Scope of Project

1. General shop that will cover all kinds of grocery & and food items.
2. A variation will be specialized for the furniture Industry.

General Concept/Requirements for Development

1. Cashing System
2. Cashing on/Off from the front end
3. CDN System (Aws/Azure)
4. Should have a plugin installation & and an on/Off system
5. Demo Mode Active/Inactive
6. Translation
7. Left To Right Option
8. Admin Option to central/vendor/Mixed Item, if central then the store will not able to add any item.
9. Central Date format Option
10.

1. Introduction

1.1. Purpose

1.2. Dynamic Futures Overview

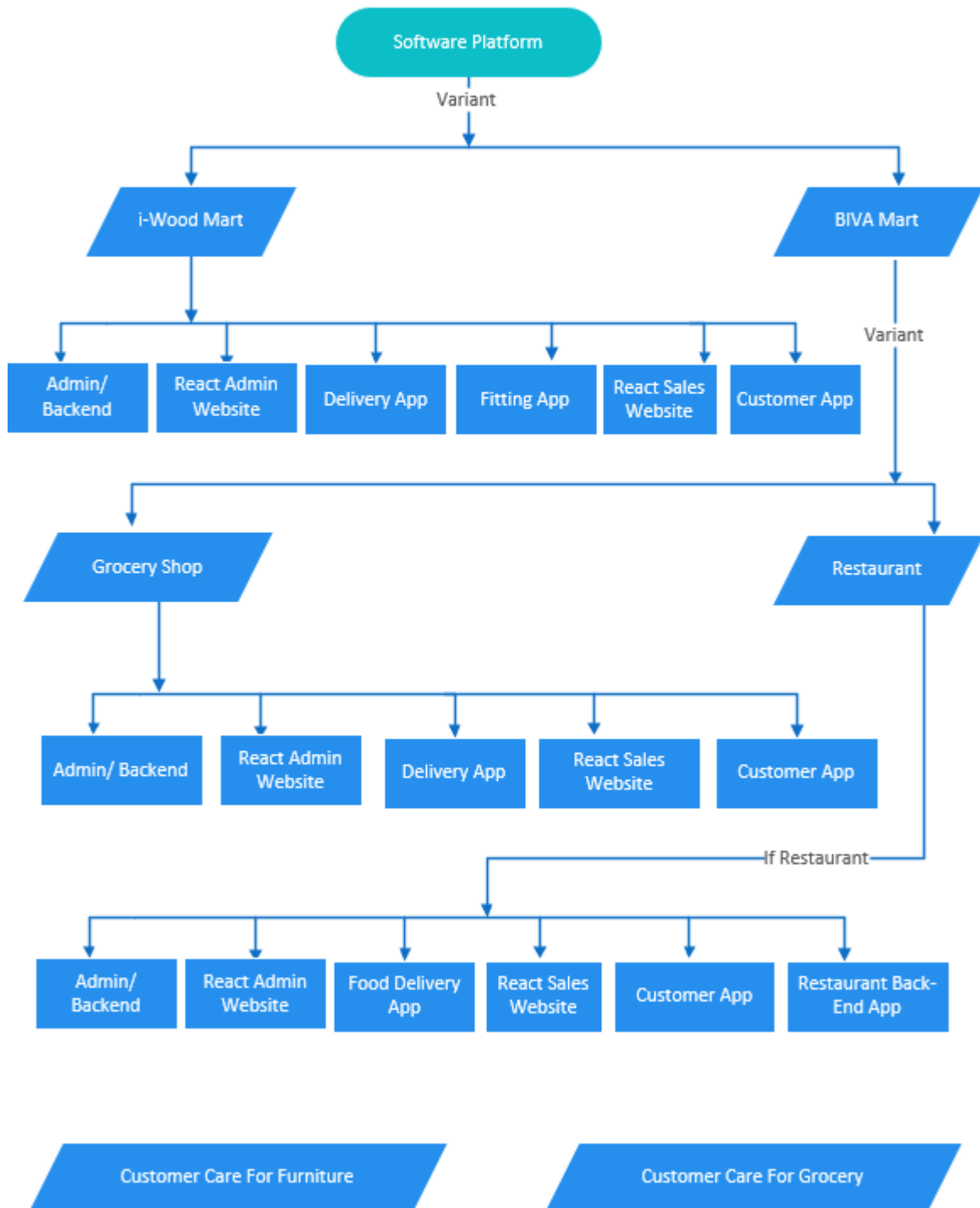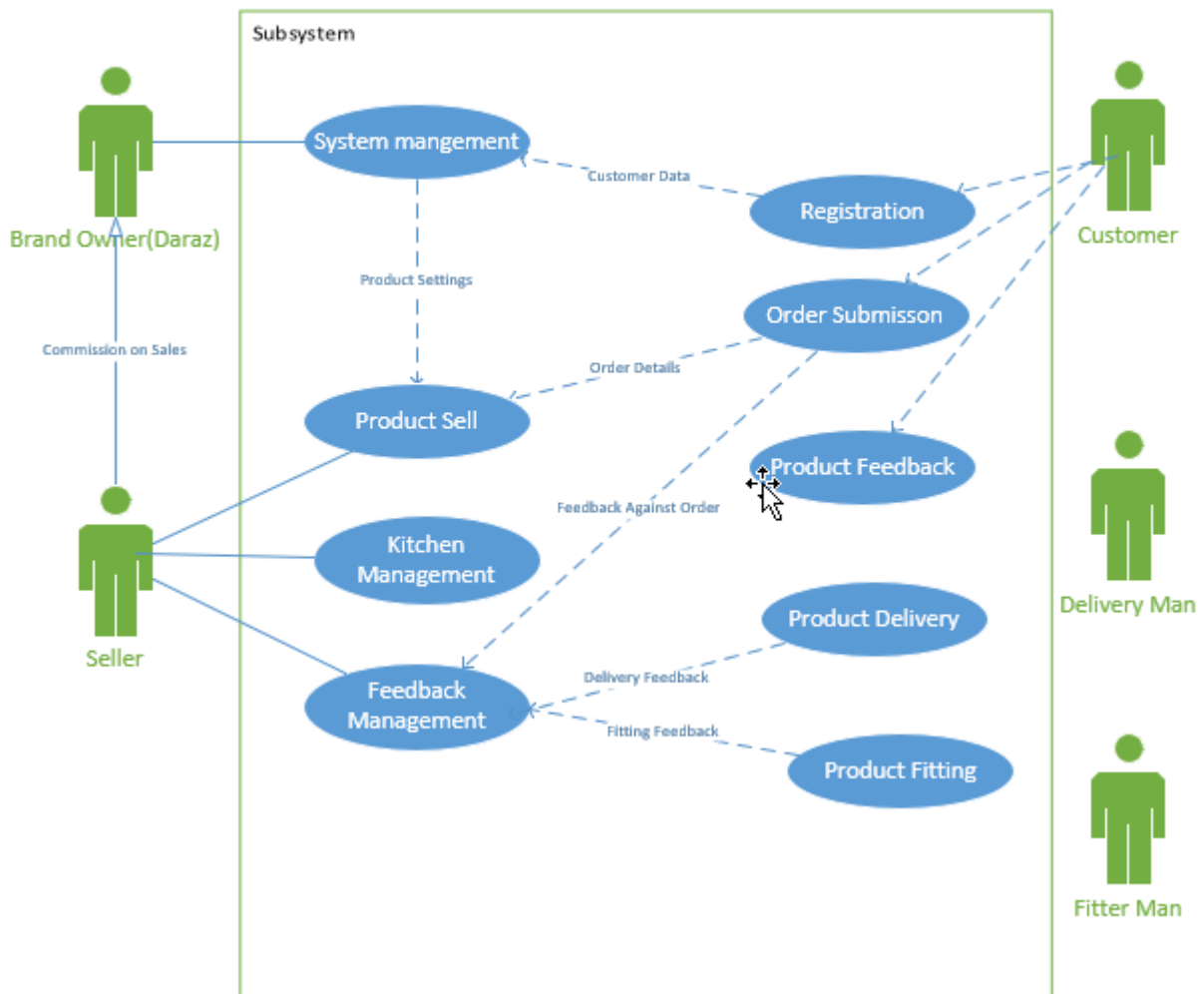### 1.3. Scope of Project

2. Overall Description

2.1. System Environment
2.2. CMS Variations (Product Concept)

Product Concept

2.3. Functional Requirements Specification

General Use Cases

Subsystem

- Brand Owner(Daraz)
- System mangement
- Customer Data
- Registration
- Customer
- Commission on Sales
- Product Settings
- Order Submisson
- Order Details
- Product Sell
- Product Feedback
- Feedback Against Order
- Kitchen Management
- Product Delivery
- Delivery Man
- Seller
- Delivery Feedback
- Feedback Management
- Fitting Feedback
- Product Fitting
- Fitter Man

2.3.1. **Brand Owners (like Daraz) use case**

2.3.2. **Vendor Use Case**

2.3.3. **Customer Use Case**

2.3.4. **Delivery-Man/Fitter-Man Use Case**

2.3.5. **Restaurant (Kitchen) Use Case**

2.3.6. **User Use Case**

#designtemplate_sp

◎E **Add New Attribute**

Default English(EN) Arabic – العربية (AR)

Name (AR)

| Ex : new attribute |

Reset   Submit

Attribute List 3

| Ex : attribute name | 🔍 | ⬇ Export ⌄ |

SI	Name	Action
1	Color	✏️ 🗑️
2	Size	✏️ 🗑️
3	Type	✏️ 🗑️

3.1 Non-Functional Requirements

3.1.1 Architectural

1. The BIVA-MART shall be designed with a Service Oriented Architecture (SOA).
2. The BIVA-MART shall be built so that it can follow Representational State Transfer (REST).
3. The BIVA-MART must be designed with a Modal-View-Controller (MVC) architecture.
4. The BIVA-MART must use open-source programming technology.
5. The BIVA-MART code shall build on Test Driven Development (TDD)

3.1.2 Documentation

6. The MIS shall have a hardcopy user guide that explains all the functions.
7. The MIS shall have an online user guide that explains the functions of the System.

4 Requirements Specification

4.1 External Systems

MIS systems need to handle data connection with legacy systems and other peripheral systems, which are:

1. **Legacy System**: The current legacy system has a database that can be any time, and all its data must be restored completely

2. **Online System**: it is used to select students based on specific criteria, our system can connect with the database and restore data

3. **Portal**: this portal will show success stories and achievements that can bind with our system through API

4. **Students Apps**: in the future, there will be apps that can help students with specific purposes, the system can be expanded to do these tasks.

4.2 Online Selection Website Functional Requirements

4.3 Add Product Attribute (Table: product_attribute, product_attribute_line)

Form Style	:	Single Form
Title	:	Add New Attribute
Description	:	All product-related attributes will be stored here.
Actor	:	
Preconditions	:	
Postconditions	:	There should be an option to add value for newly added attributes. Ex: Size (XL,XXL, Large)
Triggers	:	
Basic Flow	:	

Field Name	Title	Description
attribute_name	Name(Lang)	Name of The attribute
attribute_type	Type	1. Fabric 2. General, the Primary target is to use in the furniture sector.

Alternative Flow	:	
Exception flows	:	

Form Style	:	Single Form
Title	:	
Description	:	
Actor	:	
Preconditions	:	
Postconditions	:	
Triggers	:	
Basic Flow	:	
Alternative Flow	:	
Exception flows	:	