|  |
| --- |
| **INTRODUCTION** |

|  |
| --- |
| This project is an E-Commerce web-app which goes by the name “TechGenix.” This web-app aims to host a variety of tech products classified based on the category they belong to. The end user must be able to access various features of the web-app depending on whether the user is an admin or a customer. Admin has extra privileges to modify data stored in the database by interacting with simple forms in the web-app. Users must login to be able to add products to cart and purchase them. The personal and order details of the user are stored in a database for better experience.  The entire backend code is created using Java to facilitate smooth and hassle free storage and accessing of data. Spring MVC facilitates the flow of data between front and backend. Hibernate is used to store the user entered data in a database. Here the database engine used is H2 and Apache Tomcat is server on which the web-app was run and tested. The entire programming was done in Eclipse IDE.  Upon opening the web-app, home page is encountered which contains a header, footer and a carousel with links to their respective pages. In fact, every page has header, footer and a breadcrumb (except the home page). All these frontend pages are created with the help of HTML5, CSS and JavaScript. Some features of Bootstrap are also used. More pages will be explained in “Views” section of this report. |

|  |
| --- |
| **ACTIVITIES** |

|  |
| --- |
| Upon launching the web-app, the home page opens.    **TOP VIEW OF THE HOME PAGE**  From the snap above, it can be seen that the user has the option to sign up or login. The header also provides links to view different categories, go come back to home page and also a link to “About Us” page. In the carousel below the header, each category is played through with a link to view the products in that category. An admin has some extra privileges.    **HEADER, WHEN LOGGED IN AS ADMIN**  The header changes when logged in. It displays an option to log out and also displays the name of the user. Features such as “Add Product”, “Add Supplier” and “Add Category” are only available for admin.  From here the admin can make changes to the data available for customers. Customers on the other hand can proceed to the check the list of products available. They will be redirected to the page shown below.    **PRODUCT LIST FROM A CUSTOMER POV**  From here the customer can view individual products and add them to their cart. From there on, they can proceed to checkout and pay. Also the changes in the header can be seen here when the user logged in is not an admin. The options to add suppliers, products and categories aren’t shown. Also the cart size is shown.  These are the activities a user can perform in this web-app. The detailed explanation of each action along with the JSP pages will be given in the next section. |

|  |
| --- |
| **VIEWS** |

|  |
| --- |
| This section provides a list of all the JSP pages used.   * Header.jsp     **HEADER FOR A CUSTOMER**    **HEADER FOR ADMIN**  The header changes depending on who logs in. The customer does not have the privilege to add/edit products, suppliers and categories. But on the other hand, a customer has access to a cart.   * AboutUs.jsp   This page contains a general information about what this company does and what is stands for. It gives a brief description of the various types of products available.   * productList.jsp   Upon clicking any particular category, the user will be redirected to this page. This gives a list of all the products available in that category with their description and price. It provides an option to view a particular product. It also show the list of all categories on the left.    **productList.jsp**   * viewProduct.jsp   If the customer is interested about any of these products, they will be directed to this webpage after clicking “View Product.”    **viewProduct.jsp**  It displays the image, name, description, supplier name and category name. It also asks for the quantity. The user can use the plus/minus button to increment/decrement the quantity in that textbox. Finally if the user is interested in buying this product, he/she can add the product to cart.  Also, we can see the breadcrumb on top which is present in every page except the home page.   * cart.jsp     **cart.jsp**  The cart shows the list of products one has added along with the category they belong to and their price. It also displays the Shipping fee, Tax and Total. The user can either choose to continue shopping or place the order.   * shippingaddress.jsp   For the order to be placed, the user must fill a simple form to indicate the address they want the product to be shipped to. After filling this, they will be redirected to the order summary page.   * orderSummary.jsp   This page shows the summary of the user’s order. The user is then given an option to proceed to pay.    **orderSummary.jsp**   * paymentPortal.jsp   Upon clicking “Proceed to pay”, the user is taken to the payment portal where the user is supposed to enter his/her payment details.    **paymentPortal.jsp**   * home.jsp   This the first page that open in the web-app. As explained earlier, it contains a carousel with all the categories and a link to view them. Just like all other pages, it has a header and footer included. The carousel indicator at the bottom also the thumbnail when hovered over.    **SNIPPET FROM HOME PAGE**   * login.jsp   For a user to be able to add products to their cart and continue shopping, they should be logged in first.    **login.jsp**   * RegistrationForm.jsp   If the user doesn’t have an account to login, he/she must create one. The details entered by the user will be stored in the database.    **RegistrationForm.jsp**  Upon clicking Register, the details of the user are stored in the database. Below is the database of registered users. The database also contains tables for products, categories, orders etc.    **USER DETAILS IN THE DATABASE**   * productForm.jsp   This form is only available for admin. It is used to add products. The controller adds a model attribute for supplier and category list. It also adds an empty product variable.    **productForm.jsp**   * updateForm.jsp   This form looks exactly like the productForm, but it already has a product value store in the model attribute. The option to update and delete products are available at the bottom of product cards in product list for admin.    **THE EDIT AND DELETE OPTIONS ONLY AVAILABLE FOR ADMINS**   * supplierForm.jsp   This is yet another form which is available only for admins. This form is used to add a supplier to the database. Each product comes with a supplier name attached to it. To facilitate wide range of suppliers, this form is used.    **supplierForm.jsp**   * category.jsp   This form is also available only for admins. This looks exactly like the supplier form, but instead it is used to add categories. The option to add category can be found in the dropdown in the header.    **OPTION TO ADD CATEGORY IN THE DROPDOWN**   * updateCategory.jsp   This form is also again only available only for users with admin access. This form looks exactly like the form to add category. The option to update and delete category can be found in the category box in product list page.    **OPTION TO EDIT AND DELETE CATEGORIES**   * Footer.jsp   Finally, last but not least is the footer JSP file. This is displayed at the bottom of every webpage in the web-app.    **FOOTER**  This has the store description on the left followed by some quick links. It also gives contact details. The link under “Start Shopping” changes depending whether the user is logged in or not. At the bottom are the copyright statements.  -------------------------------------------------------------------------------------------------------  As this project is built using Java, HTML5, CSS and JSP, I would like to show some examples for each one of them below:   * Java     **A SNIPPET FROM PRODUCT CONTROLLER IN FRONT END**    **A SNIPPET FROM CATEGORY DAO IMPLEMENTATION IN BACKEND**   * HTML5     **A SNIPPET FROM HEADER.JSP**   * JSP     **JAVASCRIPT USED IN viewProduct.jsp TO FACILITATE INCREMENT AND DECREMENT BUTTONS**   * CSS     **SNIPPET FROM CAROUSEL.CSS USED IN HOME PAGE**  Apart from these, I have also used default style elements provided by Bootstrap. All the page redirections are taken care by dispatcher servlet and the controllers which use XML and Java respectively. Spring security and web are the other configurational files used (with .xml extension). Apart from these, the images are stored in a folder named resources. This folder also houses the CSS files. Both “views” and “resources” come under the WEB-INF folder, the path of which is described in the dispatcher servlet. |