

TECHNICAL DESIGN DOCUMENT

|  |  |
| --- | --- |
| **Author** | William Watlington |
| **Last Updated** | 1/29/2023 7:19 PM |
| **Version Number** | 3.0.1 |

**Version History**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Date | Reason for Change | Version |
| *Krasso, Richard* | *10/17/2019* | *New document format* | *3.0.0* |
| Watlington, William | 01/05/2023 | Personas/Prototype | 3.0.1 |
| Watlington, William | 01/23/2023 | Sprint 2 test documents | 3.0.2 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[SECTION 1: INTRODUCTION 3](#_Toc32226123)

[1.1. Purpose 3](#_Toc32226124)

[1.2. Terminology 3](#_Toc32226125)

[1.3. User Personas 3](#_Toc32226126)

[1.4. User Stories 3](#_Toc32226127)

[1.5. Time Estimations 3](#_Toc32226128)

[SECTION 2: PROCESS DESIGN 4](#_Toc32226129)

[2.1. Prototypes 4](#_Toc32226130)

[2.2. ORD (Object Relational Diagram(s) 4](#_Toc32226131)

[2.3. NoSQL Document Diagram 4](#_Toc32226132)

[2.4. NoSQL Data Structure 4](#_Toc32226133)

[SECTION 3: QA TESTING 4](#_Toc32226134)

[3.1. QA Test Plan 4](#_Toc32226135)

[3.2. Unit Tests 4](#_Toc32226136)

[SECTION 4: REFERENCES 5](#_Toc32226137)

# SECTION 1: INTRODUCTION

## Purpose

nodebucket is task management software designed to simplify the way you create, track, edit, and delete time sensitive material. This is a full-stack MEAN (MongoDB, Express, Angular, and Node.js) application, utilizing the concepts that were taught in previous courses. MongoDB will be used for data persistence. Node.js will be used for manipulating and returning saved records, SoapUI for unit testing, and Angular for user interactions.

## Terminology

|  |  |
| --- | --- |
| **Name** | **Comments** |
| MEAN Stack | MongoDB, Express, Angular, and Node.js |
| VS Code | Development IDE |
| SoapUI | API testing tool |
| MongoDB Compass | Database development IDE |
| Angular Material | <https://material.angular.io/> |
| Flex-layout | <https://github.com/angular/flex-layout> |

## User Personas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Age** | **Gender** | **Job Title** | **Marital Status** |
| Kerry Williams | 61 | Female | Retired | Widow |
| James Bond | 47 | Male | Detective | Single |
| Shallan Davar | 18 | Female | Student | Single |

**Bio:**

Kerry Williams

* Kerry Williams is a retired widow who mainly learns of new technology from her children and grandchildren. She has minimal digital literacy and is impatient with complicated applications.

James Bond

* James Bond is a detective who often travels to exotic locations. He expects applications to be simple and effective and is willing to pay for the best in everything he uses.

Shallan Davar

* Shallan Davar is a university student. She has used modern technology such as the internet and mobile phones for her entire life and is comfortable using any application. She expects applications to have little latency and feature modern styling.

## User Stories

## Kerry Williams

* + As a user, I want to track my bill due dates, so I do not forget them.
  + As a user, I want a larger text font, so that the text is easy to read.
  + As a user, I want to be able to set reminders, so that I do not forget to pay my bills.
  + As a user, I want reoccurring payments, so that I can pay my bills online.
  + As a user, I want to color code tasks, so that I can organize tasks by paid and unpaid.
* James Bond
  + As a user, I want to keep a list of tasks, so that I do not forget them.
  + As a user, I want a simple layout, so that the application is easy to use.
  + As a user, I want notifications/alerts for upcoming tasks, so that I am reminded if I do not check the app directly.
  + As a user, I want to be able to delete items, so that I can remove completed items.
  + As a user, I want to be able to view completed tasks, so that I can see what I have completed.
* Shallan Davar
  + As a user, I want to be able to reorganize tasks, so that I can put the most important ones on top.
  + As a user, I want notifications, so that I know what tasks I am working on.
  + As a user, I want to drag and drop tasks, so that I can use the app on my phone.
  + As a user, I want to be able to set task priorities, so that I know which tasks to complete first.
  + As a user, I want to be able to log out of the application, so I do not keep getting notifications.

## Time Estimations

**Points Legend**

Most difficult – 5

Moderately difficult – 3

Least difficult - 1

|  |  |
| --- | --- |
| Story | Points |
| As a user, I want to track my bill due dates, so I do not forget them. | 3 |
| As a user, I want a larger text font, so that the text is easy to read. | 1 |
| As a user, I want to be able to set reminders, so that I do not forget to pay my bills. | 5 |
| As a user, I want reoccurring payments, so that I can pay my bills online. | 5 |
| As a user, I want to color code tasks, so that I can organize tasks by paid and unpaid. | 1 |

|  |  |
| --- | --- |
| Story | Points |
| As a user, I want to keep a list of tasks, so that I do not forget them. | 1 |
| As a user, I want a simple layout, so that the application is easy to use. | 1 |
| As a user, I want notifications/alerts for upcoming tasks, so that I am reminded if I do not check the app directly. | 5 |
| As a user, I want to be able to delete items, so that I can remove completed items. | 3 |
| As a user, I want to be able to view completed tasks, so that I can see what I have completed. | 1 |

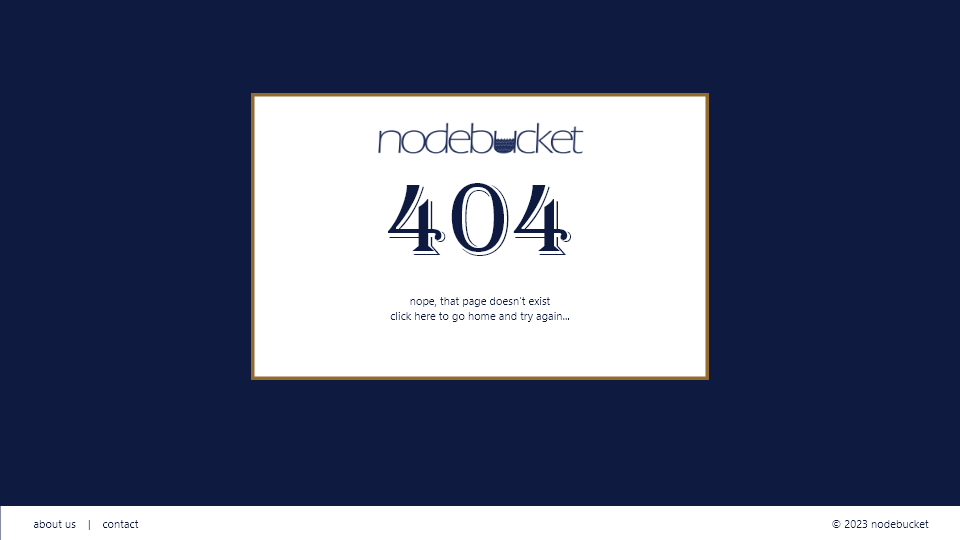
|  |  |
| --- | --- |
| Story | Points |
| As a user, I want to be able to reorganize tasks, so that I can put the most important ones on top. | 5 |
| As a user, I want notifications, so that I know what tasks I am working on. | 5 |
| As a user, I want to drag and drop tasks, so that I can use the app on my phone. | 5 |
| As a user, I want to be able to set task priorities, so that I know which tasks to complete first. | 3 |
| As a user, I want to be able to log out of the application, so I do not keep getting notifications. | 3 |

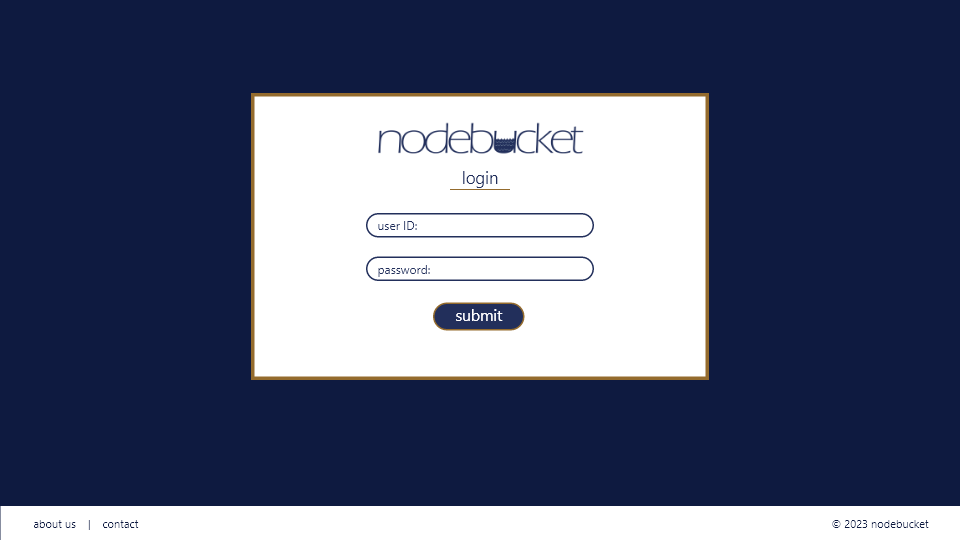
# SECTION 2: PROCESS DESIGN

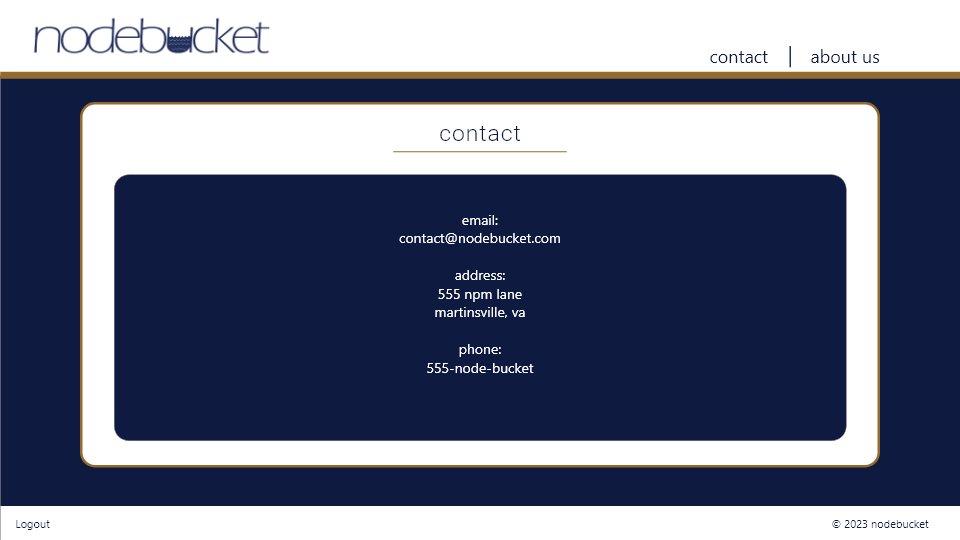
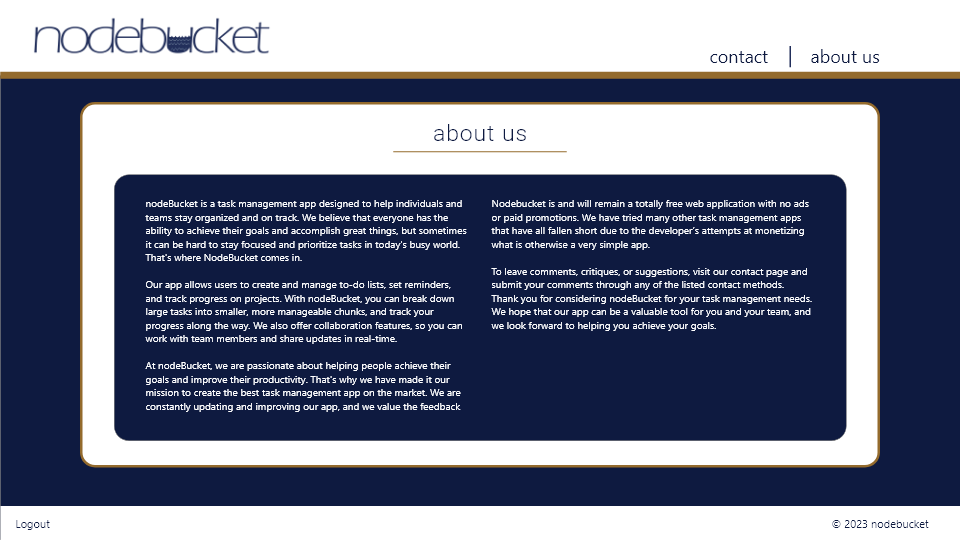
## 2.1. Prototypes

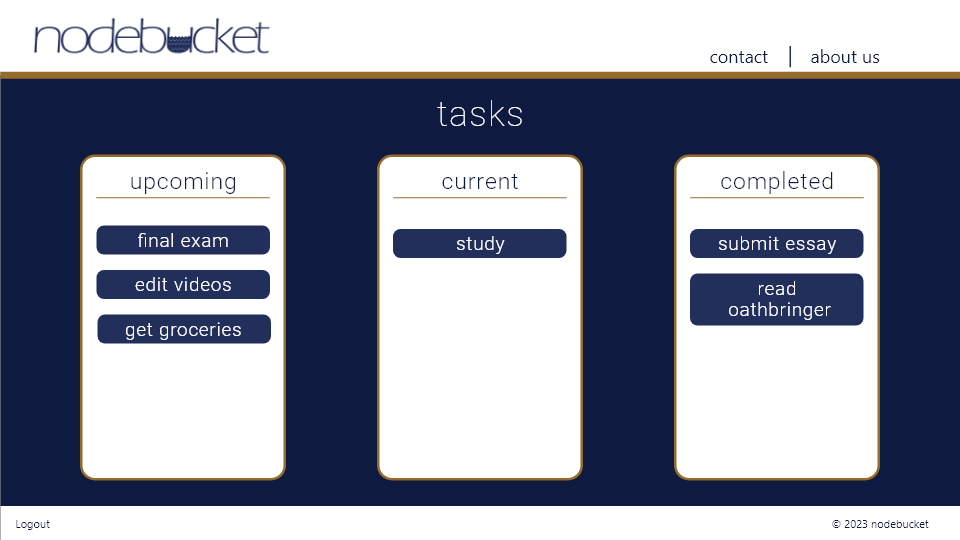
Link to interactive prototype: <https://xd.adobe.com/view/e524f0d7-5193-4fe1-a2e5-7d559775556f-b077/?fullscreen&hints=off>

Prototype images:

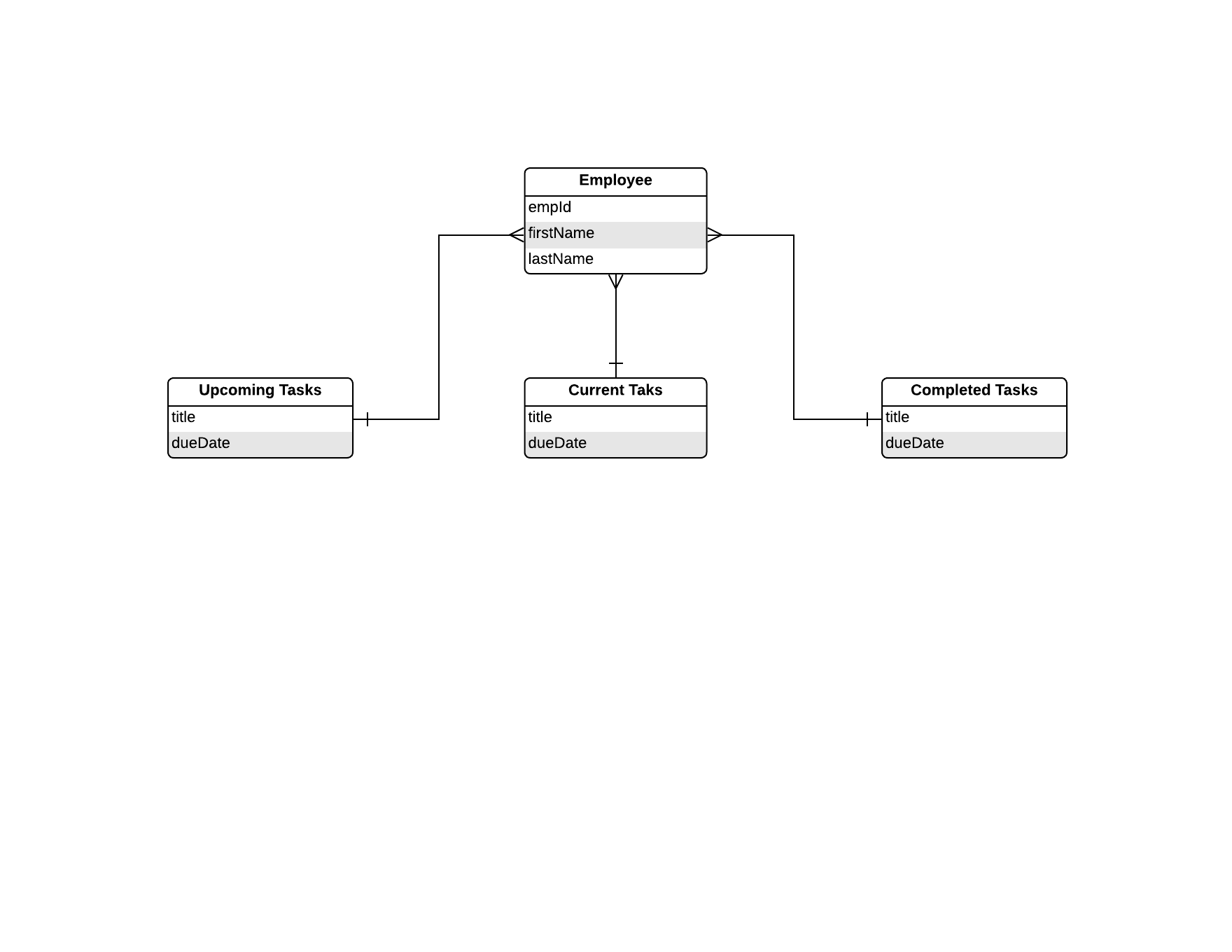








## 2.2. ORD (Object Relational Diagram(s)



## 2.3. NoSQL Data Structure

{

    "empId": 1010,

    "firstName": "Sookie",

    "lastName": "Suzuki",

    "upcomingTasks": [

        {

            "title": "task1",

            "dueDate": 20220130

        }

    ],

    "currentTasks": [

        {

            "title": "task2",

            "dueDate": 20220110

        }

    ],

    "completedTasks": [

        {

            "title": "task3",

            "dueDate": 20220103

        }

    ]

}

# SECTION 3: QA TESTING

## 3.1. QA Test Plan

**Project Name:** nodebucket.fictionfreeway.io

|  |  |
| --- | --- |
| **Test Executed by**: William Watlington | **Test Execution date:** 01/14/2023 |
| **Description:** Login Tests (Sprint 1) |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Test Steps** | **Expected** | **Actual Results** | **(Pass/Fail)** | **Notes** |
| 1 | Navigate to localhost:4200 | Login Page | Login Page | Pass |  |
| 2 | Enter valid empId and submit | Home Page | Home Page | Pass |  |
| 3 | Navigate to localhost:4200 with login cookies active | Home Page | Home Page | Fail |  |
| 4 | Enter invalid empId at login screen and submit | Error Message | Error Message | Pass |  |

|  |
| --- |
| Comments: Testing the functionality of the login page which uses cookies to verify login status. Attempting to visit localhost:4200 without logging in before causes the login page to show. Valid employee ID input generates correct cookies and takes user to the intended page (home page). Invalid employee ID input returns an error message on the login page. Navigating to localhost:4200 while login cookies are active results in user seeing home page as intended. |

|  |  |
| --- | --- |
| **Test Executed by**: William Watlington | **Test Execution date:** 01/21/2023 |
| **Description:** New task modal form test |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Test Steps** | **Expected** | **Actual Results** | **(Pass/Fail)** | **Notes** |
| 1 | Navigate to localhost:4200 while logged in | Home Page/Tasks View | Home Page/Tasks View | Pass |  |
| 2 | Click “+ new task” button | New task modal | New task modal | Pass |  |
| 3 | Click Submit with empty input fields | Nothing (button disabled) | Nothing (button disabled) | Pass |  |

|  |
| --- |
| Comments: Test attempting to create new task without inputting required task fields. Modal appears as expect when clicking “new task” button. Submit button works as expected and is disabled while input fields are empty. |

|  |  |
| --- | --- |
| **Test Executed by**: William Watlington | **Test Execution date:** 01/21/2023 |
| **Description:** Create task |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Test Steps** | **Expected** | **Actual Results** | **(Pass/Fail)** | **Notes** |
| 1 | Navigate to localhost:4200 while logged in | Home Page/Tasks View | Home Page/Tasks View | Pass |  |
| 2 | Click “+ new task” button | New task modal | New task modal | Pass |  |
| 3 | Click Submit with valid input fields | Modal close/show new task in upcoming div | Modal close/show new task in upcoming div | Pass |  |

|  |
| --- |
| Comments: Test creating new task via the new task form. When form fields are filled out, submit button is active. When submit button is clicked, the new task is successfully added to the database and appears in the correct column on the home page. Modal closes on submit. |

|  |  |
| --- | --- |
| **Test Executed by**: William Watlington | **Test Execution date:** 01/21/2023 |
| **Description:** Close new task modal |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Test Steps** | **Expected** | **Actual Results** | **(Pass/Fail)** | **Notes** |
| 1 | Navigate to localhost:4200 while logged in | Home Page/Tasks View | Home Page/Tasks View | Pass |  |
| 2 | Click “+ new task” button | New task modal | New task modal | Pass |  |
| 3 | Click “close” in top right corner of modal | Modal closes without submitting and shows home page as originally seen again | Modal closes without submitting and shows home page as originally seen again | Pass |  |

|  |
| --- |
| Comments: Testing the close function of the new task modal. Clicking the close button successfully changes the visibility value of the modal to “hidden”. |

|  |  |
| --- | --- |
| **Test Executed by**: William Watlington | **Test Execution date:** 01/21/2023 |
| **Description:** Test 404 Page |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Test Steps** | **Expected** | **Actual Results** | **(Pass/Fail)** | **Notes** |
| 1 | Navigate to nonexistent route (localhost:4200/asdf) | 404 page displayed | 404 page displayed | Pass |  |

|  |
| --- |
| Comments: Testing the routing of the 404/not found page. Navigating to routes not associated with other components correctly shows the 404 page |

|  |  |
| --- | --- |
| **Test Executed by**: William Watlington | **Test Execution date:** 01/21/2023 |
| **Description:** Test drag/drop |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Test Steps** | **Expected** | **Actual Results** | **(Pass/Fail)** | **Notes** |
| 1 | Login with valid empID | Home Page | Home Page | Pass |  |
| 2 | Attempt to click and drag an item from todo list to done list | Item moves to done list | Item moves to done list | Pass |  |
| 3 | Reload page | Items are still in new arrangement | Items are still in new arrangement | Pass |  |

|  |
| --- |
| Comments: Testing the drag/drop functionality of the todo/done lists. Items on either list can be dragged from one list to another or to the same list in a different position. Database is updated when this is performed. |

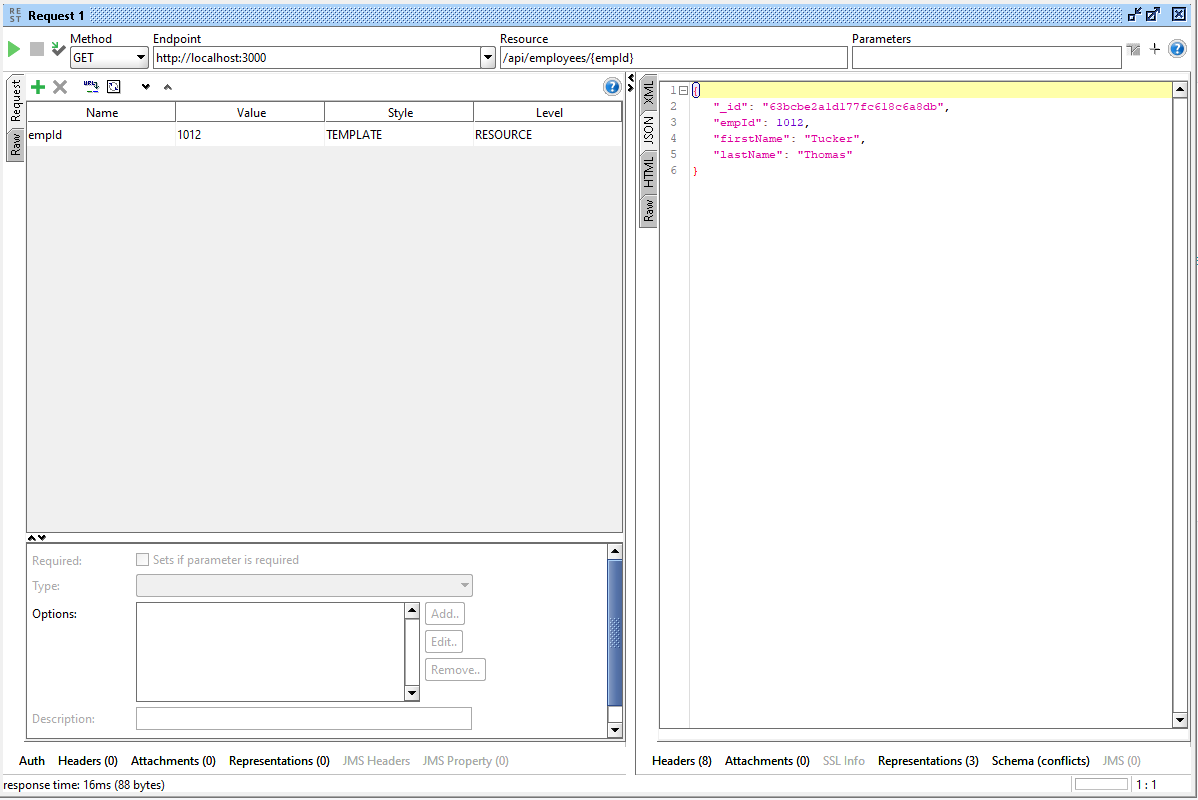
|  |  |
| --- | --- |
| **Test Executed by**: William Watlington | **Test Execution date:** 01/21/2023 |
| **Description:** Test Task Deletion |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Test Steps** | **Expected** | **Actual Results** | **(Pass/Fail)** | **Notes** |
| 1 | Navigate to localhost:4200 | Home Page | Home Page | Pass |  |
| 2 | Click trash icon next to a task item | Task Deletion confirmation modal | Task deletion confirmation modal | Pass |  |
| 3 | Click “delete” | Item deleted | Item Deleted | Pass |  |

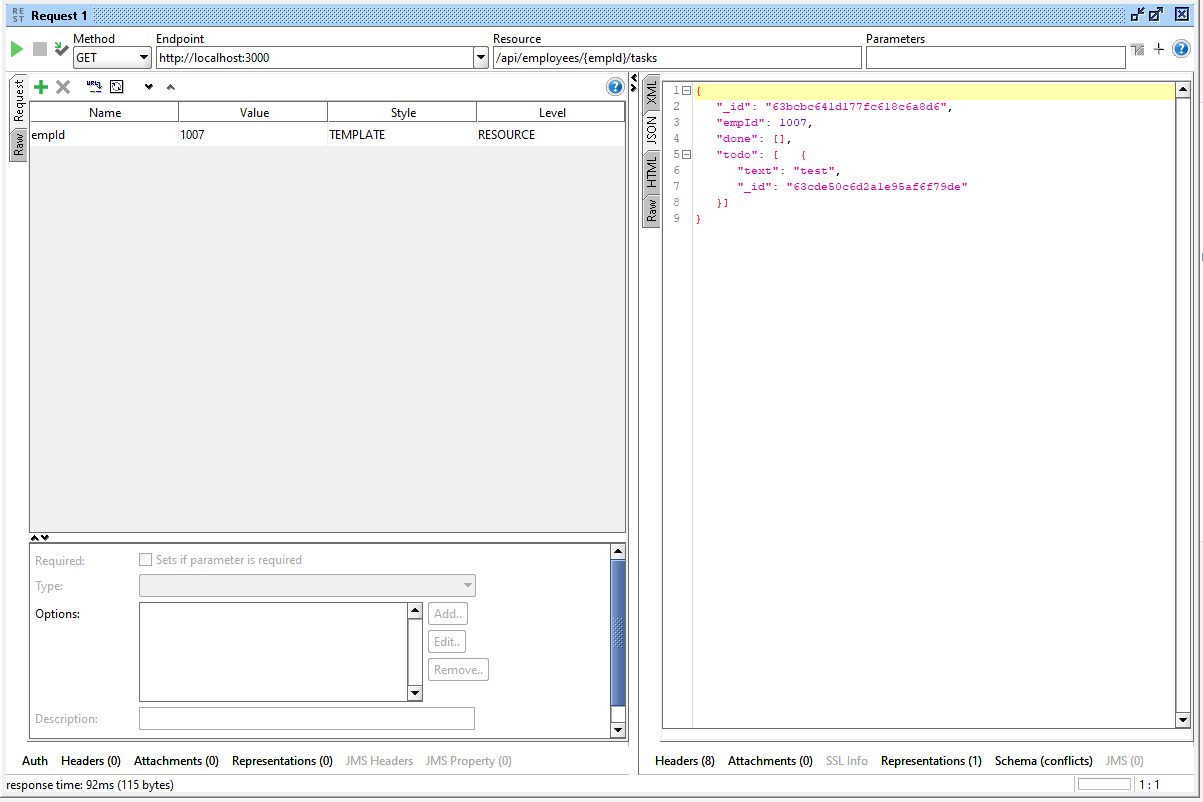
|  |
| --- |
| Comments: Testing the task deletion function and task deletion confirmation modal. Both work as intended without error. |

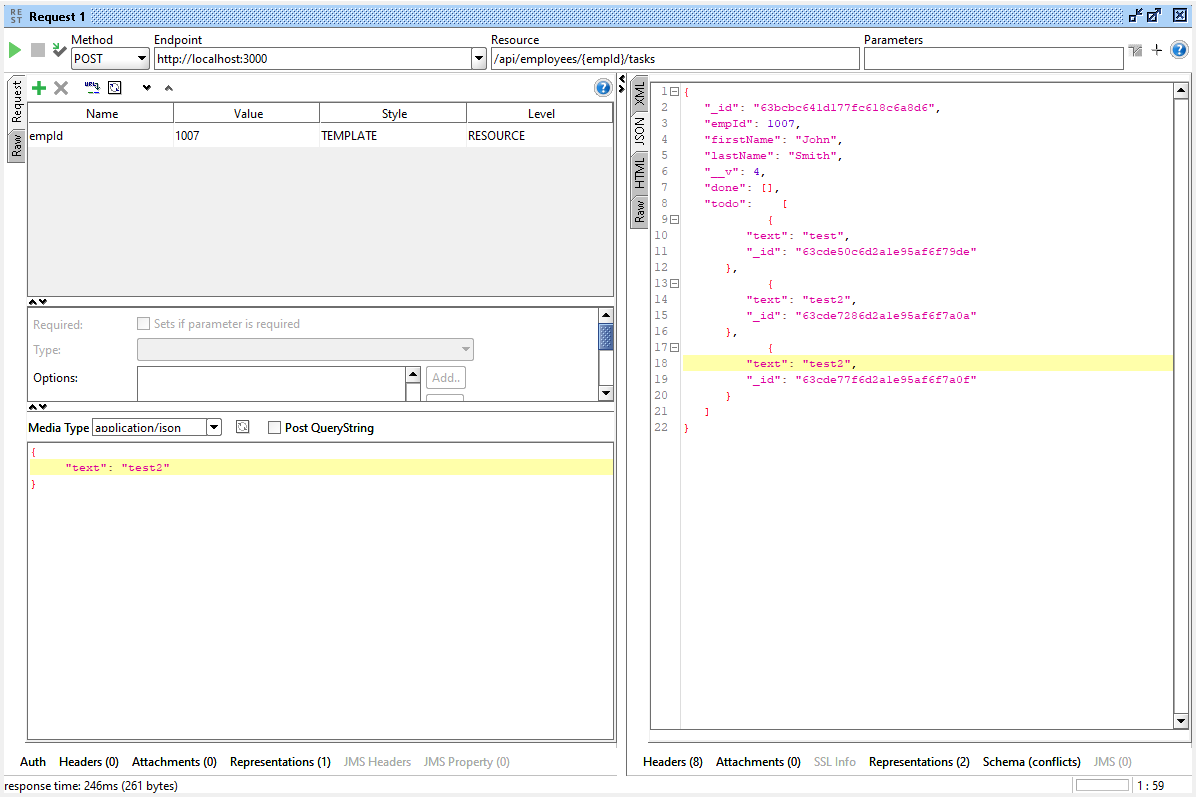
## 3.2. Unit Tests

findEmployeeById

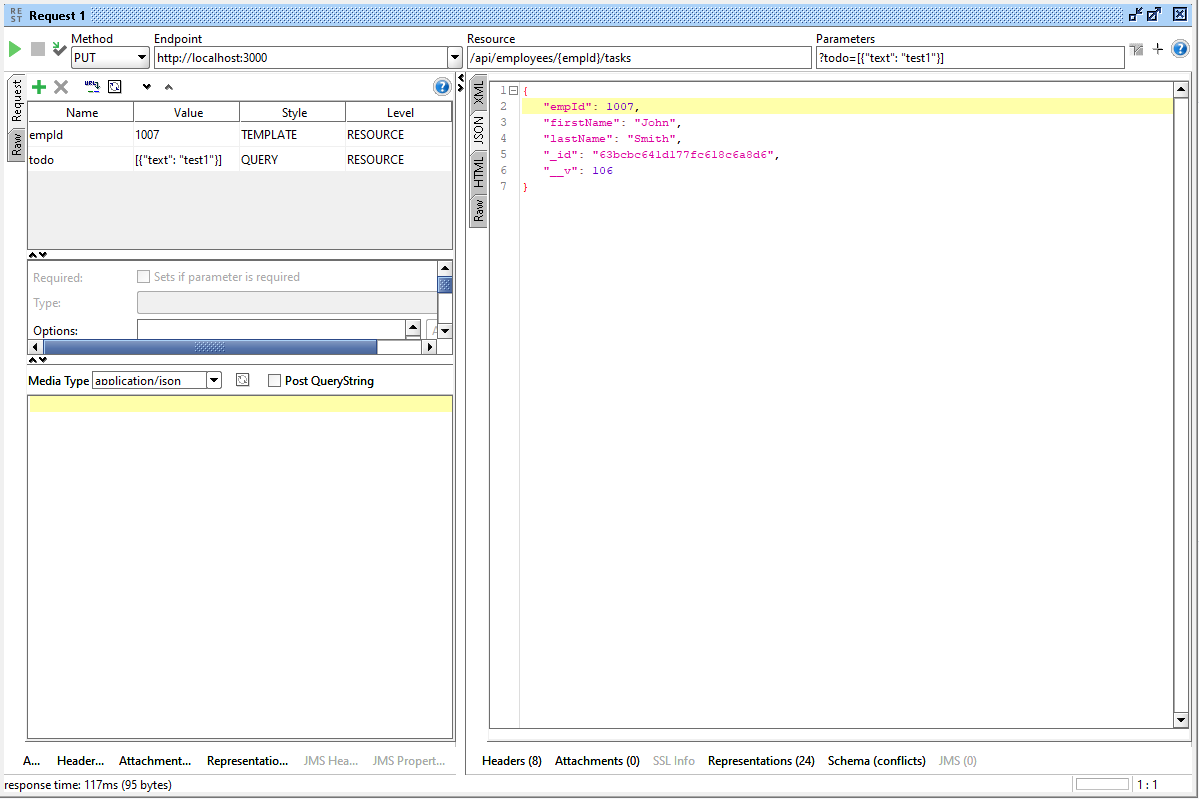


findAllTasks

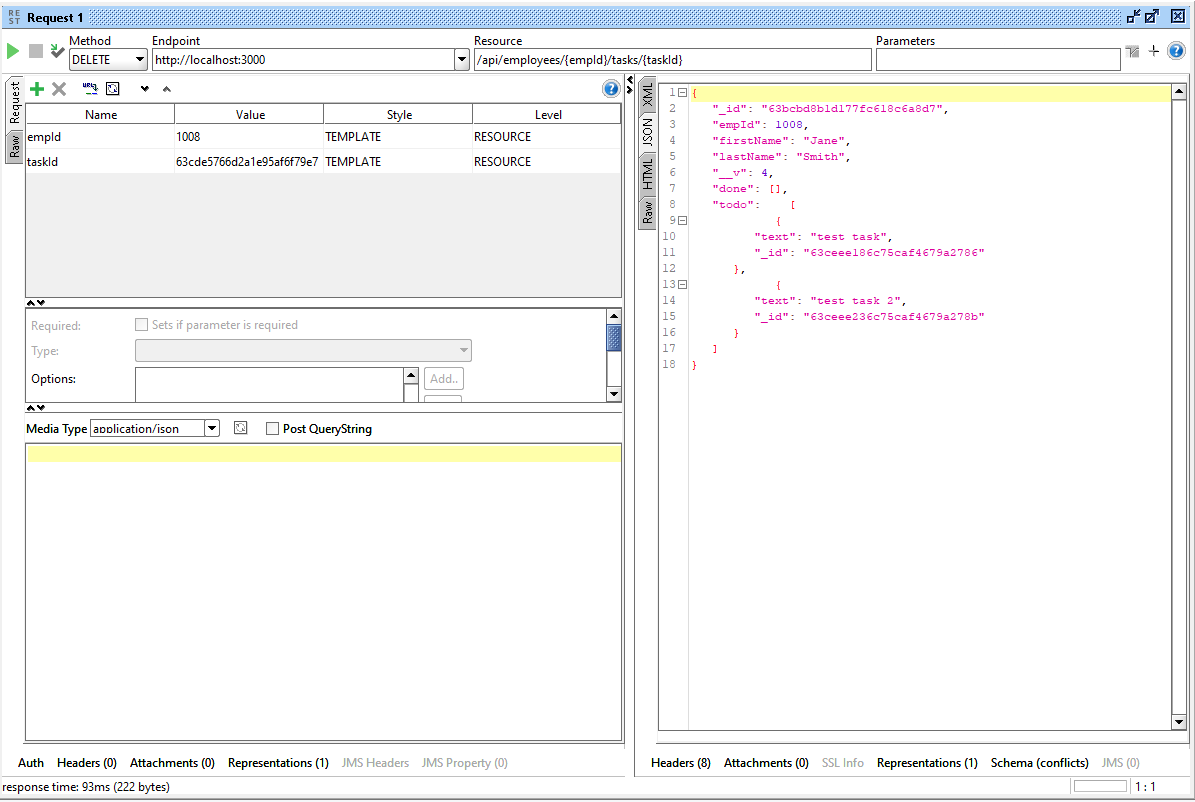


createTask

updateTask



deleteTask



# SECTION 4: REFERENCES

*[APA Reference List]*