|  |
| --- |
|  |

**FPT UNIVERSITY**

|  |  |
| --- | --- |
| Capstone Project Document | |
| **Email marketing platform with Simple Email Service of Amazon Cloud** | |
| Group 3 -JS | |
| **Group members** | Ngô Minh Tấn – SE62503 (Leader)  Nguyễn Lưu Hồng Sơn – SE62414  ~~Đặng Võ Anh Khoa – SE61550~~  Nguyễn Hữu Thắng –SE62447 |
| **Supervisor** | Lâm Hữu Khánh Phương |
| **Ext. Supervisor** | N/A |
| **Project Code** | EMP |

**– Ho Chi Minh City, May 13th, 2019 –**

This page is intentionally left blank

**Table of Contents**

[**Table of Contents** 3](#_Toc7065699)

[A. Introduction 12](#_Toc7065700)

[1. Project Information 12](#_Toc7065701)

[2. Introduction 12](#_Toc7065702)

[3. Current Situation 12](#_Toc7065703)

[4. Problem Definition 13](#_Toc7065704)

[5. Proposed Solution 13](#_Toc7065705)

[5.1. Functions 13](#_Toc7065706)

[5.2. Advantages and disadvantages 14](#_Toc7065707)

[6. Functional Requirements 15](#_Toc7065708)

[6.1. Registered User website and mobile application 15](#_Toc7065709)

[6.2. Admin website 15](#_Toc7065710)

[7. Role & Responsibility 16](#_Toc7065711)

[B. Software Project Management Plan 17](#_Toc7065712)

[1. Problem Definition 17](#_Toc7065713)

[1.1. Name of this Capstone Project 17](#_Toc7065714)

[1.2. Problem Abstract 17](#_Toc7065715)

[1.3. Project Overview 17](#_Toc7065716)

[2. Project Organization 21](#_Toc7065717)

[2.1. Software Process Model 21](#_Toc7065718)

[2.2. Roles and responsibilities 21](#_Toc7065719)

[2.3. Tools and Techniques 23](#_Toc7065720)

[3. Project Management Plan 23](#_Toc7065721)

[3.1. Product Backlog 23](#_Toc7065722)

[3.2. Sprint Backlog 23](#_Toc7065723)

[3.3. Deliverables 23](#_Toc7065724)

[3.4. All Meeting Minutes 24](#_Toc7065725)

[4. Coding Convention 24](#_Toc7065726)

[C. Software Requirement Specification 25](#_Toc7065727)

[1. User Requirement Specification 25](#_Toc7065728)

[1.1. Guest requirement 25](#_Toc7065729)

[1.2. Registered User requirement 25](#_Toc7065730)

[1.3. Staff requirement 26](#_Toc7065731)

[1.4. Admin requirement 27](#_Toc7065732)

[2. System Requirement Specification 27](#_Toc7065733)

[2.1. External Interface Requirement 27](#_Toc7065734)

[2.2. System Overview Use Case 29](#_Toc7065735)

[2.3. List of use case 31](#_Toc7065736)

[3. Software System Attribute 205](#_Toc7065737)

[3.1. Usability 205](#_Toc7065738)

[3.2. Reliability 205](#_Toc7065739)

[3.3. Availability 206](#_Toc7065740)

[3.4. Security 206](#_Toc7065741)

[3.5. Maintainability 206](#_Toc7065742)

[3.6. Portability 206](#_Toc7065743)

[3.7. Performance 206](#_Toc7065744)

[4. Conceptual Diagram 207](#_Toc7065745)

[D. Software Design Description 210](#_Toc7065746)

[1. Design Overview 210](#_Toc7065747)

[2. System Architecture Design 211](#_Toc7065748)

[2.1. Admin web application: 212](#_Toc7065749)

[2.2. Registered User web system: 212](#_Toc7065750)

[3. Component Diagram 213](#_Toc7065751)

[4. Detailed Description 215](#_Toc7065752)

[4.1. Class Diagram 215](#_Toc7065753)

[4.2. Class Diagram Explanation 217](#_Toc7065754)

[4.3. Interaction Diagram 239](#_Toc7065755)

[5. Interface 257](#_Toc7065756)

[5.1. Component Interface 257](#_Toc7065757)

[5.2. User Interface Design 257](#_Toc7065758)

[6. Database Design 307](#_Toc7065759)

[6.1. Entity Relationship Diagram (ERD) 307](#_Toc7065760)

[6.2. Data Dictionary 309](#_Toc7065761)

[7. AlgorithEMP 329](#_Toc7065762)

[7.1. Price suggestion model 329](#_Toc7065763)

[E. System implementation & Test 335](#_Toc7065764)

[1. Introduction 335](#_Toc7065765)

[1.1. Overview 335](#_Toc7065766)

[1.2. Test Approach 335](#_Toc7065767)

[2. Database Relationship Diagram 335](#_Toc7065768)

[2.1. Physical Diagram 335](#_Toc7065769)

[2.2. Data dictionary 337](#_Toc7065770)

[3. Test Plan 339](#_Toc7065771)

[3.1. Features to be tested 339](#_Toc7065772)

[3.2. Features not to be tested 339](#_Toc7065773)

[4. System Testing Test Case 340](#_Toc7065774)

[4.1. Registered User testing case: 340](#_Toc7065775)

[F. Software User’s Manual 343](#_Toc7065776)

[1. Installation Guide 343](#_Toc7065777)

[1.1. Setting up environment at server 343](#_Toc7065778)

[1.2. Deployment at server side 344](#_Toc7065779)

[1.3. Deploy web application on server 345](#_Toc7065780)

[2. User Guide 346](#_Toc7065781)

[2.1. Web application 346](#_Toc7065782)

[2.2. Mobile application 346](#_Toc7065783)

[G. Appendix 357](#_Toc7065784)

**List of tables**

[Table 1 : Roles and Responsibilities 18](#_Toc7065785)

[Table 2 - Hardware Requirements for Server 21](#_Toc7065786)

[Table 3 Hardware requirement for PC 22](#_Toc7065787)

[Table 4 Hardware requirement for the mobile 22](#_Toc7065788)

[Table 5 Development Environment - Software Requirement 22](#_Toc7065789)

[Table 6 Roles and responsibilities 25](#_Toc7065790)

[Table 7 Tools and techniques 25](#_Toc7065791)

[Table 8 Deliverables 26](#_Toc7065792)

[Table 9 <Guest> Login 35](#_Toc7065793)

[Table 10 <Guest> Register 37](#_Toc7065794)

[Table 11 <Guest> Search 39](#_Toc7065795)

[Table 12 <Registered User> Log out 46](#_Toc7065798)

[Table 13 <Registered User> Search equipment 48](#_Toc7065799)

[Table 14 <Registered User> Search materials 50](#_Toc7065800)

[Table 15 <Registered User> Search debris post 53](#_Toc7065801)

[Table 16 <Registered User> Add subscription 54](#_Toc7065802)

[Table 17 <Registered User> Remove subscription 56](#_Toc7065803)

[Table 18 <Registered User> Edit subscription 58](#_Toc7065804)

[Table 19 <Registered User> Edit profile 59](#_Toc7065805)

[Table 20 <Registered User> Report another Registered User 61](#_Toc7065806)

[Table 21 <Registered User> Add construction 63](#_Toc7065807)

[Table 22 <Registered User> Edit construction 65](#_Toc7065808)

[Table 23 <Registered User> Delete construction 66](#_Toc7065809)

[Table 24 <Supplier> View hiring transaction as supplier 69](#_Toc7065810)

[Table 25<Supplier> Accept pending hiring transaction 71](#_Toc7065811)

[Table 26 <Supplier> Deny hiring transaction 72](#_Toc7065812)

[Table 27 <Supplier> Confirm equipment delivering 74](#_Toc7065813)

[Table 28: <Supplier> Confirm equipment returned 75](#_Toc7065814)

[Table 29 : <Supplier> Cancel hiring transaction 77](#_Toc7065815)

[Table 30 <Supplier> Accept request for time extension 79](#_Toc7065816)

[Table 31 <Supplier> Deny request for time extension 80](#_Toc7065817)

[Table 32 <Supplier> Post equipment 83](#_Toc7065818)

[Table 33 <Supplier> Delete equipment 85](#_Toc7065819)

[Table 34 : <Supplier> Edit equipment 86](#_Toc7065820)

[Table 35 : <Supplier> Post material 88](#_Toc7065821)

[Table 36 <Supplier> Post material 90](#_Toc7065822)

[Table 37 <Supplier> Delete material 91](#_Toc7065823)

[Table 38 <Supplier> View material transaction 93](#_Toc7065824)

[Table 39 : <Supplier> Accept pending material transaction 95](#_Toc7065825)

[Table 40 <Supplier> Deny pending material transaction 96](#_Toc7065826)

[Table 41 <Supplier> Confirm material delivering 98](#_Toc7065827)

[Table 42 <Supplier> Cancel material transaction 99](#_Toc7065828)

[Table 43 <Supplier>View debris transaction 101](#_Toc7065829)

[Table 44 <Supplier> Bid on debris post 103](#_Toc7065830)

[Table 45 <Supplier>Confirm preparing debris service 105](#_Toc7065831)

[Table 46 <Supplier>Confirm working on debris service 106](#_Toc7065832)

[Table 47 <Supplier>Cancel debris transaction 107](#_Toc7065833)

[Table 48 <Requester>View hiring transaction 110](#_Toc7065834)

[Table 49 <Requester>Confirm equipment received 112](#_Toc7065835)

[Table 50 <Requester>Request equipment 114](#_Toc7065836)

[Table 51 <Requester>Cancel hiring transaction 116](#_Toc7065837)

[Table 52 <Requester>Request to extend hiring time 118](#_Toc7065838)

[Table 53 <Requester>Cancel time extension request 119](#_Toc7065839)

[Table 54 <Requester>Return equipment 122](#_Toc7065840)

[Table 55 <Requester>Feedback hiring transaction 123](#_Toc7065841)

[Table 56 <Requester>View material transaction 125](#_Toc7065842)

[Table 57 <Requester>Send material request 127](#_Toc7065843)

[Table 58 <Requester>Cancel material transaction 129](#_Toc7065844)

[Table 59 <Requester>Confirm material received 130](#_Toc7065845)

[Table 60 <Requester>Feedback material transaction 131](#_Toc7065846)

[Table 63 <Requester>Publish debris post 133](#_Toc7065847)

[Table 64 <Requester>Edit debris post 135](#_Toc7065848)

[Table 65 <Requester>Close debris post 136](#_Toc7065849)

[Table 66 <Requester>Delete debris post 138](#_Toc7065850)

[Table 67 : <Requester>Edit debris post 139](#_Toc7065851)

[Table 68 <Requester>Accept debris bid 141](#_Toc7065852)

[Table 69 Requester>Cancel debris transaction 143](#_Toc7065853)

[Table 70 <Requester>Confirm debris transaction finished 144](#_Toc7065854)

[Table 71 <Requester>Feedback debris transaction 146](#_Toc7065855)

[Table 72 <Scheduler>Check outdate equipment 147](#_Toc7065856)

[Table 73 <Scheduler>Check subscription matched 149](#_Toc7065857)

[Table 74 <Scheduler> Train price suggestion model 151](#_Toc7065858)

[Table 75 <Staff>Verify Registered User 154](#_Toc7065859)

[Table 76 <Staff>Verify Registered User 155](#_Toc7065860)

[Table 77 <Staff>Verify Registered User 156](#_Toc7065861)

[Table 78 <Staff>Verify report 158](#_Toc7065862)

[Table 79 <Staff>Edit material 159](#_Toc7065863)

[Table 80 <Staff>Edit equipment 161](#_Toc7065864)

[Table 81 <Staff>Change password 162](#_Toc7065865)

[Table 82 <Staff>Adjust hiring transaction 164](#_Toc7065866)

[Table 83 <Staff>Adjust material transaction 165](#_Toc7065867)

[Table 84 <Staff>Adjust debris transaction 167](#_Toc7065868)

[Table 85 <Staff>Send custom notification 168](#_Toc7065869)

[Table 86 <Staff>View equipment analytics 169](#_Toc7065870)

[Table 87 : <Staff>View material analytics 171](#_Toc7065871)

[Table 88 <Staff>View debris analytics 172](#_Toc7065872)

[Table 89 <Admin>Add staff 175](#_Toc7065873)

[Table 90 <Admin>Delete staff 176](#_Toc7065874)

[Table 91 <Admin>Add equipment category 178](#_Toc7065875)

[Table 92 <Admin>Edit equipment category 179](#_Toc7065876)

[Table 93 <Admin>Migrate equipment category 181](#_Toc7065877)

[Table 94 <Admin>Delete equipment category 182](#_Toc7065878)

[Table 95 <Admin>Add equipment type 183](#_Toc7065879)

[Table 96 <Admin>Edit equipment type 185](#_Toc7065880)

[Table 97 <Admin>Delete equipment type 186](#_Toc7065881)

[Table 98 <Admin>Add equipment type additional fields 188](#_Toc7065882)

[Table 99 <Admin>Edit equipment type additional fields 189](#_Toc7065883)

[Table 100 Conceptual Diagram Dictionary 211](#_Toc7065884)

[Table 101 <Fields> Login 259](#_Toc7065885)

[Table 102 <Buttons/Hyperlinks> Login 259](#_Toc7065886)

[Table 103 <Fields> Sign Up 261](#_Toc7065887)

[Table 104 <Buttons/Hyperlinks> Sign Up 261](#_Toc7065888)

[Table 105 <Buttons/Hyperlinks> Main menu 262](#_Toc7065889)

[Table 106 <Buttons/Hyperlinks> Equipment menu 263](#_Toc7065890)

[Table 107 <Buttons/Hyperlinks> Material menu 264](#_Toc7065891)

[Table 108 <Buttons/Hyperlinks> Debris menu 264](#_Toc7065892)

[Table 109 <Buttons/Hyperlinks> User menu 265](#_Toc7065893)

[Table 110 <Fields> Equipment search 266](#_Toc7065894)

[Table 111 <Buttons/Hyperlinks> Search equipment 266](#_Toc7065895)

[Table 112 <Fields> Post new equipment step 1 267](#_Toc7065896)

[Table 113 <Buttons/Hyperlinks> Post new equipment step 1 268](#_Toc7065897)

[Table 114 <Fields> Post new equipment step 2 268](#_Toc7065898)

[Table 115 <Buttons/Hyperlinks> Post new equipment step 2 269](#_Toc7065899)

[Table 116 <Fields> Post new equipment step 3 269](#_Toc7065900)

[Table 117 <Buttons/Hyperlinks> Post new equipment step 3 269](#_Toc7065901)

[Table 118 <Fields> Post new equipment step 2 270](#_Toc7065902)

[Table 119 <Buttons/Hyperlinks> View and delete equipment 271](#_Toc7065903)

[Table 120 <Fields> Edit equipment 272](#_Toc7065904)

[Table 121 <Buttons/Hyperlinks> Edit equipment 273](#_Toc7065905)

[Table 122 <Fields> Manage equipment transactions 274](#_Toc7065906)

[Table 123 <Buttons/Hyperlinks> Manage equipment transactions 274](#_Toc7065907)

[Table 124 <Fields> Manage equipment subscriptions 275](#_Toc7065908)

[Table 125 <Buttons/Hyperlinks> Manage equipment subscriptions 276](#_Toc7065909)

[Table 126 <Fields> Login 290](#_Toc7065910)

[Table 127 <Buttons/Hyperlinks> Login 290](#_Toc7065911)

[Table 128 <Buttons/Hyperlinks> Main menu 291](#_Toc7065912)

[Table 129 <Buttons/Hyperlinks> Equipment menu 292](#_Toc7065913)

[Table 130 <Buttons/Hyperlinks> Material menu 293](#_Toc7065914)

[Table 131 <Buttons/Hyperlinks> Debris menu 294](#_Toc7065915)

[Table 132 <Buttons/Hyperlinks> Report menu 294](#_Toc7065916)

[Table 133 <Fields> Dashboard 296](#_Toc7065917)

[Table 134 <Buttons/Hyperlinks> Dashboard 296](#_Toc7065918)

[Table 135 <Fields> Manage Staff 298](#_Toc7065919)

[Table 136 <Buttons/Hyperlinks> Manage Staff 298](#_Toc7065920)

[Table 137 <Fields> Manage Registered User 300](#_Toc7065921)

[Table 138 <Buttons/Hyperlinks> Manage Registered User 300](#_Toc7065922)

[Table 139 <Fields> Manage Equipment Category 301](#_Toc7065923)

[Table 140 <Buttons/Hyperlinks> Manage Equipment Category 302](#_Toc7065924)

[Table 141 <Fields> Manage Equipment Type 303](#_Toc7065925)

[Table 142 <Buttons/Hyperlinks> Manage Equipment Type 304](#_Toc7065926)

[Table 143 <Fields> Manage Equipment 306](#_Toc7065927)

[Table 144 <Buttons/Hyperlinks> Manage Equipment 306](#_Toc7065928)

[Table 145 <Fields> Manage Hiring Transaction 308](#_Toc7065929)

[Table 146 <Buttons/Hyperlinks> Manage Hiring Transaction 308](#_Toc7065930)

[Table 147 Entity Relationship Diagram Dictionary 312](#_Toc7065931)

**List of Figures**

[Figure 1 - Scrum Framework 23](#_Toc7065932)

[Figure 2: <Guest> Overview use case 33](#_Toc7065933)

[Figure 3 <Guest> Login 33](#_Toc7065934)

[Figure 4 Conceptual Diagram 209](#_Toc7065935)

[Figure 5 System Architecture diagram 213](#_Toc7065936)

[Figure 6 Component Diagram - Registered User Web Api 215](#_Toc7065937)

[Figure 7 Component Data Dictionary 217](#_Toc7065938)

[Figure 8 Class Diagram 218](#_Toc7065939)

[Figure 9 Class Diagram Dictionary 222](#_Toc7065940)

[Figure 10 Sequence Diagram - Search equipment 242](https://d.docs.live.net/fe447eae7a2d9c8b/Capstone/reports/Report-full.docx#_Toc7065941)

[Figure 11 Sequence Diagram - Login 243](https://d.docs.live.net/fe447eae7a2d9c8b/Capstone/reports/Report-full.docx#_Toc7065942)

[Figure 12 Sequence Diagram - Check Subscription 244](https://d.docs.live.net/fe447eae7a2d9c8b/Capstone/reports/Report-full.docx#_Toc7065943)

[Figure 13 Sequence Diagram - Post equipment 245](#_Toc7065944)

[Figure 14 Sequence Diagram - Request equipment 246](#_Toc7065945)

[Figure 15 Sequence Diagram - Chang hiring status 247](https://d.docs.live.net/fe447eae7a2d9c8b/Capstone/reports/Report-full.docx#_Toc7065946)

[Figure 16 Sequence Diagram - Checkout Materials 248](#_Toc7065947)

[Figure 17 Sequence Diagram - Publish Debris Post 249](#_Toc7065948)

[Figure 18 Sequence Diagram - Bid on Debris Post 250](#_Toc7065949)

[Figure 19 Sequence diagram - Verify Registered User 255](#_Toc7065950)

[Figure 20 Sequence diagram - Add staff 256](https://d.docs.live.net/fe447eae7a2d9c8b/Capstone/reports/Report-full.docx#_Toc7065951)

[Figure 21 Sequence diagram - Add equipmen type 257](#_Toc7065952)

[Figure 22 Edit equipment type 258](https://d.docs.live.net/fe447eae7a2d9c8b/Capstone/reports/Report-full.docx#_Toc7065953)

[Figure 23 Login 259](#_Toc7065954)

[Figure 24 Entity Relation Diagram 310](#_Toc7065955)

[**Figure 25 Linear regression plot - from Wikipedia** 332](#_Toc7065956)

[**Figure 26 Gradient Descent Plot** 334](#_Toc7065957)

[Figure 27: Physical Diagram 338](#_Toc7065958)

[Figure 28: Data Dictionary 341](#_Toc7065959)

# Introduction

## Project Information

* Project name:  **Email marketing platform with Simple Email Service of Amazon Cloud**
* Project Code: **EMP**
* Product Type: **Website Application**
* Start Date: **May 13th, 2019**
* End Date: **July 24th, 2019**

## Introduction

This document introduce a solution for email marketing. At its most basic level, email marketing is the use of email to promote your business. It is used to cultivate relationships with potential customers, keep current customers informed and updated on your brand, offer coupons to encourage customer loyalty, and more.

It is a direct form of marketing, similar to marketing through snail mail, but email marketing is much more efficient for your wallet and for the environment since it’s totally paperless.

Email marketing can be used to build trust with customers over time to turn them into repeat customers. It is also an effective way to keep your customers informed about new sales or promotions you are running. People want to stay informed about your brand, and email marketing is one of the best ways to stay engaged with that audience.

Because of that we have built a software called Mindsending. Unlike others email marketing applications on the market, we not only bring every main function of the email marketing application but also help the user by supporting them the drawing of BPMN (Business Process Modeling Notation) which can be applied to the system so that they can manage and track their work flow in the easiest way.

Beside that our web system uses Amazon SES which will help you for sending a large amount of email from sender to many recipients without sending spam mails, porn mails or in-appropriate content.

## Current Situation

- Email marketing is the act of sending a commercial message, typically to a group of people, using email. Email marketing is significantly cheaper and faster than traditional mail, mainly because with email, most of the cost falls on the recipient. Email marketing carries the benefit of allowing marketers to identify returns on investment and measure and improve efficiency.

- But email marketing also has some of disadvantages: spam emails, porn emails, in-appropriate content in the message which send to the customers.

## Problem Definition

Below are disadvantages of current situation:

* **Traditional marketing costs are very high,** sales companies spend a lot of money on 30s advertising on television or on other media such as newspapers and flyers.
* **Sending traditional emails is easy to send your mail to the spam mailbox,** making the approach to customers more restrictive.
* **Others email marketing applications on the market, it is only for users to create a simple pure automated email** with a non-general look that will make it difficult for new entrants to set up an email marketing.

## Proposed Solution

Our proposed solution is to build a system called MindSending (EMP) to resolve the current situation by helping people has a more general view of the system to automatically send emails to customers by supporting the drawing of BPMN (Business Process Modeling Notation) and then apply to the system so that they can manage and track their work flow in the convenient way.

The EMP system includes two web applications with BPMN drawing tool.

### Functions

* Administrator system (Web application):
  + **Manage Report**: Admin or staff can manage, adjust information with high authority actions.
  + **Manage account**: Admin or staff can check and verify sensitive information, account profile information.
  + **Manage sever**: Admin can view overall sever and manage sever with high authority actions.
* Client applications (Web application):
  + **Manage own account:** The customer has an overview of the account information….
  + **Manage report:** The customer can view report generated from the system by date, month, year.
  + **Manage contact:** When a contact is brought into the system, the customer can add, delete, edit contact and can export contact lists.
  + **Manage group:** Customers can categorize contacts into groups to facilitate contact entry into the appropriate sending email list.
  + **Manage templates:** Customers can add a new template or edit an existing template of the system.
  + **Manage campaign:** Customers can add a new campaign or edit an existing campaign of the system.
  + **Draw BPNM model:** Customers can draw a BPNM model to set up automation campaign.

### Advantages and disadvantages

The advantages and disadvantages of the proposed solution:

* Advantages
  + **Easy for managing and review:** Managing all workflow automation email marketing through BPMN model. Customer can drag/drop object from BPMN drawing tool.
  + **Helping for using contact database effectively through contact and group system EMP:** Help customers categorize customers according to each status.
* Disadvantages:
  + **Can not satisfy all demands:** Marketing is a very flexible and complicated field and there are special business cases that the system cannot fit. Staff involvement may be required in order to resolve these special situations.

## Functional Requirements

Function requirements of the system are listed as below:

### Email marketing website

* Customer can login
* Customer can logout
* Customer can register
* Customer can view account information
* Customer can edit account information
* Customer can search for contact with filter and sorting
* Customer can manage contact in the system
* Customer can manage and change their contact status
* Customer can search for group contact with filter and sorting
* Customer can manage group contact
* Customer can import contact
* Customer can edit contact
* Customer can add new template
* Customer can edit templates suitable for work purposes
* Customer can add new BPNM model
* Customer can edit BPNM model
* Customer can view analysis report by date, month, year

### Admin website

* Admin can manage report
* Admin can manage all accounts on the system
* Admin can manage sever

## Role & Responsibility

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Full Name** | **Role** | **Position** | **Contact** |
| 1 | Lâm Hữu Khánh Phương | Project Manager | Supervisor |  |
| 2 | Ngô Minh Tấn | Developer | Leader |  |
| 3 | Nguyễn Lưu Hồng Sơn | Developer | Member |  |
| 4 | ~~Đặng Võ Anh Khoa~~ | ~~Developer~~ | ~~Member~~ |  |
| 5 | Nguyễn Hữu Thắng | Developer | Member |  |

Table 1 : Roles and Responsibilities

# Software Project Management Plan

## Problem Definition

### Name of this Capstone Project

* Official name: Email marketing platform with Simple Email Service of Amazon Cloud
* Vietnamese name: Ứng dụng khởi tạo email tiếp thị kết hợp dịch vụ email của Amazon
* Abbreviation: EMP

### Problem Abstract

This project concentrates on creating an automation email marketing. It is called Mindsending (EMP). EMP is to assist customer for managing and initiating an automation email marketing.

In marketing, planning a marketing plan that suits the needs of customers is a central issue. Use email that has been collected effectively and can be reused.

The Mindsending web application supports customer an optimal solution for that problem, providing a BPNM model for easy manage and create a workflow.

### Project Overview

#### Current Situation

Below are the problems encountered in this project:

* **Limited time and human resource.** The team only has 4 members and 13 weeks to execute the project from requirement analytics, design, implement and testing.
* **Lack of business view.** Team members’ major is Software engineering, leading to little experience in general business fields, especially in the construction business field. Therefore, research and training effort would be made to adjust and improve the business model.
* **Lack of experience**. Team members do not have much experience in applying new technology into real projects. There would be new probleEMP arise that require research time and expertise.
* **Schedule conflict.** Team members have different schedule arrangement and lack of co-working time, leading to lack of synchronization of the team.
* **Lack of UX/UI skills.** As developers, the team do not have much knowledge in graphics design and user experience designing, which is a critical factor for a successful C2C product (customer-to-customer product)

#### The Proposed System

According to the recent trend in e-commerce, email marketing is a promising field that can solve access customer’s problem. Therefore, applying that model with necessary modification would be a possible solution.

The basic idea is that we create an email marketing platform with Simple Email Service of Amazon Cloud. We provide a website for end user (customer) to use the system. We also have an admin website to control the system with high authority.

#### Boundaries of the System

* The language of the system is English
* (pending)

#### Future Plans

* Current system can only support admin to manage equipment. We design the system to provide analytic views for admin:
  + Admin can view statistics and visualized graph about transaction process
  + Admin can view strategic information about equipment trends, hiring trends, Registered User needs trends
* Current system can only suggest price to Registered User based on equipment type. We design the system to support data for further machine learning features:
  + Suggest equipment price based on equipment name, location, time
* Current system can only provide delivery status to delivery management. We design the system to support more feature to delivery phase:
  + Support direction and route guidance for delivery
  + Support delivery time estimate based on city, time, traffic

#### Development environment

##### Hardware requirements

For server

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

Table 2 - Hardware Requirements for Server

For PC

|  |  |  |
| --- | --- | --- |
| PC | Minimum Requirements | Recommended |
| Internet Connection | Cable, Wi-Fi (4 Mbps) | Cable, Wi-Fi (8 Mbps) |
| Operating System | Window 7 | Window 7 or more. |
| Computer Processor | Intel® Core i3 1.4GHz | Intel® Core i5 2.50GHz |
| Computer Memory | 1GB RAM | 2GB RAM or more |
| Web Browser | Chromes (v42 or higher) | Chrome latest stable version |

Table 3 Hardware requirement for PC

##### Software requirement

|  |  |  |
| --- | --- | --- |
| Software | Name / Version | Description |
| Operation System | Windows Server 2014 | Operating system and platform for development |
| Environment | Java EE 8.0, Node v10, npm v6 | Specification for developing web application |
| Modeling tool | StarUML, Draw.io | Used to design diagram |
| IDE | Intellij IDEA 2019.1.5, Visual Studio Code 1.27.2 | Programming tools |
| DBEMP | MYSQL 8.0 | Used to create & manage the database for system |
| Source control | Git (Github server) | Used for source control |
| Web browser | Chrome 69 or above | Testing browser |

Table 4 Development Environment - Software Requirement

## Project Organization

### Software Process Model

This project is developed using Scrum model – part of an agile framework for Software development project. Scrum model is chosen because of the following reasons:

* Scrum is the most suitable model for small and medium project, and we only have 4 members
* The term “machine learning” is new to the team and there would be many trial-and-error tests with the system design before applying to the official release
* There are many new technologies that team members have to learn and develop at the same time, so SCRUM would allow us to do that
* The business model may not be fully tested in real market condition so changes are highly possible. Using Scrum team members can adapt to changes better

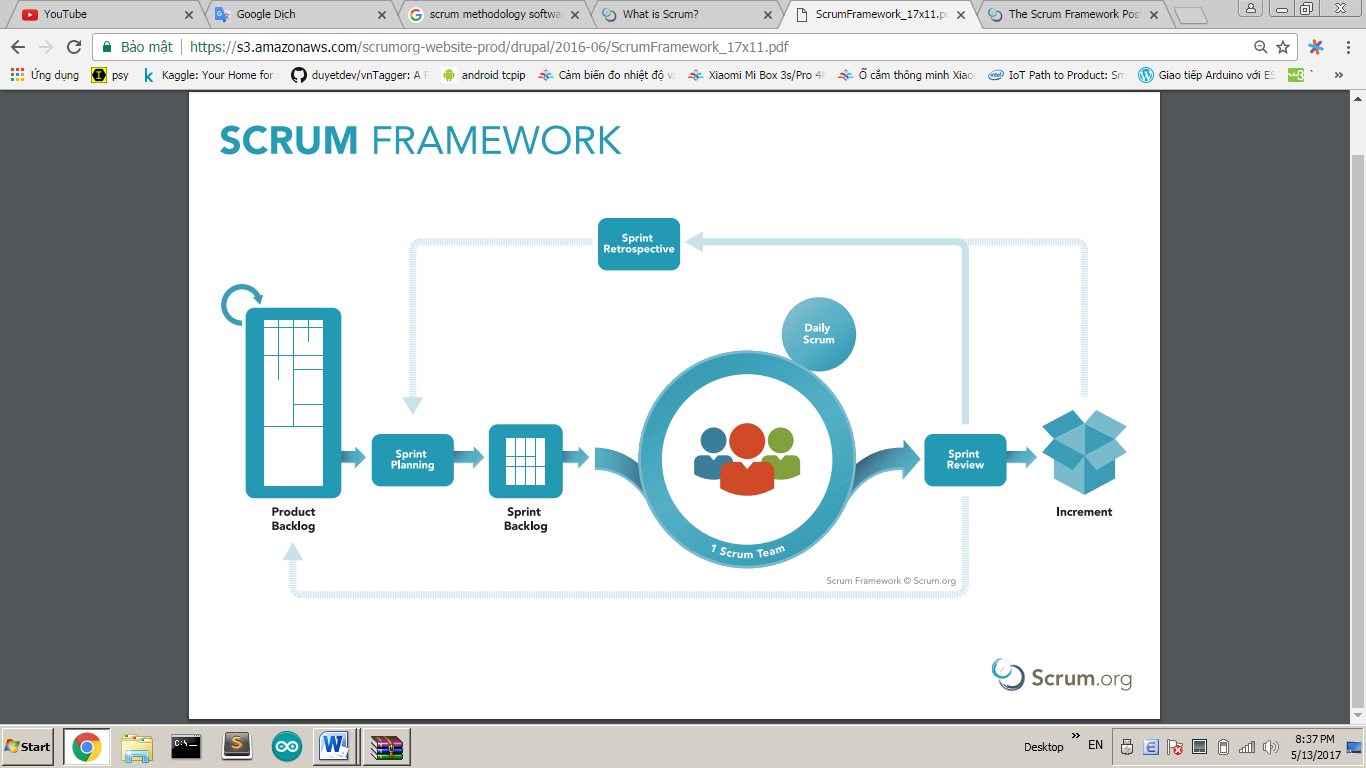


Figure 1 - Scrum Framework

### Roles and responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| No | Full name | Role in Group | Responsibilities |
| 1 | Ngô Minh Tấn | Product Owner | * Specify user requirement * Control the development process * Give out technique and business analysis support |
| 2 | Nguyễn Hữu Thắng | Scrum Master | * Managing project * Designing databases * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing * Arrange Meeting * Risk Management |
| 3 | Nguyễn Lưu Hồng Sơn | Scrum team member | * Clarifying requirements * Preparing documents * Mobile GUI Design * Coding * Testing |

Table 6 Roles and responsibilities

### Tools and Techniques

|  |  |
| --- | --- |
| Tool/Technique | Name |
| Front-end | HTML, CSS, JavaScript, jQuery, ReactJs, React Native |
| Back-end | JavaEE, Spring, JPA, Hibernate |
| IDE | IntelliJ IDEA 17.2, Visual Studio Code |
| DBEMP | MySQL |
| Source Control | Git, Github |
| Modelling tool | StarUML, Draw.io |

Table 7 Tools and techniques

## Project Management Plan

### Product Backlog

Product Backlog could be found [here.](https://drive.google.com/open?id=1hJ2loRGJDPmh-D2Zm1oSJTu9u1qBt99ybJAj1JLJSt0)

### Sprint Backlog

Sprint Backlog can be found [here](https://docs.google.com/spreadsheets/d/1ei8C6OGqnKYWK0I5HX1T0Bcu-IQfVVulU_9_QmHVhEY/edit?usp=sharing)

### Deliverables

|  |  |  |
| --- | --- | --- |
| No | Deliverable | Note |
| 1 | Introduction, Entity Relationship Diagram, Business model detail, | Sprint 1 |
| 2 | Use case Overview, UI for mobile application, Conceptual Diagram, First deployment | Sprint 2 |
| 3 | Class Diagram, Physical Diagram, UI for Website Admin, Implement Search equipment | Sprint 3 |
| … |  |  |

Table 8 Deliverables

### All Meeting Minutes

All sprint meeting minutes could be found [here](https://drive.google.com/open?id=1LIwLFeSkVkFuURBqeYgy2I-vJnobL-u3).

## Coding Convention

Summary:

* Naming Convention:
  + Variable names should be short yet meaningful. The choice of a variable name should be designed to indicate to the casual observer the intent of its use.
  + Methods should be verbs, in mixed case with the first letter lowercase, with the first letter of each internal word capitalized.
* Indentation:
  + One declaration per line is recommended since it encourages commenting.
  + In absolutely no case should variables and functions be declared on the same line.
  + Do not put different types on the same line.
* Declarations Convention:
  + One declaration per line is recommended since it encourages commenting.
  + Using Java Code Convention from:

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

# Software Requirement Specification

## User Requirement Specification

### Guest Requirement

***Guest*** *is a person who doesn’t have access to the system. Guest can use some*

*functions in the system. To use all functions, guest must login. These are some*

*functions that guest can use*:

* Login.
* Register.

### User requirement

***User*** *is a guest who logged into the system with user’s role. There are some functions that user can use:*

* Manage own account
* Manage report:

-View report

-Analysis report

* Manage contact:

-Add contact

-Delete contact

-Edit information of contact

-Import list contact

* Manage group:

-Add contact to group

-Delete group

-Edit group

-Search group

-Create group

* Manage templates:

-Add formatted template

-Edit inside template

* Manage campaign:

-Add campaign

-Edit campaign

-Delete campaign

* Draw BPNM model:

-Add BPNM

-Edit BPNM

-Delete BPNM

* View embed form

-Copy script

* Manage invite mail

### Admin requirement

***Admin*** *is an authorized user has permission to log in to the system. An admin has higher authority than user and can do whatever a user can. In addition, an admin can do the following functions:*

* Admin can manage account
* Admin can manage report
* Admin can manage sever

## System Requirement Specification

### External Interface Requirement

#### User Interface:

* GUI should be simple, clear, intuitive, and reminiscent.
* The interface design is an iterate process includes: design, sketching, prototyping, user assessment.
* The language for User Interface is English

#### Hardware Interface

* Mobile Device
* Desktop PC or Laptop

#### Software Interface

* Web application: work with browsers Firefox (v52 or higher), Chromes (v28 or higher), Internet Explorer (v10 or above) or with any web browser that supports HTML5 & CSS3.
* Mobile application: iOS (9.0 or higher)

#### Communication Protocol

* Use HTTP protocol 1.1 for communication between the web browser and the web server.

### 2.2. System Overview Use Case

Figure 1: System overview use case



### List of use case

#### <Guest> Overview Use Case

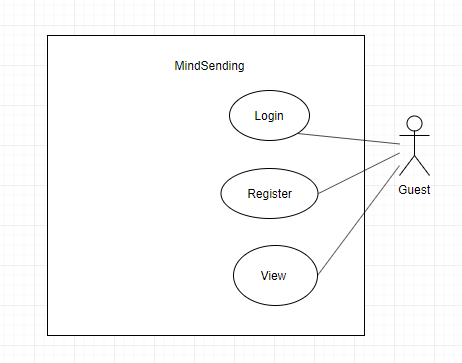


Figure 2: <Guest> Overview use case

##### <Guest> Login

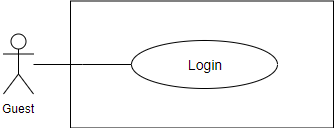


Figure 3 <Guest> Login

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_01** | | | |
| **Use Case No.** | **EMP\_UC\_01** | Use Case Version | 1.1 |
| **Use Case Name** | Login | | |
| **Author** | Nguyễn Hữu Thắng | | |
| **Date** | 15/07/2019 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use case allows user to login to the system.   **Goal:**   * User can login to the system   **Trigger:**   * User send the login command. * User clicks on “Get Started” button   **Precondition:**   * User is viewing the login screen   **Post Conditions:**   * Success: User login to the system successfully * Fail: Show error messages   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User go to the login view | The system requires the following input from User includes:   * Username: text, required * Password: text, required | | 2 | User inputs information. |  | | 3 | User sends command to login | The system show message that the user has logged in successfully  [Exception 1] |   **Exception 1:**   |  |  |  | | --- | --- | --- | | **Step** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. |   **Relationship:** N/A.  **Business rule:**   * User password character is secured and hidden with special character * The system generates an access token * The system generates a refresh token * System can use access token to verify identity without reenter username and password again * Refresh token can be used to retrieve new access token with no username or password needed * Access token have expiration time of 15 mins * Refresh token has expiration time of 90 days * Refresh token and access token is signed with RSA256 algorithm * Refresh token and access token also contain at least following information:   + Username   + User id   + Expire time   + Created time   + Signature for security * User is required a new refresh token if password changed or refresh token is expired | | | |

Table 1 <Guest> Login

##### <Guest> Register

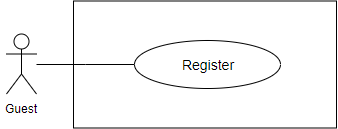


Figure 4: <Guest> Register

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - EMP\_UC\_02** | | | |
| **Use Case No.** | EMP\_UC\_02 | Use Case Version | 1.1 |
| **Use Case Name** | Register | | |
| **Author** | Nguyễn Hữu Thắng | | |
| **Date** | 15/07/2019 | **Priority** | Normal |
| **Actor**:   * Guest   **Summary**:   * This use case allows user to register to the system.   **Goal**:   * User can register to the system   **Trigger**:   * User send the register command. * User clicks “Sign up”   **Precondition**:   * User is viewing the register screen   **Post Conditions:**   * Success: User register to the system successfully * Fail: Show error messages   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User go to the guess view | The system requires the following input from User includes:   * Username: text, required * Password: text, required * Confirm password: text, required * Name: text, required * Email: text, required * Phone: text as number, required * Profile image: image, required | | 2 | User inputs information. |  | | 3 | User sends the command to register | The system show message that the user has registered successfully  [Exception 1] |   Exception 1:   |  |  |  | | --- | --- | --- | | **Step** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. |   **Relationship**: N/A.  **Business** rule:   * User password must not be saved directly plain text, and must be hashed * The system validates user input with the following constraints:   + Username: unique   + Phone number: number format   + Email: as an email format   + The confirm password matches the password | | | |

Table 2 <Guest> Register

#### <Registered User> Overview Use Case

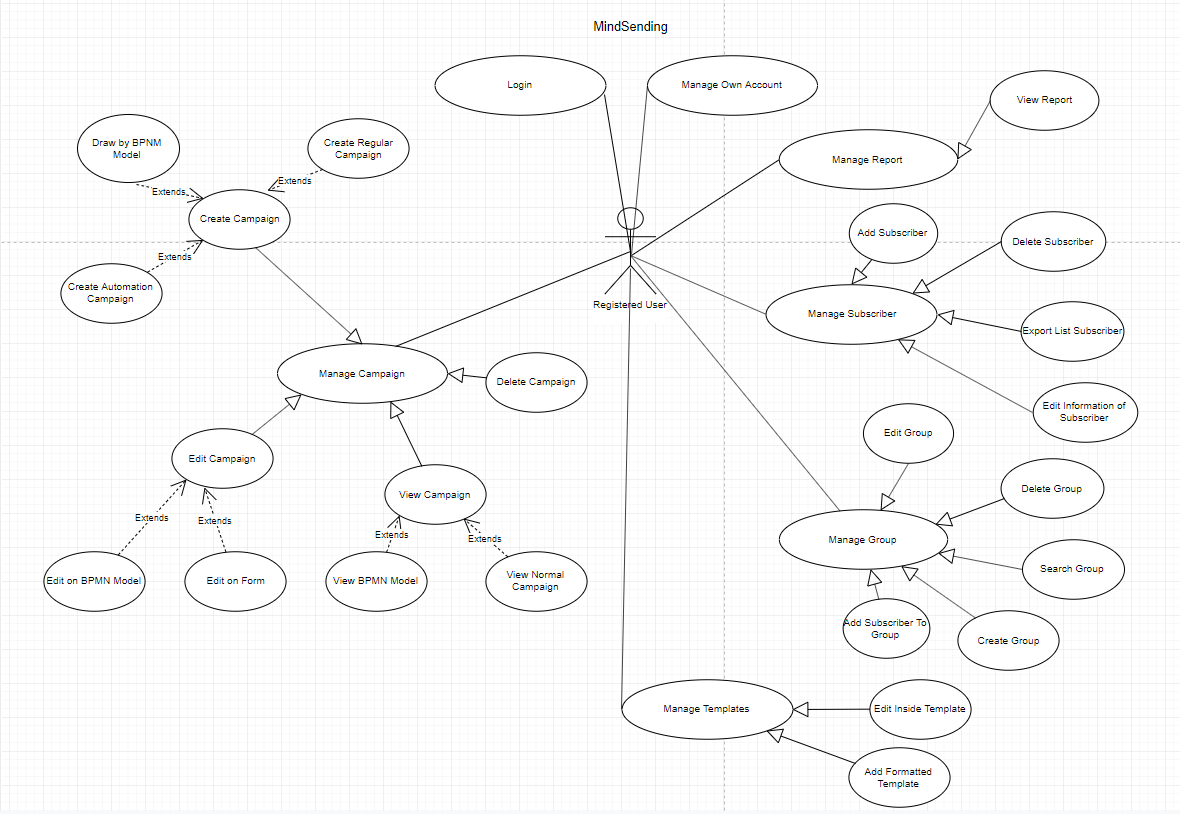


Figure 8: <Registered User>Overview use case

##### <Registered User> Logout

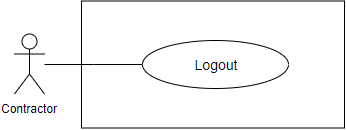


Figure 9: <Registered User> Logout

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - EMP\_UC\_03** | | | |
| **Use Case No.** | **EMP\_UC\_03** | **Use Case Version** | 1.1 |
| **Use Case Name** | Logout | | |
| **Author** | Nguyễn Hữu Thắng | | |
| **Date** | 15/07/2019 | **Priority** | Normal |
| **Actor:**   * Registered User   **Summary:**   * This use case allows user to logout to the system.   **Goal:**   * User can logout from the system   **Trigger:**   * User send a logout command.   **Precondition:**   * User is viewing the account screen   **Post Conditions:**   * Success: User logout to the system successfully * Fail: Show error messages   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User go to the account view |  | | 2 | User send a logout command. | The system show message that the user has logged out successfully  [Exception 1] |   **Exception 1:**   |  |  |  | | --- | --- | --- | | **Step** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. |   **Relationship:** N/A.  **Business rule:**   * After logged out, user access the system as role “Guest” | | | |

##### <Registered User> Manage Own Account

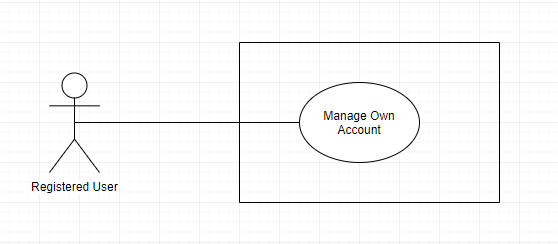


Figure 16: <Registered User> Manage User Profile

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - EMP\_UC\_04** | | | |
| **Use Case No.** | **EMP\_UC\_04** | **Use Case Version** | 1.0 |
| **Use Case Name** | Manage User Profile | | |
| **Author** |  | | |
| **Date** | 15/07/2019 | **Priority** | Normal |
| **Actor:**   * Registered User   **Summary:**   * This use case allows user to manage own account   **Goal:**   * User can edit their own account   **Triggers:**   * User clicks “User profile”. * User send manage account command to the system.   **Preconditions:**   * User must login into the system with role Registered User   **Post conditions:**   * Success: The profile successful updated. * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User goes to manage account view | The system requires input from user includes:   * Username: text, * First Name: text, * Last Name:text, * Email Address: text, * City: text, * Profile image: image, * Country: text, * Information: text, * Password: text, password | | 2 | User inputs information. |  | | 3 | User send edit profile command. | System shows message profile has been edited successfully.  [Exception 1]  [Exception 2] |   **Exception 1:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | Edit profile fail. (Input contain special characters) | The system shows an error message “Input may not contain special characters” |   **Exception 2:**   |  |  |  | | --- | --- | --- | | **Step** | **Cause** | **System Response** | | 1 | When the connection to the server is lost | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A.  **Business Rules:**   * After editing profile, Registered User must be verified again by staff * Registered User’s status change into “Not verified” | | | |

Table 21 <Registered User> Edit profile

##### <Registered User> Manage Report

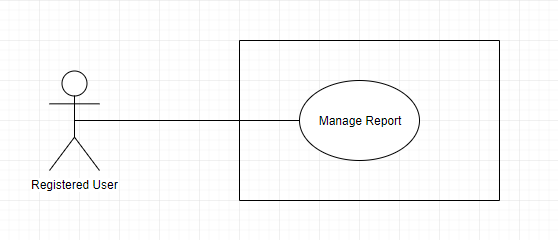


Figure 17: <Register User> Manage Report

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - EMP\_UC\_05** | | | |
| **Use Case No.** | **EMP\_UC\_05** | **Use Case Version** | 1.0 |
| **Use Case Name** | Manage report | | |
| **Author** |  | | |
| **Date** | 16/07/2019 | **Priority** | Normal |
| **Actor:**   * Registered User   **Summary:**   * This use case allows user to view report   **Goal:**   * User can view the report by week, month   **Triggers:**   * User send manage report command. * User clicks on ”Dashboard”   **Preconditions:**   * User must login into the system with role Registered User.   **Post conditions:**   * Success: The report display on screen * Fail: The report cannot displayed on screen   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User goes to manage report | System display the report by date, month, year |   **Exception:**   |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 2 | When the connection to the server is lost | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A.  **Business Rules:**   * User can view the report of sent email, email pending by week, month * User can analyses the report | | | |

Table 22 <Registered User> Manage Report

##### <Registered User> Manage Contact

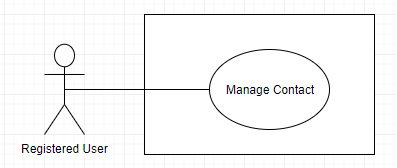


Figure 18: <Registered User> Manage Contact

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - EMP\_UC\_6** | | | |
| **Use Case No.** | **EMP\_UC\_6** | **Use Case Version** | 1.0 |
| **Use Case Name** | Manage contact | | |
| **Author** |  | | |
| **Date** | 16/07/2019 | **Priority** | Normal |
| **Actor:**   * Registered User   **Summary:**   * This use case allows registered user to manage contacts   **Goal:**   * User can view all the list of contact   **Triggers:**   * User click “View contact” to see the contact list   **Preconditions:**   * User must login into the system with role Registered User.   **Post conditions:**   * Success: List all contact. * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User sends command to view the contact list | * The system show list of all contact [Exception 1] |   **Exception 1:**   |  |  |  | | --- | --- | --- | | **Step** | **Cause** | **System Response** | | 1 | When the connection to the server is lost | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A.  **Business Rules:**   * User can view can view all the contact related to him, created in the past or the present * Each transaction has one of the following statuses at a time:   + New subscriber: contact is newly created   + Subscriber: contact is added for a time * User can see the following information about each contact:   + Email   + First name   + Last name   + Phone number   + Address   + Date of birth | | | |

Table 23 <Registered User> Manage contact

##### <Registered User> Add contacts

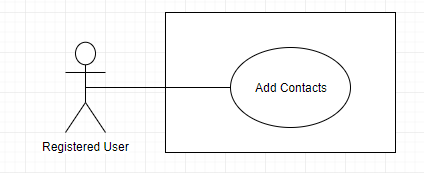


Figure 19: <Registered User> Add Contacts

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - EMP\_UC\_7** | | | |
| **Use Case No.** | **EMP\_UC\_7** | **Use Case Version** | 2.0 |
| **Use Case Name** | Add contacts | | |
| **Author** |  | | |
| **Date** | 14/06/2019 | **Priority** | Normal |
| **Actor:**   * Registered User   **Summary:**   * This use case allows Registered User to add a contact   **Goal:**   * User successfully add contacts   **Triggers:**   * User send add contact command to the system. * User clicks on “Manual add”   **Preconditions:**   * User must login into the system with role Registered User.   **Post conditions:**   * Success: New contact will be add to the default list * Fail: Show error messages   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User goes to add contact screen | Display a table show all contacts with information:   * Email Address : @email.com * Status: * First Name * Last Name * Action | | 2 | User choose add contacts button | Display two option:   * Upload CSV file * Manual Add | | 3 | User chooses to upload CSV file | * Display two options: * Add contacts * Add contacts and include in an existing list   Choose CSV file to upload | | 4 | User chooses manual add | * Display three option: * Add contacts * Add contacts and include in an existing list | | 4.1 | User fill in contact info | * Display six field to fill * First name * Last name * Email * Phone number * Address * Date of birth | | 4.2 | User add more contact | * System add more six field   [Exception 1] | | 4.3 | User save contact | * System save new contact   [Exception 1]  [Exception 2] |   **Exception 1:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | When the contact already in the list | System shows message the "This email is existed" when the contact already added to the list. |   **Exception 2:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 2 | When the connection to the server is lost | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A.  **Business Rules:** | | | |

Table 24 <Registered User> Add Contacts

##### <Registered User> Delete contacts

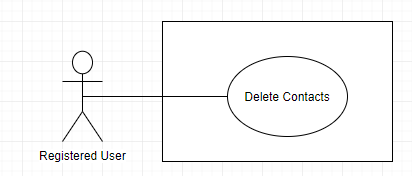


Figure 20: <Registered User> Delete contacts

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_8** | | | |
| **Use Case No.** | **EMP\_UC\_8** | **Use Case Version** | 2.0 |
| **Use Case Name** | Delete contacts | | |
| **Author** |  | | |
| **Date** | 16/07/2019 | **Priority** | Normal |
| **Actor:**   * Authenticated Registered User   **Summary:**   * This use case allows User to delete contacts.   **Goal:**   * User successfully delete contacts in the system.   **Triggers:**   * User send delete command to the system.   **Preconditions:**   * User must login into the system with role Registered User.   **Post conditions:**   * Success: contacts is removed by user. * Fail: Show error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User goes to manage contacts screen. | Display a table show list of contacts. | | 2 | User send delete contacts command. | System start removing contacts.  [Exception 1]  [Exception 2]  System show message remove successful |   **Exception 1:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | Removed fail. | Show message to notify User that remove failed. |   **Exception 2:**   |  |  |  | | --- | --- | --- | | **Step** | **Cause** | **System Response** | | 1 | When the connection to the server is lost | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A  **Business Rules:**   * In case of success scenarios, contacts will be removed from the system. | | | |

Table 25 <Registered User> Delete contacts

##### <Registered User> Mange Templates

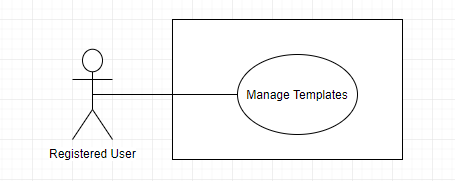


Figure 21: <Registered User> Manage Templates

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_9** | | | |
| **Use Case No.** | **EMP\_UC\_9** | **Use Case Version** | 2.0 |
| **Use Case Name** | Manage templates | | |
| **Author** |  | | |
| **Date** | 16/07/2019 | **Priority** | High |
| **Actor:**   * Registered User   **Summary:**   * This use case allows registered user to manage templates   **Goal:**   * User can view all the list of templates   **Triggers:**   * User click to templates to see gallery   **Preconditions:**   * User must login into the system with role Registered User.   **Post conditions:**   * Success: List all templates. * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User goes to gallery | System show a list of Custom templates and Mindsending templates  [Exception 1] |   **Alternative Scenario: N/A**  **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A  **Business Rules:**   * User can view all the templates in the system: * Custom templates * Mindsending templates * User can preview template before choose * User can filter template by: | | | |

Table 26 <Registered User> Manage templates

##### <Registered User> Create new template

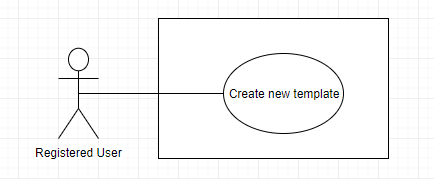


Figure 22: <Registered User> Create new template

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_10** | | | |
| **Use Case No.** | **EMP\_UC\_10** | **Use Case Version** | 2.0 |
| **Use Case Name** | Create new template | | |
| **Author** |  | | |
| **Date** | 16/07/2019 | **Priority** | High |
| **Actor:**   * Registered user   **Summary:**   * This use case allows user to create new template   **Goal:**   * User can create new template   **Triggers:**   * User send command to create new template   **Preconditions:**   * User must login to the system with role Registered User   **Post conditions:**   * Success:   + New template will be added to the system * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User send the command to create new template | System suggest two option:   * Export HTML * Load HTML | | 2 | User save template | System show message “Đã thêm thành công”  [Exception ] |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A  **Business Rules:**   * User can review the template before save. * User can export HTML or load HTML on device to system | | | |

Table 27<Supplier> Accept pending hiring transaction

##### <Registered User> Edit template

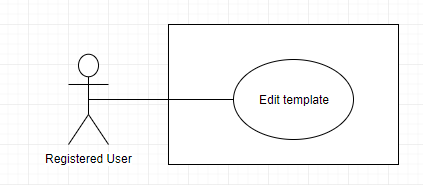


Figure 23: <Registered User> Edit template

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_11** | | | |
| **Use Case No.** | **EMP\_UC\_11** | **Use Case Version** | 1.1 |
| **Use Case Name** | Edit template | | |
| **Author** |  | | |
| **Date** | 16/07/2019 | **Priority** | High |
| **Actor:**   * Registered User   **Summary:**   * This use case allows user to edit template   **Goal:**   * User edit template successfully   **Triggers:**   * User send command to edit template   **Preconditions:**   * User must login to the system   **Post conditions:**   * Success: Template edit successfully * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User view template before edit | System show template on screen | | 2 | User edit template |  | | 3 | User save edited template | System show message “ Template edited successfully” |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A  **Business Rules:**   * User can view the template before edit | | | |

Table 28 <Registered User> Edit template

##### <Registered User> Manage Lists

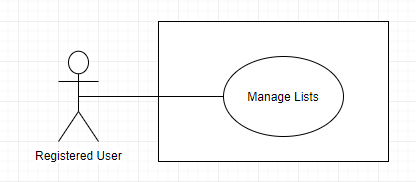


Figure 24: <Registered User> Manage Lists

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_12** | | | |
| **Use Case No.** | **EMP\_UC\_12** | **Use Case Version** | 1.0 |
| **Use Case Name** | Manage Lists | | |
| **Author** |  | | |
| **Date** | 16/07/2019 | **Priority** | Normal |
| **Actor:**   * Registered User   **Summary:**   * This use case allows user to manage lists   **Goal:**   * User can manage lists   **Triggers:**   * User click into lists   **Preconditions:**   * User must login to the system   **Post conditions:**   * Success: Contact lists display on screen * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User goes to lists screen | Display a table show all lists with information:   * List name: * Description: * Contacts: * Actions: |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A  **Business Rules:**   * By this action, user can view all contact lists. | | | |

Table 29 <Registered User> Manage lists

##### <Registered User> Create Lists

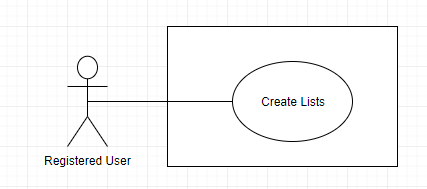


Figure 25: <Registered User> Create Lists

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_13** | | | |
| **Use Case No.** | **EMP\_UC\_13** | **Use Case Version** | 1.0 |
| **Use Case Name** | Create lists | | |
| **Author** |  | | |
| **Date** | 16/07/2018 | **Priority** | Normal |
| **Actor:**   * Registered user   **Summary:**   * This use case allows user to create lists   **Goal:**   * User create lists successfully   **Triggers:**   * User send command to create list   **Preconditions:**   * User must login to the system   **Post conditions:**   * Success:   + The list added succesfully * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User send the command to create list | System show create list window | | 2 | User fill list name and select list to add |  | | 3 | User click save list | System show message “Đã thêm thành công” |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A  **Business Rules:** | | | |

Table 30: <Supplier> Confirm equipment returned

##### <Registered User> Search Lists

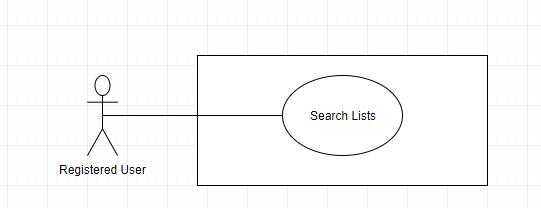


Figure 26: <Registered User> Search Lists

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_14** | | | |
| **Use Case No.** | **EMP\_UC\_14** | **Use Case Version** | 1.1 |
| **Use Case Name** | Search Lists | | |
| **Author** |  | | |
| **Date** | 16/07/2019 | **Priority** | Normal |
| **Actor:**   * Registered User   **Summary:**   * This use case allows user to search lists by typing keyword   **Goal:**   * User search lists   **Triggers:**   * User send command to search lists   **Preconditions:**   * User must login to the system * The keyword must be included   **Post conditions:**   * Success: User can search list * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User send command to search lists | System ask user to input keyword to search | | 2 | User input the keyword | System show the lists that match keyword |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | User input wrong keyword | System donot show anything |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A  **Business Rules:**   * The keyword must be include @ to search email | | | |

Table 31 : <Registered User> Search Lists

##### <Registered User> Delete Lists

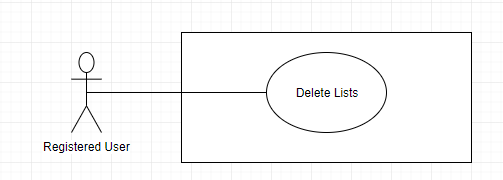


Figure 27: <Registered User> Delete Lists

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_15** | | | |
| **Use Case No.** | **EMP\_UC\_15** | **Use Case Version** | 1.0 |
| **Use Case Name** | Delete Lists | | |
| **Author** |  | | |
| **Date** | 16/07/2019 | **Priority** | High |
| **Actor:**   * Registered user   **Summary:**   * This use case allows user to delete lists   **Goal:**   * User delete lists successfully   **Triggers:**   * User send command to delete lists   **Preconditions:**   * User must login to the system   **Post conditions:**   * Success:   + The request status changed to “Accepted”   + Related hiring transaction end date changed * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User send the command to delete lists | System ask user to confirm the action | | 2 | User choose to confirm the accept action [Alternative 1] | System show message that the request is approved  [Exception 1] |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A  **Business Rules:** | | | |

Table 32 <Registered User> Delete Lists

##### <Registered User> Manage campaigns

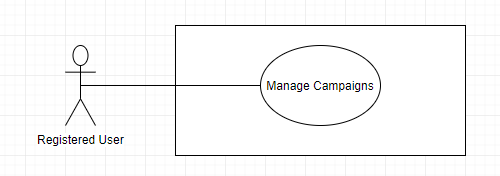


Figure 28: <Registered User> Manage Campaigns

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_16** | | | |
| **Use Case No.** | **EMP\_UC\_16** | **Use Case Version** | 1.0 |
| **Use Case Name** | Manage Campaigns | | |
| **Author** |  | | |
| **Date** | 17/06/2019 | **Priority** | High |
| **Actor:**   * Registered User   **Summary:**   * This use case allows user to manage campaigns   **Goal:**   * User can manage all campaigns   **Triggers:**   * Supplier send command to show all campaigns   **Preconditions:**   * User must login to the system   **Post conditions:**   * Success: System will show all campaigns on screen * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User send the command to show all campaigns | System show all campaigns |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. |   **Relationships:** N/A  **Business Rules:** | | | |

Table 33 <Registered User> Manage Campaigns

##### <Registered User> Create regular campaigns

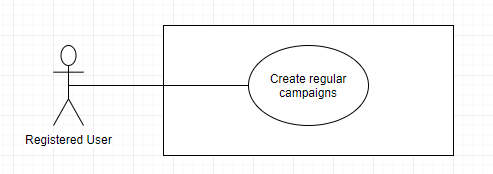


Figure 29: <Registered User> Create regular campaigns

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_17** | | | |
| **Use Case No.** | **EMP\_UC\_17** | **Use Case Version** | 1.1 |
| **Use Case Name** | Create regular campaigns | | |
| **Author** |  | | |
| **Date** | 16/07/2019 | **Priority** | High |
| **Actor:**   * Registered User   **Summary:**   * This use case allows user to create regular campaigns   **Goal:**   * User can create regular campaigns   **Triggers:**   * User send command to create regular campaigns   **Preconditions:**   * User must login to the system   **Post conditions:**   * Success: New campaigns is added * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User goes to regular campaigns | System show all campaigns and campaign statistics   * Status * Campaign Name * Clicks * Opens * Unsubscribe | | 2 | User click to create campaign | System show message to confirm create campaigns | | 3 | User input campaigns information | System show campaigns information to input:   * To * From * Subject * Content | | 4 | User confirm the action  [Alternative 1] | System display successful message  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | User input |  |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. | | 2 | User input invalidate information. | System show error message for the input mistake |   **Relationships:** N/A  **Business Rules:** | | | |

Table 34 <Registered User> Create Regular Campaigns

##### <Registered User> Create Automation Campaigns

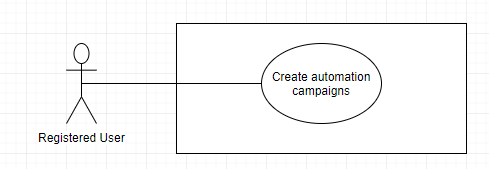


Figure 30: <Registered User> Create automation campaigns

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_18** | | | |
| **Use Case No.** | **EMP\_UC\_18** | **Use Case Version** | 1.1 |
| **Use Case Name** | Create automation campaigns | | |
| **Author** |  | | |
| **Date** | 16/07/2019 | **Priority** | Normal |
| **Actor:**   * Registered User   **Summary:**   * This use case allows user to create automation campaigns   **Goal:**   * User can create automation campaigns   **Triggers:**   * User send command to create automation campaigns   **Preconditions:**   * User must login to the system   **Post conditions:**   * Success: The automation campaigns is created * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User send the command to create automation campaigns | System show information to input | | 2 | User choose to confirm the approve action  [Alternative] | System show message that the campaign is created |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | User choose not to confirm the action | System hides the message and does nothing |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. | | 2 | The condition of the equipment is not valid to delete | System shows message about the invalid condition |   **Relationships:** N/A  **Business Rules:** | | | |

Table 35 <Registered User> Create automation campaigns

##### <Registered User> Edit Campaigns

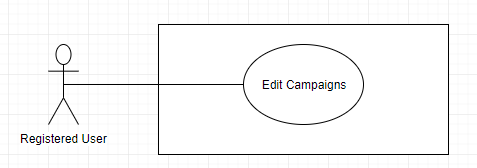


Figure 31: <Registered User> Edit Campaigns

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_19** | | | |
| **Use Case No.** | **EMP\_UC\_19** | **Use Case Version** | 1.0 |
| **Use Case Name** | Edit Campaigns | | |
| **Author** |  | | |
| **Date** | 18/07/2019 | **Priority** | Normal |
| **Actor:**   * Registered User   **Summary:**   * This use case allows user to edit campaigns   **Goal:**   * User can edit campaigns   **Triggers:**   * User send command to edit campaigns   **Preconditions:**   * User must login to the system   **Post conditions:**   * Success: The campaigns is edited * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User go to editing campaigns | System requires information about campaigns to edit, which contains:   * To * From * Subject * Content | | 2 | User input information |  | | 3 | User send command to save new campaign | System validate information, display request for confirmation | | 4 | User confirm the action  [Alternative 1] | System display successful message  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | User choose not to confirm the action | System hides the message and go back to inputting view |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. | | 2 | User input invalidate information. | System show error message for the input mistake |   **Relationships:** N/A  **Business Rules:** | | | |

Table 36 : <Registered User> Edit Campaigns

##### <Registered User> Delete Campaigns



Figure 32: <Registered User> Delete Campaigns

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_20** | | | |
| **Use Case No.** | **EMP\_UC\_20** | **Use Case Version** | 1.1 |
| **Use Case Name** | Delete Campaigns | | |
| **Author** |  | | |
| **Date** | 18/07/2019 | **Priority** | High |
| **Actor:**   * Registered User   **Summary:**   * This use case allows user to delete campaigns   **Goal:**   * User can delete campaigns   **Triggers:**   * User send command to delete campaigns   **Preconditions:**   * User must login to the system * Account of user must be activated   **Post conditions:**   * Success: Campaigns is deleted * Fail: Show error message   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User go to campaign view | System show all campaigns on screen | | 2 | User choose campaign to delete |  | | 3 | User send command to delete campaigns | System delete campaign to be choosen | | 4 | User confirm the action  [Alternative 1] | System display a successful message  [Exception 1] |     **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | No internet connection | System shows message the "Please check your connection" when the internet is lost. | | 2 | User input invalidate information. | System show error message for the input mistake |   **Relationships:** N/A  **Business Rules:** | | | |

Table 37 : <Registered User> Delete Campaigns

#### <Admin guest> Overview Use Case

Figure 109: <Admin guest>Overview use case

##### <Admin guest> Login to web admin

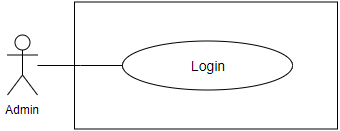


Figure 110: <Admin guest>Login to web admin

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_102** | | | |
| **Use Case No.** | **EMP\_UC\_102** | **Use Case Version** | 1.1 |
| **Use Case Name** | Login to web admin | | |
| **Author** |  | | |
| **Date** |  | **Priority** | Normal |
| **Actor:**   * Admin guest   **Summary:**   * This use case allows guest to log in the system.   **Goal:**   * Guest can log in the system with specific role.   **Triggers:**   * Guest sends login command.   **Preconditions:** N/A  **Post conditions:**   * Success: Guest is exactly authorized with specific role in the system. * Fail: System shows error messages.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | Guest go to login screen. | Login screen is shown with following labels and fields:   * Username: text input, required. * Password: text input, required. | | 2 | Guest inputs information. |  | | 3 | Guest sends command to login to system. | Guest logins system with his specific role. [Alternative 1] [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | Guest enter wrong identity information. | System shows error message “Wrong username or password. Please try again.” |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | Login failed. | Show message to notify admin that login failed. |   **Relationships:** N/A  **Business Rules:**   * Password is encrypted before being sent to server. | | | |

##### <Admin guest> Forgot your password

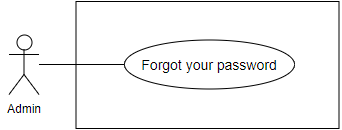


Figure 111: <Admin guest>Forgot your password

**Use Case Specification**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EMP\_UC\_103** | | | |
| **Use Case No.** | **EMP\_UC\_103** | **Use Case Version** | 1.1 |
| **Use Case Name** |  | | |
| **Author** |  | | |
| **Date** |  | **Priority** | Normal |
| **Actor:**   * Admin guest   **Summary:**   * This use case allows guest to reset password by email.   **Goal:**   * Guest can reset password by email.   **Triggers:**   * Guest sends reset password command.   **Preconditions:** N/A  **Post conditions:**   * Success: A staff password successful update in database and send to staff’s email. * Fail: A staff password cannot be updated.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | Guest go to forgot your password screen. | Forgot your password screen is shown with following labels and fields:   * Email: text input, required. | | 2 | Guest inputs information. |  | | 3 | Guest sends command to reset password to system. | Update staff password in the system.  [Alternative 1]  [Exception 1]  System shows message reset password has been updated successfully |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | Guest enter wrong identity information. | System shows error message” Wrong email.” |   **Exceptions:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | Reset password failed. | Show message to notify admin that reset password failed. |   **Relationships:** N/A  **Business Rules:**   * Password is encrypted before being sent to server. * Email have been registered before. | | | |

## Software System Attribute

### Usability

Website’s UI is fit for each browser on each device

* Font style: Arial, Sans-serif
* Font size: 12px -26px
* Color: green, black, red, white, blue, Light Slate Grey, Fuego, Honeysuckle…
* Background: White, Catalina Blue, Whisper…

### Reliability

* The information storing on the database is permanent.

### Availability

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

### Security

* Only Manager has fully access to the system
* Input data is validated before saving to database.

### Maintainability

* The system is divided into separated modules for easy maintain.

### Portability

* Web application can be run on Chrome browser version 42 or later

### Performance

* System has successfully test with basic standard

## Conceptual Diagram

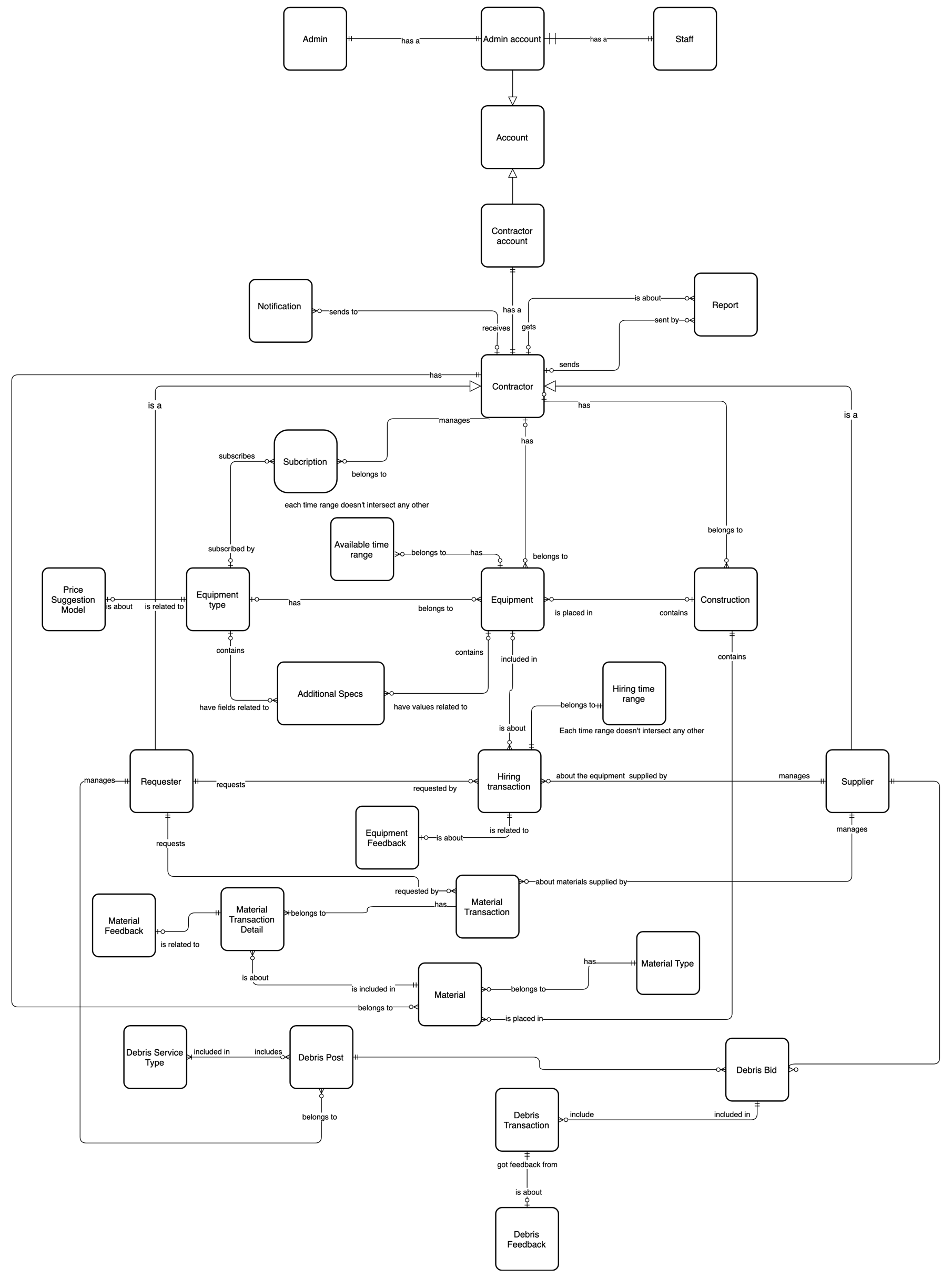


Figure 4 Conceptual Diagram

**Data Dictionary**

|  |  |
| --- | --- |
| **ENTITY DATA DICTIONARY: DESCRIBE ALL CONTENT OF ALL ENTITIES** | |
| **Entity Name** | **Description** |
| Additional Specs | Contain all additional specs information |
| Account | Contain account information |
| Admin account | Contain admin account information |
| Admin | Contain the admin information |
| Available time range | Contain available time range information of an equipment |
| Construction | Contain construction information |
| Registered User | Contain Registered User profile information |
| Registered User account | Contain Registered User account information |
| Equipment | Contain equipment information |
| Equipment type | Contain equipment type information |
| Report | Contain information of reported behavior |
| Hiring transaction | Contain hiring transaction information |
| Hiring time range | Contain hiring time range information |
| Notification | Describe a handler which will manage notifications of the system |
| Subscription | Contain subscription information |
| Material | Contain material information |
| Material transaction | Contain material transaction information |
| Material type | Contain material type information |
| Debris bid | Contain bid information for a debris post |
| Debris feedback | Contain feedback information for a finished debris transaction |
| Debris post | Contain post information for a debris service |
| Debris service type | Contain debris service type information |
| Debris transaction | Contain debris transaction information |
| Material transaction detail | Contain material transaction detail information |
| Material feedback | Contain feedback information of a finished material transaction |
| Equipment feedback | Contain feedback information for a finished hiring transaction |
| Supplier | Contain supplier information |
| Staff | Contain the staff information |
| Requester | Contain requester information |
| Price suggestion model | Contain information of the price suggestion model training result |

Table 100 Conceptual Diagram Dictionary

# Software Design Description

## Design Overview

This document describes the technical and user interface design of EMP system. It includes the architectural design, the detailed design of common functions and business functions and the design of database model.

The architectural design describes the overall architecture of the system and the architecture of each main component and subsystem.

The detailed design describes static and dynamic structure for each component and functions. It includes class diagram EMP, class explanations and sequence diagram EMP for each use cases.

 The database design describes the relationships between entities and details of each entity.

Document overview:

* Section 2: gives an overall description of the system architecture design.
* Section 3: gives component diagram EMP that describe the connection and integration of the system.
* Section 4: gives the detail design description which includes class diagram, class explanation, and sequence diagram to details the application functions.
* Section 5: describe screens design.
* Section 6: describe a fully attributed ERD.
* Section 7: describe algorithm EMP.

## System Architecture Design

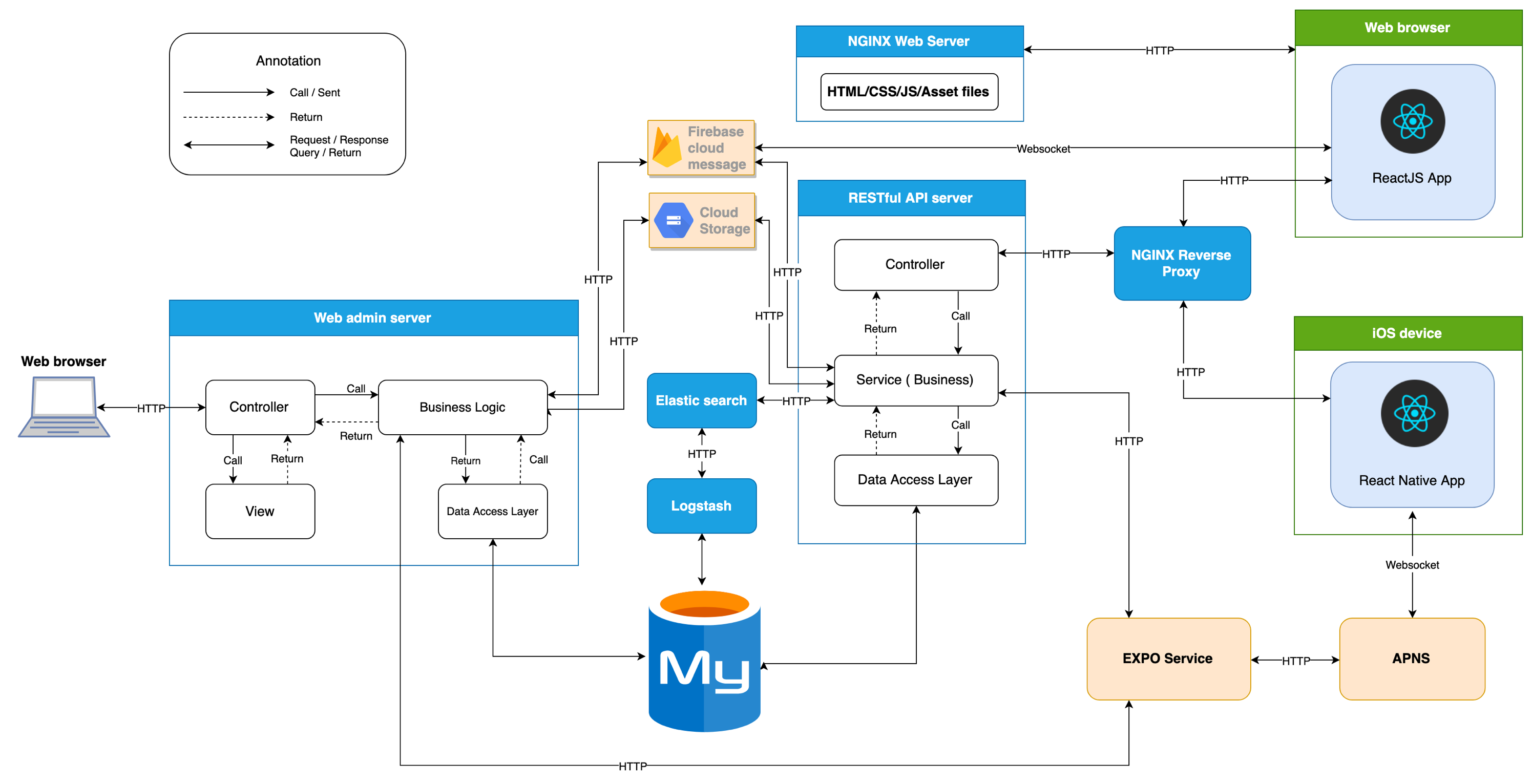


Figure 5 System Architecture diagram

### Admin web application:

In Admin Web Application, the system is developed under MVC architecture style. We choose this architecture for Admin Web Application because the following advantages:

* Quickly to develop and maintain, because the lack of human resource of our team
* Have enough flexibility and maintainability, comprehension, convention because MVC model is a classical model with variety resources and eco system.
* Save memory leaks with small amount of Javascript because of server-side rendering
* No need heavy loading in client, save computing resources for devices in offices
* Preferable for simple applications

### Registered User web system:

In Registered User web application, we choose to develop the system in Single Page Application approach because of the following advantages:

* Provide rich user interface with many features, preferable for C2C or B2C products
* Better UX/UI for end users for quick data loading and background fetching
* ReactJS is easy to master
* It is a light-weighted framework comparing to other popular ones today

In Registered User mobile application, we choose to develop the system with React Native because of the following advantages:

* Mobile app can be built with the help of Web technology which enables any web developer to easily enhance his/her skills and create React apps.
* Saves time and cost to build mobile apps on multiple platform EMP.
* A React Native app ensures speed and agility for the mobile apps with responsiveness and a great native app based user experience.

## Component Diagram

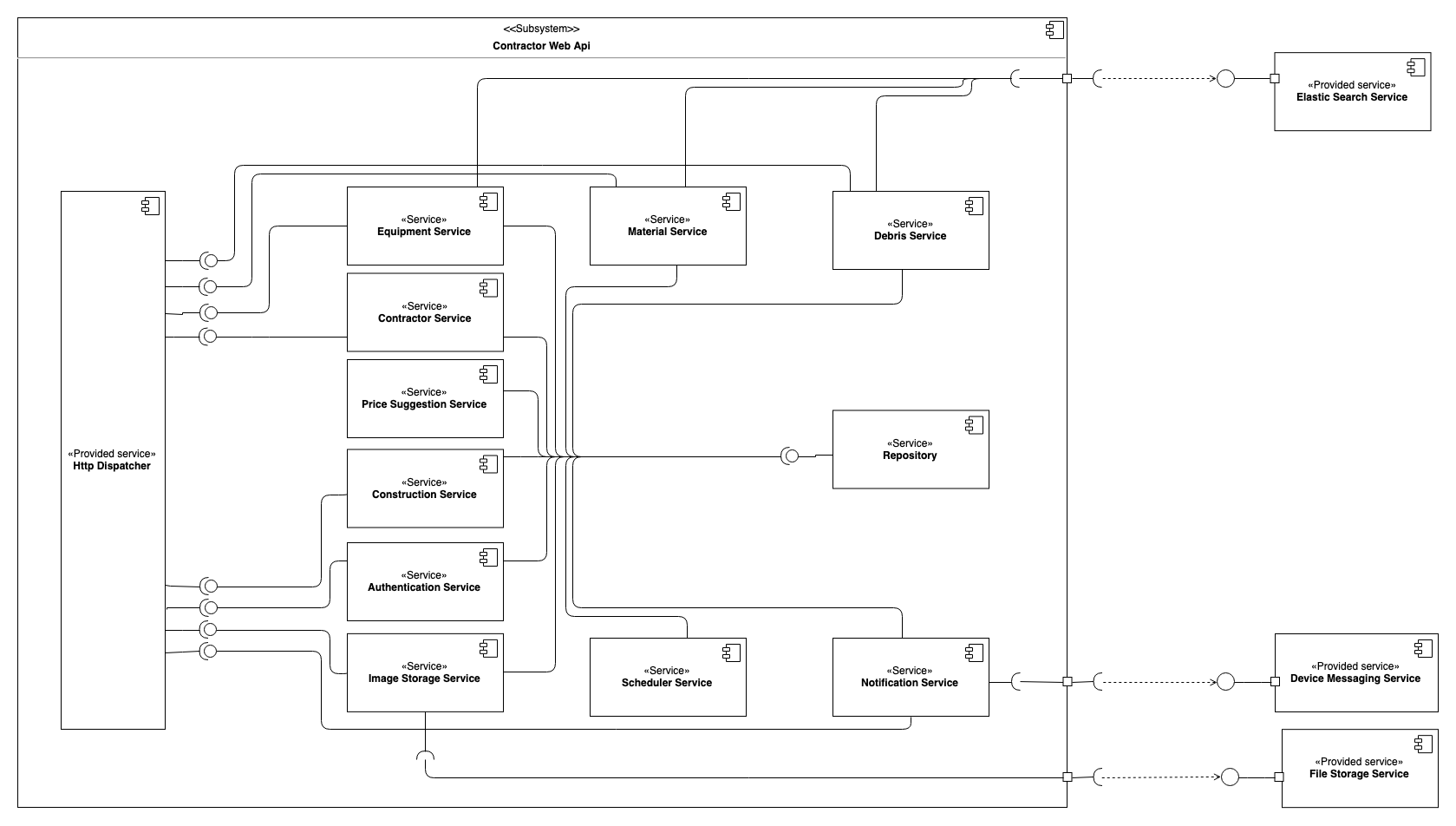


Figure 6 Component Diagram - Registered User Web Api

|  |  |
| --- | --- |
| **Component dictionary** | |
| **Component name** | **Description** |
| Equipment Service | A service responsible for managing information related to equipment, hiring transaction, hiring transaction feedbacks. |
| Material Service | A service responsible for managing information related to materials, material transactions, material transaction feedbacks. |
| Debris Service | A service responsible for managing information related to debris, debris post, debris bids, debris transactions, debris transaction feedbacks. |
| Registered User Service | A service responsible for managing information related to Registered Users. |
| Construction Service | A service responsible for managing information related to constructions. |
| Authentication Service | A service responsible for authentication and authorization. |
| Image Storage Service | A service responsible for uploading images, including equipment images, material images, debris images, profile images. |
| Scheduler Service | A service responsible for time-related tasks. |
| Notification Service | A service responsible for managing notification |
| Repository | A service responsible for data management. |
| Elastic Search Service | A provided service of Elastic search, responsible for low-latency performance information gathering and managing searchable resource |
| Device Messaging Service | An external service responsible for send notifications to actual devices. This service can be realized by Firebase Cloud Messaging service and Expo Messaging service. |
| File Storage Service | An external service responsible for raw resource storage. This service can be realized by Google Cloud Storage service, Imgur Storage service, etc. |
| Price Suggestion Service | A service responsible for model training, data collecting data, data processing, model applying for price suggestion. |

Figure 7 Component Data Dictionary

## Detailed Description

### Class Diagram

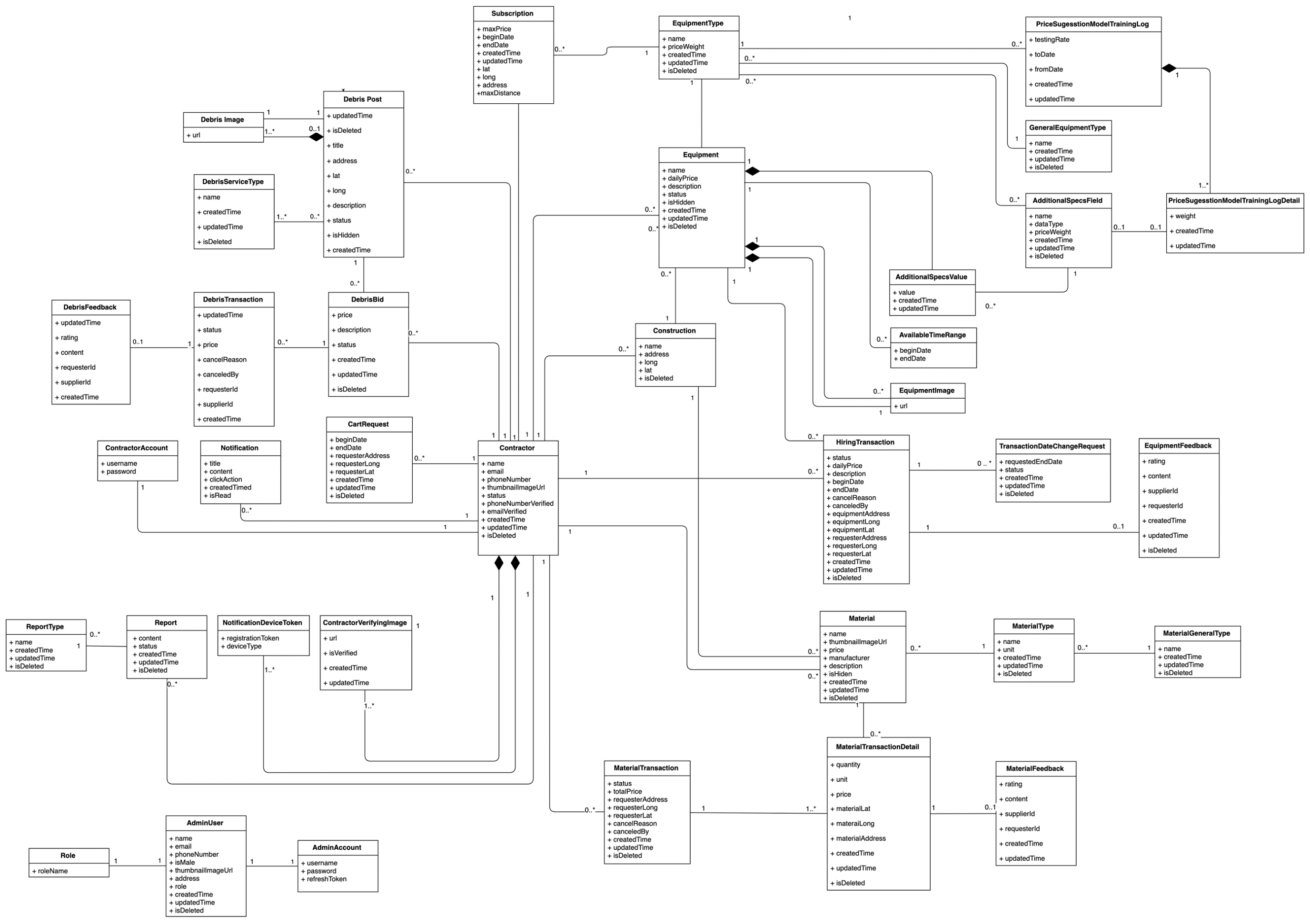


Figure 8 Class Diagram

### Class Diagram Explanation

|  |  |  |
| --- | --- | --- |
| **CLASS DICTIONARY: DESCRIBE CLASS** | | |
| **Class Name** | **Mapping column with Conceptual diagram** | **Description** |
| Registered UserAccount | Registered User Account | Contain Registered User account information |
| AdditionalSpecsField | N/A | Not exist in conceptual diagram but need this table to separate responsibility with AdditionalSpecs |
| AdditionalSpecsValue | N/A | Not exist in conceptual diagram but need this table to separate responsibility with AdditionalSpecs |
| AdminUser | Admin | Contain admin user profile information |
| AdminAccount | Admin account | Contain admin account information |
| AvailableTimeRange | Available time range | Contain available time range information of a equipment |
| Construction | Construction | Contain construction information |
| Registered User | Registered User | Contain Registered User profile information |
| EquipmentImage | N/A | Not exist in conceptual diagram but need this table to contain information of equipment’s image |
| Equipment | Equipment | Contain equipment information |
| EquipmentType | Equipment type | Contain equipment type information |
| Report | Report | Contain information of reported behavior |
| ReportType | N/A | Not exist in conceptual diagram but need this table to contain information of report type |
| GeneralEquipmentType | N/A | Not exist in conceptual diagram but need this table to contain information of equipment category, which has multiple equipment types |
| HiringTransaction | Hiring transaction | Contain hiring transaction information |
| Notification | Notification | Contain notification information |
| Subscription | Subscription | Contain subscription information |
| TransactionDateChangeRequest | N/A | Not exist in conceptual diagram but need this table to contain information of hiring transaction’s date changing request |
| NotificationDeviceToken | N/A | Not exist in conceptual diagram but need this table to contain information of the token of each device for notification |
| GeneralMaterialType | N/A | Not exist in conceptual diagram but need this table to contain information of material category, which has multiple material types |
| Material | Material | Contain material information |
| MaterialTransaction | Material transaction | Contain material transaction information |
| MaterialType | Material type | Contain material type information |
| DebrisBid | Debris bid | Contain bid information for a debris post |
| DebrisFeedback | Debris feedback | Contain feedback information for a finished debris transaction |
| DebrisPost | Debris post | Contain post information for a debris service |
| DebrisServiceType | Debris service type | Contain debris service type information |
| DebrisTransaction | Debris transaction | Contain debris transaction information |
| DebrisImage | N/A | Not exist in conceptual diagram but need this table to contain information of image of a debris post |
| MaterialTransactionDetail | Material transaction detail | Contain material transaction detail information |
| MaterialFeedback | Material feedback | Contain feedback information of a finished material transaction |
| EquipmentFeedback | Equipment feedback | Contain feedback information for a finished hiring transaction |
| Registered UserVerifyingImage | N/A | Not exist in conceptual diagram but need this table to contain information of the Registered User’s images for verifying |
| Role | N/A | Contain role information |
| PriceSuggestionModalTrainingLogDetail | Price Suggestion Model | Contain log information of Price Suggestion Model Training process |
| PriceSuggestionModalTrainingLog | N/A | Contain Details of Price Suggestion Model Training Log |

Figure 9 Class Diagram Dictionary

#### Registered UserAccount

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of Registered User account |
| username | String | Private | Registered User account’s username |
| password | String | Private | Account’s password |
| Registered UserId | Integer | Private | Registered User’s id |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### AdditionalSpecsField

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an additional specs field |
| name | String | Private | Additional specs field’s name |
| dataType | String | Private | Additional specs field’s data type |
| equipmentTypeId | Integer | Private | Equipment type’s id |
| idDeleted | Tiny Integer | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### AdditionalSpecsValue

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an additional specs value |
| value | String | Private | Additional specs value’s name |
| additionalSpecsFieldId | Integer | Private | Contain information of the mentioned attribute |
| equipmentId | Integer | Private | Contain information of the mentioned attribute |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### AdminUser

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an additional specs value |
| name | String | Private | Contain information of the mentioned attribute |
| isMale | Bit | Private | Contain information of the mentioned attribute |
| phone | String | Private | Contain information of the mentioned attribute |
| email | String | Private | Contain information of the mentioned attribute |
| roleId | Integer | Private | Contain information of the mentioned attribute |
| thumbnailImage | String | Private | Thumbnail image url |
| address | String | Private | Contain information of the mentioned attribute |

#### AdminAccount

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an admin account |
| username | String | Private | Contain information of the mentioned attribute |
| password | String | Private | Contain information of the mentioned attribute |
| adminUserId | Integer | Private | Contain information of the mentioned attribute |

#### AvailableTimeRange

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an available time range |
| beginDate | Date | Private | The begin date of hiring transaction |
| endDate | Date | Private | The end date of hiring transaction |
| equipmentId | Integer | Private | Contain information of the mentioned attribute |

#### Construction

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of the construction |
| name | String | Private | Contain information of the mentioned attribute |
| address | String | Private | Contain information of the mentioned attribute |
| long | Double | Private | Contain information of the mentioned attribute |
| lat | Double | Private | Contain information of the mentioned attribute |
| isDeleted | Tiny Integer | Private | Soft delete information |
| Registered UserId | Integer | Private | Contain information of the mentioned attribute |

#### Registered User

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a Registered User |
| name | String | Private | Contain information of the mentioned attribute |
| email | Bit | Private | Contain information of the mentioned attribute |
| phoneNumber | String | Private | Contain information of the mentioned attribue |
| thumbnailImage | String | Private | Contain information of the mentioned attribue |
| isActivated | Integer | Private | Contain information of the mentioned attribute |
| createdTime | String | Private | The created time of the data |
| updatedTime | String | Private | The updated time of the data |
| isDeleted | Tiny Integer | Private | Soft delete information |

#### EquipmentImage

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an equipment image |
| url | String | Private | Contain information of the mentioned attribute |
| equipmentId | Integer | Private | Contain information of the mentioned attribute |
| createdTime | String | Private | The created time of the data |
| updatedTime | String | Private | The updated time of the data |

#### Equipment

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an equipment |
| name | String | Private | Contain information of the mentioned attribute |
| dailyPrice | Double | Private | Contain information of the mentioned attribute |
| description | String | Private | Contain information of the mentioned attribute |
| status | String | Private | Contain information of the mentioned attribute |
| thumbnailImageId | Integer | Private | Contain information of the mentioned attribute |
| isDeleted | Tiny Integer | Private | Soft delete information |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| equipmentTypeId | Integer | Private | Contain information of the mentioned attribute |
| Registered UserId | Integer | Private | Contain information of the mentioned attribute |
| constructionId | Integer | Private | Contain information of the mentioned attribute |
| equipmentImageId | Integer | Private | Contain information of the mentioned attribute |

#### EquipmentType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an equipment type |
| name | String | Private | Contain information of the mentioned attribute |
| generalEquipmentTypeId | Integer | Private | Contain information of the mentioned attribute |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| priceWeight | Double | Private | Contain information of the mentioned attribute |

#### Report

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of report |
| content | String | Private | Additional specs field’s name |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| toRegistered UserId | Datetime | Private | Contain information of the mentioned attribute |
| fromRegistered UserId | Datetime | Private | Contain information of the mentioned attribute |
| reportTypeId | Double | Private | Contain information of the mentioned attribute |
| status | String | Private | Contain information of the mentioned attribute |
| isDeleted | Bit | Private | Soft delete information |

#### ReportType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of report type |
| name | String | Private | Additional specs field’s name |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### GeneralEquipmentType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a general equipment type |
| name | String | Private | General equipment type name |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### HiringTransaction

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of hiring transaction |
| status | String | Private | Hiring transaction’s status |
| dailyPrice | Double | Private | Contain information of the mentioned attribute |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| beginDate | Date | Private | The begin date of hiring transaction |
| endDate | Date | Private | The begin date of hiring transaction |
| equipmentAddress | String | Private | Contain information of the mentioned attribute |
| equipmentLong | Double | Private | Contain information of the mentioned attribute |
| equipmentLat | Double | Private | Contain information of the mentioned attribute |
| requesterAddress | String | Private | Contain information of the mentioned attribute |
| requesterLong | Double | Private | Contain information of the mentioned attribute |
| requesterLat | Double | Private | Contain information of the mentioned attribute |
| isDeleted | Bit | Private | Soft delete information |
| equipmentId | Integer | Private | Contain information of the mentioned attribute |
| requesterId | Integer | Private | Contain information of the mentioned attribute |
| cancelReason | String | Private | Contain information of the mentioned attribute |
| canceledBy | Integer | Private | Contain information of the mentioned attribute |

#### Notification

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of notification |
| title | String | Private | Additional specs field’s name |
| content | String | Private | Contain information of the mentioned attribute |
| isRead | Tiny Integer | Private | Contain information of the mentioned attribute |
| Registered UserId | Integer | Private | Contain information of the mentioned attribute |
| clickAction | String | Private | Contain information of the mentioned attribute |

#### Subscription

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an additional specs value |
| equipmentTypeId | String | Private | Additional specs field’s name |
| Registered UserId | Bit | Private | Contain information of the mentioned attribute |
| maxPrice | Double | Private | Contain information of the mentioned attribute |
| beginDate | Date | Private | Contain information of the mentioned attribute |
| endDate | Date | Private | Contain information of the mentioned attribute |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| lat | Double | Private | Contain information of the mentioned attribute |
| long | Double | Private | Contain information of the mentioned attribute |
| maxDistance | Double | Private | Contain information of the mentioned attribute |
| address | String | Private | Contain information of the mentioned attribute |

#### TransactionDateChangeRequest

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an additional specs value |
| requestedEndDate | Date | Private | Additional specs field’s name |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| status | String | Private | Contain information of the mentioned attribute |
| transactionId | Integer | Private | Contain information of the mentioned attribute |

#### NotificationDeviceToken

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a general equipment type |
| registrationToken | String | Private | General equipment type name |
| Registered UserId | Integer | Private | Soft delete information |
| deviceType | String | Private |  |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### GeneralMaterialType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a general material type |
| name | String | Private | General material type name |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### Material

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a material |
| name | String | Private | Material’s name |
| price | Double | Private | Material’s price |
| manufacturer | String | Private | Material’s manufacturer |
| description | String | Private | Material’s description |
| isHidden | Tiny Integer | Private | Hide material post |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| constructionId | Integer | Private | Construction id |
| Registered UserId | Integer | Private | Registered User id |
| materialTypeId | Integer | Private | Material type id |
| thumbnailImageUrl | String | Private | Material thumbnail image |

#### MaterialTransaction

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a material transaction |
| totalPrice | Double | Private | Material transaction total price |
| requesterAddress | String | Private | Material transaction requester address |
| requesterLat | Double | Private | Requester latitude |
| requesterLong | Double | Private | Requester longitude |
| status | String | Private | Material transaction’s status |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| requesterId | Integer | Private | Requester’s id |
| supplierId | Integer | Private | Supplier’s id |
| cancelReason | String | Private | Material transaction cancel reason |
| canceledBy | Integer | Private | Material transaction canceled by id |

#### MaterialType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a material type |
| name | String | Private | Material type name |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| generalMaterialTypeId | Integer | Private | Id of general material type |
| unit | Integer | Private | Define unit of material |

#### DebrisBid

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a debris bid |
| price | Double | Private | Debris bid’s price |
| status | String | Private | Debris bid’s status |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| isDeleted | Bit | Private | Soft delete information |
| supplierId | Integer | Private | Supplier’s id |
| debrisPostId | Integer | Private | Debris post’s id |
| description | String | Private | Description of debris bid |

#### DebrisFeedback

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a debris feedback |
| rating | Double | Private | Feedback rating |
| content | String | Private | Feedback content |
| requesterId | Integer | Private | Requester’s id |
| supplierId | Integer | Private | Supplier’s Id |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| debrisTransactionId | Integer | Private | Debris transaction’s id |

#### DebrisPost

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a debris post |
| title | String | Private | Debris post’s title |
| address | String | Private | Debris post’s address |
| longitude | Double | Private | Debris post’s longitude |
| latitude | Double | Private | Debris post’s latitude |
| description | String | Private | Debris post’s description |
| status | String | Private | Debris post’s status |
| isHidden | Tiny Integer | Private | Soft remove debris post |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| requesterId | Integer | Private | Requester’s id |
| thumbnailImageId | Integer | Private | Thumbnail image’s id |

#### DebrisServiceType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a debris service type |
| name | String | Private | Debris service type name |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### DebrisTransaction

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a debris transaction |
| price | Double | Private | Debris transaction’s price |
| status | String | Private | Debris transaction’s status |
| cancelReason | Integer | Private | Debris transaction’s cancel reason |
| requesterId | Integer | Private | Requester’s id |
| supplierId | String | Private | Debris transaction name |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| debrisPostId | Integer | Private | Debris post’s id |
| debrisBidId | Integer | Private | Debris bid’s id |
| canceledBy | Integer | Private | Debris transaction canceled by id |

#### DebrisImage

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a debris image |
| url | String | Private | Debris image’s url |
| debrisPostId | Bit | Private | Debris post ‘s id |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### MaterialTransactionDetail

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a material transaction detail |
| quantity | Double | Private | Material transaction detail’s quantity |
| unit | String | Private | Material transaction detail’s unit |
| price | Double | Private | Material transaction detail’s price |
| materialLat | Double | Private | Material transaction detail’s latitude |
| materialLong | Double | Private | Material transaction detail’s longitude |
| materialAddress | String | Private | Material transaction detail’s address |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| materialId | Integer | Private | Material’s id |
| materialTransactionId | Integer | Private | Material transaction’s id |

#### MaterialFeedback

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a material feedback |
| rating | Double | Private | Material feedback’s rating |
| content | String | Private | Material feedback’s  content |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| supplierId | Integer | Private | Supplier’s id |
| requesterId | Integer | Private | Requester’s id |
| materialTransactionDetailId | Integer | Private | Material transaction detail’s id |

#### EquipmentFeedback

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a equipment feedback |
| rating | Double | Private | Equipment feedback’s rating |
| content | String | Private | Equipment feedback’s  content |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| supplierId | Integer | Private | Supplier’s id |
| requesterId | Integer | Private | Requester’s id |
| hiringTransactionId | Integer | Private | Equipment transaction’s id |

#### Registered UserVerifyingImage

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a Registered User verifying image |
| url | String | Private | Registered User verifying image’s url |
| Registered UserId | Integer | Private | Registered User’s id |
| isVerified | Tiny Integer | Private | Verify image information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

Role

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of role |
| roleName | String | Private | Role’s name |

PriceSuggestionModalTrainingLogDetail

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of price suggestion detail |
| weight | Double | Private | Contain information of the mentioned attribute |
| priceSuggestionModelTraninglog Id | Integer | Private | Contain information of the mentioned attribute |
| additionalSpecsFieldId | Integer | Private | Contain information of the mentioned attribute |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

PriceSuggestionModalTrainingLog

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of price suggestion |
| weight | Double | Private | Contain information of the mentioned attribute |
| toDate | Integer | Private | Contain information of the mentioned attribute |
| fromDate | Integer | Private | Contain information of the mentioned attribute |
| equipmentTypeId | Integer | Private | Contain information of the mentioned attribute |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

### Interaction Diagram

#### Registered User web application

##### Search equipment

Summary:

* This diagram shows process of Registered User searching for equipment after providing all required information
* Described in use case EMP\_UC\_2

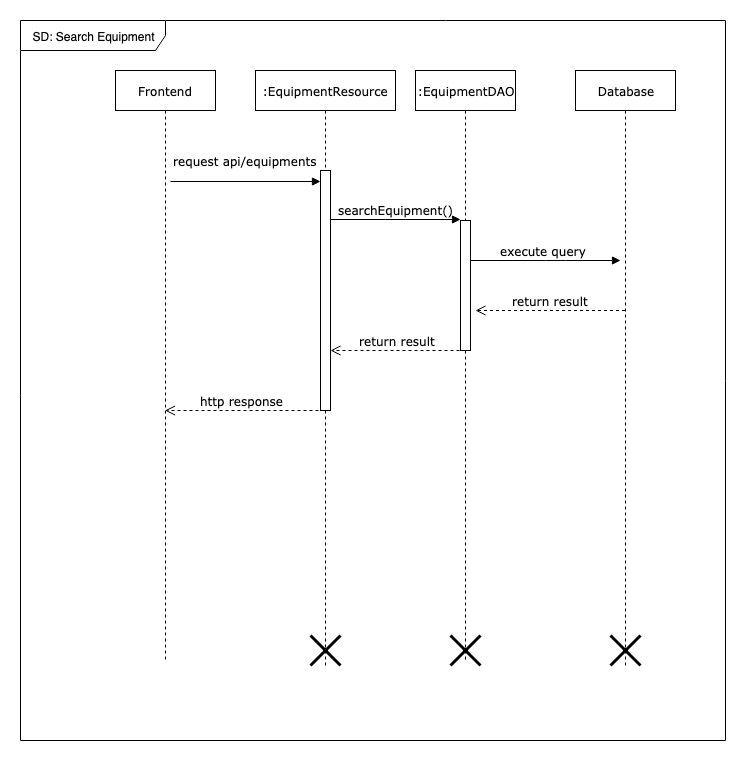


Figure 10 Sequence Diagram - Search equipment

##### Login

Summary:

* This diagram shows process of guest logging into the system
* Described in use case EMP\_UC\_5

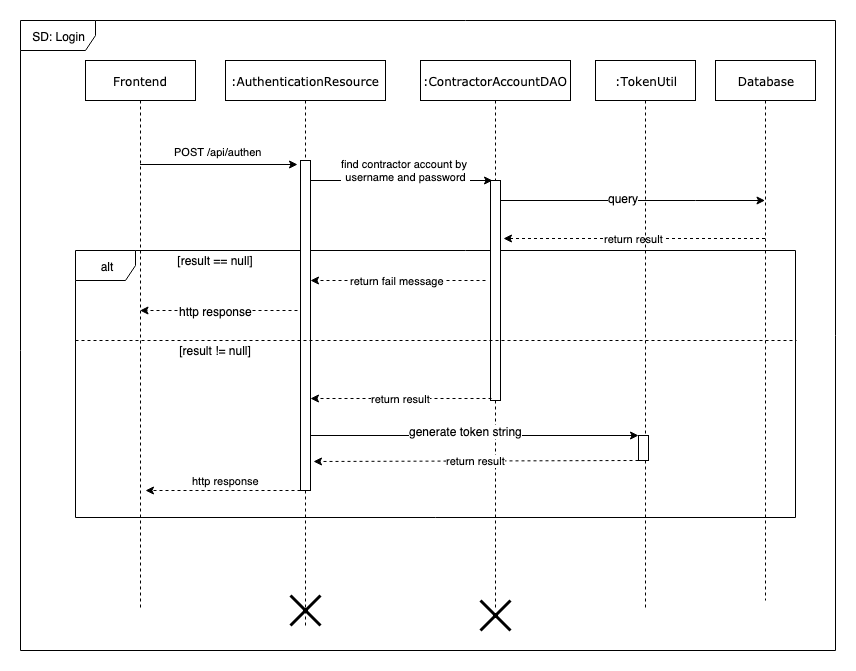


Figure 11 Sequence Diagram - Login

##### Check subscription matched

Summary:

* This diagram shows the process of Scheduler check if there are any equipment match the subscriptions
* Described in use case EMP\_UC\_68

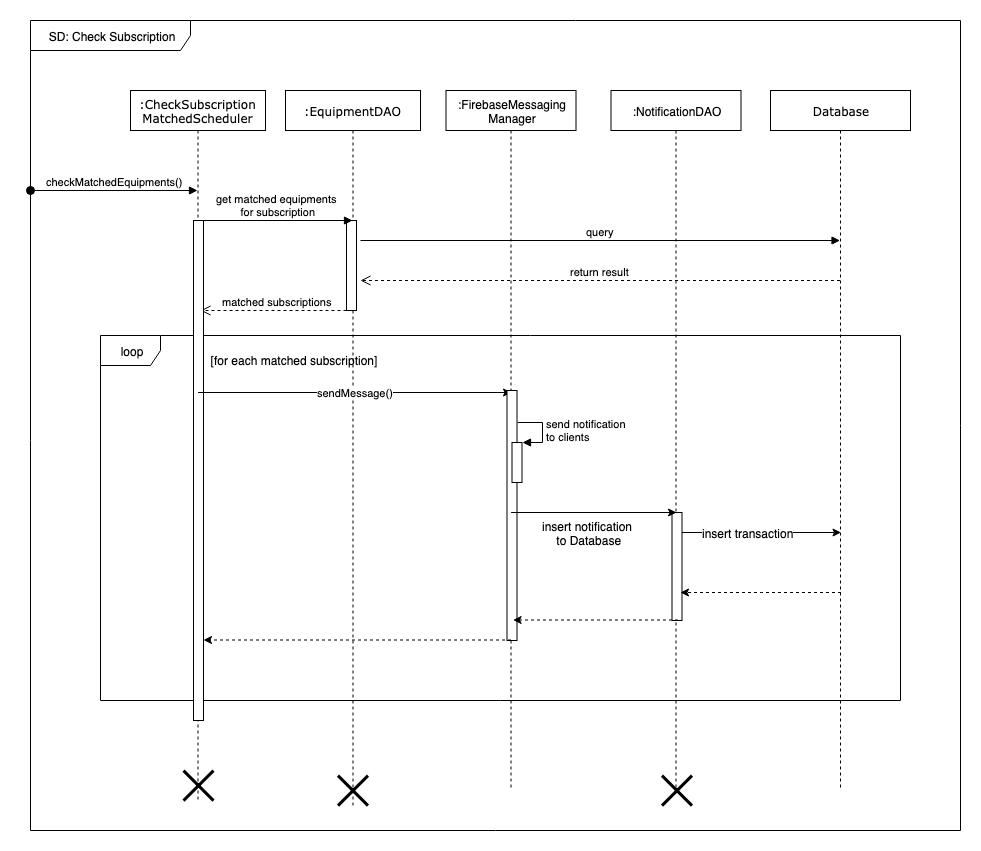


Figure 12 Sequence Diagram - Check Subscription

##### Post equipment

Summary:

* This diagram shows the process of Registered User posting a new equipment to the system after input required information
* Described in use case EMP\_UC\_31

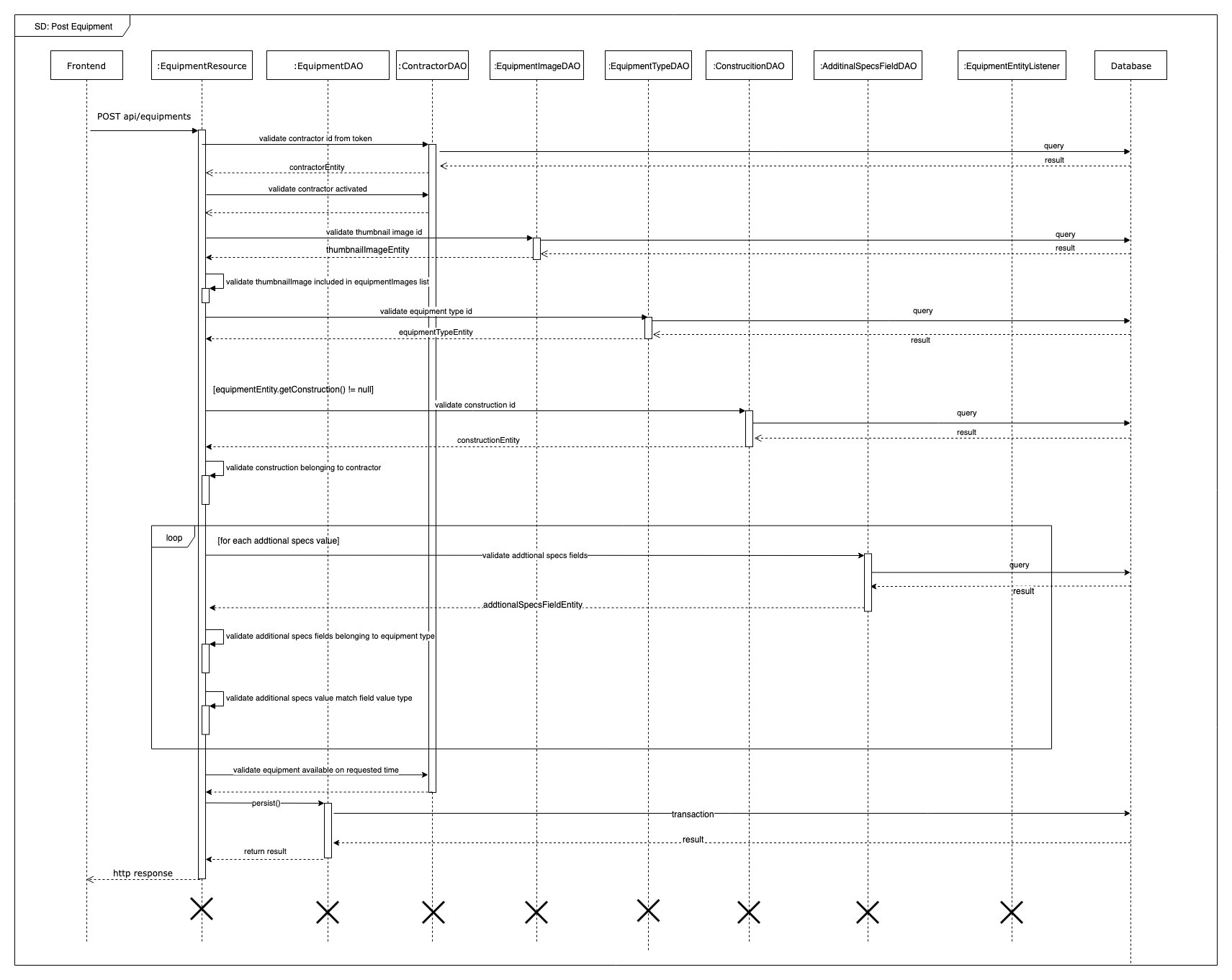


Figure 13 Sequence Diagram - Post equipment

##### Request equipment

Summary:

* This diagram shows the process of Registered User request an equipment of a supplier
* Described in use case EMP\_UC\_46

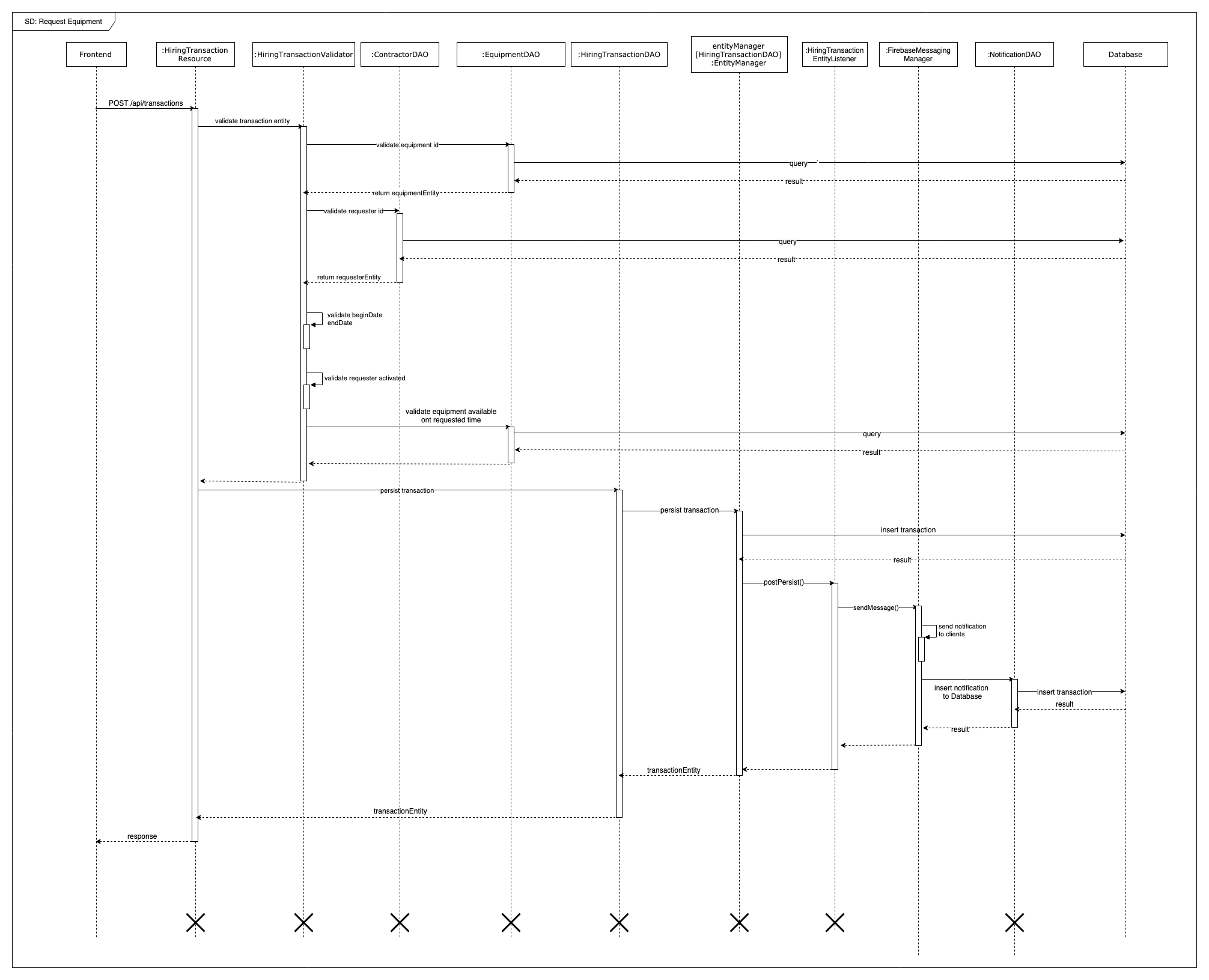


Figure 14 Sequence Diagram - Request equipment

##### Change hiring transaction status

Summary:

* This diagram shows the process of Registered User changing status of hiring transaction
* Described in use case EMP\_UC\_45, EMP\_UC\_46, EMP\_UC\_47, EMP\_UC\_50, EMP\_UC\_21, EMP\_UC\_22, EMP\_UC\_23, EMP\_UC\_24 EMP\_UC\_25

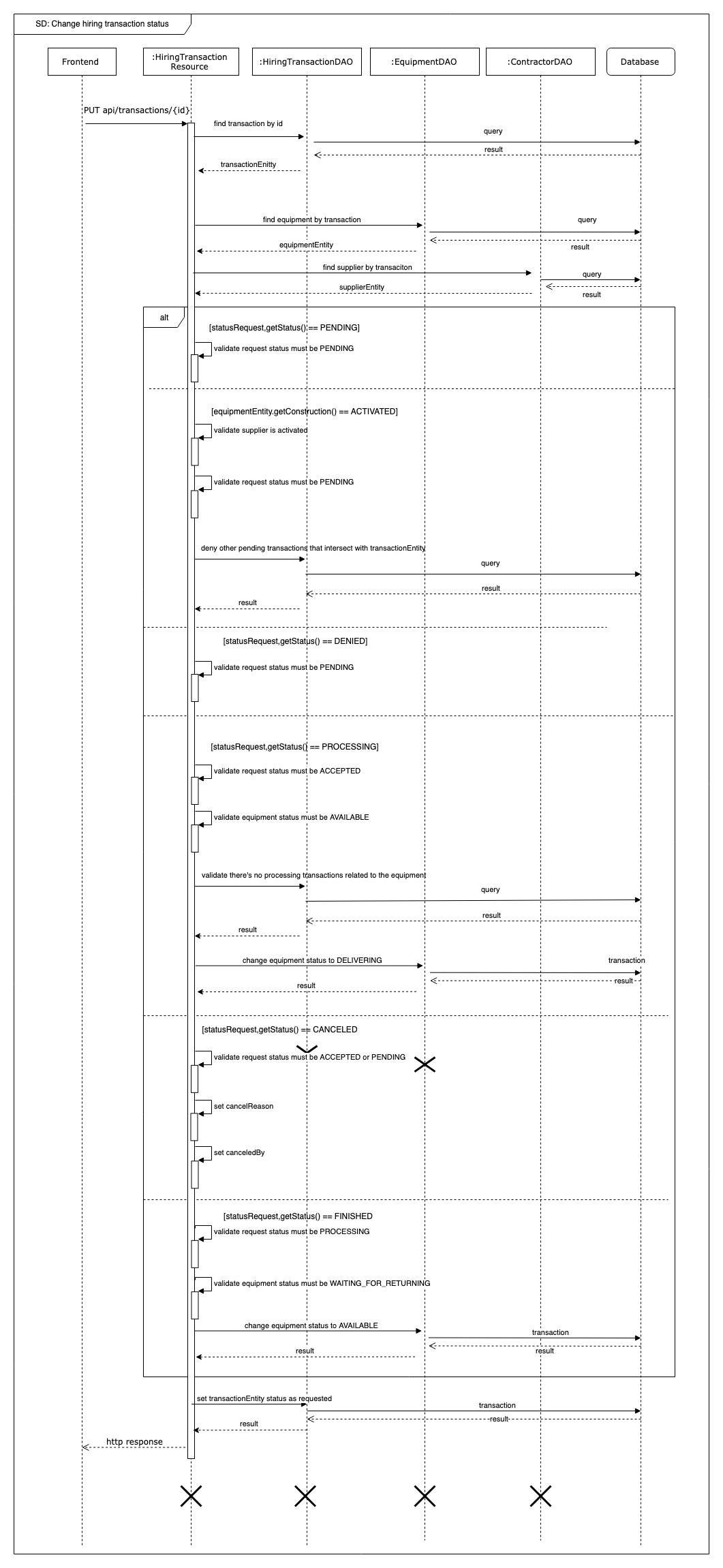


Figure 15 Sequence Diagram - Chang hiring status

##### Checkout material

Summary:

* This diagram shows the process of Registered User checkout multiple materials from cart after providing required information
* Described in use case EMP\_UC\_53



Figure 16 Sequence Diagram - Checkout Materials

##### Publish debris post

Summary:

* This diagram shows the process of Registered User publishing debris post with provided requirements to the system
* Described in use case EMP\_UC\_57

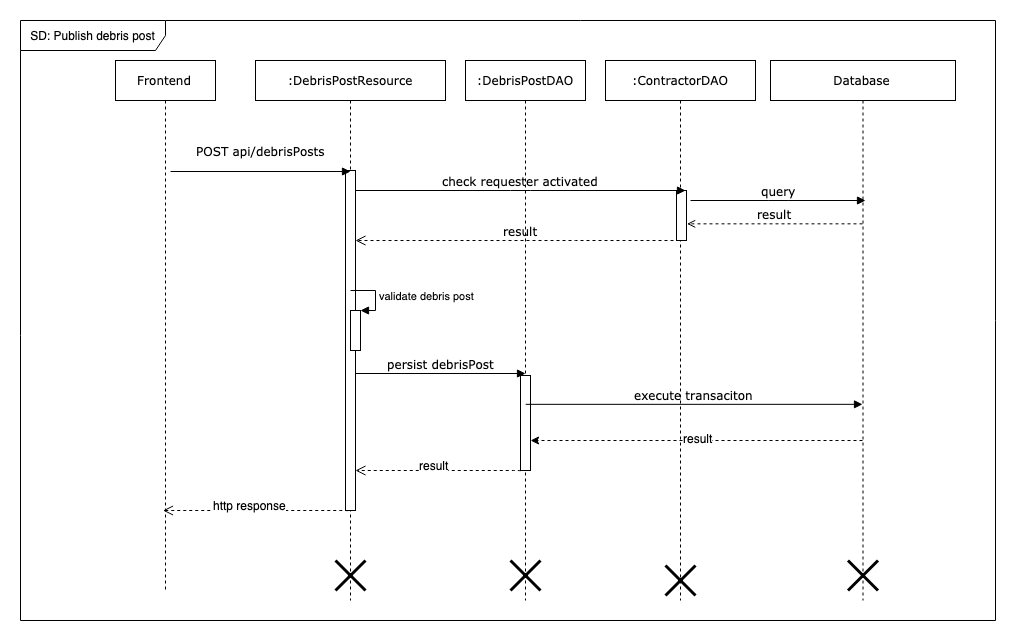


Figure 17 Sequence Diagram - Publish Debris Post

##### Bid on debris post

Summary:

* This diagram shows the process of Supplier bid on a debris post for a service offer
* Described in use case EMP\_UC\_40

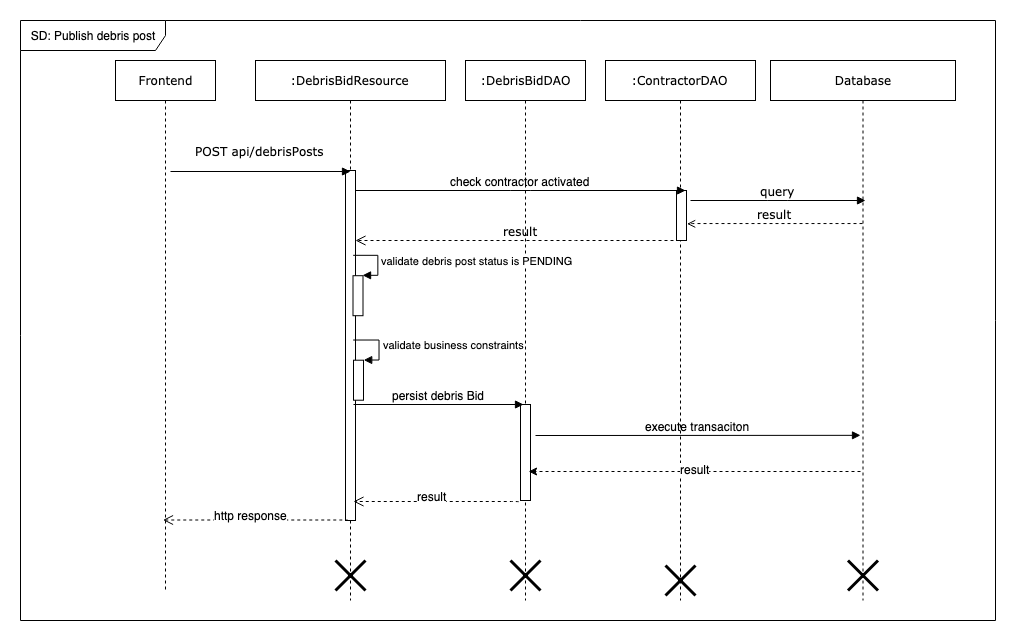
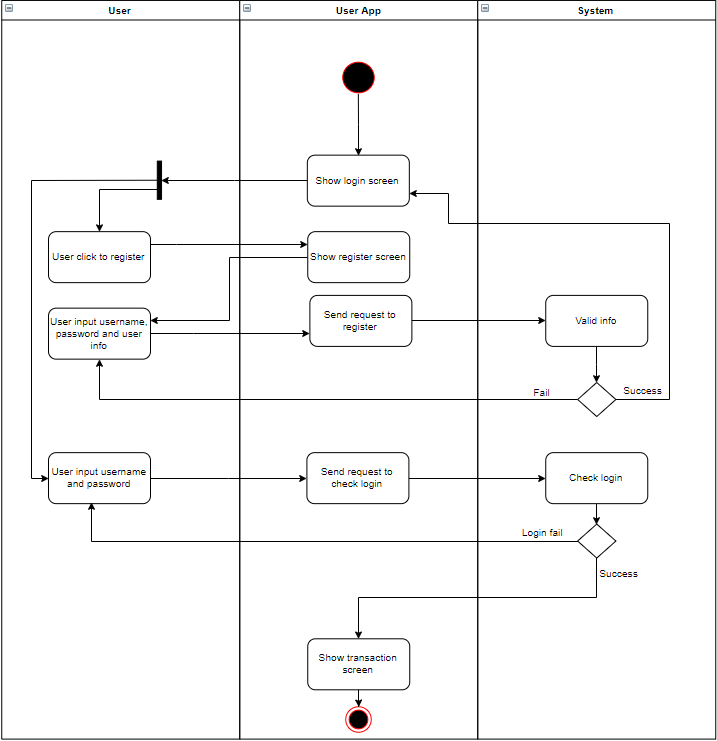


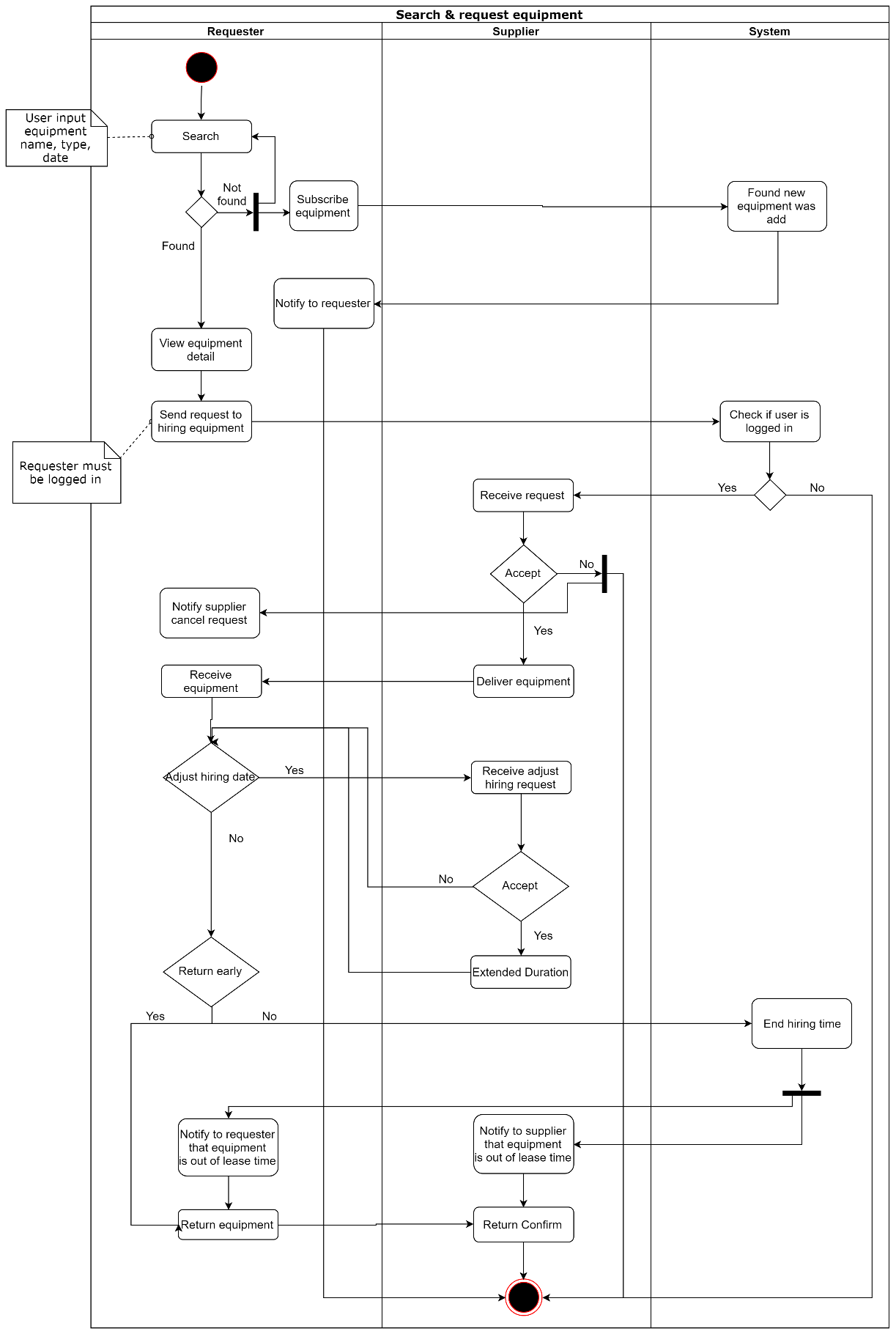
Figure 18 Sequence Diagram - Bid on Debris Post

#### Mobile application

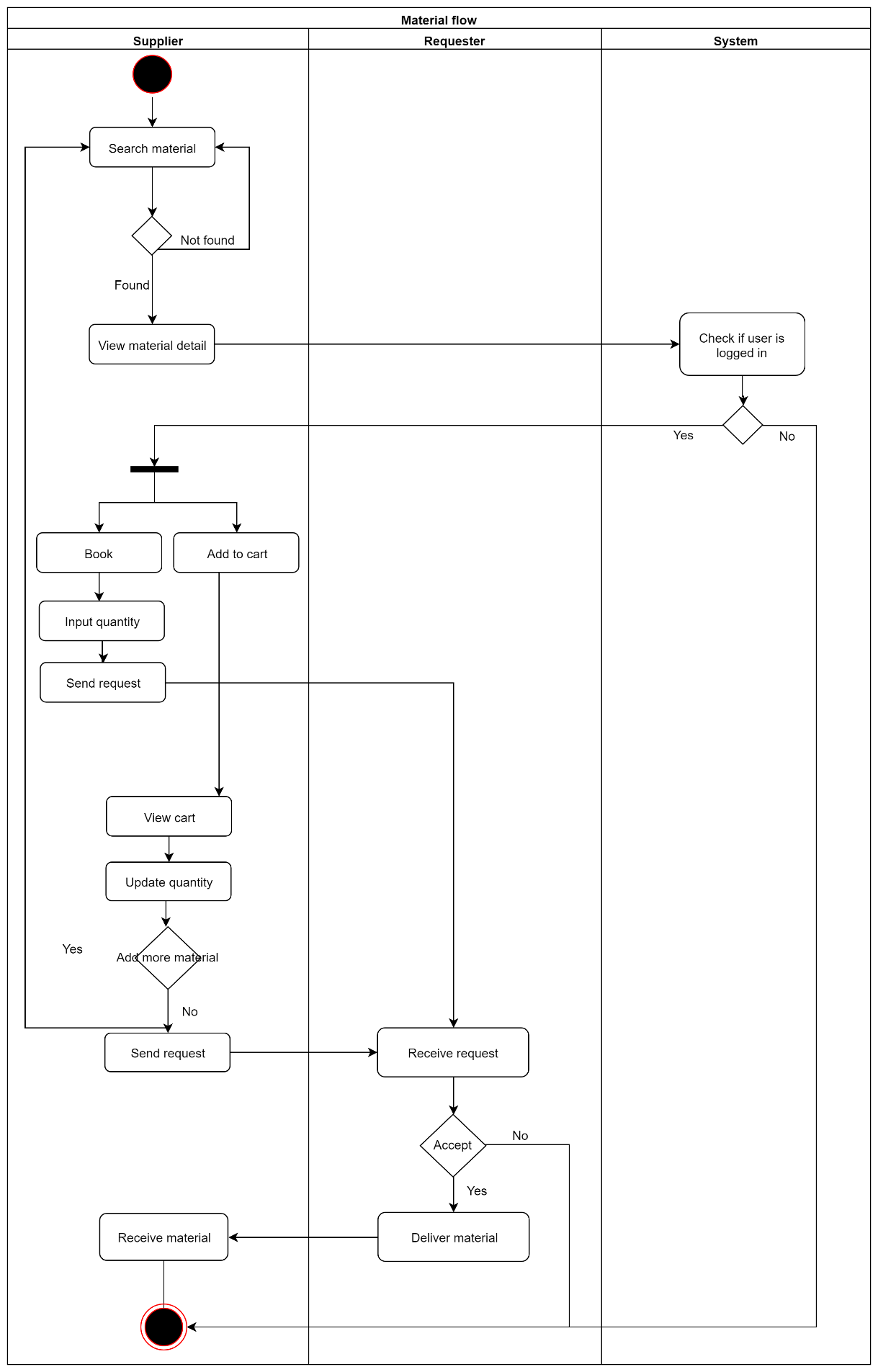
##### Authentication



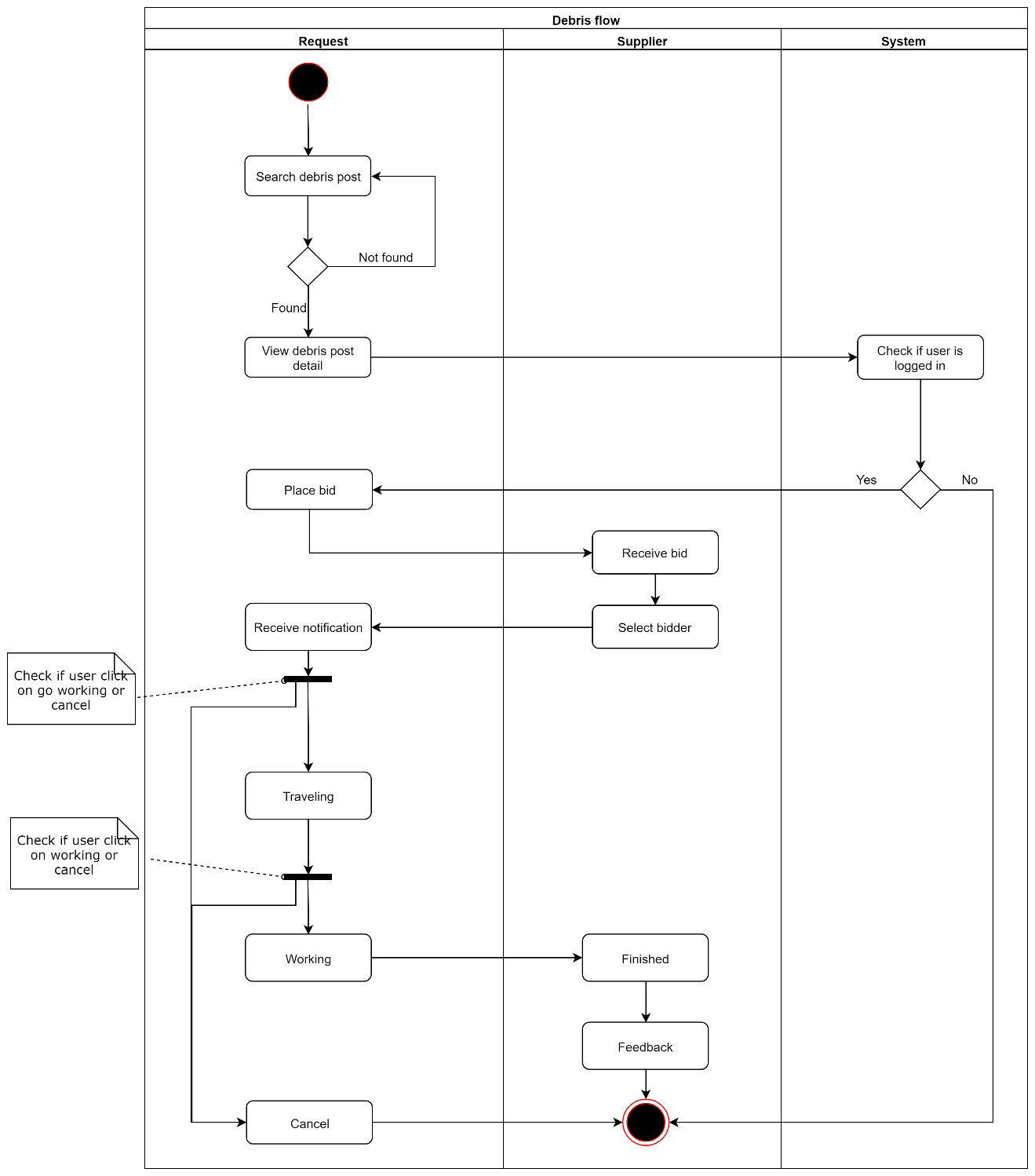
##### Rent equipment



##### Buy material



##### Bid on debris service



#### Admin web application

##### <Staff> Verify Registered User

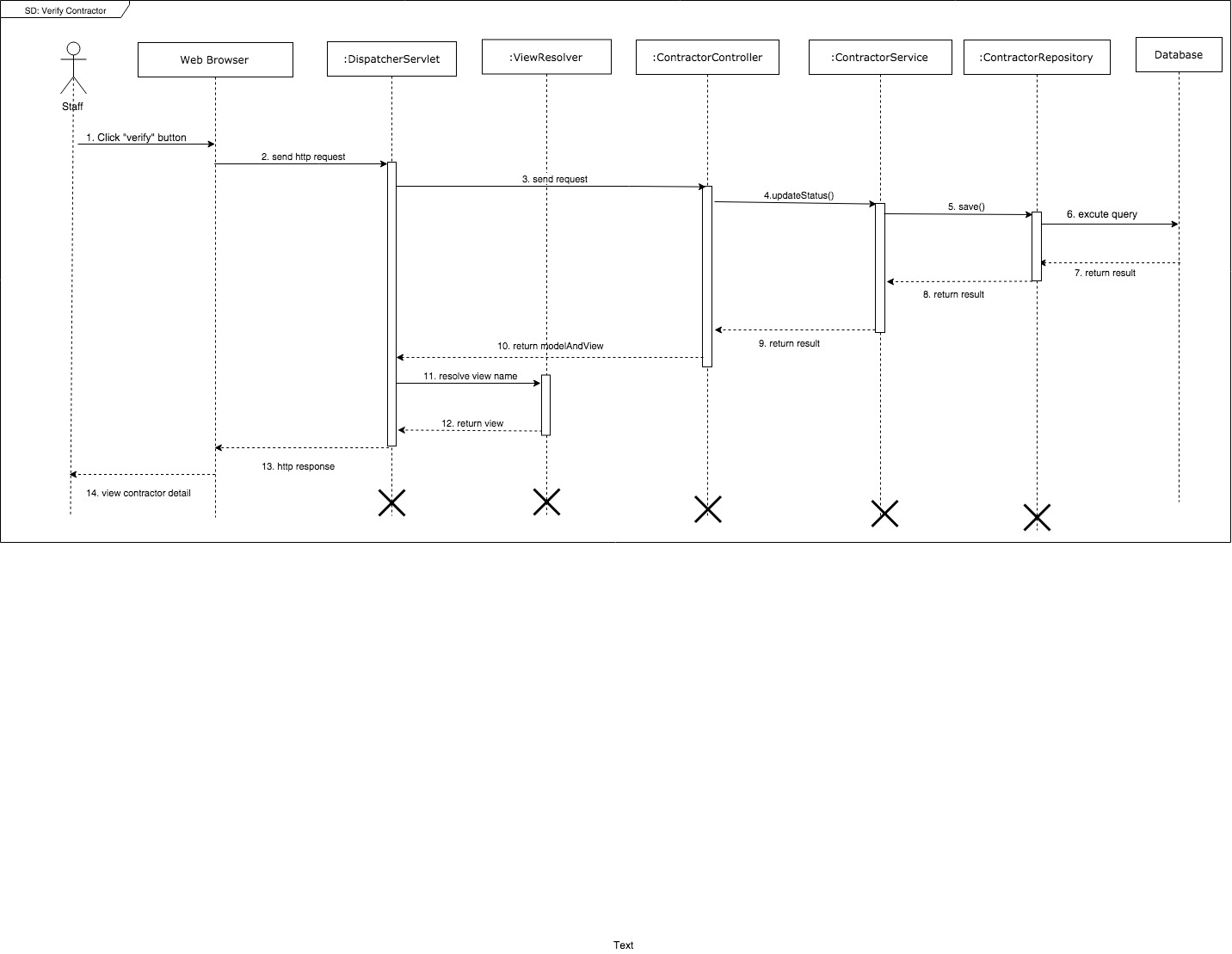
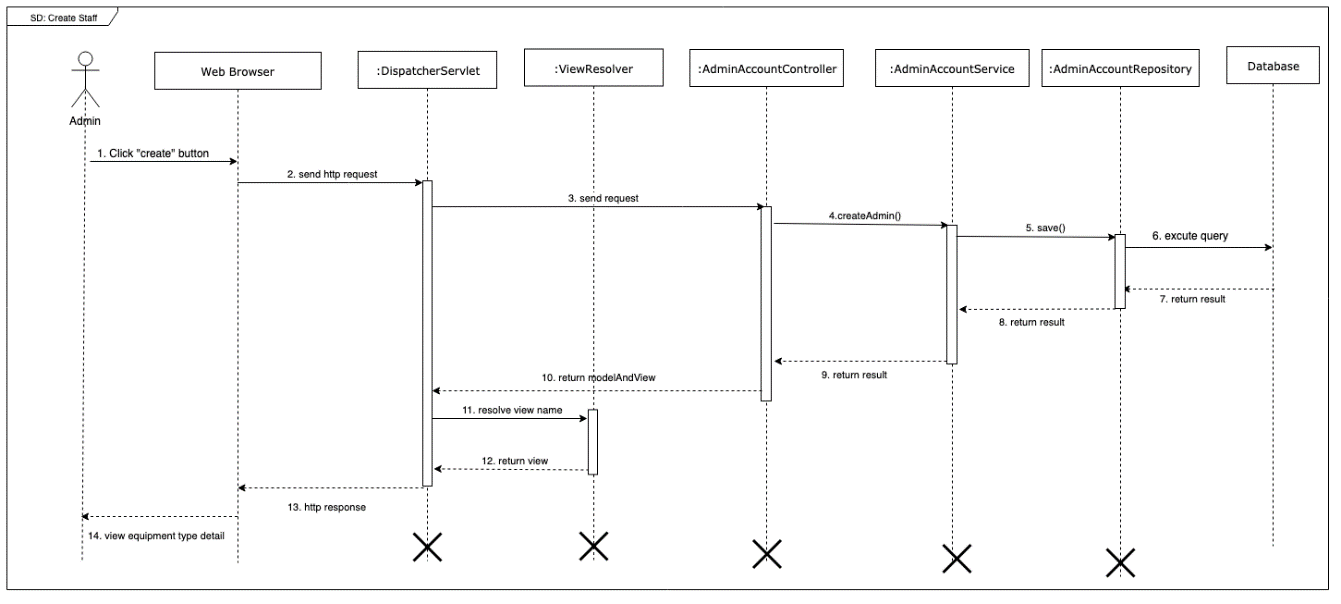


Figure 19 Sequence diagram - Verify Registered User

##### <Admin> Add staff

Figure 20 Sequence diagram - Add staff



##### <Admin> Add equipment type

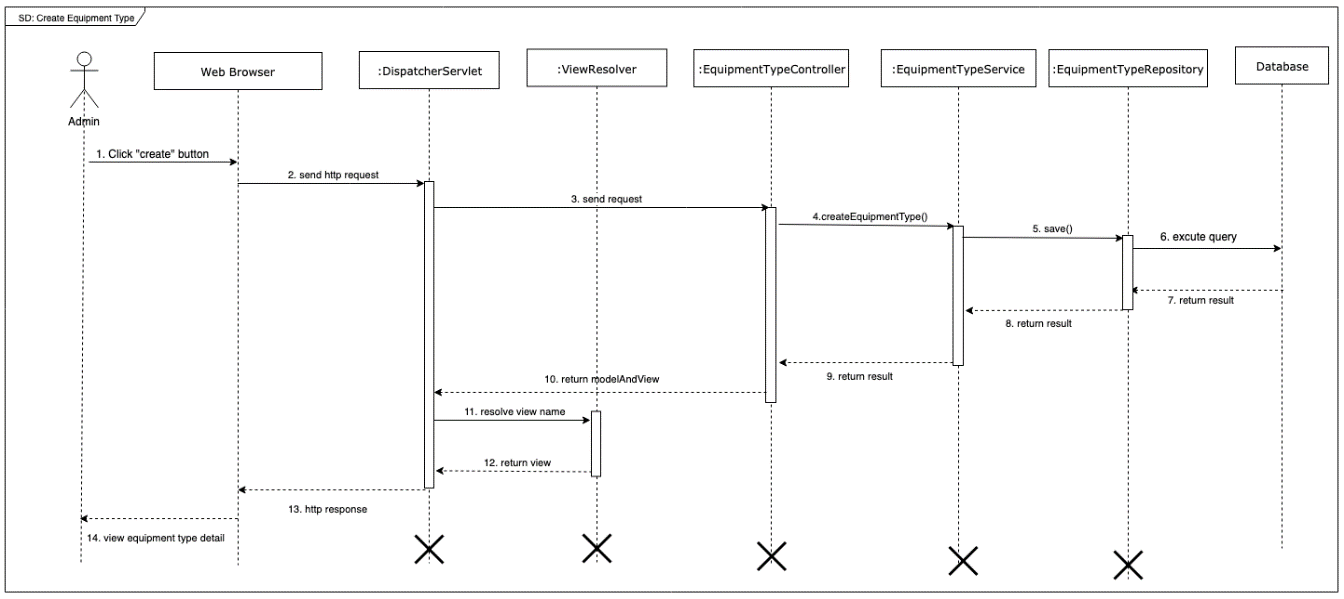


Figure 21 Sequence diagram - Add equipmen type

##### <Admin> Edit equipment type

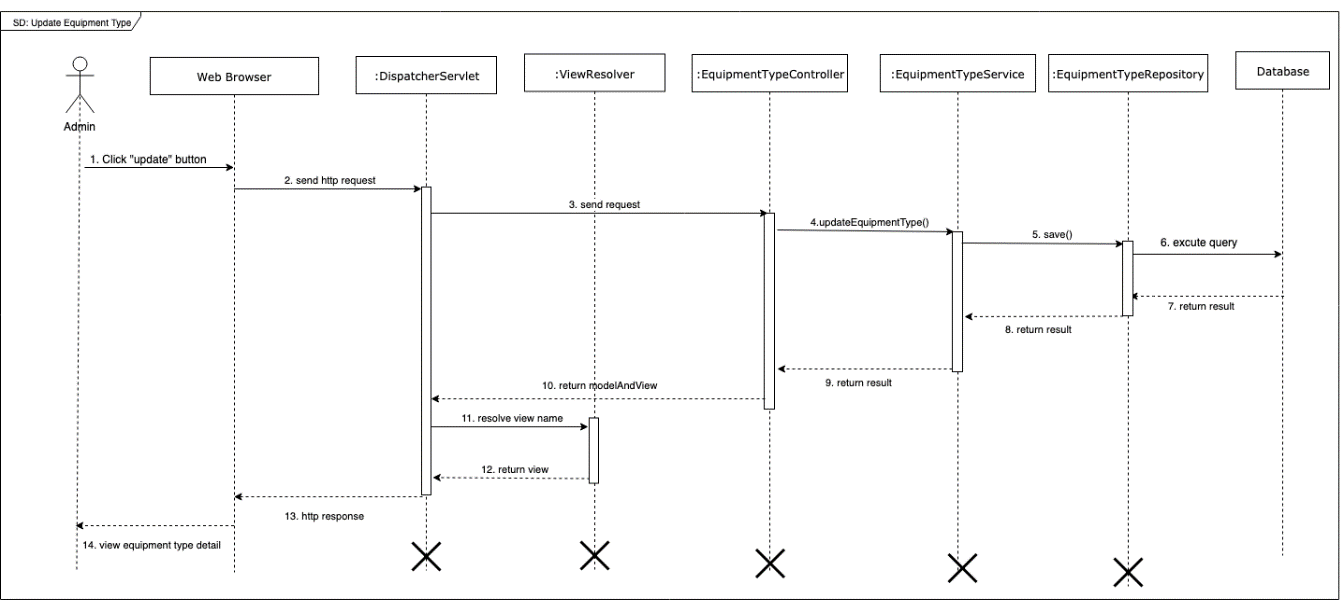


Figure 22 Edit equipment type

## Interface

### Component Interface

### User Interface Design

#### Registered User web application

##### Login



Figure 23 Login

**Fields**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** |
| 1 | Username | Username to login | No | Yes | TextBox | String |
| 2 | Password | Password to login | No | Yes | TexBox | String |

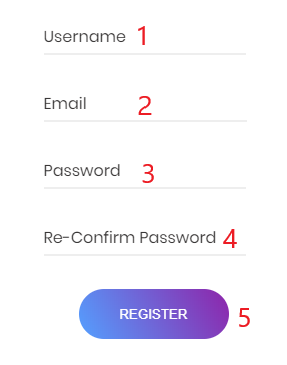
Table 101 <Fields> Login

**Buttons/Hyperlinks:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 3 | Login | Login into the system | Required | User logged in into the system |
| 4 | Forgot Password | User forget and want to take back password | N/A | User is redirected to a page for filling in new password |
| 5 | [Not a member? SIGN UP](http://localhost:3000/register) | User sign up new account | N/A | User is redirected into the sign up page |

Table 102 <Buttons/Hyperlinks> Login

##### Sign Up



**Fields**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** |
| 1 | Username | Username to sign up | No | Yes | TextBox | String |
| 2 | Email | Email to sign up | No | Yes | Textbox | String |
| 3 | Password | Password to sign up | No | Yes | TexBox | String |
| 4 | Re-Confirm password | Confirm enterred password | No | Yes | TextBox | String |

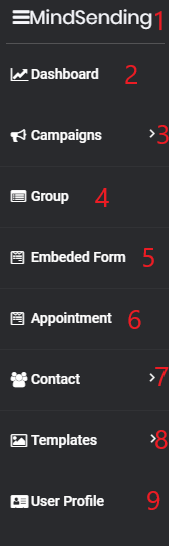
Table 103 <Fields> Sign Up

**Buttons/Hyperlinks:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 5 | Register | User sign up new account | Required | User signed up new account successfully |

Table 104 <Buttons/Hyperlinks> Sign Up

##### Main menu

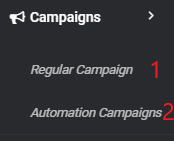


**Buttons/Hyperlinks:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | MindSending | Home | User must had been authorized | Redirect to home page |
| 2 | Dashboard | Report page | User must had been authorized | Redirect to report page |
| 3 | Campaign | Campaign page | User must had been authorized | Redirect to campaign page |
| 4 | Group | Group contact page | User must had been authorized | Redirect to group contact page |
| 5 | Embed Form | Embed form page | User must had been authorized | Redirect to embed form page |
| 6 | Appointment | Appointment page | User must had been authorized | Redirect to appointment page |
| 7 | Contact | Contact page | User must had been authorized | Redirect to contact page |
| 8 | Template | Template page | User must had been authorized | Redirect to template page |
| 9 | User profile | User profile page | User must had been authorized | Redirect to user profile page |

Table 105 <Buttons/Hyperlinks> Main menu

##### Campaign menu

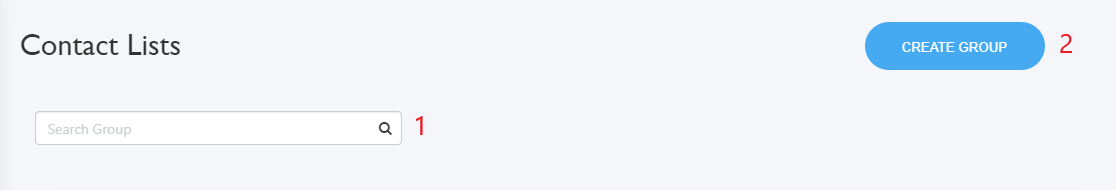


**Buttons/Hyperlinks:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | Regular Campaign | Regular Campaign page | User must had been authorized | Redirect to Regular Campaign page |
| 2 | Automation Campaigns | Automation Campaigns page | User must had been authorized | Redirect to Automation Campaigns page |

Table 106 <Buttons/Hyperlinks> Equipment menu

##### Group

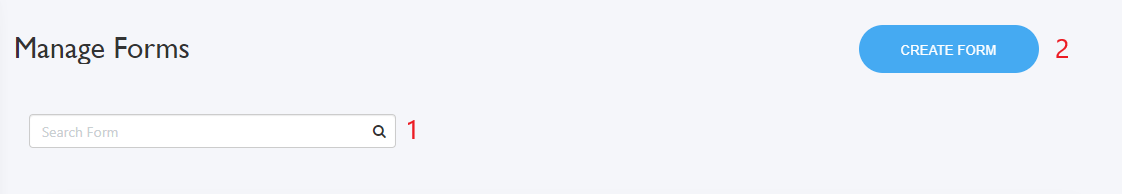


**Buttons/Hyperlinks:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | Find materials | Search group by name | N/A | Group with corresponding result |
| 2 | Create Group | Button for creating group | User must had been authorized | Redirect to create group pop-up |

Table 107 <Buttons/Hyperlinks> Material menu

##### Embed form

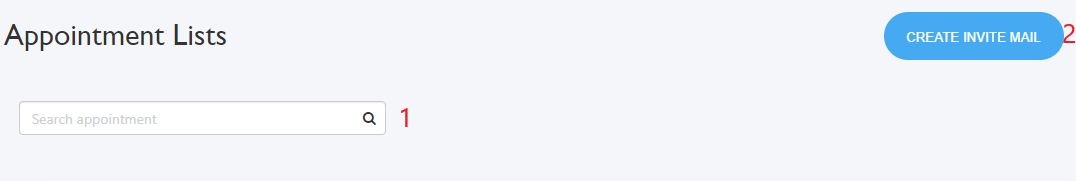


**Buttons/Hyperlinks:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | Search Form | Search form by name | N/A | Form with corresponding result |
| 2 | Create Form | Button for creating form | User must had been authorized | Redirect to create form page |

Table 108 <Buttons/Hyperlinks> Debris menu

##### Appointment

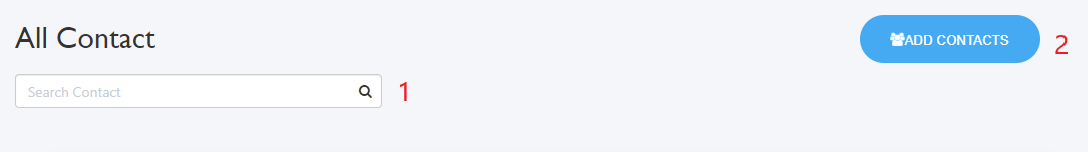


**Buttons/Hyperlinks:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | Search Appointment | Search appointment by name | N/A | appointment with corresponding result |
| 2 | Create Invite Mail | Button for creating appointment | User must had been authorized | Redirect to create appointment page |

Table 109 <Buttons/Hyperlinks> User menu

##### Contact

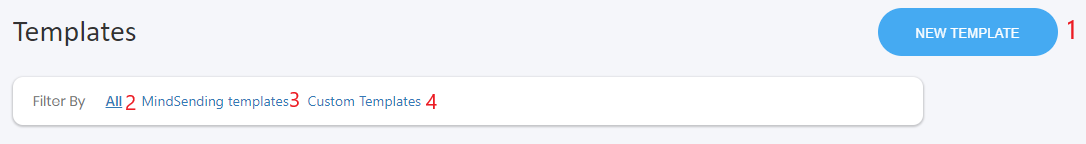


**Buttons/Hyperlinks:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | Search Contact | Search contact by name | N/A | Contact with corresponding result |
| 2 | Create Contacts | Button for creating Contact | User must had been authorized | Redirect to create Contacts page |

Table 110 <Fields> Equipment search

##### Templates



**Buttons/Hyperlinks:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | New Template | Button for creating template | User must had been authorized | Redirect to create template page |
| 2 | All | List all template | N/A | All templates are listed |
| 3 | MindSending templates | List all MindSending templates | N/A | All MindSending templates are listed |
| 4 | Custom templates | List all Custom templates | N/A | All Custom templates are listed |

## Database Design

### Entity Relationship Diagram (ERD)

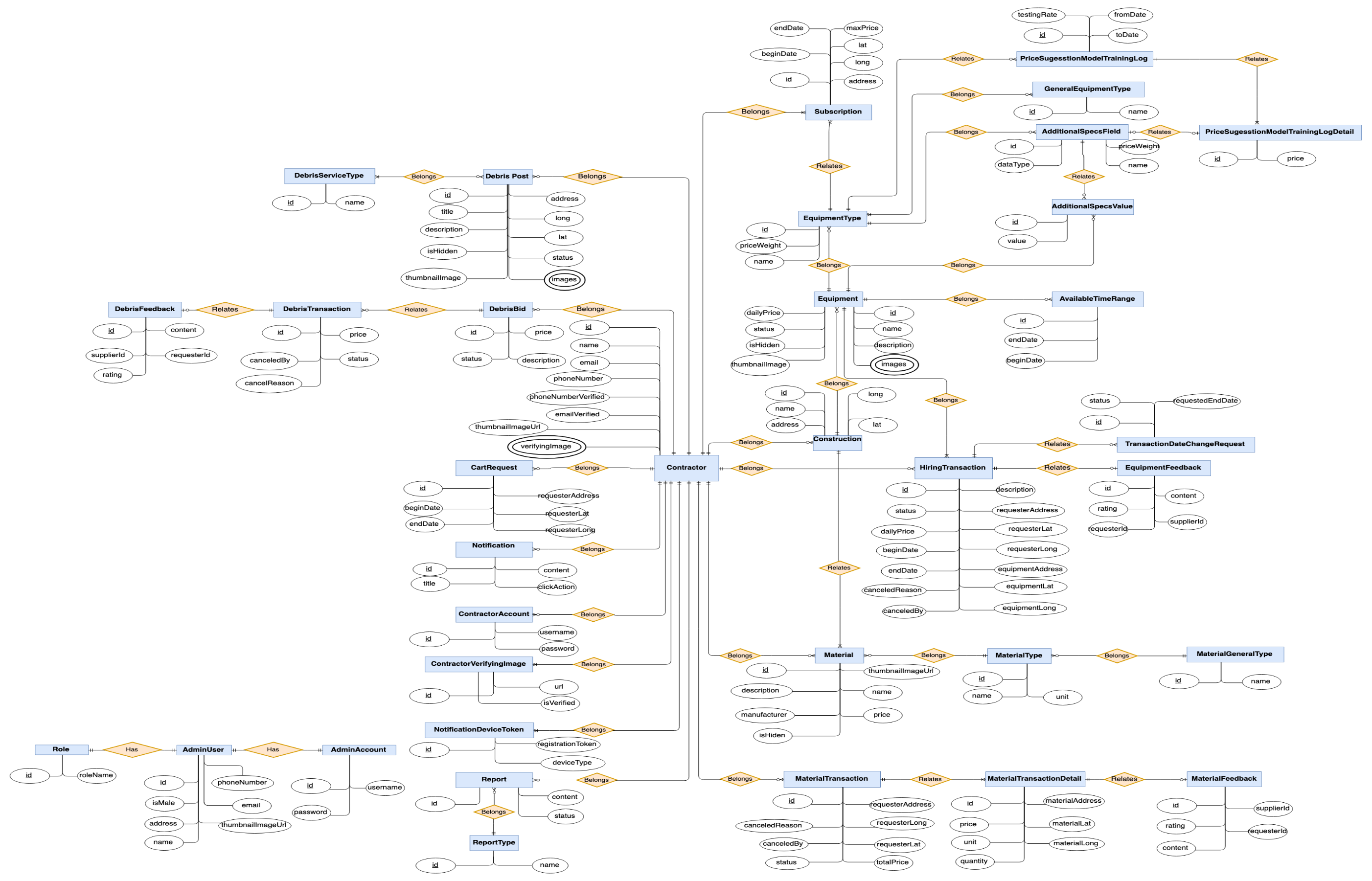


Figure 24 Entity Relation Diagram

### Data Dictionary

|  |  |
| --- | --- |
| Entity Data Dictionary: describe content of all entities | |
| **Entity Name** | **Description** |
| Registered UserAccount | Contain Registered User account information |
| AdditionalSpecsField | Contain additional specs fields information |
| AdditionalSpecsValue | Contain additional specs value information |
| AdminUser | Contain admin user profile information |
| AdminAccount | Contain admin account information |
| AvailableTimeRange | Contain available time range information of a equipment |
| Construction | Contain construction information |
| Registered User | Contain Registered User profile information |
| Equipment | Contain equipment information |
| EquipmentType | Contain equipment type information |
| Report | Contain information of reported behavior |
| ReportType | Contain report type information |
| GeneralEquipmentType | Contain equipment category information |
| HiringTransaction | Contain hiring transaction information |
| Notification | Contain notification information |
| Subscription | Contain subscription information |
| TransactionDateChangeRequest | Contain information of hiring transaction’s date changing request |
| NotificationDeviceToken | Contain information of the token of each device for notification |
| GeneralMaterialType | Contain information of material category, which has multiple material types |
| Material | Contain material information |
| MaterialTransaction | Contain material transaction information |
| MaterialType | Contain material type information |
| DebrisBid | Contain bid information for a debris post |
| DebrisFeedback | Contain feedback information for a finished debris transaction |
| DebrisPost | Contain post information for a debris service |
| DebrisServiceType | Contain debris service type information |
| DebrisTransaction | Contain debris transaction information |
| MaterialTransactionDetail | Contain material transaction detail information |
| MaterialFeedback | Contain feedback information of a finished material transaction |
| EquipmentFeedback | Contain feedback information for a finished hiring transaction |
| Registered UserVerifyingImage | Contain verifying image information |
| Role | Contain role information |
| PriceSuggestionModalTrainingLogDetail | Contain Price Suggestion Model training log detail information |
| PriceSuggestionModalTrainingLog | Contain Price Suggestion Model training log information |

Table 147 Entity Relationship Diagram Dictionary

#### Registered UserAccount

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of Registered User account |
| username | String | Private | Registered User account’s username |
| password | String | Private | Account’s password |
| Registered UserId | Integer | Private | Registered User’s id |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### AdditionalSpecsField

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an additional specs field |
| name | String | Private | Additional specs field’s name |
| dataType | String | Private | Additional specs field’s data type |
| equipmentTypeId | Integer | Private | Equipment type’s id |
| idDeleted | Tiny Integer | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### AdditionalSpecsValue

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an additional specs value |
| value | String | Private | Additional specs value’s name |
| additionalSpecsFieldId | Integer | Private | Contain information of the mentioned attribute |
| equipmentId | Integer | Private | Contain information of the mentioned attribute |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### AdminUser

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an additional specs value |
| name | String | Private | Contain information of the mentioned attribute |
| isMale | Bit | Private | Contain information of the mentioned attribute |
| phone | String | Private | Contain information of the mentioned attribute |
| email | String | Private | Contain information of the mentioned attribute |
| roleId | Integer | Private | Contain information of the mentioned attribute |
| thumbnailImage | String | Private | Thumbnail image url |
| address | String | Private | Contain information of the mentioned attribute |

#### AdminAccount

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an admin account |
| username | String | Private | Contain information of the mentioned attribute |
| password | String | Private | Contain information of the mentioned attribute |
| adminUserId | Integer | Private | Contain information of the mentioned attribute |

#### AvailableTimeRange

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an available time range |
| beginDate | Date | Private | The begin date of hiring transaction |
| endDate | Date | Private | The end date of hiring transaction |
| equipmentId | Integer | Private | Contain information of the mentioned attribute |

#### Construction

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of the construction |
| name | String | Private | Contain information of the mentioned attribute |
| address | String | Private | Contain information of the mentioned attribute |
| long | Double | Private | Contain information of the mentioned attribute |
| lat | Double | Private | Contain information of the mentioned attribute |
| isDeleted | Tiny Integer | Private | Soft delete information |
| Registered UserId | Integer | Private | Contain information of the mentioned attribute |

#### Registered User

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a Registered User |
| name | String | Private | Contain information of the mentioned attribute |
| email | Bit | Private | Contain information of the mentioned attribute |
| phoneNumber | String | Private | Contain information of the mentioned attribue |
| thumbnailImage | String | Private | Contain information of the mentioned attribue |
| isActivated | Integer | Private | Contain information of the mentioned attribute |
| createdTime | String | Private | The created time of the data |
| updatedTime | String | Private | The updated time of the data |
| isDeleted | Tiny Integer | Private | Soft delete information |

#### EquipmentImage

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an equipment image |
| url | String | Private | Contain information of the mentioned attribute |
| equipmentId | Integer | Private | Contain information of the mentioned attribute |
| createdTime | String | Private | The created time of the data |
| updatedTime | String | Private | The updated time of the data |

#### Equipment

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an equipment |
| name | String | Private | Contain information of the mentioned attribute |
| dailyPrice | Double | Private | Contain information of the mentioned attribute |
| description | String | Private | Contain information of the mentioned attribute |
| status | String | Private | Contain information of the mentioned attribute |
| thumbnailImageId | Integer | Private | Contain information of the mentioned attribute |
| isDeleted | Tiny Integer | Private | Soft delete information |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| equipmentTypeId | Integer | Private | Contain information of the mentioned attribute |
| Registered UserId | Integer | Private | Contain information of the mentioned attribute |
| constructionId | Integer | Private | Contain information of the mentioned attribute |
| equipmentImageId | Integer | Private | Contain information of the mentioned attribute |

#### EquipmentType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an equipment type |
| name | String | Private | Contain information of the mentioned attribute |
| generalEquipmentTypeId | Integer | Private | Contain information of the mentioned attribute |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| priceWeight | Double | Private | Contain information of the mentioned attribute |

#### Report

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of report |
| content | String | Private | Additional specs field’s name |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| toRegistered UserId | Datetime | Private | Contain information of the mentioned attribute |
| fromRegistered UserId | Datetime | Private | Contain information of the mentioned attribute |
| reportTypeId | Double | Private | Contain information of the mentioned attribute |
| status | String | Private | Contain information of the mentioned attribute |
| isDeleted | Bit | Private | Soft delete information |

#### ReportType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of report type |
| name | String | Private | Additional specs field’s name |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### GeneralEquipmentType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a general equipment type |
| name | String | Private | General equipment type name |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### HiringTransaction

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of hiring transaction |
| status | String | Private | Hiring transaction’s status |
| dailyPrice | Double | Private | Contain information of the mentioned attribute |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| beginDate | Date | Private | The begin date of hiring transaction |
| endDate | Date | Private | The begin date of hiring transaction |
| equipmentAddress | String | Private | Contain information of the mentioned attribute |
| equipmentLong | Double | Private | Contain information of the mentioned attribute |
| equipmentLat | Double | Private | Contain information of the mentioned attribute |
| requesterAddress | String | Private | Contain information of the mentioned attribute |
| requesterLong | Double | Private | Contain information of the mentioned attribute |
| requesterLat | Double | Private | Contain information of the mentioned attribute |
| isDeleted | Bit | Private | Soft delete information |
| equipmentId | Integer | Private | Contain information of the mentioned attribute |
| requesterId | Integer | Private | Contain information of the mentioned attribute |
| cancelReason | String | Private | Contain information of the mentioned attribute |
| canceledBy | Integer | Private | Contain information of the mentioned attribute |

#### Notification

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of notification |
| title | String | Private | Additional specs field’s name |
| content | String | Private | Contain information of the mentioned attribute |
| isRead | Tiny Integer | Private | Contain information of the mentioned attribute |
| Registered UserId | Integer | Private | Contain information of the mentioned attribute |
| clickAction | String | Private | Contain information of the mentioned attribute |

#### Subscription

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an additional specs value |
| equipmentTypeId | String | Private | Additional specs field’s name |
| Registered UserId | Bit | Private | Contain information of the mentioned attribute |
| maxPrice | Double | Private | Contain information of the mentioned attribute |
| beginDate | Date | Private | Contain information of the mentioned attribute |
| endDate | Date | Private | Contain information of the mentioned attribute |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| lat | Double | Private | Contain information of the mentioned attribute |
| long | Double | Private | Contain information of the mentioned attribute |
| maxDistance | Double | Private | Contain information of the mentioned attribute |
| address | String | Private | Contain information of the mentioned attribute |

#### TransactionDateChangeRequest

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of an additional specs value |
| requestedEndDate | Date | Private | Additional specs field’s name |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| status | String | Private | Contain information of the mentioned attribute |
| transactionId | Integer | Private | Contain information of the mentioned attribute |

#### NotificationDeviceToken

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a general equipment type |
| registrationToken | String | Private | General equipment type name |
| Registered UserId | Integer | Private | Soft delete information |
| deviceType | String | Private |  |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### GeneralMaterialType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a general material type |
| name | String | Private | General material type name |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### Material

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a material |
| name | String | Private | Material’s name |
| price | Double | Private | Material’s price |
| manufacturer | String | Private | Material’s manufacturer |
| description | String | Private | Material’s description |
| isHidden | Tiny Integer | Private | Hide material post |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| constructionId | Integer | Private | Construction id |
| Registered UserId | Integer | Private | Registered User id |
| materialTypeId | Integer | Private | Material type id |
| thumbnailImageUrl | String | Private | Material thumbnail image |

#### MaterialTransaction

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a material transaction |
| totalPrice | Double | Private | Material transaction total price |
| requesterAddress | String | Private | Material transaction requester address |
| requesterLat | Double | Private | Requester latitude |
| requesterLong | Double | Private | Requester longitude |
| status | String | Private | Material transaction’s status |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| requesterId | Integer | Private | Requester’s id |
| supplierId | Integer | Private | Supplier’s id |
| cancelReason | String | Private | Material transaction cancel reason |
| canceledBy | Integer | Private | Material transaction canceled by id |

#### MaterialType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a material type |
| name | String | Private | Material type name |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| generalMaterialTypeId | Integer | Private | Id of general material type |
| unit | Integer | Private | Define unit of material |

#### DebrisBid

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a debris bid |
| price | Double | Private | Debris bid’s price |
| status | String | Private | Debris bid’s status |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| isDeleted | Bit | Private | Soft delete information |
| supplierId | Integer | Private | Supplier’s id |
| debrisPostId | Integer | Private | Debris post’s id |
| description | String | Private | Description of debris bid |

#### DebrisFeedback

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a debris feedback |
| rating | Double | Private | Feedback rating |
| content | String | Private | Feedback content |
| requesterId | Integer | Private | Requester’s id |
| supplierId | Integer | Private | Supplier’s Id |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| debrisTransactionId | Integer | Private | Debris transaction’s id |

#### DebrisPost

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a debris post |
| title | String | Private | Debris post’s title |
| address | String | Private | Debris post’s address |
| longitude | Double | Private | Debris post’s longitude |
| latitude | Double | Private | Debris post’s latitude |
| description | String | Private | Debris post’s description |
| status | String | Private | Debris post’s status |
| isHidden | Tiny Integer | Private | Soft remove debris post |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| requesterId | Integer | Private | Requester’s id |
| thumbnailImageId | Integer | Private | Thumbnail image’s id |

#### DebrisServiceType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a debris service type |
| name | String | Private | Debris service type name |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### DebrisTransaction

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a debris transaction |
| price | Double | Private | Debris transaction’s price |
| status | String | Private | Debris transaction’s status |
| cancelReason | Integer | Private | Debris transaction’s cancel reason |
| requesterId | Integer | Private | Requester’s id |
| supplierId | String | Private | Debris transaction name |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| debrisPostId | Integer | Private | Debris post’s id |
| debrisBidId | Integer | Private | Debris bid’s id |
| canceledBy | Integer | Private | Debris transaction canceled by id |

#### DebrisImage

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a debris image |
| url | String | Private | Debris image’s url |
| debrisPostId | Bit | Private | Debris post ‘s id |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

#### MaterialTransactionDetail

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a material transaction detail |
| quantity | Double | Private | Material transaction detail’s quantity |
| unit | String | Private | Material transaction detail’s unit |
| price | Double | Private | Material transaction detail’s price |
| materialLat | Double | Private | Material transaction detail’s latitude |
| materialLong | Double | Private | Material transaction detail’s longitude |
| materialAddress | String | Private | Material transaction detail’s address |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| materialId | Integer | Private | Material’s id |
| materialTransactionId | Integer | Private | Material transaction’s id |

#### MaterialFeedback

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a material feedback |
| rating | Double | Private | Material feedback’s rating |
| content | String | Private | Material feedback’s  content |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| supplierId | Integer | Private | Supplier’s id |
| requesterId | Integer | Private | Requester’s id |
| materialTransactionDetailId | Integer | Private | Material transaction detail’s id |

#### EquipmentFeedback

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a equipment feedback |
| rating | Double | Private | Equipment feedback’s rating |
| content | String | Private | Equipment feedback’s  content |
| isDeleted | Bit | Private | Soft delete information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |
| supplierId | Integer | Private | Supplier’s id |
| requesterId | Integer | Private | Requester’s id |
| hiringTransactionId | Integer | Private | Equipment transaction’s id |

#### Registered UserVerifyingImage

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of a Registered User verifying image |
| url | String | Private | Registered User verifying image’s url |
| Registered UserId | Integer | Private | Registered User’s id |
| isVerified | Tiny Integer | Private | Verify image information |
| createdTime | Datetime | Private | The created time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

Role

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of role |
| roleName | String | Private | Role’s name |

PriceSuggestionModalTrainingLogDetail

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of price suggestion detail |
| weight | Double | Private | Contain information of the mentioned attribute |
| priceSuggestionModelTraninglog Id | Integer | Private | Contain information of the mentioned attribute |
| additionalSpecsFieldId | Integer | Private | Contain information of the mentioned attribute |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

PriceSuggestionModalTrainingLog

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | Private | Unique identifier of price suggestion |
| weight | Double | Private | Contain information of the mentioned attribute |
| toDate | Integer | Private | Contain information of the mentioned attribute |
| fromDate | Integer | Private | Contain information of the mentioned attribute |
| equipmentTypeId | Integer | Private | Contain information of the mentioned attribute |
| createdTime | Datetime | Private | The updated time of the data |
| updatedTime | Datetime | Private | The updated time of the data |

## AlgorithEMP

### Price suggestion model

#### Definition

Price suggestion model is a method to suggest the proper price using historical data

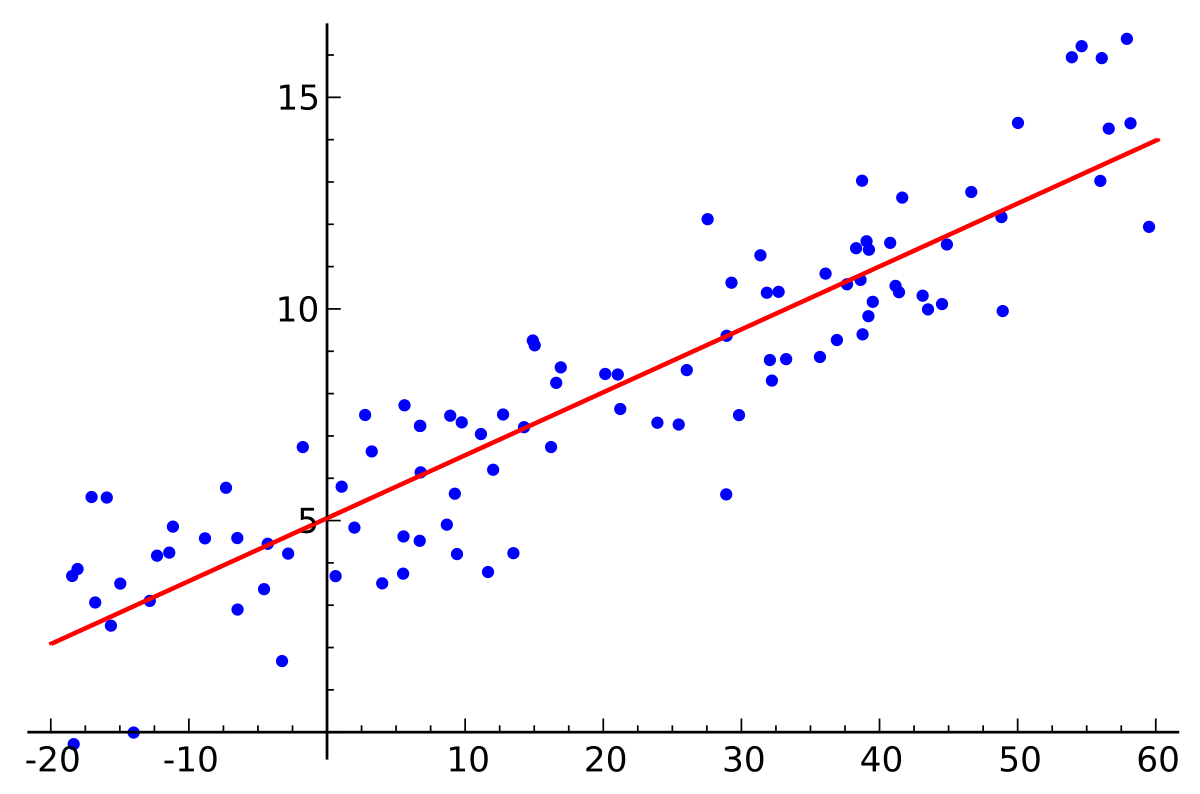
#### Problem

The system is expected to provide a price suggestion based on user input of a equipment information. Each equipment have different additional specifications as mentions in user requirements. Therefore, each equipment type must be provided with a different model for the most precision results

#### Solution

##### Linear regression

Linear regression can be understood as "finding the best fitting straight line for a set of data". Linear regression is used to show or predict the linear relationship between variables. Linear regression is simple and commonly used in machine learning fields.



**Figure 25 Linear regression plot - from Wikipedia**

The simple form of Linear Regression can be described by the following equations:

Y= β0 + β1x

Linear regression can also applied in multi-variable related probleEMP where there are more than 1 value of (x), like the current situation. Therefore, multi-variable Linear Regression can be represented by the following equation:

In above equation:

X:  Input feature value of each instance. ( matrix mxn)

θ: hypothesis parameters (co-efficient) of the model (vector (nx1))

with:

m: number of samples or dataset

n: number of features of the model

Linear regression has the following benefits:

* Benefits
* Simple
  + Fast to train
  + Can used for multiple-variable probleEMP (more than 1 variable)
  + Works in most cases
* Trade-offs:
  + Validation rate is not high because not all variable relationships are purely linear
  + Sensitive to noise (outliers)

##### Cost function for linear regression:

A cost function is a function that tell us "how good" our model is. For linear regression, one of the most commonly used cost function is Least Squared Error function:

With   
**:**prediction result

**:** actual price

**m:** number of samples used to train

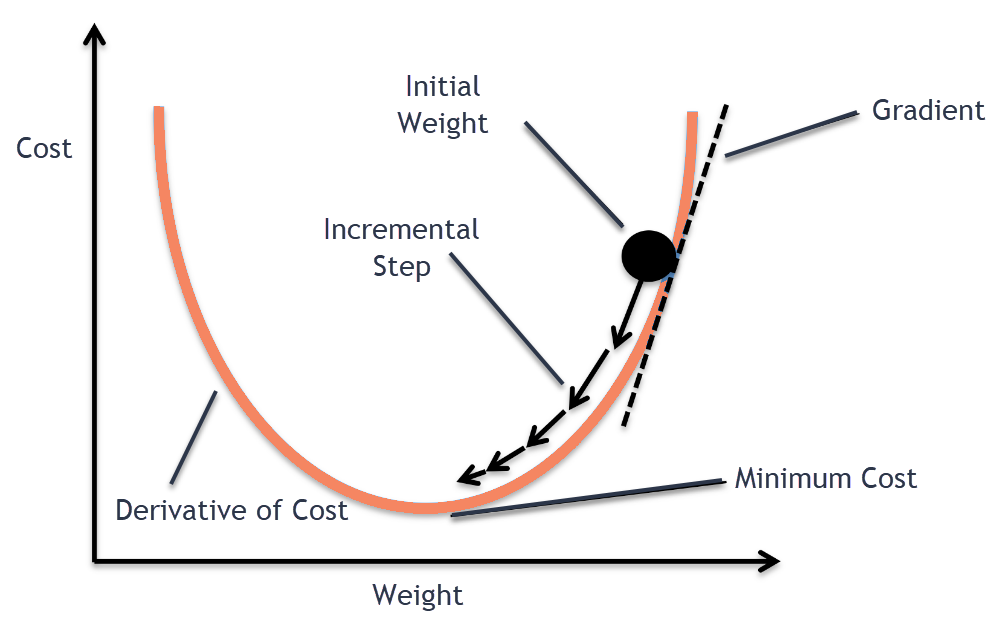
**The smaller the cost function, the better the model is**

##### Training approach with linear regression

To train a machine learning model with linear regression, there are 2 available approaches:

* 1. **Gradient descent**

Gradient descent is an optimization algorithm used to minimize the cost function by iteratively moving a point toward its local minimum point. In machine learning, we use gradient descent to update the weights of our model.



**Figure 26 Gradient Descent Plot**

Gradient descent allows us to set up hyper-parameters before training process, including number of iterations, initial weights, learning rates. Therefore, Gradient descent is highly customizable and stable with noise and outliers.

Gradient descent haves the following benefits and trade-offs:

* Benefits:
  + Can manually adjust model result by setting the number of iteration loops
  + Can endure noise thanks to high customizability
* Trade off:
  + Slow because of the iteration needed for training
  + Require initial parameters
  + Need many manual run to figure out the best result
  1. **Normal equation**

Normal Equation is an analytical approach to directly calculate the model weight θ without using Gradient Descent. The normal equation can be described as the following:

In the above equation,

**θ** : hypothesis parameters that define it the best. (vector nx1)

**X** : Input feature value of each instance. ( matrix mxn)

**Y** : Output value of each instance. ( vector mx1)

with :

**m**: number of samples or dataset

**n**: number of features of the model

Normal equation is inferred by applying derivative on the cost function, which is Least Squared Error function to find the local minimuEMP. Because of the characteristic of the squared function, the local minimum is also the global minimuEMP, which is our desired final result. Because the final hypothesis parameters is calculated directly with 1 step, the hypothesis is very fast. However, because there are no manually tuning and setting involved, Normal Equation approach can easily affected by noise and outliers in the dataset.

Using normal equation would also have benefits and trade-offs:

* Benefits:
  + Fast with 1-step learning
  + No related initial settings needed
* Trade-offs:
  + Cannot manually adjusted, so it is very sensible to noise

##### Implementation

We choose to use **normal equations** approach instead of **gradient descent** because:

* Faster
* Because of the project scope, there would be no human involved

For each equipment type, we would train a separate model

Input:

* X: additional specs values of all the equipments of the equipment type
* Y: price of the equipments

Output:

* : hypothesis parameters of the model

Hypothesis parameters would be used to calculate suggested price by linear regression equation:

With:

* **X**: input values of the equipment need to suggest price
* hypothesis parameters of the model
* : suggested price

# System implementation & Test

## Introduction

### Overview

This section provides in detail all necessary information about implementation information and testing procedure of EMP includes test plans, test cases, test result and risks estimations.

### Test Approach

* Goal: To test the whole system based on the core workflow.
* Method: System Testing, Black-box Testing.

## Database Relationship Diagram

### Physical Diagram

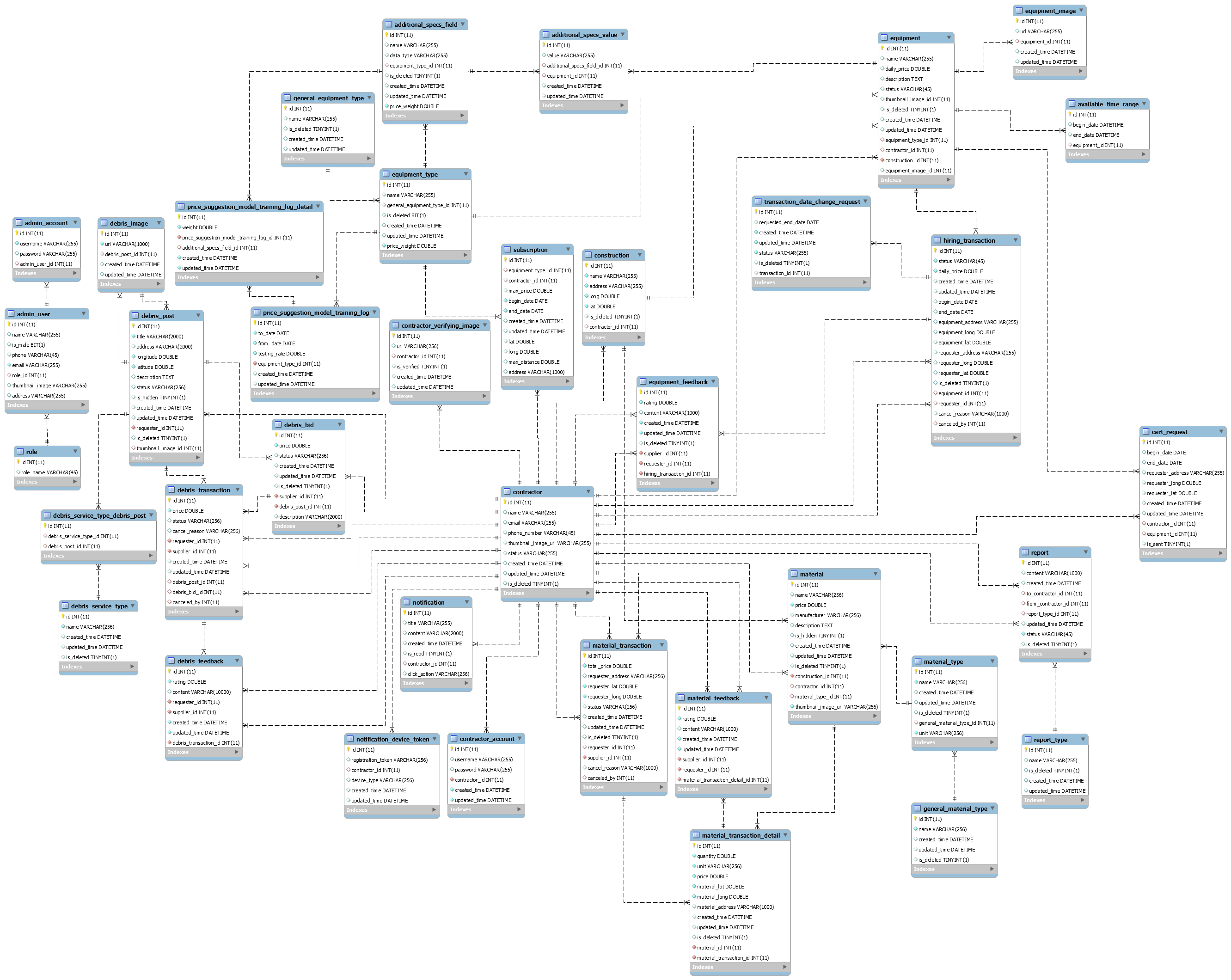


Figure 27: Physical Diagram

### Data dictionary

|  |  |
| --- | --- |
| **Data dictionary: describe content of all tables** | |
| **Entity name** | **Description** |
| additional\_specs\_field | Describe all additional specifications field of each equipment type in the system |
| additional\_specs\_value | Describe all additional specifications value of each equipment in the system |
| admin\_account | Describe all admin account in the system |
| admin\_user | Describe all admin user in the system |
| available\_time\_range | Describe all available time range of each equipment in the system |
| construction | Describe all constructions of each Registered User in the system |
| Registered User | Describe all Registered User in the system |
| Registered User\_account | Describe all accounts of Registered Users in the system |
| Registered User\_verifying\_image | Describe all images to verify account of each Registered User in the system |
| debris\_bid | Describe all bids of each debris post in the system |
| debris\_feedback | Describe all feedbacks of each debris post in the system |
| debris\_image | Describe all images of each debris post in the system |
| debris\_post | Describe all debris post in the system |
| debris\_service\_type | Describe all debris service types in the system |
| debris\_service\_type\_debris\_post | Describe all relations between debris posts and debris service types in the system |
| debris\_transaction | Describe all debris transactions in the system |
| equipment | Describe all equipments in the system |
| equipment\_feedback | Describe all feedbacks of each equipment transaction in the system |
| equipment\_image | Describe all images of each equipment in the system |
| equipment\_type | Describe all equipment types in the system |
| general\_equipment\_type | Describe all equipment type categories in the system |
| general\_material\_type | Describe all material type categories in the system |
| hiring\_transaction | Describe all hiring equipment transactions in the system |
| material | Describe all materials in the system |
| material\_feedback | Describe all feebacks of each detail of material transaction in the system |
| material\_transaction | Describe all material transactions in the system |
| material\_transaction\_detail | Describe all details of each material transaction in the system |
| material\_type | Describe all material types in the system |
| notification | Describe all notifications of each Registered User in the system |
| notification\_device\_token | Describe all device’s notification token of each Registered User in the system |
| report | Describe all reports about bad behaviors of each Registered User in the system |
| report\_type | Describe all bad behavior types in the system |
| role | Describe all role in the system |
| subscription | Describe all subscription for equipment in the system |
| transaction\_date\_change\_request | Describe all requests about changing transaction’s date in the system |
| PriceSuggestionModalTrainingLogDetail | Describe price Suggestion Model training log detail in the system |
| PriceSuggestionModalTrainingLog | Contain Price Suggestion Model training log in the system |

Figure 28: Data Dictionary

## Test Plan

The purpose of this section is to verify and ensure that EMPSC meets its design specification and other requirements from user. The following part will describe which features to be tested and which will not.

### Features to be tested

* Registered User:
  + Create new regular campaign
  + Create new automation campaign
  + Create new group
  + Create new embedded form
  + Create new appointment
  + Create new contact
  + Create new template
  + Update profile

### Features not to be tested

* Admin:
  + Reset password
  + Reply feedback
  + View statistic

## System Testing Test Case

### Registered User testing case:

#### Create new regular campaign:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | Registered User request for creating a new regular campaign | 1. Click Create Campaign  2. Select option template or simple text  3. Fill nothing in textbox  4. Click “Begin” button | Begin button is disabled. Can not click to begin. | N/A | Pass | 25/07/2019 |  |
| EMP\_02 | Registered User request for creating a new regular campaign | 1. Click Create Campaign  2. Select option template or simple text  3. Fill campaign name in textbox  4. Click “Begin” button | User will be redirected to design campaign page | N/A | Pass | 25/07/2019 |  |
| EMP\_03 | Registered User request for creating a new regular campaign | 1. Choose List  2. Fill in Sender Name  3. Click “Design Email” button | Design Email button is disabled. Can not click to design. | N/A | Pass | 25/07/2019 |  |
| EMP\_04 | Registered User request for creating a new regular campaign | 1. Choose List  2. Fill in Sender Name  3. Fill in Email Address  4. Click “Design Email” button | Design Email button is disabled. Can not click to design. | N/A | Pass | 25/07/2019 |  |
| EMP\_05 | Registered User request for creating a new regular campaign | 1. Choose List  2. Fill Sender Name  3. Fill in Email Address  4. Fill in Subject  5. Click “Design Email” button  6. Choose and edit template  7. Save campaign | New regular campaign is created. | N/A | Pass | 25/07/2019 |  |

#### Create new automation campaign:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | Registered User request for creating a new automation campaign | 1. Click Create Automation  2. Fill nothing in textbox  3. Click “Begin” button | Begin button is disabled. Can not click to begin. | N/A | Pass | 25/07/2019 |  |
| EMP\_02 | Registered User request for creating a new automation campaign | 1. Click Create Automation  2. Fill campaign name in textbox  3. Click “Begin” button | User will be redirected to design automation campaign page | N/A | Pass | 25/07/2019 |  |

#### Create new group:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | Registered User request for creating a new group | 1. Click Create Group  2. Fill nothing in “New Name”  3. Click “Create” button | Create button is disabled. Can not click to create. | N/A | Pass | 25/07/2019 |  |
| EMP\_02 | Registered User request for creating a new group | 1. Click Create Group  2. Fill in “New Name”  3. Click “Begin” button | New group will be created | N/A | Pass | 25/07/2019 |  |

#### Create new embedded form:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | Registered User request for creating a new embedded form | 1. Click Create Form  2. Fill nothing in “Form Name”  3. Click “Save this form” button | System shows “You must fill in name for this form”. | N/A | Pass | 25/07/2019 |  |
| EMP\_02 | Registered User request for creating a new embedded form | 1. Click Create Group  2. Fill in “Form Name”  3. None of group is chosen  4. Click “Save this form” button | System shows “You must choose a group” | N/A | Pass | 25/07/2019 |  |
| EMP\_03 | Registered User request for creating a new embedded form | 1. Click Create Group  2. Fill in “Form Name”  3. Choose one group  4. Click “Save this form” button | New embedded form is created | N/A | Pass | 25/07/2019 |  |

#### Create new appointment:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | Registered User request for creating a new appointment | 1. Click Create appointment  2. Fill nothing in appointment name  3. Click “Save draft” button | System shows “You need to enter a name that’s at least 3 character long” | N/A | Pass | 25/07/2019 |  |
| EMP\_02 | Registered User request for creating a new appointment | 1. Click Create appointment  2. Fill in appointment name  3. Date and time are not chosen  4. Click “Save draft” button | System shows “Enter date and time of your appointment” | N/A | Pass | 25/07/2019 |  |
| EMP\_03 | Registered User request for creating a new appointment | 1. Click Create appointment  2. Fill in appointment name  3. Date and time are chosen  4. Groups are not chosen  5. Click “Save draft” button | System shows “Choose at least one group” | N/A | Pass | 25/07/2019 |  |
| EMP\_04 | Registered User request for creating a new appointment | 1. Click Create appointment  2. Fill in appointment name  3. Date and time are chosen  4. Groups are chosen  5. Sender name and email address are not filled in  6. Click “Save draft” button | System shows “Enter sender name and email address” | N/A | Pass | 25/07/2019 |  |
| EMP\_05 | Registered User request for creating a new appointment | 1. Click Create appointment  2. Fill in appointment name  3. Date and time are chosen  4. Groups are chosen  5. Sender name and email address are filled in  6. Subject is not filled in  7. Click “Save draft” button | System shows “This field can not be empty” | N/A | Pass | 25/07/2019 |  |
| EMP\_06 | Registered User request for creating a new appointment | 1. Click Create appointment  2. Fill in appointment name  3. Date and time are chosen  4. Groups are chosen  5. Sender name and email address are filled in  6. Subject is filled in  7. Click “Save draft” button | New appointment is created | N/A | Pass | 25/07/2019 |  |

#### Create new contact:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | Registered User request for creating a new contact | 1. Click Add contact  2. Choose “Manual add”  3. Choose option “add contacts” or “add contacts and include in an existing list”  4. Fill nothing in email field  5. Click “Save Contact” button | System shows “Please enter email address” | N/A | Pass | 25/07/2019 |  |
| EMP\_02 | Registered User request for creating a new contact | 1. Click Add contact  2. Choose “Manual add”  3. Choose option “add contacts” or “add contacts and include in an existing list”  4. Fill in email field  5. Click “Save Contact” button | New contacts are created | N/A | Pass | 25/07/2019 |  |
| EMP\_03 | Registered User request for creating a new contact | 1. Click Add contact  2. Choose “Upload CSV”  3. Choose option “add contacts” or “add contacts and include in an existing list”  4. CSV file is not chosen  5. Click “Save Contact” button | System shows “Please choose a CSV file” | N/A | Pass | 25/07/2019 |  |
| EMP\_04 | Registered User request for creating a new contact | 1. Click Add contact  2. Choose “Upload CSV”  3. Choose option “add contacts” or “add contacts and include in an existing list”  4. CSV file is chosen  5. Click “Save Contact” button | New contacts are created | N/A | Pass | 25/07/2019 |  |

#### Create new template:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | Registered User request for creating a new template | 1. Click New Template  2. Drag and drop nothing component  3. Click “Save template” button | System shows “You must drag and drop at least one component” | N/A | Pass | 25/07/2019 |  |
| EMP\_02 | Registered User request for creating a new template | 1. Click New Template  2. Drag and drop component  3. Fill nothing in template name  4. Click “Save template” button | System shows “Please enter a name for this template” | N/A | Pass | 25/07/2019 |  |
| EMP\_03 | Registered User request for creating a new template | 1. Click New Template  2. Drag and drop component  3. Fill in template name  4. Click “Save template” button | A new template is created | N/A | Pass | 25/07/2019 |  |

#### Update profile:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | Registered User request for updating profile | 1. Click User profile  2. Edit nothing  3. Click “Update profile” button | System shows “Nothing has been changed” | N/A | Pass | 25/07/2019 |  |
| EMP\_02 | Registered User request for updating profile | 1. Click User profile  2. Edit one or more field  3. Click “Update profile” button | System shows “User profile has been updated” | N/A | Pass | 25/07/2019 |  |

* + 1. テストケース

#### 新しい通常のキャンペーンを作成する:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | 新しい通常のキャンペーンを作成するための登録ユーザーのリクエスト | 1. [キャンペーンの作成]をクリックします  2.オプションテンプレートまたは単純なテキストを選択します  3.テキストボックスに何も記入しない  4.「開始」ボタンをクリックします | 開始ボタンは無効です。クリックして開始することはできません。 | N/A | パス | 25/07/2019 |  |
| EMP\_02 | 新しい通常のキャンペーンを作成するための登録ユーザーのリクエスト | 1. [キャンペーンの作成]をクリックします  2.オプションテンプレートまたは単純なテキストを選択します  3.テキストボックスにキャンペーン名を入力します  4.「開始」ボタンをクリックします | ユーザーはデザインキャンペーンページにリダイレクトされます | N/A | パス | 25/07/2019 |  |
| EMP\_03 | 新しい通常のキャンペーンを作成するための登録ユーザーのリクエスト | 1.リストを選択  2.送信者名を入力します  3. [メールのデザイン]ボタンをクリックします | [デザインメール] ボタンは無効になっています。クリックして設計することはできません。 | N/A | パス | 25/07/2019 |  |
| EMP\_04 | 新しい通常のキャンペーンを作成するための登録ユーザーのリクエスト | 1.リストを選択  2.送信者名を入力します  3.メールアドレスを入力してください  4. [メールのデザイン]ボタンをクリックします | [デザインメール] ボタンは無効になっています。クリックして設計することはできません。 | N/A | パス | 25/07/2019 |  |
| EMP\_05 | 新しい通常のキャンペーンを作成するための登録ユーザーのリクエスト | 1.リストを選択  2.送信者名の入力  3.メールアドレスを入力してください  4.件名を記入  5.「メールのデザイン」ボタンをクリックします  6.テンプレートを選択して編集する  7.キャンペーンを保存 | 新しい通常のキャンペーンが作成されます。 | N/A | パス | 25/07/2019 |  |

#### 新しい自動化キャンペーンを作成する:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | 新しい自動化キャンペーンを作成するための登録ユーザーのリクエスト | 1. [オートメーションの作成]をクリックします  2.テキストボックスに何も記入しない  3.「開始」ボタンをクリックします | 開始ボタンは無効です。クリックして開始することはできません。 | N/A | パス | 25/07/2019 |  |
| EMP\_02 | 新しい自動化キャンペーンを作成するための登録ユーザーのリクエスト | 1. [オートメーションの作成]をクリックします  2.テキストボックスにキャンペーン名を入力します  3.「開始」ボタンをクリックします | ユーザーはデザイン自動化キャンペーンページにリダイレクトされます | N/A | パス | 25/07/2019 |  |

#### 新しいグループを作成する:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | 新しいグループを作成するための登録ユーザーのリクエスト | 1. [グループの作成]をクリックします  2.「新しい名前」には何も記入しません  3.「作成」ボタンをクリックします | [作成] ボタンは無効です。クリックして作成できません。 | N/A | パス | 25/07/2019 |  |
| EMP\_02 | 新しいグループを作成するための登録ユーザーのリクエスト | 1. [グループの作成]をクリックします  2.「新しい名前」を入力します  3.「開始」ボタンをクリックします | 新しいグループが作成されます | N/A | パス | 25/07/2019 |  |

#### 新しい埋め込みフォームの作成:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | 新しい埋め込みフォームを作成するための登録ユーザーのリクエスト | 1.フォームの作成をクリックします  2.「フォーム名」に何も記入しない  3.「このフォームを保存」ボタンをクリックします | システムには「このフォームの名前を入力する必要があります」と表示されます。 | N/A | パス | 25/07/2019 |  |
| EMP\_02 | 新しい埋め込みフォームを作成するための登録ユーザーのリクエスト | 1. [グループの作成]をクリックします  2.「フォーム名」を入力します  3.グループが選択されていません  4. [このフォームを保存]ボタンをクリックします | システムに「グループを選択する必要があります」と表示される | N/A | パス | 25/07/2019 |  |
| EMP\_03 | 新しい埋め込みフォームを作成するための登録ユーザーのリクエスト | 1. [グループの作成]をクリックします  2.「フォーム名」を入力します  3. 1つのグループを選択します  4. [このフォームを保存]ボタンをクリックします | 新しい埋め込みフォームが作成されます | N/A | パス | 25/07/2019 |  |

#### 新しい予定の作成:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | 登録済みユーザーが新しい予定を作成するためのリクエスト | 1. [予定を作成]をクリックします  2.予約名に何も記入しない  3.「下書きを保存」ボタンをクリックします | システムに「少なくとも3文字以上の名前を入力する必要があります」と表示される | N/A | パス | 25/07/2019 |  |
| EMP\_02 | 登録済みユーザーが新しい予定を作成するためのリクエスト | 1. [予定を作成]をクリックします  2.予定名を入力します  3.日付と時刻が選択されていません  4.「下書きを保存」ボタンをクリックします | システムに「予約の日時を入力してください」と表示される | N/A | パス | 25/07/2019 |  |
| EMP\_03 | 登録済みユーザーが新しい予定を作成するためのリクエスト | 1. [予定を作成]をクリックします  2.予定名を入力します  3.日付と時刻が選択されます  4.グループが選択されていません  5.「下書きを保存」ボタンをクリックします | システムに「少なくとも1つのグループを選択してください」と表示される | N/A | パス | 25/07/2019 |  |
| EMP\_04 | 登録済みユーザーが新しい予定を作成するためのリクエスト | 1. [予定を作成]をクリックします  2.予定名を入力します  3.日付と時刻が選択されます  4.グループが選択されます  5.送信者名とメールアドレスが入力されていません  6.「下書きを保存」ボタンをクリックします | システムに「送信者名とメールアドレスを入力してください」と表示される | N/A | パス | 25/07/2019 |  |
| EMP\_05 | 登録済みユーザーが新しい予定を作成するためのリクエスト | 1. [予定を作成]をクリックします  2.予定名を入力します  3.日付と時刻が選択されます  4.グループが選択されます  5.送信者名とメールアドレスが入力されます  6.件名は記入されていません  7.「下書きを保存」ボタンをクリックします | システムに「このフィールドは空にできません」と表示される | N/A | パス | 25/07/2019 |  |
| EMP\_06 | 登録済みユーザーが新しい予定を作成するためのリクエスト | 1. [予定を作成]をクリックします  2.予定名を入力します  3.日付と時刻が選択されます  4.グループが選択されます  5.送信者名とメールアドレスが入力されます  6.件名が記入されています  7.「下書きを保存」ボタンをクリックします | 新しい予定が作成されます | N/A | パス | 25/07/2019 |  |

#### 新しい連絡先の作成:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | 登録済みユーザーが新しい連絡先を作成するためのリクエスト | 1. [連絡先を追加]をクリックします  2.「手動追加」を選択します  3.「連絡先を追加」または「連絡先を追加して既存のリストに含める」オプションを選択します  4.メールフィールドに何も記入しない  5.「連絡先を保存」ボタンをクリックします | システムに「メールアドレスを入力してください」と表示される | N/A | パス | 25/07/2019 |  |
| EMP\_02 | 登録済みユーザーが新しい連絡先を作成するためのリクエスト | 1. [連絡先を追加]をクリックします  2.「手動追加」を選択します  3.「連絡先を追加」または「連絡先を追加して既存のリストに含める」オプションを選択します  4.メールフィールドに入力  5.「連絡先を保存」ボタンをクリックします | 新しい連絡先が作成されます | N/A | パス | 25/07/2019 |  |
| EMP\_03 | 登録済みユーザーが新しい連絡先を作成するためのリクエスト | 1. [連絡先を追加]をクリックします  2.「CSVをアップロード」を選択します  3.「連絡先を追加」または「連絡先を追加して既存のリストに含める」オプションを選択します  4. CSVファイルが選択されていません  5.「連絡先を保存」ボタンをクリックします | システムに「CSVファイルを選択してください」と表示される | N/A | パス | 25/07/2019 |  |
| EMP\_04 | 登録済みユーザーが新しい連絡先を作成するためのリクエスト | 1. [連絡先を追加]をクリックします  2.「CSVをアップロード」を選択します  3.「連絡先を追加」または「連絡先を追加して既存のリストに含める」オプションを選択します  4. CSVファイルが選択されます  5.「連絡先を保存」ボタンをクリックします | 新しい連絡先が作成されます | N/A | パス | 25/07/2019 |  |

#### 新しいテンプレートの作成:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | 登録済みユーザーが新しいテンプレートを作成するためのリクエスト | 1. [新しいテンプレート]をクリックします  2.何もドラッグアンドドロップするコンポーネント  3.「テンプレートを保存」ボタンをクリックします | システムに「少なくとも1つのコンポーネントをドラッグアンドドロップする必要があります」と表示される” | N/A | パス | 25/07/2019 |  |
| EMP\_02 | 登録済みユーザーが新しいテンプレートを作成するためのリクエスト | 1. [新しいテンプレート]をクリックします  2.コンポーネントをドラッグアンドドロップする  3.テンプレート名に何も記入しない  4.「テンプレートを保存」ボタンをクリックします | システムに「このテンプレートの名前を入力してください」と表示される | N/A | パス | 25/07/2019 |  |
| EMP\_03 | 登録済みユーザーが新しいテンプレートを作成するためのリクエスト | 1. [新しいテンプレート]をクリックします  2.コンポーネントをドラッグアンドドロップする  3.テンプレート名を入力します  4.「テンプレートを保存」ボタンをクリックします | 新しいテンプレートが作成されます | N/A | パス | 25/07/2019 |  |

#### プロファイルの更新:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test case description** | **Test case procedure** | **Expected output** | **Inter-dependence test case** | **Result** | **Test date** | **Note** |
| EMP\_01 | 登録ユーザーのプロファイル更新リクエスト | 1. [ユーザープロフィール]をクリックします  2.何も編集しない  3.「プロファイルの更新」ボタンをクリックします | システムに「何も変更されていません」と表示される | N/A | パス | 25/07/2019 |  |
| EMP\_02 | 登録ユーザーのプロファイル更新リクエスト | 1. [ユーザープロフィール]をクリックします  2. 1つ以上のフィールドを編集します  3.「プロファイルの更新」ボタンをクリックします | システムに「ユーザープロファイルが更新されました」と表示される | N/A | パス | 25/07/2019 |  |

# Software User’s Manual

## Installation Guide

### Setting up environment at server

#### Hardware requirements

|  |  |  |
| --- | --- | --- |
| **Server** | **Minimum Requirements** | **Recommended** |
| **Internet Connection** | Cable, Wi-Fi (8 Mbps) | Cable, Wi-Fi (50 Mbps or more) |
| **Operating System** | CentOS 6 | CentOS 7 or Ubuntu 18 |
| **Computer Processor** | Intel® Xeon ® 1.4GHz | Intel® Xeon ® Quad Core (12M Cache, 2.50 GHz) |
| **Computer Memory** | 2GB RAM | 8GB or more |
| **Storage space** | 3GB | 5GB or more |

#### Software requirement

|  |  |  |
| --- | --- | --- |
| **Name** | **Name/Version** | **Description** |
| **Environment** | Java EE 8.0, Node v10 | Specification for developing web application |
| **Dependency manager, build tool** | Yarn v1.15.2, Gradle 5.3.1, Maven 3.6.1 | Dependency managers, build tools for development |
| **Operating System** | Window 10 or Ubuntu 16 or MacOS | Operating system and platform for development |
| **Modeling tool** | StarUML, Draw.io | Used to design diagram |
| **IDE** | IntelliJ IDEA 2018.1.5, Visual Studio Code 1.27.2 | Programming tools |
| **DBEMP** | MySQL 8.0 | Used to create & manage the database for development |
| **Web browser** | Chrome 69 or above. | Testing browser |
| **Mobile OS** | iOS 10 or above | Testing mobile device operating system |
| **Deployment Server** | Docker Compose 1.24.0 or above | Deployment environment for production |

### Deployment at server side

#### Prepare deployment package

* **Development**

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

* + Download and install IntelliJ IDEA (Includes Gradle and Maven)  
    Link download:  
    [https://www.jetbrains.com/idea/download/#section=windows](https://www.jetbrains.com/idea/download/)
  + Download and install MySQL 8.0  
    Link download:  
    <https://dev.mysql.com/downloads/installer/>
  + Download and install NodeJS v10 (Inludes NPM 6)  
    Link download:  
    <https://nodejs.org/en/download/>
  + Download and install Yarn  
    Link download:  
    <https://yarnpkg.com/lang/en/docs/install/#windows-stable>
* **Production**  
  Download and install Docker Compose:  
  Link download: <https://docs.docker.com/compose/install/>

#### Configure Server before deploy

* Development:
  + MySQL 8.0: create new database and new user and grant all privileges on the new database to the new user.
  + Web API application:
    - Copy capstone-EMP/client-api/src/main/tomee/deploy-conf/ tomee.xml to capstone-EMP/client-api/src/main/tomee/deploy/tomee.xml
    - Edit connection string
  + Web Admin application:
    - Edit connection string in file: capstone-EMP/webadmin/src/main/resources/application.properties

### Deploy web application on server

#### Development

* Database:
  + Restore database from file capstone-EMP/database/init.sql
* Web API application:
  + Open IntelliJ IDEA and open folder capstone-EMP/client-api as a project
  + At the top menu, click Run button
* Web admin application:
  + Open IntelliJ IDEA and open folder capstone-EMP/webadmin as a project
  + At the top menu, click Run button
* Web client:
  + Open VSCode and open folder capstone-EMP/client-web
  + Open intergrated terminal and run command: “yarn” to install dependency packages and “yarn dev” to start web client for development

#### Production

* For web application (includes Web API, Web Admin, Web Client, database):
  + Open terminal at project folder and run command: “npm start”
  + Install dependency packages with command: “npm install”

## User Guide

### Web application

# Appendix

1. **Spring Framework** *[Online]* Available: https:/spring.io/docs/reference.
2. **MySQL** *[Online]* Available: <https://dev.mysql.com/doc/>.
3. **ReactJS** *[Online]* Available: <https://reactjs.org/docs/getting-started.html>
4. **React Native** *[Online]* Available: <https://facebook.github.io/react-native/docs/getting-started>
5. **IntelliJ IDEA** *[Online]* Available: <https://www.jetbrains.com/idea/documentation/>
6. **WebStorm** *[Online]* Available: <https://www.jetbrains.com/webstorm/features/>
7. **Visual Studio Code** *[Online]* Available: <https://code.visualstudio.com/docs>
8. **Nodejs** *[Online]* Available: <https://nodejs.org/en/docs/>