# Requirements

A To Do Application involving functions such as

* Add new task
* Delete new task
* Update an existing task
* Delete an existing task

# Design

The spring boot To Do App project will have

* Thymeleaf with bootstrap.
* MongoDb as backend.

Analyzing the requirements, the app contains the basic CRUD operations as functionalities such as the Create, Read, Update and Delete.

Web

* A home page that shows all the tasks along with options to delete and update them as well.
* An insertion form with which the user can insert new tasks with fields of Name, Description and Date.
* An updation form to update the entries.

Spring Boot Structure



# Implementation

Model Class:

It is essentially the entity that the system stores in the database.



Repository



Service Layer



Methods:

* addNewTasks()

Adds a new task on the database with the current timestamp as the primary key.

* getAllTasks()

Returns a list of all the tasks that are of the type ‘Task’.

* getTaskById(int id)

Returns a task that is of the given id.

* deleteTask(int id)

Deletes a task from the database with the given id.

* saveUpdatedTask(Task existing,Task task)

Saves an existing task with the attributes of the new task object.

* getSameDayTask()

Returns a list of tasks that are of type ‘Task’ with the date attribute set to the current date.

Controller

* @RequestMapping(“/index”)

Handles the request to the index page. The controller adds all the existing task as well as task for the current day to display.

* @GetMapping(“/addNew”)

Handles the request to the add task form. Wraps a entity object to the Model and redirects to the form page.

* @PostMapping(“/addTask”)

Makes use of the service layer to add new task data to the database.

* GetMapping(“/delete/{id}”)

Deletes the task entry with the given id using the service layer.

* GetMapping(“/update/{id}”)

Handles the request to the update form. The handler finds the task with the given id, wraps it into the model and redirects to the updation form.

* GetMapping(“/change/{id}”)

This handler changes the actual value of the existing variable using methods defined in the service layer.

HTTP Requests







