

<u>Home</u> / Quick Start



( Last updated: May 5th, 2021

<u>Installation & Running Project</u> <u>locally</u>

<u>Installation</u>

View in Browser

Code

package.json

Code Walkthrough

<u>Design Library</u>

# **Installation & Running project locally**

# Installation

The first step is to clone the project. Once succeeded, then run the following commands from project directory where **package.json** exist.

**Default code example:** 

npm install

npm start

# **View in Browser**

Once all the required packages are installed by running the above command, open the browser and browse to: https://localhost:3000

# **Code Intro**

### **Framework:**

ReactJS is used as a FE application development library. The time when project was started, version 17 was the latest.

package.json

```
<!DOCTYPE html>
<html lang="en">
    {
        "name": "kmyynti",
        "version": "0.1.0",
        "private": true,
        "dependencies": {
          "@storybook/addon-actions": "^6.0.28",
          "@storybook/react": "^6.0.28",
          "@testing-library/jest-dom": "^5.11.4",
          "@testing-library/react": "^11.1.0",
          "@testing-library/user-event": "^12.1.10",
          "axios": "^0.21.0",
          "babel-loader": "^8.1.0",
          "bootstrap": "^4.5.3",
          "env-cmd": "^10.1.0",
          "jquery": "^3.5.1",
          "react": "^17.0.1",
          "react-adal": "^0.5.0",
          "react-bootstrap": "^1.4.0",
          "react-bootstrap-typeahead": "^5.1.4",
          "react-dom": "^17.0.1",
          "react-loader-spinner": "^4.0.0",
          "react-loading-overlay": "^1.0.1",
          "react-modal": "^3.12.1",
          "react-redux": "^7.2.2",
          "react-router": "^5.2.0",
          "react-router-dom": "^5.2.0",
          "react-scripts": "4.0.0",
          "react-spinners": "^0.10.6",
          "react-table": "^6.8.6",
          "react-table-latest": "npm:react-table@^7.6.2",
          "react-table-stable": "npm:react-table@^6.10.3",
          "react-table-v6": "^6.8.6",
          "react-to-print": "^2.12.1",
          "react-tooltip": "^4.2.13",
          "redux": "^4.0.5",
          "redux-form": "^8.3.7",
          "redux-thunk": "^2.3.0",
          "rimraf": "^3.0.2",
          "typescript": "^4.0.5",
          "web-vitals": "^0.2.4",
          "webpack": "^4.44.2"
        },
        "scripts": {
          "start": "react-scripts start",
          "build": "react-scripts build",
          "build:dev": "env-cmd -f .env.development npm run build",
          "build:test": "env-cmd -f .env.test npm run build",
          "build:qa": "env-cmd -f .env.qa npm run build",
          "test": "react-scripts test",
          "eject": "react-scripts eject"
        },
        "eslintConfig": {
          "extends": [
            "react-app",
            "react-app/jest"
         ]
        },
        "browserslist": {
          "production": [
            ">0.2%",
            "not dec
            "not op_mini all"
          "development": [
            "last 1 chrome version",
            "last 1 firefox version",
            "last 1 safari version"
          ]
        }
      }
</html>
```

The code walkthrough will be done after giving introduction of whole application.

# **Design Library**

ReactBootrap is used as a design library. <a href="https://react-bootstrap.github.io">https://react-bootstrap.github.io</a> Other libraries which have been used in project are:



## jquery

**jquery** has been used at many places in the project to implement different requirements of the project.



### react-table-v6

This is the main and very important part of application. The library has been used in **TILAUS** page for creating the orders, on **Paivan Tilaukset** to show the orders of the day, and **Vanhat Tilaukset**.



# react-modal

This library has been used to display the dialogues against various actions including **Errors Messages, Success of some action** etc...



### **Redux**

The core of this application is state management and **REDUX** store has been used for this purpose. All the states of the application are managed through Redux Store.

For demonstration, I will show you the examples by running the Application, by going through the code, and by using browser extension **Profiler** 



Home / Components

# Components

Open Kmyynti FE

**App Components** 

<u>Tilaus</u>

Paivan Tilaus

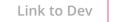
Vanhat Tilaus

Single Order

<u>Redux</u>

# **Open Kmyynti FE**

Below are the links to Dev and Test environments.



Link to Test

# **App Components**

## **Tilaus**

Tilaus page's implementation resides inside 'Container' folder and the page is divided into 04 main sections. Each section is independent component combined in main component.

#### **Overview**



#### **Details**

As shown in the image, there are 05 components for this page.

- 1. OrderAndCustomerInfo.js
- 2. BranchInfo.js
- 3. OrderList.js
- 4. Actions.js
- 5. **Orders.js** (Combine above child components)

# **Paivan Tilaukset**

### Overview



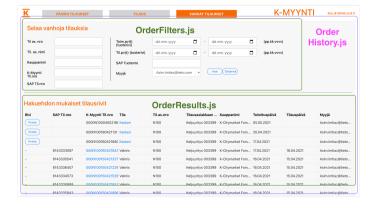
### **Details**

As shown in the image, there are 03 components for this page.

- 1. OrderFilters.js
- 2. OrderResults.js
- 3. **DayOrders.js** (Combine above child components)

# Vanhat Tilaukset

### **Overview**



### **Details**

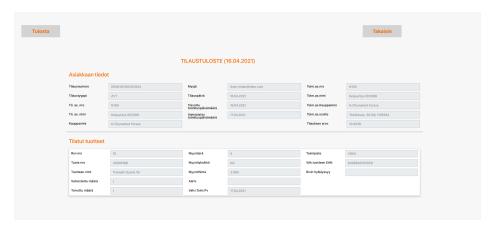
As shown in the image, there are 03 components for this page.

- 1. OrderFilters.js
- 2. OrderResults.js
- 3. **OrderHistory.js** (Combine above child components)

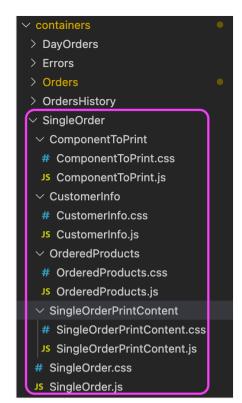
# **Single Order**

This page is rendered when the user click on 'Kmyynti Til.nro' from either 'Vanhat Tilaukset' or 'Paivan Tilaukset'. The components for this page are shown below in right image.

Overview Details



The header section in above image is the information of the customer for the clicked order and section below shows the associated Orders with that orderNumber. Tulosta button will make the possibility to print the order and Takaisin will take back to the page from where the link was clicked.



# **Redux**

Redux store has been used for state management of kmyynti application. Actions are dispatched from each component to update the state. These actions can include updating the value of any flagVariable or after getting the data from API. The Reducer always keep the state up-to-date to make the consistent data available for any component any time. There are separate **action** files and **reducer** files for each main component.

Redux documentation

Designed for documentation purposes



Home / API



**API Calls** 

**Endpoint details** 

# **API Calls**

# 1. Axios

Axios has been used to communicate with the end-points.

# 2. Global Configurations

**Source:** <a href="https://www.npmjs.com/package/axios">https://www.npmjs.com/package/axios</a>

# **Endpoint details**

The application is calling different endpoints for different environments. There are 04 types of environments and for each environment there is a separate file with **.env** prefix. The other details can be seen in the code in the **action files**.