**Technical Story Card**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version No.** | **Date** | **Prepared By** | **Significant Changes** |
| **1** | 25-04-2022 | Team  Code Panache |  |
|  |  |  |  |

|  |  |
| --- | --- |
| **Abbreviation** | **Description** |
| BSC | Business Story Card |
| TSC | Technical Story Card |
|  |  |
|  |  |

Table of Contents

[1 Introduction](#_Toc379364881)

[2 Scope of Change](#_Toc379364882)

[3 List of impacted modules](#_Toc379364883)

[4 Design and Detailed technical updates](#_Toc379364884)

[4.1 Process model](#_Toc379364885)

[4.1.1 Use case Model](#_Toc379364886)

[4.1.2 Sequence diagram](#_Toc379364887)

# Introduction

Task Organizer is a concept, describing a set of technology-based solutions that aim to provide ease to the user in organizing his/her daily tasks in a systematic way.

Under this paradigm, clients add the tasks along with the deadline and priority. The tasks can also be updated in case there is a shift in deadline or change in plan. The facility of striking off the tasks by clicking on the checkbox provides a sense of accomplishment. The tasks can also be deleted to keep the dashboard clean and ready for the next set of tasks.

# Scope of Change

Change will not be entertained and whatever specifications mentioned in this document is final.

# List of impacted modules

All the functional modules will be created from scratch.

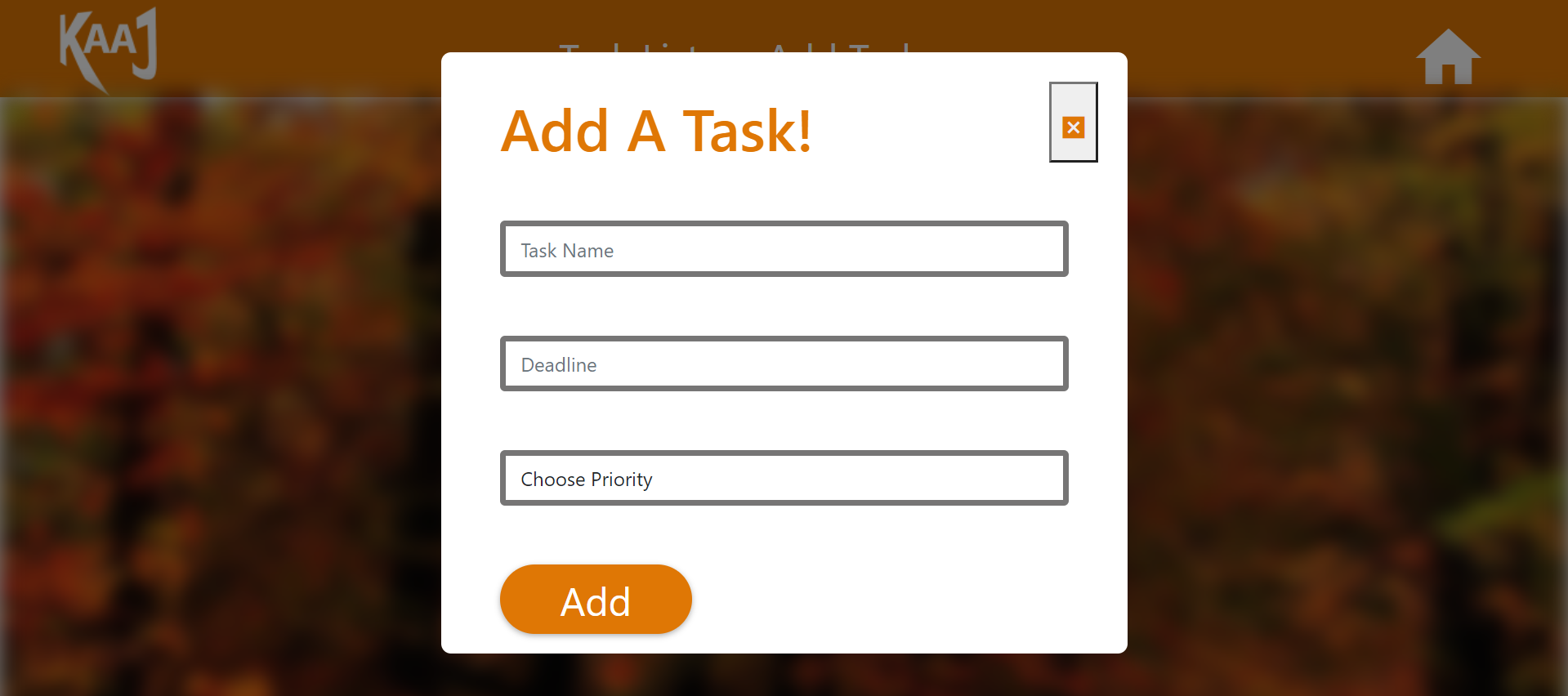
# Design and Detailed technical updates

## Process model

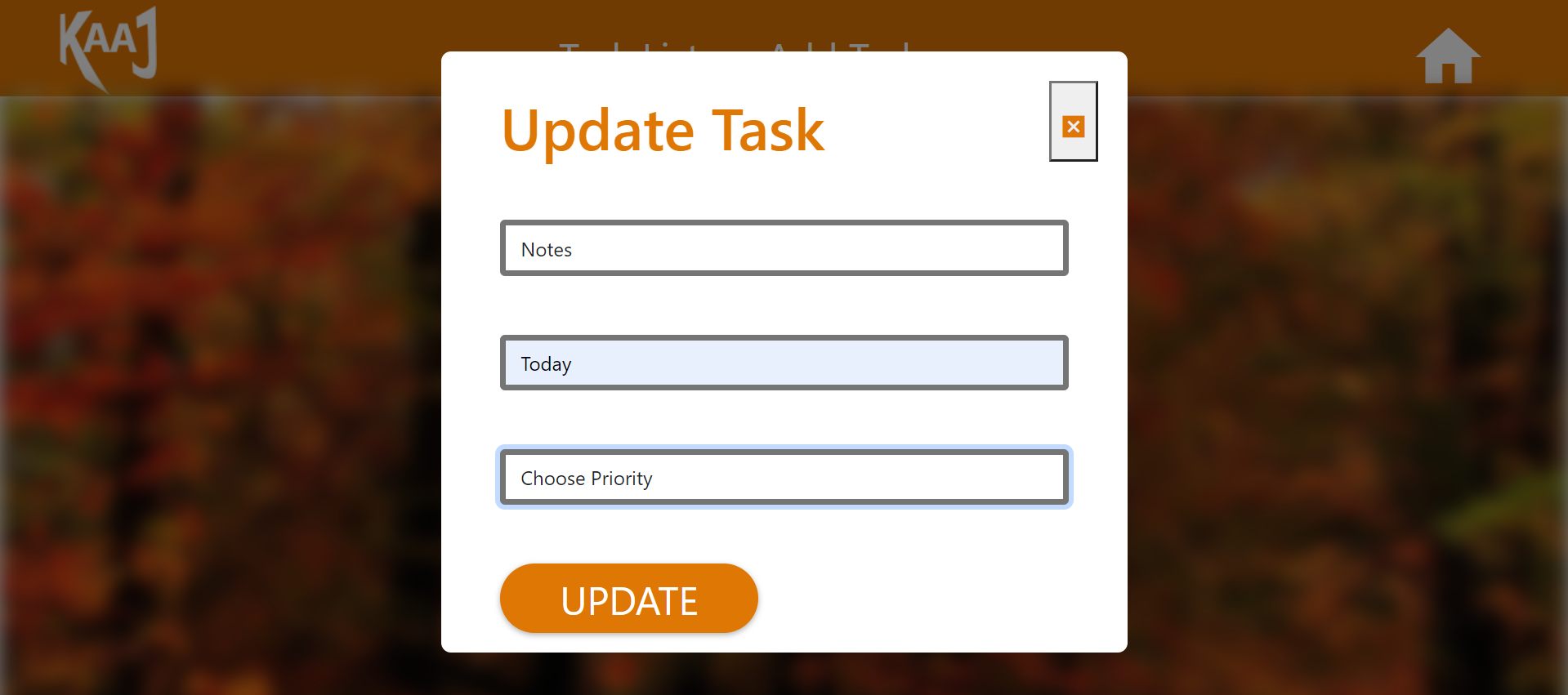
### Use case Model

### 

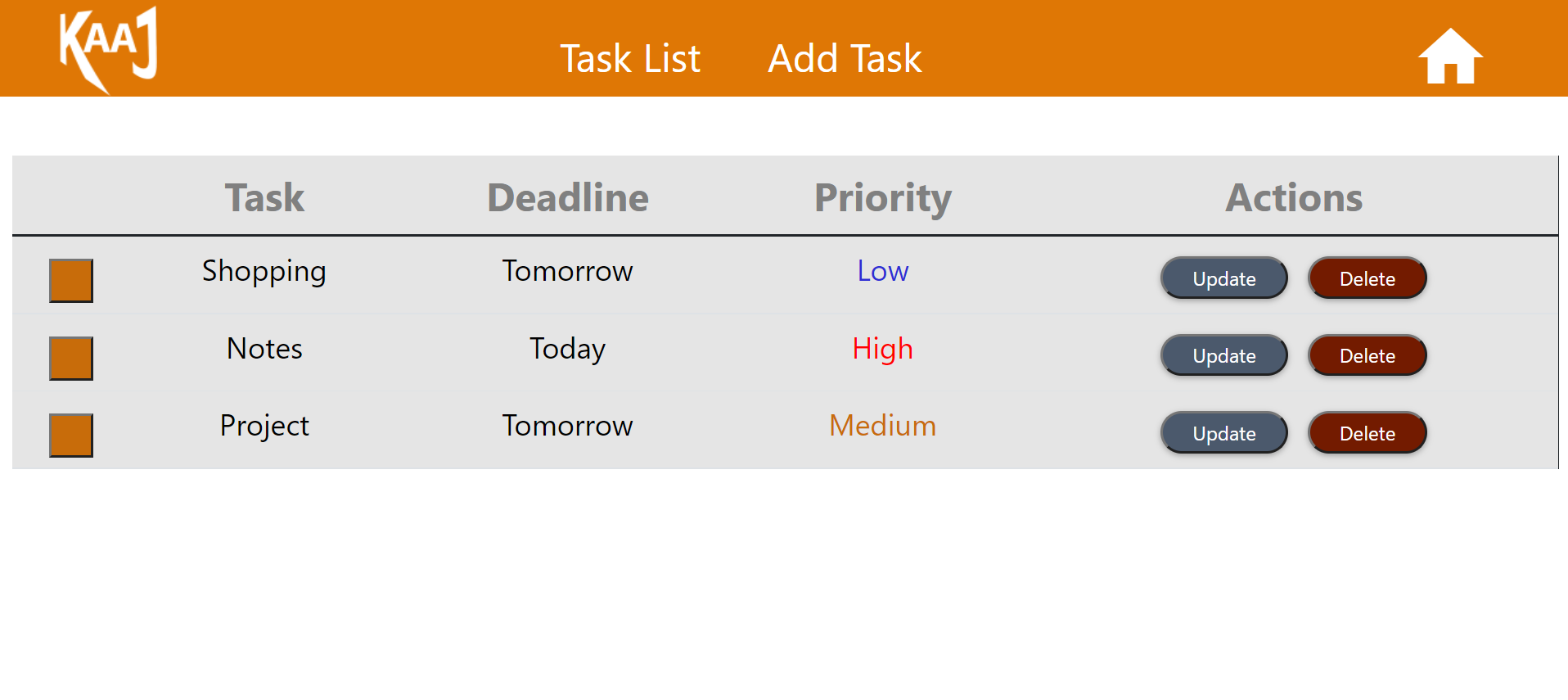
|  |  |
| --- | --- |
| **Brief Description** | **Add Task** |
| **Basic Flow** | 1. **The following details are required to add tasks:**  * **Task Name** * **Deadline** * **Priority**  1. **The following steps will be followed to add tasks:**  * **Click on add task** * **Fill the required task details** * **Click on “Add”** |
| **Alternate Flow** | **Not Applicable** |
| **Validation** | **All the fields should have been filled.** |
| **Pre-Conditions** | **Not Applicable** |
| **Post-Conditions** | **Redirected to Task list page** |



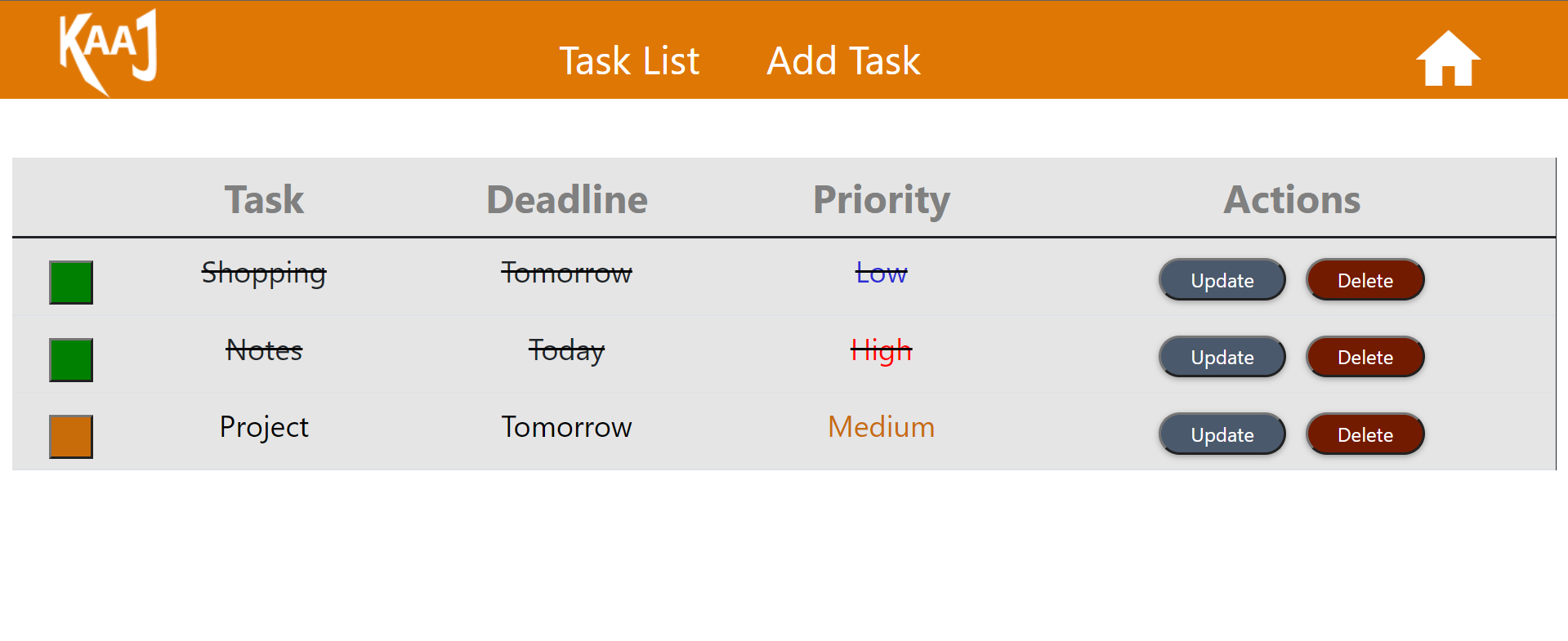
|  |  |
| --- | --- |
| **Brief Description** | **Update Task** |
| **Basic Flow** | 1. **The following fields are required to update task**  * **Task Name** * **Deadline** * **Priority**  1. **After clicking on update button tasks will be**   **updated in the following steps:**   * **Click on update task** * **All the details previously filled will be visible** * **User can update the required details** * **Click on “Update”** |
| **Alternate Flow** | **Not Applicable** |
| **Validation** | **All the fields should have been filled.** |
| **Pre-Conditions** | **Tasks should have been added.** |
| **Post-Conditions** | **Redirected to Task list page.** |



|  |  |
| --- | --- |
| **Brief Description** | **Delete Task** |
| **Basic Flow** | **Task will be deleted by clicking on “Delete” button.** |
| **Alternate Flow** | **Not Applicable** |
| **Validation** | **All the fields should have been filled.** |
| **Pre-Conditions** | **Task should have been added.** |
| **Post-Conditions** | **Redirected to Task List page.** |



|  |  |
| --- | --- |
| **Brief Description** | **Strike Task** |
| **Basic Flow** | 1. **Check box is used to strike tasks.** 2. **Check box in “orange” color will turn “green” if user clicks on it and the respective task will be striked.**   **This can be toggled.** |
| **Alternate Flow** | **Not Applicable** |
| **Validations** | **All fields should have been filled.** |
| **Pre-Conditions** | **Tasks should have been added.** |
| **Post-Conditions** | **Not Applicable** |



Architecture Diagram

**Logical View**

**Technology/ Framework**

**Layer**

UI Components

HTML/ CSS/ JavaScript/ AJAX/ Angular

Presentation Layer

Resources

Validations

Export (pdf)

Report

Web-Server Classes

Java 1.8/ Spring MVC

Application/ Business Layer

Controllers

Configuration

REST Controllers

Service

Entity/ Model Classes

Data Access Layer

Hibernate ORM

Repository classes

Oracle 11g Database

**Presentation Layer:**

This layer consists of all UI components.

* Export component: The plug-in is used to export the reports in pdf format. It helps to reduce the workload on the web/ application server.
* Validation component: All basic level data validations should be done at UI level.
* Report component: All reports to be generated to the user, will be processed and generated at the browser end.
* Resources: All HTML/CSS/ images, which are required for the page design.

**Application Layer:**

This layer comprises of all server and business classes.

* Configuration settings (XML or Class based) will define the application and server configuration.
* Spring Controllers defines the server classes, which are required to process the incoming http requests.
* Service classes are required to perform the required business services.
* REST Controllers to process the HTTP AJAX requests.
* Model classes are used the define the functions of the entities present in the system.

**Data Layer:**

Data layer is implemented through Hibernate ORM. It will contain the repository classes, which provides interface to the table.

### Sequence diagram

**MIDDLEWARE**

**DATABASE**

**USER**

**UI**

Get started

Add Task page

Task List Page

Delete Data

Delete Task

Task List Page

Update Data

Client-Side Validation

Update Task

Task List Page

Insert Data

Client-Side Validation

Add a Task

Home

Landing Page