

Task Manager

Add Task [View Task](#)

Task: Parent Task:
Priority From: Priority To: Start Date: End Date:

Task	Parent	Priority	Start	End		
Task 1	Parent Task 1	10	20/10/2017	21/10/2017	Edit	End Task
Task 2	Parent Task 2	15	22/10/2017	25/10/2017	Edit	End Task
Task 3	This Task Has NO Parent	30	18/10/2017	21/10/2017	Edit	End Task
Task 4	Parent Task Can Be Any One From Main Task List	1	20/10/2017	21/10/2017	Edit	End Task

SINGLE PAGE APPLICATION ON TASK MANAGER

Java FSD 8 Hour Code Capsule

ABSTRACT

The Task Manager solution should be a Single Page Application (SPA) to keep track of upcoming tasks, finished tasks and their respective priorities.

Developer

Java Full Stack Developer

Contents

Important Instructions	2
Business-Requirement: An Overview	2
Technical Spec – Solution Development Environment	3
Front End Layer	3
Middle Tier Layer	3
Database & Integration Layer	3
Ancillary Layer	3
Deployment & Infrastructure	3
Editors	3
Wireframes	4
Add Party and Search Party Screen	4
Manage Political Leader and its Party and State	5
Manage Development Activity	6
Rate Development Activity (Up-Vote and Down-Vote)	7
Database Tables	8
Software Requirements	4
Architecture	6
Architecture Diagram for Java Enterprise Full Stack	7
Important Instructions	8
Assessment Deliverables	8

Important Instructions

1. Adhere to the design specifications mentioned in the case study.
2. Please make sure that your code does not have any compilation errors while submitting your case study solution.
3. The final solution should be a zipped code having solution. Solution code will be used to perform Static code evaluation.
4. Implement the code using best design standards.
5. Use Internationalization for all the labels and messages in Rest API Development.
6. Do not use System.out statements or console.log for logging in Rest API and FrontEnd respectively. Use appropriate logging methods for logging statements/variable/return values.
7. If you are using Spring Restful or Jersey JAX-RS to develop Rest API, then use Maven to build the project and create WAR file.
8. If you are using Node and Express to develop Rest API, then use Grunt/Gulp/NPM to build/minify the project and create application for deployment.
9. If you are using C#.Net/VB.Net, ASP.net, ASP.net MVC, WCF, Web API to build Rest API, then use relevant tools to build the project.
10. Write web service which takes input and return required details from database.
11. Use JSON format to transfer the results.

Business-Requirement: An Overview

The Task Manager Single Page Application allows you to manage task and set their priorities in range of 0 to 30. Each task will have parent task (optional/another task can be made parent), start date and end date. Below are the features of Task Manager:

1. Add/Edit/View/Delete Task
2. You can make one task a parent of another task.
3. On view task screen:
 - a. User can search task by task name, parent task, start date, end date and priority.
 - b. User can edit task
 - c. User can end the task once it is finished. Once ended, user cannot edit the task. User can only view the task once finished.
4. All search fields on view task screen are using auto-filter property of angularjs.

Technical Spec – Solution Development Environment

Front End Layer

Framework(s)/SDK/Libraries	Version
AngularJS	1.5 or above
Bootstrap	3.0 or above
CSS	3
HTML	5
JavaScript	1.8 or above
JQuery	1.3

Middle Tier Layer

Technology	Framework(s)/SDK/Libraries	Version
Java Stack	Spring Boot	1.5 or above
	Spring MVC	4.0 or above
	JDK	1.7 or above
	Maven	3.x or above

Database & Integration Layer

Technology	Framework(s)/SDK/Libraries	Version
Java Stack	Hibernate	4.0 or above
	JAX-RS Jersey/ Spring Restful	
	MySQL	5.7.19
MongoDB	MongoDB	3.4
	NoSQL	

Ancillary Layer

Technology	Framework(s)/SDK/Libraries	Version
Source Code Management Tool	GIT	2.14.2
Build Tool/JAVA Stack	Maven	3.x
Testing Tool/JAVA Stack	JUnit	4.x
Testing Tool/JAVA Stack	Spring Test	4.x

Deployment & Infrastructure

Technology	Framework(s)/SDK/Libraries	Version
Docker	-	
Apache Tomcat	-	
Apache HTTP (XAMPP)	-	
Node	-	
Dependency Management Tool	NPM	

Editors

Name	Version
Adobe Brackets	
Sublime Text	
Atom Editor	
Eclipse J2EE Editor	

Wireframes

Add Task

Task Manager

[Add Task](#) [View Task](#)

Task:

Priority:

Parent Task:

Start Date:

End Date:

View Task

Task Manager

[Add Task](#) [View Task](#)

Task: **Parent Task:**

Priority From: **Priority To:** **Start Date:** **End Date:**

Task	Parent	Priority	Start	End		
Task 1	Parent Task 1	10	20/10/2017	21/10/2017	<input type="button" value="Edit"/>	<input type="button" value="End Task"/>
Task 2	Parent Task 2	15	22/10/2017	25/10/2017	<input type="button" value="Edit"/>	<input type="button" value="End Task"/>
Task 3	This Task Has NO Parent	30	18/10/2017	21/10/2017	<input type="button" value="Edit"/>	<input type="button" value="End Task"/>
Task 4	Parent Task Can Be Any One From Main Task List	1	20/10/2017	21/10/2017	<input type="button" value="Edit"/>	<input type="button" value="End Task"/>

Task Manager

Task:

Priority:

0

30

Parent Task:

Start Date:

End Date:

Update

Cancel

Database Tables

Parent Task Table
Parent_ID
Parent_Task

Task Table
Task_ID
Parent_ID
Task
Start_Date
End_Date
Priority

Software Requirements

This case study assumes knowledge of programming and hands-on with below mentioned skills.

The technologies included in Full Stack are not limited to following but may consist of:

- UI Layer (HTML5, CSS3, Bootstrap, JavaScript, JQuery, AngularJS, JSP)
- Middleware Restful API (Spring Restful, JAX-RS, Spring MVC)
- Database Persistence (Hibernate)
- Database layer (MySQL, MongoDB)
- Ancillary skills (GIT, Docker, Maven) etc.

To complete this case study, you should be comfortable with basic single page web application concepts including REST and CRUD. The environment setup is built into virtual environment you are logged in to.

You may use angular-cli to create your template project.

Ref1: <https://cli.angular.io/>

Ref2: <https://github.com/angular/angular-cli>

Architecture

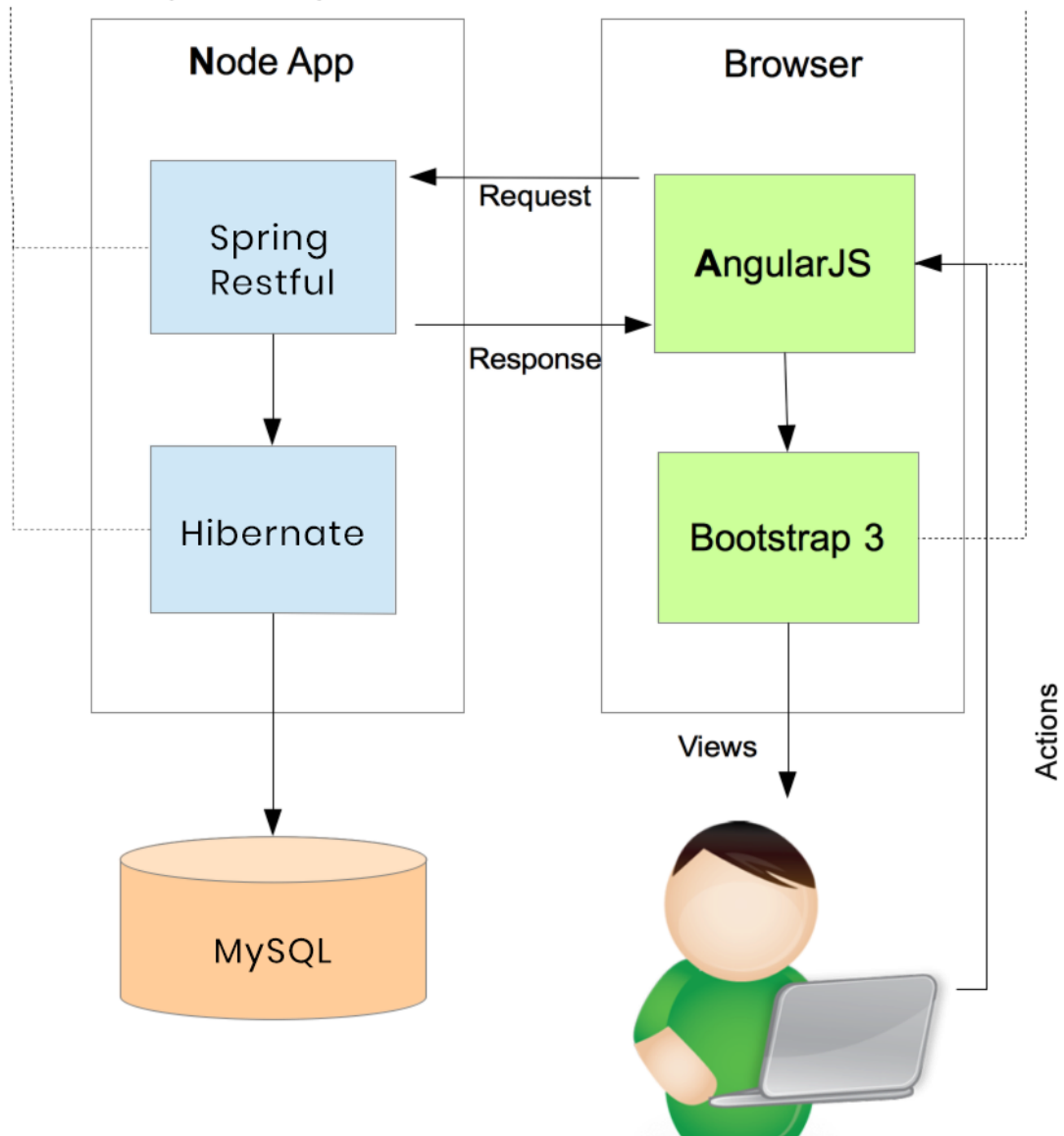
A physical architecture is an arrangement of physical elements, (system elements and physical interfaces) that provides the designed solution for a product, service, or enterprise. It is intended to satisfy logical architecture elements and system requirements. Workout Tracker follows a three layered architecture namely presentation layer, business logic layer and data access layer.

- **Presentation Tier** is the tier in which the users interact with an application. It is a single-page-application of responsive nature. Presentation Tier will consume restful API implemented in Business Tier to display content to the user.
- **Business Tier** is mainly working as the bridge between Data Tier and Presentation Tier. All the Data passes through the Business Tier before passing to the presentation Tier. Business Tier is the sum of Business Logic Layer, Data Access Layer and Value Object and other components used to add business logic. It exposes Rest API which can be called by Presentation Tier to display content to the user. It will also send the data from Presentation tier to Data Tier using Rest API.
- **Data Tier** is basically the server which stores all the application's data. Data tier contents Database Tables, XML Files and other means of storing Application Data.

Architecture Diagram for Java Enterprise Full Stack

Java Enterprise Layer

NPM Modules



Important Instructions

12. Adhere to the design specifications mentioned in the case study.
13. Feel free to create front-end and back-end of single page application from scratch. You are free to use angular-cli commands at command prompt to create SPA (Single Page Application) template.
14. Please make sure that your code does not have any compilation errors while submitting your case study solution.
15. The final solution should consist of three parts:
 - a. Front-end built using HTML5, CSS3, Bootstrap and Angular JS as a SPA. Controllers written in Angular JS should consume restful API coded in business layer.
 - b. Business-layer built using Spring Restful/JAX-RS Jersey or Node.JS/Express.JS.
 - c. Database-layer built using MongoDB or MySQL.

Assessment Deliverables

1. For Front End – zipped application
2. For Backend - Packaged code files (Source code and WAR).
3. For SCM* – Project Code should be present in active GIT repository
4. Few Steps on how to run the solution.

*SCM – Source Code Management