E-Commerce applications using Web-Services

Gumma, Yatesh Laxman Kumar
Department of Computer Science
Lakehead University
Thunder Bay, Canada
ygumma@lakeheadu.ca

Seethamraju, Phaneendra
Department of Computer Science
Lakehead University
Thunder Bay, Canada
seethamraju@lakeheadu.ca

Tirumala, Sailesh
Department of Computer Science
Lakehead University
Thunder Bay, Canada
stirumal@lakeheadu.ca

Fiaidhi Jinan
COMP5112WD Supervisor
Lakehead University
Thunder Bay, Canada
jfiaidhi@lakeheadu.ca

Abstract - Web-Services can be referred as a family of technologies that standardized the communication of applications through world wide web in a cost-effective manner. Few of the major software vendors like IBM, Microsoft, Oracle, SAP are all embracing the web service standards into the newer versions of their applications which are web services enabled. All the constraints like cost, time, space for discovering, and e-business transactions can be solved easily by using web-services. With the introduction of these Web services design of the business application to provide service, integrating with other business entities, and conducting business transactions completely changed the working of a business organization. These web-services became a viable component in distributed E-Commerce platforms. In determining server response time, high speed communication systems along with computing capacity and network latency have become important. In web technologies a new model of architecture for distributed E-Commerce applications has been proposed which aims at integration and inter-operation between different platforms. In this paper we have proposed a model for developing an E-Commerce application using the Spring framework which helps in easy integration with other frameworks and solves most difficulties in an enterprise application development.

Keywords – E-Commerce, Web-Service, H2 Database, Spring Framework, Spring Security, JSTL, JSP.

I. INTRODUCTION

In the ancient days, the barter system is used for the exchange of goods and service which had further been replaced with money. With the advancement in technology saw the rise of the internet which paved the way for E-Commerce. Hence, most of the users prefer to sit at home and do shopping in a traditional way in ease. E-Commerce is an online market where trading of products and services or exchanging assets or information from place to place put, individual to individual or organization to the organization through an electronic system like the web. E-commerce is generalized into four domains. The following domains are discussed below:

- 1) **Business to Business (B2B):** The trading of goods and services between the organizations are engaged with this model.
- 2) **Business to Consumer (B2C):** At the point, when a business offers for sale of goods and services to an individual customer.
- 3) **Consumer to Consumer (C2C):** These involve the exchange of goods and service between individuals.
- 4) **Consumer to Business (C2B):** In this model consumers trade their goods and service to the organizations.

The desirability and significance of promoting adoption of information technology in regular business practices have been broadly perceived by governments all through the world. In Australia, both the Commonwealth and State Governments have set up procedures and projects to raise familiarity with online business systems, to promote the circulation of online technologies and to exhibit the advantages prone to gather from the adoption of E-Commerce practices.

E-Commerce can go up against an assortment of structures including distinctive transactional connections among organizations and shoppers, and additional distinct items being traded as a component of these transactions.

Few components of these transactions are given below:

1. **Retail:** The sale of an item by a business straightforwardly to a client with no intermediary.
2. **Wholesale:** The sale of items in mass, frequently to a retailer that at that point sells them directly to buyers.
3. **Outsourcing:** The sale of an item, which is made and transported to the customer by a third party.
4. **Crowd funding:** The collection of cash from clients in advance of an item being accessible to raise the startup capital necessary to put it up for sale to the market.
5. **Subscription:** The automatic repeating purchase of an item or service all the time until the point when the subscriber cancelled.
6. **Physical products:** Any tangible good that requires the stock to be replenished and requests to be physically shipped to clients as sales are made.

7. Digital products: Downloadable Digital products, layouts, and courses, or media that must be obtained for utilization or authorized for use.

8. Services: A skill or set of skills gave in return to pay. The service provider's time can be obtained for a charge.

Technologies that are used for implementing any E-Commerce application play a crucial role, as application must provide critical guarantees to consumers and suppliers of the services. In particular it is fundamental that such technologies ensure not only the security of transaction, but also adequate quality of service which is, availability of services should be guaranteed for all the users. One of the solutions for such kind of problems is developing the application by using Web-Services among the emerging technologies. Web-Services provides a means of expressing complex business to business process in terms of fine-grained sub tasks by implementing the service-oriented architecture. All the functional building blocks that are made by using web-services are accessible over a standard internet protocols and are independent of programming languages, platform. These services can be new applications or be wrapped around the legacy systems to make them network enabled.

Day by day new web technologies are being introduced but still the developments in business to business applications have been trailing the dynamics of consumer oriented electronic commerce. With the advancement in number of E-Commerce applications, business organizations slowly started adapting to the E-Commerce platforms in order to cut the operation costs by streamlining the vendor process selection and improving the collaborations. Today, many products that are sold online are designed as configurable products to satisfy the diversified requirements of customers.

II. BACKGROUND WORK

Wu Xiaoli and Tian Hui[1] have stated the wide range of popularity for the e-commerce in china and the people there have a clear idea about how significant is, e-commerce in their lives. As the e-commerce services are rising and being the standard for the enterprise level application. They have analyzed the growth in e-commerce at an initial state in China and then examined the main mode of e-commerce and the way e-commerce technology is realized. They explained the relationship between various bodies of e-commerce activities, the classification of e-commerce models and different views of transactions between the models such as Business to Consumer, enterprises and administrative institutions. Also, described in depth the creation and advancement of e-commerce and its key models, presents a traditional e-commerce platform, functional modules, network structure and other associated e-commerce technology combined with real projects.

Zhou Jie and Xie Hong[2] have termed e-commerce security as complicated in terms of technical point of view and to remain protected different kinds of legislative problems in e-commerce. They have also discussed about the applications analyzed its security. Have mentioned about some e-commerce elements like data integrity, certification and repudiation. They have also mentioned E-

commerce Security Measures such as different kinds of encryptions like Secret and public Key encryptions, Authentication, Digital Signature, Secure Electronic Transaction Protocol and other security technologies. They concluded that E-commerce security is at risk and can lead to data loss if the confidentiality, integrity related requirements are not included in the policy and that it is necessary to improve legislation of e-commerce security to reduce the problems in the rapidly developing e-business.

Yarong Zeng et al.[3] have expressed E-commerce as an important activity of Social economy and can create opportunities for the economic growth along with handling issues to the tax collection systems. Based on the complete analysis of then existing tax system they have proposed some strategies to tackle the problems faced by the administration. They have found several issues with the tax system in China 's e-commerce tax collection model such as problems in measuring the tax base, underdeveloped legal system, unable to specify the tax object and remaining. Finally, they have suggested many strategies for the development through e-commerce taxation related laws.

They have compared the e-commerce tax collection in china with their then existing data and projected the required data from rapid development of contacts between the enterprise and consumers and business' where the transactions happen more transparent, efficient and much cheaper. Along with which the authors have also explained regarding the Situation of legalization that electronic records the main reason for the electronic records to be a key element of tax collection. Pointed the problems in the system like Taxpayer and object of taxation were not very much clear even the business scope is increasing the trade tax registration was not suitable for the e-commerce, the pain in determining the tax revenue in e-commerce. Suggested to make appropriate changes to the tax clear elements and improve the relativeness in the taxation categories. Completely implement the E-invoice management where taxpayers can ask their invoice through internet and receive from ticket window by the registration and strengthening the co-operation in International Anti-Tax avoidance. In brief, tax collection and ecommerce administration is a long- and impossible task and should be focused on a long- and forward- viewpoint. Even though e-commerce has the negative effect on current tax registration, tax returns, tax collection, tax audit, tax sources tracking, tax services and so on, it is growing with great potential in China.

Hong Yu Et al. [4] proposed a reference model for knowledge management in order to do so, they combined information system strategies with business strategies. They used data mining techniques to support knowledge management in e-commerce. They described four functional modules, namely data collection, data preprocessing, pattern discovery and knowledge analysis of the integrated architecture The Data Mining techniques presented in this article are Path Analysis, Detection of Association Rules, Discovery of Sequence Pattern, Classification Analysis and Clustering Analysis. Implementing data mining technology to gain competitive advantage has been proved feasible and effective, for e-commerce enterprise. The article also presented some challenges in e-commerce DM which are as

follows for a large e-commerce website, collecting all click-stream data may not be feasible in both storage and cost. One solution to the problems mentioned above is to sample the data at the point of collection, but this solution creates more problems like sampled data will not be able to precisely capture rare events such as searching for a particular term or credit card authorization failure.

Lu Xia [5] presented the advantages of E-Commerce application for foreign trade companies and its application in China. He suggests government should optimize E-commerce application's external application environment, and to find its basic functions and advantages in foreign trade application, ensure its standard and legislation and guide the cooperation and coordination of pertinent departments. In the article, to perfect the service of intermediary platform of E-commerce he suggests some functions of the platform to be perfected like platform promotion ID- based authentication checking and online feedback.

Hongxin Li Et al. [6] stated that E-Commerce logistics integrates logistic services and e-commerce application services based on theories and by studying the developing trend of international logistic company and also advices that the e-commerce logistics is the future trend of logistic industries. In the paper the explain the concept of E-Commerce Logistics and he believes that e-logistics will undergo three phases the first stage deals with the process of logistic services, the second stage is to enable logistic services like activities of business electronic and the third stage is e-commerce logistics which is known as the ultimate stage of e-logistics. In the paper he clarifies that E-Commerce logistic is not to provide logistics services to e-commerce but it is to extend its services to the e-commerce while maintaining its own business in logistics.

III. PROPOSED SYSTEM

Application is developed using Spring framework. It can be considered as a collection of different sub frameworks like Spring ORM, Spring SOAP, Spring Security, Spring MVC. All the modules can be used either individually or separately based on the application requirement. One of the main advantages of the spring framework is the use of Stereotype annotations which reduced the effort of defining the configuration for each and every class in the separate XML file. In our application we have used all the modules together to achieve better functionalities in the application for example Spring Web-Flow module is used for providing flow between different UI by clearly defining the web flow rules in an XML file.

And for storing the data H2 database is used as it is an open source and lightweight java database. It can also be configured to run as in-memory database that is data will not persist on the disk. Because of its embedding feature its never used in the production phase, it is only used for the development and testing phases of software development cycle.

The complete application will be accessed by two different kind of user's customers and admins or the suppliers. Both the users have different views which are sorted based on the user type. Admins will be able to be managing products and

orders. Managing products includes adding new products or updating the pre-existing product details, along with activating or deactivating them based on stock availability. And customer will be able view the products based on the category, if they wish to purchase it can be added to the cart and proceed for the checkout. Following flowchart gives the in detailed picture of the application flow based on the user type.

Fig. 1 Application Flow.

Based on the MVC design pattern system the complete application has been developed in two different modules one which completely deals with the User Interface and the second module is used for interactions with the database. All the dependencies that has been used in the application development are taken from maven repository. All the dependencies related to the project are written inside the pom.xml file. Both the modules are linked by adding the database module dependency into the UI module as a maven dependency where groupId indicate the reversed domain that we have entered while creating the project, artifactId indicates the project name and version is standard, so it looks as below.

```

<dependency>
  <groupId>ecommerce</groupId>
  <artifactId>backend</artifactId>
  <version>0.0.1-SNAPSHOT</version>
</dependency>

```

Few other dependencies that has been included in the file are Spring Security which helps us in tackling the security challenges in the application and as well rendering the views based on the authorization. JUnit to ensure that all the persist and retrieval methods are working properly. Hibernate Validators for the form validations , H2 database dependency for interacting with the database and the Spring ORM which converts the data into object format. So, if there is any request from the UI module the Database module needs to respond back with the appropriate response so that the application can run without any errors. This MVC design pattern help us in separating the business logic from

the UI part. In this way we can get rid of some of the cyber-attacks.

A. Database Module

As said earlier the database we have used is H2 database as it supports most of the SQL, JDBC APIs and also it has strong security features. In order to work with database, we need to configure the database dependency in the pom.xml file. The dependency of H2 database looks as below.

```
<dependency>
    <groupId>com.h2database</groupId>
    <artifactId>h2</artifactId>
    <version>1.4.196</version>
</dependency>
```

Instead of closing the connection with the database, it is kept open with in cache file which is known as connection pooling. So, each and every time when the application interacts with the database it gets the connection status from the cache file. And all the parameters related to the database like connection URL, driver, username and password are maintained in a Hibernate configuration file. Data source is made available to every method in the application by creating the java bean file in the hibernate configuration file with the help of `getDataSource()` method whose return type is connection status. Hence, when ever the application needs to interact with the database, we just call the data source method to get the connection status.

As we are working with the remote interface so each call's between the interface is expensive. In order to reduce the number of calls we have created DTO classes which stands for Data Transfer Objects. DTO is an entity that carries data between processes and contains the complete call data. To make it go across the connection we need to serialize the DTO classes. On the server side the transfer data between the DTO and domain objects, Assembler is used. Usually DTO class contains bunch of fields along with the getter and setter methods for them. So, whatever data we are transferring from object to database will reside in the DTO package.

Data Access Objects (DAO) which is usually a design pattern is used for accessing the data from the data sources. Data Access Object Interfaces, Data Access Object Concrete Class, are two components of the DAO Design pattern. The interfaces for data access objects describe the basic operations on a model system. As we know that interface can contain only method definition but not any implementation. So, to work with those methods we need to implement those methods into a new class which is Data Access Object Concrete Class. And the implemented class is responsible to get the data from the source, which in our scenario is an H2 database.

Connection with the data source is managed by DAO only, once the connection is established DAO abstracts and encapsulates all access to the data source. All the access mechanism which is required to work with the data source is implemented in DAO and the data-source could be persistent storage like an RDBMS or a business service accessed via REST or SOAP. And whatever the classes that

we create to access the data from the database will reside in the DAO package.

The above class diagram demonstrates the usage of DTO, DAO and DAOImpl classes and their flow for a category class on which product type is being classified. Similar implementations were done on Products, Users, Cart, CartLine and Addresses classes.

Fig. 2 class diagram.

The proposed model was analyzed by using Junit Testing Framework which is most popular in Java. It is explicitly recommended for unit testing. However, it doesn't require any serve for testing the application which makes the process of testing faster and effective. In order to work with the Junit, we need to add the dependency of the Junit into our application in pom.xml file.

```
<dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>4.12</version>
    <scope>test</scope>
</dependency>
```

All the implemented methods in the DAOImpl classes are tested separately. In order to make the method Junit testable the method should be annotated with `@Test` annotation. All the resource that are required for the test method executing successfully should be defined in a separate method and annotating with `@BeforeClass`. The return value is then compared with the exact value by using the `assertEquals()` method. If they are not equal's, then an Assertion Error without a message is thrown. Following line of the code checks if the category has been added into database or not.

```
assertEquals("Successfully added a category
inside the table!",
true,
categoryDAO.add(category));
```

B. UI Module

All the web pages in our application are written using JSP, which is a server-side program and they enable to create the content on the web pages dynamically. Java Serve Pages (JSP) is a built on the top of servlet specifications and are platform independent method which is used for developing web-based applications. In servlets we write the code in java and then embed client-side markup into it but in case of JSP we start with the client-side markup then embed JSP tags to connect our page to java backend. These are similar to Java Serve Faces which are used for building MVC application. In our application the complete web page is divided into four different sections namely header, footer, side navigation bar, and main body content. In any web page main body content will be updated based on the web page and all the other sections will be common for all the webpages. All three common section are written into three different files respectively and then included into the main page whenever code by using the following line of code.

```
<%@include file="./shared/navbar.jsp"%>
```

Similarly, main body content sections for all the web pages are written in separate files. All the four sections are included in one file that is page.jsp and the main body content it will be displayed based on the user click handling the JSTL tags.

```
<c:if test="${userClickAllProducts== true}">
  <%@include file="listProducts.jsp"%>
</c:if>
```

So, if the user clicks on the 'View Products' then listProducts.jsp file will be rendered in the main body content section. We are fetching the user click from the controller. So, if the url is in the format of 'show/all/products', it will be returning the user click type along with the title of the page.

```
@RequestMapping(value = {"show/all/products"})
public ModelAndView showAllProducts() {
    ModelAndView mv = new ModelAndView("page");
    mv.addObject("title", "All Products");
    mv.addObject("categories", categoryDAO.list());
    mv.addObject("userClickAllProducts", true);
    return mv;
}
```

All the static resources like css, java-script and image are loaded using the JSTL url function by giving it a var name in order differentiate it.

```
<spring:url var="css" value="/resources/css" />
```

Dispatcher Servlet plays a vital role in spring mvc framework as it acts as a front controller in a spring-based web application. Provides a mechanism for the processing of requests where the actual work is carried out by configurable and delegated components. In any web application that is developed using Spring MVC the dispatcher is responsible for handling all the requests and responses in a web application so, the dispatcher servlet receives the Http request and maps it to the appropriate resources may be controller or models or views and the sends the appropriate response. It is inherited from

javax.servlet.HttpServlet and it is generally configured in web.xml file.

Any number of dispatcher servlet instance can be used in a web application. Each servlet will operate its own namespace, loading its own application context with mapping, handlers etc., In most of cases there will be only one dispatcher servlet that contains context url ("/"), where all the requests coming to that domain will be handled by it.

Fig. 3 Dispatcher Servlet.

Request from the client or the browser is initially handled by dispatcher servlet. Then the dispatcher servlet will be forwarding request to the page controller to identify the matching url pattern. If there is any pattern found it will be returning the view name and our view resolver is responsible for rendering the view on to the browser. But there is an while we are loading the static resource in to the initial home in order to solve the issues we have disabled the authorization feature on the static resources.

```
<http pattern="/resources/**" security="none"/>
```

All the views in our applications are render based on the authorization of the user which is handle with the help of spring security. So even if the user tries to modify the url access denied page will be displayed as we are checking authorization of the user before rendering the view. So, customers are restricted to view the products and letting them buy but in case of admin or supplier will be able to manage the products. Any unregistered user will be able to view products but if customer wish to buy them he need to be registered. It done based on the intercept-url pattern so if the url pattern is in the form of '/manage/**' then user need to have admin or supplier access to view those web pages.

Admin have an access for activating and deactivating or even adding the new product through the spring form. Each form is validated using an order to prevent the inserting null values into the database. We have two different kind of validators jQuery for user registration and user login form and for managing form we have used hibernate validators. So, while adding a product the image-type is also checked where we are allowing user to upload either PNG or JPG formats with a restriction in the file size.

If the admin wants to update the product details we will be redirecting them to the same product form adding with the

pre-existing details to the form. In similar way, form validation is performed when user clicks on submit, making the image upload option not mandatory. It will be validated only if user adds a new image in to the form.

We tried to implement the Spring Web-Flow module for the user registration page. The flow is completely defined in a separate XML file which is spring-security.xml. For displaying the products on the web-page jQuery data tables plugin is used. This provides all the features such as scanning, sorting and paging without any prior configuration. Data binding to the table is done by assigning the fields of JSON to the columns of the table.

For the product code we are using the randomly generated number from the randomUID() from java.util package. All the product images are stored in the local machine with the product code as the file name with jpg as an extension. So that it can retrieved easily by using the product code.

IV. RESULTS

Fig. 4. View products based on category

Fig. 5 shopping cart.

Fig. 6. Add product form validation

Fig. 7 Manage Products.

V. CONCLUSION

As per our proposed model we were able to create secured E-Commerce application using Spring MVC design pattern. And by using Spring Security we increased the confidential integrity by checking the authorization of the user for any view before rendering it on to the screen. For all the views apache tiles approach has been used by dividing complete view into different sections. In future, we will be integrating payment gateway in our application while user is checking out. Even the order confirmation will be sent to e-mails for both customer and supplier notifying about the order. Displaying the most viewed products as a suggestion for the customer based on the number of views, for this reason, we have been storing the view count in the product.

REFERENCES

- [1] Wu Xiaoli, Tian Hui. 2010. Discussion on E-Commerce Key Technology.
- [2] Zhou Jie and Xie Hong, 2010. E-Commerce Security Policy Analysis, International Conference on Electrical and Control Engineering.
- [3] Yarong Zeng, Xuejiao Guo, Yanni Yang, and Huijuan Huang. 2012. E-commerce Tax Collection and Administration in China, International Conference on Information Management, Innovation Management and Industrial Engineering
- [4] H. Yu, X. Huang, X. Hu and C. Wan, "Knowledge Management in E-commerce: A Data Mining Perspective," 2009 International Conference on Management of e-Commerce and e-Government, Nanchang, 2009, pp. 152-155.
- [5] Lu Xia, "The study of E-commerce application in foreign trade enterprise," 2010 International Conference on Future Information Technology and Management Engineering, Changzhou, 2010, pp. 16-19.
- [6] Hongxin Li, Ying Zhang and Jinfang Wu, "The future of e-Commerce logistics," 2008 IEEE International Conference on Service Operations and Logistics, and Informatics, Beijing, 2008, pp. 1403-1408.
- [7] <https://docs.spring.io/spring-security/site/docs/3.0.x/reference/el-access.html>
- [8] <http://www.h2database.com/html/commands.html>