

# **SMART BIDS**

## **CS6314 PROJECT**

*by*

**(TECH DIVAS)**

**SUSHMA EATI (sxe160530)**

**MANISHA GALLA (mxg161430)**

**SAI MANI LIKHITA KANCHARLA (sxk167530)**

# Contents

<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
<b>2</b>	<b>TECHNOLOGIES USED</b>	<b>3</b>
2.1	LIST OF TECHNOLOGIES CONSIDERED . . . . .	3
<b>3</b>	<b>FUNCTIONALITIES</b>	<b>5</b>
3.1	. . . . .	5
3.2	FUNCTIONALITIES . . . . .	5
<b>4</b>	<b>WEBSITE</b>	<b>7</b>
4.1	FRONTEND WEBSITE: . . . . .	7
4.2	LIST OF WEB SERVICES . . . . .	8
<b>5</b>	<b>SUMMARY</b>	<b>10</b>
5.1	Problems Encountered . . . . .	10
5.2	Conclusion . . . . .	10

# List of Figures

1.1	Web Architecture . . . . .	2
4.1	Login Page . . . . .	7
4.2	Home Page . . . . .	8

# Chapter 1

## INTRODUCTION

A responsive e-commerce bidding website Smart Bid was developed as part of our Web Programming Languages Coursework through which user can post an item to sell which is available to other users for bidding. It was designed in a responsive fashion using Bootstrap templates. The detailed architectural diagram showing the interaction between various components is shown below

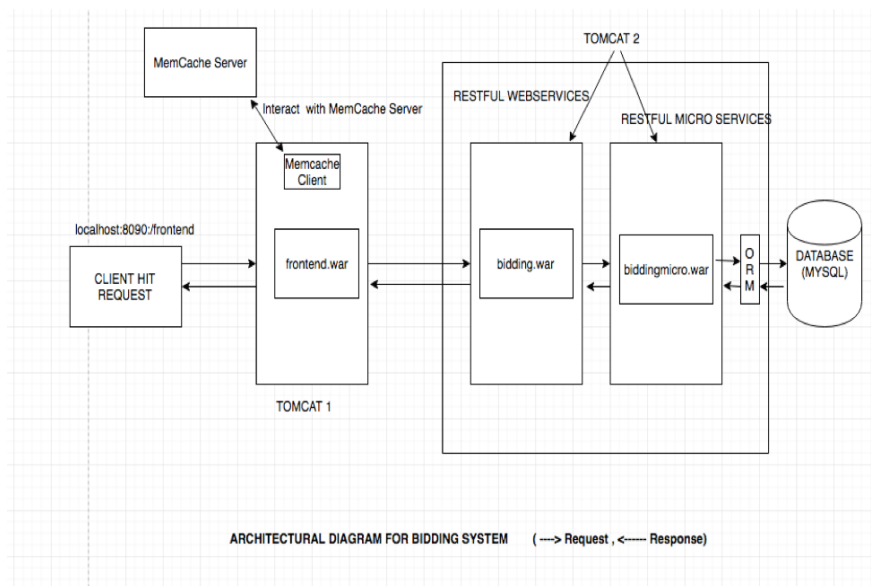


Figure 1.1: Web Architecture

# Chapter 2

## TECHNOLOGIES USED

- Front-End Technologies: Spring MVC for Controllers,HTML,CSS, JSP and BootStrap.
- Back-End Technologies ( Web Service Micro Services) : Spring MVC for hosting Web-services and Micro-services, MySql as Database and Hibernate for communication with database.
- Mem-Cache as caching technology.

### 2.1 LIST OF TECHNOLOGIES CONSIDERED

#### **For Building Web Services:**

-Apache CXF, Spring MVC and Java Servlets.

#### **Spring MVC**

-Easy to build web services. Many built in methods and annotation are present. So no over work of creating STUB classes. Works with XML as well as JSON.

#### **Front-End: JSP BootStrap**

-Template are already available on BootStrap site, so no overhead of creating from scratch. Templates alignment according to the screen resolutions is already taken care. **ORM:** (Hibernate, JPA Spring)

#### **Hibernate:**

-No need to change queries with the databases even if the database changes.

Table 2.1: Technologies

FRONT END TECHNOLOGIES	BACKEND TECHNOLOGIES
HTML	Java
CSS	Spring
BOOTSTRAP Template	Hiberante
JavaScript	MySQL
JSON	JSON
JQuery	RESTful
Memcached	

# Chapter 3

## FUNCTIONALITIES

### 3.1

#### NON FUNCTIONAL REQUIREMENTS

- HTTPS: Make tomcat secure by importing certificate to the ketstore, and make appropriate changes in the server.xml of tomcat.
- HTTP Compress Forwarding: Requests between all the web servers are compressed and send forward.
- Authentication: authorization key is added in the header parameter.
- MemCache: Used as caching technology in the front end.

### 3.2 FUNCTIONALITIES

- New user registration : A new user should register by filling out the registration form to create an account. Upon successful registration the user will be redirected to the login page.
- Existing user login and logout : Upon successful login user can access options to create a new bid, to bid for an item, viewing his/or her profile, check out the posted items and purchase history. The user can logout anytime after finishing his/her task is accomplished.
- User profile information display and editing : The user can view /edit his/her respective .
- User login information: The login information of the user and the last time and location of valid login will be displayed on the homepage of the website.
- Ability to post items to sell : A user can post an item which can be bid by the valid users.
- Ability to bid for items: A user can bid for item posted by other users
- Ability to automatically sell the item to the highest bidder : Click the close bid option which will automatically send a mail to the highest bidder.



- Page listing all your posted items : A page listing all the posted items where in an item can be highlighted using mouse over and ability to delete an item is provided.
- Page displaying all the bids for your posted items: Displays all the bids that are bidden for an item posted by a user.
- Search for items : A user can search for items by providing a keyword about the item and list of items relate to the keyword are displayed.
- Page listing all your bids and their statuses: Navigate to MyBids page to see all the bids bidden by the current user. 12 Results should be displayed in a sortable manner : Once the results are displayed. There is a sort option on various fields like price, Date, Name etc.
- A generic 404 page: A customized generic 404 page is displayed when the user tries to access an unavailable page .

# Chapter 4

## WEBSITE

### 4.1 FRONTEND WEBSITE:

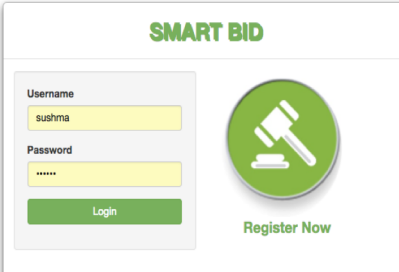
All pages of the website are built using Bootstrap Framework. When user clicks on <http://localhost:8090/frontend> he will be redirected to the landing page of the website, where he has the option to login if he is a registered user or register to the site if he is a new user.

#### 1. LANDING PAGE(LOGIN PAGE):

- Page with options to enter login.
- Register as new user.

#### 2. HOME PAGE:

- (a) Create New Bid: User can create post new item for bidding.
- (b) View all my Bids: User can find all the bids that he has bidden.
- (c) Search Item: Enter Keyword and search for an item.
- (d) My Profile: User can view his details from which he can navigate to edit details too.
- (e) Bid an item: User finds all the items posted by other users and can bid for an item.



SMART BID

Username  
sushma

Password  
\*\*\*\*\*

Login

Register Now

Figure 4.1: Login Page

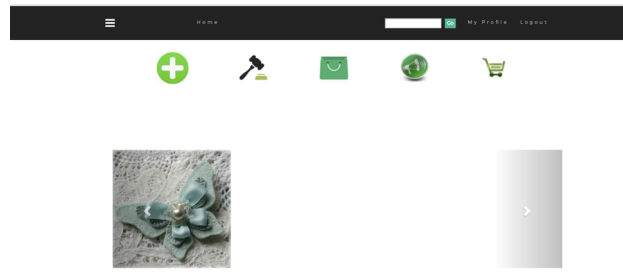


Figure 4.2: Home Page

- (f) Posted Items: Where user can find all the items posted by him and a link to navigate all the bids for that particular item, and an option to close the bid.
- (g) Purchase History: Where user can find all the bids that he has won.

## 4.2 LIST OF WEB SERVICES

### Register Web Service:

Forwards requests from client and forwards the request to the Micro Service.

- Register Micro Service 1: Checks whether username already exists, if so return appropriate message.
- Register Micro Service 2: If user doesn't exist, save him.

### Login Web Service:

Forwards requests from client and forwards the request to the Micro Service.

- Login Micro Service: Check whether the given username and password match with one another.

### Get User Profile Web Services:

Forwards requests from client and forwards the request to the Micro Service.

- User Profile Micro Service: Queries the database with the userid of the logged in user and returns the user json object.

### Create Item Web Service:

Forwards requests from client and forwards the request to the Micro Service.

- Create Item Micro Service: Saves the item created on the website in the item table.

### Bid Item WebService:

Forwards requests from client and forwards the request to the Micro Service.

- Bid Item Micro Service: Makes the bid successful and updates the status to ONGOING.

### Purchase History Web Service:

Forwards requests from client and forwards the request to the Micro Service.

- Purchase History Micro Service: Retrieves the purchase history of the logged in user.

**Search Web Service:**

Forwards requests from client and forwards the request to the Micro Service.

- Micro Service: Searches in the name description field of item table with the given keyword.

**Close Bid Web Service:**

Forwards requests from client and forwards the request to the Micro Service.

- Micro Service: Sends mail to the highest bidder user and changes status from ONGOING to BIDDED.

# Chapter 5

## SUMMARY

### 5.1 Problems Encountered

The main challenges encountered during the course of this project are

- We initially started individually working on frontend and Web Services on two separate projects. Later integrating them together was tough task where we faced some building errors which were resolved successfully.
- We explored Git and Bit Bucket version control systems to work as a team but we faced some issues while committing the changes and merging them together.

### 5.2 Conclusion

During the course of this project we have learnt how to integrate the various technologies and implementing the techniques we learnt during the lectures in the modules. It is a good opportunity to work in a team and work collaboratively by sharing each other ideas in achieving the required task.