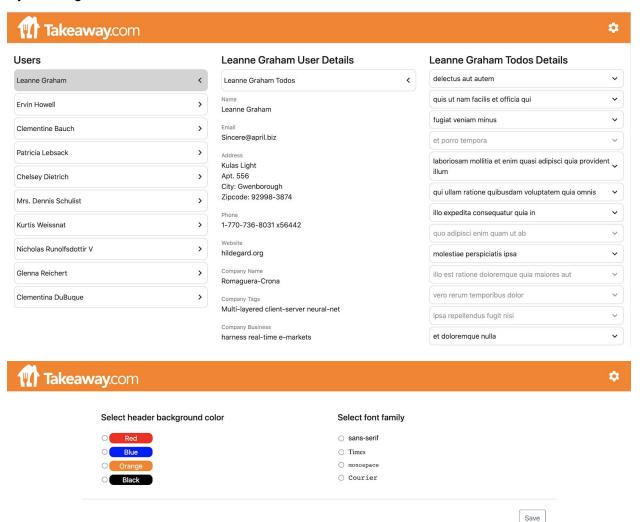
Takeaway - Order list App

Firstly, I added **vue-resource** plugin to my project making web requests and handle responses, explicitly to send a GET request to the APIs that return JSON metadata to get the users and todos data.

Then, **Bootstrap** library was added so that I used the Bootstrap CSS classes for styling the user interface of my application. One of the reasons is that Bootstrap makes the entire process of creating responsive designs easy.

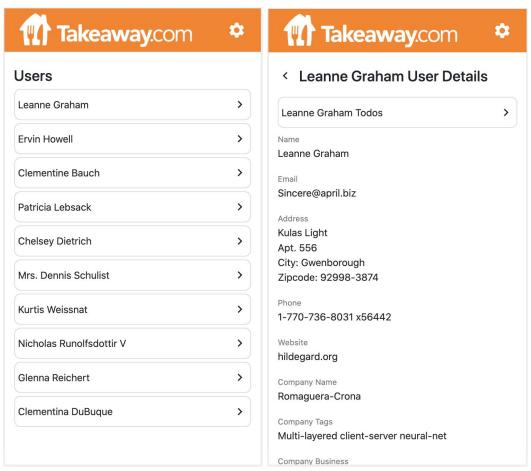
My UI design like below:



After that, Components are implemented according to the UI design. To achieve the UI design, I created 5 components:

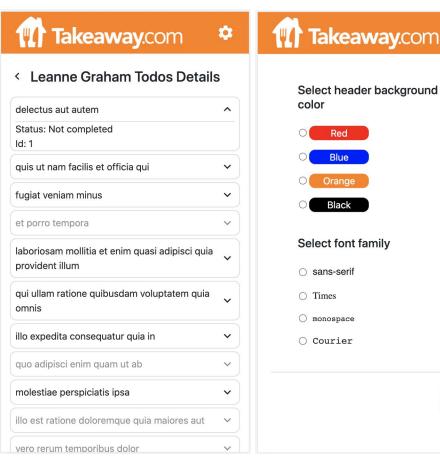
- **UsersList:** Firstly, the component gets the users list from the store and displays it. If the user clicks on any user, that user's user details is been displaying in the next column. To show that user details, it calls the UserDetails component. Moreover, if the user clicks on the user's todos, it calls the TodoDetails component to display the todos list.
- UserDetails: When this component created, it also gets the selected user's todos from
 the store. Then it displays the selected user's user details as well as the todos button.
 Lastly, it emits the todos data to the UsersList component.
- **TodoDetails**: This component gets the todos list as a prop from the UserList component. It shows all the todos lists of the selected user by calling the TodoListItem component for each todo.
- **TodoListItem:** It gets the todo as a prop and displays the todo detail. When the user clicks on a todo, it displays the details of the selected todo.
- AppSettings: In this component, the user can change the color of the header and the font family of the entire app.

Also, the mobile UI design of my app like below:



*

Save



i18n package is for adapting the app or products to specific local languages. For now, I added only the English option, other languages can be added easily in the future. The main reason to add this package is that It makes the maintenance of the app easier and simpler.

Also, I added **Vuex(store)** for the state management. So, Vuex is a state management library, and it helps us deal with shared states. In my application, I put the API call methods in the actions and I can dispatch from the components. After dispatching, it will mutate the state values. Lastly, we will get these state values to use in the components.

For the testing, I used **Nightwatch.js** framework for end-2-end(functional) testing. I wrote E2E tests because they assure me that I have a working system. And also, It Can implicitly test many things at once. There are 3 test suites to check:

- Header is visible
- User-list, user details and todos details are visible
- Setting changes work

To run tests first go to project directory and write the below command to the terminal npx vue-cli-service test:e2e --url http://localhost:8080/