CONTENTS

1 CATALOG

1.1 CREATE MENU

Use Case ID	CT.0001
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows an authorized user to create a new menu configuration in the system. A menu defines a curated set of products organized by categories.
Pre- Conditions	- User is logged in and has the required role.
Post- Condition	- A new menu is saved in the system and listed in menus page. - Associated categories and their active products will be visible under the new menu.
Normal flow	1.User navigates to **Catalog > Menu**, clicks on '**Add Menu**' button where a new entry page opens. 2.User fills in the following info: .Enters a unique name for menu (*required)* .Selects one or more category *(optional)* .Sets a menu color in which color the menu shows on POS *(optional)* .Sets a menu image *(optional)* 3.User clicks on 'save'. 4.If validation passes, system stores the new menu.
Alternative flow	3.1, Instead of clicking '**save**' *(step 3)*, the user might click **'save and add another'** .System validates input and, if valid, stores the new menu. .System clears the form and resets it for a new entry
Flow diagram	

1.2 Create Category

Use Case ID	CT.0002
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows an authorized user to create categories for organizing products within the system. These categories help structure the menu and determine how products are grouped and displayed."
Pre- Conditions	- User is logged in and has the required role.
Post- Condition	- A new category is saved in the system and listed in categories page. - Created categories will be visible in 'categories' drop-down in

	'Create Product' page. -Created categories will be visible while creating promotions
Normal flow	1.User navigates to **Catalog > Categories**, clicks on '**Create Category**' button where a new entry page opens. 2. User fills in the form : .Enters a unique category name (*required).* .Selects a parent category *(optional)*. .Sets the display order in which sequence the category will be displayed.*(required)* .Sets a category color in which color that category shows on POS *(optional)* .Attach to modifier groups where each product of that category will be linked automatically with the selected modifier group .uploads a category icon & cover images (optional). 3.User clicks on 'save'. 4.System validates input and if passes, system stores the new category & redirects to the category list page.
Alternative flow	3.1.Instead of clicking '**save**' *(step 7)*, the user might click **'save & add another'** .System validates input and, if valid, stores the new category. .System clears the form and resets it for a new entry
Flow diagram	

1.3 CreaTe Modifier Group

Use Case ID	**CT.0003**
actor(s)	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	Allows authorized users to define customizable options (e.g., sizes, toppings) that can be attached to products either directly or via categories. These options (modifiers) enhance product customization during the ordering process.
Pre- Conditions	- User is logged in and has the required role. - '**View As'** options are pre-populated from the backend. - At least one **station type** exists in the system (only *required* if **'Preparation Station Routing**' is enabled and the user selects either "Modifier Group Level" or "Group Values Level" as the preparation station level.
Post- Condition	- A new modifier group is saved in the system and listed in modifiers page. - Associated modifiers become visible during product ordering.. -The group becomes available in the modifier dropdown on both the 'Create Product' and 'Create Category' pages.

	- If preparation station levels are configured, they will be reflected in kitchen routing.
Normal flow	**1**.User navigates to the "Create Modifier Group" page. **2**.User fills in the following section: **Basic Info**: User enters a **unique Name** for the modifier group (*required),* sets **display order** to define in which sequence the group will be displayed *(required)*, uploads a **Group Image** *(optional)*.and sets **Min** & **Max** values to define the maximum and minimum number of options a customer can select from that group during the order *(optional)*. **Settings**: - User selects the **View As** option to define how options are viewed on the order screen *(required)*. - Toggles **Mandatory?** switch if the customer must pick at least one option from that group *(optional)*. - Selects the modifier group **type which determines the behavior of values:** **Discount**: Used to apply percentage-based discounts. **Other**: Used for all other types of product customization (*e.g.*, Toppings, Sweetness). *Some input fields for group values will change depending on the selected group type* - If **Prep Station Routing** is **enabled,** user selects the **preparation station level:** .**On Product** (default), no station type assignment from the modifier group. .**On Modifier Group**, display 'station types' dropdown and the user select one (*required*). .**On Group Values Level**, after adding values, a "Preparation Station" tab appears for per-value assignment. **Group Values:** User can add group values immediately or later. Upon clicking **"Add Value"**, the user chooses one of two value types: **a. Add new modifier (manual entry):** - **Name** *(required)*. *E*.g., "Extra Cheese"

	- **Display Order** *(required):* determines order of appearance - **Adjustable Price** *(required):* the price added to the product price - **Active Toggle**: enables/disables the value - **Image:** optionally uploads modifier picture. - **Default Flag**: marks value as default selection - **Modifier Level** Toggle *(optional)*: Enable this to allow **size-based** customization of the value. -Useful when the product has defined sizes *(e.g., Pizza: Small, Medium, Large)*. When **ON**, user can define levels under the modifier *(e.g., Extra Cheese - Small, Medium, Large)* and set individual **prices** per level, overriding the general modifier price. ✅ This structure ensures that when a customer selects Medium Pizza, they are shown the corresponding Extra Cheese (Medium) modifier with its specific pricing and settings. **b. add product as modifier, o**pens a product selection modal - User selects one or more existing products to act as modifiers - Sets **Display Order** and **Adjustable Price** for each - Product sizes are inherited from the selected product *If the group type is set to "**Discount**", the user will enter a discount percentage instead of an adjustable price for each value.* **3**. User clicks **'save'** **4**. System validates input like: - Max selection ≥ Min selection. - Prices ≥ 0. - Display order is an integer. **5**.If validation passes, system stores the new group & redirects to the modifier groups list page
Alternative flow	**3.1**.Instead of clicking '**save**', the user might click **'save & add another'** .System validates input and, if valid, stores the new group. .System clears the form and resets it for a new entry
-KMS FLOW	**1.** When the user adds options under the **Group Values** section, an **Ingredients** section appears, allowing them to optionally define the components of each option. Additionally,

	a **Recipe** tab appears, where the user can optionally specify the quantity of each ingredient used. **2**. If **Group Values Level** is selected as the preparation station level, an **Ingredients Level** toggle appears for each value. This allows users to assign station types per individual ingredient, enabling precise routing in kitchen operations. ***NOTE: "KMS Flow (Kitchen Management System) is an optional module that enables ingredient-level preparation tracking.***

1.4 CREATE PRODUCT

Use Case ID	CT.0004
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows an authorized user to add a menu-level product that can be sold directly to a customer. It may have default ingredients and can support optional customization through attached modifier groups
Pre-Conditions	- User is logged in and has the required role. - At least one category exists in the system. - At least one station type exists in the system (required only if routing is enabled). - "Sales Channels" options are pre-populated* from the backend.
Post-Condition	- A new product is saved in the system and listed in products page. - Active added products will be visible on ordering screen (*varies depending on selected sales channel*) -The active added products will be available while creating promotions. - Preparation station levels configured will be reflected in kitchen routing.
Normal flow	1.User navigates to the **Catalog > Products**, clicks on **'Create Product'** button. A new entry page opens where there're many tabs, each tab has its own sections to fill different types of data: **A. Information tab**: This tab contains the general & base info needed to create a product with the following sections:

	.**Basic Information**: User enters the product name (required),types a short description (required) and optionally types directions of use. .**Product Images:** User optionally uploads product Images. .**Settings**: This section includes configuration details that define how the product behaves, appears, and is managed within the system where user sets a unique **SKU** number (*optional*), sets the **display order** (required) to define in which sequence the product will be displayed, selects a product **category** (*required*), selects at least one **sales channel** to choose how to promote and sell the product, via [mobile app, pos, online store] (required).**Track Inventory** switch is toggled in case the product quantity need to be trackable or adjustable in the inventory (*optional*), **Hidden** switch is toggled **ON** if the product is *not available* for selling or ordering by its own ,where such product could be used as a component of a **COMBO meal**, **active** switch controls whether the product is **visible and accessible in branches** for further configuration, if toggled **OFF** then the product will show on Products page only with 'inactive' flag, and **Could Be Purchased** switch is toggled **ON** if this product can be purchased through 'Purchase Orders'. If Prep Stations Routing is enabled, user selects a preparation station level. 'On products' then the user selects a station type from the displayed drop down (required). .**Pricing**: User sets the product's cost (*optional*).and sets the product's price (*required*). .**Service Charges:** This section is optionally used to define extra fees for different dining methods. User clicks **'Add Charges',** selects dining method [Dine-in, Pick up, Delivery], and sets the additional charges for each selected option *(required). If the option isn't wanted anymore then can be deleted by clicking 'bin' icon.* .**Modifier Groups**: This is an *optional* section where user associates the product with modifier groups to allow customers customize that product during the ordering process. These groups are added either by selecting a group from the dropdown or by enabling "**Inherit Modifier Groups from Category"** toggle or both. After added users can customize the options' status and pricing at the product level. .**Available Sizes:** This optional section allows the user to define the available sizes for the product. If 'Add Size' was clicked then the user must enter a **size name** (required), specify the **adjustable price** — the additional amount added to the base price (required), and select a **default size** (required). ⊖***Note:*** *Sizes cannot be set for* ***purchasable items****. This is recommended for maintaining accurate inventory tracking, as the system does not support tracking inventory at the size level.*

	The following tabs are optionally filled: **B. Custom Prices:** This tab is for setting the product price per branch linked to the sales channels selected in Information tab. Note that if the product price has been updated the custom prices won't be changed automatically, the user will have to change them manually (either from this tab or from inventory page) **C. Properties tab:** enter product attributes like width, height color, ...etc **D. SEO:** enter keywords for search optimization 2. Use clicks 'save' 3.System validates input
Alternative flow	2.1. Instead of clicking **'save'**, the user might click **'save & add another'** . System validates input and, if valid, stores the new group. . System clears the form and resets it for a new entry
-KMS FLOW	For KMS flow, additional tab and additional section appear on the entry page: **.Used ingredients** section where user defines what materials or prep recipes used to make the product (optional). **.Recipe tab:** after ingredients been selected this tab appears for setting amount of each ingredient (optional). .In case **'Preparation station routing'** is enabled, **'On Ingredients level'** option will be available as a routing option in addition to 'On Product' option, and as a result a new tab 'Preparation Station' tab will appear to assign a station type for each used ingredient (optional). ***SPECIAL CASE:*** If the product is attached to a **modifier group** whose **Preparation Station Routing** is set to **"On Product"**, an additional assignment option will appear. This allows the user to **map the modifier group to one of the preparation stations already assigned to the product's ingredients**, ensuring the modifier is routed to a relevant station within the context of the recipe.

1.5 Create Ingredients Category

Use Case ID	CT.0005
actor	Admin, Company Owner, Branch Owner *(for their own branches)*

Description	This use case allows an authorized user to add categories for grouping ingredients and organizing them.
Pre- Conditions	- User is logged in and has the required role.
Post- Condition	- A new ingredients category is saved in the system and listed in categories page. - Created categories will be visible in 'category' drop-down on 'Create Ingredient' page.
Normal flow	1.User navigates to the "Create Category" interface, enters category name(*required)* and *optionally* uploads a category image. 2.User Clicks 'save'
Alternative flow	2.1.Instead of clicking '**save**', the user might click **'save & add another'.** .System validates input and, if valid, stores the new category. .System clears the form and resets it for a new entry
Flow diagram	

1.6 Create Ingredients

Use Case ID	CT.0006
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows an authorized user to add ingredients needed in the system. These ingredients are needed for giving accurate specification for products and recipes building
Pre- Conditions	- User is logged in and has the required role. - At least one ingredients category exists in the system. - At least one UOM exists in the system.
Post- Condition	- A new ingredient is saved in the system and listed in Ingredients page. - Created Ingredients will be visible in 'Used Ingredients' drop-down on 'Create product' page. - Created ingredient will be visible for selection while adding a 'Prep Recipe' - Created ingredient will be visible for selection while creating inventory transaction. - Created ingredients will be visible while adding a purchase order. - Created ingredients will be added to its storage area inventory. - Created ingredients will be available while creating
Normal flow	1.User navigates to **Catalog > Ingredients,** clicks ‘add ingredient’, enters ingredient name(*required),* uploads an image (optional), selects a purchase UOM that is used while placing a

	purchase order *(required)* selects a default UOM which is the UOM used while preparing the ingredient *(required),* sets min qty alert which indicates the threshold that ingredients stock shouldn't drop below to avoid stockouts *(required),* sets **max qty** alert to help prevent overstocking *and* optionally toggles **'Returnable'** switch which means that this item could be returned and restored in the inventory in case of order cancellation. 2.User clicks save
Alternative flow	2.1.Instead of clicking '**save**', the user might click **'save & add another'** .System validates input and, if valid, stores the new category. .System clears the form and resets it for a new entry
Flow diagram	

1.7 Price Adjustment

Use Case ID	**CT.0007**
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows an authorized user to **adjust product prices by category or by individual items** for one or more specific branches. This helps businesses manage localized pricing without affecting the base product settings.
Pre- Conditions	-User is logged in and has the required role. -At least one active product exists in the system. -At least one category exists in the system. -At least one branch exists in the system.
Post- Condition	The prices of the selected products or categories are updated **only in the specified branches**, overriding default prices.
Normal flow	**1**.User navigates to **Products** page and clicks on 'Price adjustment' button. Completes the following required fields: • Select one or more **branches.** • Choose whether to adjust price **by Category** or **by Product** • Based on the above selection, **add items** (categories or specific products). • Enter the **adjusted price** or **adjustment amount** for each selected item. **2**.User Clicks save
Alternative flow	**1.** User can adjust products price from **'Inventory'** page by: .selecting a branch from the drop-down. .entering the new price directly in price field next to each

	product or combo meal. **2.1.** Instead of clicking '**save**', the user might click **'save & add another'** . If valid, it **saves the changes** and **resets the form** for a new entry. ***NOTE:*** *This flow is limited to **products and combo meals only**, and does not support price adjustment by category.*
Flow diagram	

1.8 CREATE COMBO MEAL

Use Case ID	**CT.0008**
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows authorized users to create a **Combo Meal**, which is a bundled product consisting of multiple existing items (e.g., Burger + Fries + Drink) that can be sold as a single offering. Combo meals can be promoted independently of the standard menu.
Pre- Conditions	. The user is logged in and has the required role. . At least one sales channel is already configured in the system. . At least **two active products** are available in the product catalog.
Post- Condition	. A new combo meal is successfully added to the **Combo Meals List**. . The combo is now available to order through the selected sales channels, even if not listed on the main product menu.
Normal flow	1.User navigates to **Catalog > Combo meals**, and clicks on **'Add combo meal'** button. 2.User fills in the following sections: **Information tab**: This tab contains the general & base info needed to create a product with the following sections: .**Basic Information**: User enters the combo **name** (*required*),types a **short description** (*required*) and *optionally* types **directions of use**. **Product Images:** User optionally uploads combo meal **Images**. .**Settings**: This section includes configuration details that define how the product behaves, appears, and is managed within the system where user sets a **unique SKU** number (*optional*), sets the **display order** (required) to define in which sequence the product will be displayed, selects at least one **sales channel** to choose

	how to promote and sell the combo, via **[mobile app, pos, online store]** (*required*). **Active** toggle: This switch controls whether the combo meal is **visible and accessible in branches** for further configuration, **ON then t**he combo meal will appear in branch-level settings, allowing it to be made available, priced, or customized per branch. **OFF** then the combo meal will remain hidden from all branches — it won't be visible in the Inventory or available for ordering anywhere. **If Prep Stations Routing is enabled**, then it *automatically* follows the routing rules of its components items. **.Combo Items:** This section is used to build the combo meal by selecting which existing products *(e.g., burgers, sides, drinks)* will be bundled together. User clicks **'Add items'**, selects the products to o include as part of the combo, for each selected item: Choose the **size**, if applicable (required), specify the **quantity** of that item to be included in the combo meal (required). .**Pricing**: User sets the meal's cost (*optional*).and sets the meal's price (*required*). .**Service Charges:** This section is optionally used for setting **extra charges** on the product for specific dining options [Dine-in, Pick up, Delivery], user clicks **'Add charge'**, selects a dining option from the dropdown and sets the charges for each selected option *(required). If the option isn't wanted anymore then can be deleted by clicking 'bin' icon.* . **Modifier Groups**: This is an *optional* section where user associates the combo meal with modifier groups to allow customers customize that meal during the ordering process. This groups are added by selecting a group from the dropdown. After added users can customize the options' status and pricing at the combo meal level. 3.User Clicks save
Alternative flow	**3.1.** Instead of clicking '**save**', the user might click **'save & add another'** . System validates input and, if valid, stores the new category. . System clears the form and resets it for a new entry ***NOTE: This alternative flow can be applied only on products while Normal Flow includes products & categories.***
Flow diagram	

1.9 CREATE OPEN COMBO

Use Case ID	CT.0008
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows an authorized user to create and configure open meals—special menu items offered **free of charge** to customers during a defined period. The user can specify the active days, open hours, and applicable sales channels to control availability. Open meals are typically used for promotional purposes.
Pre- Conditions	. The user is logged in and has the required role. . At least one sales channel is already configured in the system. . At least **one active product** is available in the product catalog.
Post- Condition	. A new open meal is successfully added to the **Open Combo Meals List**. . The combo is now available to order through the selected sales channels, even if not listed on the main product menu.
Normal flow	1.User navigates to **Catalog > open meals**, and clicks on **'Add open combo** button. 2.User fills in the following sections: **Information tab**: This tab contains the general & base info needed to create a product with the following sections: .**Basic Information**: User enters the **item name** (*required*), types a **short description** (*required*) and enters the item **price** *(required)*. **Product Images:** User optionally uploads combo meal **Images**. .**Settings**: This section to configure the meal availability, by specifying active days, open hours and sales channels .**Combo Items:** This section is used to build the combo meal by selecting which existing products *(e.g., burgers, sides, drinks)* will be bundled together. User clicks **'Add items'**, selects the products to include as part of the combo, for each selected item: Choose the **size**, if applicable (required), specify the **quantity** of that item to be included in the combo meal (required). .**Pricing**: User sets the meal's cost (*optional*).and sets the meal's price (*required*). .**Service Charges:** This section is optionally used for setting **extra charges** on the product for specific dining options [Dine-in, Pick up, Delivery], user clicks '**Add charge**', selects a dining option from the dropdown and sets the charges for each selected option *(required). If the option isn't wanted anymore then can be deleted by clicking 'bin' icon.* . **Modifier Groups**: This is an *optional* section where user associates the combo meal with modifier groups to allow customers customize that meal during the ordering process. This

	groups are added by selecting a group from the dropdown. After added users can customize the options' status and pricing at the combo meal level. 3.User Clicks save
Alternative flow	**3.1.** Instead of clicking '**save**', the user might click **'save & add another'** . System validates input and, if valid, stores the new category. . System clears the form and resets it for a new entry ***NOTE: This alternative flow can be applied only on products while Normal Flow includes products & categories.***
Flow diagram	

2 PREP STATIONS

NOTE: *This is a part of the flow only when 'Prep Stations Routing' is enabled from company settings.*

2.1 CREATE STATION TYPE

Use Case ID	**ST.0001**
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows an authorized user to add station types for grouping prep stations that can belong to the same type and can be used for same purpose while preparing and fulfilling an item.
Pre- Conditions	- User is logged in and has the required role. - 'Preparation Station Routing' is enabled in the system.
Post- Condition	- A new station type is saved in the system and listed in station types page. - Created stations will be visible in Stations drop-down on 'Create kitchen area' page. - Created stations will be visible for selection while creating products, Modifiers Groups and every place that applies routing assignment.
Normal flow	1.User navigates to **prep stations -> station types ,** clicks on 'add station type' button and enters station name (*required)* 2.User Clicks save.

Alternative flow	2.1.Instead of clicking '**save**', the user might click **'save & add another'** .System validates input and, if valid, stores the new station. .System clears the form and resets it for a new entry
Flow diagram	

2.2 CREATE STORAGE AREA

Use Case ID	**ST.0002**
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows an authorized user to add storage areas for storing & tracking materials (row materials and recipes).
Pre- Conditions	- User is logged in and has the required role. - **KMS** flow is enabled. - At least one branch exists in the system.
Post- Condition	- A new storage area is saved in the system and listed in **Storage Areas** page. - Created areas will be visible in 'Areas' drop-down on **'Create kitchen area'** page. - Created areas will be visible for selection while creating **inventory transaction**. - Created areas will be visible for selection while creating **preparation orders** in **KMS** flow. - Created areas will be available while adding a **purchase order**. - Created areas will be available while adding a **wastage order**.
Normal flow	1.User navigates to **prep stations > storage area** and clicks on 'add storage area' button, enters area name (*required),* selects a branch *(required)* and *optionally* fills description field. 2.User Clicks save.
Alternative flow	2.1.Instead of clicking '**save**', the user might click **'save & add another'** .System validates input and, if valid, stores the new area. .System clears the form and resets it for a new entry
Flow diagram	

2.3 Create Kitchen Area

Use Case ID	ST.0003
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows an authorized user to add kitchen areas represents a location in the kitchen where part of an order is prepared (e.g., Grill, Salad Station), order items are routed based on preassigned areas for each item (at product/ modifier/ ingredients level)
Pre- Conditions	- User is logged in and has the required role. - Preparation Station Level Routing is enabled. - At least one station type exists in the system. - At least one branch exists in the system.
Post- Condition	- A new kitchen area is saved in the system and listed in Kitchen areas page. - Created areas will be available in 'Areas' drop-down on 'Create Product' & 'Create Modifier Group' pages. - Items of each order are routed to their assigned kitchen area in order to be prepared and fulfilled.
Normal flow	1.User navigates to **prep stations -> kitchen Area** and clicks on 'add kitchen area' button and fills in the following sections fields: **.Kitchen Area Details:** enters kitchen name (*required*). selects a station type *(required)* and selects a storage area *(optional)*. **.Kitchen Area Settings:** this is used to configure how the kitchen area operates, in other words how tickets or orders are **delivered** to this station (via Printer or KDS), and whether it's the **default** kitchen area. **.Is Default (*optional*):** The user enables this toggle if this area should act as the default routing destination. Products with no assigned kitchen area will be automatically routed to this kitchen area. *This typically occurs when products were created **before routing was enabled and** thus have no kitchen area configured so once routing is turned on, the system needs a fallback, and the default kitchen area is used for those products.* **.Output Device Configuration (*required*):** .User must select at least one method to define how this kitchen area receives orders - Printer enabled: sends the order to an integrated kitchen printer. - KDS enabled: the order tickets will be displayed on the connected KDS screen. - both are enabled: the order tickets will be diaplayed on the

	KDS screens and printed simultaneously. 2.User Clicks save
Alternative flow	2.1. Instead of clicking '**save**', the user might click **'save & add another'** . System validates input and, if valid, stores the new area. . System clears the form and resets it for a new entry
Flow diagram	

3 INVENTORY

3.1 MANAGE INVENTORY

Use Case ID	**INV.0001**
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows authorized users to manually manage inventory per branch. This includes managing quantities, pricing, stock status and availability at branch. Additionally, the system automatically updates inventory based on actions such as purchase orders and fulfilling customer orders.
Pre- Conditions	- User is logged in and has the required role. - At least one product exists in the system, - At least one branch exists in the system. -'**Track Inventory**' toggle is **enabled** for the product (required for quantity adjustments)
Post- Condition	- Product pricing will be updated at the selected branch. - Product quanitity will be updated at the selected branch. - Product availability will be updated at the selected branch. - Product stock status will be updated at the selected branch. - All changes are saved and reflected in reports, listings, and checkout logic.
Normal flow	1.User navigates to **Inventory** page, selects a branch to view its products, updates price of the products, updates quantities, toggles the "**Available**" switch to indicate whether the product

	is offered for sale or fulfillment at the selected branch and toggles the "**In Stock**" switch to reflect whether the product **physically** exists at the branch. ***NOTE: these changes are auto saved.***
Alternative flow	. Pricing can be adjusted through 'products' page by clicking 'adjust price' button. ***Review Use Case* CAT.0007**
Flow diagram	

3.2
inventory transfer

Use Case ID	**INV.0002**
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows authorized users to transfer ingredients from storage area to another.
Pre- Conditions	- User is logged in and has the required role. - At least one ingredient exists in the system. - At least one storage area exists in the system.
Post- Condition	- storage area inventory update.
Normal flow	1.User navigates to **Inventory > Inventory Transfer**, selects both source area and destination area, sets the transfer date then clicks **'Add Ingredients'** to selects which items to be transferred with their quantities. 2. User clicks 'save'.
Alternative flow	
Flow diagram	

4 BRANCHES

3.2 CREATE BRANCH (THIS NEEDS REVIEW WHEN I CAN TEST IT)

4 BRANCHES

Use Case ID	BR.0001
actor	Admin, Company Owner
Description	This use case allows authorized users to create a new branch under the company. Branches serve as operational units where products are stored, managed, and offered to customers. Each branch can process orders independently and may have its own settings for staff and inventory
Pre- Conditions	- User is logged in and has the required role. - At least one sales channel exists in the system. - At least one item of each of the following exists in the system [payment provider, tax provider, and shipping provider] - Countries & States are available for filling address.
Post- Condition	-A new branch will be stored & listed in 'Branches' page. - Created branches will be available for storing and offering products according to the sales channel the product was linked to - Customizing products pricing per branch will be available while creating products. - Added branch will be added to the inventory for tracking its products.
Normal flow	1.User navigates to **Branches** page, clicks 'Add branch' button and fill in the following sections: **.Basic Information:** user enters branch name (required), enters branch phone number (required), selects a default tax provider [fixed, stripe] (required), sets working hours (optional), enable/ disable **Is Active** toggle for branch status and *optionally* uploads a branch **logo**. . **Branch Address:** **Address** line (required), **ZIP code** (required), select **country** (required), select **state** (required) and selects city (optional). **.Branch Settings:** This section is visible **only if company has Routing feature enabled.** User can enable **Routing** toggle if Preparartion Station Routing will be applied at the created branch (*optional).* Optionally enter the branch location on map **Work Hours:** Optionally set the work hours per day by setting the opening time and closing time for each day **.Sales Channels:** in this section user enables one or more sales channel that the branch will offer its products through [POS, Online store, Mobile App], each has its own configuration tab appears after switching its toggle **ON.** **For each sales channel, the user configures one or more Dining Options:** • **Dine-In** • **Take-out**

4 BRANCHES

	• **Delivery**
	
	*When a dining option is enabled, a new section opens with specific settings that include __ **for Dine in & Take-out** options:*
	
	• **Enable Table Service** – ON if waiters serve tables *(only relevant for Dine-in)*.
	• **Override Tax Provider** – ON if this dining method should use a different tax provider than the branch default one. .Requires selecting a **Tax Provider** if enabled.
	• **Payment Provider** – select a payment method for this dining flow.
	• **Apply Extra Charge** *(optional)*:
	  • Toggle ON to enable a service fee.
	  • Choose **Charge Type** (Fixed, Percentage, Open).
	  • Enter **Amount** (required if enabled).
	  • **Taxed?** – ON if the extra charge should include tax.
	• **Auto Gratuity** - *only relevant for Dine-in (optional)*:
	  • Toggle ON if a gratuity will be automatically added.
	  • If ON, set the amount.
	
	*For **Delivery** Dining option**:***
	
	• **Shipping Provider** *(required)*:
	Select the delivery service responsible for fulfilling orders. This can be a third-party provider or a custom in-house delivery setup.
	
	• **Fixed Fees?** *(toggle)*:
	ON: A fixed delivery fee will be charged to the customer. Requires setting a **Charge Amount**, which is the exact fee applied to the order (e.g.$5.00)
	
	OFF: May support flexible or dynamic delivery pricing *(e.g. distance-based fees)*.
	
	• **Waive Free Threshold?** (toggle):
	ON: Enables the option to automatically **waive (remove)** the delivery fee when the order total meets or exceeds a certain amount. Requires entering a **Waive Threshold Amount**, which defines the minimum **pre-discount order value** needed for free delivery *(e.g., $30.00)*.
	OFF: Delivery fee applies regardless of order size.

	2. User clicks 'save' 3. System validates input and, if valid, stores the new branch.
Alternative flow	2.1. Instead of clicking '**save**', the user might click '**save & add another'.** .System validates input and, if valid, stores the new area. .System clears the form and resets it for a new entry.
Flow diagram	

4 ORDERS

4.1 MANAGE ORDER STATUS

Use Case ID	ORD.0001
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case allows an authorized user to view & update orders status, orders that were placed via mobile app, website or pos.
Pre- Conditions	-User is logged in and has the required role. - At least one order exists in the system. - Order status is neither 'canceled' nor 'completed'
Post- Condition	Order status is updated, and notifications are sent to the customer's app
Normal flow	1.User navigates to the "**orders**" page, clicks on the order name or on the 'eye' icon to view details and change status as required [mark as in preparation, mark as failed or cancel order].
Alternative flow	
Flow diagram	

4.2 CREATE PURCHASE ORDERS

Use Case ID	ORD.0002
actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case is used to log orders purchased via suppliers connected to the company

Pre- Conditions	- User is logged in and has the required role. - At least one active supplier in the system. - At least one **active purchasable** product exists in the system - At least one ingredient exists in the system and one storage area.
Post- Condition	-A new purchase order is logged in the system. -Purchased quantity will be added to the product's quantity in the inventory. -For **KMS** flow, the purchased quantity of ingredients will be added to the **materials inventory** in 'Storage Area' page.
Normal flow	1. User navigates to **Orders > purchase order** page, clicks 'Add purchase order' button then fills in the following sections: **.Order Details:** user enters the order number (required), selects the supplier (required), and sets purchase date (required) **.Order Items:** in this section user clicks **'add items'** to open a modal for selecting the purchased items [products or ingredients], marks the desired items then clicks **'add selected'.** For each item, the user enters the Purchase Price and Quantity (*required*). If the item is an ingredient, the user must also select a Storage Area for where it will be stored (*required*) **.Summary:** In this section system displays the subtotal of the order automatically, if a discount was applied user enters it and if there are extra charges (*e.g. shipping*) then user clicks '+add charges' enters the reason behind these charges (*required*), the amount of the charges (*required*) and may add a note for internal reference (*optional*). 2.User clicks '**Submit Order'**.
Flow diagram	

5.3

CREATE WASTAGE ORDERS

Use Case ID	**ORD.0003**

actor	Admin, Company Owner, Branch Owner *(for their own branches)*
Description	This use case is used to **record and track ingredient loss** in the branch.it helps ensure accurate inventory levels and reduce shrinkage by accounting for items that can no longer be sold or used due to spoilage, damage, expiration, over-prep, or operational errors.
Pre- Conditions	- **KMS** flow is **enabled** at the branch. - User is logged in and has the required role. - At least one storage area exists in the system. - At least one ingredient exists in the system - At least one active employee exists.
Post- Condition	-A new wastage order is logged in the system. -Wasted quantity will be discarded from the item's quantity in the inventory.
Normal flow	1.User navigates to **Orders -> Wastage Orders** clicks 'Add Wastage Order' and fills in the following details: selects the storage area from which the items will be wasted (*required*), adds the items to be wasted, specifying the quantity for each (*required*), selects the employee responsible for the wastage (*required*) and adds details (*optional*). 2. User clicks **'save'**.
Alternative flow	
Flow diagram	

5 USERS

5.1 CREATE EMPLOYEES

Use Case ID	**USR.0001**
actor	Admin, Company Owner, Branch Owner
Description	This use case is used to add employees to the system.
Pre- Conditions	- User is logged in and has the required role. - At least one company exists in the system. - At least one branch exists in the system (needed if role isn't Company Owner) - At least one role exists in the system.
Post- Condition	-A new employee is added to 'Employees' list. - Active added employee can access the system. -Active added employee will show in 'wasted by' drop down on '**Create Wastage Order**' page.
Normal flow	1.User navigates to **Users > Employees** page, clicks 'Add Employee' button, enters contact info [name, phone, email, password] all are *required* fields. Selects the employee role [company owner, branch owner, employee, waiter, cashier], based on the selected role and the permission level of the person creating the employee, the

	system will dynamically display the appropriate company and/or branch assignment fields: *.If an Admin is creating the employee and the selected role is **Company Owner**, a Company dropdown will appear to assign the employee to a company. If the selected role is **Branch Owner** or lower, both Company and Branch dropdowns will be shown, allowing assignment to a specific branch within the selected company.* *.If a Company Owner is creating the employee they can create Branch Owners or lower roles then only the Branch will be assigned, since the company is already implied.* Finally, user enables **'active'** so the employee can access the system (optional) 2.User clicks save.
Alternative flow	
Flow diagram	

6.2 CREATE CUSTOMERS

Use Case ID	**USR.0002**
actor	Admin, Company Owner
Description	This use case allows an authorized user to manually create a customer account from the dashboard
Pre- Conditions	- User is logged in and has the required role. - Customer's email doesn't exist in the system meaning the customer haven't signed up with the same email before.
Post- Condition	-A new customer is added. -Customer can log in to the app with the created account credentials. -
Normal flow	1.User navigates to **Users > Customers**, clicks on 'Add Customer' button and fills in the following customer's information: name, phone number, email and password (all are *required fields*). 2.User Clicks 'save'
Alternative flow	
Flow diagram	

7 PROMOTIONS

7.1 ADD PROMOTION

7.1.A ADD DISCOUNT PROMOTION

Use Case ID	PROM 0001 (should be changed)
actor	Admin, Company Owner, Branch Owner
Description	This use case allows an authorized user to add sale on products.
Pre- Conditions	-User is logged in and has the required role. - At least one product exists in the product catalog. - At least one branch exists in the system.
Post- Condition	- A new promotion is saved and listed in 'Promotions' page. - If marked as active, the promotion will be applied to the selected products or category based on the defined conditions, and pricing will be updated accordingly across the selected sales channels and branches
Normal flow	1.User navigates to **Promotions page**, clicks on **Add Promotion > Discount promotion'** and fills the following sections on the entry page**:** **.Basic Info:** user enter promotion title (*required*), discount amount (*required*) and toggles the **Is Active** status to enable/disable the promotion(*optional*). **.Promotion Items**: Chooses how the discount will be applied: On all products: discount applies on all products in the branch selected. On Product: user selects one or more specific products (*required*) On Category: user selects one product category where the discount will apply to all products under it (*required*) **.Application Date/time**: set the time frame for the promotion: Selects whether the promotion should apply **all week** or on specific **weekdays, d**efines the **start and end dates** of the promotion and sets whether the promotion is valid only during **store open hours** or for a custom time window. **.Settings:** user selects on which branches the promotion will be applied, then selects applicable dining options and sales channels
Alternative flow	
Flow diagram	

7.1.B ADD FREE ITEM PROMOTION

Use Case ID	PROM 0001
actor	Admin, Company Owner, Branch Owner
Description	This use case allows an authorized user to create a promotion where **customers receive free item(s) after spending a defined amount** on eligible products. Once the specified spending threshold is met during an order, the system automatically applies the free item(s) as part of the promotion.
Pre- Conditions	-User is logged in and has the required role. - At least one product exists in the product catalog. - At least one branch exists in the system. - System has valid sales channels and dining options defined.
Post- Condition	- A new promotion is saved and listed in ‘Promotions’ page. - If marked as active, the promotion will be applied on the selected products.
Normal flow	1.User navigates to **Promotions page**, clicks on **Add Promotion > Fee Item promotion’** and fills the following sections on the entry page**:** **.Basic Info:** user enter promotion title (*required*), spend amount which refers to amount customers should pay to get the free item (*required*) and toggles the **Is Active** switch to enable/disable the promotion(*optional*). **.Promotion Items**: Chooses which items that will be offered as free items On Product: user selects at least one product (*required*) **.Application Date/time**: set the time frame for the promotion: Selects whether the promotion should apply **all week** or on specific **weekdays, d**efines the **start and end dates** of the promotion and sets whether the promotion is valid only during **store open hours** or for a custom time window. **.Settings:** user selects on which branches the promotion will be applied, then selects applicable dining options and sales channels 2. User clicks **‘save’**.
Alternative flow	
Flow diagram	

7.1.C ADD BOGO PROMOTION

Use Case ID	PROM 0003 (should be changed)
actor	Admin, Company Owner, Branch Owner
Description	This use case allows an authorized user to create a **BOGO (Buy One, Get One)** promotion, where purchasing specified product(s) automatically grants the customer one or more free item(s). This can be used to incentivize sales by offering additional items at no cost when certain conditions are met.
Pre- Conditions	-User is logged in and has the required role and permissions to manage promotions. - At least one active product exists in the product catalog. - At least one branch is defined and active in the system. - System has valid sales channels and dining options defined.
Post- Condition	- A new promotion is saved and listed on the 'Promotions' page. - If marked as **Active**, the system evaluates this promotion during eligible order transactions and automatically applies it when the qualifying conditions are met.
Normal flow	1.User navigates to **Promotions page**, clicks on **Add Promotion > BOGO promotion'** and fills the following sections on the entry page**:** **.Basic Info:** user enter promotion title (*required*), and toggles the **Is Active** status to enable/disable the promotion(*optional*). **.Must Buy Items**: This section is to select item(s) which purchasing will trigger the promotion. User clicks 'add item' which opens a modal to select the products with their sizes. *Note: The system applies **AND logic** — all selected items must be present in the order to trigger the promotion.* **.Free Items To Get:** Choose the free item(s) customers receive after spending a defined amount. User clicks 'add item' which opens a modal to select the products with their sizes. Once the specified spending threshold is met during an order, the system automatically applies the free item(s) as part of the promotion. **.Application Date/time**: set the time frame for the promotion: Selects whether the promotion should apply **all week** or on specific **weekdays, d**efines the **start and end dates** of the promotion and sets whether the promotion is valid

	only during **store open hours** or for a custom time window. **.Settings:** User selects the **Branches** where the promotion applies, selects applicable dining options and sales channels.
Alternative flow	
Flow diagram	

7.1.C ADD LARGER ORDER PROMOTION

Use Case ID	**PROM 0003 (should be changed)**
actor	Admin, Company Owner, Branch Owner
Description	This use case allows authorized users to create and configure a promotion where customers receive a fixed discount when their order subtotal reaches a specified minimum spend threshold. Once saved and activated, the promotion is automatically applied to qualifying orders during the defined period and under the configured conditions.
Pre- Conditions	-User is logged in and has the required role. - At least one product exists in the product catalog. - At least one branch exists in the system. - System has valid sales channels and dining options defined.
Post- Condition	- A new promotion is saved and listed in 'Promotions' page. - If marked as active, the promotion will be applied to the selected products across the selected sales channels and branches
Normal flow	1.User navigates to **Promotions page**, clicks on **Add Promotion > Larger Order Promotion'** and fills the following sections on the entry page**:** **.Basic Info:** user enter promotion name (*required*), sets the **spend amount** which is the minimum order subtotal required to qualify for the promotion, sets the **save amount** to be discounted when the condition is met,and toggles **"Is Active"** to enable or disable the promotion. .**Application Date/time**: set the time frame for the promotion: Selects whether the promotion should apply **all week** or on specific **weekdays, d**efines the **start and end dates** of the promotion and sets whether the promotion is valid

	only during **store open hours** or for a custom time window. **.Settings:** user selects on which branches the promotion will be applied, then selects applicable dining options and sales channels
Alternative flow	
Flow diagram	

CONFIGURATION

8.1 SETTINGS

Use Case ID	**CONFG.0001 (will be updated)**
actor	Admin, Company Owner
Description	This use case allows administrators to manage essential operational settings and loyalty configurations that apply across all branches of the company. It supports customizing reward program rules, app version requirements, delivery policies, and system behavior to ensure smooth operation across platforms and customer experiences.
Pre- Conditions	- User is logged in and has the required role.
Post- Condition	- System behavior updated, global behavior (e.g., delivery fee logic, loyalty point calculations) reflects the new settings.
Normal flow	1.User navigates to **Configuration > Branch configuration** and applies the following settings: • **Delivery Fee Waiver Threshold** Set the minimum order amount that qualifies for free delivery. If the order total exceeds this value, the delivery fee is waived. • **Registration Points Gift** Define the number of loyalty points gifted to new users upon registration, encouraging user sign-up and engagement. • **Exchange Rate** Set the monetary value equivalent for 1 loyalty point (e.g., $1 = 2 points). Used when redeeming points at checkout.

	• **Points-to-Cash Redemption Ratio** Specify how many points are required to deduct $1 from the customer's bill (e.g., 25 points = $1 discount). • **Server Time Adjustment** Modify the local server time by increasing or decreasing it (in hours). Useful for syncing across time zones or correcting drift. • **Android & iOS App Version** Set the minimum required version of the mobile app that can access the platform, helping enforce updates and compatibility. • **Routing Configuration:** Toggle 'Preparation Station Routing' switch if order items routing is used by company (optional) • **Kitchen Management System Configuration:** toggle KMS switch if enabled at the company (optional). 2.User Clicks **'save'**
Alternative flow	
Flow diagram	

8.2 ADD SHIPPING PROVIDER

Use Case ID	**CONFG.0002**
actor	Admin, Company Owner
Description	This use case allows authorized users to register a shipping provider, which represents a company, service, or method responsible for delivering orders (*e.g.*, FedEx, Custom Delivery, Shippo). Providers serve as a parent entity for specific shipping methods and help categorize delivery types by functionality or integration.
Pre- Conditions	- User is logged in and has the required role. - At least one shipping way (type) is configured in the system.
Post- Condition	- A new shipping provider is created and available to be linked to shipping methods.
Normal flow	1.User navigates to **Configuration > Shipping Providers**, clicks on **'Add Shipping Provider'** button 2. User fills in the following information: Enters the name of the service in **provider title** field *(required)*, selects a **shipping way** [FedEx, FedEx tracking, shippo]. This is the type of delivery service

	supported by the provider *(required).* *Based on the selected type, the form will display dynamic fields as follows:* - If **FedEx** or **FedEx Tracking** is selected: **a. Account Number** assigned by the shipping company. Used for billing and identification. *For example, FedEx assigns this when you register a business account. (required)* **b. Client ID**: This is a **unique code** issued by the shipping provider when you register for API access. The system uses this value when calling the shipping provider's server to authenticate your business. *(required)* **c. Client secret:** A **confidential** password/key that is used with the Client ID to securely connect your system with the provider's system. This should be **kept private and never shared**. *(required)* - If **Shippo** is selected then **Client Secret** iis the only field *required* for connecting to shippo's system. Finally, user *optionally* switch **'status'** toggle to enable/disable the provider. 3.User Clicks **'save'.**
Alternative flow	
Flow diagram	

8.3 ADD SHIPPING METHOD

Use Case ID	**CONFG.0003**
actor	Admin, Company Owner
Description	This use case allows users to define specific shipping methods that are offered through a selected shipping provider. This is the **specific delivery option** that the customer or staff will select at checkout.
Pre- Conditions	- User is logged in and has the required role. - At least one **active** shipping provider.
Post- Condition	- A new shipping method is saved and linked to its provider. - It becomes available for use in order workflows *(e.g., POS, online order, delivery tracking).*
Normal flow	1.User navigates to **Configuration > Shipping Methods**, clicks on **'Add Shipping Method'** and fills in the following info: enters the method **title** *(required)*, selects a **shipping provider** from list of saved providers (required), sets the delivery **cost** (charge) for this method *(required),* enters the estimated number of days it takes to complete the

	delivery in **Expected Days To Deliver** field, and optionally enters additional notes in **Description.** 2.User clicks **'save'**.
Alternative flow	
Flow diagram	

8.4 ADD PAYMENT PROVIDER

Use Case ID	**CONFG.0004**
actor	Admin, Company Owner
Description	This use case allows authorized users to set and configure **payment providers** such as Cash, Card, or integrated services like Stripe. These providers are used in the system to define how payments are processed at the time of order checkout, refund, or settlement.
Pre- Conditions	- User is logged in and has the required role. - Payment types are already defined in the system *(e.g., Card, Cash, Points).* - If configuring an **external** provider *(e.g., Stripe or PayPal)*, the user has obtained the necessary credentials from the provider's developer or business dashboard.
Post- Condition	-A new payment provider is created and becomes available for use at checkout. -Added providers will be available for selection while configuring branches' sales channels.
Normal flow	1. User navigates to **Configuration > Payment Providers**, clicks on **'Add Payment Provider'** button. 2. User fills in the following information: Enters the name of the payment option in **provider title** field *(required)*, selects a **payment way** [cash, stripe, PayPal, wallet, points]. This is the type of payment service supported by the provider *(required).* *Based on the selected type, the form will display dynamic fields as follows:* • If **Stripe** is selected: **a. Publishable Key** *(required):* A public key provided by Stripe used to securely send payment info. **b. Secret Key** *(required)*: A private code from Stripe that lets your system accept payments. **c. Merchant ID** *(required)*: This ID is used to identify your Stripe business account. **d. Webhook Secret** *(optional)*: Used to securely receive automatic updates from Stripe, such as when a payment is completed.

	e. Apple Merchant ID & Name *(optional)*: Used if you want to accept Apple Pay. **f. Google Merchant ID & Name** *(optional)*: Used if you want to accept Google Pay. • If **PayPal** is selected: User enters **Publishable Key** and **Secret Key** *(required fields)*. • If **Wallet, cash or points** is selected then no additional fields show up. Finally, user *optionally* switch **'status'** toggle to enable/disable the provider. **3.** User clicks **'save'**.
Alternative flow	
Flow diagram	

8.5 ADD TAX PROVIDER

Use Case ID	**CONFG.0003**
actor	Admin, Company Owner
Description	This use case allows an authorized user to add a **Tax Provider** to the system. A Tax Provider defines how taxes are calculated and applied to orders. Providers can be **manual types** (*e.g., fixed or estimated tax*) or **automated integrations** with external tax services (*e.g., Stripe tax service*).
Pre- Conditions	- User is logged in and has the required role. - At least one **Tax Way** (tax method type) is available in the system.
Post- Condition	- A new tax provider is saved and available for use in order calculation workflows. - Added providers will be available for selection while creating branches.
Normal flow	1.User navigates to **Configuration > Tax Providers**, clicks on **'Add Tax Provider'.** 2. User fills in the following information: Enters the name of the payment option in **Tax Provider title** field *(required)*, selects a **Tax Way** (required) could be one of the following [Fixed, Stripe, PayPal, Estimated]. *(Based on the selected way, the form will display dynamic fields as follows:* • If **Stripe** is selected: **a. Publishable Key** *(required):* A public key provided by Stripe used to securely send payment info. **b. Secret Key** *(required)*: A private code from Stripe that lets your system accept payments.

	c. Merchant ID *(required)*: This ID is used to identify your Stripe business account. **d. Webhook Secret** *(optional)*: Used to securely receive automatic updates from Stripe, such as when a payment is completed. • If **PayPal** is selected: User enters **Publishable Key** and **Secret Key** *(required fields)*. • If **Wallet, cash or points** is selected then no additional fields show up. Finally, user *optionally* switch **'status'** toggle to enable/disable the provider. **3.**
Alternative flow	
Flow diagram	

8.6 ADD SUPPLIER

Use Case ID	CONFG.0006
actor	Admin, Company Owner
Description	This use case allows authorized users to create a new supplier record by entering supplier information, contact details, and location data.
Pre- Conditions	- User is logged in and has the required role. - The system has a preloaded **Countries** list.
Post- Condition	- New supplier record is stored and accessible in the supplier list. - Added supplier can be linked to purchase orders.
Normal flow	1.User navigates to **Configuration > Suppliers**, clicks on **'Add Supplier'.** 2. User fills in the following information: .**Supplier Name** which is the business or company name (*required*) **.Contact Name** *(required)* .**Phone Number** (required) **.Contact Email** (required) **.Active (toggle)**: activate/ deactivate the supplier. **.Address Info** (address, country, state, ZIP) all *required* fields and selects the city is *optional.* *3.* User clicks **'save'**.
Alternative flow	
Flow diagram	

8.7 CREATE UNIT OF MEASUREMENT CATEGORY

Use Case ID	CONFG.0007
actor	Admin, Company Owner
Description	This use case allows authorized users to create UOM categories for grouping used for products, used for managing preparing recipes, fulfilling orders, and wastage orders
Pre- Conditions	- User is logged in and has the required role. - The system has a preloaded **Countries** list.
Post- Condition	- New UOM category is added to the system. - Added UOMs are visible for selection to set ingredients amounts while creating prep recipes and while recording wastage orders
Normal flow	1.User navigates to **Configuration > Unit Of Measurements**, clicks on **'Add UOM category'.** 2. User fills in the following information: .**UOM category Name** *(required).* Clicks **'Add Unit':** • enters the unit's name (required). • **Reference Unit:** this flag refers to the **base unit** of the category. All other units are defined relative to this one. *(e.g. Gram is the reference unit in Weight category)* • **Ratio** (required): defines how much of this unit equals **1 reference unit**. (*e.g., 1 kilogram is 1000 grams) so you set the ratio for KG to 1000* ***(it has 1000 units of the reference unit)*** **3.** User clicks **'save'*.***
Alternative flow	
Flow diagram	

9 KMS (KITCHEN MANAGEMENT SYSTEM)

This Shows in the sidebar only if KMS is enabled from Configuration

9.1 CREATE PREP RECIPE

Use Case ID	KMS.0001
actor	Admin, Company Owner, Branch Owner
Description	This use case allows authorized users to create a new recipe within the system. A recipe consists of a name, storage assignment, serving size details, and a list of ingredients with their respective quantities. Recipes serve as the foundation for food preparation, inventory tracking. Once saved, the recipe can be referenced for production, menu planning, and inventory deductions.
Pre- Conditions	- User is logged in and has the required role. - KMS is enabled at the company. - At least one unit of measurement exists in the system. - At least one available ingredient exists in the system
Post- Condition	-A new prep recipe is added to 'Prep Recipes' list. -Added recipe will be added to **'Used Ingredients'** drop-down on **Create Product** page.
Normal flow	1.User navigates to **KMS > Prep Recipes,** clicks '**Add recipe**' and fill in the following fields: enter recipe name *(required)*, sets the serving size which represents how much food this recipe prepares per batch *(required),selects the unit for the serving size (required), and finally clicks 'add ingredients'* to begin adding components: .Selects an **Ingredient** from the dropdown. .Enters the **Amount** needed per serving. 2. User clicks **'save'**.
Alternative flow	
Flow diagram	

9.2 CREATE PREP PREP RECIPE (PREPARATION ORDER)

Use Case ID	KMS.0001
actor	Admin, Company Owner, Branch Owner and
Description	This use case allows an authorized user (e.g., kitchen manager or inventory staff) to initiate a **preparation order** for a previously defined recipe. The system calculates the total output based on the recipe's serving size and the desired batch quantity. This process helps in tracking inventory consumption and increasing available stock for prepared items.

Pre- Conditions	- User is logged in and has the required role. - **KMS** is enabled at the company. - At least one available prep recipe exists in the system. - At least one storage area exists in the system.
Post- Condition	-A new preparation order is added to 'Prep Prep Recipes' list. -The required ingredients are deducted from inventory based on the used amount fro each batch quantity, *(e.g. 5 grams of **flour** are used to add a dpzza dough recipe, then 2 batches was preparared through 'prep prep recipe'* The resulting prepared item is added to the specified storage area.*, then 10 grams of **flour** will be deducted from the inventory)* -Added recipes will be added to the storage area Inventory. - Inventory logs updates for tracking.
Normal flow	1.User navigates to **KMS > Prep Prep Recipes,** clicks '**Add preparartion order**' and fill in the following fields: select **recipe** from 'Recipe' dropdown *(required) where* system auto-fills the **Serving Size** from the recipe. User sets the **batch quantity** *(required)* which represents number of batches to prepare *(required).* System calculates the **Total Serving** as: `Batch Quantity × Serving Size.` User selects the **Storage Area** where the output will be stored. 2. User clicks **'save'**.
Alternative flow	
Flow diagram	

10 CRM (CUSTOMER RELATION MANAGEMENT)

10.1 ADD JOBS

Use Case ID	CRM.0001
actor	Admin, Company Owner, Branch Owner
Description	This use case allows authorized users to create and publish job openings (careers) so they can appear on the customer portal for applicants to submit job applications.
Pre- Conditions	. User is logged in and has the required role. . At least one active branch exists in the system.

Post- Condition	. A new job/career entry is successfully added to the system. . The added career becomes visible on the **customer portal**, allowing job seekers to apply.
Normal flow	1.User navigates to **CRM > Jobs,** clicks '**Add career**' and fill in the following fields: **.Job Title** *(required):* Enter the position name. **.Branch** *(required):* Select the branch where this job is available. **.Minimum Experience** *(required):* Enter the minimum years of experience required. **.Application Deadline** *(optional):* Set the last date to apply for this position. **.Status** *(toggle):* Switch ON to publish the job (visible to applicants), or OFF to keep it disabled/draft. **.Job Description** *(optional):* Add details about responsibilities, qualifications, etc. **2.** User clicks **'save'**.
Alternative flow	
Flow diagram	

10.2 MANAGE GALLERY

Use Case ID	**CRM.0002**
actor	Admin, Company Owner
Description	This use case allows authorized users to manage gallery. This includes adding, editing and deleting images
Pre- Conditions	. User is logged in and has the required role.
Post- Condition	. Added galleries will be visible on the HomePage.
Normal flow	1.User navigates to **CRM > Gallery,** where they can view the added images, if they want to add a new image they click '**Add gallery**' then fill in the image **title (reuired) and uploadsnd uploads the image (required).**

11.1 configure gamification settings

	If the user wants to **remove** an image from the gallery they click on the up-right 3 dots and clicks **'delete image'.** **2.** User clicks **'save'**.
Alternative flow	
Flow diagram	

10.3 ADD HOME SECTION

Use Case ID	CRM.0003
actor	Admin, Company Owner
Description	This use case allows authorized users to manage homepage sections specifically targeted to either the **Website** or **Mobile App**. Each section controls how content is structured and displayed on the corresponding platform.
Pre- Conditions	. User is logged in and has the required role.
Post- Condition	. Added galleries will be visible on the HomePage.
Normal flow	1.User navigates to **CRM > Gallery,** where they can view the added images, if they want to add a new image they click '**Add gallery**' then fill in the image **title (reuired) and uploadsnd uploads the image (required).** If the user wants to **remove** an image from the gallery they click on the up-right 3 dots and clicks **'delete image'.** **2.** User clicks **'save'**.
Alternative flow	
Flow diagram	

11 GAMIFICATION

11.1 configure gamification settings

Use Case ID	GAM.0001
actor	Admin, Company Owner
Description	This use case allows an authorized user to configure the gamification system within the platform. The gamification

11.1 configure gamification settings

	module rewards customers with XP (Experience Points) for specific actions, enables level-based progression, and assigns badges and prizes for engagement.
Pre- Conditions	. User is logged in and has the required role. . At least one prize is predefined in the system. . At least one customer-related action *(e.g., order placement)* exists in the system.
Post- Condition	. A gamification program is created and stored in the system. .Customers interacting with the system apps will begin accumulating XP based on the defined rules. .The XP system is reflected in the customer-facing app, including badges, progress indicators, and prize eligibility.
Normal flow	**1.** User navigates to **Gamification > Settings.** **2.** User selects the gamification program **type** [**Disabled. Permanant, Expiry**], each type has its own configuration settings ❖ **Disabled:** Deactivates any existing program. XP earning is paused, and customers no longer see progress. ❖ **Permanant**: The gamification program stays active indefinitely. If selected it requires the following settings: user enters the program **name,** sets the **number of levels** of the program**, and for each level:** ▢ Set **XP** required for each level. ▢ Enter **level title** (XP **Tier**) ▢ Select **Badge** type: .If **points:** enter reward points .If **Others** select a **prize** from predefined option. And finally defines the specific **actions** that will award XP. ***(e.g., "New Customer" = 10 XP).*** ❖ **Expired**: The program is available for a limit period. User sets the program **duration/period** which is how long the program lasts *(e.g., "1 Year")*. This type requires only **one level** configuration with XP, badges and prize. *After the expiration period ends, the program becomes inactive. XP stops being earned, and*

	rewards may no longer be available or visible to users. **3.** User clicks **'save'**
Alternative flow	
Flow diagram	

11.2 manage challenges

Use Case ID	GAM.0002
actor	Admin, Company Owner
Description	This use case allows an authorized user to **manage gamification challenges** within the system. Challenges can be applied to specific products or categories and are used to reward customers based on defined completion criteria *(e.g., ordering certain items).*
Pre- Conditions	. User is logged in and has the required role. . Products or categories must already exist in the system to be assigned to challenges.
Post- Condition	. A new challenge is created and saved in the system. . The challenge becomes visible to customers and staff through their respective apps. . Customers interacting with relevant products or categories will begin accumulating progress toward these challenges.
Normal flow	**1.** User navigates to **Gamification > Game challenges.** **2.** User clicks **'Add Challenge'** and fills in the challenge form: • Challenge **Name** (*required*) • **Number of completions** required to complete the challenge *(required)* • **Applied On**: Product or Category • **Select**: Specific product or category from dropdown • **Starting Date** and **Ending Date** • **Active** toggle to enable/disable the challenge • **Prize Type**: Choose between **Reward Points** or **XP**, then specify the amount **3.** User clicks **'save'**
Alternative flow	
Flow diagram	