Folder structure part 2:

The next folder is redux which containts the main state management code flow the store, slices and selectors in it and I have worked on it before. And also I have done a review on the routers folder which containts the routing flow and code bases in it with 3 folders in it admin, seller and waiter in it.

The next folder is also the services folder which has 5 folders in it.(deliveryman, deliveryzone, rest, seller, waiter) and also many components outside these folders inside the root folder services.  
Deliveryman folder has 3 components in it:-

order.js: defines an [orderService](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") object that provides methods for interacting with deliveryman order-related API endpoints. It includes methods to get all orders, get an order by ID, and update the status of an order. These methods use the [request](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") module to make HTTP requests to the specified endpoints.

payment-from-partner: defines a service object called [paymentFromPartnerService](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") that provides methods for interacting with the API endpoint related to payments from partners. This service uses the [request](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") module to make HTTP requests to the specified endpoint.

statistics.js: defines a service object called [statisticService](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") that provides methods for interacting with the API endpoint related to deliveryman statistics. This service uses the [request](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") module to make HTTP requests to the specified endpoint.

The next on which is deliveryzone folder has 5 files in it which are related to locations and so on. lets see one of them atleast:

area.js: defines a service object called [areaService](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") that provides methods for interacting with the API endpoints related to delivery zones or areas. This service uses the [request](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") module to make HTTP requests to the specified endpoints.

The next folder is the rest folder let’s see twofiles in it:

catagories.js: defines a service object called [categoryService](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") that provides methods for interacting with the API endpoints related to categories. This service uses the [request](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") module to make HTTP requests to the specified endpoints.

information.js: defines a service object called [informationService](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") that provides methods for interacting with the API endpoints related to various informational resources. This service uses the [request](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") module to make HTTP requests to the specified endpoints.

The next folder is the seller folder. Let’s see some files in it:

advert.js: defines a service object called [advertService](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") that provides methods for interacting with the API endpoints related to advertisements and ad packages for sellers. This service uses the [request](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") module to make HTTP requests to the specified endpoints.

discount.js: efines a service object called [discountService](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") that provides methods for interacting with the API endpoints related to discounts for sellers. This service uses the [request](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") module to make HTTP requests to the specified endpoints.

order.js: efines a service object called [orderService](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") that provides methods for interacting with the API endpoints related to orders for sellers. This service uses the [request](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") module to make HTTP requests to the specified endpoints.

The last folder is waiter folder which contains two files let’s analyze them:

order.js: defines a service object called [orderService](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") that provides methods for interacting with the API endpoints related to orders for waiters. This service uses the [request](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") module to make HTTP requests to the specified endpoints.

statistics.js: efines a service object called [statisticService](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") that provides methods for interacting with the API endpoint related to waiter statistics. This service uses the [request](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") module to make HTTP requests to the specified endpoint.

All of the above files in each folder uses request in the root folder of services let us see it’s use.

request.js: configures an Axios instance for making HTTP requests with additional functionality such as request and response interceptors. This setup ensures that all API requests and responses are handled consistently across the application, including adding authorization headers, handling errors, and displaying notifications.

* Axios Instance Configuration: Creates an Axios instance with a base URL and timeout setting.
* Request Interceptor: Adds an authorization token to the headers if available. Adds the current language to the query parameters for GET requests. Handles request errors by displaying a notification.
* Response Interceptor: Returns the response data directly. Handles response errors by displaying notifications, removing the token, and dispatching actions to clear user data if necessary.

Restaurant.js: defines a service object called [shopService](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") that provides methods for interacting with the API endpoints related to restaurant shops. This service uses the [request](vscode-file://vscode-app/c:/Users/ermid/AppData/Local/Programs/Microsoft VS Code/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html" \o ") module to make HTTP requests to the specified endpoints.

This setup centralizes API calls with:

* Error handling
* Auto-authentication
* Global notifications
* Better code organization

Views fodler holds vast amount of folders and files in it. The views folder typically contains the main components of an application that represent different pages or sections of the app. These components usually serve as the interface for users to interact with specific parts of the system. Each folder or component within the views folder generally corresponds to a particular feature, page, or section of the app, and includes all necessary subcomponents, forms, and utilities related to that feature.