



Project layout

Edit

New Page

caprepo edited this page 16 minutes ago · 6 revisions

The project layout is as follows:

- config - this is where specific configuration files are stored (ex: Maven)
- docs - Any relevant documents
- src - All Java, Javascript, application configuration files

- config
 - maven

Table of Contents

- [src](#)
- [src/main/java](#)
- [src/main/webapp](#)
- [src/main/webapp/js](#)
- [index.html](#)

src

- ◦ main
 - java - All Java code
 - resources - spring context xml
 - webapp - All HTML, CSS and Javascript

src/main/java

- ◦ com.capgemini.googlecloud package
 - ImportBillsSessionListener - Has methods for session creation and destroy
 - ImportBillsServlet - This is a servlet that is set to load at startup. It has a parameter ImportBillsApplication
 - ImportBillsApplication - This is where the Jersey REST configuration is done. com.capgemini.googlecloud.web has been setup as the package where all REST resources are sitting. We can add more packages.
- com.capgemini.googlecloud.web package
 - Add your REST resource classes here
- com.capgemini.googlecloud.model package
 - Add your POJO (get/set) here

▼ Pages 3

[Home](#)[Local environment setup](#)[Project layout](#)[+ Add a custom sidebar](#)

Clone this wiki locally

<https://github.com/caprepo/>[Clone in Desktop](#)

- com.capgemini.googlecloud.service package
 - Add your business logic here
- src/main/resources
 - Here all spring configuration files will be located

src/main/webapp

- ◦ css - All css files
- fonts - All font files
- images - All images
- includes - All HTML partials
- js - Javascript files
- jslib - All vendor provided Javascript libraries
- META-INF
- WEB-INF - web.xml mainly

src/main/webapp/js

(files in bold is where our most changes will go)

- ◦ app.js - imports all angular modules written by the team
- **controllers.js - Controllers are defined here. we can also define new controllers in other files but they should be coming under importBillsControllers.controller**
- directives.js - Any AngularJS directives go here
- filters.js - Any AngularJS filters go here
- logger.js - AngularJs Logging
- plugin.js - plugins
- **route-config.js - This is where we define the routes and controllers.**
- **services.js - Typically the controller will call the service. Services will call external google api services.**
- tinyMce.Config - HTML wysiwig editor
- underscore-module.js - Using the Javascript underscore library

index.html

- 1. Defines the app as an angularJS app.
- `<html ng-app="importBillsApp">`
- 2. Imports all CSS
- 3. Include header (header.html)
- `<div ng-include="'includes/header.html'"></div>`
- 4. Include body (partial html defined by the route)
- `<div ng-view></div>`
- 5. Include footer (footer.html)
- `<div ng-include="'includes/footer.html'"></div>`

+ Add a custom footer

