CONFIG config.json Contains all the info our dependencies, the configuration for Database access as well as the data for the app

passport.js It's using passport strategy package for authentication. This file was built requiring folder models and authenticates the user.

MIDDLEWARE isauthenticated.js is not used for the login or signup pages. Has a function with three parameters being the third one a function (called next). If the user logs in he/she can proceed, if not, the user will be redirected to login html page.

MODELS index.js Uses Sequelize to create DB with MySql to be exported and suggests server is running on "dev" environment

user.js Requires BCRYPTJS which can be a problem for Window machines but it hashes user's passwords, protecting it before storing into db. This file is creating a function with an object that will define the standards for the log in.

PUBLIC JS login.js It's the logic for the form component code in the login.HTML file where user will add email and password. If no email or password, nothing happens. Once both info are received, a function will make a post to API route and if all correct, user will be addressed to member page.

members.js helper for the member.html file, it does a get request for the user email once its name is received.

signup.js logic for the signup.html form once button submit is clicked. This will do a post request to the route and re direct to member page, otherwise an error message will be thrown

STYLESHEETS style.css contains the styling for the form located on HTML, specifying the margin top value

ROUTES api-routes.js Routes API calls requiring access to the database and uses passport.js to authenticate middleware with our local strategy. Has two Post ROUTES:1. If login is validated is then sent members page.2. Creates a new user and redirect to login page. Has also two GET routes: 1. With a function that logs out the user and redirect 2. With and empty object in case the user is not logged in otherwise it will send back user's email and id only.

html-routes This file / route is passing in the "isauthenticated" middleware so only logged in users can have access. Route for the for all three HTML pages, it authenticates the users and redirects to members page

Server.js This file uses express, which is a minimal and flexible Node.js web application that helps to manage all components. It is activating routes and listening to the port so the app can be opened on the browser

package-lock.json contains all dependencies detailed so any user can check and use same ones