

---

# System Requirements Specification Index

For

## Donation Management System

Version 1.0

IIHT Pvt. Ltd.  
fullstack@iiht.com

## TABLE OF CONTENTS

|     |  |    |
|-----|--|----|
| 1   | Project Abstract                               | 3  |
| 2   | Assumptions, Dependencies, Risks / Constraints | 4  |
| 2.1 | NGO Constraints:                               | 4  |
| 2.2 | Donor Constraints                              | 4  |
| 2.3 | Donations Constraints                          | 4  |
| 2.4 | Donation Request Constraints                   | 4  |
| 3   | Business Validations                           | 5  |
| 4   | Rest Endpoints                                 | 6  |
| 4.1 | NgoController                                  | 6  |
| 4.2 | DonarController                                | 6  |
| 5   | Template Code Structure                        | 8  |
| 5.1 | Package: com.iiht.training.ngo                 | 8  |
| 5.2 | Package: com.iiht.training.ngo.entity          | 8  |
| 5.3 | Package: com.iiht.training.ngo.dto             | 10 |
| 5.4 | Package: com.iiht.training.ngo.model.exception | 11 |
| 5.5 | Package: com.iiht.training.ngo.repository      | 11 |
| 5.6 | Package: com.iiht.training.ngo.service         | 12 |
| 5.7 | Package: com.iiht.training.ngo.service.impl    | 13 |
| 5.8 | Package: com.iiht.training.ngo.exception       | 14 |
| 5.9 | Package com.iiht.training.ngo.controller       | 15 |
| 6   | Considerations                                 | 16 |
| 7   | Execution Steps to Follow                      | 17 |

# Donation Management APPLICATION

## System Requirements Specification

---

### 1 PROJECT ABSTRACT

---

**Donation Management** Application is Spring boot RESTful application with MySQL, where NGOs can raise the funds by inviting the donors online and sending the notification about the events and donations.

**Following is the requirement specifications:**

|                                 |  |
|---------------------------------|--|
|                                 | Donation Management System Application     |
|                                 |  |
| Modules                         |  |
| 1                               | NGO  |
| 2                               | Donor                                      |
| 3                               | Donation                                   |
| 4                               | Donation Request                           |
|                                 |  |
| NGO Module Functionalities      |  |
|                                 |  |
| 1                               | Register a NGO                             |
| 2                               | Update the existing NGO details            |
| 3                               | Get the NGO by Id                          |
| 4                               | Fetch all registered NGOs                  |
| 5                               | Delete an existing NGO                     |
|                                 |  |
| Donor Module Functionalities    |  |
| 1                               | Register a Donor                           |
| 2                               | Update the existing Donor                  |
| 3                               | Get a Donor by Id                          |
| 4                               | Fetch all registered Donors                |
| 5                               | Delete an existing Donor                   |
| 6                               | Fetch all the Donors registered with a NGO |
|                                 |  |
| Donation Module Functionalities |  |
| 1                               | Create a Donation                          |
| 2                               | Update the existing Donation details       |
| 3                               | Get the Donation by Id                     |
| 4                               | Fetch all Donations                        |
| 5                               | Delete an existing Donation                |
| 6                               | Fetch all Donations done by a Donor        |
| 7                               | Fetch all Donations done for a NGO         |
|                                 |  |

|   |  |
|---|--|
| Donation Request Module Functionalities |  |
| 1                                       | Create a Donation Request                    |
| 2                                       | Get the Donation Notification by the NGO     |
| 3                                       | Get all the Donation request sent to a Donor |

## 2 ASSUMPTIONS, DEPENDENCIES, RISKS / CONSTRAINTS

---

### 2.1 NGO CONSTRAINTS:

- While deleting an NGO, if ngold does not exist then the operation should throw a custom exception.
- While fetching the NGO details by id, if ngold does not exist then the operation should throw a custom exception.

### 2.2 DONOR CONSTRAINTS

- While deleting the Donor, if donorld does not exist then the operation should throw a custom exception.
- While fetching the Donor details by id, if donorld does not exist then the operation should throw a custom exception.
- While fetching all the Donor details by NGO id, if ngold does not exist then the operation should throw a custom exception.

### 2.3 DONATIONS CONSTRAINTS

- While deleting the Donation, if donationId does not exist then the operation should throw a custom exception.
- While fetching the Donation details by id, if donationId does not exist then the operation should throw a custom exception.
- While fetching all the Donations done by a donor id, if donorld does not exist then the operation should throw a custom exception.
- While fetching all the Donation details by NGO id, if ngold does not exist then operation should throw custom exception.

### 2.4 DONATION REQUEST CONSTRAINTS

- While fetching the all the Donation request done by NGO, if ngold does not exist then operation should throw custom exception.
- While fetching all the Donation requests sent to a donor, if donorld does not exist then the operation should throw a custom exception.

## Common Constraints

- For all rest endpoints receiving @RequestBody, validation check must be done and must throw custom exception if data is invalid
- All the business validations must be implemented in dto classes only.
- All the database operations must be implemented on entity object only
- Do not change, add, remove any existing methods in service layer
- In Repository interfaces, custom methods can be added as per requirements.
- All RestEndpoint methods and Exception Handlers must return data wrapped in **ResponseEntity**

## 3 BUSINESS VALIDATIONS

---

- NGO name is not null, min 3 and max 100 characters.
- NGO username is not null, min 3 and max 50 characters.
- NGO password is not null, min 3 and max 50 characters.
- NGO address is not null, min 3 and max 100 characters.
- NGO phone number is not null and have min 10 and max 10 digits
- NGO started In is not null, have 'yyyy-mm-dd' format and should be past date
- NGO documents is not null, min 3 and max 100 characters.
- Donor name is not null, min 3 and max 100 characters.
- Donor username is not null, min 3 and max 50 characters.
- Donor password is not null, min 3 and max 50 characters.
- Donor email is not null, min 3 and max 100 characters and should be in email format
- Donor phone number is not null and have min 10 and max 10 digits
- Donor address is not null, min 3 and max 100 characters.
- Donation type is not null, min 3 and max 100 characters.
- Donation amount is not null
- Donation date is not null, have 'yyyy-mm-dd' format and should be future date
- Donation Request amount is not null
- Donation Request status is not null, min 3 and max 100 characters
- Donation request end date is not null, have 'yyyy-mm-dd' format and should be future date

## 4 REST ENDPOINTS

Rest End-points to be exposed in the controller along with method details for the same to be created

### 4.1 NgoCONTROLLER

| URL Exposed   | Purpose                  |        |             |                    |        |                          |   |
|---|--------------------------|--------|-------------|--------------------|--------|--------------------------|---|
| <div>1. /ngos/register-ngo</div> <table><tr><td>Http Method</td><td>POST</td></tr><tr><td>Parameter 1</td><td>NgoDto</td></tr><tr><td>Return</td><td>NgoDto</td></tr></table>   | Http Method              | POST   | Parameter 1 | NgoDto             | Return | NgoDto                   | Register a NGO                            |
| Http Method   | POST                     |        |             |                    |        |                          |   |
| Parameter 1   | NgoDto                   |        |             |                    |        |                          |   |
| Return  | NgoDto                   |        |             |                    |        |                          |   |
| <div>2. /ngos/update-ngo</div> <table><tr><td>Http Method</td><td>PUT</td></tr><tr><td>Parameter 1</td><td>NgoDto</td></tr><tr><td>Return</td><td>NgoDto</td></tr></table>  | Http Method              | PUT    | Parameter 1 | NgoDto             | Return | NgoDto                   | Update the NGO                            |
| Http Method   | PUT                      |        |             |                    |        |                          |   |
| Parameter 1   | NgoDto                   |        |             |                    |        |                          |   |
| Return  | NgoDto                   |        |             |                    |        |                          |   |
| <div>3. /ngos/get/{ngold}</div> <table><tr><td>Http Method</td><td>GET</td></tr><tr><td>Parameter 1</td><td>Long(ngold)</td></tr><tr><td>Return</td><td>NgoDto</td></tr></table>  | Http Method              | GET    | Parameter 1 | Long(ngold)        | Return | NgoDto                   | Fetches the details of NGO by Id          |
| Http Method   | GET                      |        |             |                    |        |                          |   |
| Parameter 1   | Long(ngold)              |        |             |                    |        |                          |   |
| Return  | NgoDto                   |        |             |                    |        |                          |   |
| <div>4. /ngos/delete/{ngold}</div> <table><tr><td>Http Method</td><td>DELETE</td></tr><tr><td>Parameter 1</td><td>Long (ngold)</td></tr><tr><td>Return</td><td>Boolean</td></tr></table>  | Http Method              | DELETE | Parameter 1 | Long (ngold)       | Return | Boolean                  | Delete the Ngo detail                     |
| Http Method   | DELETE                   |        |             |                    |        |                          |   |
| Parameter 1   | Long (ngold)             |        |             |                    |        |                          |   |
| Return  | Boolean                  |        |             |                    |        |                          |   |
| <div>5. /ngos/all</div> <table><tr><td>Http Method</td><td>GET</td></tr><tr><td>Parameter 1</td><td>-</td></tr><tr><td>Return</td><td>List&lt;NgoDto&gt;</td></tr></table>  | Http Method              | GET    | Parameter 1 | -                  | Return | List<NgoDto>             | Fetch all registered NGOs                 |
| Http Method   | GET                      |        |             |                    |        |                          |   |
| Parameter 1   | -                        |        |             |                    |        |                          |   |
| Return  | List<NgoDto>             |        |             |                    |        |                          |   |
| <div>6. /ngos/create-donation-request</div> <table><tr><td>Http Method</td><td>POST</td></tr><tr><td>Parameter 1</td><td>DonationRequestDto</td></tr><tr><td>Return</td><td>DonationRequestDto</td></tr></table>                    | Http Method              | POST   | Parameter 1 | DonationRequestDto | Return | DonationRequestDto       | Create a Donation request                 |
| Http Method   | POST                     |        |             |                    |        |                          |   |
| Parameter 1   | DonationRequestDto       |        |             |                    |        |                          |   |
| Return  | DonationRequestDto       |        |             |                    |        |                          |   |
| <div>7. /ngos/donation-request-by-ngo/{ngold}</div> <table><tr><td>Http Method</td><td>GET</td></tr><tr><td>Parameter 1</td><td>Long(ngold)</td></tr><tr><td>Return</td><td>List&lt;DonationRequestDto&gt;</td></tr></table>        | Http Method              | GET    | Parameter 1 | Long(ngold)        | Return | List<DonationRequestDto> | Get all the donation request by the NGO   |
| Http Method   | GET                      |        |             |                    |        |                          |   |
| Parameter 1   | Long(ngold)              |        |             |                    |        |                          |   |
| Return  | List<DonationRequestDto> |        |             |                    |        |                          |   |
| <div>8. /ngos/ donation-request-by-donar/{donarId}</div> <table><tr><td>Http Method</td><td>GET</td></tr><tr><td>Parameter 1</td><td>Long(donarId)</td></tr><tr><td>Return</td><td>List&lt;DonationRequestDto&gt;</td></tr></table> | Http Method              | GET    | Parameter 1 | Long(donarId)      | Return | List<DonationRequestDto> | Get all the donation requests for a donar |
| Http Method   | GET                      |        |             |                    |        |                          |   |
| Parameter 1   | Long(donarId)            |        |             |                    |        |                          |   |
| Return  | List<DonationRequestDto> |        |             |                    |        |                          |   |

|  |  |
|--|--|
|  |  |
|--|--|

## 4.2 DONARCONTROLLER

| URL Exposed                   |                | Purpose  |
|-------------------------------|----------------|--|
| 1. /donars/register-donar     |                | Register a Donar                               |
| Http Method                   | POST           |  |
| Parameter 1                   | DonarDto       |  |
| Return                        | DonarDto       |  |
| 2. /donars/update-donar       |                | Update the existing donar                      |
| Http Method                   | PUT            |  |
| Parameter 1                   | DonarDto       |  |
| Return                        | DonarDto       |  |
| 3. /donars/get/{donarId}      |                | Fetches the donar details by id                |
| Http Method                   | GET            |  |
| Parameter 1                   | Long(donarId)  |  |
| Return                        | DonarDto       |  |
| 4. /donars/all                |                | Fetch the details of all the registered donars |
| Http Method                   | GET            |  |
| Parameter 1                   | -              |  |
| Return                        | List<DonarDto> |  |
| 5. /donars/delete/{donarId}   |                | Delete the existing Donar                      |
| Http Method                   | DELETE         |  |
| Parameter 1                   | Long (donarId) |  |
| Return                        | Boolean        |  |
| 6. /donars/get-by-ngo/{ngold} |                | Fetch all the donars registered with the NGO   |
| Http Method                   | GET            |  |
| Parameter 1                   | Long (ngold)   |  |
| Return                        | List<DonarDto> |  |
| 7. /donations/add-donation    |                | Create a Donation                              |
| Http Method                   | POST           |  |
| Parameter 1                   | DonationDto    |  |
| Return                        | DonationDto    |  |
| 8. /donations/update-donation |                | Update the existing Donation details           |
| Http Method                   | PUT            |  |
| Parameter 1                   | DonationDto    |  |
| Return                        | DonationDto    |  |

|                                       |                   |  |
|---------------------------------------|-------------------|--|
| 9. /donations/delete/{donationId}     |                   | Delete an existing Donation                        |
| Http Method                           | DELETE            |  |
| Parameter 1                           | Long (donationId) |  |
| Return                                | Boolean           |  |
| 10. /donations/get/{donationId}       |                   | Get the donation details by id                     |
| Http Method                           | GET               |  |
| Parameter 1                           | Long (donationId) |  |
| Return                                | DonationDto       |  |
| 11. /donations/all                    |                   | Fetch all the existing donations                   |
| Http Method                           | GET               |  |
| Parameter 1                           | -                 |  |
| Return                                | List<DonationDto> |  |
| 12. /donations/get-by-donar/{donarId} |                   | Fetch all the donations for a particular donar     |
| Http Method                           | GET               |  |
| Parameter 1                           | Long(donarId)     |  |
| Return                                | List<DonationDto> |  |
| 13. /donations/get-by-ngo/{ngold}     |                   | Fetch all the donations raised by a particular NGO |
| Http Method                           | GET               |  |
| Parameter 1                           | Long(ngold)       |  |
| Return                                | List<DonationDto> |  |
| 14.                                   |                   |  |

## 5 TEMPLATE CODE STRUCTURE

### 5.1 PACKAGE: COM.IIHT.TRAINING.NGO

#### Resources

|  |   |                     |
|--|---|---------------------|
| <b>DonationManagementSystemApplication (Class)</b> | This is the Spring Boot starter class of the application. | Already Implemented |
|--|---|---------------------|

### 5.2 PACKAGE: COM.IIHT.TRAINING.NGO.ENTITY

#### Resources

| Class/Interface          | Description                                     | Status                 |
|--------------------------|---|------------------------|
| <b>NgoEntity (class)</b> | o Annotate this class with proper annotation to | Partially implemented. |



|                                      |  |                        |
|--------------------------------------|--|------------------------|
|                                      | <p>declare it as an entity class with <b>ngold</b> as primary key.</p> <ul style="list-style-type: none"> <li>o Map this class with the ngo_details table.</li> <li>o Generate the <b>ngold</b> using <b>IDENTITY</b> strategy</li> </ul>  |                        |
| <b>DonarEntity(class)</b>            | <ul style="list-style-type: none"> <li>o This class is partially implemented.</li> <li>o Annotate this class with proper annotation to declare it as an entity class with <b>donarId</b> as primary key.</li> <li>o Map this class with <b>donar</b> table.</li> <li>o Generate the <b>donarId</b> using the <b>IDENTITY</b> strategy</li> </ul>                           | Partially implemented. |
| <b>Donation(class)</b>               | <ul style="list-style-type: none"> <li>o This class is partially implemented.</li> <li>o Annotate this class with proper annotation to declare it as an entity class with <b>donationId</b> as primary key.</li> <li>o Map this class with <b>donation</b> table.</li> <li>o Generate the <b>donationId</b> using the <b>IDENTITY</b> strategy</li> </ul>                  | Partially implemented. |
| <b>DonationRequestEntity (class)</b> | <ul style="list-style-type: none"> <li>o This class is partially implemented.</li> <li>o Annotate this class with proper annotation to declare it as an entity class with <b>requestId</b> as primary key.</li> <li>o Map this class with <b>donation_request</b> table.</li> <li>o Generate the <b>requestId</b> using the <b>IDENTITY</b> strategy</li> <li>o</li> </ul> | Partially implemented  |

### 5.3 PACKAGE: COM.IIHT.TRAINING.NGO.DTO

#### Resources

| Class/Interface                   | Description  | Status                 |
|-----------------------------------|--|------------------------|
| <b>NgoDto (class)</b>             | Use appropriate annotations from the <b>Java Bean Validation API</b> for validating attributes of this class. (Refer <b>Business Validation</b> section for validation rules). | Partially implemented. |
| <b>DonarDto (class)</b>           | Use appropriate annotations from the <b>Java Bean Validation API</b> for validating attributes of this class. (Refer <b>Business Validation</b> section for validation rules). | Partially implemented. |
| <b>DonationDto (class)</b>        | Use appropriate annotations from the <b>Java Bean Validation API</b> for validating attributes of this class. (Refer <b>Business Validation</b> section for validation rules). | Partially implemented. |
| <b>DonationRequestDto (class)</b> | Use appropriate annotations from the <b>Java Bean Validation API</b> for validating attributes of this class. (Refer <b>Business Validation</b> section for validation rules). | Partially implemented  |

## 5.4 PACKAGE: COM.IIHT.TRAINING.NGO.MODEL.EXCEPTION

### Resources

| Class/Interface                  | Description   | Status               |
|----------------------------------|---|----------------------|
| <b>ExceptionResponse (class)</b> | Object of this class is supposed to be returned in case of exception through exception handlers | Already implemented. |

## 5.5 PACKAGE: COM.IIHT.TRAINING.NGO.REPOSITORY

### Resources

| Class/Interface                              | Description   | Status                |
|--|---|-----------------------|
| <b>NgoRepository (interface)</b>             | <ol style="list-style-type: none"><li>1. Repository interface exposing CRUD functionality for <b>NGO</b> Entity.</li><li>2. You can go ahead and add any custom methods as per requirements</li></ol>         | Partially implemented |
| <b>DonarRepository (interface)</b>           | <ol style="list-style-type: none"><li>1. Repository interface exposing CRUD functionality for <b>DonarEntity</b> Entity.</li><li>2. You can go ahead and add any custom methods as per requirements</li></ol> | Partially implemented |
| <b>DonationRepository (interface)</b>        | <ol style="list-style-type: none"><li>1. Repository interface exposing CRUD functionality for <b>Donation</b> Entity.</li><li>2. You can go ahead and add any custom methods as per requirements</li></ol>    | Partially implemented |
| <b>DonationRequestRepository (interface)</b> | <ol style="list-style-type: none"><li>1. Repository interface exposing CRUD functionality for <b>DonationRequest</b> Entity.</li></ol>  | Partially implemented |

|  |  |  |
|--|--|--|
|  | 2. You can go ahead and add any custom methods as per requirements |  |
|--|--|--|

## 5.6 PACKAGE: COM.IIHT.TRAINING.NGO.SERVICE

### Resources

| Class/Interface                           | Description  | Status               |
|---|--|----------------------|
| <b>NgoService (interface)</b>             | Interface to expose method signatures for NGO related functionality.<br><br>Do not modify, add or delete any method              | Already implemented. |
| <b>DonarService (interface)</b>           | Interface to expose method signatures for Donar related functionality.<br><br>Do not modify, add or delete any method            | Already implemented. |
| <b>DonationService (interface)</b>        | Interface to expose method signatures for Donations related functionality.<br><br>Do not modify, add or delete any method        | Already implemented. |
| <b>DonationRequestService (interface)</b> | Interface to expose method signatures for Donation request related functionality.<br><br>Do not modify, add or delete any method | Already implemented  |

## 5.7 PACKAGE: COM.IIHT.TRAINING.NGO.SERVICE.IMPL

### Resources

| Class/Interface             | Description   | Status             |
|-----------------------------|---|--------------------|
| NgoServiceImpl (class)      | <ul style="list-style-type: none"><li>Implements <b>NgoService</b>.<br/>Contains template method implementation.</li><li>Need to provide implementation for NGO related functionalities</li><li>Add required repository dependency</li><li>Do not modify, add or delete any method signature</li></ul>            | To be implemented. |
| DonarServiceImpl (class)    | <ul style="list-style-type: none"><li>Implements <b>DonarService</b>.<br/>Contains template method implementation.</li><li>Need to provide implementation for Donar related functionalities</li><li>Add required repository dependency</li><li>Do not modify, add or delete any method signature</li></ul>        | To be implemented. |
| DonationServiceImpl (class) | <ul style="list-style-type: none"><li>Implements <b>DonationService</b>.<br/>Contains template method implementation.</li><li>Need to provide implementation for donations related functionalities</li><li>Add required repository dependency</li><li>Do not modify, add or delete any method signature</li></ul> | To be implemented. |

|   |  |                   |
|---|--|-------------------|
| <b>DonationRequestServiceImpl (class)</b> | <ul style="list-style-type: none"> <li>Implements <b>DonationRequestService</b>.<br/>Contains template method implementation.</li> <li>Need to provide implementation for donation request and notifications related functionalities</li> <li>Add required repository dependency</li> <li>Do not modify, add or delete any method signature</li> </ul> | To be implemented |
|---|--|-------------------|

## 5.8 PACKAGE: COM.IIHT.TRAINING.NGO.EXCEPTION

### Resources

| Class/Interface              | Description  | Status                 |
|------------------------------|--|------------------------|
| <b>GlobalHandler (class)</b> | <ul style="list-style-type: none"> <li>RestControllerAdvice Class for defining global exception handlers.</li> <li>Contains Exception Handler for <b>InvalidDataException</b> class.</li> <li>Use this as a reference for creating exception handler for other custom exception classes</li> </ul> | Partially implemented. |

| Class/Interface                          | Description   | Status           |
|--|---|------------------|
| <b>NgoNotFoundException (Class)</b>      | <ul style="list-style-type: none"> <li>• Custom Exception to be thrown when trying to fetch or delete the NGO info which does not exist.</li> <li>• Need to create Exception Handler for same wherever needed (local or global)</li> </ul>    | Already created. |
| <b>DonarNotFoundException (Class)</b>    | <ul style="list-style-type: none"> <li>• Custom Exception to be thrown when trying to fetch or delete Donar info which does not exist.</li> <li>• Need to create Exception Handler for same wherever needed (local or global)</li> </ul>      | Already created. |
| <b>DonationNotFoundException (Class)</b> | <ul style="list-style-type: none"> <li>• Custom Exception to be thrown when trying to fetch or delete a donation info which does not exist.</li> <li>• Need to create Exception Handler for same wherever needed (local or global)</li> </ul> | Already created. |

## 5.9 PACKAGE: COM.IIHT.TRAINING.NGO.CONTROLLER

### Resources

| Class/Interface                | Description   | Status            |
|--------------------------------|---|-------------------|
| <b>NgoController (Class)</b>   | <ul style="list-style-type: none"><li>• Controller class to expose all rest-endpoints for NGO and donation request related activities.</li><li>• May also contain local exception handler methods</li></ul> | To be implemented |
| <b>DonarController (Class)</b> | <ul style="list-style-type: none"><li>• Controller class to expose all rest-endpoints for Donar and Donations related activities.</li><li>• May also contain local exception handler methods</li></ul>      | To be implemented |

## 6 CONSIDERATIONS

---

- A. There is no roles in this application
- B. You can perform the following 3 possible actions

|                  |
|------------------|
| NGO              |
| Donor            |
| Donation         |
| Donation Request |



## 7 EXECUTION STEPS TO FOLLOW

---

1. All actions like build, compile, running application, running test cases will be through Command Terminal.
2. To open the command terminal the test takers, need to go to Application menu (Three horizontal lines at left top) -> Terminal -> New Terminal.
3. To build your project and run test cases use command:  
**mvn clean package**
4. To launch your application, move into the target folder (**cd target**). Run the following command to run the application:  
**java -jar donation-management-system-0.0.1-SNAPSHOT.jar**
5. This editor Auto Saves the code.
6. If you want to exit(logout) and continue the coding later anytime (using Save & Exit option on Assessment Landing Page) then you need to use **CTRL+Shift+B**-command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository. Else the code will not be available in the next login.
7. These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.
8. To test any Restful application, the last option on the left panel of IDE, you can find ThunderClient, which is the lightweight equivalent of POSTMAN.
9. This is a web-based application, to run the application on a browser, use the internal browser in the workspace. Click on the second last option on the left panel of IDE, you can find Browser Preview, where you can launch the application.

**Note: The application will not run in the local browser**

10. Default credentials for MySQL:
  - a. Username: **root**
  - b. Password: **pass@word1**
11. To login to mysql instance: Open new terminal and use following command:
  - a. **sudo systemctl enable mysql**
  - b. **sudo systemctl start mysql**
  - c. **mysql -u root -p**

**The last command will ask for password which is 'pass@word1'**
12. Mandatory: Before final submission run the following command:  
**mvn test**

13. You need to use **CTRL+Shift+B** - command compulsorily on code IDE, before final submission as well. This will push or save the updated contents in the internal git/repository, and will be used to evaluate the code quality.