**Unit 14 Sequelize Homework: Reverse Engineering Code Tutorial**

In this tutorial, I will reverse engineer the starter code provided and provide an explanation for \*every\* file in the `Develop` folder, it’s dependency on other files, and its purpose:

**Config.json:**

Environment variable contains the name of our application's deployment environment. **Dependencies**: mysql, burger\_sequelize\_db, JAWSDB\_URL.

**Burgers\_controller.js:**

A controller is a function you write to control your data and is used to define the server routes. **Dependencies:** express, database up and running.

**Burger.js:**

Module export, tells Node. js which bits of code (functions, objects, strings, etc.) to “export” from a given file so other files are allowed to access the exported code. **Dependencies:** sequelize

**Index.js:**

Handles app startup, routing, and other functions of your application and does require other modules to add functionality. If you're running a website or web app it would also become a basic HTTP web server. **Dependencies:** sequelize, database up and running.

**Style.css:**

The style. css is a stylesheet (CSS) file that controls the presentation (visual design and layout) of the website pages. **Dependencies:** mysql, database up and running.

**Burger.png:**

Image file that renders on the nav-bar of every page.

**Hexi.psd:**

Photoshop Document, is an image file format native to Adobe Photoshop Application. It's an image editing friendly format that supports multiple image layers and various imaging options.

**Script.js:**

The ready() method is used to make a function available after the document is loaded. Whatever code you write inside the $(document ).ready() method will run once the page DOM is ready to execute JavaScript code.

**Main.handlebars:**

We are using handlebars templating library to build our application interface and telling it to use main.handlebars as the defaultLayout and also telling it where to find other layouts. **Dependencies:** style.css, script.js.

**Index.handlebars:**

An index of all the divs, classes, and column/rows and headers in the API-reference part of the site. **Dependencies:** burger.png

**Package.json:**

Is a JSON file that exists at the root of a Javascript/Node project. It holds metadata relevant to the project and it is used for managing the project's dependencies, scripts, version and a whole lot more. **Dependencies:** server.js, express handlebars, express, mysql2, mysql, seqeulize, handlebars-helper-css

**Package-lock.json:**

Is automatically generated for any operations where npm modifies either the node\_modules tree, or **package**. **json**.

**Readme.md:**

This file is added to provide information regarding the application or site. Information such as App Name, Author, point of contact, Purpose, Installation Instructions, and system requirements.

**Server.js:**

This file points to the tools, libraries, databases, routes, and ports necessary to start a backend server. **Dependencies:** express, express-handlebars,burgers\_controller.js, port 3000, seqeuelize, database up and running.