

**Project Synopsis**

**On**

**Women’s E-shopping Web Application**

**Submitted By:**

**Murthy**

**Bala subrahmanyam**

**Amarendra**

**Tharunnath**

**Lalitha**

**Priyanka**

**Objective**

The system helps in buying of  goods, products and services online by choosing the listed products from  website(E-Commerce site).

**Description**

1. In day to day life, we will need to buy lots of goods or products from a shop. It may be food items, electronic items, house hold items etc. Nowadays, it is really hard to get some time to go out and get them by ourselves due to busy life style or lots of works. In order to solve this, B2C E-Commerce websites have been started. Using these websites, we can buy goods or products online just by visiting the website and ordering the item through online.
2. However, this project is specific for Sarees for women. An online store is a virtual store on the Internet where customers can browse the catalog and select different kinds sarees of their interest. A customer can go through the site and they can know the offers in festive season and discounts as well.





**Assumptions**

1. The user will register via this application, once user is register, user can login.
2. The product does require backend database server MongoDb for storing the username and password for different types of users of the system as well as various database .
3. After login, the page will be redirected to user profile page and list of products will be displayed to the user.
4. The user can filter the products by Color, Category and Price.
5. Once the user selects the particular product then the page will be redirected to product details page.
6. The user can see the similar products in same page and they can go through it.

**RISK**

1. Improper User Credentials leads to malfunctioning of user authentication .

**Approach**

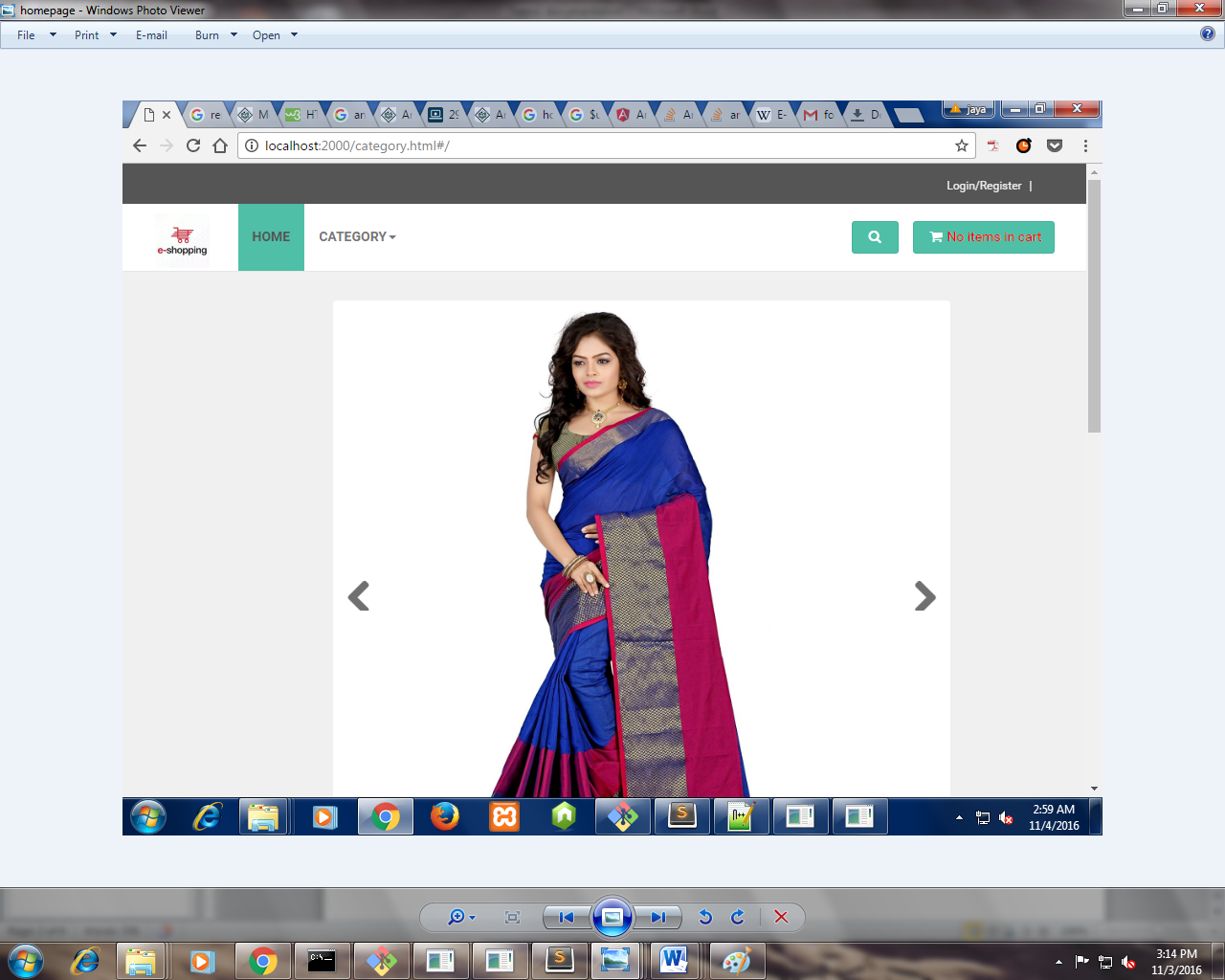
1. Initially the front view will be created using HTML5, Angular JS.
2. Based on Node JS the backend will be connected.
3. Major APIs like ServerConfiguration file for connecting to data base will be implemented using Node JS.
4. Validations for users input and authentications with database credentials, error handling.



**Modules**

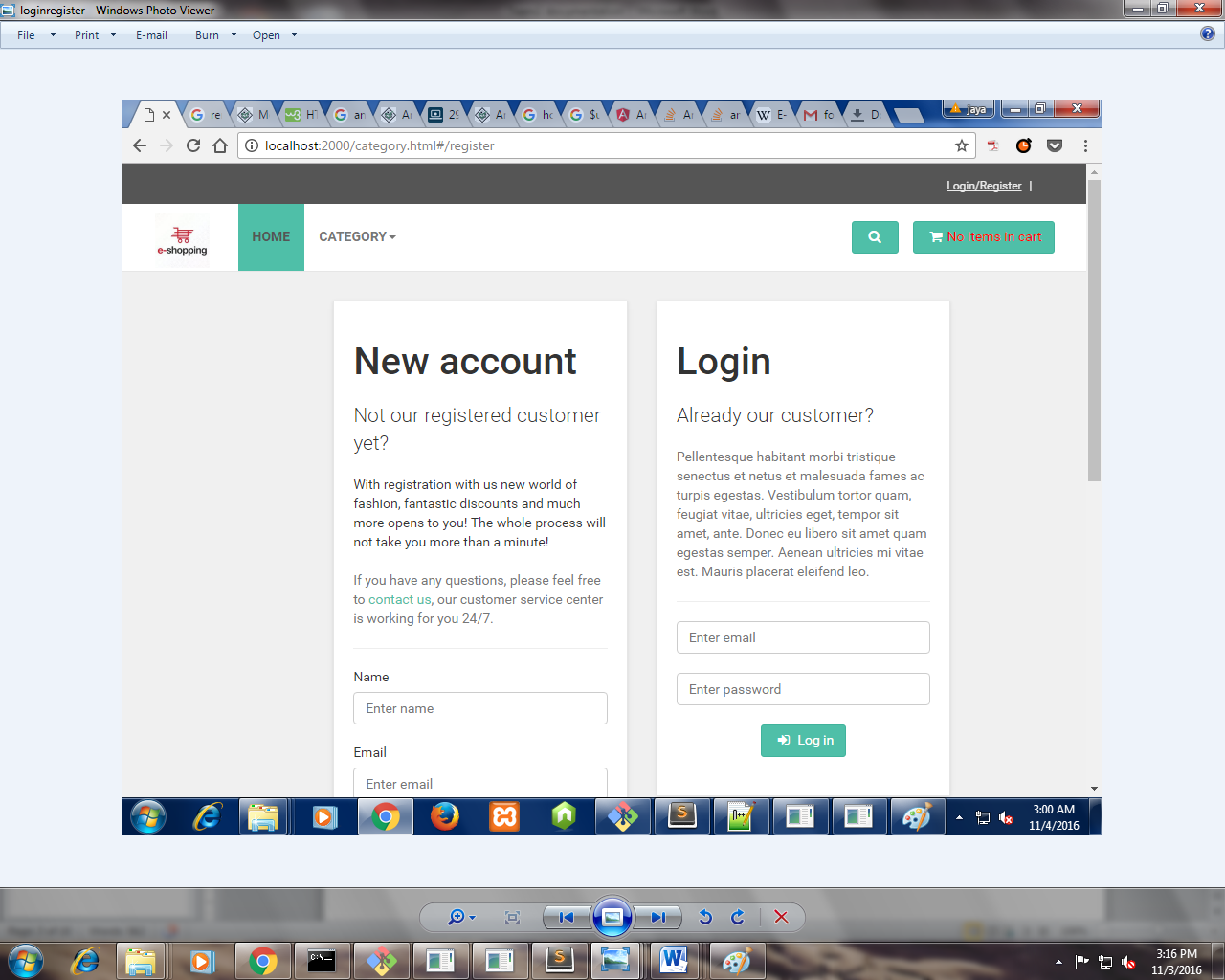
This project includes four modules

1. Landing/Home screen.

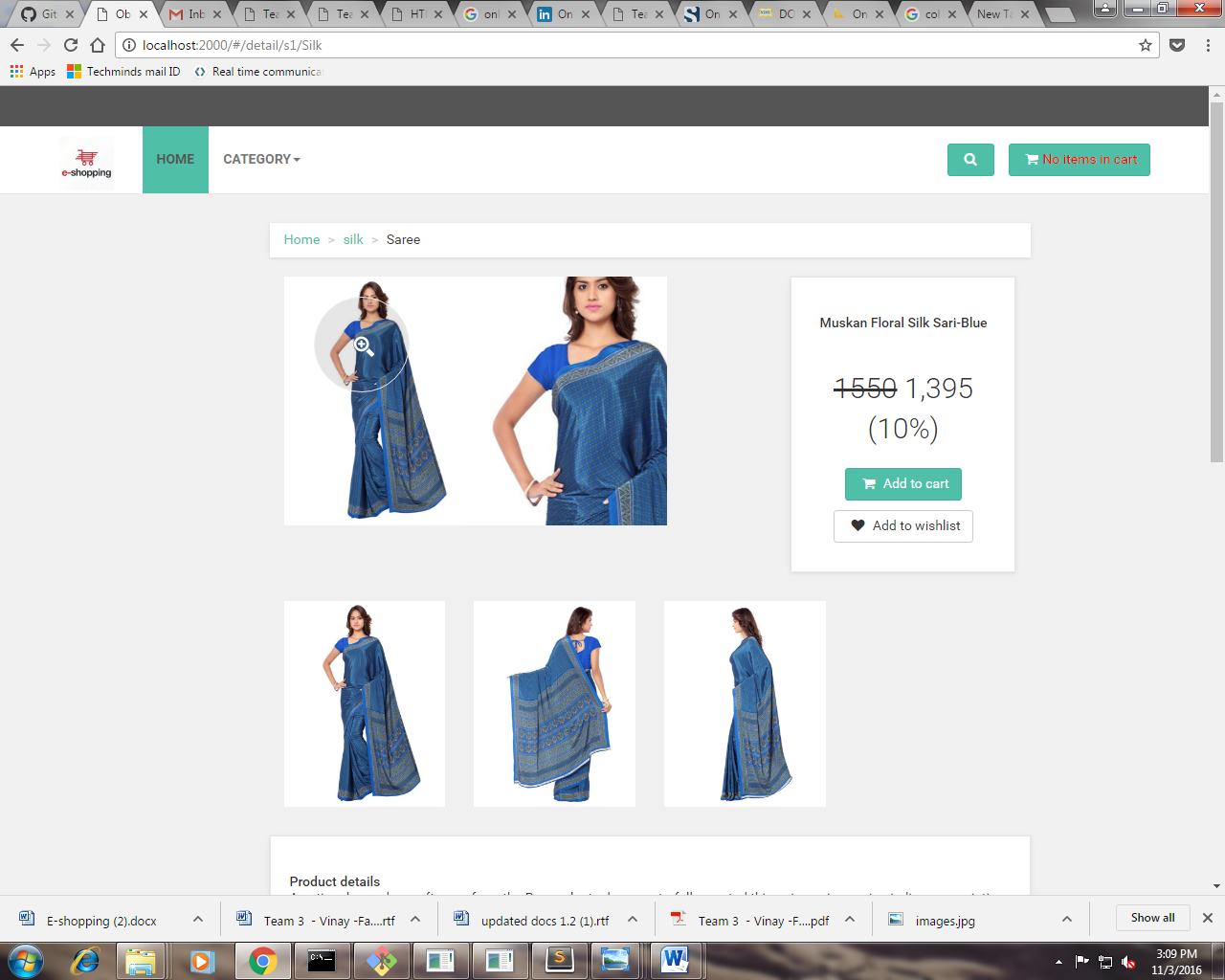




1. Login/Registration: This module is for the users who do not have their account. Here user is allowed to create an account to login.



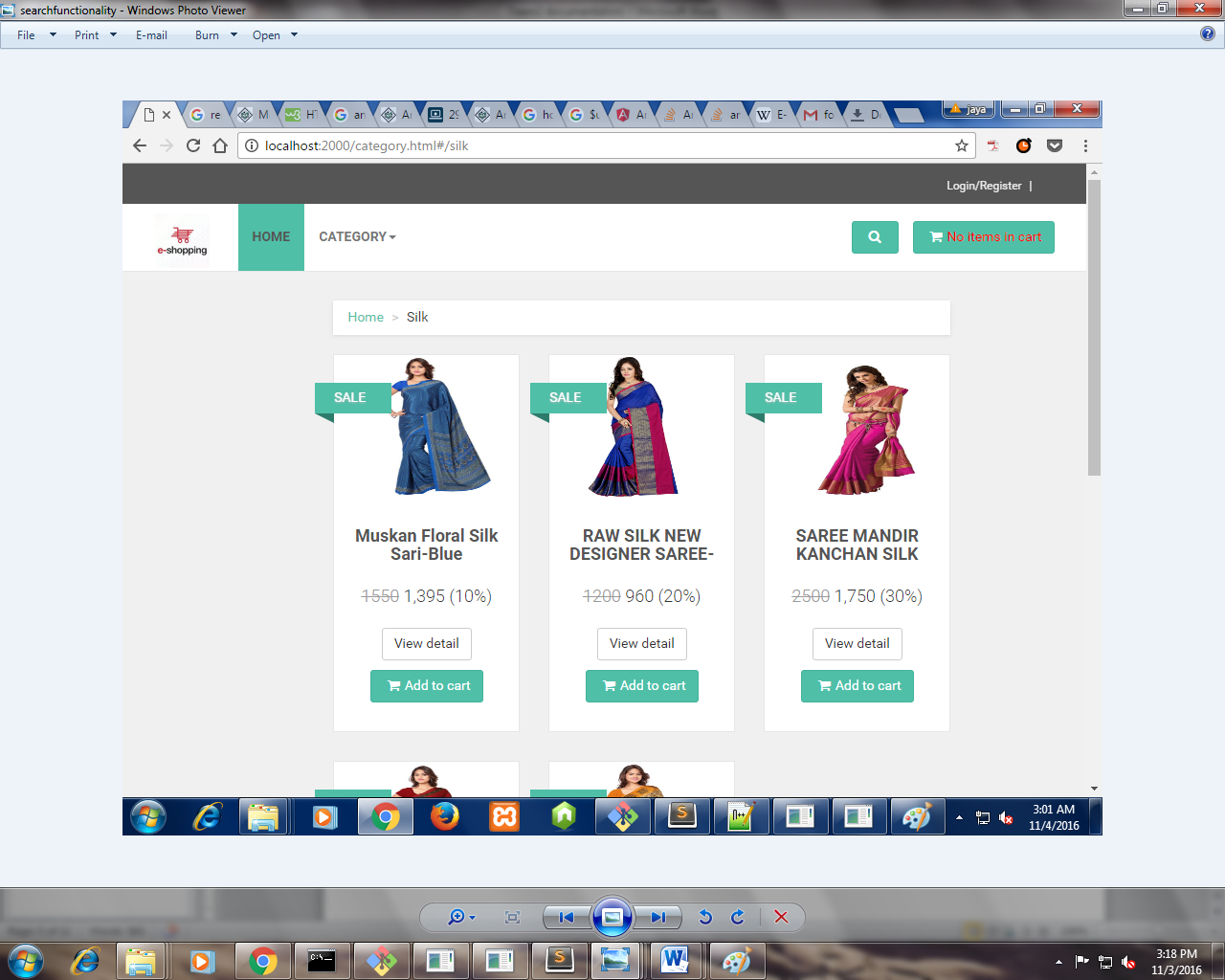
1. Product details:  This module has information regarding the products such as its name, model, color, price information, its features etc.







1. Search functionality: This module helps the customer to ease his search based on his  budget or interest. The search can be done on different categories like mobile model name, model number, colour, price etc



**API’S**

1. Login/Register and Connection API: It is used to connect the server and do the database operations.

* Murthy and Amarendar are involved in connection API.

1. User Profile API: It is used to retrieve and valid the userdata.

* Lalitha and Balasubrahmanyam are involved in User profile API.

1. Pagination API: It is used to load more data about product.

* Priyanka and Balasubrahmanyam are involved in Pagination API.

1. Search APL: It is used to search the products by price, category and color.

* Balasubrahmanyam and Tharun are involved in Search API.

**Test Cases**

1. Unit Testing

Bug: Once the user filled the registration form and registered, still the form contains the user details in respective fields.

Approach: we have used the user-defined refresh() function to referesh the details once registered.

Bug: When we developed registration and login module using nodeJs and mongoDb.

We got "Module Undefined Mongoose" error,as system unable to run mongod execution file.

Approach: we have used the command "mongod --storageEngine mmapv1 --dbpath 'c:/data/db'"

from the command prompt to the mongod exe file

Bug: once the user registered into the database,

As we used refresh() function,feilds get empty so we got validations problems.

Approach: $setPristine  $setUntouched

1. Integration Testing

* Naming conventions
* Missing of Script files
* Misplace of CSS files
* Binding of directives
* Call of functions

**Expected Result**

1. If all the credentials of the user is filled, it should allow user to register.
2. If the credentials matches, it should allow user to log-in.
3. Once the user search the product in search box, it should display the filtered data.
4. It should append the product once the user will select and it will be redirected to product details page.



**Technologies Used :**

1. Angular JS.
2. Node JS.
3. MongoDB.

**Conclusion**

1. Helps to improve enterprise application.

**Questions**

1. Here we are implementing only particular product(Sarees).
2. Approach for exception handling.
3. J Unit testing.
4. Code coverage.





