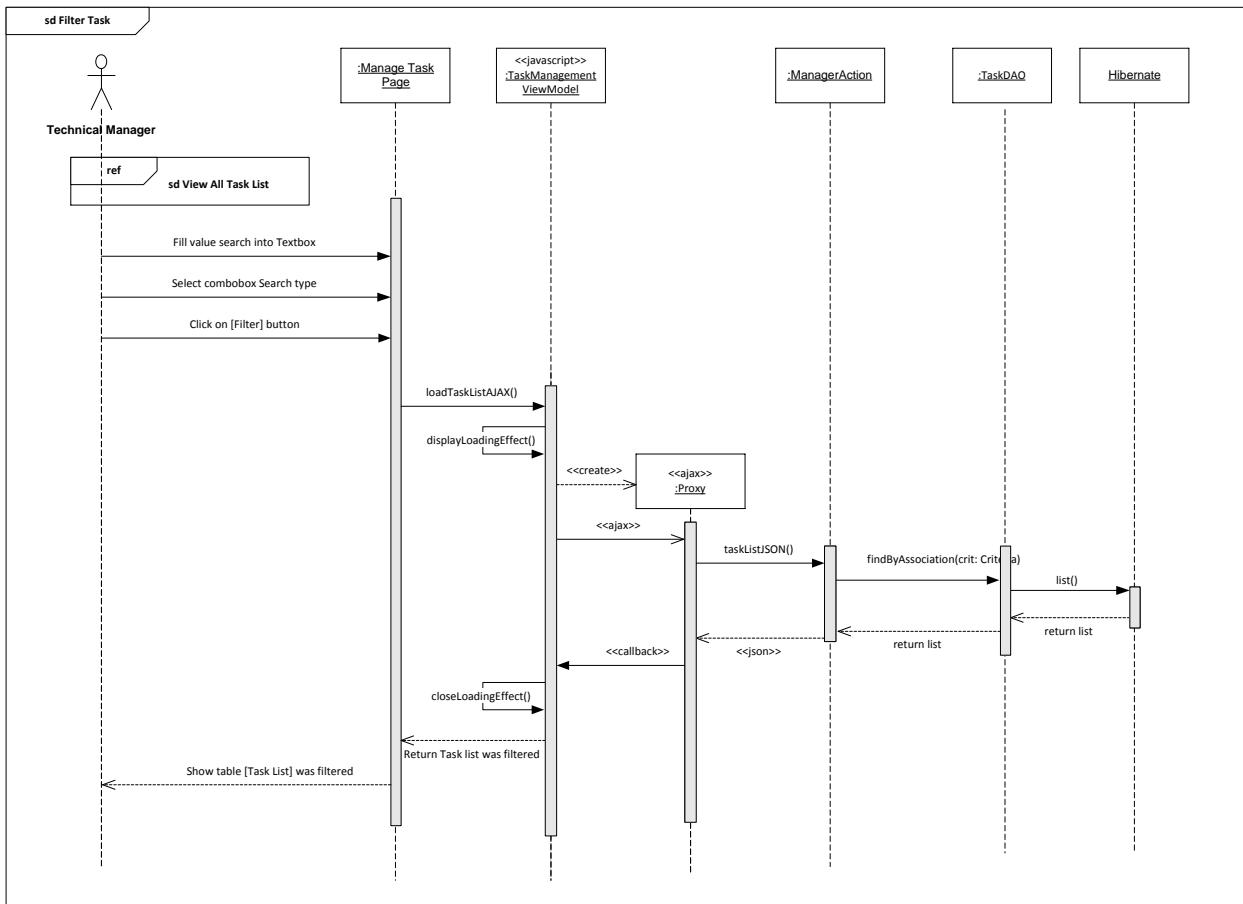


Figure 153-Technical Manager Filter Account sequence diagram

5.49. < Technical Manager> [Manage User] Update Account Information

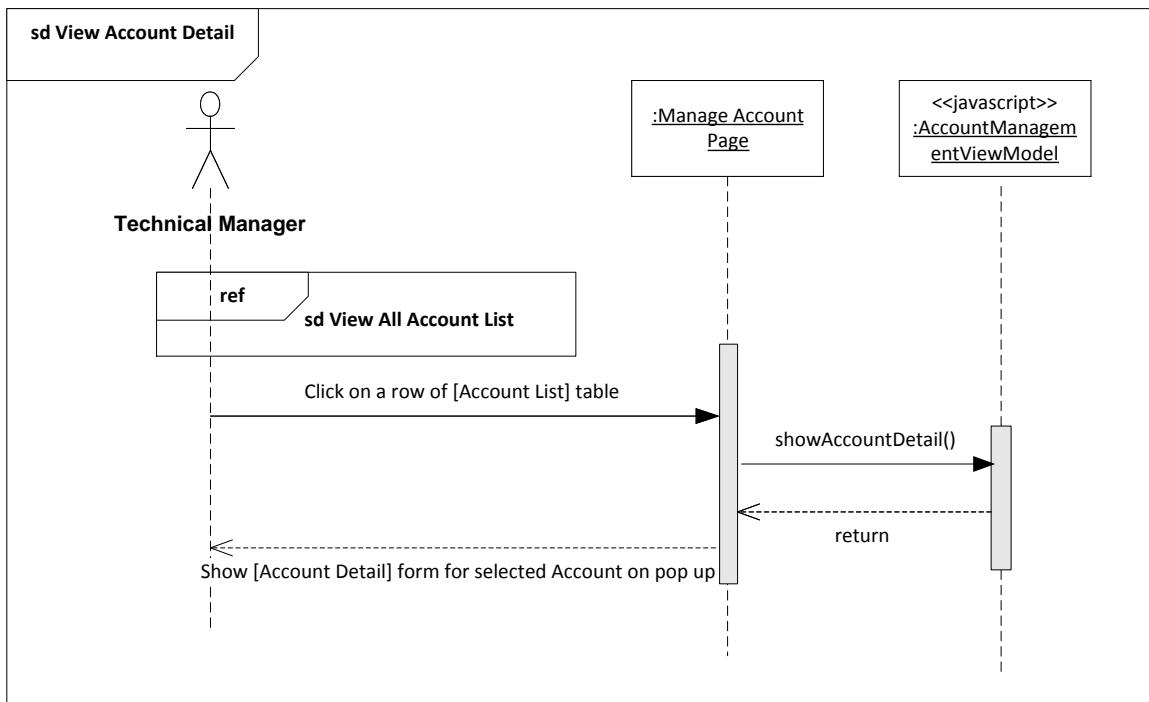


Figure 154-Technical Manager View Account Detail sequence diagram

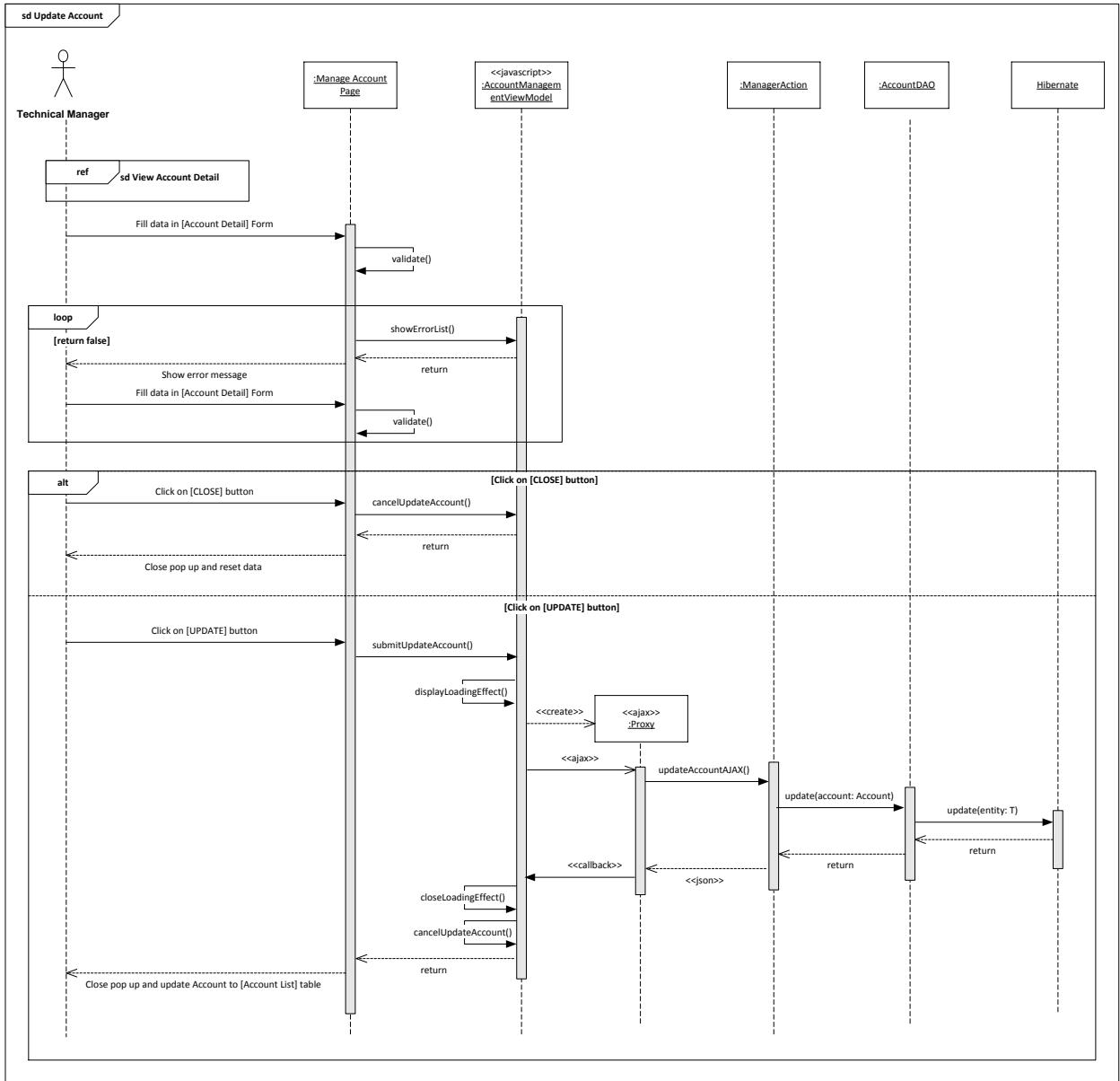


Figure 155-Technical Manager Update Account Information sequence diagram

5.50. < Technical Manager> [Manage Product] Add Product Category

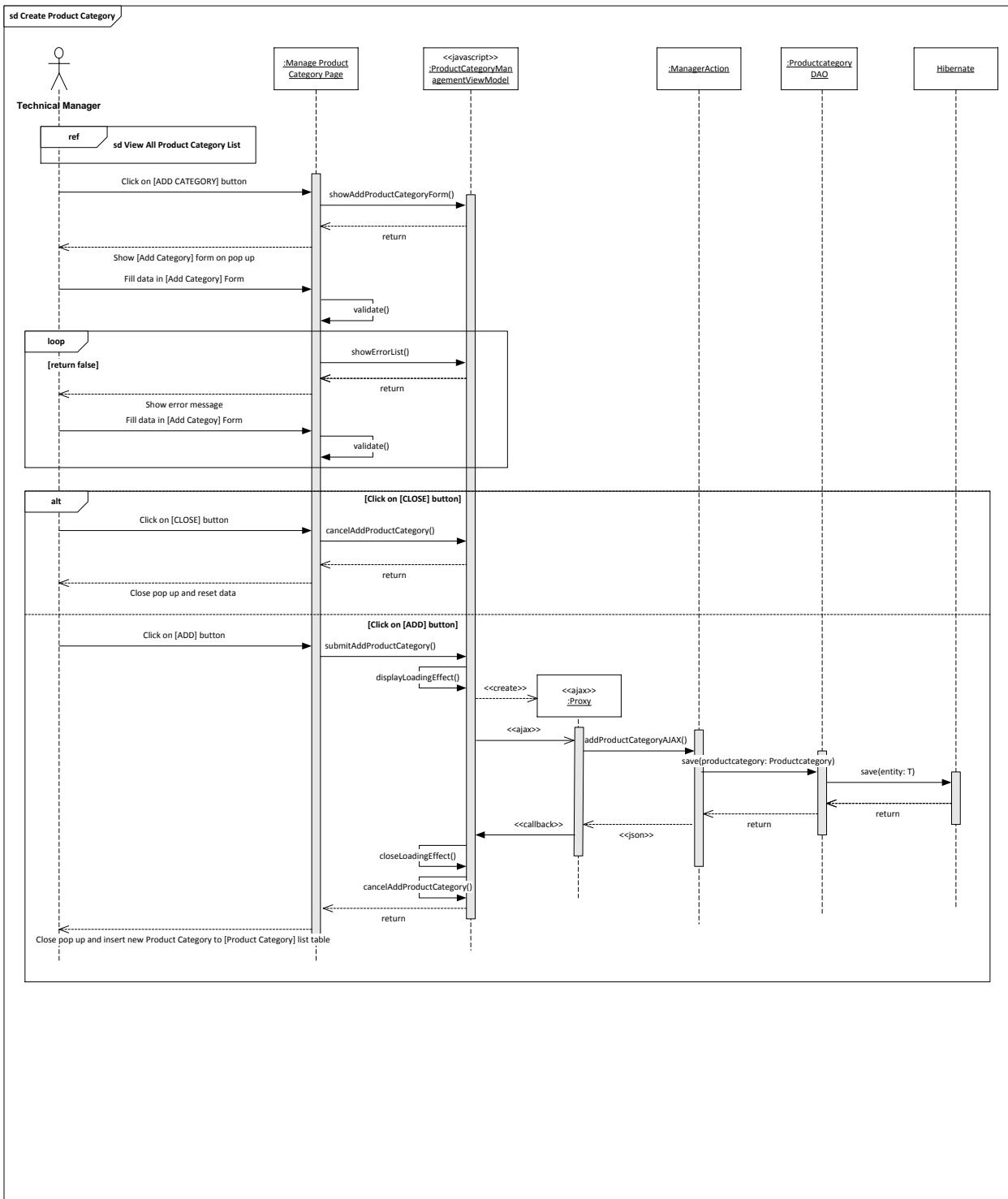


Figure 156-Technical Manager Add Product Category sequence diagram

5.51. < Technical Manager> [Manage Product] View All Product Category List

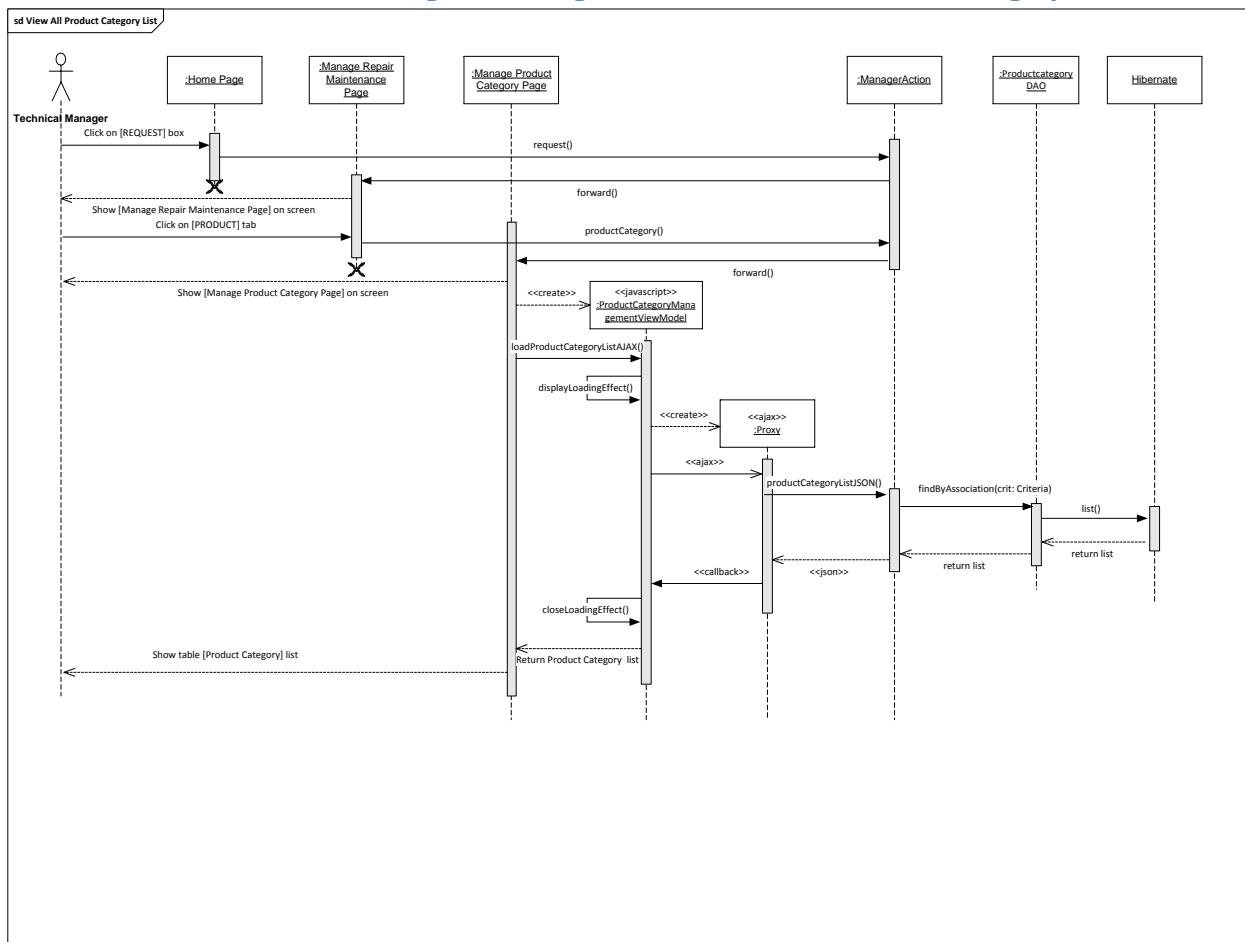


Figure 157-Technical Manager View All Product Category List sequence diagram

5.52. < Technical Manager> [Manage Product] Update Product Category

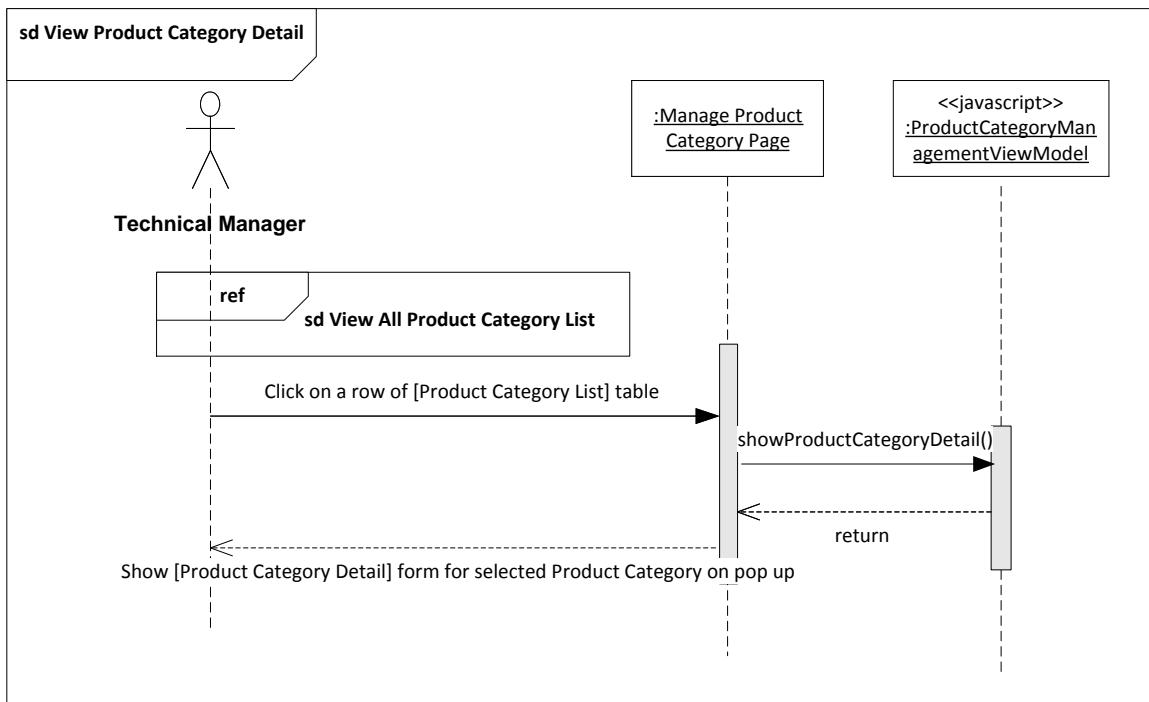


Figure 158-Technical Manager View Product Category Detail sequence diagram

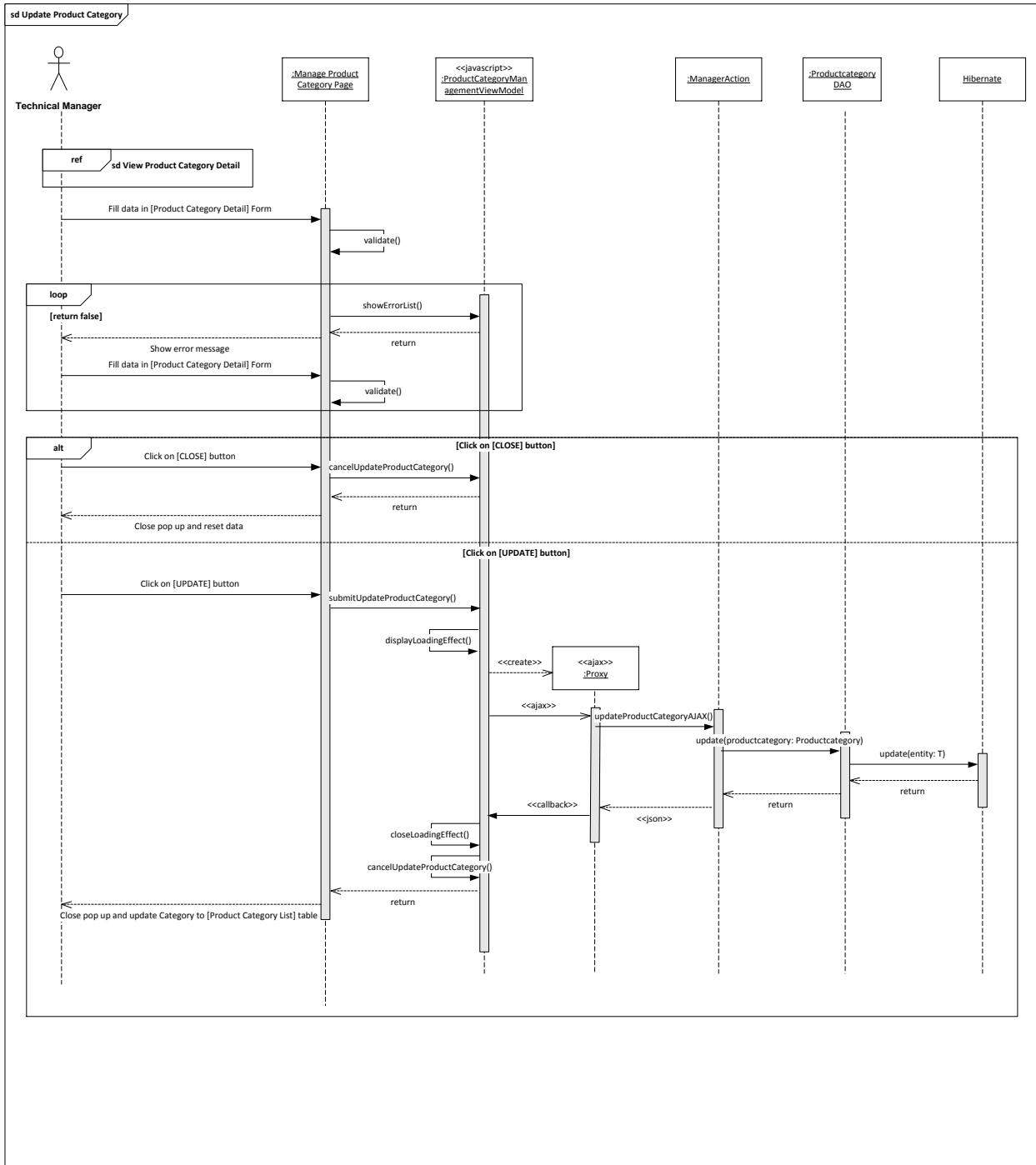


Figure 159-Technical Manager Update Product Category sequence diagram

5.53. < Technical Manager> [Manage Product] Add Product Prototype

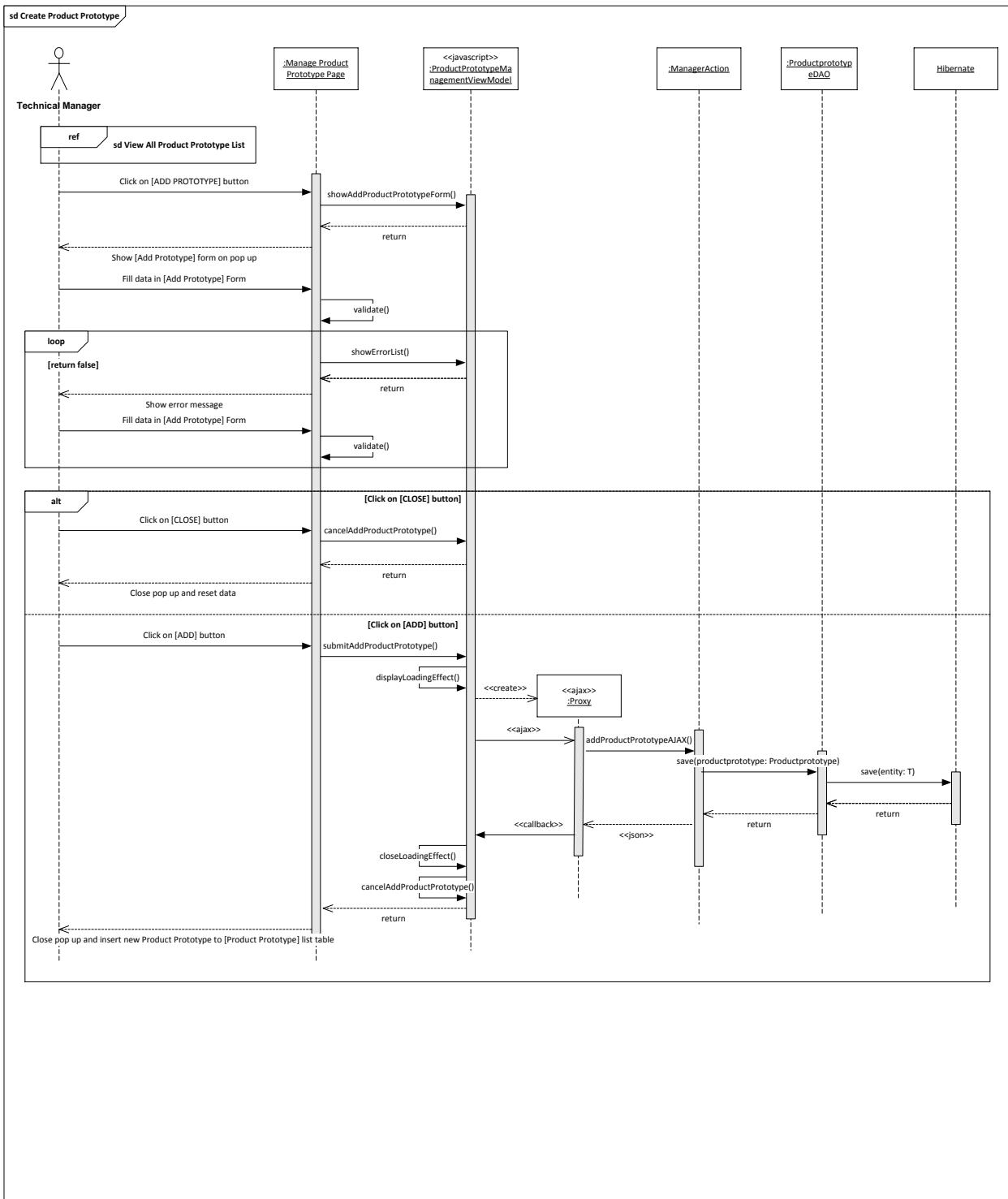


Figure 160-Technical Manager Add Product Prototype sequence diagram

5.54. < Technical Manager> [Manage Product] View All Product Prototype List

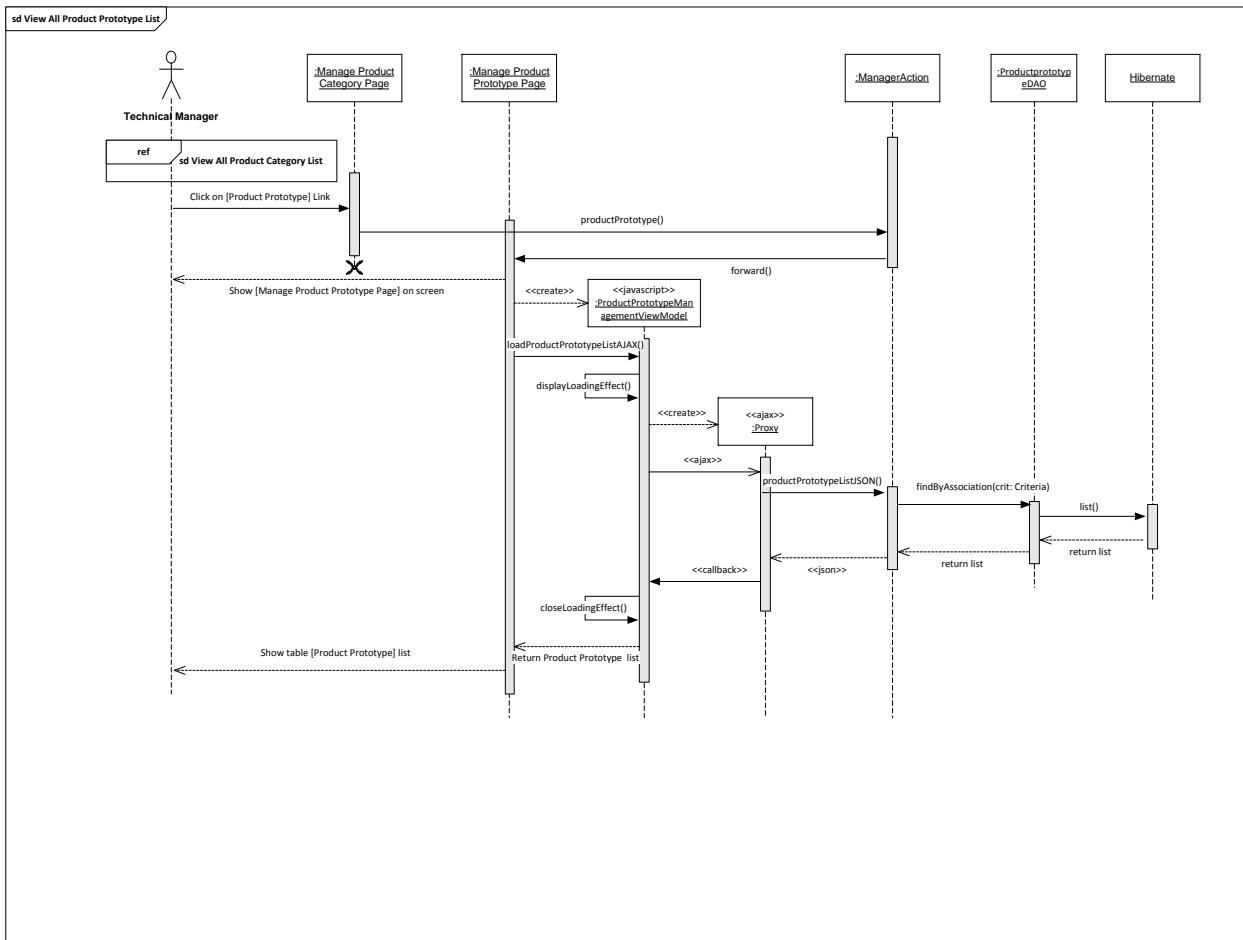


Figure 161-Technical Manager View All Product Prototype List sequence diagram

5.55. < Technical Manager> [Manage Product] Update Product Prototype

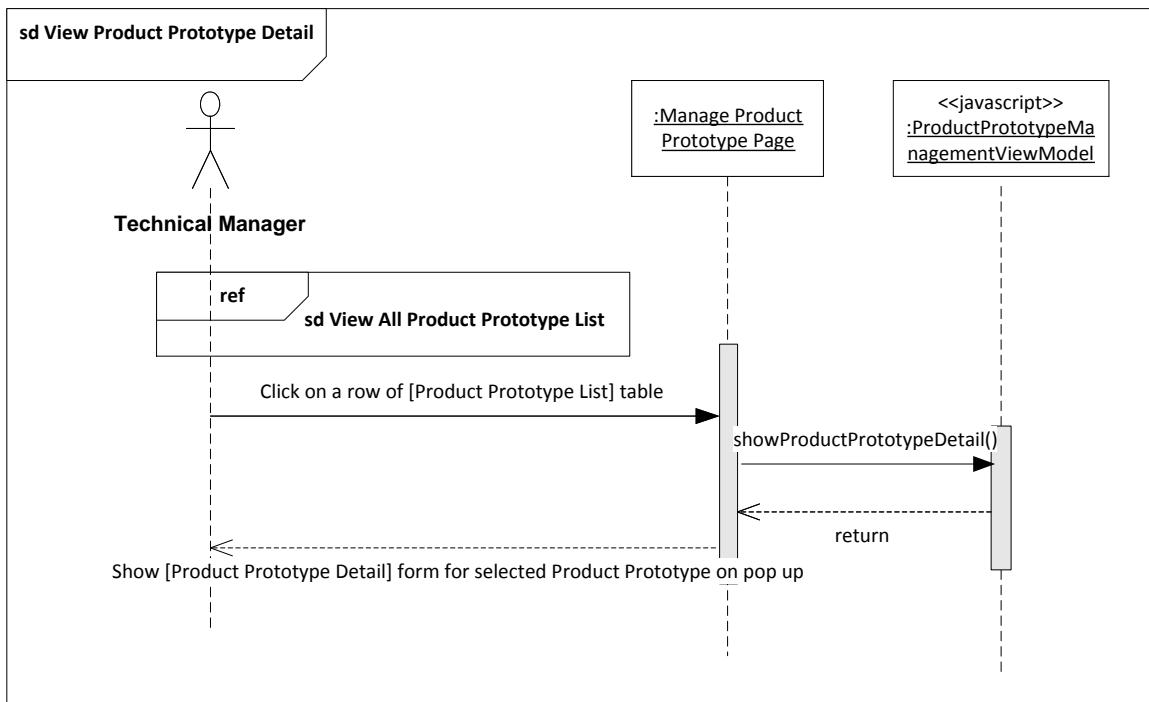


Figure 162-Technical Manager View Product Prototype Detail sequence diagram

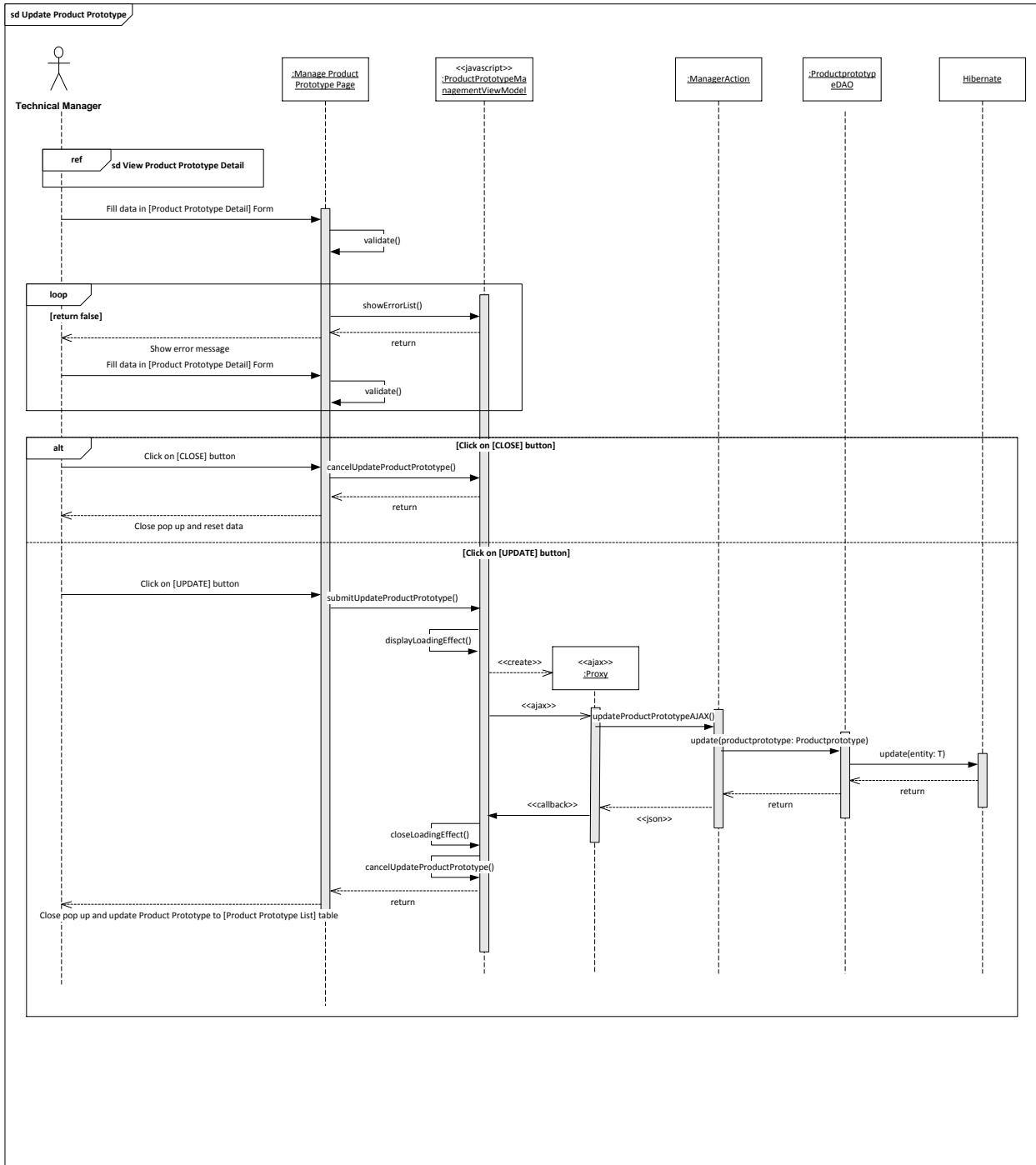


Figure 163-Technical Manager Update Product Prototype sequence diagram

5.56. < Technical Manager> [Manage Product] Add Product Model

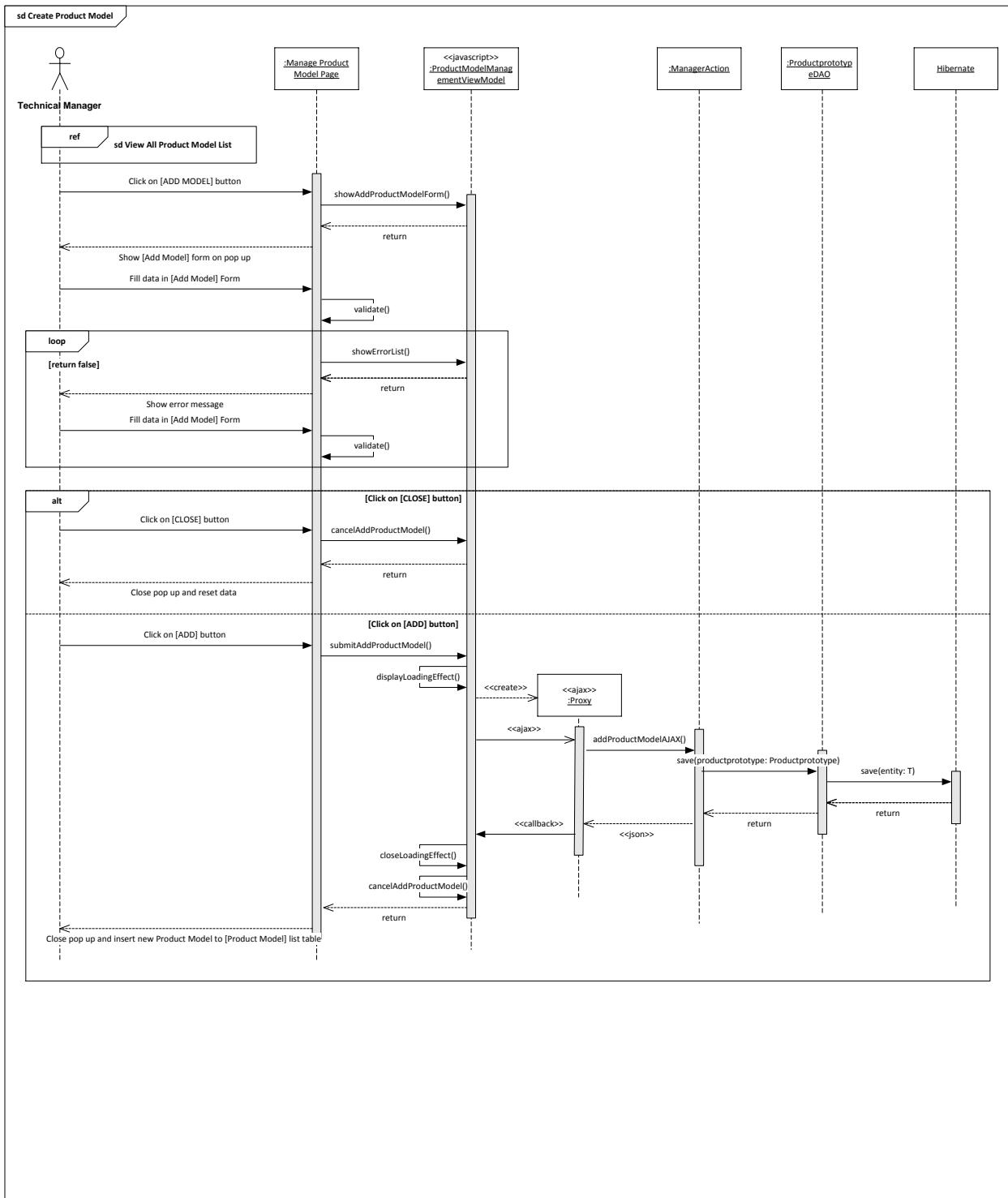


Figure 164-Technical Manager Add Product Model sequence diagram

5.57. < Technical Manager> [Manage Product] View All Product Model List

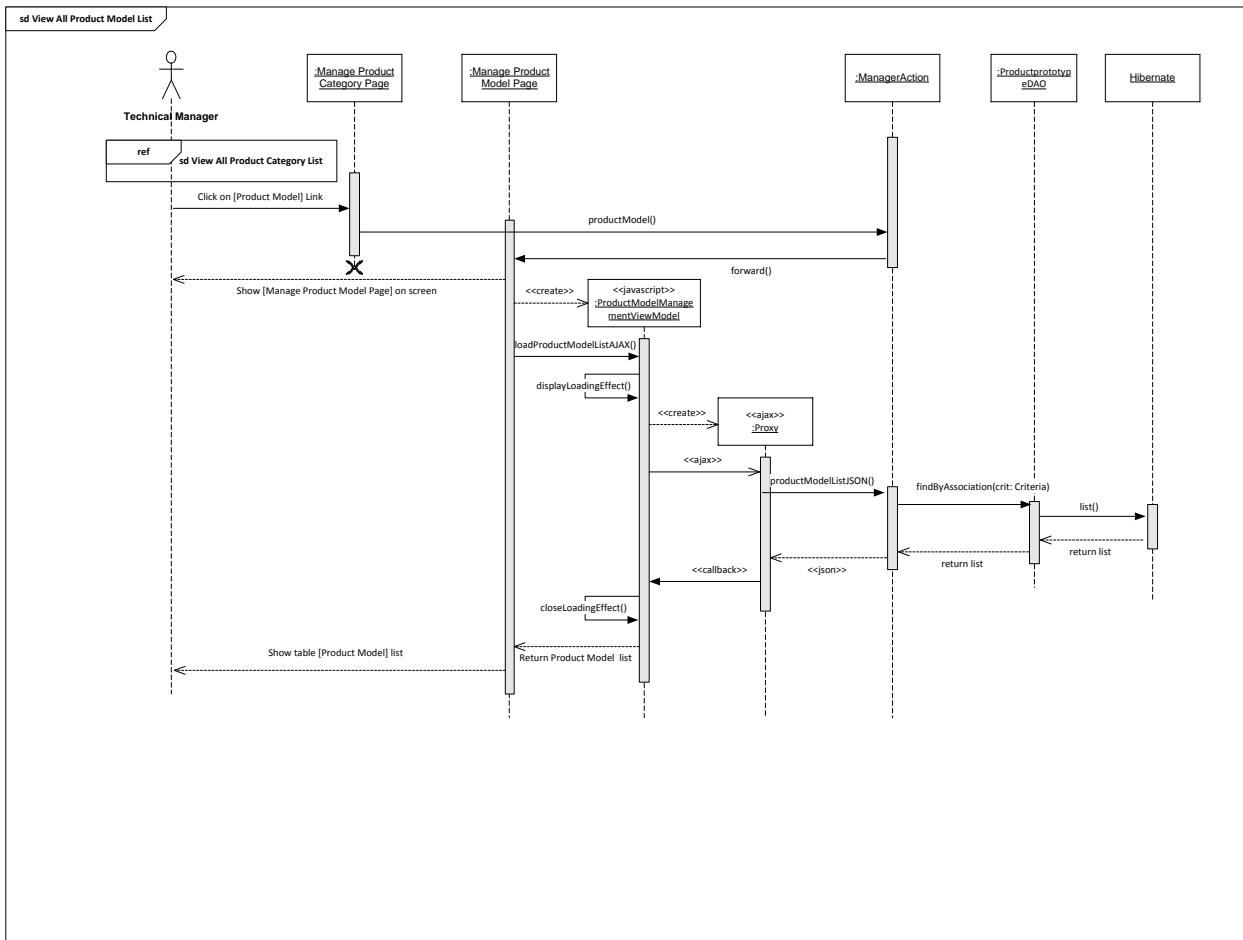


Figure 165-Technical Manager View All Product Model List sequence diagram

5.58. < Technical Manager> [Manage Product] Update Product Model

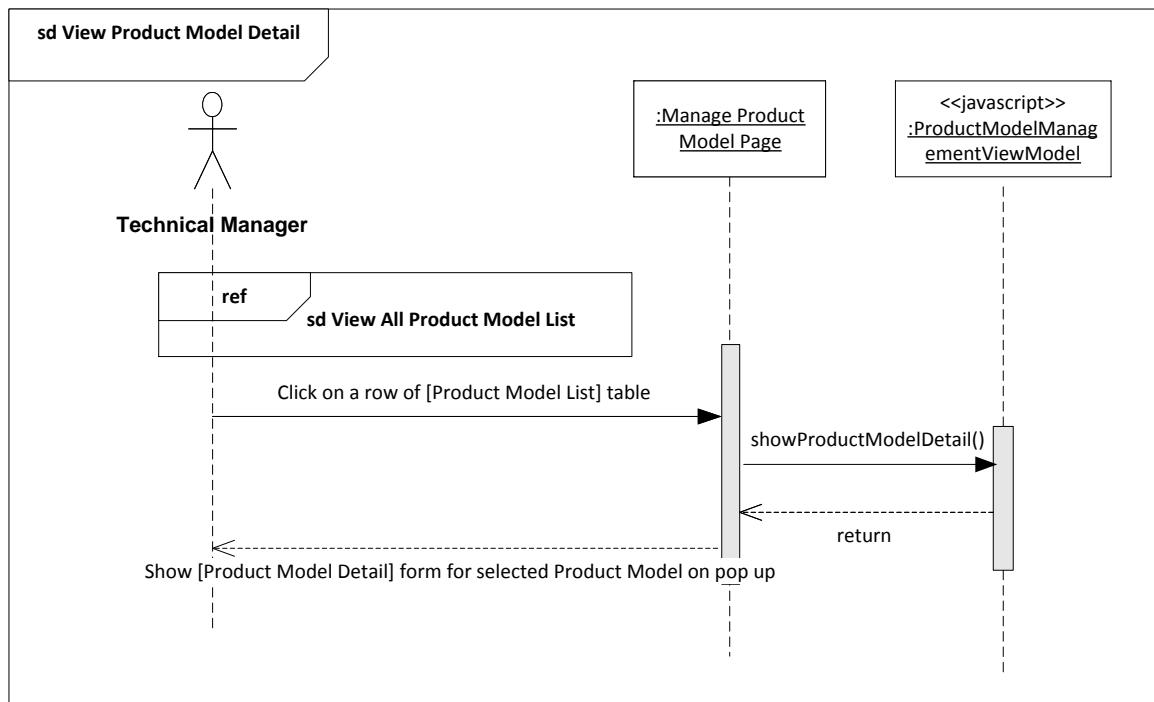


Figure 166-Techical Manager View Product Model Detail sequence diagram

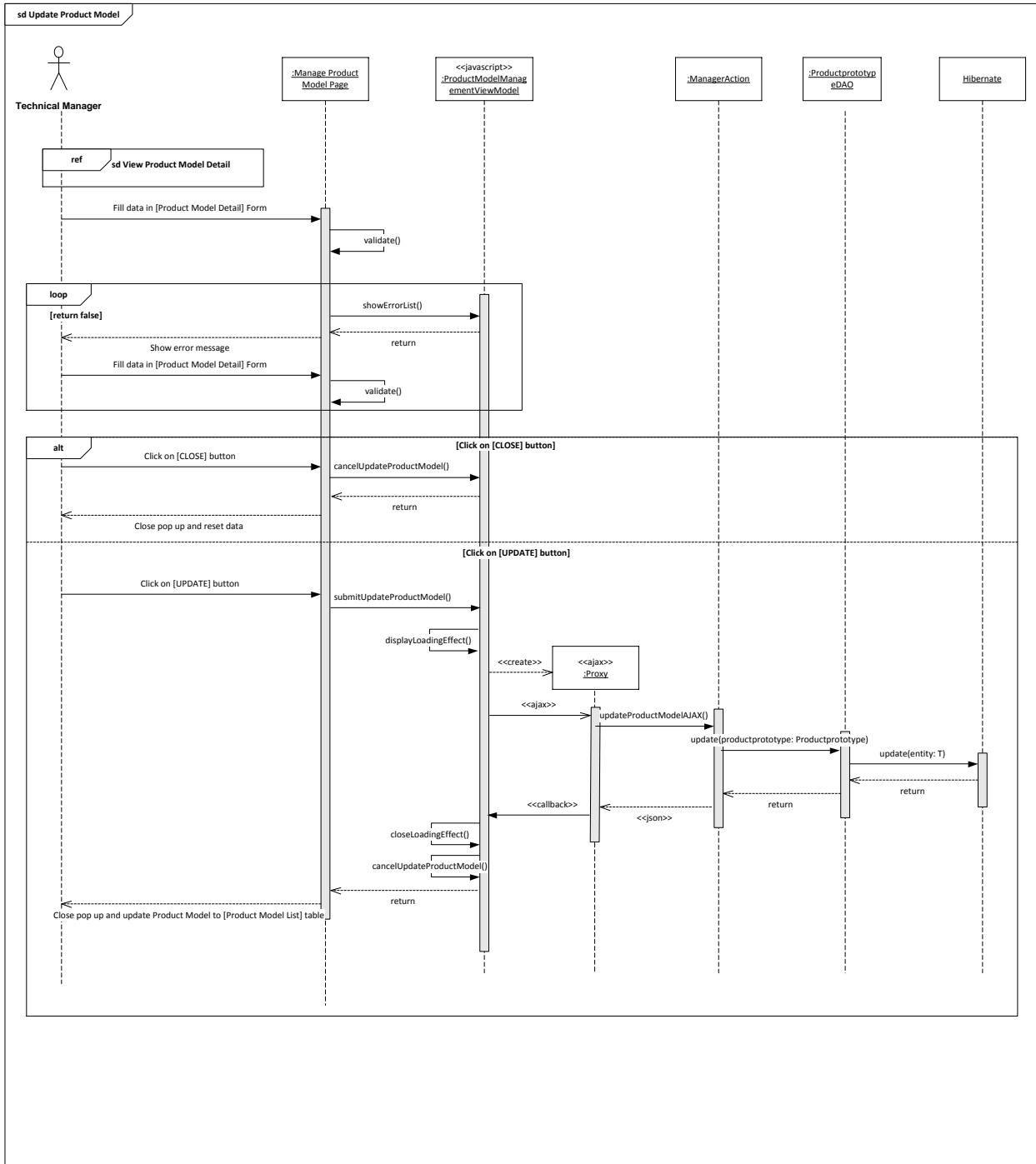


Figure 167-Technical Manager Update Product Model sequence diagram

5.59. < Technical Manager> [Manage Invoice] View Finished Request

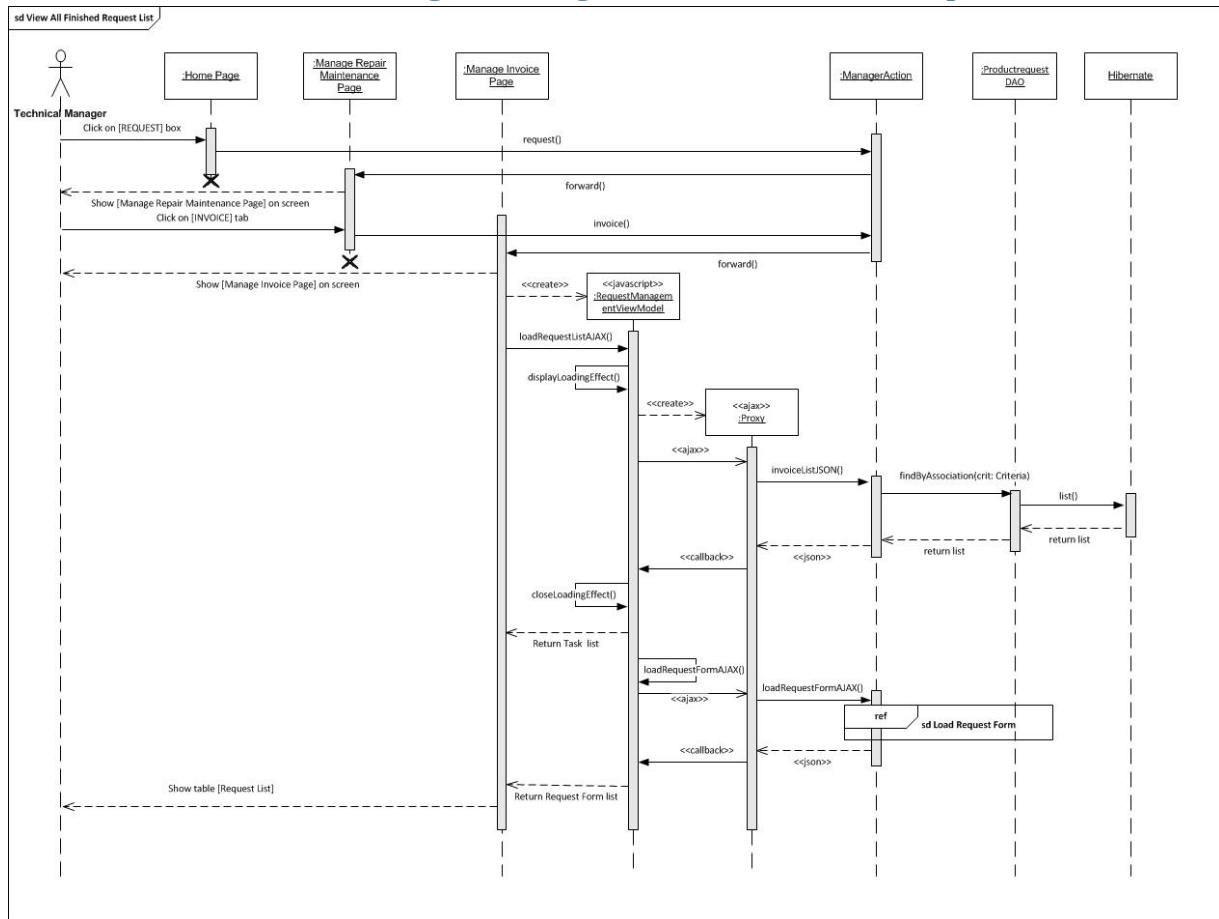
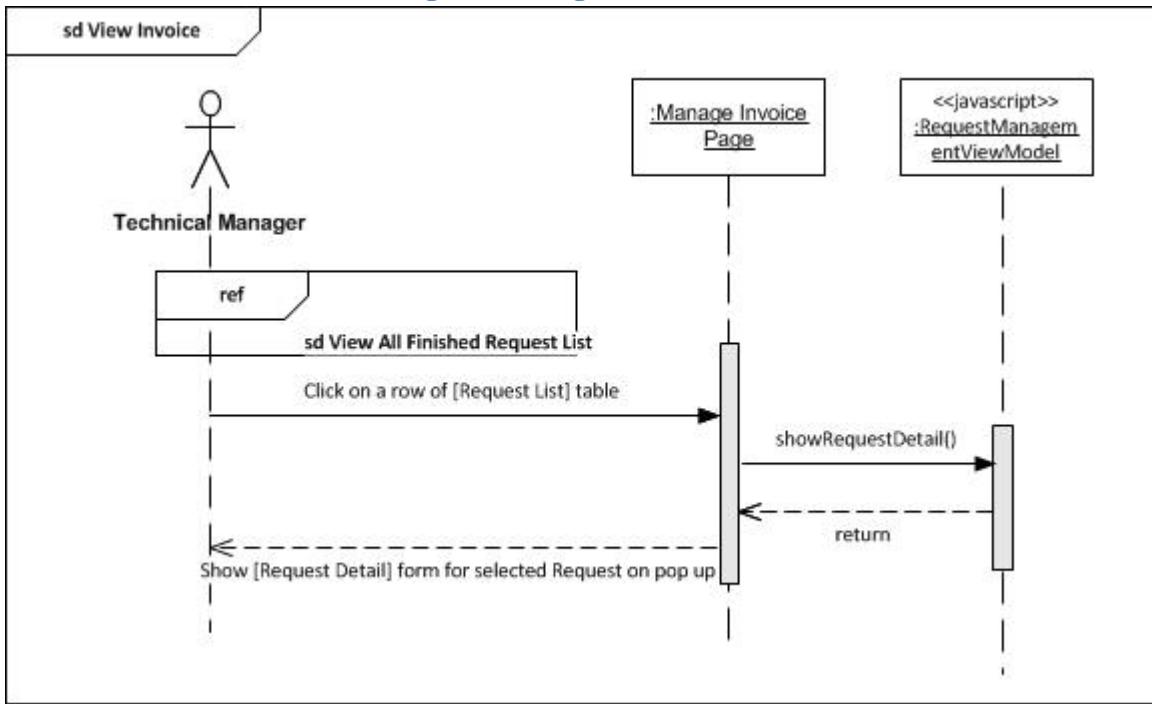
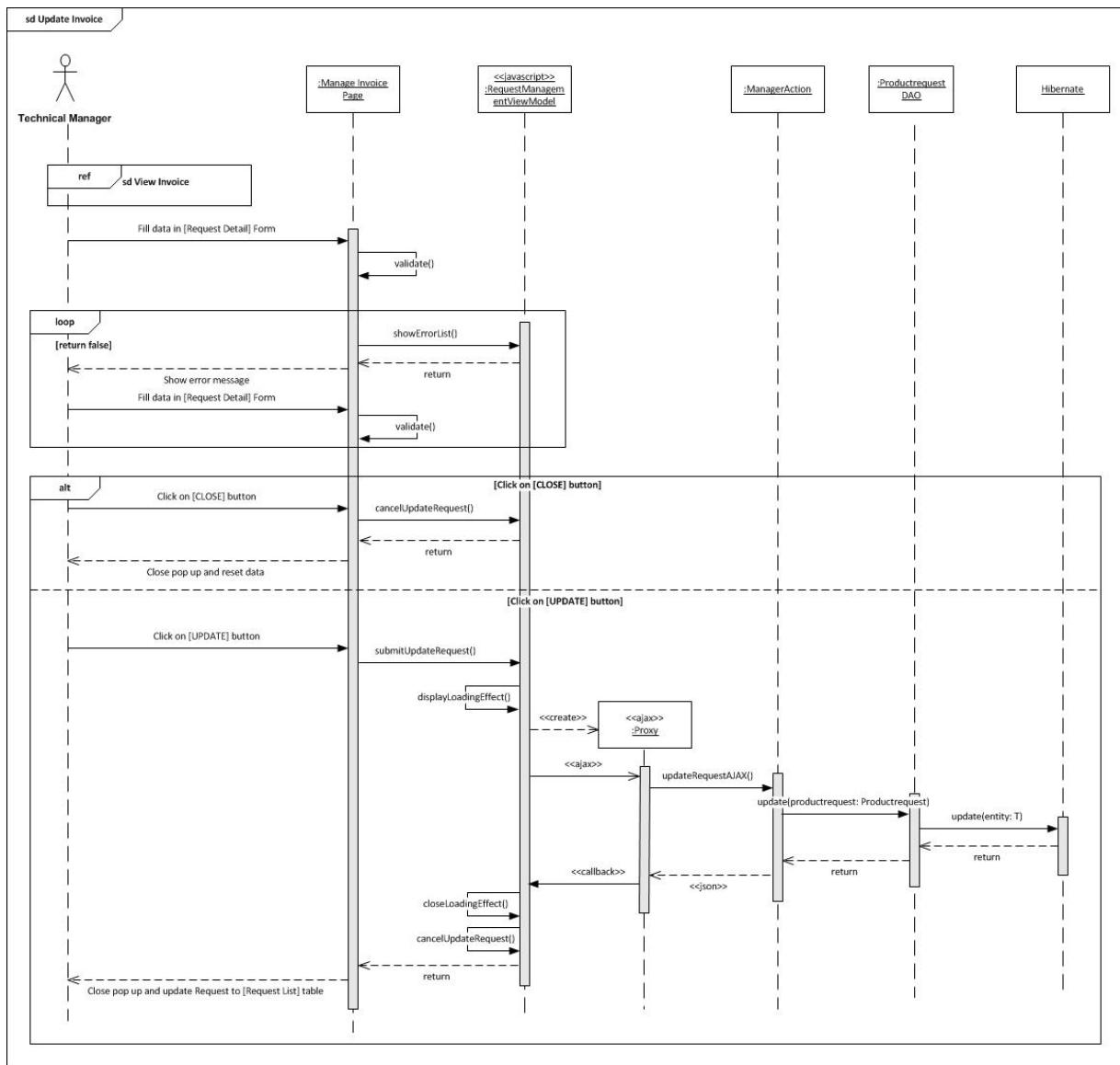


Figure 168 - Technical Manager View Finished Request

5.60. < Technical Manager> [Manage Invoice] View Invoice**Figure 169 - Technical Manager View Invoice**



5.61. < Technical Manager> [Manage Invoice] Update Invoice

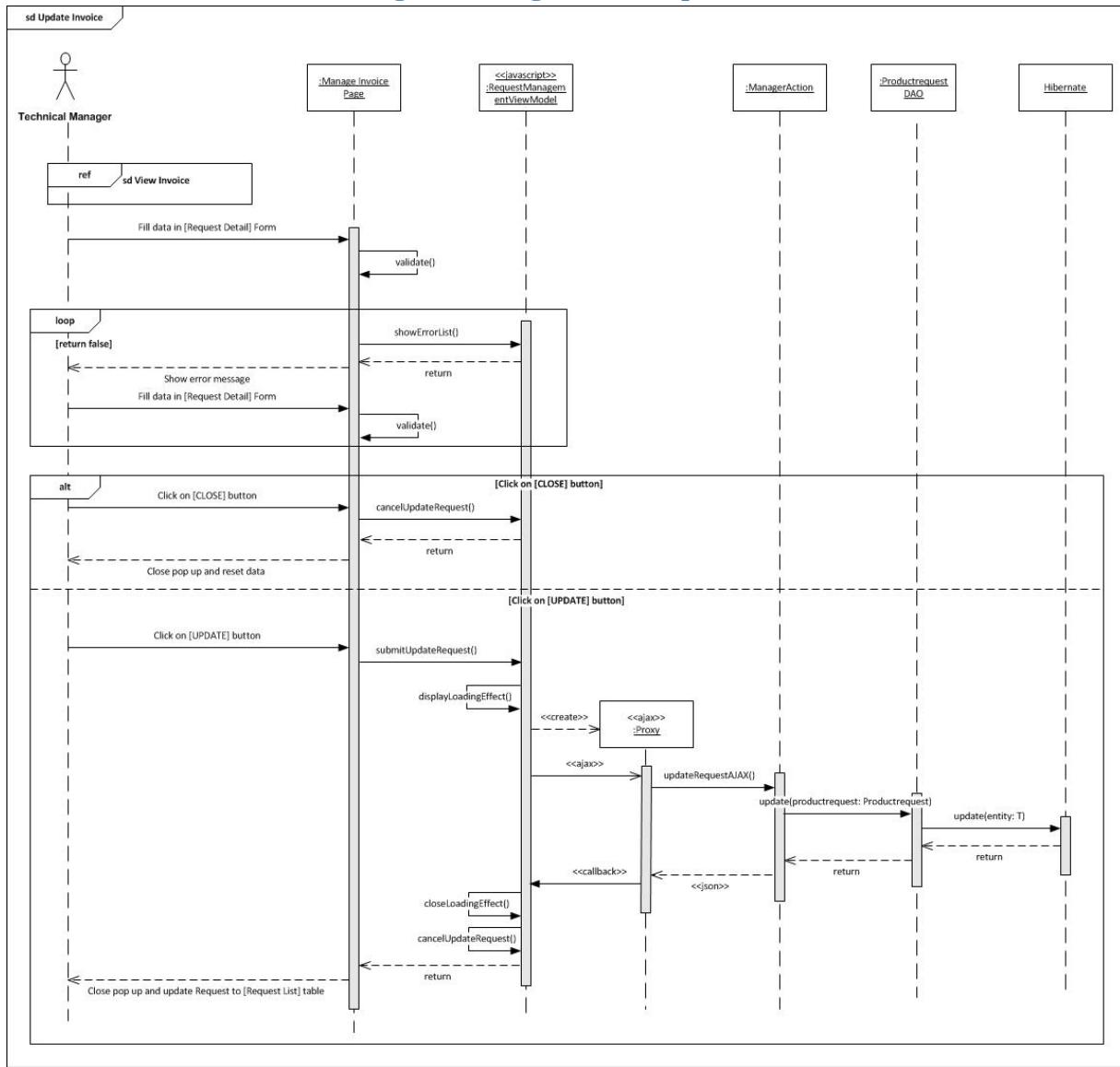


Figure 170 - Technical Manager Update Invoice

5.62. < Technical Manager> [Manage Promotion] View All Promotion List

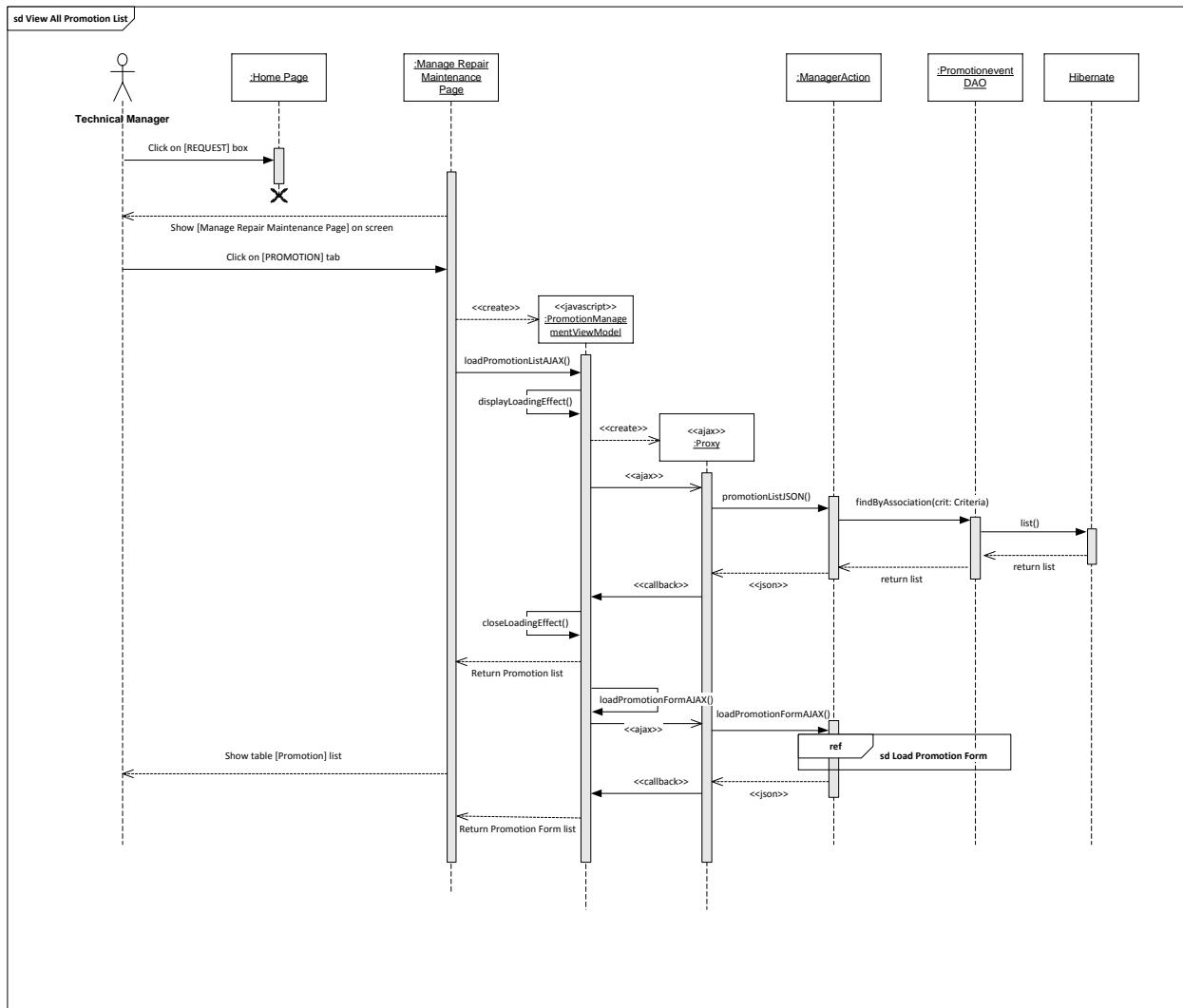


Figure 171-Technical Manager View All Promotion List sequence diagram

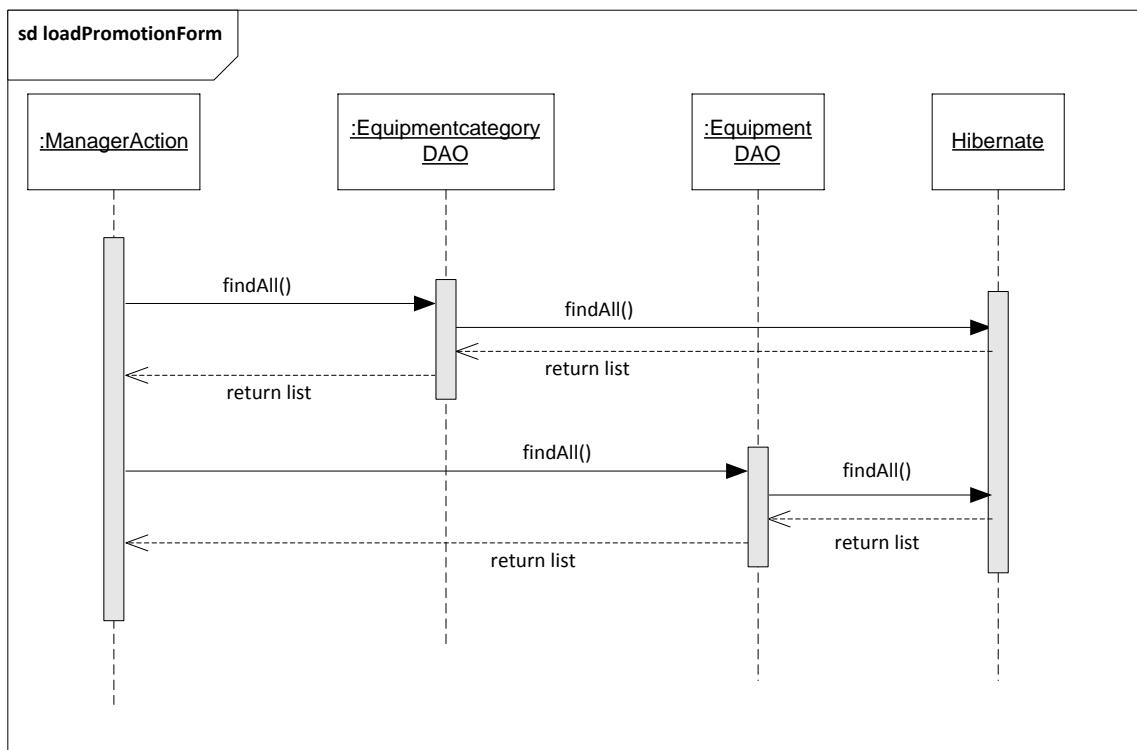


Figure 172-Technical Manager Load Promotion Form sequence diagram

5.63. < Technical Manager> [Manage Promotion] Create Promotion

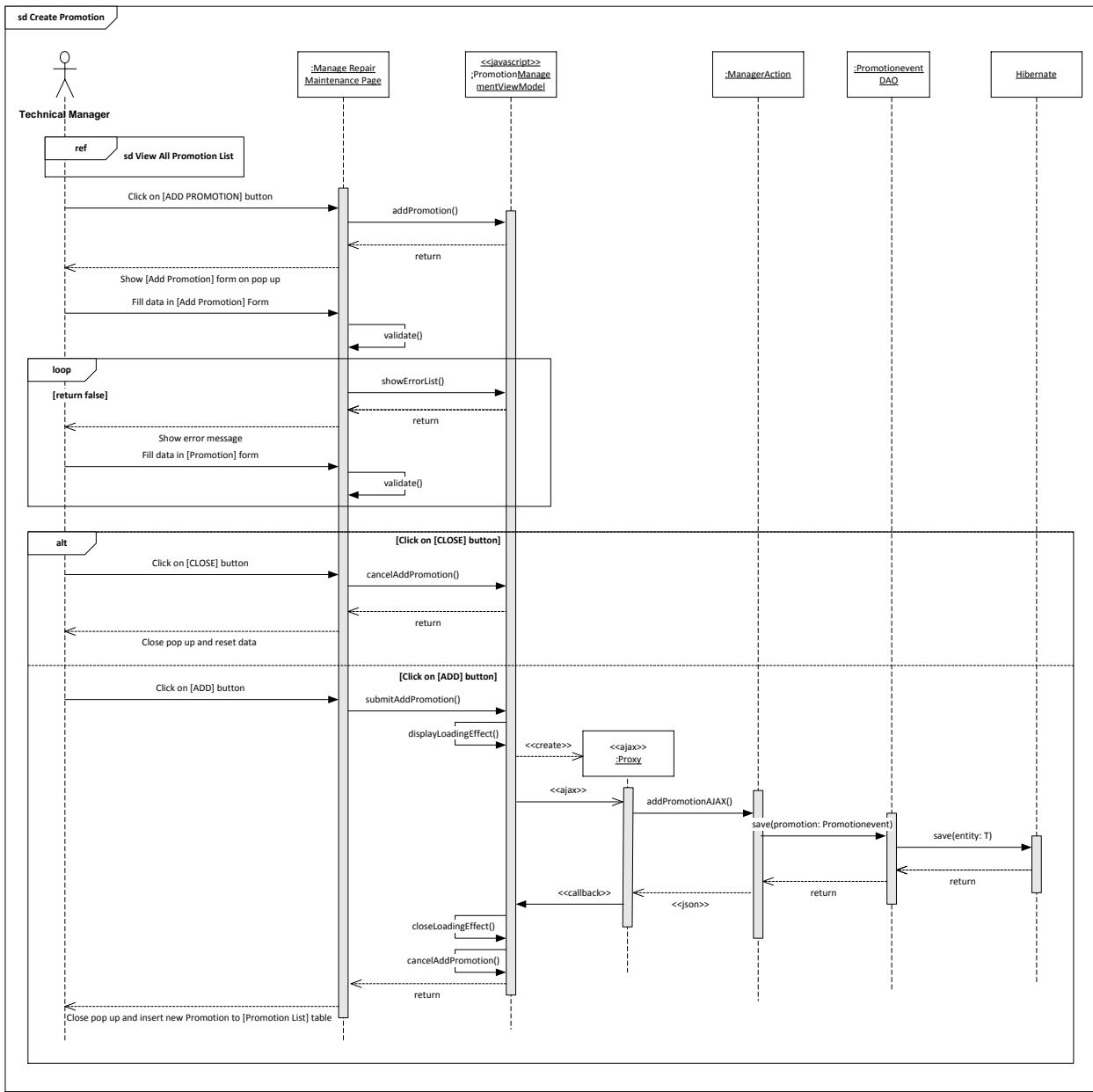


Figure 173-Technical Manager Create Promotion sequence diagram

5.64. < Technical Manager> [Manage Promotion] View Promotion Detail

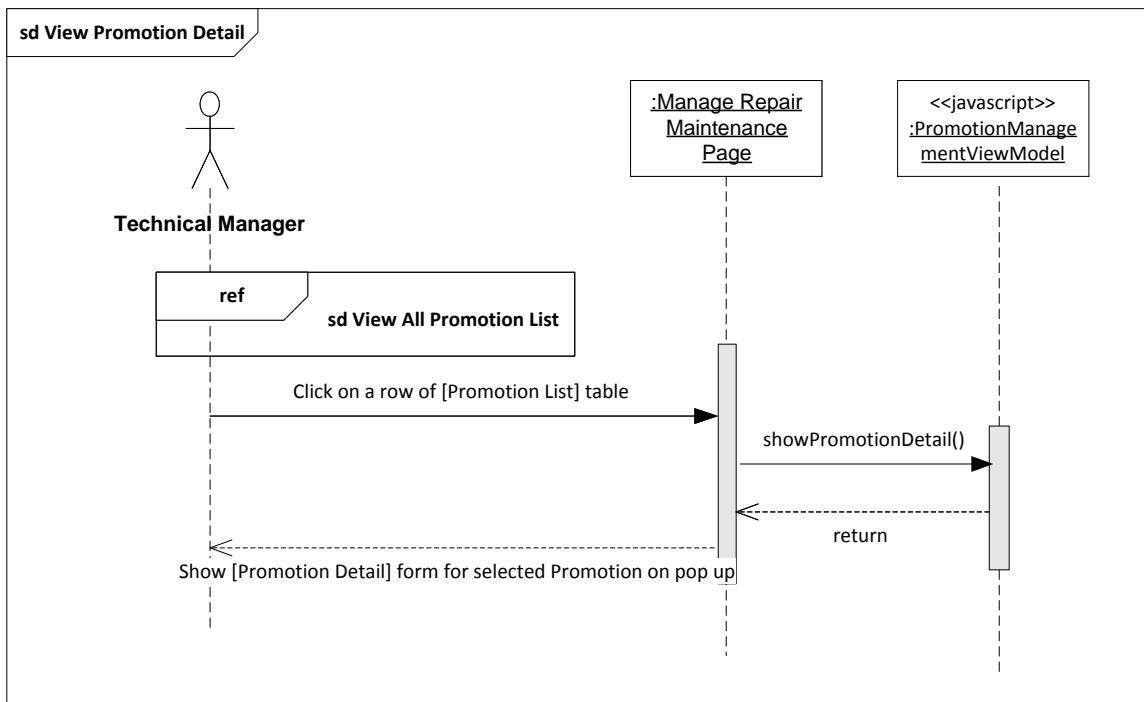


Figure 174-Technical Manager View Promotion Detail sequence diagram

5.65. < Technical Manager> [Manage Promotion] Update Promotion

5.66. < System> Auto Notification

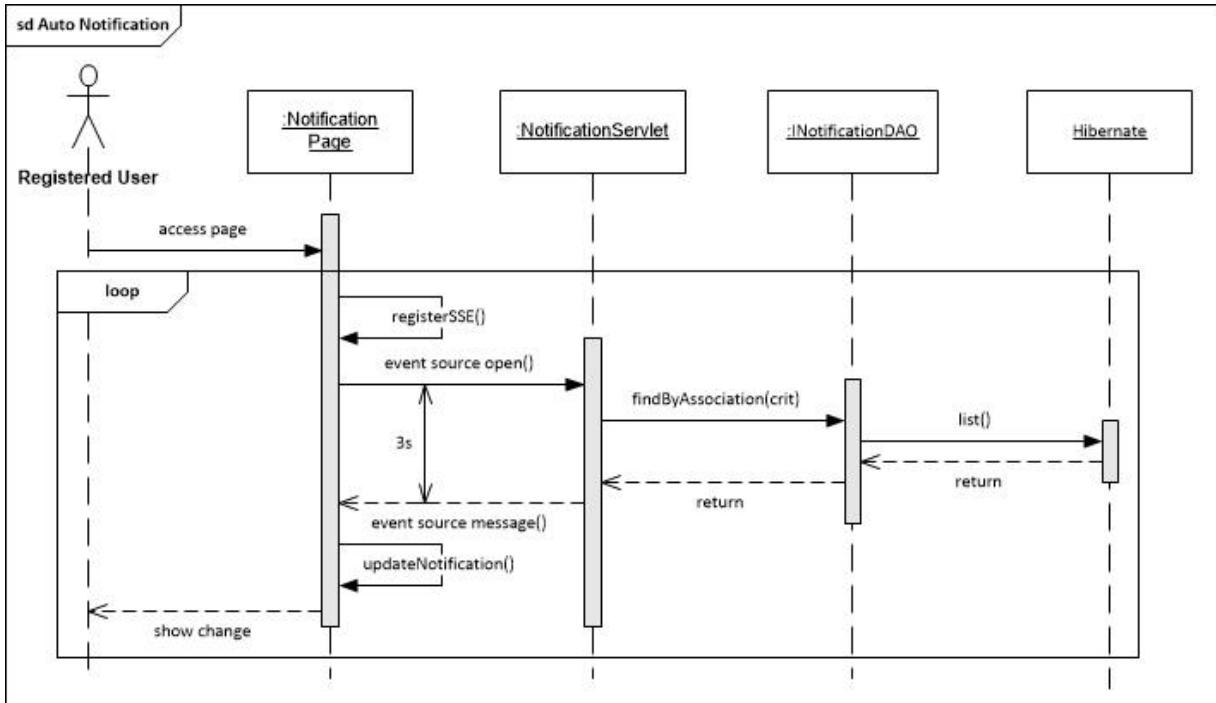


Figure 175-System Auto Notification

5.67. < System> Auto Check Missed Deadline Request / Task

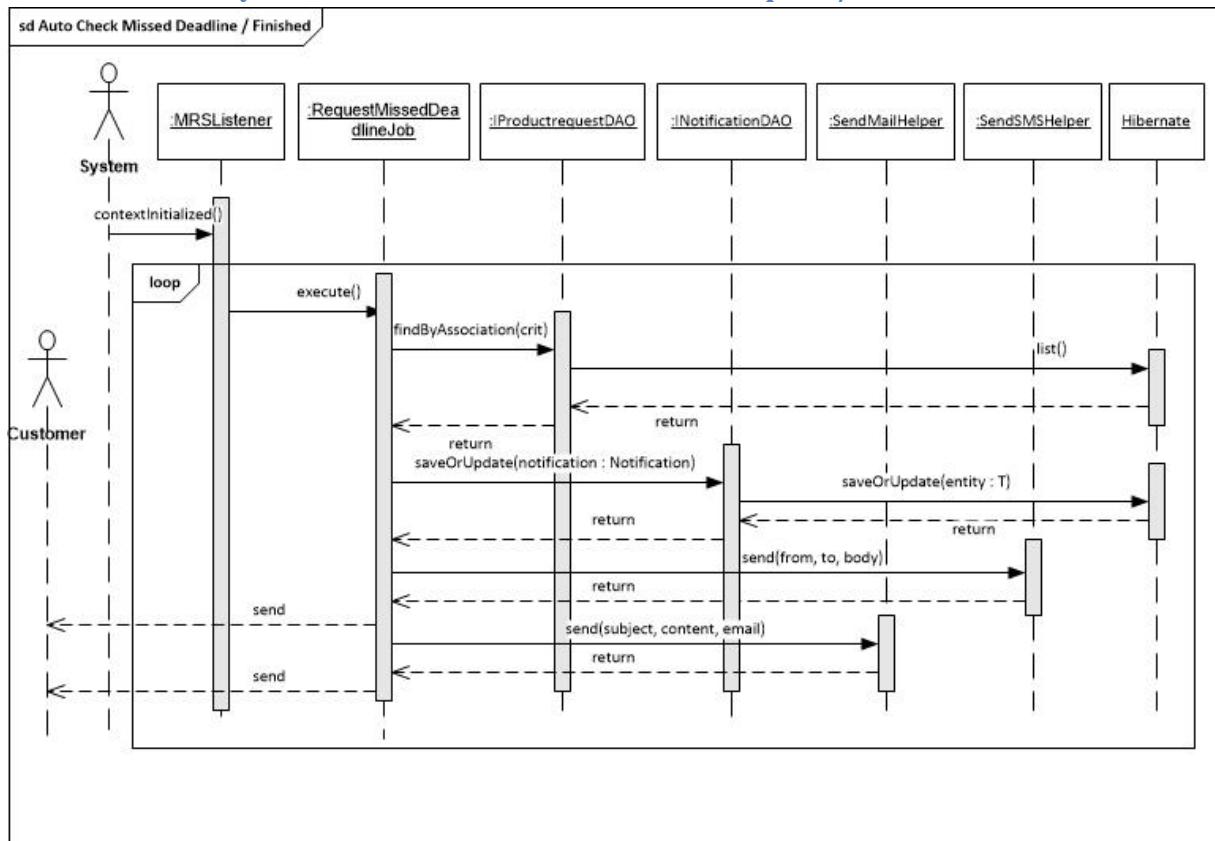


Figure 176-System Auto Check Missed Deadline Request / Task

6. User Interface Design

6.1. Page Flow Diagram

N/A

6.2. Detail Design of the User Interface

N/A

7. Database Design

7.1. Database Diagram

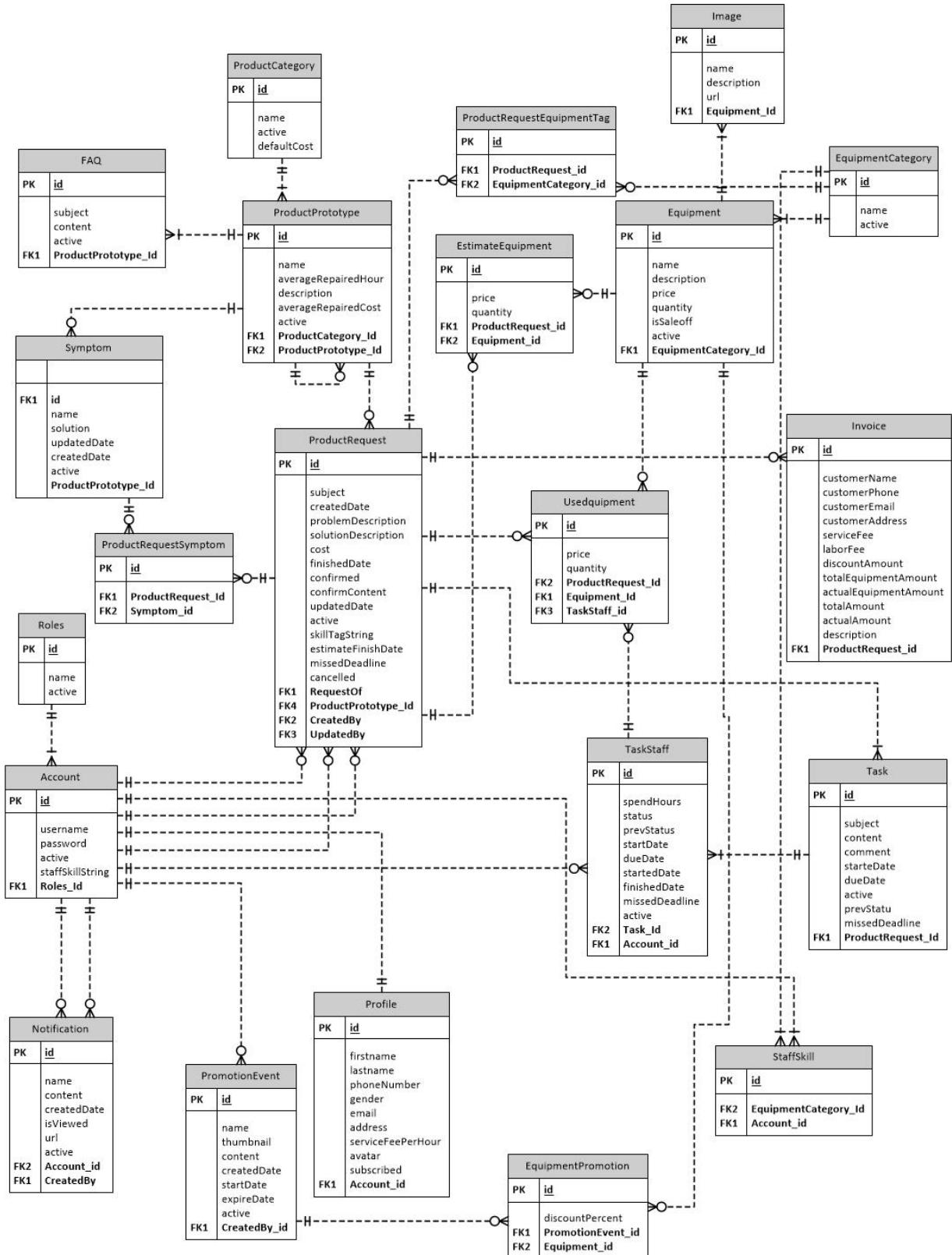


Figure 177 Logical Diagram**7.2. Relations Schema**

Name	Description
Account	Stores accounts in the system
Profile	Stores basic information of an account
Notification	Stores actions using for notification
Roles	Stores some roles using in the system
ProductRequest	Stores some basic information of requests in the system
Invoice	Stores invoice information of request
ProductPrototype	Stores some information of product prototypes exist in the system
ProductCategory	Stores some information of categories of product prototype
Equipment	Stores some information of equipments in the system
Image	Stores images of equipment
EquipmentCategory	Stores some information of categories of equipment
FAQ	Stores some information of faqs provide for customer
Symptom	Stores some symptoms information after repair or maintenance
Task	Stores some information of tasks of staff
Taskstaff	Association table between Task and Account table
PromotionEvent	Store promotion information
EquipmentPromotion	Association table between Equipment and PromotionEvent, stores discount percent of equipment
UsedEquipment	Association table between Equipment and ProductRequest, stores used equipments when repair or maintenance
EstimateEquipment	Association table between Equipment and ProductRequest, stores estimate equipments before repair or maintenance
ProductRequestEquipmentTag	Association table between ProductRequest and EquipmentCategory, stores tags of a request
ProductrequestSymptom	Association table between Product and Symptom
StaffSkill	Association table between Account and EquipmentCategory, stores skills of staffs

Table 25-Relationship Schema description

7.3. Relations Schema Details

Account

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of account Identity (seed = 1, increment = 1)
username	VARCHAR(20)		x	x	username of system
password	VARCHAR(20)		x		password of system
active	BIT		x		current account is active or not
staffSkillString	VARCHAR(500)				skill of an staff, delemiter by comma
Roles_id	INT	FK			id of Roles

Table 26-Table Account data dictionary

Profile

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of profile Identity (seed = 1, increment = 1)
firstname	NVARCHAR(20)		x		firstname of account
lastname	NVARCHAR(20)				lastname of account
phoneNumbe r	VARCHAR(13)		x		phone number of account
gender	VARCHAR(10)		x		gender of account
email	VARCHAR(50)		x		email of account
address	NVARCHAR(20 0)		x		address of account
serviceFeePe rHour	DOUBLE				fee per hour of a staff
avatar	VARCHAR(200)				avatar of account
subscribed	BIT		x		account is subscribed to get promotion event or not
Account_id	INT	FK	x	x	id of Account

Table 27-Table Profile data dictionary

Notification

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of Notification Identity (seed = 1, increment = 1)
name	NVARCHAR(20 0)		x		title of notification

content	NVARCHAR(300)		x		content of notification
createdDate	DATETIME		x		date of created notification
isViewed	BIT		x		notification is viewed or not
url	VARCHAR(100)		x		url of notification
active	BIT		x		notification is active or not
Account_id	INT	FK	x		id of Account that to be notified
CreatedBy	INT	FK	x		id of Account that created notification

Table 28-Table Notification data dictionary**Roles**

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of Roles Identity (seed = 1, increment = 1)
name	NVARCHAR(45)		x		name of role
Active	BIT		x		is active or not

Table 29-Table Roles data dictionary**ProductRequest**

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of ProductRequest Identity (seed = 1, increment = 1)
subject	NVARCHAR(100)		x		name of request
createdDate	DATETIME		x		date of created request
problemDescription	NVARCHAR(300)		x		problem of request
solutionDescription	NVARCHAR(300)				solution of request
cost	DOUBLE				final cost of request
finishedDate	DATETIME				date of finished request
confirmed	BIT		x		customer is confirmed or not
confirmedContent	NVARCHAR(100)				content of customer confirmed
updateDate	DATETIME		x		date of update request
active	BIT		x		request is active or not
skillTagString	VARCHAR(500)				request is related with which equipment category, delimiter by comma
requesttype	NVARCHAR(20)		x		type of request (Repair / Maintenance)

status	NVARCHAR(20)		x		status of request
estimateFinishDate	DATETIME		x		date of estimate finish request
missedDeadline	BIT		x		is request missed deadline or not
cancelled	BIT		x		is request cancelled or not
RequestOf	INT	FK	x		id of Account of customer
ProductPrototype_id	INT	FK	x		id of related ProductPrototype
CreatedBy	INT	FK	x		id of Account that created request
UpdateBy	INT	FK	x		id of Account that updated request

Table 30-Table ProductRequest data dictionary**Invoice**

Column Name	Type	P/F Key	Not Null	Unique	Description
Id	INT	PK	x		Id of Invoice Identity (seed = 1, increment = 1)
customerName	NVARCHAR(40)		x		full name of customer
customerPhone	VARCHAR(13)		x		phone number of customer
customerEmail	VARCHAR(50)		x		email of customer
customerAddress	NVARCHAR(200)		x		address of customer
serviceFee	DOUBLE		x		default service fee depend on which product category
laborFee	DOUBLE		x		total cost that staff spent for request
discountAmount	DOUBLE		x		total discount amount
totalEquipmentAmount	DOUBLE		x		total price of used equipment
actualEquipmentAmount	DOUBLE		x		actual price of used equipment
totalAmount	DOUBLE		x		total amount of invoice
actualAmount	DOUBLE		x		actual amount of invoice
Description	NVARCHAR(100)				description of invoice
ProductRequest_id	INT	FK	x	x	id of ProductRequest

Table 31-Table Invoice data dictionary**ProductPrototype**

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of ProductPrototype Identity (seed = 1, increment = 1)
name	NVARCHAR(100)		x		name of product prototype
description	NVARCHAR(500)		x		description of product prototype
averageRepairedCost	DOUBLE				average repaired cost of request belong to prototype
averageRepairedHour	DOUBLE				average repaired hour of request belong to prototype
active	BIT		x		is active or not
ProductCategory_id	INT	FK	x		id of ProductCategory
ProductPrototype_id	INT	FK			id of ProductPrototype

Table 32-Table ProductPrototype data dictionary**ProductCategory**

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of ProductCategory Identity (seed = 1, increment = 1)
name	NVARCHAR(100)		x		name of product category
active	BIT		x		is active or not
defaultCost	DOUBLE		x		default cost when repair

Table 33-Table ProductCategory data dictionary**Equipment**

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of Equipment Identity (seed = 1, increment = 1)
name	NVARCHAR(100)		x		name of equipment
description	NVARCHAR(300)		x		description of equipment
price	DOUBLE		x		price of equipment

isSaleoff	BIT		x		is equipment in sale off period
active	BIT		x		is active or not
EquipmentCategory_id	INT	FK	x		id of equipment category

Table 34-Table Equipment data dictionary**Image**

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of Image Identity (seed = 1, increment = 1)
name	NVARCHAR(100)		x		name or image
description	NVARCHAR(200)		x		description of image
url	VARCHAR(100)		x		url of image
Equipment_id	INT	FK	x		id of equipment

Table 35-Table Image data dictionary**EquipmentCategory**

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of EquipmentCategory Identity (seed = 1, increment = 1)
name	NVARCHAR(100)		x		name of equipment category
active	BIT		x		is active or not

Table 36-Table EquipmentCategory data dictionary**Symptom**

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of Symptom Identity (seed = 1, increment = 1)
name	NVARCHAR(100)		x		name of symptom
solution	NVARCHAR(200)				solution of symptom
updatedDate	DATETIME		x		date of updated symptom
createdDate	DATETIME		x		date of created symptom
active	BIT		x		is active or not
ProductProto	INT	FK	x		id of ProductPrototype

type_id					
---------	--	--	--	--	--

Table 37-Table Symptom data dictionary**FAQ**

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of FAQ Identity (seed = 1, increment = 1)
subject	NVARCHAR(200)		x		title of FAQ
content	NVARCHAR(1000)		x		content of FAQ
active	BIT		x		is active or not
ProductProto type_id	INT	FK	x		id of ProductPrototype

Table 38-Table FAQ data dictionary**Task**

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of Task Identity (seed = 1, increment = 1)
subject	NVARCHAR(100)		x		title of task
content	NVARCHAR(1000)		x		content of task
comment	NVARCHAR(1000)				comment of task
startDate	DATETIME		x		date of estimate start task
dueDate	DATETIME		x		date of estimate finish task
active	BIT		x		is active or not
status	NVARCHAR(20)		x		current status of task
preStatus	NVARCHAR(20)		x		previous status of task
missedDeadline	BIT		x		is missed deadline or not
ProductRequest_id	INT	FK	x		id of ProductRequest

Table 39-Table Task data dictionary**TaskStaff**

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of TaskStaff

					Identity (seed = 1, increment = 1)
spendHours	DOUBLE		x		time staff spent for task
status	NVARCHAR(20)		x		status of current task of staff
prevStatus	NVARCHAR(20)		x		previous status of task of staff
startDate	DATETIME		x		date of estimate start task
dueDate	DATETIME		x		date of estimate finish task
startedDate	DATETIME		x		date of started task
finishedDate	DATETIME		x		date of finished task
missedDeadline	BIT		x		is missed deadline or not
active	BIT		x		is active or not
Account_id	INT	FK	x		id of Account of staff
Task_id	INT	FK	x		id of Task

Table 40-Table TaskStaff data dictionary

PromotionEvent

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of PromotionEvent Identity (seed = 1, increment = 1)
name	NVARCHAR(200)		x		name of promotion
thumbnail	VARCHAR(200)		x		url image of thumbnail
content	TEXT		x		content of promotion
createdDate	DATETIME		x		date of created promotion
startDate	DATETIME		x		date of start promotion
expireDate	DATETIME		x		date of finish promotion
active	BIT		x		is active or not
CreateBy_id	INT	FK	x		id of Account

Table 41-Table PromotionEvent data dictionary

EquipmentPromotion

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of EquipmentPromotion Identity (seed = 1, increment = 1)
discountPercent	DOUBLE		x		percent of discount
PromotionEvent_id	INT	FK	x		id of PromotionEvent
Equipment_id	INT	FK	x		id of Equipment

Table 42-Table EquipmentPromotion data dictionary

UsedEquipment

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of UsedEquipment Identity (seed = 1, increment = 1)
price	DOUBLE		x		price of equipment
quantity	INT		x		quantity of equipment
ProductRequest_id	INT	FK	x		id of ProductRequest
Equipment_id	INT	FK	x		id of Equipment
TaskStaff_id	INT	FK	x		id of TaskStaff

Table 43-Table UsedEquipment data dictionary

EstimateEquipment

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of EstimateEquipment Identity (seed = 1, increment = 1)
price	DOUBLE		x		price of equipment
quantity	INT		x		quantity of equipment
ProductRequest_id	INT	FK	x		id of ProductRequest
Equipment_id	INT	FK	x		id of Equipment

Table 44-Table EstimateEquipment data dictionary

ProductRequestEquipmentTag

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of ProductRequestEquipmentTag Identity (seed = 1, increment = 1)
ProductRequest_id	INT	FK	x		id of ProductRequest
EquipmentCategory_id	INT	FK	x		id of EquipmentCategory

Table 45-Table ProductRequestEquipmentTag data dictionary

ProductRequestSymptom

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of ProductrequestSymptom

					Identity (seed = 1, increment = 1)
ProductRequest_id	INT	FK	x		id of ProductRequest
Symptom_id	INT	FK	x		id of Symptom

Table 46-Table ProductRequestSymptom data dictionary**StaffSkill**

Column Name	Type	P/F Key	Not Null	Unique	Description
id	INT	PK	x		Id of StaffSkill Identity (seed = 1, increment = 1)
Account_id	INT	FK	x		id of Account of staff
EquipmentCategory_id	INT	FK	x		id of EquipmentCategory

Table 47-Table StaffSkill data dictionary

7.4. Database Implementation

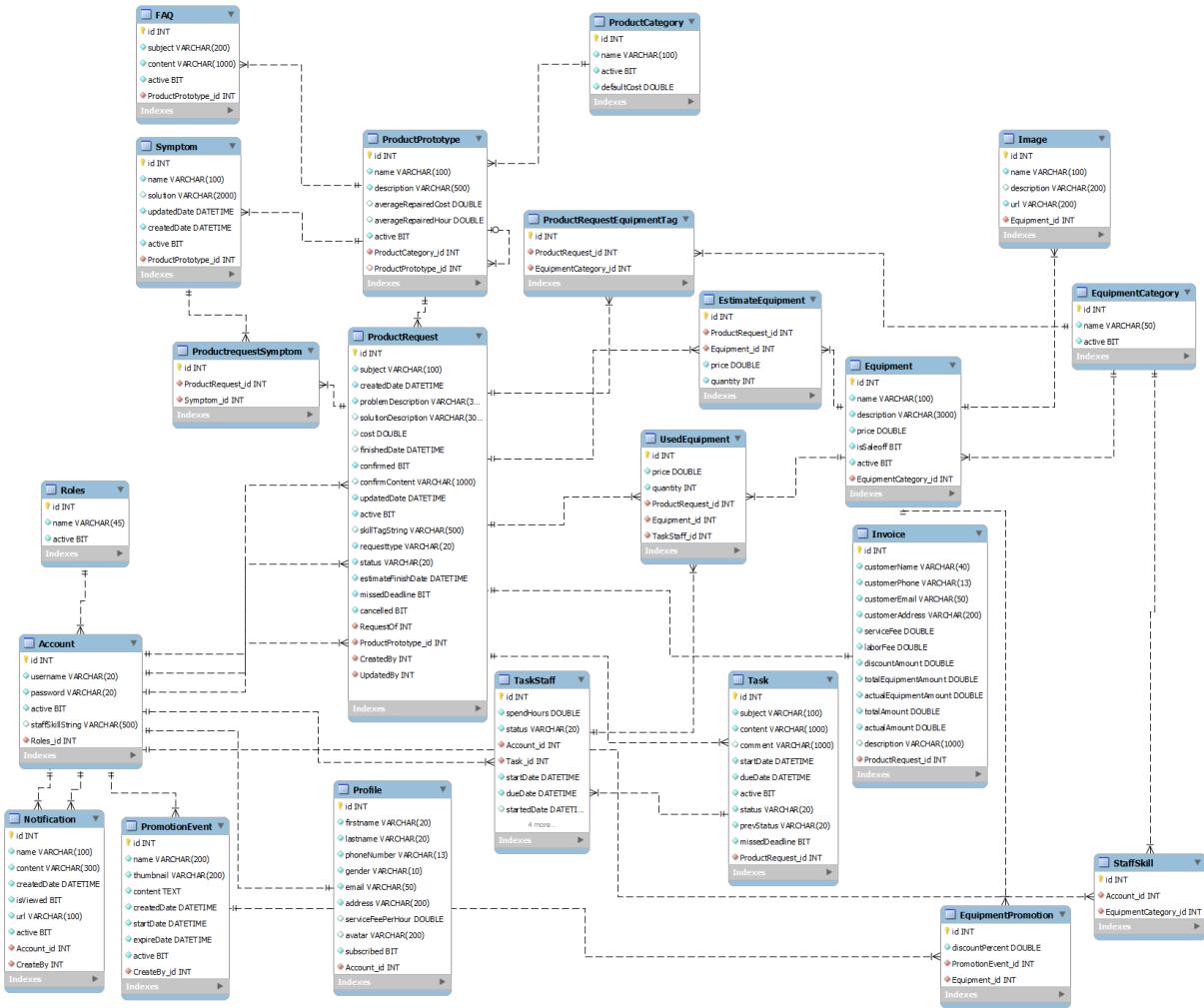


Figure 178 - Physical Diagram

V. Software Test Documentation

1. Introduction

This section provides all the features about test plans, test cases, test results, test environments, pass/fail criteria and risks estimation as well as checklist to cover all cases that can happen in MRS system.

2. Test Approach

- Test goal: To find all remain bugs in MRS system.
- Test type: White box testing.
- Test size: System Component.
- Test technique: Check list.

3. Test Plan

Test plan document is used to verify and ensure that MRS system meets its design specifications and other user requirements. The following part will describe which features will be tested and which is not, test tools and test environment.

3.1. Features to be tested

3.1.1. *Request Management*

3.1.2. *Task Management*

3.1.3. *Symptom Management*

3.1.4. *Product Management*

3.1.5. *Equipment Management*

3.1.6. *FAQ Management*

3.1.7. *Statistics*

3.1.8. *Account Management*

3.1.9. *Invoice Management*

3.1.10. *Staff Tracking Task*

3.1.11. *Customer Tracking Request*

3.2. Features not to be tested

3.2.1. *Login*

3.2.2. *Register*

3.2.3. *Change Password*

3.2.4. *Update Profile*

3.2.5. *View Equipment*

3.3. Test Tool and Environment

- Test tool: There is no special testing tool used in testing this product.
- Test environment:
 - o Hardware: Intel Core 2 Duo 2.00GHz or above, RAM at least 2GB, HDD free at least 1GB, Screen with 1024x768 or higher resolution.
 - o Operating System: Microsoft Windows 7.
 - o Browser: Firefox Nightly 25+, Google Chrome 27+.

4. Test Cases

4.1. Customer - [Tracking Request] View Request List

The screenshot shows a web application interface. At the top, there is a navigation bar with the logo "MRS solution", links for "HOME", "ABOUT", "EQUIPMENT", and "FAQ", and a user session indicator "Hello customer" with "0 News" and a "LOGOUT" button. Below the navigation bar is a sidebar titled "PERSONAL" containing "Basic" and "Change Password" links, and a "SERVICES" section with a "Tracking Request" link. The main content area is titled "Tracking Request" and contains a "REQUEST LIST" table. The table has columns for "NO.", "REQUEST NAME", "MODEL", "STATUS", and "REQUEST DATE". It lists four requests:

NO.	REQUEST NAME	MODEL	STATUS	REQUEST DATE
1	Request 1	NB Acer Aspire AO756 - 877BCrr (001) (Đỏ)	In Progress	2013-07-22T23:26:39
2	Request 5	NB Acer Aspire AO756 - 877BCrr (001) (Đỏ)	Confirmed	2013-07-23T13:46:41
3	Request 11	NB Acer Aspire AO756 - 877BCrr (001) (Đỏ)	Finished	2013-07-23T15:30:06
4	request test	NB Acer Aspire AO756 - 877BCrr (001) (Đỏ)	Pending	2013-07-24T15:46:11

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps	1. Go to [Home page]. 2. Login with roles of Customer 3. Click on [Request box].					
	Components						
TC_CTR_V_001	View all request of Customer.	-	- Show table Request List with fields contain.		Pass	Jul-23-2013	

4.2. Customer - [Tracking Request] Update Request

The screenshot shows a 'REQUEST DETAIL' window with the following details:

- Request Name:** Request 1
- Request Type:** Repair
- Model:** NB Acer Aspire AO756 - 87"
- Status:** Pending (highlighted with a red box)
- Problem describe:** Bad sector
- Equipment:** MB Intel Box DB85FL
- Customer Confirmation:** A rich text editor toolbar.
- Buttons:** CANCEL (orange) and CONFIRM (orange).

The screenshot shows the same 'REQUEST DETAIL' window, but the status has been updated to Confirmed (highlighted with a red box). The other fields remain the same as in the previous screenshot.

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note	
	Steps	<ol style="list-style-type: none"> 1. Go to [Home page]. 2. Login with roles of Customer 3. Click on [Request box]. 						
	Components							
TC_CTRU_001	View detail of request	4. Click on a [Pending] request.	- Show [Request Detail] form on popup with its components. - Display 3 buttons: [Close], [Cancel], [Submit].		Pass	Jul-23-2013		

TC_CTRU_002	View detail of request	5. Click on a [In Progress] request.	- Show [Request Detail] form on popup with its components. - Display [Close] button.		Pass	Jul-23-2013
TC_CTRU_003	View detail of request	6. Click on a [Finished] request.	- Show [Request Detail] form on popup with its components. - Display [Close] button.		Pass	Jul-23-2013
TC_CTRU_004	Customer confirm the request	4. Click on a [Pending] request. 5. Fill in [Confirmation] textbox. 6. Click on [Submit] button.	- Request's status is updated to [Confirmed]. - Push a notification to relate Manager - The status of request is updated in [Request List] table.		Pass	Jul-23-2013
TC_CTRU_005	Customer confirm the request	4. Click on a [Pending] request. 5. Let the [Confirmation] textbox empty. 6. Click on [Submit] button.	- Display error message on [Error] bar. - Status of request is not updated.		Pass	Jul-23-2013
TC_CTRU_006	Customer cancel the request	4. Click on a [Pending] request. 5. Click on [Cancel] button.	- Request's status is updated to [Canceled]. - Push a notification to relate Manager - The status of request is updated in [Request List] table.		Pass	Jul-23-2013

4.3. Customer - [Tracking Request] Push/Receive Notification

4.4. Staff - [Tracking Task] View All Tasks List

The screenshot shows a task management interface. At the top, there's a navigation bar with 'HOME', 'ABOUT', 'EQUIPMENT', 'FAQ', and a user profile 'Hello staff1'. Below the navigation is a search bar with '0 News' and a 'LOGOUT' button. The main area is titled 'TASK LIST' and displays a table with three rows of task data. The columns are labeled 'NO.', 'TASK NAME', 'STATUS', 'START DATE', 'FINISH DATE', and 'ACTIVATION'. The first two rows have 'New' status, while the third has 'In Progress'. The 'SEARCH' input field at the top right and the 'Activation' column are highlighted with a red box.

TASK LIST					
Total tasks: 3					
NO.	TASK NAME	STATUS	START DATE	FINISH DATE	ACTIVATION
1	task 2.1	New	07/26/2013 22:28	07/31/2013 22:28	<input checked="" type="checkbox"/>
2	Task 2.1	New	07/23/2013 01:32	07/27/2013 01:38	<input checked="" type="checkbox"/>
3	Task 1.1	In Progress	07/23/2013 01:30	07/24/2013 06:30	<input type="checkbox"/>

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps	<ol style="list-style-type: none"> 1. Go to [Home page]. 2. Login with roles of Staff 3. Click on [Task box]. 					
TC_STT_V_001	Components		- Show table Task List with Tasks belong to that account.		Pass	Jul-23-2013	

4.5. Staff - [Tracking Task] Filter Task

The screenshot shows a task management interface similar to the previous one, but with a different filter applied. The 'SEARCH' input field at the top right is highlighted with a red box. The main area displays a table with three rows of task data, matching the structure of the previous screenshot but with different dates in the 'START DATE' and 'FINISH DATE' columns.

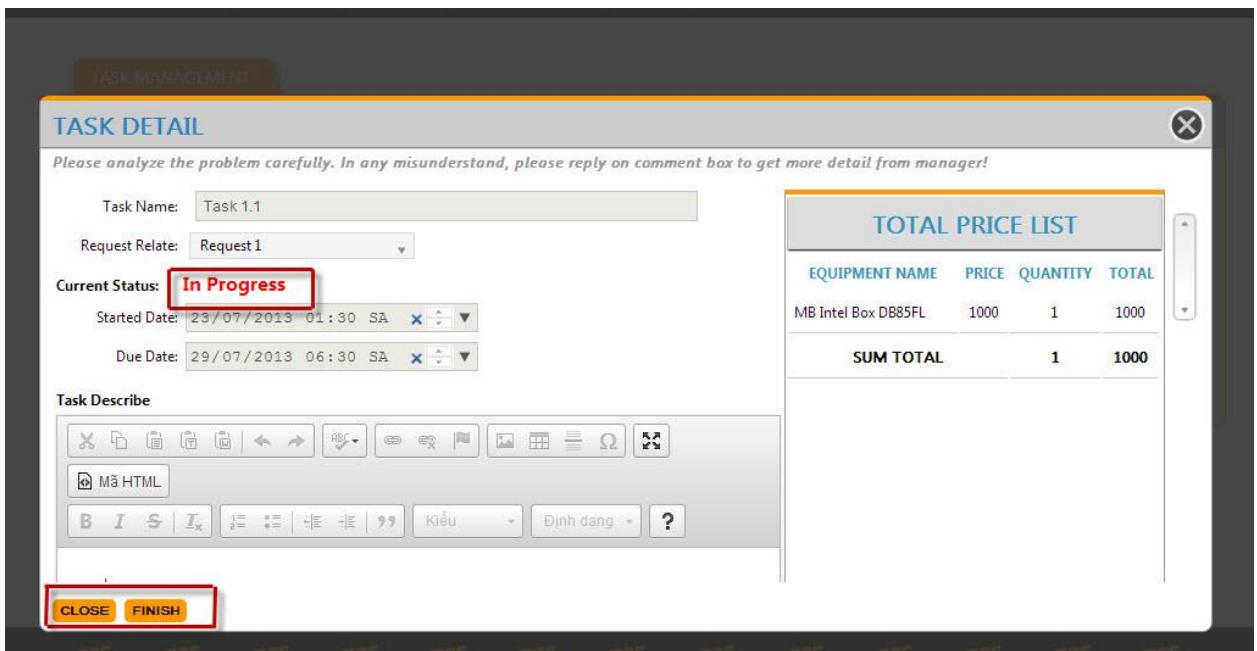
TASK LIST					
Total tasks: 3					
NO.	TASK NAME	STATUS	START DATE	FINISH DATE	ACTIVATION
1	task 2.1	New	07/26/2013 22:28	07/31/2013 22:28	<input checked="" type="checkbox"/>
2	Task 1.1	New	07/23/2013 01:30	07/29/2013 06:30	<input checked="" type="checkbox"/>
3	Task 2.1	New	07/23/2013 01:32	07/27/2013 01:38	<input checked="" type="checkbox"/>

ID	Description	Test Case	Expected Output	Dependence	Result	Test	Note
----	-------------	-----------	-----------------	------------	--------	------	------

		Procedure				date	
	Steps	1. Go to [Home page]. 2. Login with roles of Staff 3. Click on [Task box].					
	Components						
TC_STTF_001			- Show form [Search] above table Task List.		Pass	Jul-23-2013	
TC_STTF_002		4. Fill value want to search. 5. Choose [Name] to search belong to [Task Name], [Status]: to search belong to status of tasks. 6. Click on [Search] button.	- Display Task List belongs to the result.		Pass	Jul-23-2013	

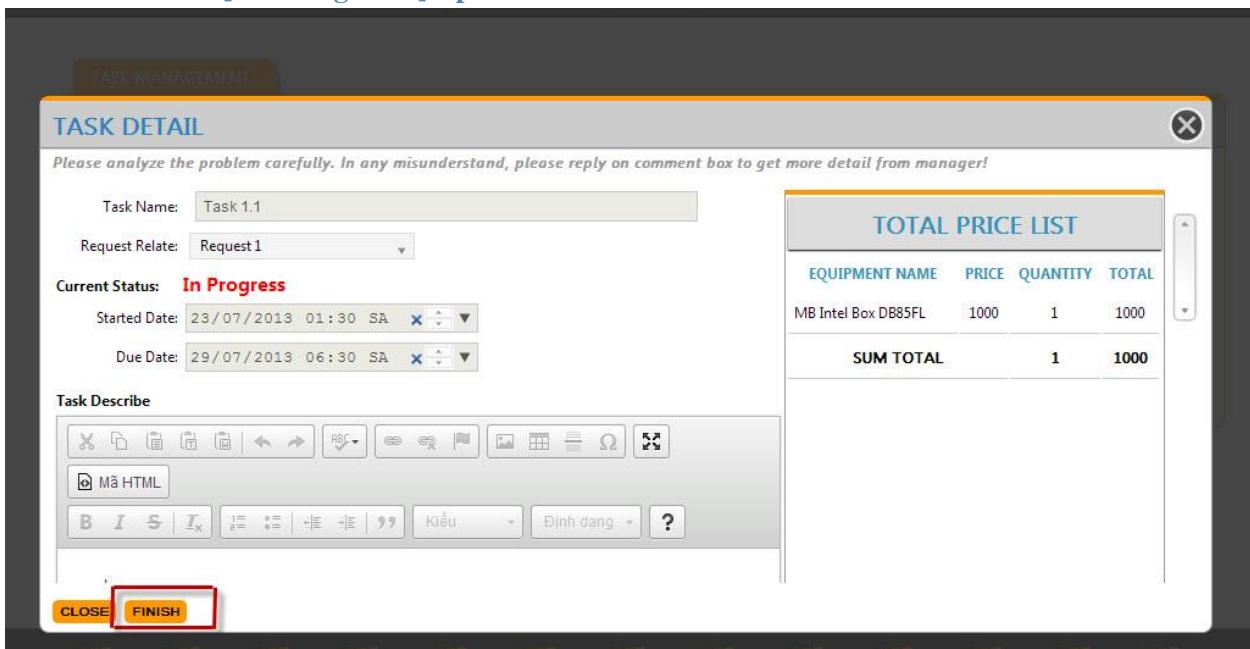
4.6. Staff - [Tracking Task] View Task Detail

The screenshot shows a software interface titled "TASK MANAGEMENT". A modal window titled "TASK DETAIL" is open. Inside, there's a message: "Please analyze the problem carefully. In any misunderstand, please reply on comment box to get more detail from manager!". The "Task Name" field contains "task 2.1". The "Request Relate" dropdown is set to "Request 1". The "Current Status" field is highlighted with a red box and shows "New". Below it, the "Started Date" is "26/07/2013 10:28 CH" and the "Due Date" is "31/07/2013 10:28 CH". A "Task Describe" section contains the text "thay CPU" and includes a rich text editor toolbar. At the bottom of the modal are two buttons: "CLOSE" and "START", with "START" also being highlighted with a red box.



ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps	1. Go to [Home page]. 2. Login with roles of Staff 3. Click on [Task box]. 4. Click on a record of table Task List.					
	Components						
TC_STTD_001	Case 1: Task has not start yet.		- Show [Task Detail] form on popup with details information and [Close], [Start] button.		Pass	Jul-23-2013	
TC_STTD_002	Case 2: Task start already		- Show [Task Detail] form on popup with details information and [Close], [Finish] button.		Pass	Jul-23-2013	
TC_STTD_003	Case 3: Task was finish.		- Show [Task Detail] form on popup with details information and [Close] button.		Pass	Jul-23-2013	

4.7. Staff - [Tracking Task] Update Task



ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
Steps							
1. Go to [Home page]. 2. Login with roles of Staff 3. Click on [Task box]. 4. Click on a record of table Task List.							
Components							
TC_STTU_001	According Case 2: Task start already		- Show [Task Detail] form on popup with details information about selected task.		Pass	Jul-23-2013	
TC_STTU_002		5. Fill in fields that want to update. 6. Click [Finish] button	- Updated task is updated to database		Pass	Jul-23-2013	
TC_STTU_003		5. [Comment for more issues] is blank 6. Click [Finish] button	- Display error message - Updated task is not updated to database		Pass	Jul-23-2013	

TC_STTU_004		5. Spend hours is not real	- Display error message - Updated task is not updated to database		Pass	Jul-23-2013	
TC_STTU_005		5. Fill in fields that want to update. 6. Click [Close] button	- Updated task is not updated to database - Close popup [Task Detail]		Pass	Jul-23-2013	

4.8. Staff - [Tracking Task] Push/Receive Notification

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note	
	Steps	1. Go to [Home page]. 2. Login with roles of Staff 3. Click on [Task box].						
	Components							
TC_STTP_00		4. Update a task to Task List	- New notification is send to Manager.		Pass	Jul-23-2013		

4.9. Technical Manager - [Manage FAQ] Create FAQ

- 4.10. Technical Manager - [Manage FAQ] View all FAQ list
- 4.11. Technical Manager - [Manage FAQ] View FAQ detail
- 4.12. Technical Manager - [Manage FAQ] Update FAQ
- 4.13. Technical Manager - [Manage repair/maintenance order] Create Request

The screenshot shows the 'ADD REQUEST - PRODUCT PROBLEM' dialog box. At the top, it says 'Please describe the problem, MRS system will help you to determine more exactly!'. It has two dropdown menus: 'Product Prototype' set to 'Acer' and 'Product Model' set to 'NB Acer Aspire AO756 - 8...'. Below these is a section for 'Estimate Equipment' with a 'ADD EQUIPMENT' button. The main area is for 'Problem describe(*)' and contains a rich text editor with a toolbar for text, tables, and images. At the bottom of the dialog are 'CLOSE', 'ADD', and 'REFRESH' buttons.

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
Steps	7. Go to [Home page]. 8. Login with roles of Manager 9. Click on [Request box]. 10. Click on [ADD REQUEST] button.						
Components							
TC_MRC_001			- Show [Add Request- Product Problem] form on popup with its components.		Pass	Jul-24-2013	
TC_MRC_002		11. Fill all required fields. 12. Click [Add] button.	- New request is inserted to database.		Pass	Jul-24-2013	

TC_MRC_003		5. Fill some required fields are blank. 6. Click [Add] button.	- Display error message on required fields. - New request is not inserted to database		Pass	Jul-24-2013	
TC_MRC_004		5. Fill all required information. 6. Choose [Estimate Finish] before the day execute test. 7. Click [Add] button.	- Display invalid message on [Estimate Finish] input. - New request is not inserted to database		Pass	Jul-24-2013	
TC_MRC_005		5. Fill all required information. 6. [Problem description] is blank. 7. Click [Add] button.	- Display error message. - New request is not inserted to database		Fail	Jul-24-2013	
TC_MRC_006		5. Fill in fields on [Add Request- Product Problem] form. 6. Click [Close] button.	- New request is not inserted to database. - Close popup [Add Request- Product Problem]		Pass	Jul-24-2013	

4.14. Technical Manager - [Manage repair/maintenance order] View All Request List

The screenshot shows the 'REQUEST LIST' page of the MRS Solution software. At the top, there is a navigation bar with links for HOME, ABOUT, EQUIPMENT, FAQ, STATISTIC, ACCOUNT, and INVOICE. On the right side of the header, there is a 'Hello manager' greeting, a '0 News' badge, and a 'LOGOUT' button. Below the header, there is a toolbar with buttons for REQUEST, TASK, SYMPTOM, PRODUCT, EQUIPMENT, FAQ, STATISTIC, ACCOUNT, and INVOICE. The main content area is titled 'REQUEST LIST' and contains a table with 10 rows of data. The table has columns for NO., CODE, REQUEST NAME, STATUS, CREATED DATE, ESTIMATE FINISH DATE, and ACTIVE. The first row is highlighted in blue. At the bottom of the table, there are buttons for 'Prev', '1', '2', 'Next', and 'REFRESH'. The entire table area is enclosed in a red box.

NO.	CODE	REQUEST NAME	STATUS	CREATED DATE	ESTIMATE FINISH DATE	ACTIVE
1	PR094	Request test FPT	Pending	07/24/2013 15:56	07/25/2013 15:53	<input checked="" type="checkbox"/>
2	PR093	request test	Pending	07/24/2013 15:46	07/24/2013 15:45	<input checked="" type="checkbox"/>
3	PR092	request 2	Finished	07/24/2013 15:44	07/24/2013 15:43	<input checked="" type="checkbox"/>
4	PR091	request 1	Finished	07/24/2013 15:30	07/24/2013 15:29	<input checked="" type="checkbox"/>
5	PR090	Request 12	Pending	07/23/2013 15:41	07/23/2013 15:40	<input checked="" type="checkbox"/>
6	PR089	Request 11	Finished	07/23/2013 15:30	07/23/2013 15:29	<input checked="" type="checkbox"/>
7	PR088	Request 10	Finished	07/23/2013 15:23	07/23/2013 15:22	<input checked="" type="checkbox"/>
8	PR087	Request 9	Confirmed	07/23/2013 15:15	07/23/2013 15:15	<input checked="" type="checkbox"/>
9	PR086	Request 8	Confirmed	07/23/2013 15:09	07/23/2013 15:08	<input checked="" type="checkbox"/>
10	PR085	Request 7	Finished	07/23/2013 15:00	07/23/2013 15:00	<input checked="" type="checkbox"/>

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps						
	4. Go to [Home page]. 5. Login with roles of Manager 6. Click on [Request box].						
TC ₁ MRV_001	Components		- Show table Request List with fields contain and show on paging.		Pass	Jul-24-2013	

4.15. Technical Manager - [Manage repair/maintenance order] Filter Request

The screenshot shows the 'REQUEST LIST' page of the MRS solution software. At the top, there is a navigation bar with links: HOME, ABOUT, EQUIPMENT, FAQ, STATISTIC, ACCOUNT, and INVOICE. On the far right of the header, it says 'Hello manager' with a 'LOGOUT' button, and there is a 'News' icon with the number '0'.

The main area is titled 'REQUEST LIST'. It shows a table with 5 rows of request data:

NO.	CODE	REQUEST NAME	STATUS	CREATED DATE	ESTIMATE FINISH DATE	ACTIVE
1	PR091	request 1	Finished	07/24/2013 15:30	07/24/2013 15:29	<input checked="" type="checkbox"/>
2	PR090	Request 12	Pending	07/23/2013 15:41	07/23/2013 15:40	<input checked="" type="checkbox"/>
3	PR089	Request 11	Finished	07/23/2013 15:30	07/23/2013 15:29	<input checked="" type="checkbox"/>
4	PR088	Request 10	Finished	07/23/2013 15:23	07/23/2013 15:22	<input checked="" type="checkbox"/>
5	PR076	Request 1	Confirmed	07/22/2013 23:26	07/23/2013 23:26	<input checked="" type="checkbox"/>

Below the table, there are buttons for 'Prev', '1', 'Next', and 'REFRESH'. To the right of the table, there is a search bar with fields for 'Name' and 'Status' and a 'SEARCH' button.

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps	1. Go to [Home page]. 2. Login with roles of Manager 3. Click on [Request box].					
	Components						
TC_MRF_001			- Show form [Search] above table Request List.		Pass	Jul-23-2013	
TC_MRF_002		4. Fill value want to search. 5. Choose [Name] to search belong to [Request Name], [Status]: to search belong to status of requests. 6. Click on [Search] button.	- Display Request List belong to the result.		Pass	Jul-23-2013	

4.16. Technical Manager - [Manage repair/maintenance order] View Request Detail

The following information is the detail of this request

TOTAL ESTIMATE EQUIPMENTS PRICE LIST			
EQUIPMENT NAME	PRICE	QUANTITY	TOTAL
8GB DDR3 Kingmax 1333	1410	2	2820
VGA 1GB Zotac GT420	1230	1	1230
SUM TOTAL		3	4050

Request Name: Request test FPT
Estimate Finish: 25/07/2013 03:53 CH
Customer Relate: customer
Request Type: Repair
Product Prototype: Acer
Product Model: NB Acer Aspire A0756 - 8...
 Active

Estimate Equipment

Category: RAM	Equipment: 8GB DDR3 Kingmax 1333	Quantity: 2
Category:	Equipment:	Quantity:

CLOSE **UPDATE** **CANCEL REQUEST**

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note	
	Steps	<ol style="list-style-type: none"> 1. Go to [Home page]. 2. Login with roles of Manager 3. Click on [Request box]. 4. Click on a record of table Request List. 						
	Components							
TC_MR01			- Show [Request Detail] form on popup with details information about selected request.		Pass	Jul-23-2013		

4.17. Technical Manager - [Manage repair/maintenance order] Update Request

The following information is the detail of this request

Request Name:	Request test FPT
Estimate Finish:	25/07/2013 03:53 CH
Customer Relate:	customer
Request Type:	Repair
Product Prototype:	Acer
Product Model:	NB Acer Aspire A0756 - ...

Active

Estimate Equipment

Category:	Equipment:	Quantity:
RAM	8GB DDR3 Kingmax 1333	2

TOTAL ESTIMATE EQUIPMENTS PRICE LIST

EQUIPMENT NAME	PRICE	QUANTITY	TOTAL
8GB DDR3 Kingmax 1333	1410	2	2820
VGA 1GB Zotac GT420	1230	1	1230
SUM TOTAL		3	4050

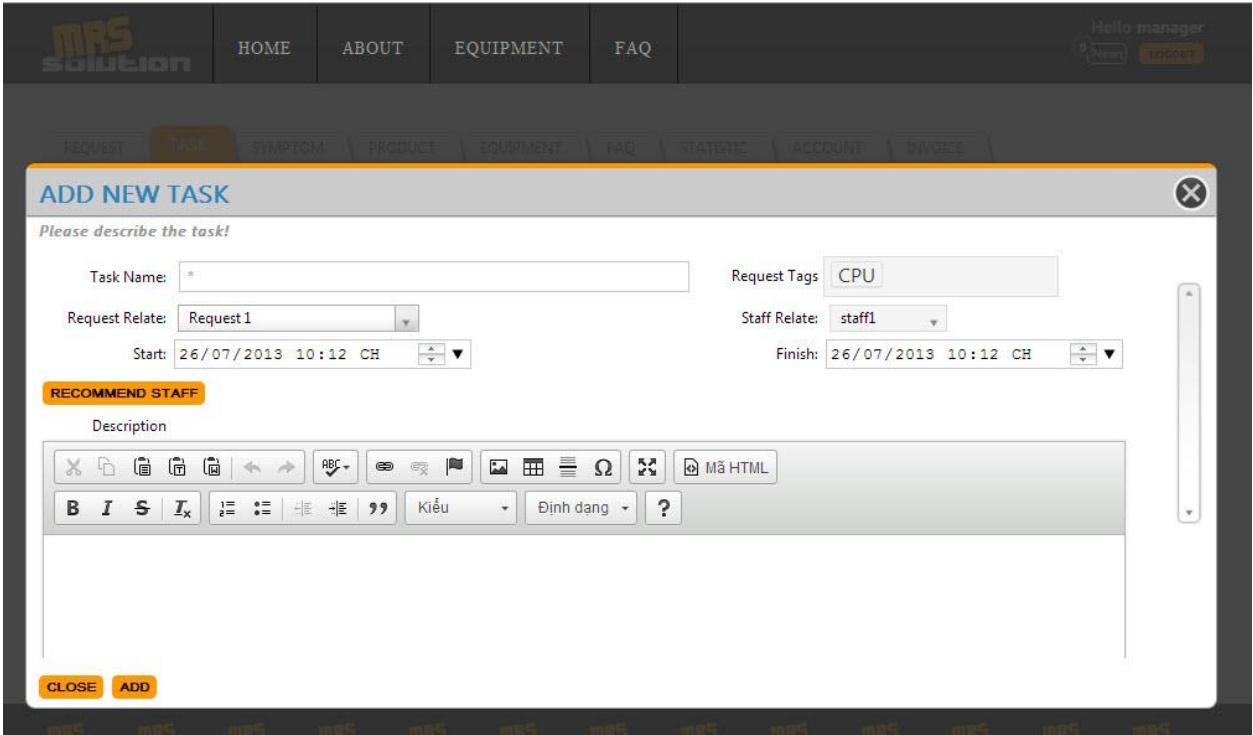
Buttons: CLOSE, UPDATE (highlighted), CANCEL REQUEST

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps	1. Go to [Home page]. 2. Login with roles of Manager 3. Click on [Request box]. 4. Click on a record of table Request List.					
	Components						
TC_MRU_001			- Show [Request Detail] form on popup with details information about selected request.		Pass	Jul-23-2013	
TC_MRU_002		5. Fill in fields that want to update. 6. Click [Update] button	- Updated request is updated to database		Pass	Jul-23-2013	
TC_MRU_003		1. Fill in fields that want to update. 2. Click [Close] button	- Updated request is not updated to database - Close popup [Request Detail]		Pass	Jul-23-2013	

4.18. Technical Manager - [Manage repair/maintenance order] Push/Receive Notification

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note	
	Steps	1. Go to [Home page]. 2. Login with roles of Manager 3. Click on [Request box].						
	Components							
TC ₋ MRP_001		4. Add new request to Request List.	- New notification is send to according customer.		Pass	Jul-23-2013		
TC ₋ MRP_002		4. Update a request to Request List	- New notification is send to according customer.		Pass	Jul-23-2013		

4.19. Technical Manager - [Manage Task] Create Task



The screenshot shows the MRS Solution software interface. At the top, there's a navigation bar with links like HOME, ABOUT, EQUIPMENT, FAQ, and STATISTIC. A user 'Hello manager' is logged in. Below the navigation bar, there's a main menu with tabs: REQUEST, TASK (which is selected), SYMPTOM, PRODUCT, EQUIPMENT, FAQ, STATISTIC, ACCOUNT, and INVOICE. A sub-menu for 'REQUEST' is open, showing 'ADD NEW REQUEST'. A large 'ADD NEW TASK' dialog box is centered on the screen. It has fields for 'Task Name' (with a placeholder '*'), 'Request Tags' (set to 'CPU'), 'Request Relate' (set to 'Request1'), 'Staff Relate' (set to 'staff1'), 'Start' (date and time: 26/07/2013 10:12 CH), and 'Finish' (date and time: 26/07/2013 10:12 CH). Below these fields is a 'RECOMMEND STAFF' button. Underneath is a 'Description' section with a rich text editor toolbar containing various icons for bold, italic, underline, etc. At the bottom of the dialog box are 'CLOSE' and 'ADD' buttons. The background of the main window shows a repeating 'mrs' logo pattern.

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note	
	Steps	1. Go to [Home page].						

	2. Login with roles of Manager 3. Click on [Request box]. 4. Click on [Task] tab 5. Click on [ADD TASK] button.					
	Components					
TC_MTC_001			- Show [Add New Task] form on popup with its components.		Pass	Jul-23-2013
TC_MTC_002		6. Fill all required fields. 7. Click [Add] button.	- New task is inserted to database.		Pass	Jul-23-2013
TC_MTC_003		6. Fill some required fields are blank. 7. Click [Add] button.	- Display error message on required fields. - New task is not inserted to database		Pass	Jul-23-2013
TC_MTC_004		6. Fill all required information. 7. Choose [Finish Day] before [Start Day]. 8. Click [Add] button.	- Display invalid message on [Finish Day] input. - New task is not inserted to database		Pass	Jul-23-2013
TC_MTC_005		6. Fill all required information. 7. [Description] is blank. 8. Click [Add] button.	- Display error message. - New task is not inserted to database		Pass	Jul-23-2013
TC_MTC_006		6. Fill in fields on [Add New Task] form. 7. Click [Close] button.	- New task is not inserted to database. - Close popup [New Task]		Pass	Jul-23-2013

4.20. Technical Manager - [Manage Task] View All Tasks List

The screenshot shows the MRS solution software interface. At the top, there is a navigation bar with links: HOME, ABOUT, EQUIPMENT, FAQ, STATISTIC, ACCOUNT, and INVOICE. On the far right of the top bar, it says "Hello manager" with a "LOGOUT" button, and there is a "News" icon with the number "0". Below the top bar, there is a secondary navigation bar with tabs: REQUEST, TASK (which is highlighted in orange), SYMPTOM, PRODUCT, EQUIPMENT, FAQ, STATISTIC, ACCOUNT, and INVOICE. The main content area is titled "TASK LIST" and contains a table with three rows of task data. At the bottom of the table, there are buttons for "Prev", "1", "Next", and "REFRESH".

NO.	TASK NAME	STATUS	CREATED DATE	FINISH DATE	ASSIGNED TO
1	Task 1.1	New	07/23/2013 01:30	07/29/2013 06:30	staff1
2	Task 1.2	New	07/23/2013 01:30	07/29/2013 01:55	staff3
3	Task 2.1	New	07/23/2013 01:32	07/27/2013 01:38	KieuDTD

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps	1. Go to [Home page]. 2. Login with roles of Manager 3. Click on [Request box]. 4. Click on [Task] tab.					
	Components	TC_001 MTV					
			- Show table Task List with fields contain and show on paging.		Pass	Jul-23-2013	

4.21. Technical Manager - [Manage Task] Filter Task

The screenshot shows the 'TASK LIST' page. At the top, there's a navigation bar with tabs: REQUEST, TASK (which is selected and highlighted in orange), SYMPTOM, PRODUCT, EQUIPMENT, FAQ, STATISTIC, ACCOUNT, and INVOICE. On the far right of the header, there's a user interface with 'Hello manager', a '0 News' badge, and a 'LOGOUT' button. Below the header, the main content area is titled 'TASK LIST' and displays a table of tasks. The table has columns: NO., TASK NAME, STATUS, CREATED DATE, FINISH DATE, and ASSIGNED TO. There are three tasks listed:

NO.	TASK NAME	STATUS	CREATED DATE	FINISH DATE	ASSIGNED TO
1	Task 1.1	New	07/23/2013 01:30	07/29/2013 06:30	staff1
2	Task 1.2	New	07/23/2013 01:30	07/29/2013 01:55	staff3
3	Task 2.1	New	07/23/2013 01:32	07/27/2013 01:38	KieuDTD

At the bottom left of the table, there are navigation buttons: 'Prev', '1', 'Next'. On the right side of the table, there's a 'REFRESH' button. Above the table, there's a search bar with the number '1' in it, and radio buttons for 'Name' and 'Status' with 'Name' being selected. A 'SEARCH' button is also part of the search bar.

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps	1. Go to [Home page]. 2. Login with roles of Manager 3. Click on [Request box]. 4. Click on [Task] tab					
	Components						
TC_MTF_001			- Show form [Search] above table Task List.		Pass	Jul-23-2013	
TC_MTF_002		5. Fill value want to search. 6. Choose [Name] to search belong to [Task Name], [Status]: to search belong to status of tasks. 7. Click on [Search] button.	- Display Task List belong to the result.		Pass	Jul-23-2013	

4.22. Technical Manager - [Manage Task] View Task Detail

The screenshot shows the 'TASK DETAIL' window from the MRS Solution application. At the top, there's a navigation bar with links like HOME, ABOUT, EQUIPMENT, FAQ, STATISTIC, ACCOUNT, and INVOICE. A 'Hello manager' message is visible on the right. The main window has a title 'TASK DETAIL' and a sub-instruction: 'Please analyze the problem carefully. In any misunderstand, please reply on comment box to get more detail from manager!'. It contains several input fields and controls:

- Task Name:** Task 1.1
- Staff Relate:** staff1 (dropdown), Reassign to
- Request Relate:** Request 1 (dropdown)
- Start:** 23/07/2013 01:30 SA
- Finish:** 29/07/2013 06:30 SA
- Spend Hours:** 0
- Status:** New
- Description:** A rich text editor with a toolbar containing icons for bold, italic, underline, etc. It also includes a 'RECOMMEND STAFF' button and a checked 'Active' checkbox.
- Buttons at the bottom:** CLOSE, UPDATE, FINISH.

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps	1. Go to [Home page]. 2. Login with roles of Manager 3. Click on [Request box]. 4. Click on [Task] tab. 5. Click on a record of table Task List.					
Components							
TC_MTD_001			- Show [Task Detail] form on popup with details information about selected task.		Pass	Jul-23-2013	

4.23. Technical Manager - [Manage Task] Update Task

The screenshot shows the 'TASK DETAIL' window. At the top, there's a message: "Please analyze the problem carefully. In any misunderstand, please reply on comment box to get more detail from manager!" Below this, the 'Task Name' field contains "Task 1.1". The 'Staff Relate' dropdown shows "staff1" and has a 'Reassign to' checkbox. The 'Request Relate' dropdown shows "Request 1". The 'Start' date is set to "23/07/2013 01:30 SA" and the 'Finish' date is "29/07/2013 06:30 SA". The 'Spend Hours' field is set to "0". A 'RECOMMEND STAFF' button is visible. The 'Description' section contains the text "asd" and features a rich text editor toolbar. At the bottom, there are three buttons: 'CLOSE', 'UPDATE' (which is highlighted with a red box), and 'FINISH'.

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps	<ol style="list-style-type: none"> 1. Go to [Home page]. 2. Login with roles of Manager 3. Click on [Request box]. 4. Click on [Task] tab 5. Click on a record of table Task List. 					
Components							
TC_MTU_001			- Show [Task Detail] form on popup with details information about selected task.		Pass	Jul-23-2013	
TC_MTU_002		<ol style="list-style-type: none"> 6. Fill in fields that want to update. 7. Click [Update] button 	- Updated task is updated to database		Pass	Jul-23-2013	

TC ₊ MTU_003		6. Fill in fields that want to update. 7. Click [Close] button	- Updated task is not updated to database - Close popup [Task Detail]		Pass	Jul-23-2013	
----------------------------	--	---	--	--	------	-------------	--

4.24. Technical Manager - [Manage Task] Push/Receive Notification

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps	1. Go to [Home page]. 2. Login with roles of Manager 3. Click on [Request box]. 4. Click on [Task] tab.					
	Components						
TC ₊ MTP_001		5. Add new task to Task List	- New notification is send to according staff.		Pass	Jul-23-2013	
TC ₊ MTP_002		5. Update a task to Task List	- New notification is send to according staff.		Pass	Jul-23-2013	

4.25. Technical Manager - [Manage Symptom] Create Symptom

4.26. Technical Manager - [Manage Symptom] View All Symptom List

4.27. Technical Manager - [Manage Symptom] View Symptom Detail

4.28. Technical Manager - [Manage Symptom] Update Symptom

4.29. Technical Manager - [Manage Equipment] Add Equipment

4.30. Technical Manager - [Manage Equipment] View All Equipment List

4.31. Technical Manager - [Manage Equipment] View Equipment Detail

4.32. Technical Manager - [Manage Equipment] Update Equipment

4.33. Technical Manager - [Statistics] Statistics by request type

4.34. Technical Manager - [Statistics] Statistics by top fixed model

4.35. Technical Manager - [Statistics] Statistics by symptom

4.36. Technical Manager - [Manage User] Add new Account

The screenshot shows a modal dialog titled "ADD NEW ACCOUNT". It contains the following fields:

- First name:
- Last name:
- User Name:
- Email:
- Gender: Male
- Phone number:
- Address:
- Role: Customer

At the bottom are two buttons: "CLOSE" and "ADD".

Below the dialog, the main application interface shows the following details:

- User: sherlockholmes
- Name: Hung Dang
- Gender: Male
- Email: hanhdi@gmail.com
- Phone: 1234567890
- Address: Gò Vấp, Hồ Chí Minh
- Role: Customer

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps	6. Go to [Home page]. 7. Login with roles of Manager 8. Click on [Request box]. 9. Click on [Account] tab 10. Click on [Add Account] button.					
	Components						
TC_MUC_001	View Add New Account form		- Show [Add New Account] form on popup with its components.		Pass	Jul-23-2013	
TC_MUC_002	Add a new account successfully	8. Fill all required fields. 9. Click [Add] button.	- New account is inserted to database. - An email is sent automatic to registered email of account.		Pass	Jul-23-2013	
TC_MUC_003	Add a new account with error	8. Fill in fields. 9. Keep some required fields blank. 10. Click [Add] button.	- Display error message on required fields or on [Error] tab. - New account is not inserted to database		Pass	Jul-23-2013	

TC_MUC_004	Add a new account. Username is automatic generated.	9. Fill all required information. 10. Fill [First Name] and [Last Name] same as a registered account 11. Click [Add] button.	<ul style="list-style-type: none"> - [Username] is automatic generated with added number behind. 		Pass	Jul-23-2013
TC_MUC_005	Add new account with role is [Staff]	6. Fill all required information 7. Select role [Staff] 8. Click on [StaffSkill] field to select skill. 9. Click [Add] button	<ul style="list-style-type: none"> - Display [StaffSkill] field for staff. - Display list of skill to select. - New account is inserted to database. 		Pass	Jul-23-2013
TC_MUC_006	View Add account form but do not add.	8. Fill in fields on [Add New Account] form. 9. Click [Close] button.	<ul style="list-style-type: none"> - New account is not inserted to database. - Close [Add New Account] popup. 		Pass	Jul-23-2013

4.37. Technical Manager - [Manage User] View List of Accounts

The screenshot shows the MRS solution application interface. At the top, there is a navigation bar with links for HOME, ABOUT, EQUIPMENT, FAQ, STATISTIC, ACCOUNT (which is highlighted in orange), and INVOICE. On the right side of the header, there is a user session indicator showing 'Hello manager' with a news count of 0, and a LOGOUT button.

The main content area is titled 'ACCOUNT LIST'. It features a table with columns: NO., USER NAME, FULL NAME, GENDER, EMAIL, PHONE NUMBER, ADDRESS, ROLE, and ACTIVATION. The table contains 10 rows of account data. Below the table are navigation buttons for Prev, Next, and a Refresh button.

NO.	USER NAME	FULL NAME	GENDER	EMAIL	PHONE NUMBER	ADDRESS	ROLE	ACTIVATION
1	customer	Johny English	Male	abcd1234@gmail.com	0964123456	Gò Vấp, HCM	Customer	<input checked="" type="checkbox"/>
2	HungDD	Hung Dang Duong	Male	hbd@da.1sda	123343342	dåasd	Staff	<input type="checkbox"/>
3	KieuDTD	Kieu Duong Thi Diem	Female	123@fsdf.23	1243423423	åadasd	Staff	<input checked="" type="checkbox"/>
4	KieuDTD2	Kieu D T D 2	Female	abd@fdf.f	1232432	ådd	Staff	<input checked="" type="checkbox"/>
5	manager	Technical Manager	Male	hanbdi1@gmail.com	0964987678	Gò Vấp, HCM	Manager	<input checked="" type="checkbox"/>
6	manager2	Hung Dang	Male	hanbdi@gmail.com5	1234567890	Go Vap, Ho Chi Minh	Manager	<input checked="" type="checkbox"/>
7	sherlockholmes	Hung Dang	Male	hanbdi@gmail.com	1234567890	Go Vap, Ho Chi Minh	Customer	<input checked="" type="checkbox"/>
8	staff1	Hung Dang	Male	hanbdi3@gmail.com	1234567890	Go Vap, Ho Chi Minh	Staff	<input checked="" type="checkbox"/>
9	staff2	Hung Dang	Male	hanbdi4@gmail.com	1234567890	Go Vap, Ho Chi Minh	Staff	<input checked="" type="checkbox"/>
10	staff3	Hung Dang	Male	hanbdi2@gmail.com	1234567890	Go Vap, Ho Chi Minh	Staff	<input checked="" type="checkbox"/>

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
	Steps <ol style="list-style-type: none"> 5. Go to [Home page]. 6. Login with roles of Manager 7. Click on [Request box]. 8. Click on [Account] tab. 						
	Components						
TC_MUV_001			- Show [Account List] table with information contain and show on paging of 10 account per page.		Pass	Jul-23-2013	

4.38. Technical Manager - [Manage User] Update Account Information

The screenshot shows a modal dialog titled "ACCOUNT DETAIL - UPDATE". Inside the dialog, there are input fields for "User Name" (KieuDTD), "Full name" (full name), a checked "Active" checkbox, a dropdown "Role" set to "Staff", and a list of "Staff Skill" items: CPU, HDD, and Keyboard-Mouse. At the bottom of the dialog are two buttons: "CLOSE" and "UPDATE", with "UPDATE" being highlighted by a red border. Below the dialog, a table displays account information for three users: manager, manager2, and manager3.

ID	Manager	Role	Gender	Email	Phone	Address	City
5	manager	Technical Manager	Male	hanbd1@gmail.com	0964507670	Gò Vấp, HCMC	Manager
6	manager2	Hung Dang	Male	hanbd1@gmail.com5	1234567890	Gò Vấp, Ho Chi Minh	Manager

ID	Description	Test Case Procedure	Expected Output	Dependence	Result	Test date	Note
Steps							
8. Go to [Home page]. 9. Login with roles of Manager 10. Click on [Request box]. 11. Click on [Account] tab 12. Click on a record of [Account List] table.							
Components							
TC_MUU_001	View detail of an account		- Show [Account Detail] form on popup with details information about selected account.		Pass	Jul-23-2013	
TC_MUU_002	Update new information for an account	13. Fill in fields that want to update. 14. Click [Update] button	- Selected account is updated to database with new information. - Close popup [Account Detail]		Pass	Jul-23-2013	
TC_MUU_003	View Account detail but do not update.	6. Fill in fields that want to update. 7. Click [Close] button	- Selected account is not updated to database - Close popup [Account Detail]		Pass	Jul-23-2013	

5. Checklists

5.1. Checklist of Validation

N/A

5.2. Submission Checklist

N/A

6. Other material

N/A

VI. Installation Guide

1. Setting up the environment at server side

The following software must be installed into the server machine:

1.1. Hardware requirements

- Personal computers for developing with the minimum configuration: CPU Core 2 Duo 2.0GHz, 2GB of RAM, 120GB of hard disk, and internet.

1.2. Software requirements

- Operating system: Windows 7
- Web Server: Apache Tomcat 7.0.34
- MySQL Server 5.6: used to create and manage the database for web application

2. Deployment at server side

2.1. Prepare deployment package

- Extract the deployment package to a folder on the server. For example: *D:\Capstone Project*

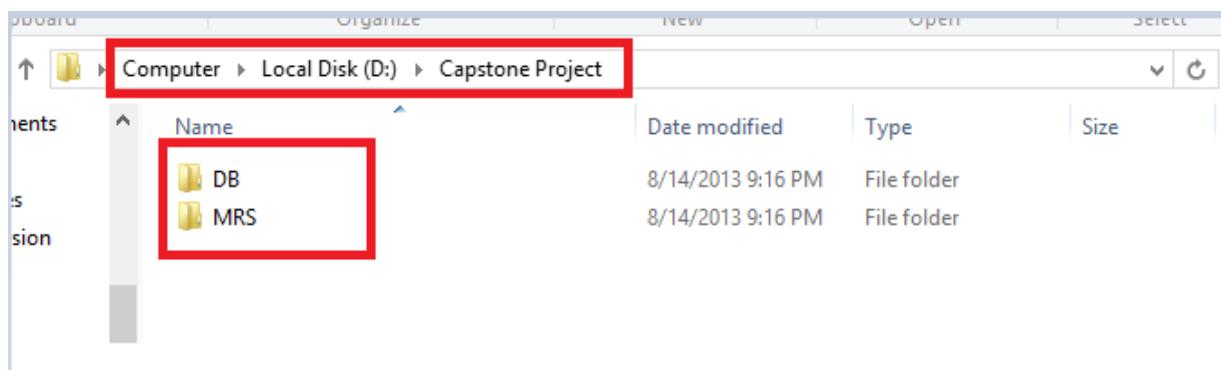


Figure 179 - Prepare Deployment Package

2.2. Deploy database

- Open DB folder.
- Using MySQL Workbench to import file “MRS_Script.sql”

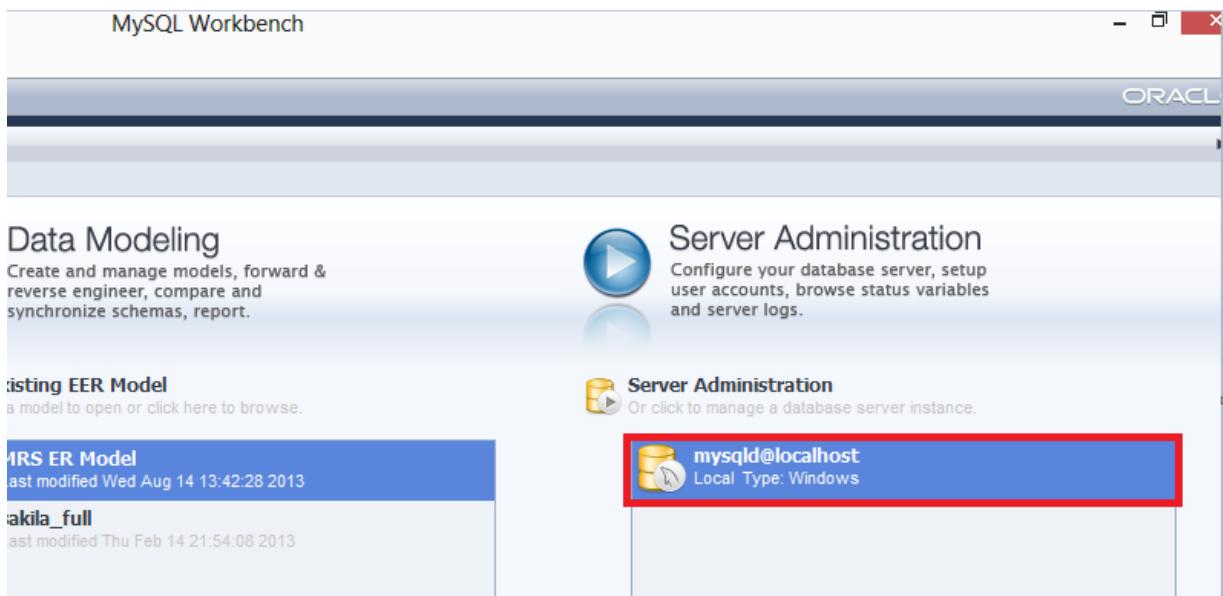


Figure 180-Deploy Database (1)

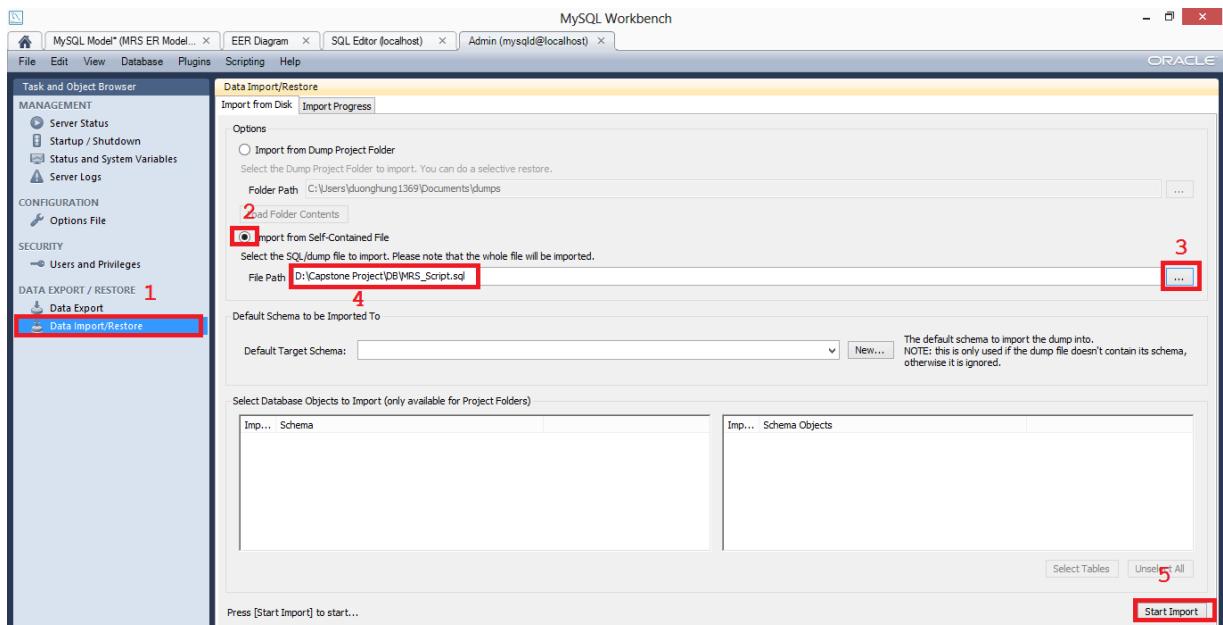


Figure 181-Deploy Database (2)

- A new database name mrsdb will be created with all table and sample data.

2.3. Deploy web application

- Copy **MRS_Capstone_v1.war** in D:\Capstone Project\MRS into folder **webapps** of Tomcat and then start server.
- Url to connect to MRS: http://localhost:8083/MRS_Capstone_v1 (check corresponding Tomcat port) to deploy **MRS_Capstone_v1.war**

2.4. Change configurations

2.4.1. Change database connection configuration

- After deploy **MRS_Capstone_v1.war** to **webapps**, browse to **./webapps/MRS_Capstone_v1/WEB-INF/classes** and open **hibernate.cfg.xml** to edit database connection.

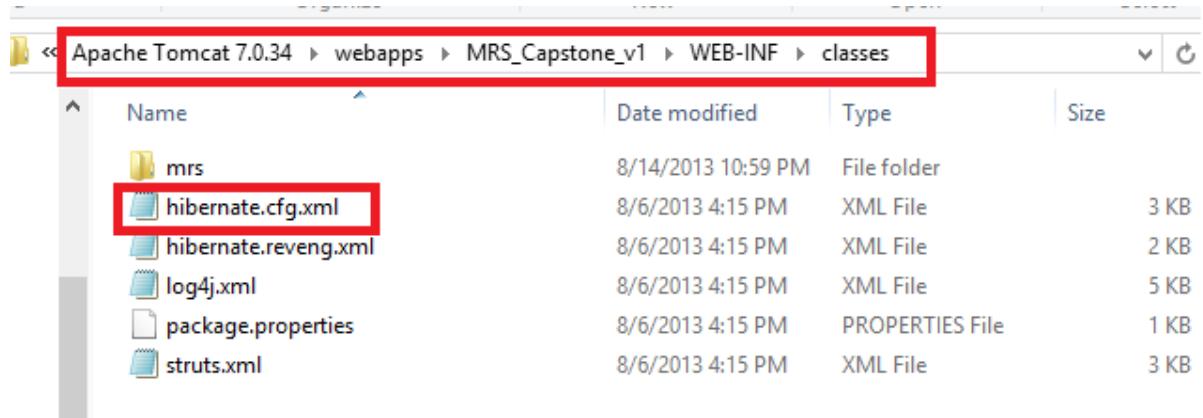


Figure 182 - Change database connection configuration (1)

- Change database url connection, username, password corresponding installed MySQL information.

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <!DOCTYPE hibernate-configuration PUBLIC "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
   http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">
3  <session-factory>
4      <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>
5      <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
6      <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/mrsdb?useUnicode=true&characterEncoding=UTF-8</property>
7      <property name="hibernate.connection.username">root</property>
8      <property name="hibernate.connection.password">123456</property>
9
10 <property name="hibernate.show_sql">true</property>

```

Figure 183 - Change database connection configuration (2)

3. Setting up the environment at client side

The system is accessed via web browsers. The client machine should be installed with one of the following browsers:

- Google Chrome (latest version is recommended)
- Mozilla Firefox (latest version is recommended)

4. User's Guide

4.1. Guest's Guide

4.1.1. Register

To able to use MRS website's functions, guests should register to be "Customer".

To register an account, guest can do by following steps:

- Click **REGISTER** button in header of page.

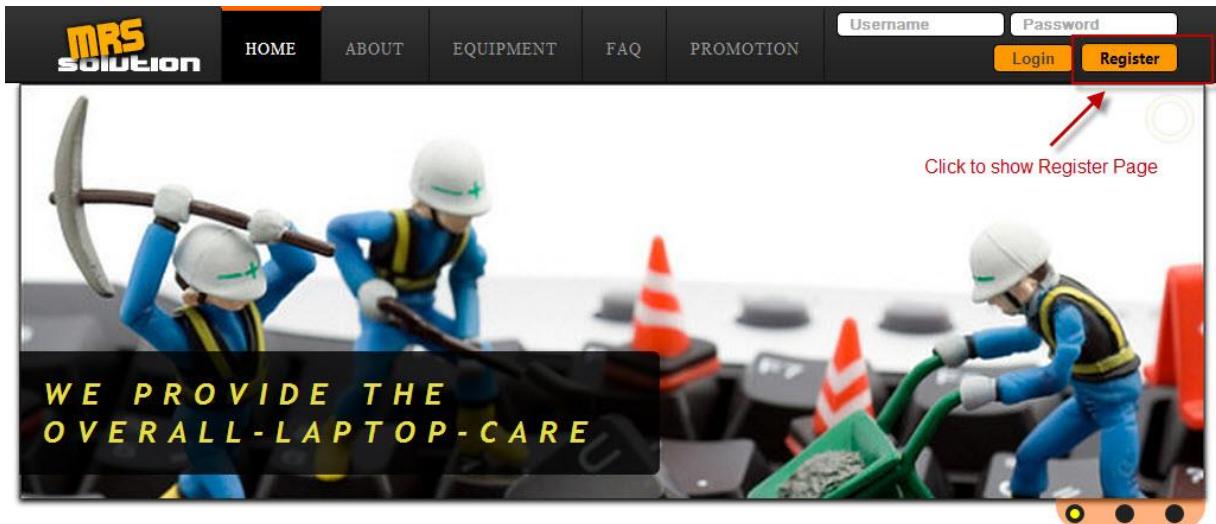


Figure 184 - Click Register button

- Redirect to [Register Page] with Register form:

register

FIRST NAME :

LAST NAME :

USER NAME :

PASSWORD :

RETYPE
PASSWORD :

EMAIL :

RETYPE EMAIL :

PHONE NUMBER
(FORMAT:
+849XXXXXXXX) :

ADDRESS :

GENDER : Male

totkien 52-58

Nhập hai từ reCAPTCHA™ stop spam. read books.

why register?

← Form Register

already have an account? [sign in](#)

username
password

Click to Register

Figure 185 - Register Page

- “**First Name**”: is the given name of user. It can be following Unicode format and is *mandatory* information.
- “**Last Name**”: is the given name of user. It can be following Unicode format.
- “**Username**”: is the username of account which is used to login the system. Username must be valid (*not in used*) in system and has at least 6 character and at most 20 characters. Username is *mandatory* information.
- “**Password**”: is the password of account which is used to login the system. Password has at least 6 characters and at most 20 characters. Password is *mandatory* information.
- “**Retype Password**”: is the re-typed password. It must be corrected with password. Re-typed password is *mandatory* information.

- “**Email**”: is email of account which is used to send the information from system to user. Email must be valid (*not in used*) in system and be following email format, example: abc@yahoo.com,...
- “**Retype Email**”: is the re-type email. It must be corrected with email. Re-typed email is *mandatory* information.
- “**Phone number**”: is the phone number of user. Phone number is *mandatory* information.
- “**Address**”: is address f user. Phone number is *mandatory* information.
- “**Gender**”: is the gender of user. User must select one in three selections: “Male”, “Female”, “Other”.
- “**Captcha code**”: Capcha code is *mandatory*.
 - Click **[Create Account]** button to finish registration and redirect to website Home page.

4.1.2. View FAQ

This function can be used be all visitors. Visitor has to go to [FAQ page] to view all FAQs in the system.

To view FAQ list, visitor can do by following steps:

- Click **FAQ** tab on Home page to redirect to [FAQ Page].

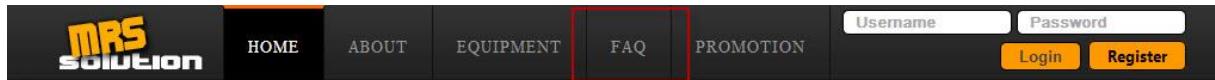


Figure 186 - Click FAQ tab

- On the Left menu have Model list, visitor can select to view FAQ list according each Model.

The screenshot shows a FAQ page with a large question mark icon at the top right. Below it, the text "F.A.Q.s Got a question?" is displayed. A message says, "If you got a question, look no further. We post the most common questions. If you have any further questions, please feel free to contact us." To the left, a sidebar lists categories: Acer, Asus, Compaq-HP, Dell, IBM-Lenovo, Samsung, Toshiba, USB 3G, Creative, Sony Walkman, Desktop Hard Drive, Portable Hard Drive, and Sony. The main content area shows three expanded FAQ entries: 1. *Ổ đĩa chạy chậm* (Slow disk), which contains text about Disk Defragmenter; 2. *Dumping RAM*; and 3. *Máy quá nóng*. Red arrows point to the first entry's title, the content area, and the "Click to view FAQ belong to Model" link below the sidebar.

Figure 187 - FAQ Page

- Click on title of FAQ to expand content.

4.1.3. View Equipment

All guests can use this function.

To view Equipment, guest can do by following steps:

- Click **Equipment** tab on Home page to redirect to [Equipment Page].

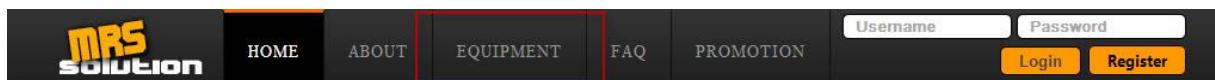
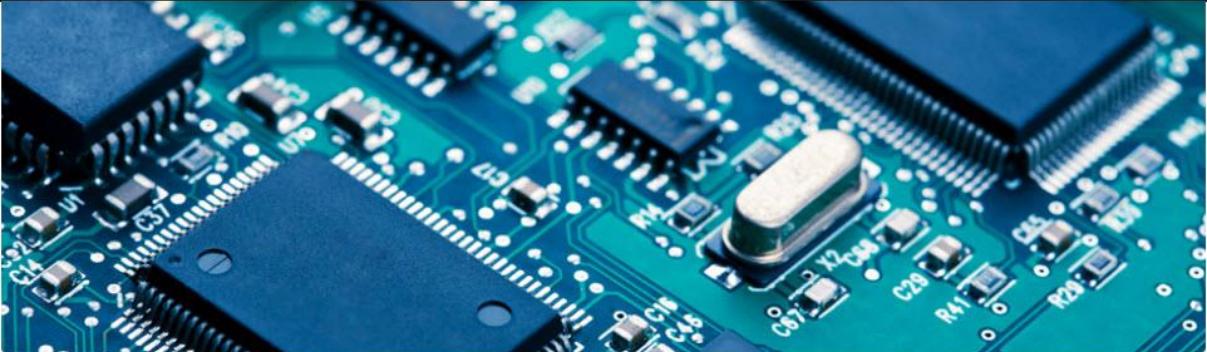


Figure 188 - Click Equipment tab

- On the Left menu have Category list, visitor can select to view Equipment list according each Category.



ALL PRODUCTS

- CPU
- RAM
- HDD
- VGA
- MAINBOARD
- CASE
- KEYBOARD-MOUSE
- ODD
- SPEAKER
- HEADPHONE
- PIN
- LCD
- REMOVABLE HDD

all products

SEARCH NAME

			
MB INTEL BOX DB85FL 1000 (VND) Instock : Sd	ZOTAC H61 M-ATX 930 (VND) Instock : Sd	FOXCONN H61MXE 950 (VND) Instock : Sd	ASROCK H61M-VS 950 (VND) Instock : Sd
			
GIGABYTE GA-B75M-D2V (REV. 1.0) 1490 (VND) Instock : Sd	GIGABYTE GA-A55M-DS2 1480 (VND) Instock : Sd	INTEL BLKDH61WWB3 1290 (VND) Instock : Sd	GIGABYTE GA-P61A-D3 1600 (VND) Instock : Sd
			
ASUS P8B75-M 1733 (VND) Instock : Sd	INTEL BLKDH77KC (ATX) 2280 (VND) Instock : Sd		

Click to view equipment by Category

Prev 1 2 3 4 5 ... 8 9 Next

Figure 189 - Equipment Page

- Click on image link to view detail



Figure 190 - Equipment Detail

4.1.4. Search Equipment

All guests can use this function. Guest has to go to [Equipment page] to search equipment.

To search equipment, guest can do by following steps:

- Click **Equipment** tab on Home page to redirect to [Equipment Page].
- Input search value to textbox and click **Search** button
- The result of searching equipment is showed at below



Figure 191 - Search Equipment

4.1.5. View Promotion Event

4.1.6. Contact via mail

4.2. Registered User's Guide

4.2.1. Login

Registered User should login the system to use specific functions of system.

To login, Registered User can do by following steps:

- Input username, password and click **LOGIN** button in header of page.



Figure 192 - Login at Home Page

- Input username, password and click **LOGIN** button in Login form at [Register Page].

The screenshot shows a registration form titled 'register'. It contains fields for First Name, Last Name, User Name, Password, Retype Password, Email, Retype Email, and Phone Number. To the right of the form is a button labeled 'why register?'. Below the form is a login section with fields for 'username' and 'password', and a 'LOGIN' button.

Figure 193 - Login at Register Page

- After login successful, header of page will display username, notification and logout button.



Figure 194 - Login success

4.2.2. Update profile

Registered User uses this function.

To update profile, they can do by following steps:

- Login the system
- Click username link on the header of page

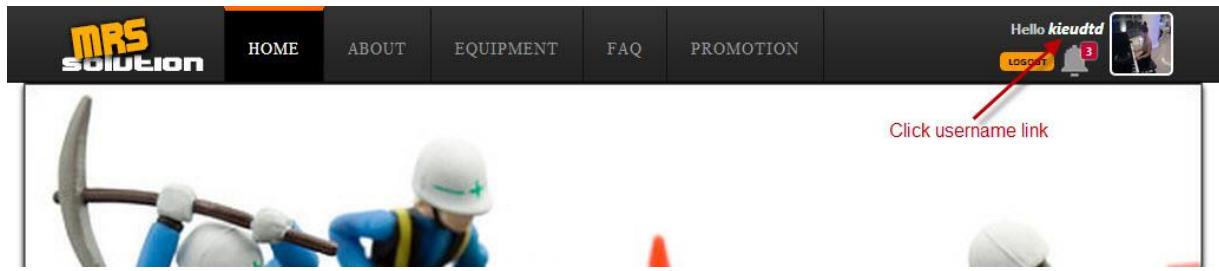


Figure 195 - Click Profile

- Display [Profile Page] on screen with Basic information that user can update.

A screenshot of the 'Basic Information' update profile page. The page has a sidebar on the left with 'PERSONAL' sections for 'Basic' and 'Change Password', and 'SERVICES' sections for 'Tracking Request'. The main area is titled 'Basic Information' and contains fields for First Name ('Kiều'), Last Name ('Dương Thị Diễm'), Email ('kieudtd60471@fpt.edu.vn'), Gender ('Male' dropdown), Phone number ('+840908783553'), and Address ('Quận 12, HCM'). A red box highlights the 'Address' field and the 'UPDATE' button below it. A red arrow points from the text 'Edit information' to the 'Address' field. Another red arrow points from the text 'Click' to the 'UPDATE' button.

Figure 196 - Update Profile

- “**Gender**”: is the gender of user. User must select one in three selections: “Male”, “Female” and “Other”.
- “**Phone number**”: is the given name of user. It can be following Unicode format.
- “**Address**”: is the middle name of user. It can be following Unicode format
- Click **Update** to finish Update profile action

4.2.3. Change Password

Registered User uses this function.

To change password, they can do by following steps:

- Login the system
- Click on username link in the header of page, [Profile Page] will be shown on screen.
- Click on Change Password link on navigate

Basic Information

First Name: Kiều
Last Name: Dương Thị Diễm
Email: kieuidd60471@fpt.edu.vn
Gender: Male
Phone number: +840908783553
Address: Quận 12, HCM

PERSONAL

- Basic
- Change Password**

SERVICES

- Tracking Request

UPDATE

Figure 197 - Link Change Password

- Fill all fields:
 - “**Old Password**” is current password. It must be correct with the password of user
 - “**New Password**” is new password. It must have at least 6 characters and at most 20 characters.
 - “**Confirm Password**” is re-typed password. It must be corrected with new password

Change Password

Old Password:
New Password:
Confirm Password:

PERSONAL

- Basic
- Change Password

SERVICES

- Tracking Request

CHANGE

Edit information

Click

Figure 198 - Change Password

- Click **Change** to finish Changing Password and redirect to Edit account Information page

4.2.4. Log Out

Registered User uses this function. They must login the system.

To logout the system, they clicks **Logout** button on the header of page. The system redirect to Home Page



Figure 199- Click Logout link



Figure 200 - Redirect to Home Page

4.3. Customer's Guide

4.3.1. View request list

Customer of system uses this function to track their entire request.

To view all request, customer can do by following steps:

- Login to the system.
- Click on **Request** box on [Home Page]

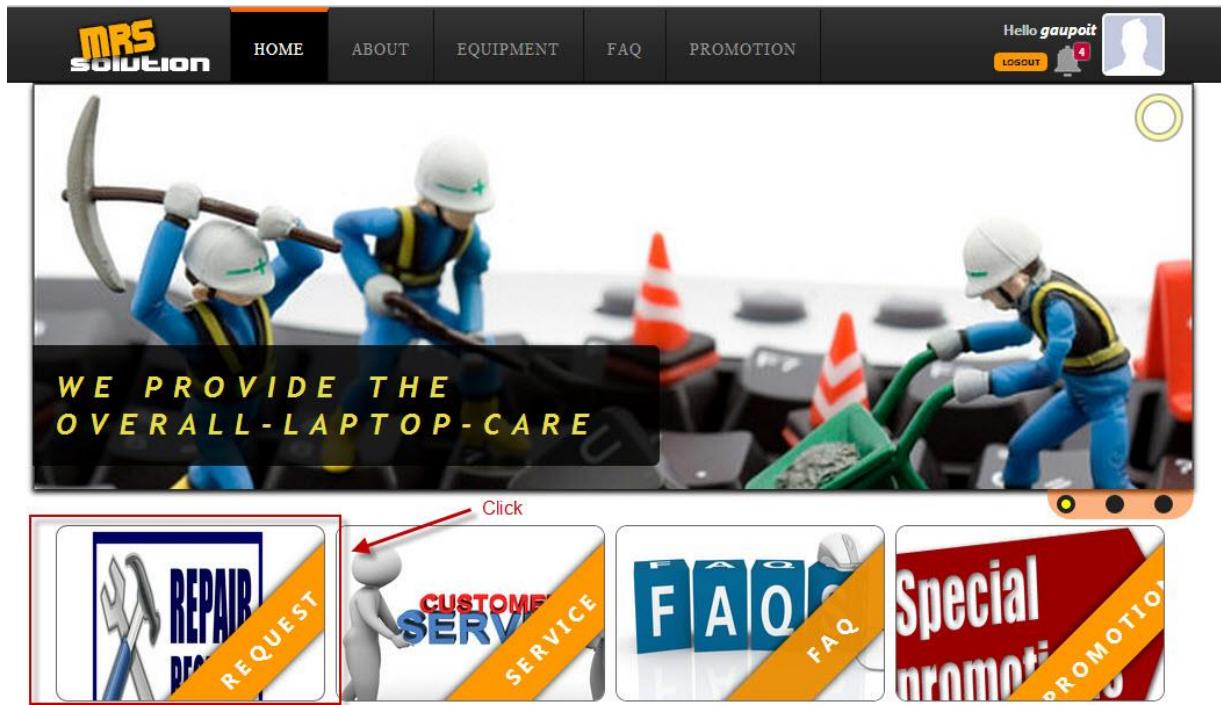


Figure 201 - Click Request box

- [Customer Request] Page will be shown on screen with table Request list

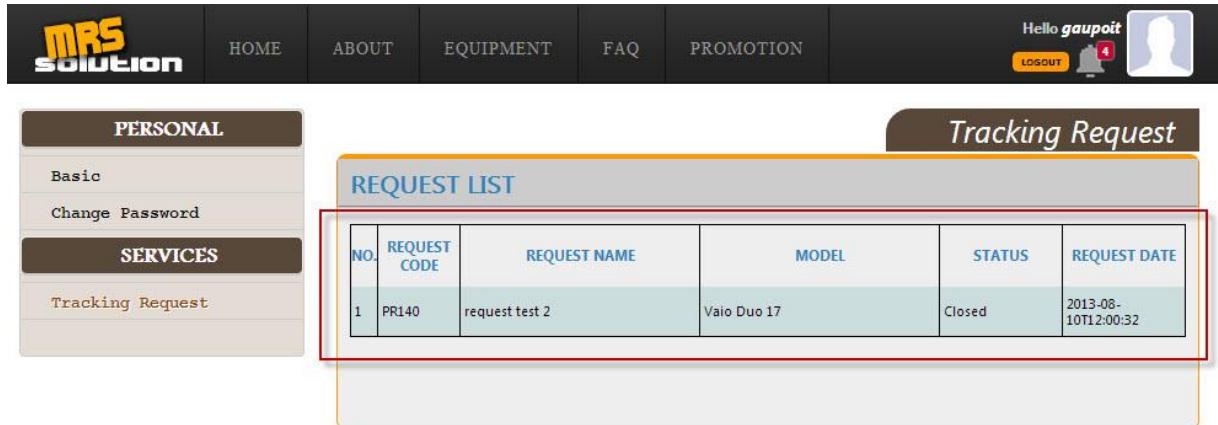


Figure 202 - Customer Tracking Request Page

- Click on a row of table Request List to view detail of this request

The screenshot shows the "REQUEST DETAIL" page. At the top, it says "The following information is the detail of this request". Below this, there are input fields for Request Name (requesttest2), Request Type (Repair), Model (Vaio Duo 17), and Status (Closed). There is also a text area for PROBLEM DESCRIBE containing the text "bi hỏng rat nhiều thứ". Under EQUIPMENT, there is a table with columns Name and Price. It lists two items: ASUS EN210 SILENT/DI/1GD3/V2(LP) at 796 and Mouse Pad Zadez at 21. Under INVOICE, there is a table with columns Equipment, Quantity, and Unit Price. It shows the same two items with quantities of 1 and unit prices of 796 and 21 respectively. At the bottom left, there is a "CLOSE" button.

Name	Price
ASUS EN210 SILENT/DI/1GD3/V2(LP)	796
Mouse Pad Zadez	21

Equipment	Quantity	Unit Price
ASUS EN210 SILENT/DI/1GD3/V2(LP)	1	796
Mouse Pad Zadez	1	21

Figure 203 - Request Detail

4.3.2. Update request

Customer of system uses this function to update their request.

To update their request, customer can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Customer Request] Page will be shown on screen with table Request list
- Click on a row of table Request List to view detail of this request.
- In case of status of request is **Pending**, form detail has **Customer Confirm** field to customer can edit.

REQUEST DETAIL

The following information is the detail of this request

Request Name:	request 31232	Request Type:	Repair
Model:	NB Acer Aspire AO756 - 87	Status:	Pending

PROBLEM DESCRIBE
asdasd

EQUIPMENT :

Name	Price
MB Intel Box DB85FL	1000

Customer Confirmation

Rich text editor toolbar with various icons for text formatting, images, and tables.

CANCEL **CONFIRM** ← Click

- Click **Cancel** to cancel request or **Confirm** to confirm approval request.

4.4. Staff's Guide

4.4.1. View All Task List

Staff of system uses this function to track their entire task.

To view all task, staff can do by following steps:

- Login to the system
- Click on **Task** box on [Home Page]



- [Staff Tracking Task] Page will be shown on screen with table Task list

NO.	TASK NAME	STATUS	START DATE	FINISH DATE	ACTIVATION
1	task moi	New	08/16/2013 00:08	08/20/2013 00:08	<input checked="" type="checkbox"/>

- Staff can filter their task by status of task.

4.4.2. Update Task

Staff of system uses this function to update their task.

To update their task, staff can do by following steps:

- Login to the system
- Click on **Task** box on [Home Page]
- [Staff Tracking Task] Page will be shown on screen with table Task list
- Click on a row of table Task List to view detail of this task

TASK					
NEW TASK		IN-PROGRESS TASK		FINISHED TASK	
TASK LIST					
Display 10 of 8 total		Click		Name	FILTER
NO.	TASK NAME	STATUS	START DATE	FINISH DATE	ACTIVATION
1	task moi	New	08/16/2013 00:08	08/20/2013 00:08	<input checked="" type="checkbox"/>
2	task for cancel	Finished	08/15/2013 14:18	08/16/2013 14:18	<input checked="" type="checkbox"/>
3	task cho Thanh tm	Missed Deadline	08/10/2013 11:05	08/13/2013 11:05	<input checked="" type="checkbox"/>
4	dffffg	Missed Deadline	08/11/2013 12:59	08/12/2013 12:59	<input checked="" type="checkbox"/>
5	task de test lai	Finished	08/10/2013 16:14	08/11/2013 16:14	<input checked="" type="checkbox"/>
6	task for request miss dealine	Missed Deadline	08/11/2013 12:57	08/11/2013 12:57	<input checked="" type="checkbox"/>
7	taskasds	Missed Deadline	08/10/2013 12:27	08/10/2013 22:27	<input checked="" type="checkbox"/>
8	task tre deadline	Missed Deadline	08/10/2013 12:25	08/10/2013 13:25	<input checked="" type="checkbox"/>

Prev 1 Next REFRESH

- In case of status of task is **New**, Detail form only showing **Start** button. Staff clicks on **Start** button to start their task.

TASK DETAIL

Please analyze the problem carefully. In any misunderstand, please reply on comment box to get more detail from manager.

BASIC INFORMATION

Task Name: task moi

Request Relate: TASK MOI

Product Relate: NB ACER ASPIRE A0756 - 877BCRR (001) (ĐỎ)

Current Status: New

Start Date: 16/08/2013 12:08 SA

Due Date: 20/08/2013 12:08 SA

TASK DESCRIBE

Task Content:

THAY RAM

CLOSE START

- In case of status of task is **In Progress**, Detail form only showing **Finish** button. Staff click on **Finish** button to finish their task

The screenshot shows the 'TASK DETAIL' window. At the top, it says 'Task Name: Task 1.1' and 'Request Relate: Request 1'. Below that, 'Current Status: In Progress' is displayed. Under 'Task Describe', there is a rich text editor toolbar and a text area containing 'Mã HTML'. On the right, a 'TOTAL PRICE LIST' table is shown:

EQUIPMENT NAME	PRICE	QUANTITY	TOTAL
MB Intel Box DB85FL	1000	1	1000
SUM TOTAL		1	1000

At the bottom of the window are 'CLOSE' and 'FINISH' buttons.

- In case of status of task is **Finished**, Detail form only showing **Close** button.

The screenshot shows the 'TASK DETAIL' window. The 'Basic Information' section includes 'Task Name: task for cancel', 'Request Relate: REQUEST FOR CANCEL', 'Product Relate: NB ACER ASPIRE AO756 - 877BCRR (001) (ĐÓ)', and 'Current Status: Finished'. The 'Task Describe' section contains 'Task Content: SAS'. On the right, a 'TOTAL PRICE LIST' table is shown:

EQUIPMENT NAME	PRICE	QUANTITY	TOTAL
SUM TOTAL		0	0

At the bottom of the window is a single 'CLOSE' button.

4.5. Technical manager's Guide

4.5.1. Manage Request

4.5.1.1. View Request list

Technical manager of system uses this function.

To view all request lists, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]

- [Manage Repair Maintenance] Page will be shown on screen with table Request list

The screenshot shows a web-based application interface titled "REQUEST LIST". At the top, there is a navigation bar with tabs: REQUEST, TASK, SYMPTOM, PRODUCT, EQUIPMENT, FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. The REQUEST tab is highlighted. Below the navigation bar, there is a search bar labeled "Display" with a dropdown set to "10" and a message "of 8 total.". To the right of the search bar is a "FILTER" button. The main area contains a table with the following columns: NO., CODE, REQUEST NAME, STATUS, CREATED DATE, ESTIMATE FINISH DATE, and ACTIVE. The table lists 8 repair requests, each with a unique ID (PR153 to PR150) and a brief description. The last column, ACTIVE, contains checkboxes. At the bottom of the table are buttons for "Prev", "1", "Next", and "REFRESH".

NO.	CODE	REQUEST NAME	STATUS	CREATED DATE	ESTIMATE FINISH DATE	ACTIVE
1	PR153	thay ổ cứng bị hư	Closed	07/24/2013 07:59	07/27/2013 07:47	<input checked="" type="checkbox"/>
2	PR152	Cài lại firmware 3G Fast connect cho anh Diệp Trần Anh Tuấn	Missed Deadline	07/24/2013 06:02	07/26/2013 08:00	<input checked="" type="checkbox"/>
3	PR149	Sửa ổ cứng Seagate 3TB cho anh Nguyễn Phước Thịnh	Closed	07/05/2013 11:52	07/07/2013 11:00	<input checked="" type="checkbox"/>
4	PR148	Sửa USB 3G cho anh Nguyễn Phước Thịnh	Closed	07/04/2013 00:17	07/05/2013 17:00	<input checked="" type="checkbox"/>
5	PR147	Thay RAM cho chị Nguyễn Phước Bộ Trần	Closed	07/03/2013 15:19	07/03/2013 17:00	<input checked="" type="checkbox"/>
6	PR146	Sửa màn hình bị sọc cho Đăng Dương Hùng	Closed	07/02/2013 22:08	07/04/2013 10:06	<input checked="" type="checkbox"/>
7	PR151	Vệ sinh laptop cho anh Diệp Trần Anh Tuấn	Finished	07/02/2013 05:13	07/04/2013 05:10	<input checked="" type="checkbox"/>
8	PR150	Sửa bàn phím cho anh Đăng Dương Hùng	Closed	06/08/2013 18:36	06/11/2013 08:00	<input checked="" type="checkbox"/>

Figure 204-Manager Request List

- Technical manager can filter request by name, status, product type or code of request.

This screenshot is similar to Figure 204, showing the "REQUEST LIST" page. A dropdown menu is open over the "FILTER" button, displaying options: Name, Code, Status, and Type. The "Name" option is currently selected. The rest of the page structure is identical to Figure 204, including the table of repair requests and navigation buttons at the bottom.

NO.	CODE	REQUEST NAME	STATUS	CREATED DATE	ESTIMATE FINISH DATE	ACTIVE
1	PR153	thay ổ cứng bị hư	Closed	07/24/2013 07:59	07/27/2013 07:47	<input checked="" type="checkbox"/>
2	PR152	Cài lại firmware 3G Fast connect cho anh Diệp Trần Anh Tuấn	Missed Deadline	07/24/2013 06:02	07/26/2013 08:00	<input checked="" type="checkbox"/>
3	PR149	Sửa ổ cứng Seagate 3TB cho anh Nguyễn Phước Thịnh	Closed	07/05/2013 11:52	07/07/2013 11:00	<input checked="" type="checkbox"/>
4	PR148	Sửa USB 3G cho anh Nguyễn Phước Thịnh	Closed	07/04/2013 00:17	07/05/2013 17:00	<input checked="" type="checkbox"/>
5	PR147	Thay RAM cho chị Nguyễn Phước Bộ Trần	Closed	07/03/2013 15:19	07/03/2013 17:00	<input checked="" type="checkbox"/>
6	PR146	Sửa màn hình bị sọc cho Đăng Dương Hùng	Closed	07/02/2013 22:08	07/04/2013 10:06	<input checked="" type="checkbox"/>
7	PR151	Vệ sinh laptop cho anh Diệp Trần Anh Tuấn	Finished	07/02/2013 05:13	07/04/2013 05:10	<input checked="" type="checkbox"/>
8	PR150	Sửa bàn phím cho anh Đăng Dương Hùng	Closed	06/08/2013 18:36	06/11/2013 08:00	<input checked="" type="checkbox"/>

4.5.1.2. Add Request

Technical manager of system uses this function.

To add new request to Request list, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]



- [Manage Repair Maintenance] Page will be shown on screen.
- Click on **Add Request** button to show Add request form

The screenshot shows a form titled 'ADD REQUEST - PRODUCT PROBLEM'. It includes sections for 'CUSTOMER RELATED' (Customer: duonghung1269, Fullname: HUNG DUONG, Phone: +84964987678, Address: PHAN HUY ICH, HO CHI MINH), 'PRODUCT INFORMATION' (Category: Tablet, Prototype: Apple, Model: iPad 4), and 'BASIC INFORMATION' (Request Name: [empty], Estimate Finish: 08/16/2013 03:13 AM, Request Type: Repair). To the right, there is a 'TOTAL PRICE LIST' table with columns EQUIPMENT NAME, PRICE, QUANTITY, and TOTAL, showing a sum total of 0. Buttons at the bottom include 'CLOSE' and 'ADD'.

Figure 205-Manager Add Request

4.5.1.3. Update Request

Technical manager of system uses this function.

To update a request, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen with table [Request List].
- Click on a row of table [Request List] to view Request detail form.

4	PR148	Sửa USB 3G cho anh Nguyễn Phước Thịnh	Closed	07/04/2013 00:17	07/05/2013 17:00	<input checked="" type="checkbox"/>
---	-------	---------------------------------------	--------	------------------	------------------	-------------------------------------

- Click **Confirm Request** button to confirm this request. Click **Cancel Request** to cancel this request.

4.5.2. Manage Task

4.5.2.1. View Task list

Technical manager of system uses this function.

To view all task lists, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]

- [Manage Repair Maintenance] Page will be shown
- Click **Task** tab in [Manage Repair Maintenance] or **Create Task** button in [Detail request] form
- [Manage Task] Page will be shown on screen with table [Task List]

NO.	TASK NAME	STATUS	CREATED DATE	FINISH DATE	ASSIGNED TO
1	task for cancel 2	In Progress	08/15/2013 14:31	08/16/2013 14:31	kieudtd
2	task for cancel	Finished	08/15/2013 14:18	08/16/2013 14:18	thanhtm
3	task request test 2 2	Finished	08/13/2013 12:03	08/15/2013 19:03	kieudtd
4	oooooooooooooooooooo	In Progress	08/13/2013 13:23	08/14/2013 13:23	kieudtd
5	assdf	In Progress	08/13/2013 13:19	08/13/2013 13:19	kieudtd
6	task request test 2 1	Finished	08/10/2013 12:02	08/13/2013 12:02	kieudtd
7	task cho Thanh tm	Finished	08/10/2013 11:05	08/13/2013 11:05	thanhtm
8	dfggg	In Progress	08/11/2013 12:59	08/12/2013 12:59	thanhtm
9	task de test lai	Finished	08/10/2013 16:14	08/11/2013 16:14	thanhtm
10	task for request miss dealine	In Progress	08/11/2013 12:57	08/11/2013 12:57	thanhtm

REFRESH

- Technical manager can filter task by name, status, or request relate.

NO.	TASK NAME	STATUS	CREATED DATE	FINISH DATE	ASSIGNED TO
1	task for cancel 2	In Progress	08/15/2013 14:31	08/16/2013 14:31	kieudtd
2	task for cancel	Finished	08/15/2013 14:18	08/16/2013 14:18	thanhtm
3	task request test 2 2	Finished	08/13/2013 12:03	08/15/2013 19:03	kieudtd

4.5.2.2. Add Task

Technical manager of system uses this function.

To add new task to Request list, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **Task** tab in [Manage Repair Maintenance]
- [Manage Task] Page will be shown on screen with table [Task List]
- Click on **Add Task** button to show Add form

REQUEST	TASK	SYMPTOM	PRODUCT	EQUIPMENT	FAQ	PROMOTION	STATISTIC	ACCOUNT	INVOICE
TASK LIST									
ADD TASK									
Display	10	of 12 total.							
NO.	TASK NAME	STATUS	CREATED DATE	FINISH DATE	ASSIGNED TO				
1	task for cancel 2	In Progress	08/15/2013 14:31	08/16/2013 14:31	kieudtd				
2	task for cancel	Finished	08/15/2013 14:18	08/16/2013 14:18	thanhtm				

Fill all fields and click Add button:

ADD NEW TASK

Please describe the task!

REQUEST RELATED

Request: task moi

Created Date: 08/16/2013 00:05

Estimate Finish Date: 08/23/2013 00:05

Problem: ASS

STAFF RELATED

Staff Relate: thanhtm

RECOMMEND STAFF

TASK INFORMATION

Name: Click

CLOSE ADD

4.5.2.3. Update Task

Technical manager of system uses this function.

To update a task, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **Task** tab in [Manage Repair Maintenance]
- [Manage Task] Page will be shown on screen with table [Task List]
- Click on a row of table [Task List] to view Task detail form.

REQUEST	TASK	SYMPOTM	PRODUCT	EQUIPMENT	FAQ	PROMOTION	STATISTIC	ACCOUNT	INVOICE
TASK LIST									
ADD TASK									
Display 10 of 12 total.									
NO.	TASK NAME	STATUS	CREATED DATE	FINISH DATE	ASSIGNED TO				
1	task for cancel 2	In Progress	08/15/2013 14:31	08/16/2013 14:31	kieudtd				
2	task for cancel	Finished	08/15/2013 14:18	08/16/2013 14:18	thanhtm				
3	task request test 2.2	Finished	08/13/2013 12:03	08/15/2013 19:03	kieudtd				
4	oooooooooooooooooooo	In Progress	08/13/2013 13:23	08/14/2013 13:23	kieudtd				
5	assdf	In Progress	08/13/2013 13:19	08/13/2013 13:19	kieudtd				
6	task request test 2.1	Finished	08/10/2013 12:02	08/13/2013 12:02	kieudtd				
7	task cho Thanh tm	Finished	08/10/2013 11:05	08/13/2013 11:05	thanhtm				
8	dffffg	In Progress	08/11/2013 12:59	08/12/2013 12:59	thanhtm				
9	task de test lai	Finished	08/10/2013 16:14	08/11/2013 16:14	thanhtm				
10	task for request miss dealine	In Progress	08/11/2013 12:57	08/11/2013 12:57	thanhtm				
						REFRESH			
Prev		1	2	Next					

Fill data and click **Update** button

TASK DETAIL

Please analyze the problem carefully. In any misunderstand, please reply on comment box to get more detail from manager!

Staff: thanhtm	TOTAL USED EQUIPMENTS										
RECOMMEND STAFF	EQUIPMENT NAME PRICE QUANTITY										
TASK INFORMATION <table border="1"> <tr> <td>Name: task moi</td> <td>Start: 16/08/2013 12:08 SA</td> <td>Finish: 20/08/2013 12:08 SA</td> <td>Status: NEW</td> <td>Spend Hours: 0</td> </tr> <tr> <td colspan="5"> Description: <div style="border: 1px solid #ccc; padding: 5px; height: 100px;"> </div> </td> </tr> </table>		Name: task moi	Start: 16/08/2013 12:08 SA	Finish: 20/08/2013 12:08 SA	Status: NEW	Spend Hours: 0	Description: <div style="border: 1px solid #ccc; padding: 5px; height: 100px;"> </div>				
Name: task moi	Start: 16/08/2013 12:08 SA	Finish: 20/08/2013 12:08 SA	Status: NEW	Spend Hours: 0							
Description: <div style="border: 1px solid #ccc; padding: 5px; height: 100px;"> </div>											
<input type="button" value="CLOSE"/> <input type="button" value="UPDATE"/> <input type="button" value="FINISH"/>											

4.5.3. Manage Symptom

4.5.3.1. View Symptom list

Technical manager of system uses this function.

To view all symptom lists, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Symptom** tab in [Manage Repair Maintenance] or **Update Symptom** button in [Detail request] form

- [Manage Symptom] Page will be shown on screen with table [Symptom List]

NO.	SYMPOTM NAME	PRODUCT PROTOTYPE	ACTIVE
1	hong cam ưng màn hình	NB Acer E1 - 571 - 32342G50Mnks (006)	<input checked="" type="checkbox"/>
2	hu o cung	Toshiba	<input checked="" type="checkbox"/>
3	Sọc màn hình	DELL Vostro 3460 (34RH44)	<input checked="" type="checkbox"/>
4	Hồng cảm ứng màn hình	Vaio Duo 17	<input type="checkbox"/>
5	Máy không hoạt động	Creative	<input checked="" type="checkbox"/>
6	Màn hình bị điểm chết, bị lỗi	NB Lenovo G480 - 1145	<input checked="" type="checkbox"/>
7	Lỗi phần mềm không kết nối được với dây mạng	DELL Inspiron 15 3521 (HNP6M4-Black)	<input checked="" type="checkbox"/>
8	Không nạp được pin	NB Asus X402CA - WX042 (Trắng)	<input checked="" type="checkbox"/>
9	Hư loa	MP3 4GB Sony NWZ-E473	<input checked="" type="checkbox"/>
10	Dumping RAM	NB Lenovo G480 - 1145	<input checked="" type="checkbox"/>

Prev | 1 | 2 | Next | REFRESH

- Technical manager can filter symptom by name.

4.5.3.2. Add Symptom

Technical manager of system uses this function.

To add new symptom to Symptom list, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **Sympotm** tab in [Manage Repair Maintenance]
- [Manage Symptom] Page will be shown on screen with table [Symptom List]
- Click on **Add Symptom** button to show Add Symptom form

NO.	SYMPOTM NAME	PRODUCT PROTOTYPE	ACTIVE
1	hong cam ưng màn hình	NB Acer E1 - 571 - 32342G50Mnks (006)	<input checked="" type="checkbox"/>
2	hu o cung	Toshiba	<input checked="" type="checkbox"/>

Fill all fields and click **Add** button

4.5.3.3. Update Symptom

Technical manager of system uses this function.

To update a symptom, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **Symptom** tab in [Manage Repair Maintenance]
- [Manage Symptom] Page will be shown on screen with table [Symptom List]
- Click on a row of table [Symptom List] to view Symptom detail form.

NO.	SYMPOTM NAME	PRODUCT PROTOTYPE	ACTIVE
1	hong cam ưng màn hình	NB Acer E1 - 571 - 32342G50Mnks (006)	<input checked="" type="checkbox"/>
2	hu o cung	Toshiba	<input checked="" type="checkbox"/>
3	Sọc màn hình	DELL Vostro 3460 (34RH44)	<input checked="" type="checkbox"/>
4	Hồng cảm ứng màn hình	Vaio Duo 17	<input type="checkbox"/>
5	Máy không hoạt động	Creative	<input checked="" type="checkbox"/>
6	Màn hình bị điểm chết, bị lỗi	NB Lenovo G480 - 1145	<input checked="" type="checkbox"/>
7	Lỗi phần mềm không kết nối được với dây mạng	DELL Inspiron 15 3521 (HNP6M4-Black)	<input checked="" type="checkbox"/>
8	Không nạp được pin	NB Asus X402CA - WX042 (Trắng)	<input checked="" type="checkbox"/>
9	Hư loa	MP3 4GB Sony NWZ-E473	<input checked="" type="checkbox"/>
10	Dumping RAM	NB Lenovo G480 - 1145	<input checked="" type="checkbox"/>

Fill data and click on **Update** button

The screenshot shows a 'SYMPOTM DETAIL' page. At the top, it says 'The following information is the detail of this symptom'. Below this is a 'BASIC INFORMATION' section with fields for 'Symptom Name' (containing 'Lỗi phần mềm không kết nối được với dây mạng'), 'Product Relate' (containing 'DELL Inspiron 15 3521 (HNP6M4-Black)'), and 'Active' (with a checked checkbox). There is also a 'Solution' section with a rich text editor toolbar and a list of troubleshooting steps. At the bottom are 'CLOSE' and 'UPDATE' buttons.

4.5.4. Manage Product Category

4.5.4.1. View Product Category list

Technical manager of system uses this function.

To view all product category lists, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Product** tab in [Manage Repair Maintenance]
- [Manage Product Category] Page will be shown on screen with table [Category List]

The screenshot shows the 'Product Category' management interface. At the top, there is a navigation bar with tabs: REQUEST, TASK, SYMPTOM, PRODUCT (which is highlighted with a red box and has an arrow pointing to it labeled 'Click'), EQUIPMENT, FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. Below the navigation bar is a sidebar with 'PRODUCT CATEGORY' and sub-options: PRODUCT PROTOTYPE and PRODUCT MODEL. The main area is titled 'Product Category' and contains a 'CATEGORY LIST' table. The table has columns: NO., CATEGORY NAME, DEFAULT COST, and ACTIVE. The data in the table is as follows:

NO.	CATEGORY NAME	DEFAULT COST	ACTIVE
1	Laptop	9990	<input checked="" type="checkbox"/>
2	may chong nong	1000	<input checked="" type="checkbox"/>
3	MP3-MP4	10000	<input checked="" type="checkbox"/>
4	Removable HDD	30000	<input checked="" type="checkbox"/>
5	Tablet	8000	<input checked="" type="checkbox"/>
6	USB 3G	500	<input checked="" type="checkbox"/>

At the bottom of the table, there are buttons for 'Prev', '1', 'Next', and 'REFRESH'. A red arrow points to the 'Category List' table with the label 'Category List'.

- Technical manager can filter product category by name.

The screenshot shows the 'Product Category' section of a software interface. At the top, there are tabs for REQUEST, TASK, SYMPTOM, PRODUCT (which is highlighted in orange), EQUIPMENT, FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. On the left, there are three buttons: PRODUCT PROTOTYPE, PRODUCT MODEL, and PRODUCT CATEGORY (which is also highlighted). The main area is titled 'Product Category' and contains a 'CATEGORY LIST'. It includes a search bar with a placeholder 'Filter content' and a 'SEARCH NAME' button. Below the search bar, it says 'Display 10 of 3 total.' A red arrow points from the text 'Filter result' to the search bar. The table below has columns: NO., CATEGORY NAME, DEFAULT COST, and ACTIVE. The data is as follows:

NO.	CATEGORY NAME	DEFAULT COST	ACTIVE
1	may chong nong	1000	<input checked="" type="checkbox"/>
2	MP3-MP4	10000	<input checked="" type="checkbox"/>
3	Removable HDD	30000	<input checked="" type="checkbox"/>

At the bottom of the table are buttons for 'Prev', '1', and 'Next', along with a 'REFRESH' button.

4.5.4.2.Add Product Category

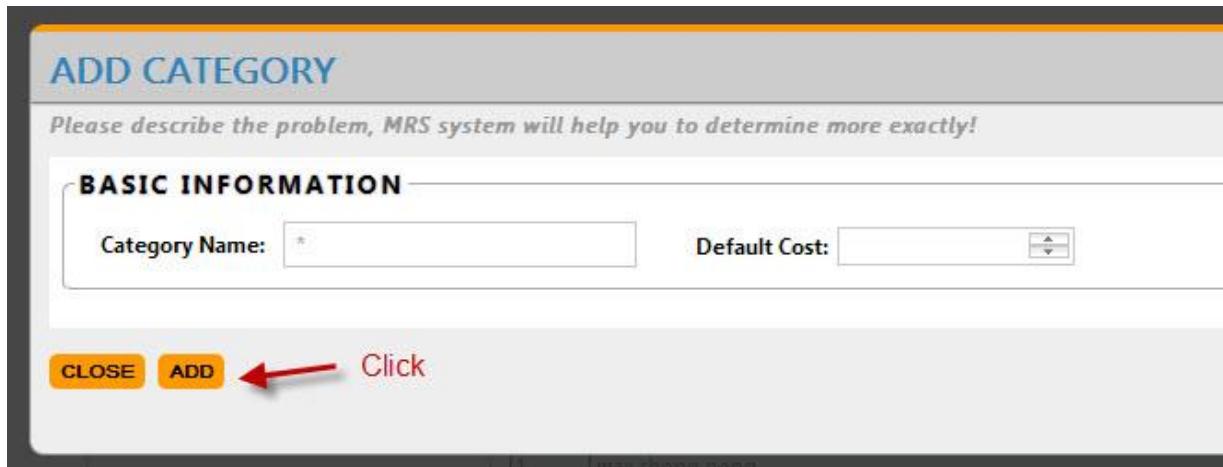
Technical manager of system uses this function.

To add new product category to Category list, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **Product** tab in [Manage Repair Maintenance]
- [Manage Product Category] Page will be shown on screen with table [Category List]
- Click on **Add Category** button to show Add Category form

This screenshot is similar to the one above, showing the 'Product Category' list. However, the 'ADD CATEGORY' button in the top-left corner of the list area is highlighted with a red box and a red arrow pointing to it, indicating where the user should click to add a new category.

Fill all fields and click **Add** button



4.5.4.3. Update Product Category

Technical manager of system uses this function.

To update a product category, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **Product** tab in [Manage Repair Maintenance]
- [Manage Product Category] Page will be shown on screen with table [Category List]
- Click on a row of table [Category List] to view Category detail form.

REQUEST **TASK** **SYMPTOM** **PRODUCT** **EQUIPMENT** **FAQ** **PROMOTION** **STATISTIC** **ACCOUNT** **INVOICE**

PRODUCT CATEGORY

PRODUCT PROTOTYPE

PRODUCT MODEL

Product Category

CATEGORY LIST

ADD CATEGORY

Display 10 of 6 total.

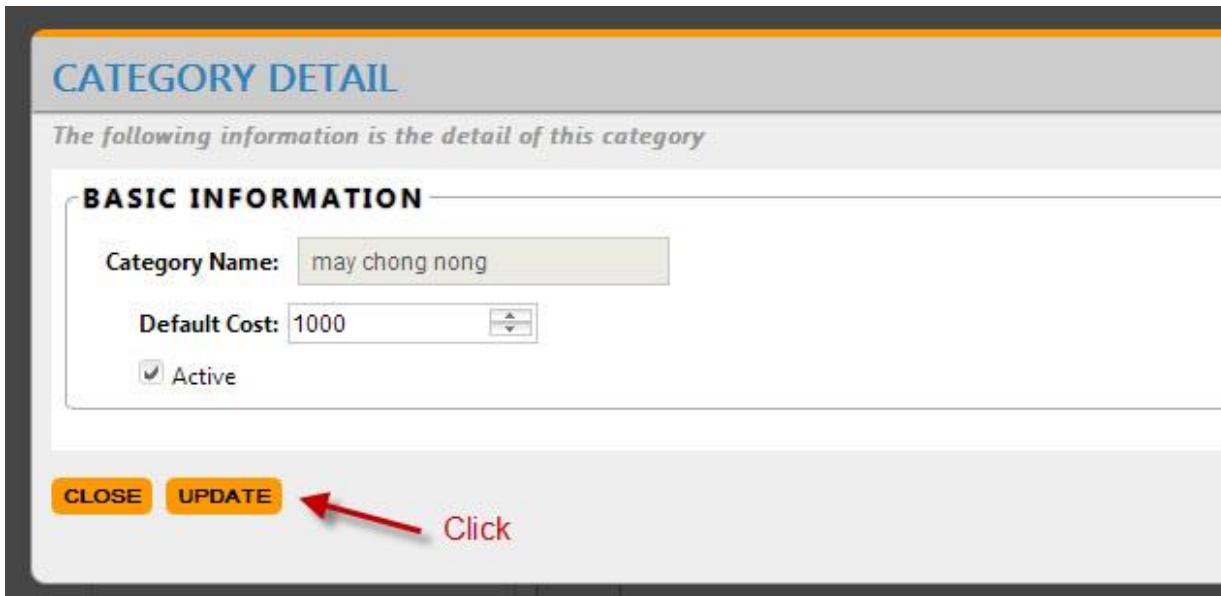
NO.	CATEGORY NAME	DEFAULT COST	ACTIVE
1	Laptop	9990	<input checked="" type="checkbox"/>
2	may chong nong	1000	<input checked="" type="checkbox"/>
3	MP3-MP4	10000	<input checked="" type="checkbox"/>
4	Removable HDD	30000	<input checked="" type="checkbox"/>
5	Tablet	8000	<input checked="" type="checkbox"/>
6	USB 3G	500	<input checked="" type="checkbox"/>

SEARCH NAME

Prev 1 Next REFRESH

Click

- Fill data and click **Update** button



4.5.5. Manage Product Prototype

4.5.5.1. View Product Prototype list

Technical manager of system uses this function.

To view all product prototype lists, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Product** tab in [Manage Repair Maintenance] to show [Manage Product Category] Page
- Click on **Product Prototype** link on the menu left

The screenshot shows a 'Product Category' page. At the top, there's a navigation bar with tabs: REQUEST, TASK, SYMPTOM, PRODUCT (which is highlighted in orange), EQUIPMENT, FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. Below the navigation bar is a sidebar with three links: PRODUCT CATEGORY (highlighted in orange), PRODUCT PROTOTYPE (which has a red arrow pointing to it with the word 'Click'), and PRODUCT MODEL. The main content area is titled 'CATEGORY LIST' and contains a table with the following data:

NO.	CATEGORY NAME	DEFAULT COST	ACTIVE
1	Laptop	9990	<input checked="" type="checkbox"/>
2	may chong nong	1000	<input checked="" type="checkbox"/>
3	MP3-MP4	10000	<input checked="" type="checkbox"/>

[Manage Product Prototype] Page will be shown with table [Prototype List]

The screenshot shows the 'Product Prototype' module with the 'PRODUCT' tab selected. The main title is 'Product Prototype' and the sub-section is 'PROTOTYPE LIST'. The list displays 10 items out of 14 total, with a 'Display' dropdown set to 10. A 'Prototype Name' filter is present. The table columns are NO., NAME, DESCRIPTION, CATEGORY, and ACTIVE. The data includes various computer components like Acer, Asus, Compaq-HP, Creative, Dell, Desktop Hard Drive, IBM-Lenovo, Portable Hard Drive, Samsung, and Sony.

NO.	NAME	DESCRIPTION	CATEGORY	ACTIVE
1	Acer	Acer	Laptop	<input checked="" type="checkbox"/>
2	Asus	Asus	Laptop	<input checked="" type="checkbox"/>
3	Compaq-HP	Compaq-HP	Laptop	<input checked="" type="checkbox"/>
4	Creative	Creative	MP3-MP4	<input checked="" type="checkbox"/>
5	Dell	Dell	Laptop	<input checked="" type="checkbox"/>
6	Desktop Hard Drive	Desktop Hard Drive	Removable HDD	<input checked="" type="checkbox"/>
7	IBM-Lenovo	IBM-Lenovo	Laptop	<input checked="" type="checkbox"/>
8	Portable Hard Drive	Portable Hard Drive	Removable HDD	<input checked="" type="checkbox"/>
9	Samsung	Samsung	Laptop	<input checked="" type="checkbox"/>
10	Sony	Hàng sony, dòng laptop vaio	Laptop	<input checked="" type="checkbox"/>

- Technical manager can filter product prototype by name.

The screenshot shows the 'Product Prototype' module with the 'PRODUCT' tab selected. The main title is 'Product Prototype' and the sub-section is 'PROTOTYPE LIST'. A red arrow points to the 'Filter Content' input field. Another red arrow points to the dropdown menu for 'Prototype Name' and 'Category Name'. A third red arrow points to the 'FILTER' button. The list displays 10 items out of 14 total, with a 'Display' dropdown set to 10. The table columns are NO., NAME, DESCRIPTION, CATEGORY, and ACTIVE. The data includes various computer components like Acer, Asus, Compaq-HP, Creative, Dell, Desktop Hard Drive, IBM-Lenovo, Portable Hard Drive, Samsung, and Sony.

NO.	NAME	DESCRIPTION	CATEGORY	ACTIVE
1	Acer	Acer	Laptop	<input checked="" type="checkbox"/>
2	Asus	Asus	Laptop	<input checked="" type="checkbox"/>
3	Compaq-HP	Compaq-HP	Laptop	<input checked="" type="checkbox"/>
4	Creative	Creative	MP3-MP4	<input checked="" type="checkbox"/>
5	Dell	Dell	Laptop	<input checked="" type="checkbox"/>
6	Desktop Hard Drive	Desktop Hard Drive	Removable HDD	<input checked="" type="checkbox"/>
7	IBM-Lenovo	IBM-Lenovo	Laptop	<input checked="" type="checkbox"/>
8	Portable Hard Drive	Portable Hard Drive	Removable HDD	<input checked="" type="checkbox"/>

4.5.5.2.Add Product Prototype

Technical manager of system uses this function.

To add new product prototype to Prototype list, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown

- Click **Product** tab in [Manage Repair Maintenance] to show [Manage Product Category] Page
- Click on **Product Prototype** link on the menu left
- [Manage Product Prototype] Page will be shown with table [Prototype List]
- Click on **Add Prototype** button to show Add Prototype form

The screenshot shows a navigation bar with tabs: REQUEST, TASK, SYMPTOM, PRODUCT (highlighted in orange), EQUIPMENT, FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. Below the navigation bar, there are three buttons: PRODUCT CATEGORY, PRODUCT PROTOTYPE (highlighted in orange), and PRODUCT MODEL. A main content area titled 'Product Prototype' displays a table titled 'PROTOTYPE LIST'. The table has columns: NO., NAME, DESCRIPTION, CATEGORY, and ACTIVE. At the top of the table, there is a red box around the 'ADD PROTOTYPE' button. A red arrow points to this button with the text 'Click'.

Fill all fields and click **Add** button:

The screenshot shows a modal dialog box titled 'ADD PROTOTYPE'. Inside the dialog, there is a message: 'Please describe the problem, MRS system will help you to determine more exactly!'. Below this is a section titled 'BASIC INFORMATION' with the following fields: 'Prototype Name:' (with an asterisk) and 'Product Category:' (set to 'Laptop'). There is also a 'Description' text area containing an asterisk (*). At the bottom of the dialog, there are two buttons: 'CLOSE' and 'ADD'. A red arrow points to the 'ADD' button with the text 'Click'.

4.5.5.3. Update Product Prototype

Technical manager of system uses this function.

To update a product prototype, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Product** tab in [Manage Repair Maintenance] to show [Manage Product Category] Page

- Click on **Product Prototype** link on the menu left
- [Manage Product Prototype] Page will be shown with table [Prototype List]
- Click on a row of table [Prototype List] to view Prototype detail form.

NO.	NAME	DESCRIPTION	CATEGORY	ACTIVE
1	Acer	Acer	Laptop	<input checked="" type="checkbox"/>
2	Asus	Asus	Laptop	<input checked="" type="checkbox"/>
3	Compaq-HP	Compaq-HP	Laptop	<input checked="" type="checkbox"/>
4	Creative	Creative	MP3-MP4	<input checked="" type="checkbox"/>
5	Dell	Dell	Laptop	<input checked="" type="checkbox"/>
6	Desktop Hard Drive	Desktop Hard Drive	Removable HDD	<input checked="" type="checkbox"/>
7	IBM-Lenovo	IBM-Lenovo	Laptop	<input checked="" type="checkbox"/>
8	Portable Hard Drive	Portable Hard Drive	Removable HDD	<input checked="" type="checkbox"/>
9	Samsung	Samsung	Laptop	<input checked="" type="checkbox"/>
10	Sony	Hàng sony, dòng laptop vaio	Laptop	<input checked="" type="checkbox"/>

- Fill data and click Update button

PROTOTYPE DETAIL
The following information is the detail of this prototype

BASIC INFORMATION

Prototype Name: Compaq-HP

Product Category: Laptop

Description
Compaq-HP

Active

CLOSE **UPDATE** ← Click

4.5.6. Manage Product Model

4.5.6.1. View Product Model list

Technical manager of system uses this function.

To view all product model lists, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Product** tab in [Manage Repair Maintenance] to show [Manage Product Category] Page
- Click on **Product Model** link on the menu left



[Manage Product Model] Page will be shown with table [Model List]

The screenshot shows a top navigation bar with tabs: REQUEST, TASK, SYMPTOM, PRODUCT (highlighted in orange), EQUIPMENT, FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. Below this is a secondary navigation bar with categories: PRODUCT CATEGORY, PRODUCT PROTOTYPE, and PRODUCT MODEL (highlighted with a red box). A red arrow points to the 'Model List' table. The main content area is titled 'MODEL LIST' with a sub-section 'ADD MODEL'. It displays 'Display 10 of 29 total.' and a filter section with 'Model Name' and 'FILTER' buttons. The table has columns: NO., NAME, CATEGORY, PRODUCT BRAND, and ACTIVE. The table data is as follows:

NO.	NAME	CATEGORY	PRODUCT BRAND	ACTIVE
1	2GB Sony NWZ B172	MP3-MP4	Sony Walkman	<input checked="" type="checkbox"/>
2	4GB Transcend MP300	MP3-MP4	Transcend	<input checked="" type="checkbox"/>
3	4GB Transcend MP330	MP3-MP4	Transcend	<input checked="" type="checkbox"/>
4	ASUS X202E CT141H // 140H // 044H	Laptop	Asus	<input checked="" type="checkbox"/>
5	Dell 3521 HNP6M6 (Den)	Laptop	Dell	<input checked="" type="checkbox"/>
6	DELL Inspiron 15 3521 (HNP6M4-Black)	Laptop	Dell	<input checked="" type="checkbox"/>
7	DELL Vostro 3460 (34RH44)	Laptop	Dell	<input checked="" type="checkbox"/>
8	HP 450 (C8J29PA)	Laptop	Compaq-HP	<input type="checkbox"/>
9	HP 450 (C8J31PA)	Laptop	Compaq-HP	<input checked="" type="checkbox"/>
10	HP Pavilion 14-e008TU (E3B86PA) (Silver)	Laptop	Compaq-HP	<input checked="" type="checkbox"/>

At the bottom of the table are buttons for 'Prev', '1' (selected), '2', '3', 'Next', and 'REFRESH'.

- Technical manager can filter product model by name.

The screenshot shows the 'Product Model' section of a software interface. At the top, there are tabs: REQUEST, TASK, SYMPTOM, PRODUCT (highlighted in orange), EQUIPMENT, FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. Below these are three buttons: PRODUCT CATEGORY, PRODUCT PROTOTYPE, and PRODUCT MODEL. The main area is titled 'MODEL LIST' and contains a table with columns: NO., NAME, CATEGORY, and ACTIVE. The table has three rows of data. Above the table are two filter boxes: 'Filter content' (with a dropdown set to 'Display 10 of 29 total.') and 'Filter type' (with a dropdown set to 'Model Name' and a 'Sony Walkman' entry). A red arrow points to the 'Model Name' dropdown, and another red arrow points to the 'FILTER' button.

NO.	NAME	CATEGORY	ACTIVE
1	2GB Sony NWZ B172	MP3-MP4	Sony Walkman
2	4GB Transcend MP300	MP3-MP4	Transcend
3	4GB Transcend MP30	MP3-MP4	Transcend

4.5.6.2.Add Product Model

Technical manager of system uses this function.

To add new product model to Model list, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Product** tab in [Manage Repair Maintenance] to show [Manage Product Category] Page
- Click on **Product Model** link on the menu left
- [Manage Product Prototype] Page will be shown with table [Model List]
- Click on **Add Model** button to show Add Model form

The screenshot shows the same 'Product Model' section as before, but the 'ADD MODEL' button in the top-left corner of the 'MODEL LIST' table is now highlighted with a red box and a red arrow pointing to it. The rest of the interface is identical to the previous screenshot.

Fill all fields and click **Add** button:

ADD MODEL

Please describe the problem, MRS system will help you to determine more exactly!

BASIC INFORMATION

Model Name: *

Product Category: Laptop

Product Prototype: Acer

Description

CLOSE ADD Click

4.5.6.3. Update Product Model

Technical manager of system uses this function.

To update a product model, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Product** tab in [Manage Repair Maintenance] to show [Manage Product Category] Page
- Click on **Product Model** link on the menu left
- [Manage Product Model] Page will be shown with table [Model List]
- Click on a row of table [Model List] to view Model detail form.

Product Model

MODEL LIST

Display 10 of 29 total.

NO.	NAME	CATEGORY	PRODUCT BRAND	ACTIVE
1	2GB Sony NWZ B172	MP3-MP4	Sony Walkman	<input checked="" type="checkbox"/>
2	4GB Transcend MP300	MP3-MP4	Transcend	<input checked="" type="checkbox"/>
3	4GB Transcend MP330	MP3-MP4	Transcend	<input checked="" type="checkbox"/>
4	ASUS X202E CT141H // 140H // 044H	Laptop	Asus	<input checked="" type="checkbox"/>
5	Dell 3521 HNP6M6 (Den)	Laptop	Dell	<input checked="" type="checkbox"/>
6	DELL Inspiron 15 3521 (HNP6M4-Black)	Laptop	Dell	<input checked="" type="checkbox"/>
7	DELL Vostro 3460 (34RH44)	Laptop	Dell	<input checked="" type="checkbox"/>
8	HP 450 (C8J29PA)	Laptop	Compaq-HP	<input type="checkbox"/>
9	HP 450 (C8J31PA)	Laptop	Compaq-HP	<input checked="" type="checkbox"/>
10	HP Pavilion 14-e008TU (E3B86PA) (Silver)	Laptop	Compaq-HP	<input checked="" type="checkbox"/>

Prev 1 2 3 Next REFRESH

- Fill data and click **Update** button

MODEL DETAIL

The following information is the detail of this model

BASIC INFORMATION

Model Name:	4GB Transcend MP330
Product Category:	MP3-MP4
Product Prototype:	Acer

Description

Máy nghe nhạc MP3 4GB MP330 Transcend TS4GMP330
(Đen // Hồng)

Active

CLOSE UPDATE Click

4.5.7. Manage Equipment

4.5.7.1. View equipment list

Technical manager of system uses this function.

To view all equipment lists, technical manager can do by following steps:

- Login to the system

- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Equipment** tab in [Manage Repair Maintenance]
- [Manage Equipment] Page will be shown on screen with table [Equipment List]

The screenshot shows the 'Equipment List' page. At the top, there is a navigation bar with tabs: REQUEST, TASK, SYMPTOM, PRODUCT, EQUIPMENT (which is highlighted in orange), FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. Below the navigation bar, there is a section titled 'EQUIPMENT CATEGORY' with sub-sections 'EQUIPMENT' and 'Equipment Category'. A red arrow points to the 'EQUIPMENT' tab. The main content area is titled 'EQUIPMENT LIST' and contains a table with 10 rows of equipment data. The columns are labeled: NO., NAME, CATEGORY, PRICE, and ACTIVE. The table shows various motherboards with their respective details. At the bottom of the table, there are navigation buttons for 'Prev', '1', '2', '3', '4', '5', '...', '8', '9', 'Next', and a 'REFRESH' button.

NO.	NAME	CATEGORY	PRICE	ACTIVE
1	MB Intel Box DB85FL	CPU	1000	<input checked="" type="checkbox"/>
2	Zotac H61 M-ATX	CPU	930	<input checked="" type="checkbox"/>
3	Foxconn H61MXE	CPU	950	<input checked="" type="checkbox"/>
4	ASRock H61M-VS	CPU	950	<input checked="" type="checkbox"/>
5	GIGABYTE GA-B75M-D2V (rev. 1.0)	CPU	1490	<input checked="" type="checkbox"/>
6	GIGABYTE GA-A55M-DS2	CPU	1480	<input checked="" type="checkbox"/>
7	Intel BLKDH61WWB3	CPU	1290	<input checked="" type="checkbox"/>
8	GIGABYTE GA-P61A-D3	CPU	1600	<input checked="" type="checkbox"/>
9	ASUS P8B75-M	CPU	1733	<input checked="" type="checkbox"/>
10	Intel BLKDH77KC (ATX)	CPU	2280	<input checked="" type="checkbox"/>

- Technical manager can filter equipment by name or category name.

The screenshot shows the 'Equipment List' page with filtering options. At the top, there is a navigation bar with tabs: REQUEST, TASK, SYMPTOM, PRODUCT, EQUIPMENT (highlighted in orange), FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. Below the navigation bar, there is a section titled 'EQUIPMENT CATEGORY' with sub-sections 'EQUIPMENT' and 'Equipment Category'. The main content area is titled 'EQUIPMENT LIST' and contains a table with 3 rows of equipment data. To the right of the table, there are three dropdown menus: 'Filter content' (set to 'Name'), 'Filter type' (set to 'Category Name'), and a 'FILTER' button. Red arrows point to each of these three elements.

NO.	NAME	CATEGORY	PRICE	ACTIVE
1	MB Intel Box DB85FL	CPU	1000	<input checked="" type="checkbox"/>
2	Zotac H61 M-ATX	CPU	930	<input checked="" type="checkbox"/>
3	Foxconn H61MXE	CPU	950	<input checked="" type="checkbox"/>

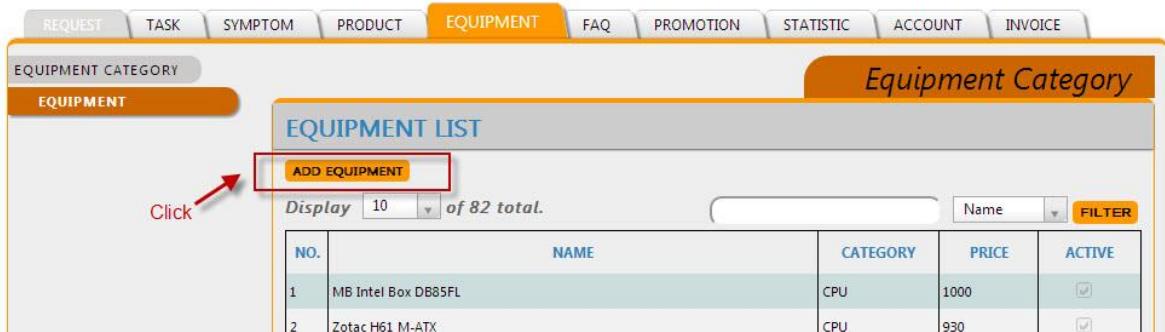
4.5.7.2.Add Equipment

Technical manager of system uses this function.

To add new equipment to Equipment list, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **Equipment** tab in [Manage Repair Maintenance]

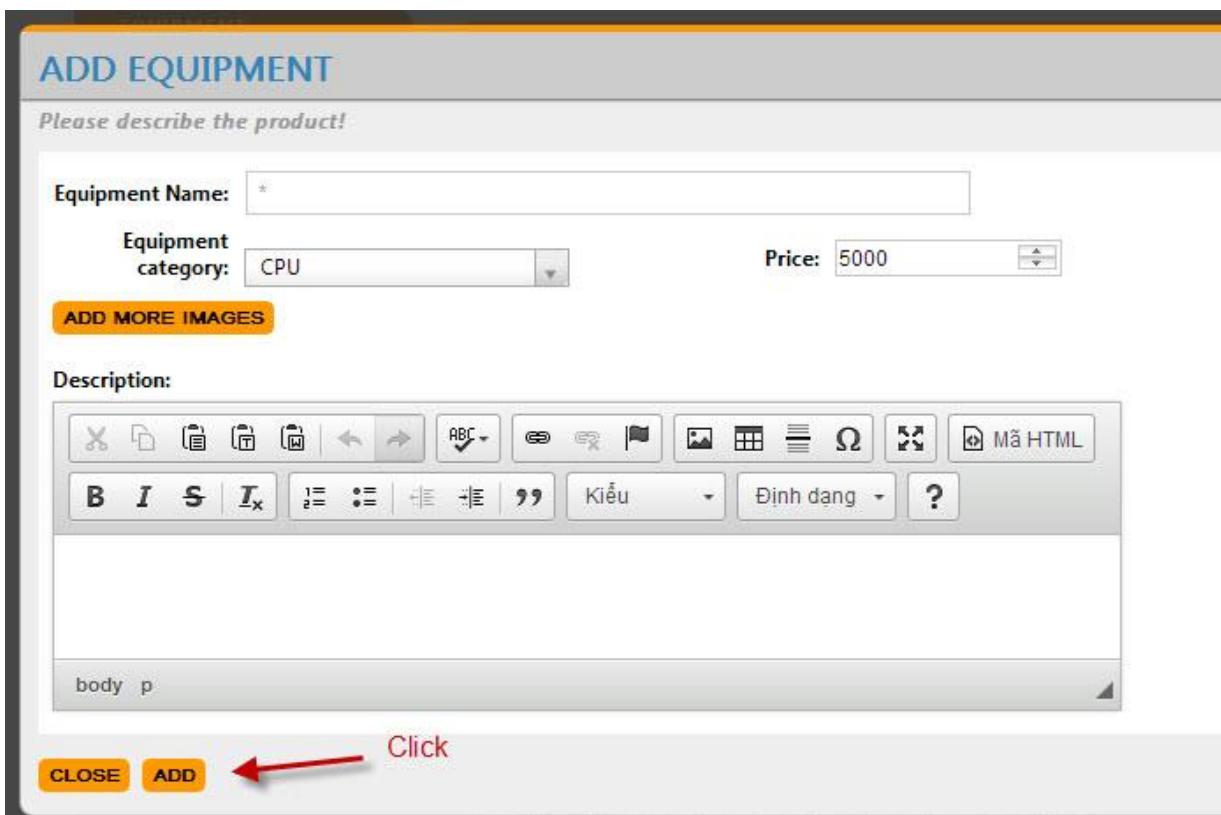
- [Manage Equipment] Page will be shown on screen with table [Equipment List]
- Click on **Add Equipment** button to show Add Equipment form



The screenshot shows the 'Equipment Category' section of a software interface. At the top, there are tabs: REQUEST, TASK, SYMPTOM, PRODUCT, EQUIPMENT (which is highlighted in orange), FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. Below the tabs, there's a sub-menu for 'EQUIPMENT' with options like EQUIPMENT CATEGORY and EQUIPMENT. The main area is titled 'EQUIPMENT LIST' and displays a table with two rows of equipment data. The columns are labeled NO., NAME, CATEGORY, PRICE, and ACTIVE. Row 1: NO. 1, NAME MB Intel Box DB85FL, CATEGORY CPU, PRICE 1000, ACTIVE checked. Row 2: NO. 2, NAME Zotac H61 M-ATX, CATEGORY CPU, PRICE 930, ACTIVE checked. Below the table are buttons for 'Display' (set to 10), 'of 82 total.', a search bar, and a 'FILTER' button.

NO.	NAME	CATEGORY	PRICE	ACTIVE
1	MB Intel Box DB85FL	CPU	1000	<input checked="" type="checkbox"/>
2	Zotac H61 M-ATX	CPU	930	<input checked="" type="checkbox"/>

Fill all fields and click **Add** button



The screenshot shows the 'ADD EQUIPMENT' form. The title is 'ADD EQUIPMENT' and a sub-instruction says 'Please describe the product!'. There are three main input fields: 'Equipment Name:' with a required asterisk, 'Equipment category:' with a dropdown menu showing 'CPU', and 'Price:' with a numeric input field set to '5000'. Below these are buttons for 'ADD MORE IMAGES' and a rich text editor toolbar. The rich text editor toolbar includes icons for bold, italic, underline, etc. Below the toolbar is a large text area with placeholder text 'body p'. At the bottom of the form are two buttons: 'CLOSE' and 'ADD', with a red arrow pointing to the 'ADD' button.

4.5.7.3. Update Equipment

Technical manager of system uses this function.

To update equipment, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **Equipment** tab in [Manage Repair Maintenance]
- [Manage Equipment] Page will be shown on screen with table [Equipment List]

- Click on a row of table [Equipment List] to view Equipment detail form.

NO.	NAME	CATEGORY	PRICE	ACTIVE
1	MB Intel Box DB85FL	CPU	1000	<input checked="" type="checkbox"/>
2	Zotac H61 M-ATX	CPU	930	<input checked="" type="checkbox"/>
3	Foxconn H61MXE	CPU	950	<input checked="" type="checkbox"/>
4	ASRock H61M-VS	CPU	950	<input checked="" type="checkbox"/>
5	GIGABYTE GA-B75M-D2V (rev. 1.0)	CPU	1490	<input checked="" type="checkbox"/>
6	GIGABYTE GA-A55M-DS2	CPU	1480	<input checked="" type="checkbox"/>
7	Intel BLKDH61WWB3	CPU	1290	<input checked="" type="checkbox"/>
8	GIGABYTE GA-P61A-D3	CPU	1600	<input checked="" type="checkbox"/>
9	ASUS P8B75-M	CPU	1733	<input checked="" type="checkbox"/>
10	Intel BLKDH77KC (ATX)	CPU	2280	<input checked="" type="checkbox"/>

Display 10 of 82 total. Name FILTER

Prev 1 2 3 4 5 ... 8 9 Next REFRESH

- Fill data and click Update button

EQUIPMENT DETAIL
The following information is the detail of this equipment

Equipment Name: Foxconn H61MXE

Equipment category: CPU Price: 950

ADD MORE IMAGES

Image Name: Foxconn H61MXE_1.jpg  

Image Name: Foxconn H61MXE_2.jpg  

CLOSE UPDATE 

4.5.8. Manage FAQ

4.5.8.1. View FAQ list

Technical manager of system uses this function.

To view all FAQ lists, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **FAQ** tab in [Manage Repair Maintenance]
- [Manage FAQ] Page will be shown on screen with table [FAQ List]

NO.	SUBJECT	PRODUCT PROTOTYPE	ACTIVE
1	Ổ đĩa chạy chậm	NB Asus X402CA - WX042 (Trắng)	<input checked="" type="checkbox"/>
2	Dumping RAM	HP Pavilion 14-e008TU (E3B86PA) (Silver)	<input checked="" type="checkbox"/>
3	Máy quá nóng	HP Probook P4440S A5K36AV-2	<input checked="" type="checkbox"/>

- Technical manager can filter FAQ by name.

NO.	SUBJECT	PRODUCT PROTOTYPE	ACTIVE
1	Dumping RAM	HP Pavilion 14-e008TU (E3B86PA) (Silver)	<input checked="" type="checkbox"/>

4.5.8.2. Add FAQ

Technical manager of system uses this function.

To add new FAQ, technical manager can do by following steps:

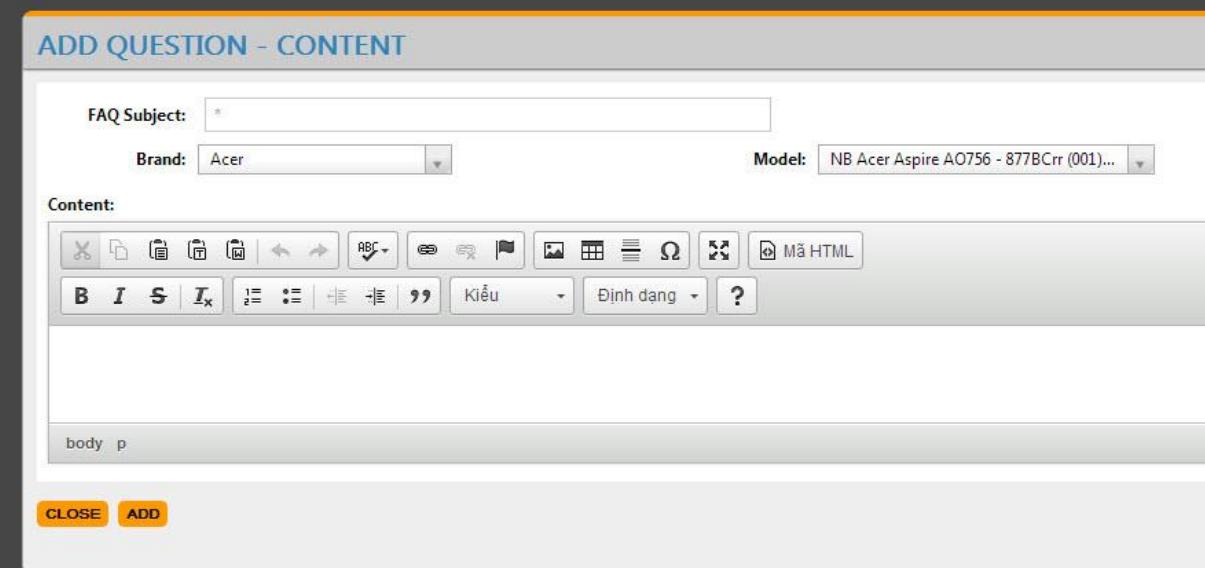
- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **FAQ** tab in [Manage Repair Maintenance]

- [Manage FAQ] Page will be shown on screen with table [FAQ List]
- Click on **Add FAQ** button to show Add FAQ form



NO.	SUBJECT	PRODUCT PROTOTYPE	ACTIVE
1	Ổ đĩa chạy chậm	NB Asus X402CA - WX042 (Trắng)	<input checked="" type="checkbox"/>
2	Dumping RAM	HP Pavilion 14-e008TU (E3B86PA) (Silver)	<input checked="" type="checkbox"/>
3	Máy quá nóng	HP Probook P4440S A5K36AV-2	<input checked="" type="checkbox"/>

Fill all fields and click **Add** button:



FAQ Subject: *

Brand: Acer

Model: NB Acer Aspire AO756 - 877BCrr (001)...

Content:

body p

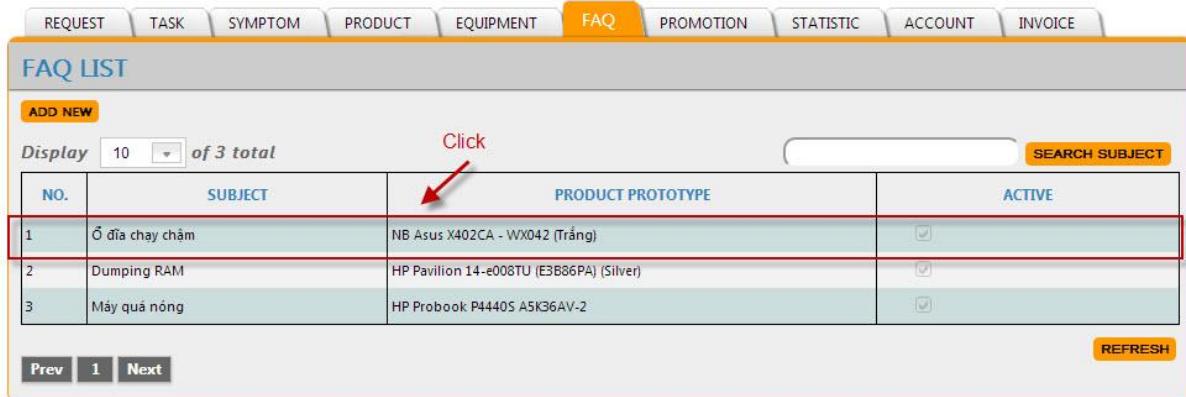
CLOSE ADD

4.5.8.3. Update FAQ

Technical manager of system uses this function.

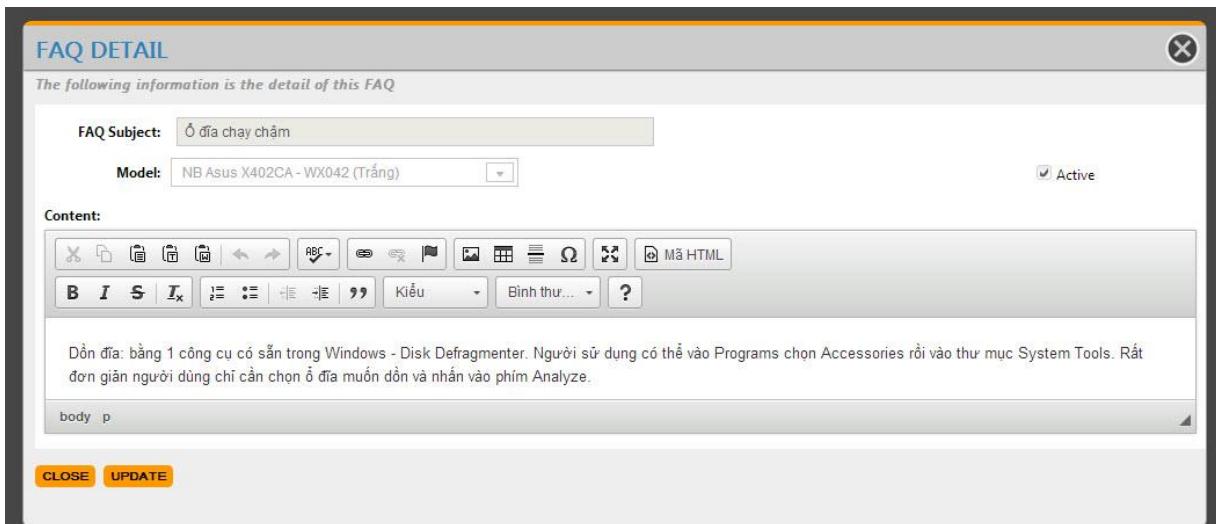
To update FAQ, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **FAQ** tab in [Manage Repair Maintenance]
- [Manage FAQ] Page will be shown on screen with table [FAQ List]
- Click on a row of table [FAQ List] to view FAQ detail form.



NO.	SUBJECT	PRODUCT PROTOTYPE	ACTIVE
1	Ổ đĩa chạy chậm	NB Asus X402CA - WX042 (Trắng)	<input checked="" type="checkbox"/>
2	Dumping RAM	HP Pavilion 14-e008TU (E3B86PA) (Silver)	<input checked="" type="checkbox"/>
3	Máy quá nóng	HP Probook P4440S A5K36AV-2	<input checked="" type="checkbox"/>

- Fill data and click Update button



The following information is the detail of this FAQ

FAQ Subject:

Model: Active

Content:

Dồn đĩa: bằng 1 công cụ có sẵn trong Windows - Disk Defragmenter. Người sử dụng có thể vào Programs chọn Accessories rồi vào thư mục System Tools. Rất đơn giản người dùng chỉ cần chọn ổ đĩa muốn dồn và nhấn vào phím Analyze.

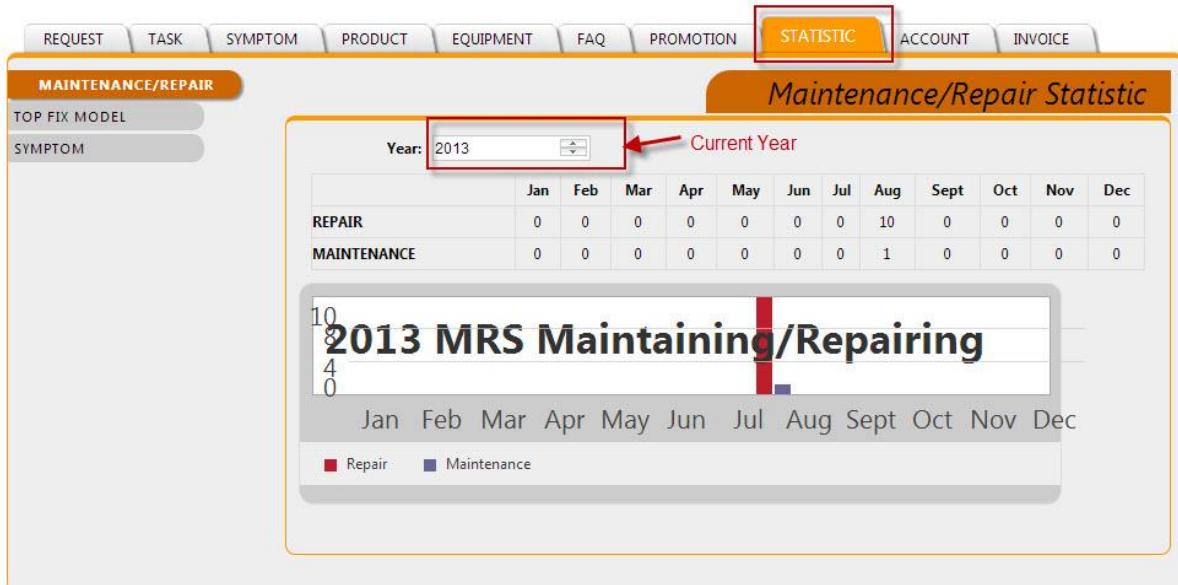
4.5.9. Manage Statistic

4.5.9.1. Statistic by Request Type

Technical manager of system uses this function.

To statistic by request type, technical manager can do by following steps:

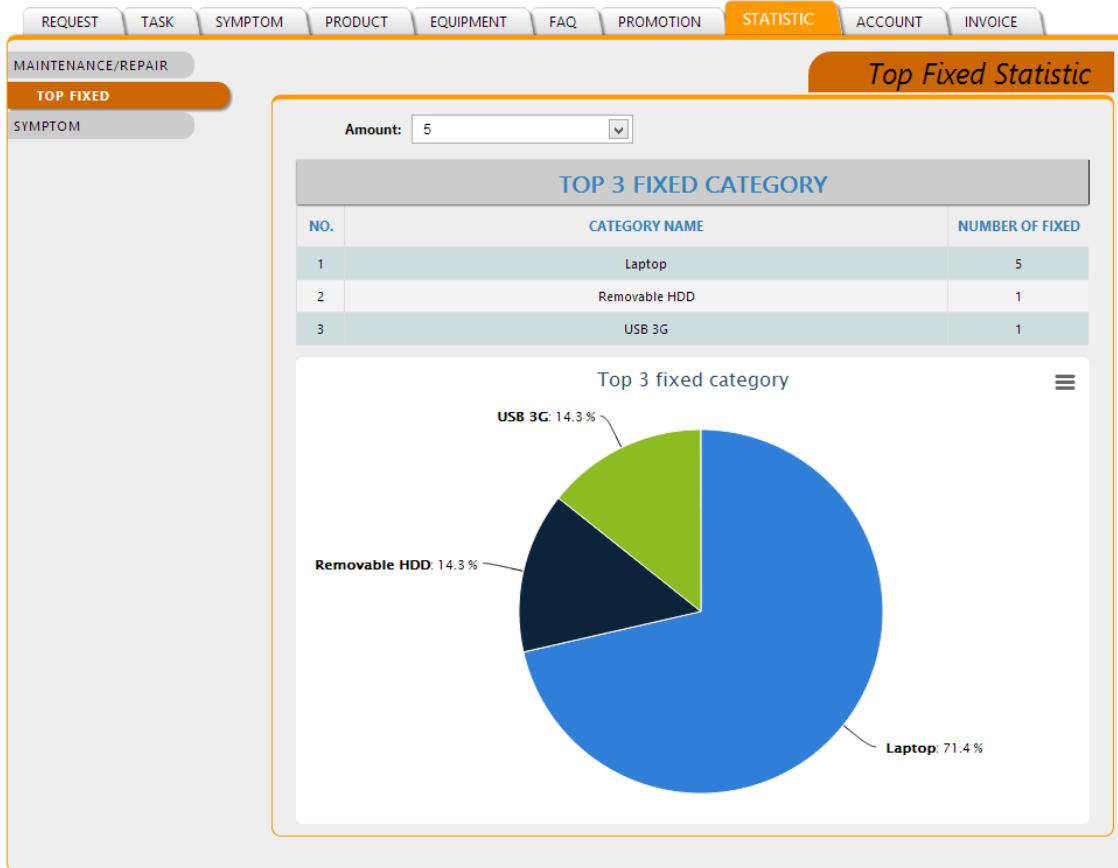
- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Statistic** tab in [Manage Repair Maintenance]
- [Manage Statistic] Page will be shown on screen with chart of current year.



- Technical manager can filter other year in **Year** textbox.

4.5.9.2. Statistic Top Fixed Model

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Statistic** tab in [Manage Repair Maintenance]
- Click **Top Fixed** link on the left menu in [Manage Statistic] Page
- [Top Fixed Model Statistics] Page will be shown on screen with chart.



- Technical manager can filter other year in **Amount** textbox.

4.5.9.3. Statistic by Symptom

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Statistic** tab in [Manage Repair Maintenance]
- Click **Symptom** link on the left menu in [Manage Statistic] Page
- [Top Fixed Symptom Statistics] Page will be shown on screen with chart.

The screenshot shows a web-based application interface titled "Symptom Statistic". At the top, there is a navigation bar with tabs: REQUEST, TASK, SYMPTOM, PRODUCT, EQUIPMENT, FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. The STATISTIC tab is highlighted. Below the navigation bar, there is a section titled "MAINTENANCE/REPAIR" and "TOP FIX MODEL". A red box highlights the "SYMPTOM" tab. On the right side of the screen, there is a search form with fields for "Amount" (set to 5), "Top Product categories" (set to Laptop), and "Top Product model" (set to Vaio Duo 17). Below the search form, a table titled "TOP 1 SYMPTOMS OF VAIO DUO 17" is displayed, showing one row with "NO.", "SYMPTOM NAME", and "SOLUTION". The first row shows "1", "Hồng cảm ứng màn hình", and "da up date".

- Technical manager can filter other year in **Amount** textbox or **Product Category**, **Product Model**

4.5.10. Manage Account

4.5.10.1. View Account list

Technical manager of system uses this function.

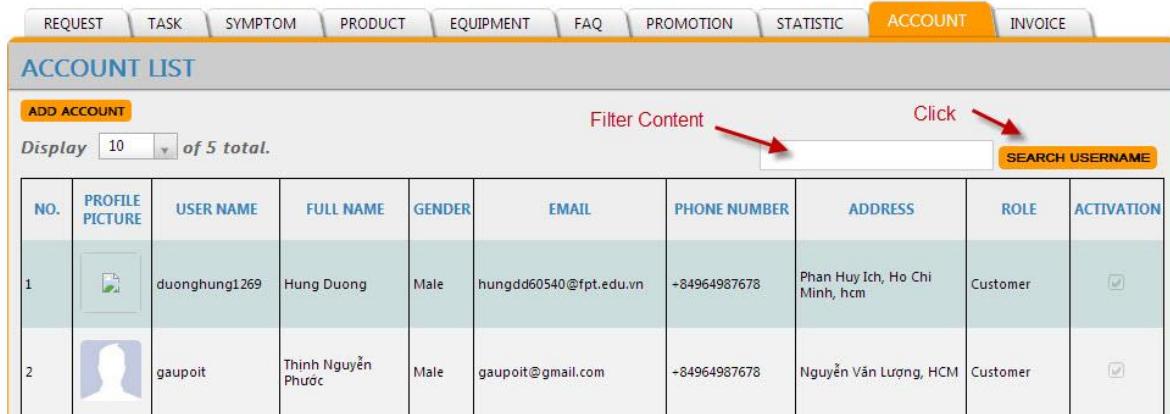
To view Account lists, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Account** tab in [Manage Repair Maintenance]
- [Manage Account] Page will be shown on screen with table [Account List]

The screenshot shows a web-based application interface titled "ACCOUNT LIST". At the top, there is a navigation bar with tabs: REQUEST, TASK, SYMPTOM, PRODUCT, EQUIPMENT, FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. The ACCOUNT tab is highlighted. Below the navigation bar, there is a search bar with "ADD ACCOUNT" and "SEARCH USERNAME" fields. A red arrow points to the "Display" dropdown menu which shows "10" and "of 5 total.". Another red arrow points to the "Click" button. A red box highlights the "ACCOUNT" tab. Below the search bar, a table titled "Account List" is displayed, showing four rows of user information. The columns are: NO., PROFILE PICTURE, USER NAME, FULL NAME, GENDER, EMAIL, PHONE NUMBER, ADDRESS, ROLE, and ACTIVATION. The data in the table is as follows:

NO.	PROFILE PICTURE	USER NAME	FULL NAME	GENDER	EMAIL	PHONE NUMBER	ADDRESS	ROLE	ACTIVATION
1		duonghung1269	Hung Duong	Male	hungdd60540@fpt.edu.vn	+84964987678	Phan Huy Ich, Ho Chi Minh, hcm	Customer	<input checked="" type="checkbox"/>
2		gaupoit	Thinh Nguyễn Phước	Male	gaupoit@gmail.com	+84964987678	Nguyễn Văn Lượng, HCM	Customer	<input checked="" type="checkbox"/>
3		hungdd	hung dang duong	Male	hanbdi@gmail.com	+84964987678	Go Vap, Ho Chi Minh	Manager	<input checked="" type="checkbox"/>
4		kieuudtd	Kiều Dương Thị Diễm	Male	kieudtd60471@fpt.edu.vn	+840908783553	Quận 12, HCM	Staff	<input checked="" type="checkbox"/>

- Technical manager can filter Account by name.



NO.	PROFILE PICTURE	USER NAME	FULL NAME	GENDER	EMAIL	PHONE NUMBER	ADDRESS	ROLE	ACTIVATION
1		duonghung1269	Hung Duong	Male	hungdd60540@fpt.edu.vn	+84964987678	Phan Huy Ich, Ho Chi Minh, hcm	Customer	<input checked="" type="checkbox"/>
2		gaupoit	Thinh Nguyễn Phước	Male	gaupoit@gmail.com	+84964987678	Nguyễn Văn Lương, HCM	Customer	<input checked="" type="checkbox"/>

4.5.10.2. Add Account

Technical manager of system uses this function.

To add new Account, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **Account** tab in [Manage Repair Maintenance]
- [Manage Account] Page will be shown on screen with table [Account List]
- Click on **Add Account** button to show Add Account form



NO.	PROFILE PICTURE	USER NAME	FULL NAME	GENDER	EMAIL	PHONE NUMBER	ADDRESS	ROLE	ACTIVATION
1		duonghung1269	Hung Duong	Male	hungdd60540@fpt.edu.vn	+84964987678	Phan Huy Ich, Ho Chi Minh, hcm	Customer	<input checked="" type="checkbox"/>

Fill all fields and click **Add** button

ADD NEW ACCOUNT

BASIC INFORMATION

Role:	Customer
First name:	*
User Name:	*
Avatar:	Chọn Tệp tin Không có tệp nào được chọn
Email:	*
Phone number:	*
Gender:	Male
Address:	*

CLOSE **ADD**

4.5.10.3. Update Account

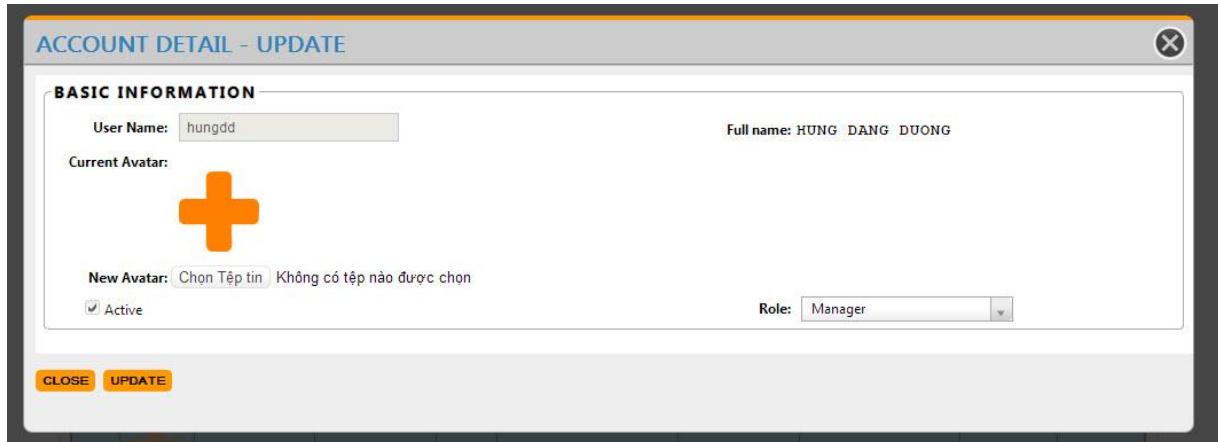
Technical manager of system uses this function.

To update Account, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **Account** tab in [Manage Repair Maintenance]
- [Manage Account] Page will be shown on screen with table [Account List]
- Click on a row of table [Account List] to view Account detail form.

ACCOUNT LIST									
ADD ACCOUNT Display 10 of 5 total. <input type="text"/> SEARCH USERNAME									
NO.	PROFILE PICTURE	USER NAME	FULL NAME	GENDER	EMAIL	PHONE NUMBER	ADDRESS	ROLE	ACTIVATION
1		duonghung1269	Hung Duong	Male	hungdd60540@fpt.edu.vn	+84964987678	Phan Huy Ich, Ho Chi Minh, hcm	Customer	<input checked="" type="checkbox"/>
2		gaupoit	Thinh Nguyễn Phước	Male	gaupoit@gmail.com	+84964987678	Nguyễn Văn Lượng, HCM	Customer	<input checked="" type="checkbox"/>
3		hungdd	hung dang duong	Male	hanbdi@gmail.com	+84964987678	Gò Vấp, Ho Chi Minh	Manager	<input checked="" type="checkbox"/>
4		kieudtd	Kiều Dương Thị Diễm	Male	kieudtd60471@fpt.edu.vn	+840908783553	Quận 12, HCM	Staff	<input checked="" type="checkbox"/>

- Update data and click **Update** button



4.5.11. Manage Invoice

4.5.11.1. View Request List

Technical manager of system uses this function.

To view Request lists that have status is **Finished**, **Cancelled** and **Closed**; technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown
- Click **Invoice** tab in [Manage Repair Maintenance]
- [Manage Invoice] Page will be shown on screen with table [Request List]

NO.	CODE	NAME	STATUS	CREATED DATE	FINISH DATE
0	PR144	request for cancel	Cancelled	08/15/2013 14:18	08/15/2013 14:33
1	PR145	request test lai	Closed	08/10/2013 16:12	08/10/2013 16:18
2	PR140	request test 2	Closed	08/10/2013 12:00	08/10/2013 12:12
3	PR139	request test	Finished	08/10/2013 11:02	08/10/2013 11:13
4	PR138	request test, request test	Cancelled	08/10/2013 10:59	

4.5.11.2. Update Invoice

Technical manager of system uses this function.

To update Invoice, technical manager can do by following steps:

- Login to the system
- Click on **Request** box on [Home Page]
- [Manage Repair Maintenance] Page will be shown on screen.
- Click **Invoice** tab in [Manage Repair Maintenance]

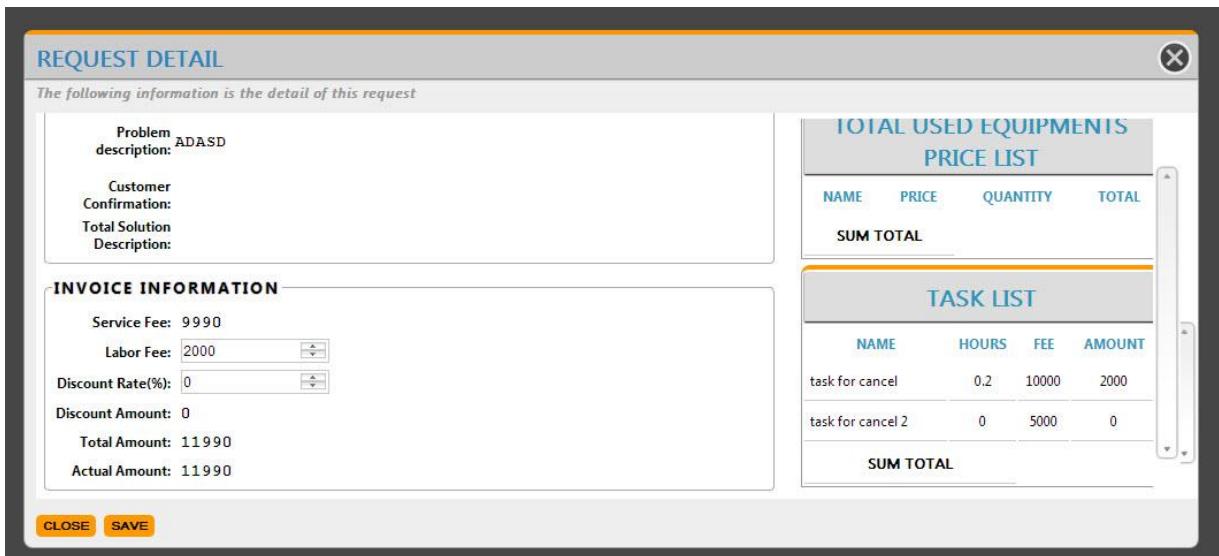
- [Manage Invoice] Page will be shown on screen with table [Request List]
- Click on a row of table [Request List] to view Request detail form.



The screenshot shows a software interface titled "REQUEST LIST". At the top, there is a navigation bar with tabs: REQUEST, TASK, SYMPTOM, PRODUCT, EQUIPMENT, FAQ, PROMOTION, STATISTIC, ACCOUNT, and INVOICE. The INVOICE tab is currently selected. Below the navigation bar, there is a search bar with the placeholder "Display 10 of 0 total." and a "SEARCH" button. The main area contains a table with columns: NO., CODE, NAME, STATUS, CREATED DATE, and FINISH DATE. The table has five rows of data. Row number 3, which corresponds to the request code PR139, is highlighted with a red border. The data for row 3 is: NO. 3, CODE PR139, NAME request test, STATUS Finished, CREATED DATE 08/10/2013 11:02, and FINISH DATE 08/10/2013 11:13. A "REFRESH" button is located at the bottom right of the table area.

NO.	CODE	NAME	STATUS	CREATED DATE	FINISH DATE
0	PR144	request for cancel	Cancelled	08/15/2013 14:18	08/15/2013 14:33
1	PR145	request test lai	Closed	08/10/2013 16:12	08/10/2013 16:18
2	PR140	request test 2	Closed	08/10/2013 12:00	08/10/2013 12:12
3	PR139	request test	Finished	08/10/2013 11:02	08/10/2013 11:13
4	PR138	request test, request test	Cancelled	08/10/2013 10:59	

- Click Save button to save invoice



The screenshot shows a modal dialog box titled "REQUEST DETAIL". Inside the dialog, there is a message "The following information is the detail of this request". On the left, there are three sections: "Problem description:" (ADASD), "Customer Confirmation:", and "Total Solution Description:". On the right, there are two summary tables. The top table is titled "TOTAL USED EQUIPMENTS PRICE LIST" and the bottom table is titled "TASK LIST". Both tables have columns: NAME, PRICE, QUANTITY, and TOTAL. The "SUM TOTAL" row is also present in both tables. At the bottom of the dialog, there are "CLOSE" and "SAVE" buttons.

TOTAL USED EQUIPMENTS PRICE LIST			
NAME	PRICE	QUANTITY	TOTAL
SUM TOTAL			

TASK LIST			
NAME	HOURS	FEE	AMOUNT
task for cancel	0.2	10000	2000
task for cancel 2	0	5000	0
SUM TOTAL			