**EGEN CODING CHALLENGE – LEVEL 1**

**Technical Design Document**

Created By: -

Ankur Bag

Contents

**[Egen Coding Challenge – Level 1](#_Toc449968962)** [3](#_Toc449968962)

[**Assumptions** 3](#_Toc449968963)

[**Questions** 3](#_Toc449968964)

[**Information** 4](#_Toc449968965)

[**Steps to Import Project** 4](#_Toc449968966)

[**Use Cases** 10](#_Toc449968967)

[**Home Screen** 10](#_Toc449968968)

[**createUser :** 11](#_Toc449968969)

[**Mongo DB** 14](#_Toc449968970)

[**getAllUsers** 14](#_Toc449968971)

[**updateUser** 16](#_Toc449968972)

[**JUNIT Test CASE:** 26](#_Toc449968973)

# **Egen Coding Challenge – Level 1**

Create a restful web service for User Management application using

* Spark Java <http://sparkjava.com/> and
* MongoDB

## **Assumptions**

* User has the following properties

{    
   **"id"**:"1630215c-2608-44b9-aad4-9d56d8aafd4c",  
   **"firstName"**:"Dorris",  
   **"lastName"**:"Keeling",  
   **"email"**:"Darby\_Leffler68@gmail.com",  
   **"address"**:{    
      **"street"**:"193 Talon Valley",  
      **"city"**:"South Tate furt",  
      **"zip"**:"47069",  
      **"state"**:"IA",  
      **"country"**:"US"  
   },  
   **"dateCreated"**:"2016-03-15T07:02:40.896Z",  
   **"company"**:{    
      **"name"**:"Denesik Group",  
      **"website"**:"http://jodie.org"  
   },  
   **"profilePic"**:"http://lorempixel.com/640/480/people"  
}

* Use MongoDB as your data store.
* The User's ID uniquely identifies a user.
* Feel free to use any framework or library in your code. But make sure you use Spark Java and Mongo.

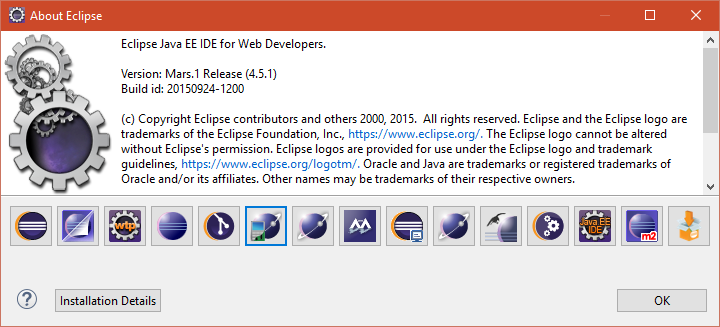
## **Questions**

* Develop the below 4 services using Spark Java
  + **createUser**
    - Takes input as JSON.
    - Creates the user if he is not already available in the data store.
  + **getAllUsers**
    - Gives the list of all users that are in the data store
  + **updateUser**
    - Takes input as JSON
    - Finds the user from the data store
      * if found, updates the required fields
      * if not found, sends back a 404 saying user not found
      * Unit tests using any unit testing library
* Create a new repo egen-be-challenge on your [github.com](http://github.com/) account and commit all your code to it. Make sure all IDE and OS related files are in .gitignore.
* Once completed, send the link of the repo to the recruiter.

## **Information**

## **Steps to Import Project**

**Tools Used: Eclipse**

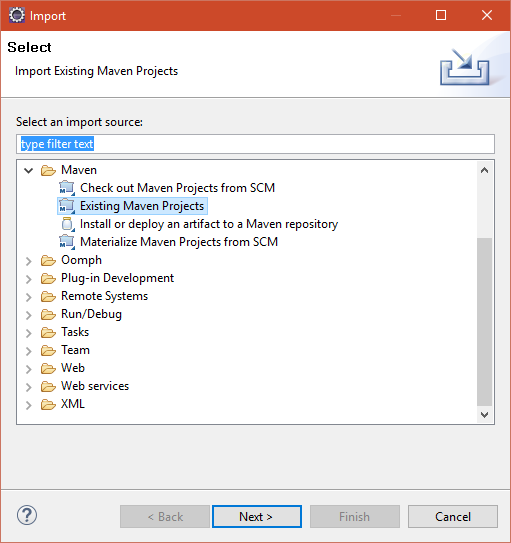


**Developed Maven Java Spark Project:**

1. **Front End :** BootStrap, Velocity
2. **Data Base :** MongoDb

**STEP 1**

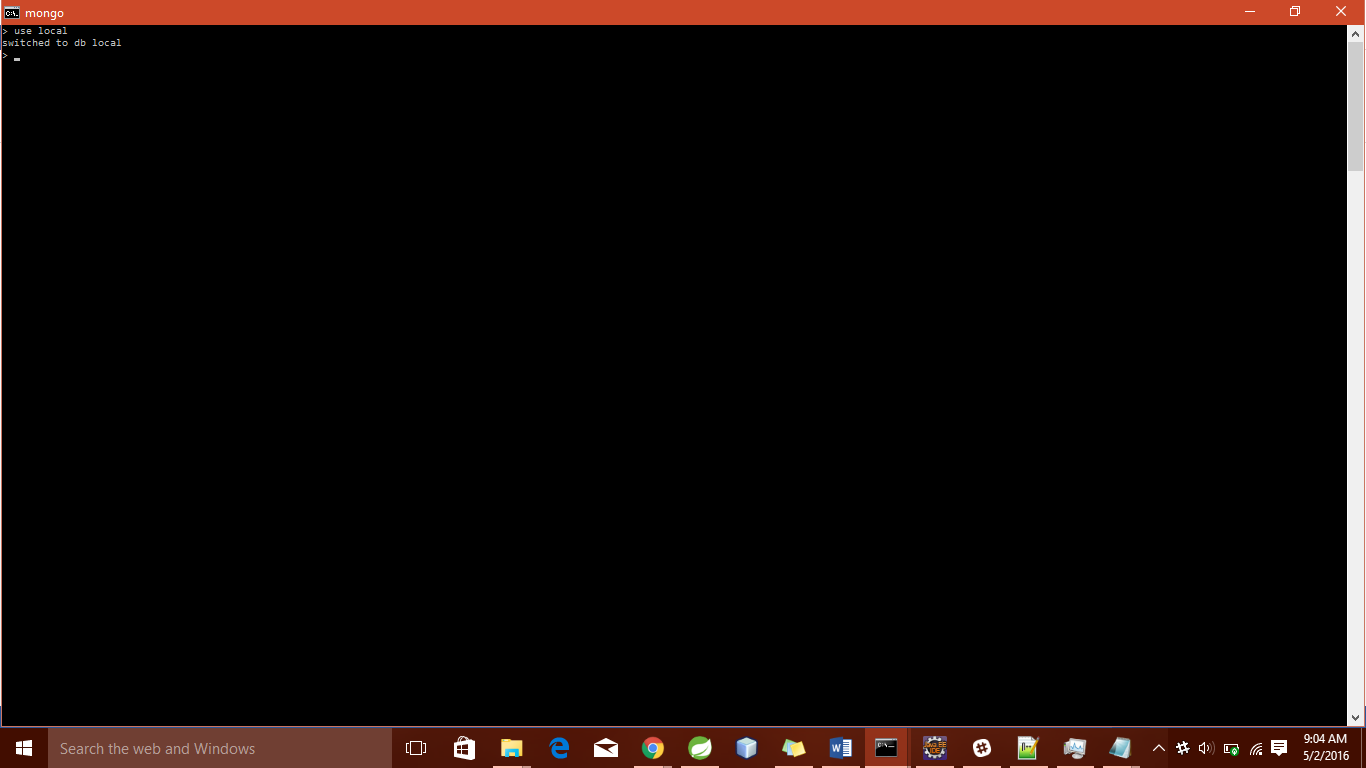
**Import As:**



**STEP 2**

**Install Mongo DB**

Use following Schema

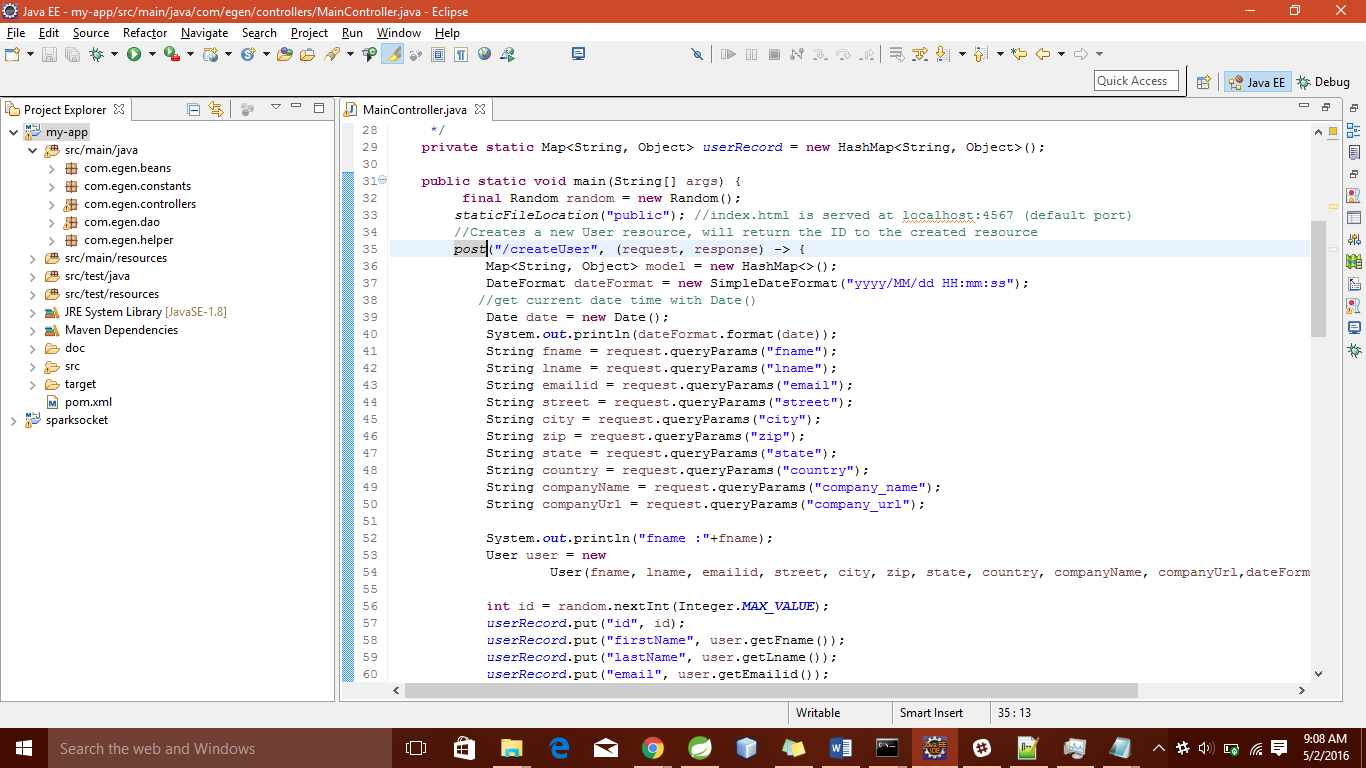


**Create Db : user collection**

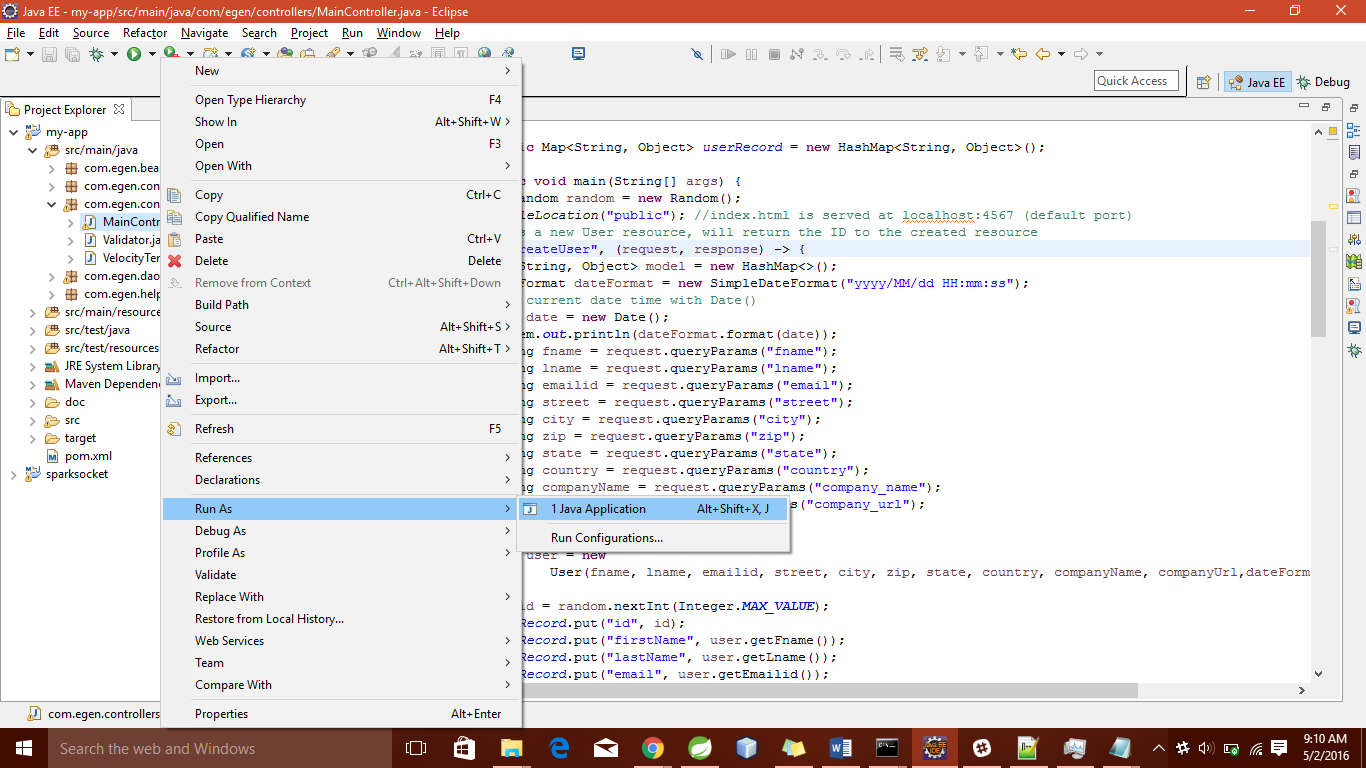
**Syntax :** db.createCollection(“**user\_collection**”)

**STEP 3**

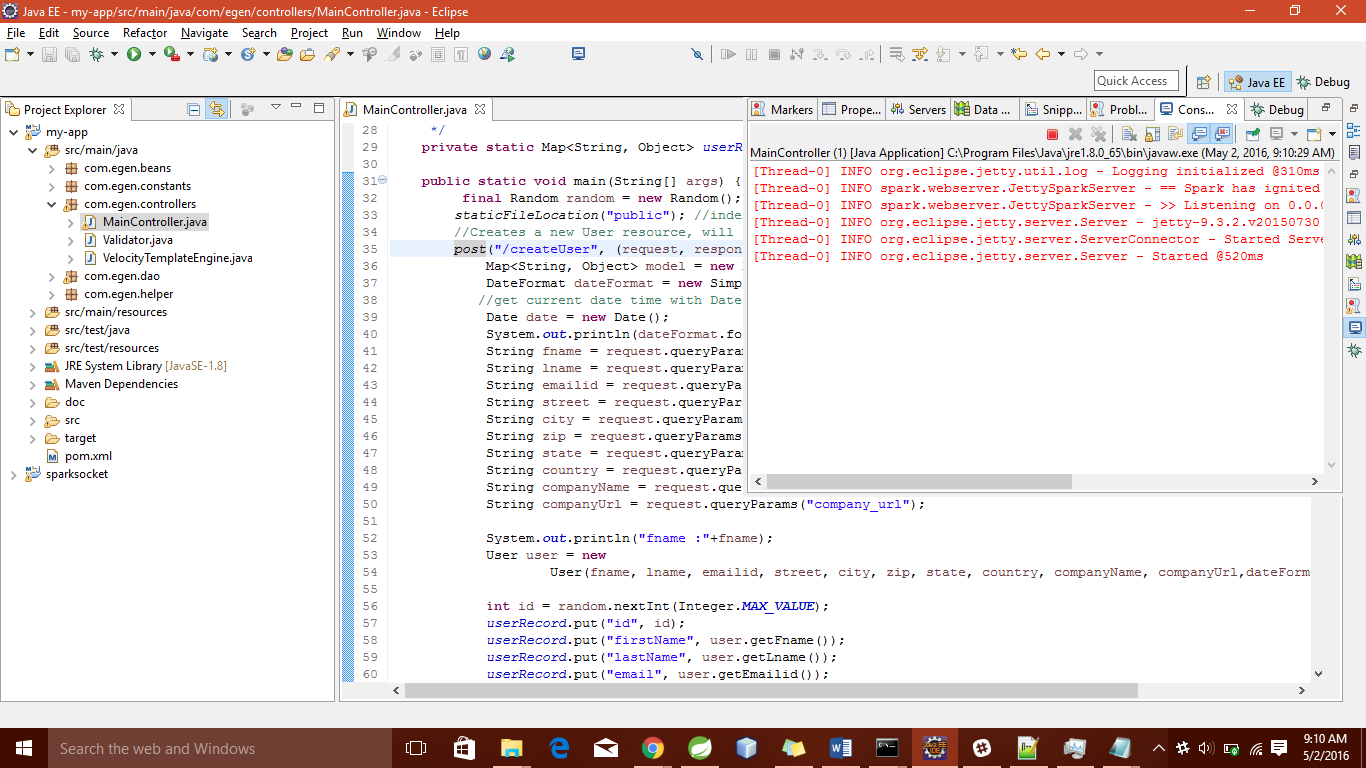
**After Importing the project :**



**Right Click on *MainController.java* to run as Java Application**

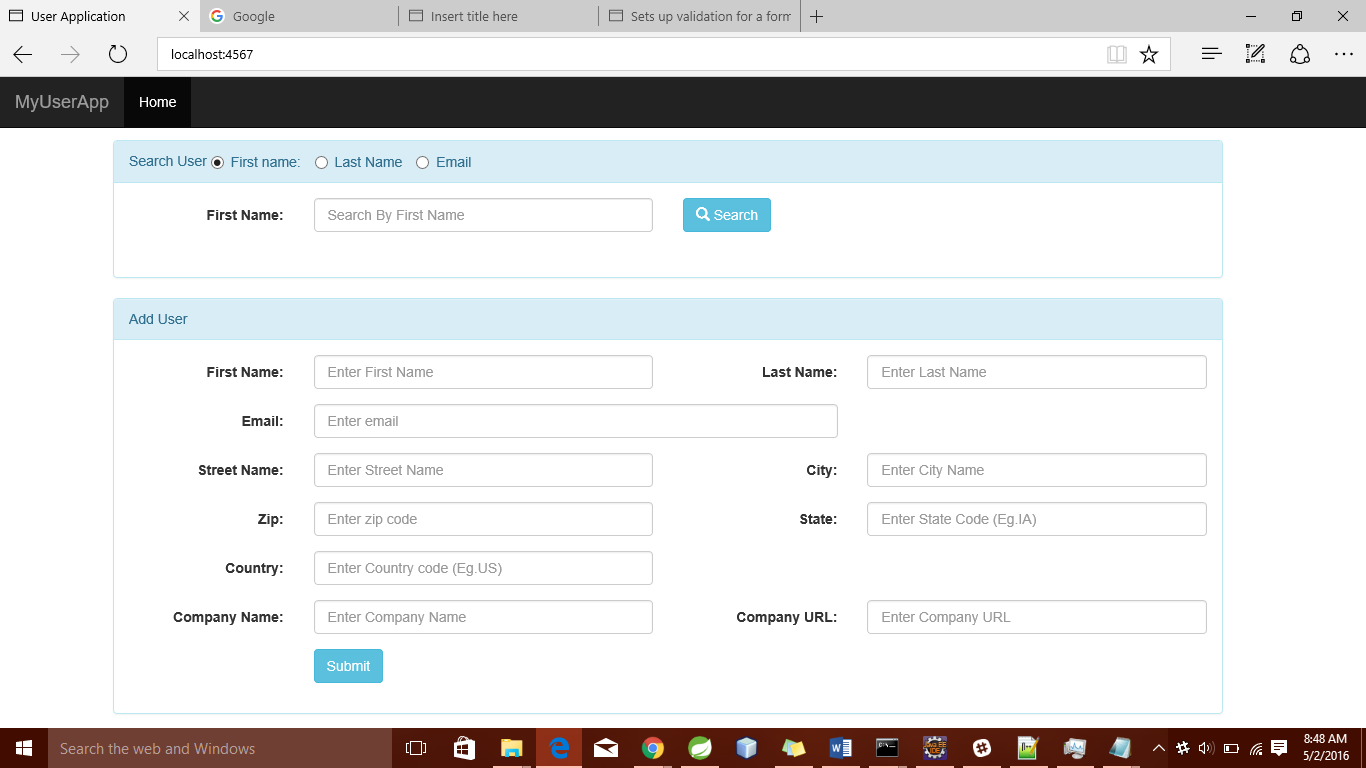


**Java Spark Server Starts on local host 4567 (Default)**

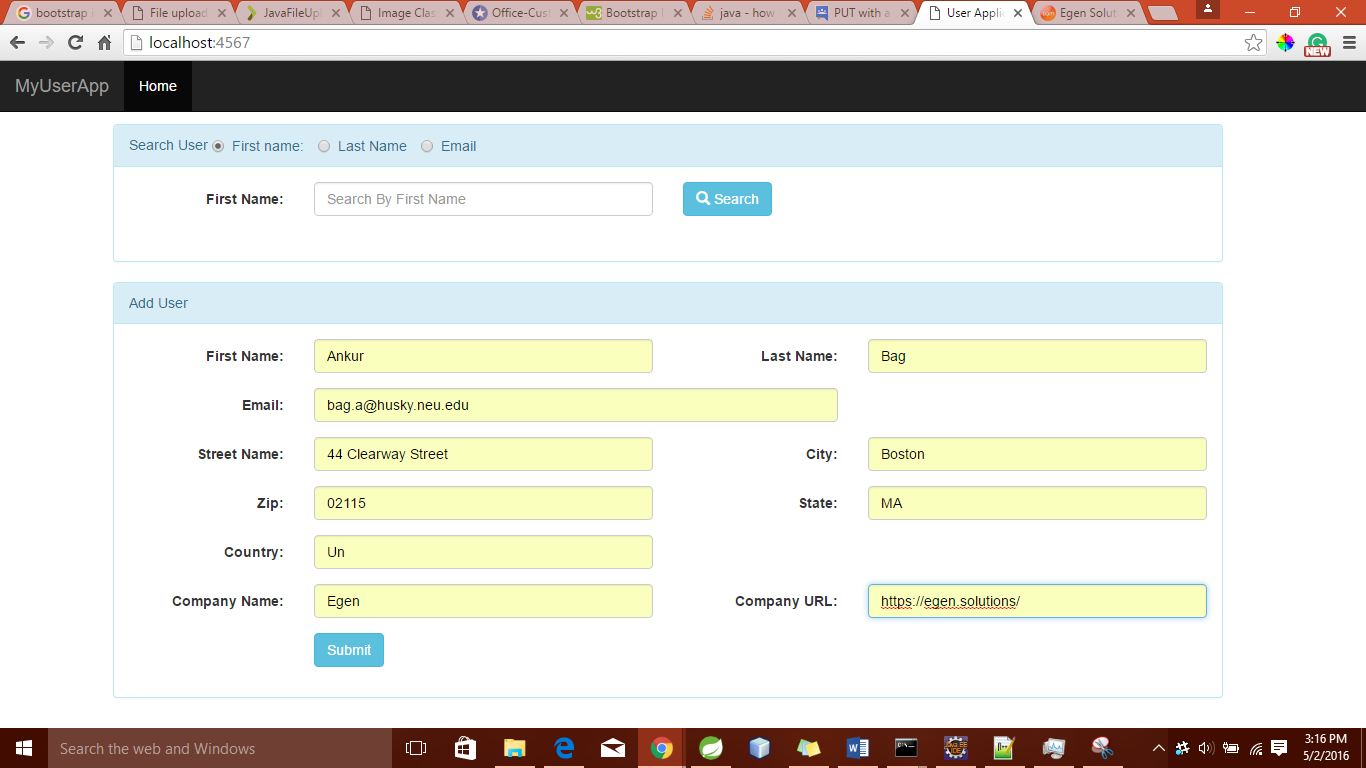


# **Use Cases**

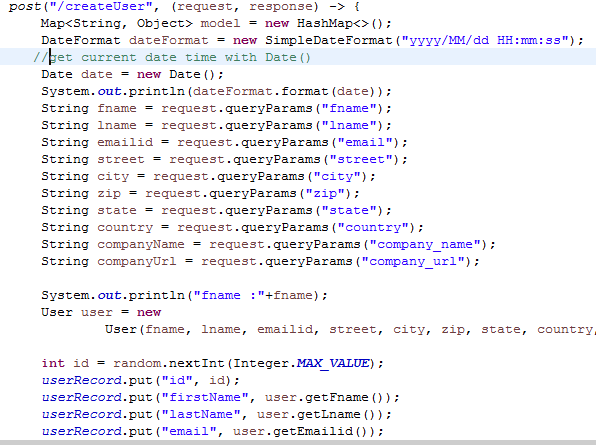
# **Home Screen**



# **createUser :**



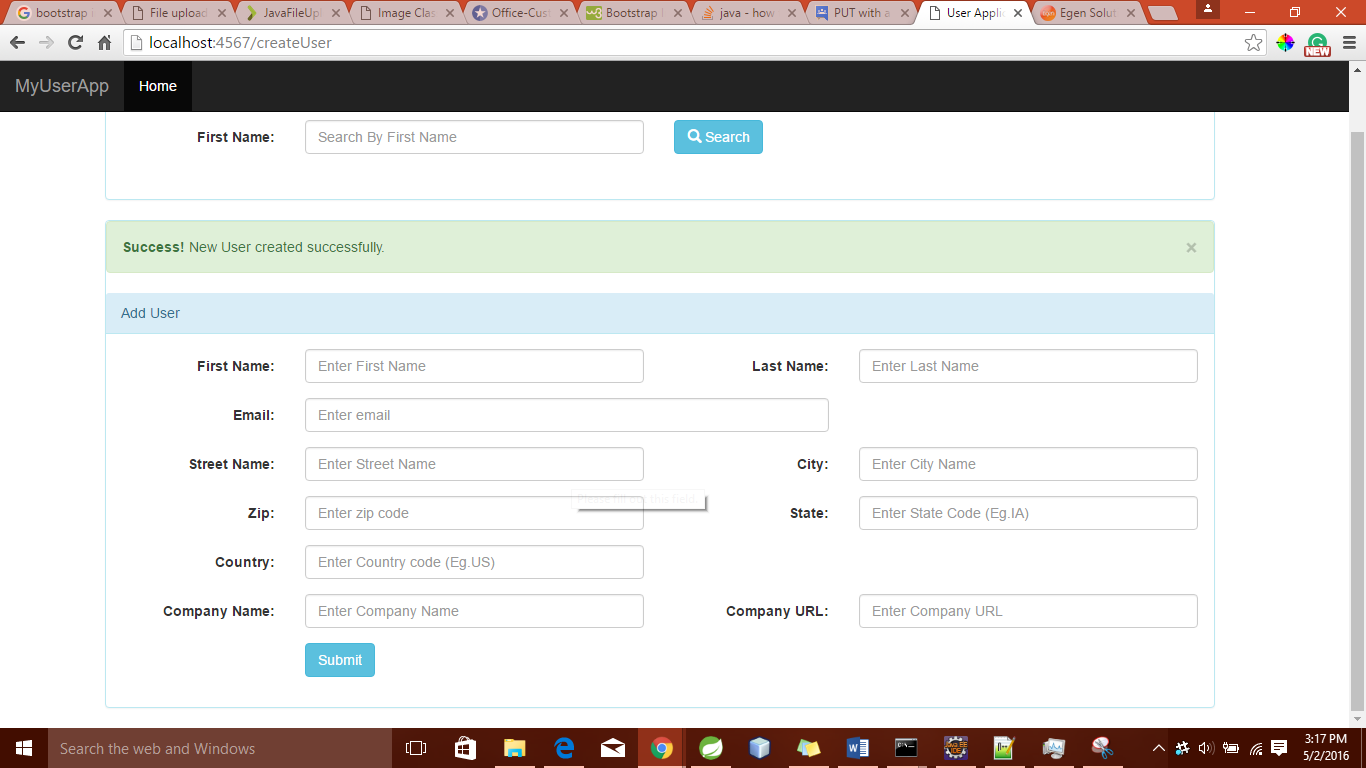
**Service invoked**



**Collection inserted in the DB**



**Success Screen**

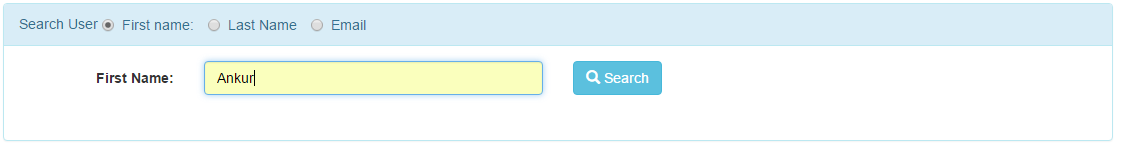


# **Mongo DB**

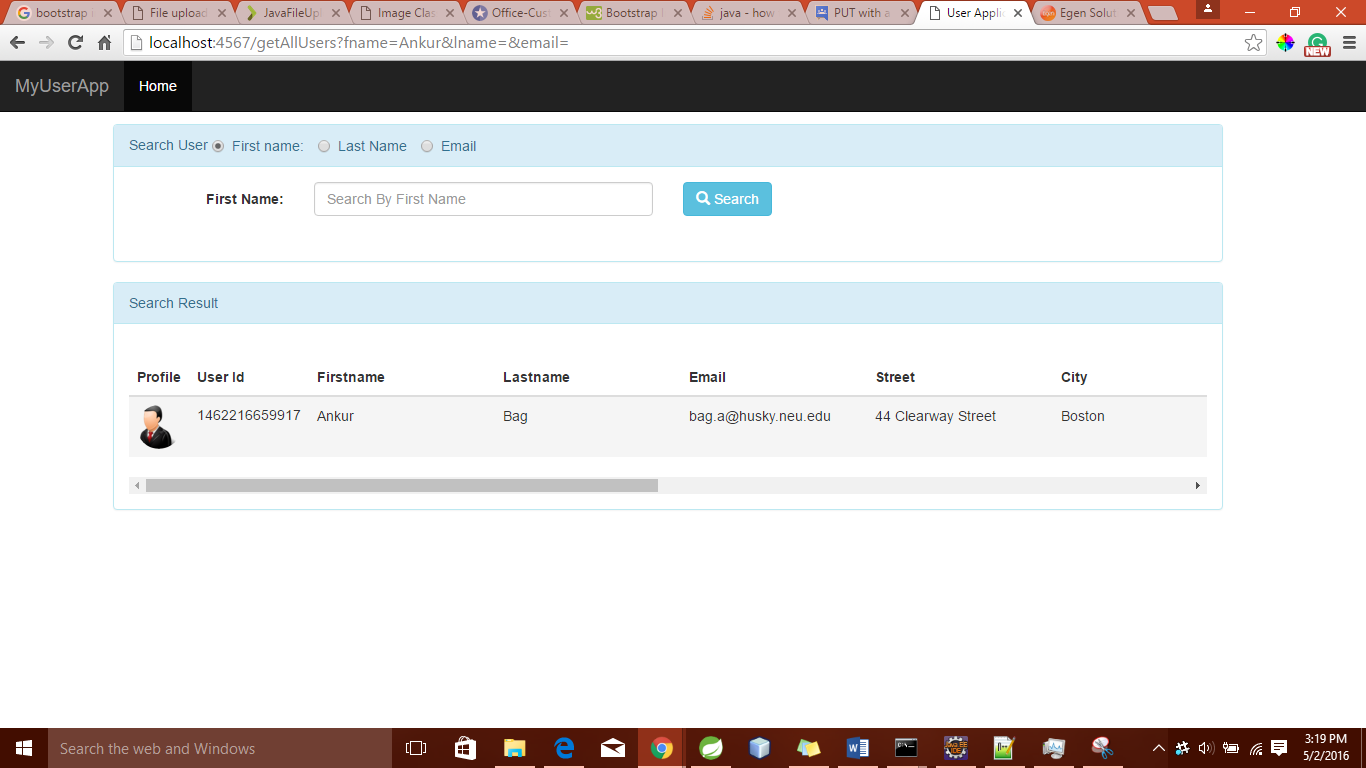
# **getAllUsers**

* + - * + Gives the list of all users that are in the data store

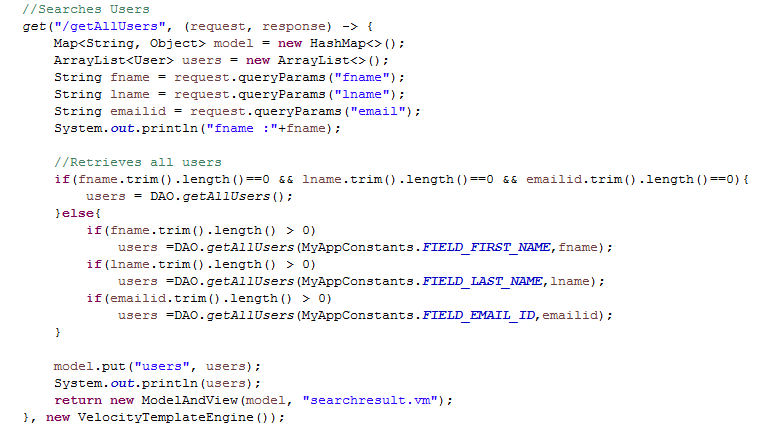
Search box is present on the Home Screen. I have provided various ways to search – Search By first name, last name and email



* ***User can search all user without any input parameter***

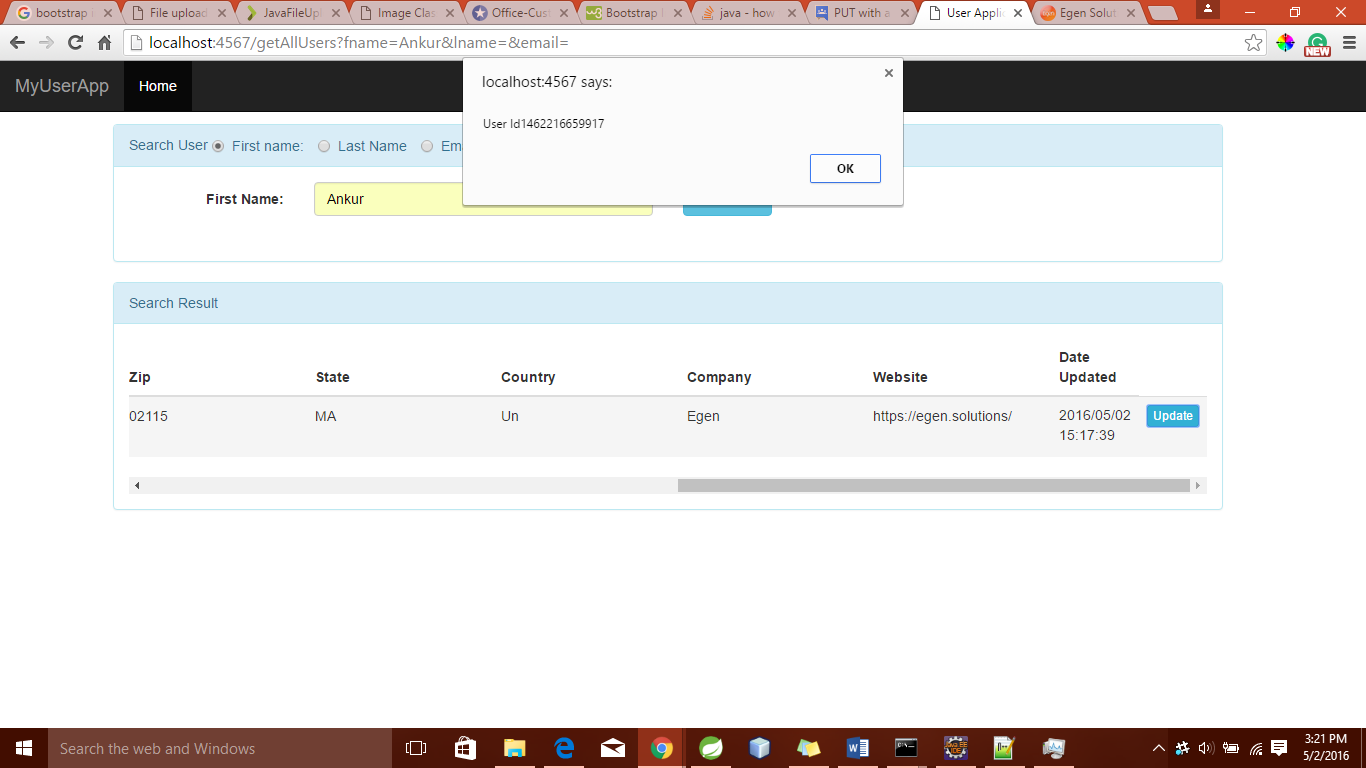


**Service Invoked**:

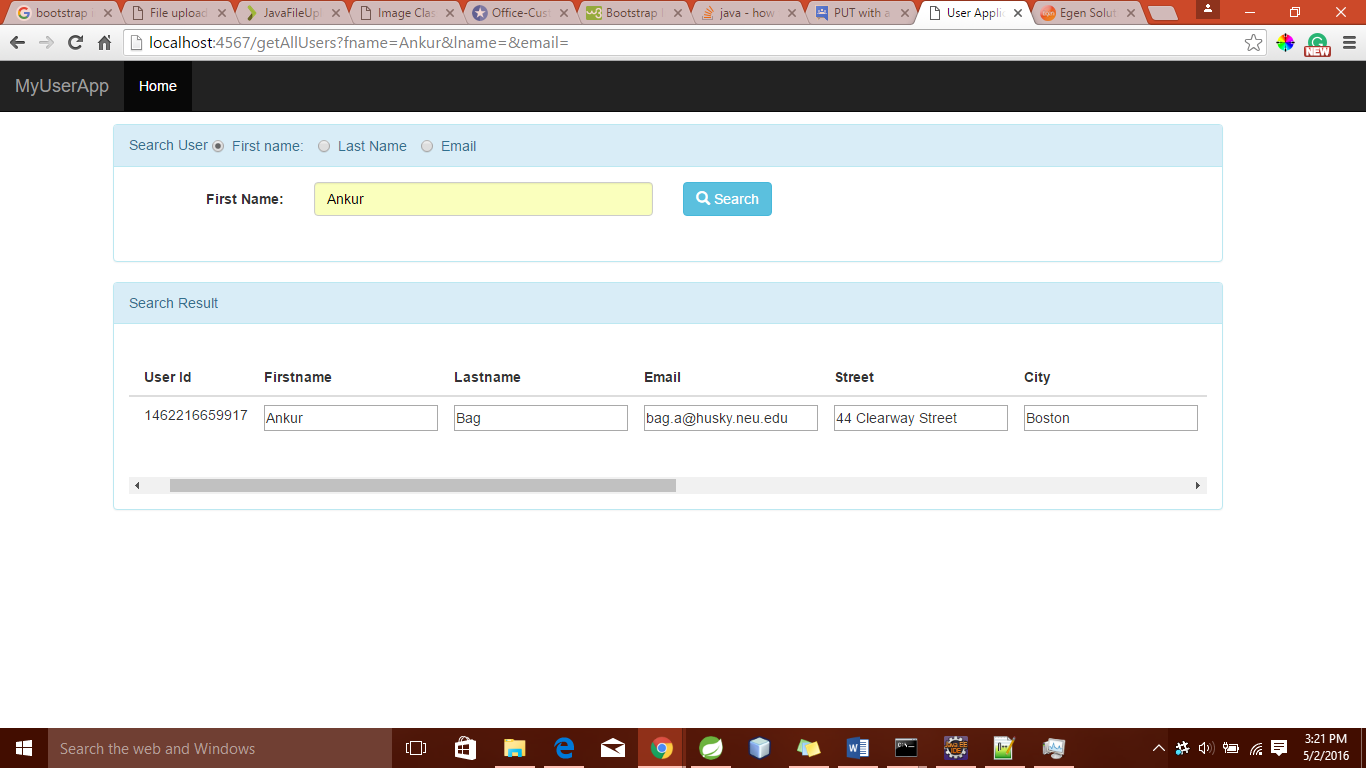


# **updateUser**

* Single Page App – where an AJAX call to the Update service and populating data on Velocity File (.vm).

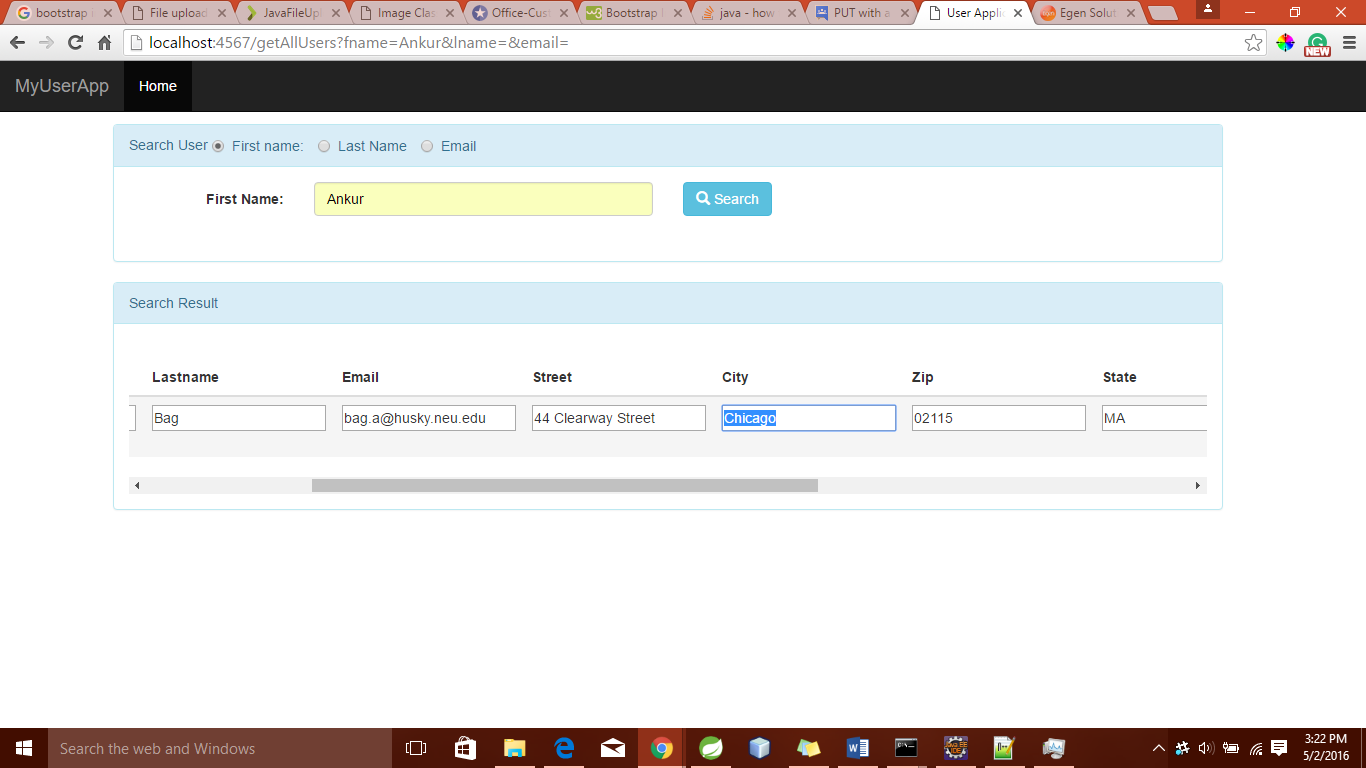


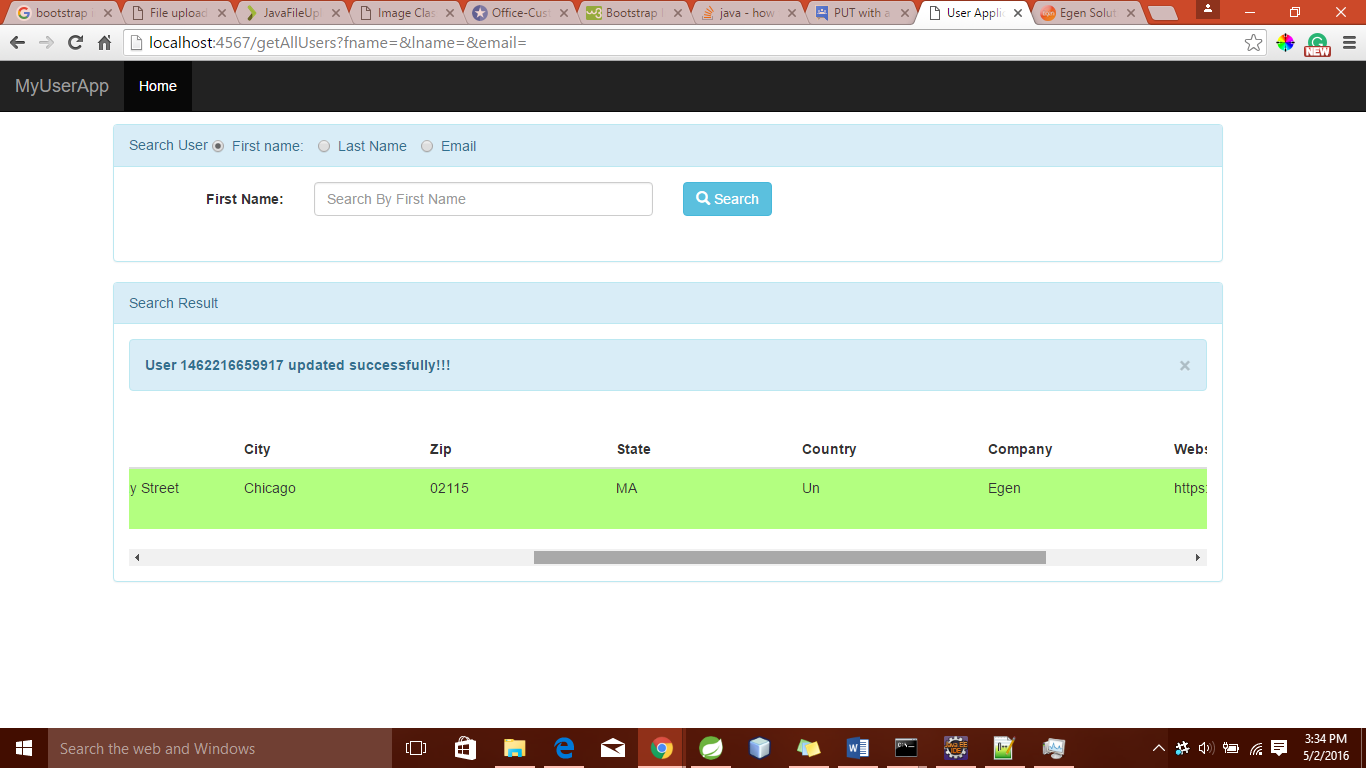
* Clicking Update Button turns the read only field into editable fields



* User Makes Changes

1. Valid Scenario





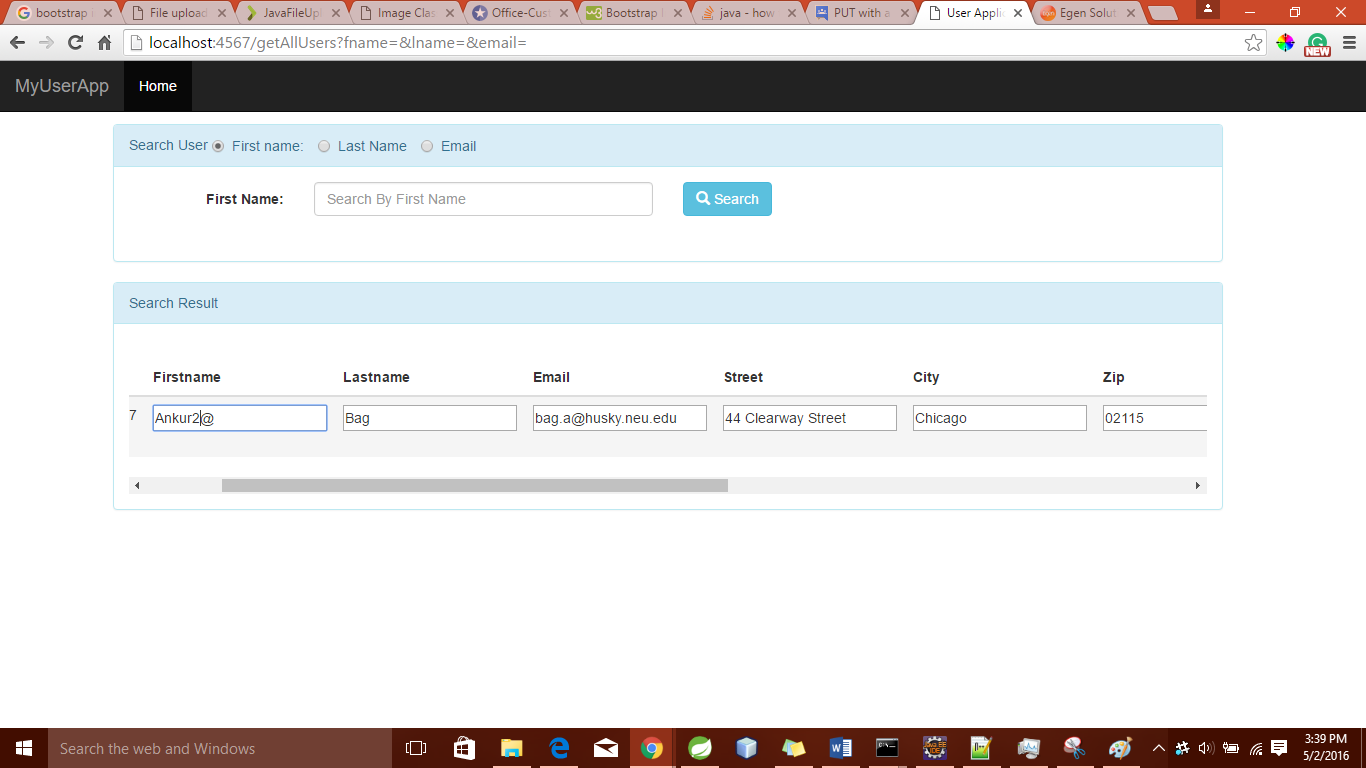
**Success Screen**

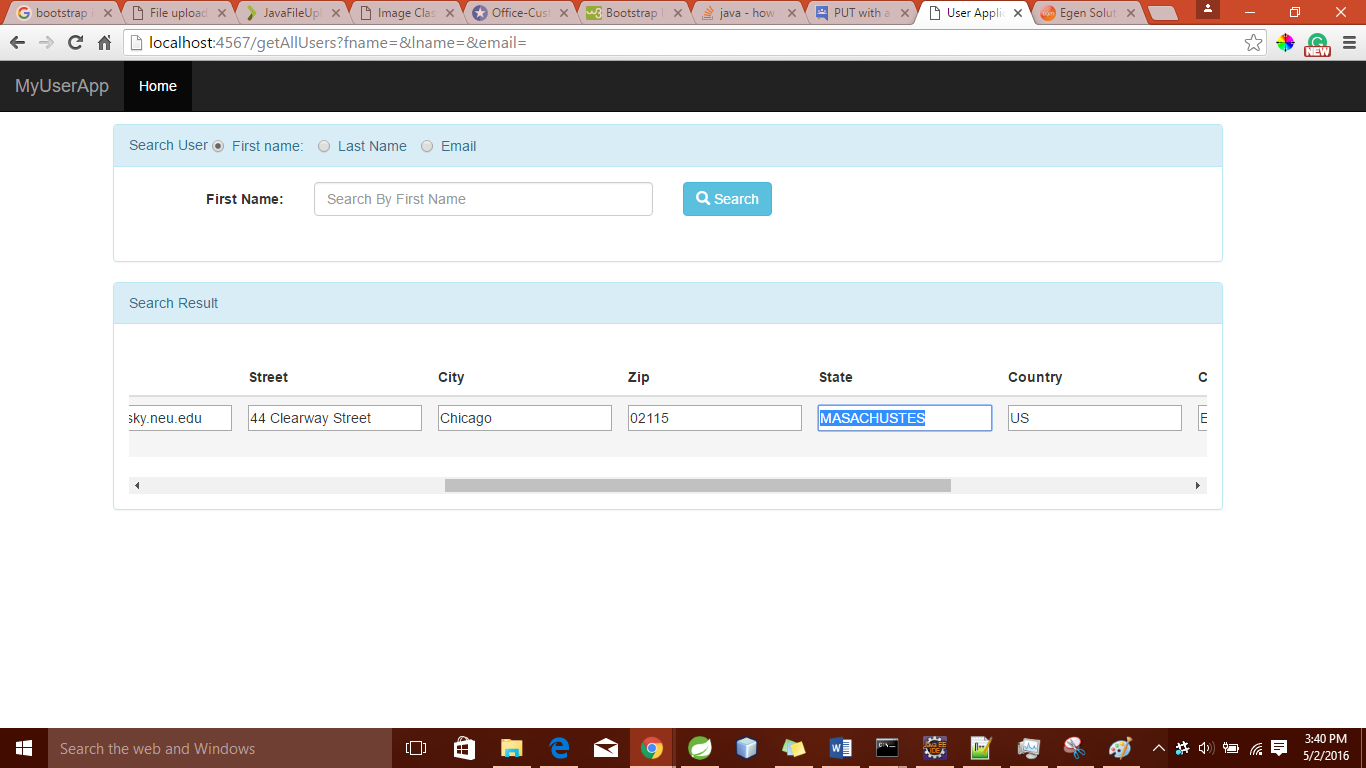
**Database**

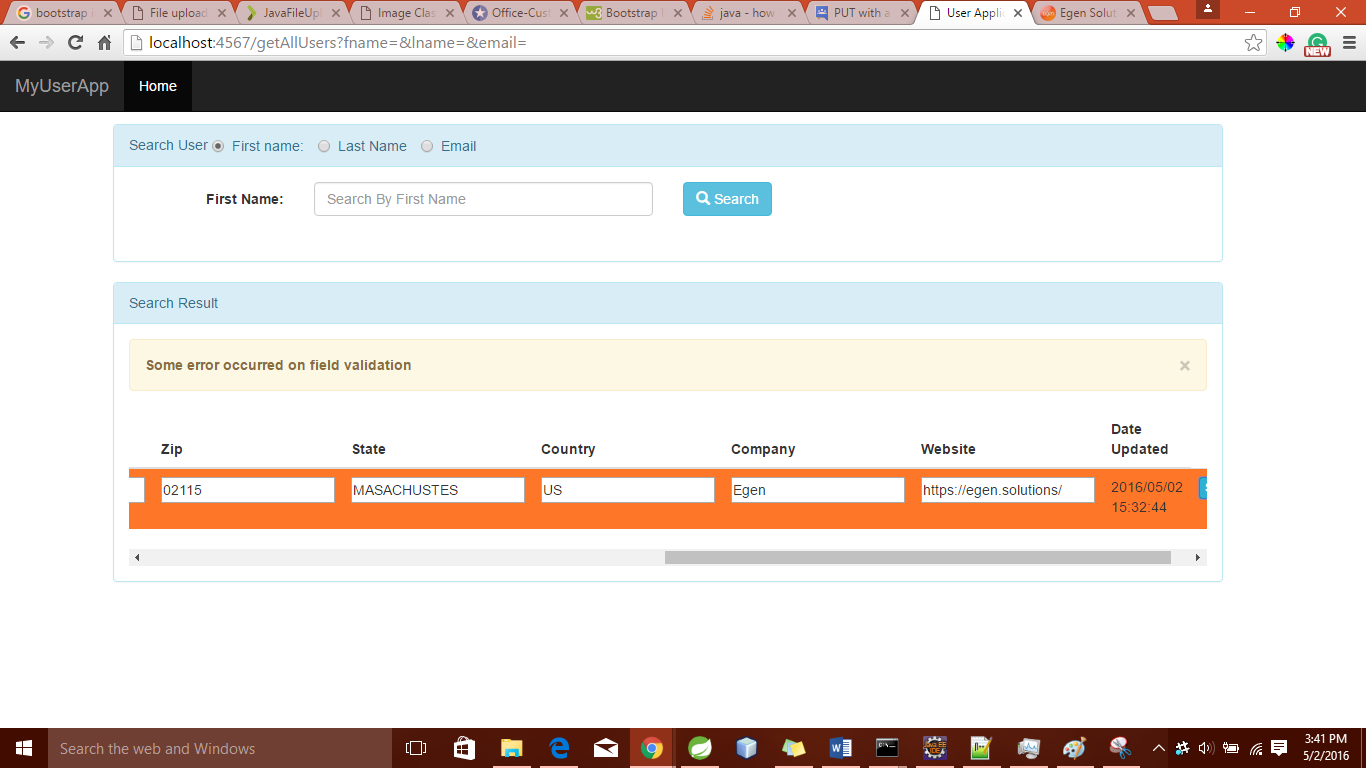


1. Invalid Scenario

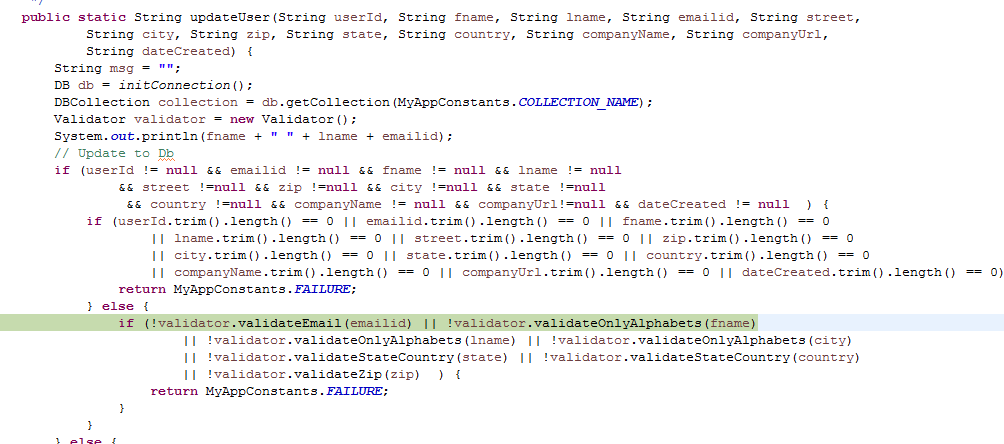
User Enters invalid input eg – Numeric Name , More than 2 Character States

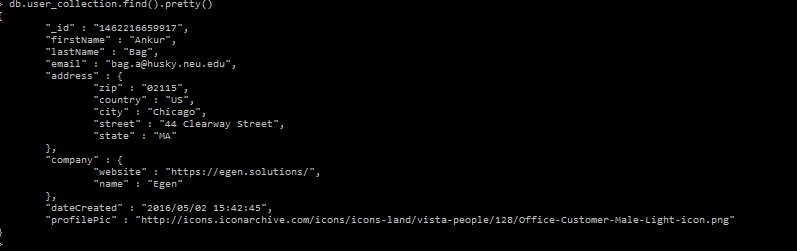




And Now Tries to save ; Gets error message

Server side Validation in place





**Ajax Call to *updateUser***



**Service Method**



# **JUNIT Test CASE:**

**Test Cases :**

1. **Check of Unique User**
2. **Updation on User**
3. **Getting users list**

