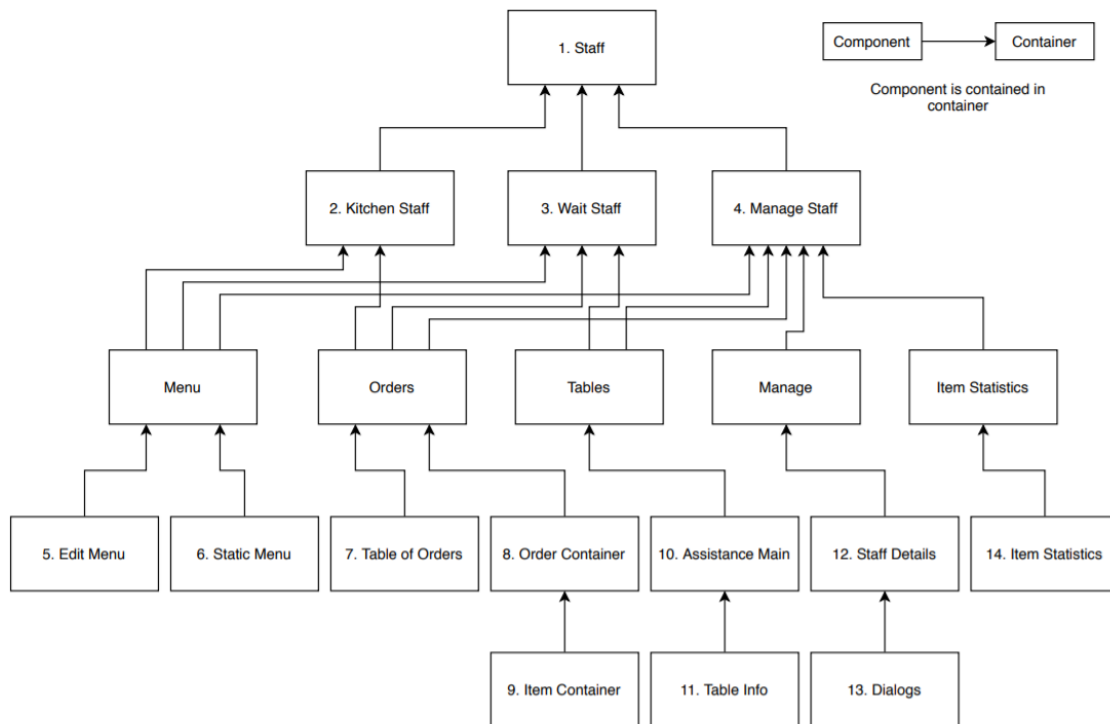# Software Quality – Staff page (frontend/src/pages/Staff)

Staff page itself is a single page application to decrease loading time required. When the page is opened, all information (except menu since it takes too long to load) is fetched and stored, which reduces subsequent loading times when going through the nav bar.

1. Staff page container (/Staff.tsx)

This handles whether or not user is a staff, if the user is a staff, then it displays the username and staff type on the app bar on the top of the page. It also has a button to allow the user to log out.

States

- staffType: type of staff that is logged in

Functions

- logout: logs the user out
- isStaff: checks if the current user is a staff
- displayStaff: displays different staff containers depending on the type of staff
- displayStaffType: used to display the type of staff in the App Bar at the top of the page

2. Kitchen staff container (/StaffComponents/Kitchen.tsx)

This container involves the nav bar, handling what screen to show and handles fetching the items to show in orders and how to move items between lists.

States

- currPage: currently open screen (either menu or orders)
- queueList: list of items in queue
- cookingList: list of items in cooking list
- readyList: list of items that are ready to be served

- lastClicked: id of last item clicked
- helpDialog: whether the help dialog is open
- preventDups: last item clicked (to prevent duplicates in list)
- noDups: list that last item entered (to prevent duplicates)
- menu: whole menu
- menuvalue: used for menu tabs

Functions

- updateOrders: fetch all lists from database again
- updateMenu: fetch menu from database
- displayCont: handles displaying the different screens depending on which tab is clicked
- changeMenuValue: for displaying menu
- helpDialog: generates the help dialog
- displayNav: displays the navigation bar
- moveToCooking: move selected item to cooking
- moveToReady: move selected item to ready list
- moveToQueue: move selected item to queue list
- removeItem: remove item from a list

3. Wait staff container (/StaffComponents/Wait.tsx)

This container handles changing screens depending on what is clicked on the navigation bar on the left. It stores most of the information that needs to be displayed, and handles moving items between lists as well.

States

- currPage: current screen that is shown (menu, orders or tables)
- toServeList: list of items that need to be served
- servedList: list of items that are served
- isOpen: whether alert is open
- lastClicked: id of last item clicked
- helpOpen: whether the help dialog is open
- noDups: last list that selected item entered
- preventDups: item that was last clicked
- alertMessage: message displayed in alert
- menu: whole menu
- menuvalue: used to help menu
- tables: a list of tables and their information
- assistance: a list of table numbers that require assistance
- bill: a list of table numbers that have requested a bill

Functions

- billRequestAlert: open alert to indicate that bill was requested
- assistanceAlert: open alert to indicate that assistance was requested
- updateMenu: fetch menu from server
- updateOrders: fetch to be served list and served list from database
- updateAssist: fetch table information from database
- changeMenuValue: used for menu
- moveToServed: moves item to 'served' list
- moveToToServe: moves item to 'to be served' list

- removeItem: removes item from list
- helpDialog: generates the help dialog
- showAlert: generates the alert
- displayCont: handles what screen to display
- displayNav: generates the navigation bar on the left

4. Manage staff container (/StaffComponents/Manage.tsx)

This container handles changing screens depending on what is clicked on the navigation bar on the left. It stores most of the information that needs to be displayed, and handles moving items between lists as well.

States

- currPage: currently displayed screen
- tables: information on all tables
- assistance: list of table numbers that need assistance
- bill: list of table numbers that have requested the bill
- orderRealData: list of all items in all lists
- staffRealData: list of all staff and their details
- itemRealData: list of all items and their details
- trevenue: total revenue
- menu: whole menu
- menuvalue: used for menu
- currCat: current category
- allItems: list of all items in item pool
- ingredientsList: list of all ingredients
- selectedItem: selected heading for item statistics (for sorting)
- selectedOrders: selected heading for orders (for sorting)
- itemSort: ascending or descending for item statistics
- orderSort: ascending or descending for orders

Functions

- updateOrders: fetch orders from database
- updateAssist: fetch table information from database
- forceUpdateMenu: fetch menu from database
- forceUpdateItemlist: fetch item list from database
- forceUpdateIngredList: fetch ingredients list from database
- updateStaff: fetch staff information
- displayCont: handles which screen to display
- displayNav: displays the navigation bar on the left

5. Edit Menu (StaffComponents/Menu/EditMenu.tsx)

Renders a menu that has editing functions.

Components

  i.   Edit/create item dialog (/Components/EditItem.tsx)
 ii.   Delete item permanently or from category (/Components/Delete.tsx)
iii.   Edit/create/delete category (/Components/EditCategory.tsx)
 iv.   Add item to category (/Components/AddItemCat.tsx)
  v.   Add/delete ingredients from ingredients list (/Components/EditIngredients.tsx)

vi.   Edit ingredients for each item (/Components/Ingredients.tsx)

vii.   Help dialog (/Components/Help.tsx)

Props

- menu: an object that contains the whole menu, including all categories and all items in each category
- value: used to indicate which tab is open
- allItems: a list of all items in item pool
- ingredientsList: a list of all ingredients in ingredients pool
- forceUpdateMenu: function that updates (fetches from server) the menu
- forceUpdateItemList: function that updates (fetches from server) the item list
- forceUpdateIngredList: function that updates (fetches from server) the ingredients list
- changeValue: a function used to change the value above

States

- openModal: whether the Modal for displaying item information is open
- modal: the item that is selected
- editItemDialog: whether the edit item dialog (components i.) is open
- editCatDialog: whether the edit category dialog (components iii.) is open
- deleteDialog: whether the delete dialog (components ii.) is open
- ingredDialog: whether the ingredients dialog (components vi.) is open
- editIngredDialog: whether the edit ingredients dialog (components v.) is open
- addItemCatDialog: whether the add item to category dialog (components iv.) is open
- helpDialog: whether the help dialog (components vii.) is open
- currItem: currently selected item
- currCat: currently selected category
- isEdit: if user is editing
- isDel: is user is deleting
- itemIngredients: ingredients in currently selected items
- alertDialog: if alert dialog is open
- severity: type of alert
- alertMessage: message that will show up in alert

Functions

- *isOpen: changes the state (open or not) of dialogs
- setAlert: sets whether the alert is open, what type of alert and it's message
- createItemIngredients: given currently selected item, it will create an array containing the id's of the ingredients in that item (later to be used for editing ingredients in an item)
- generateItemsInCategory: given a category and all items within it, it creates the cards that display those items
- handleTabChange: when another tab is clicked, it updates the value from props
- openModal: changes states so that the modal will open
- handleCloseModal: changes states so that the modal will close
- tabProps: helper function for generating the tabs that display the categories

6. Static Menu (StaffComponents/Menu/StaticMenu.tsx)

Renders a copy of the menu.

Props

- menu: an object that contains the whole menu, including all categories and all items in each category
- value: used to indicate which tab is open
- changeValue: a function used to change the value above

State

- openModal: Boolean for whether the Modal is open
- modal: currently selected item, the one that is supposed to show on the modal when an item is clicked

Functions

- generateItemsInCategory: given a category and all items within it, it creates the cards that display those items
- handleTabChange: when another tab is clicked, it updates the value from props
- openModal: changes states so that the modal will open
- handleCloseModal: changes states so that the modal will close
- tabProps: helper function for generating the tabs that display the categories

7. Table of orders (StaffComponents/Orders/ManageOrders.tsx)

Renders a screen with a sorting table for all orders of all statuses.

Props

- realData: array of all orders
- changeRealdata: function used to change the order of orders in realData, used for sorting
- order: whether it is ascending or descending
- selected: which heading was clicked, determines how to sort
- changeSelected: function used to change which heading was selected
- changeOrder: function used to change whether to order ascending or descending

Functions

- sortData: sorts orders in realData depending on order and selected
- refresh: fetches realData from server
- printArrow: displays the arrow that indicates how it is sorted
- handleClick: sorts data then sets the new sorted data
- printOrders: generates the table of orders

8. Order containers (StaffComponents/Orders/Components/OrderContainer.tsx)

Each container is a table with a table inside, used to display the items in that list.

Props

- update: function that updates the position of items in the list. It is used when item is clicked
- someList: list of items that should be displayed in the list
- lastClicked: id of item that was last clicked

Functions

- getHeading: generates the list/table header
- getBox: generates the body of the list/table used to display items

9. Item Container (/StaffComponents/Orders/Components/ItemTemplate.tsx)

A container that displays information of an item on a button.

Props

- item: the item that needs to be displayed
- itemId: fake id used to sort the item within the list
- update: function to move item between lists
- realId: id of the item
- lastClicked: id of item that was last clicked

Functions

- displayComment: displays a comment if there is one

10. Assistance Main (/StaffComponents/Assistance/AssistanceMain.tsx)

Displays tables, indicating which tables are occupied, need assistance and requested bill.

Props

- tables: list of tables and their status's
- assistance: array containing id of tables that require assistance
- update: paid function
- billRequest: array containing id of tables that requested a bill

State

- numTables: number of tables
- selectedTable: id of table that is clicked
- main: whether assistance main is displayed or table information is displayed
- helpOpen: whether help dialog is open

Functions

- createTables: generates the boxes that represent tables (includes styling)
- handleClick: function that updates selected table
- backToTables: function that helps to change back to main assistance page
- paid: function that tells the server a table has paid
- tableKey: the key displayed that the bottom, explaining what the stylings mean
- needAssistance: check if selected table needs assistance
- helpDialog: renders the help dialog

11. Table info (/StaffComopnents/Assistance/TableInfo.tsx)

Displays items the chosen table has ordered or status of table.

Props

- tableNumber: currently selected table
- assistance: whether the table requires assistance
- isEmpty: whether the table is empty
- paidFunction: function that tells the server a table has paid

State

- tableInfo: contains information about current table (items orders and total cost)
- hide: whether or not to hide the resolved button
- itemsOrdered: number of items ordered

- order_id: id of order
- isOpen: whether or not the alert is open
- alertMessage: message that will show up in the alert
- alertSeverity: type of alert

Functions

- changeAlertState: sets whether the alert is open
- printItems: prints the items that table has ordered
- problemResolved: sends to server that the problem has been resolved
- freeTable: sends to server that the table is now freed

12. Staff details (/StaffComponents/StaffDetails/StaffDetails.tsx)

Displays a table with staff information, with options to change staff type and delete staff. On the screen, there is also the option to set registration keys.

Props

- realData: list of all staff
- update: updates by fetching list of staff from server again

State

- deleteOpen: whether delete dialog is open
- resetOpen: whether change staff type dialog is open
- resetKeyOpen: whether change registration key is open
- selectedStaff: information on staff that is selected
- isOpen: whether alert is open
- alertMessage: message that will show up in alert
- selectedStaffType: 1 is wait staff, 2 is kitchen, 3 is manage
- severity: type of alert

Functions

- changeStaffType: sends to server to change staff type for selected staff
- showAlert: shows the alert
- deleteStaff: sends to server to delete staff
- printable: prints the table displaying staff details
- render: renders everything

13. Dialogs (/StaffComponents/StaffDetails/Components)

Renders dialogs for buttons on manage screen.

- Change staff type dialog (/Components/ChangeStaffType.tsx)
- Delete dialog (/Components/DeleteDialog.tsx)
- Reset registration keys (/Components/ResetRegist.tsx)

14. Item Statistics (/StaffComponents/Analytics/ItemStats.tsx)

Renders the page with a sorting table that contains item statistics.

Props

- realData: an array containing item statistics
- trevenue: total revenue

- setRealdata: function to change realData (used for sorting purposes)
- order: ascending or descending
- selected: heading that is selected for sorting
- setOrder: function to change order
- setSelected: function to change selected

Functions

- sortData: sorts items in realData
- printArrow: prints the arrow that indicates whether it's sorted in ascending or descending order
- handleClick: sorts data, then sets the sorted data as realData
- printItemTable: prints the table that displays information on item statistics
- render: renders everything