

LAB2: CONVERTING EXISTING JAVASCRIPT WEBAPP INTO ELECTRON DESKTOP APP

In this exercise, you will:

- Set up and run existing web application in the browser.
- Convert web application into working Electron Desktop App.

Set up and run existing web application

1. Check out the `Lab-WebApp-to-ElectronApp` branch from the remote repository.

```
| git checkout origin/Lab-WebApp-to-ElectronApp
```

Note: The HEAD of this branch provides the completed code for the entire exercise.

2. Run `npm install` to ensure all the dependencies have been installed.
3. Navigate to JSON Server folder in project. Start a new terminal window from within this folder and run below command.

```
| json-server --watch --port 4000 tshirts.json
```

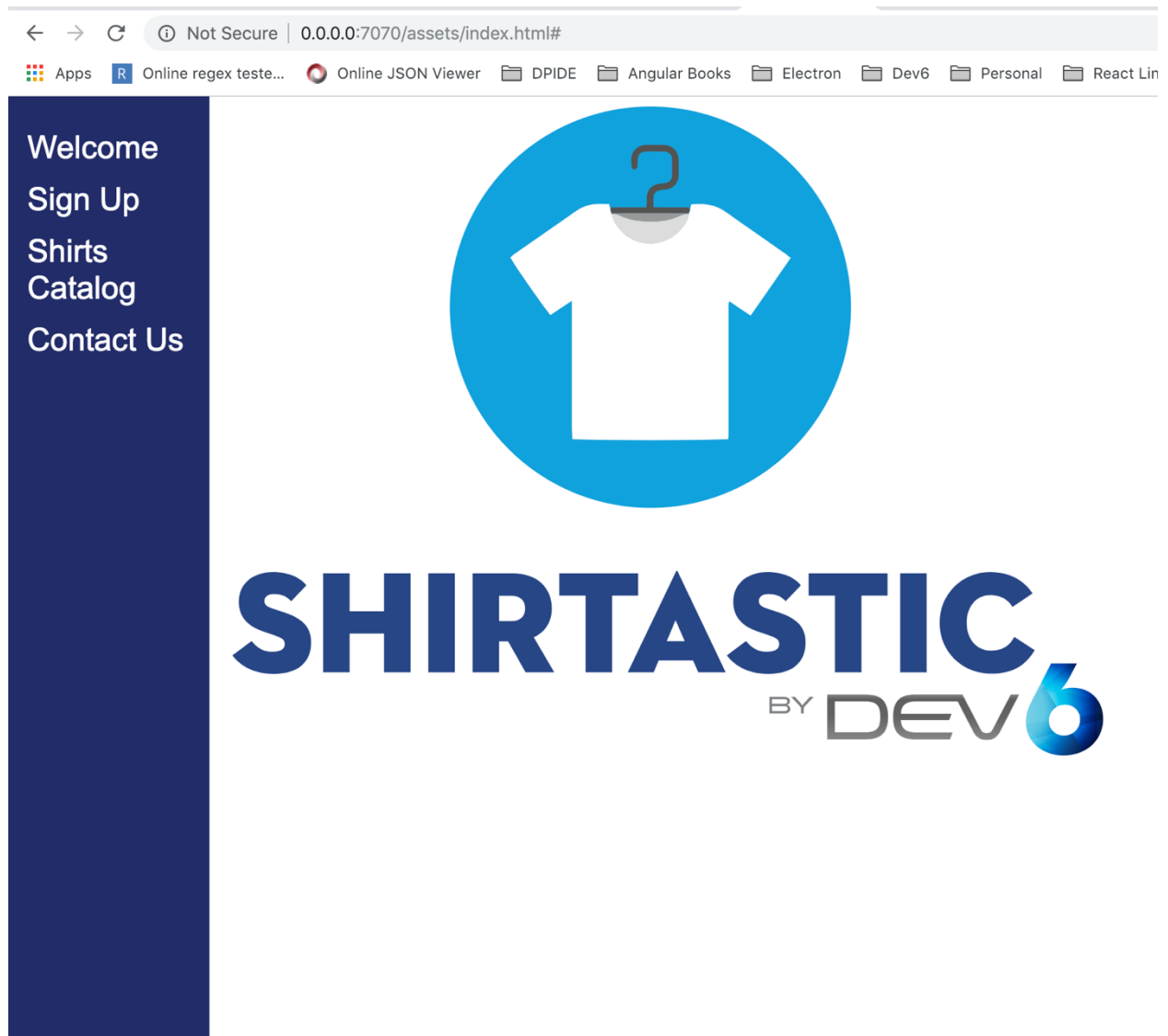
4. Now let's test our web application in the browser. In order to do so you have to run below command in your terminal.

```
| npm run start
```

5. Create a new local branch for your work based on the "starting point..." commit.

```
| git log --oneline  
| git checkout <hash for starting point commit>  
| git checkout -b Lab-WebApp-to-ElectronApp-mine
```

Note: Visit URL <http://0.0.0.0:7070/assets/index.html> in your browser. You should see the Shirtastic web application. Now, in next series of steps we will convert this web application into working Electron Desktop App.



Convert web application into working Electron Desktop App

6. First step is to install Electron. To do so, run the following command.

```
| npm install --save-dev electron
```

```
"dependencies": {  
  "light-server": "^2.6.0"  
},  
"devDependencies": {  
  "electron": "^4.0.4"  
}
```

Note: This command will install Electron in your project repository and save it in dev dependencies section in package.json. There is an interesting thing to note here,

that we have added electron as a dev dependency instead of regular dependency. As we all know, dependencies are required to run, and dev dependency are required to develop. The answer to this variation is that electron is already packaged as part of built output. So, no need of explicitly declaring it as part of dependency. To read more about this you can visit this link.

<https://stackoverflow.com/questions/50803207/why-does-electron-need-to-be-saved-as-a-developer-dependency>.

7. Second step is to create main.js file in the root level of project. You can also refer this link to copy the code for main.js file. Browse to the part where the article talks about main.js content. Just copy paste part which belongs to main.js file into your newly created main.js file.

<https://electronjs.org/docs/tutorial/first-app>

8. Once you have pasted the code from this link to main.js file. Now it is time to edit the path for index.html.

```
// and load the index.html of the app.  
win.loadFile('assets/index.html')
```

Note: Main.js will be the entry point for application. Main.js will create a browser window which will run the index.html and rest of our old web application. So, it makes sense to tell main.js from where to look up index.html in the project structure.

9. Final step is to edit package.json and add a new script to start electron desktop app. Add new entry namely startElectron in existing scripts as shown below in the snippet.

```
"scripts": {  
  "startWebApp": "light-server -s . -p 7070 -w \"assets/app/**/*.js, assets/screens/**/*.html, assets/css/**/*.css, assets/index.html, assets/app.js\"",  
  "startElectron": "electron ."  
},
```

10. And, add main field to package.json.

```
{
  "name": "shirtastic-javascript-app",
  "version": "1.0.0",
  "description": "Shirtastic Javascript App",
  "main": "main.js",
  "scripts": {
    "start": "light-server -s . -p 7070 -w \"assets/app/**/*.js, assets/sc",
    "startElectronApp": "electron ."
  },
  "keywords": [],
  "author": "gagank@dev6.com",
  "license": "ISC",
  "dependencies": {
    "light-server": "^2.6.0"
  },
  "devDependencies": {
    "electron": "^4.0.4"
  }
}
```

Note: Electron runtime looks for main field entry in order to start the electron app.

11. Just one last quick fix in the index.html before we can test our brand-new Electron app. Navigate to index.html under assets folder and add below script tag just before the closing of body tag.

```
<script>window.$ = window.jQuery = require('./lib/jquery.min.js');</script>
```

Note: This fix is not related to Electron usage. It is done to resolve the jQuery dependency in electron app, the way project structure is organised. Without this script the electron app gives error related to jQuery usage.

12. Now time for the moment of truth. Yes, let's test the electron app. Go to terminal window and run the command.

| npm run startElectronApp



Note: You should see shiny new Electron Desktop App running on your machine now.
