E-Commerce Website Using ASP.NET MVC Framework

NOMULA MOHANSATISH[1], KALPANA K[2]
[1, 2] *Department of Computer Science and Engineering, Presidency University, Bangalore, India*

Abstract- A website for e-commerce was built using the ASP.NET MVC (Model-View-Controller) framework. Customers can use the website's intuitive layout to conduct searches for items, view products, and make purchases. The owner of the website may handle the product catalogue, complete orders, and maintain track of inventories through an easy-to-use administration interface. The project utilises A dynamic and seamless experience for users may be created by using innovative web development tools including HTML 5, CSS 3, the Bootstrap framework and jQuery. With an e-commerce web page, customers may shop conveniently and safely, and owners of websites can successfully run their online enterprises.

Indexed Terms- User registration, Customer login, Orders management, Payment gateway, Admin panel, Product catalogue, Shopping cart, Check out process, product information page.

I. INTRODUCTION

The use of information technology is growing swiftly and adding new applications. Many people believe that software products are an important administrative tool for the commercial sector since they offer more advantages than the traditional services they generally replace. They help to improve factors associated to processes, such as rapidity, quality, and accuracy. The objective of the project is to develop a practical web-based application for electronic commerce that will make it easier for consumers or clients of an organisation to choose products. A solid interface for users is one of the essential elements of an efficient e-commerce website, claim D. S. Bhat et al. (2016) [1]. After careful evaluation, Microsoft's ASP.NET was selected as the foundation and technology for building e-commerce websites.

Users of the project, which is also referred to as Application Hub, can visit a platform accessible via the internet where they are able to explore a wide range of software applications made by various companies by using the specified URL. Individuals who have proper login credentials can purchase the items they require. In accordance with the products the consumer has selected, the billing section will produce an invoice. The total cost of the item being bought will be displayed, and the customer can proceed to a secure payment gateway to finish the transaction.

The objective of the project is to develop a website-based system that provides clients with a central destination to purchase software products and services that are appropriate for their requirements. The objective is to increase marketing for the products produced by an organisation while making it simpler for consumers to make purchases by decreasing the amount of human labour as well as time needed for the item's purchase process.

1.1. Overview of the Project

This web application is built using the ASP.net Core MVC architecture and the front-end technology of JavaScript, HTML, and CSS, and it uses SQL Server from Microsoft for data administration and storage.
There are two methods to sign in to the project: as a user or as an administrator. The user is able to link to the application and view the project site after his or her credentials have been correctly and successfully entered into the database. First, the user registers in the online environment by providing accurate and necessary information. The admin login can only be used by administrators, and its specified credentials are stored in the database. This enables the admin to modify existing items, add additional ones, and remove any as necessary to adhere to corporate regulations.

- Home Module

When a user accesses a secure URL (Uniform Resource Locator), they are taken to the home page, where they can view the list of products and click on any item to be taken to the product module, move to other modules, or search for any item on the same page. The Home module shows a thorough list of every product that is currently available by reading the database and includes any alterations that the administrator has made.

- Product Module

The product information module, which pulls users away from the home page, displays all the details for whatever product they choose. These details are obtained form the database and may be changed at any time by the administrator, if necessary. If the item satisfies the user's requirements, they can evaluate all of its details, including price, before adding it into their shopping basket; if not, they can go back to the site and look at some other items.

- Cart Module

The Cart module shows the user's selection of the things they have put in their cart. Users can check every product's details and price while adding as many products as they like to their shopping basket. Users are able to eliminate existing goods and add new ones if that is not necessary. After that, the customer can pay for anything in his shopping basket and get an in-depth generated bill in the form of an email.

- Admin Module

The admin can use the Admin module to add, alter, or remove any item that is available for advertising on the portal.

The user will view the identical data in the main screen and product modules once an admin has added a product, which will be recorded in the database. This implies that the admin should add or modify a product with the utmost caution.

II. REVIEW OF LITERATURE

Software protection is one of the key challenges in PC maintenance. Security is achieved by the use of heuristics and spontaneous strategies, although the issue has not typically gotten the scientific attention it merits. We talk about fictitious programming security in this essay. It makes sense to store data on information storage servers, such mail servers and data servers, in an encrypted format to minimise security and protection concerns.

However, if we consider security, utility is lost. For instance, a user may want to locate the archives which include a particular set of terms but may not be aware of how to allow the data on a storage server to conduct the search and offer the information without endangering user privacy.

III. METHODOLOGY

To ensure proper and effective operation, a range of different methods are used at each phase of the project. Everything from introducing a user to a server's data to processing a user's payment depends on database operations. The ASP.NET Core Framework's MVC Framework and other Nuget packages are used to construct the application to offer a wide range of features.

After entering the right URL for the web page, the customer first sees the Home page. Customers will then have a choice between checking in and making purchases and logging in just before making a transaction.

3.1 Design concepts

Designing an effective front-end as well as a back-end interface is the aim of this project.

In this instance, the MVC architecture was used to develop both the data model and the view.

3.2 Front-end and Back-end Development

Front-end development, often called client-side web development, is the process of creating JavaScript, HTML, and CSS for a website in order that a user is able to interact with the software directly [3]. Because this project website's design aims to create an excellent User Interface (UI), and to make sure that people perceive the material on the website in an intelligible and pertinent style, front-end development was essential.

The process of writing the code on the server side that powers a website or app's background processes is known as back-end development [4]. The logic to generate the database and connect it with the application, as well as the facilitation of API integration, were essential to this project. This programme was written in C#, a language that strongly emphasises object-oriented thinking.

3.3 Database (DB) Management

The database is a grouping of structured data that a DBMS manages. The User, Product, or Cart tables are among the many tables in the project's database. These distinct tables each provide the necessary information in a set of columns and rows. The project uses back-end logic to access information in the database. Microsoft SQL Server served as the project's database administration system, and SSMS, or SQL Server Management Studio, was used to write database queries.

3.4 Model-View-Controller

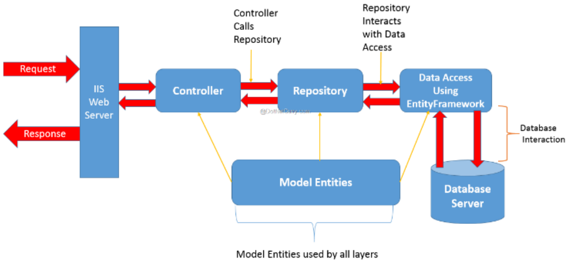
According to Krasner et al. (1998) in [5, MVC is a design pattern in which the user's experience (view), data (model), and the application logic (controller) are all separated. This pattern may help you distinguish between your worries about managing data and those about building a view of it.

Requests are sent to a the Controller, which collaborates with the Models to carry out tasks and collect data using the model of MVC for websites. The Controller informs the Model which View should be displayed. The final page is produced by the View using information from the Model. This project makes use of this layout to speed up application development.

3.5 Project Flow

A person enters the website's URL to begin the project. He then signs up as a customer before making an order via a method of payment.

The other components are connected by interfaces.


Request
Response
IIS Web Server
Controller Calls Repository
Repository Interacts with Data Access
Controller
Repository
Data Access Using EntityFramework
Database Interaction
Model Entities
Database Server
Model Entities used by all layers

3.6 Database Operations

The importance of acquiring data from databases to the creation of web apps is covered by Y. Bai (2020a) in [6]. A cross-platform, open-source, and compact variant of the classic Entity Framework (EF) is known as Entity Framework Core. It provides the Object relational Mappings (O/RM) functionality, which allows developers to work with databases while using.net objects by converting tables of database entries to objects. Most of the often required data-access software is avoided. EF Core supports a large variety of database engines.

In this project, Entity Framework is used to map the objects to the database in order to Create-Read-Update-Delete (CRUD) operations can be carried out on them. It is used to create an entirely novel entry in the database's customer table on the sign-up page. On the home page, it is used to read and display all of the products from the database. On administrative pages, it is used to update already-existing item entries, add new products, or delete existing ones from a database's product table.

IV. IMPLEMENTATION

The base of this system is Asp.net a 3.0 Core. Microsoft's Asp.net framework provides a programming environment for developing console or web apps. Nowadays, thanks to technological improvements, there are many tools accessible to create efficient and reliable websites. Web applications can be produced in an array of ways, from the user interface to the back end.

4.1 Software Requirements

Hosting

This project's hosting is provided by an IIS Server

- Database

Microsoft SQL Server 2019 is being used for this project's connection to the relational database management system (RDBMS).

- Front-end frameworks and languages

The user interface development of the website makes use of the following:
• JavaScript
• HTML
• CSS
• Bootstrap

- Back-end framework and languages

The following are utilised in this website's back-end development:

Model-View-Controller (MVC) and C#
A development environment that is integrated
The programme for the project was created using the following kinds of code editors or IDEs:
• SSMS, SQL Server Management Studio
• Visual Studio 2019

3.2. Development Platform
The platform was Microsoft's free and open-source Asp.net Core framework for the.net framework. A programming foundation for building console or web programmes is provided by the well-known.net framework. With the aid of object-oriented programming languages such C# or VB.net, programmers are able to design websites. Microsoft's Internet Information Services, also known as IIS, is required to operate web applications built with the.NET framework or any of its subsets on the server side.

C# was selected as the language used in this project due to its powerful OOP (Object Orientated Programming) methodology.

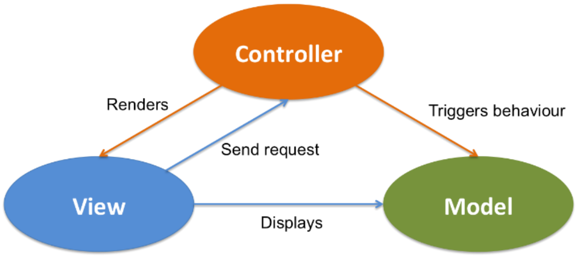
IIS, or Internet Information Services, is used to provide HTML pages.

The advantages of The Asp.net Core which rendered it a wise decision for this project are listed below:

It is free and open source, includes the Web API (Application Programmer Interface), MVC, which is and Asp.Net Web pages, and makes it simple to construct cross-platform software for Windows, Linux, and Mac.

4.2 Design Pattern
In this project, the Model-View-Controller, or MVC, design pattern was used. Under the MVC design paradigm, the user interface, information, and app functionality are all split in viewpoint, model, and controller, respectively.

Requests are sent to a the Controller, which collaborates with the Models to carry out tasks and collect data using the model of MVC for websites. The Controller informs the Model which View should be displayed. The final page is produced by the View using information from the Model. This project makes use of this layout to speed up application development. Figure shows how the MVC architecture is structured.



V. EXISTING SYSTEM

As this stage of outside application becomes more popular, hackers have started searching for dangerous programming. Given their widespread use, rogue software can make programmers a large profit.

In order to make information accessible to the public over the Internet and to expose them to a range of web-based attacks, numerous product architectures have evolved with a web-based component. A common and legitimate assault of this type is MySQL infusion, which can provide attackers unrestricted access to sensitive database that support Web applications.

Another highly automated solution for safeguarding Web applications is presented in this study. Unlike SQL infusion, which has advantages over the majority of existing solutions that are both usable and applicable. They solely offer their services in that area. The aforementioned websites offer illustrations of either pet accessories and commerce or just wellbeing. Up until now, not much has been said about the field of veterinary science in general. The only interface currently available in the system is text-based, which is less user-friendly than an interface with graphics.

4.1 Disadvantages

In order to send spam to a large number of users and their friends, the existing system can collect user private data like email addresses, hometowns, genders, among other things.

The existing mechanism has the ability to "re-produce" by making other damaging programmes popular.

VI. PROPOSED SYSTEM

SQL injection vulnerabilities are a result of inadequate information approval. Using acceptable, meticulous coding techniques is the obvious answer in this strategy for eliminating these vulnerabilities. Here, we give an overview of some of the written recommendations for mitigating SQL injection vulnerabilities that have been found to be effective.

This website offers solutions for all pet requirements in a one handy spot. An organised outline must be made before the framework is really built. The structure and distinguishing characteristics, including the data sources, production, records, databases, and techniques, are described in the logical overview. A functioning foundation, records, and real programme programming are generated by the physical growth, which adheres to the reasonable plan.

Here, we used the Web Applications SQL-injection Preventer (WASP) tool to conduct an observational analysis on a variety of web-based applications that we subjected to a wide range of attacks and actual penetration tests. WASP can thwart every single successful attempt and won't generate any false positives. Crypto and cryptology are occasionally used interchangeably since they are both employed to protect sensitive data. They differ in that steganography involves concealing data while the latter proves that no data is concealed in any way.

CONCLUSION

Many businesses have a practise of buying things. But it was pointed out that purchasing software is not an easy process. Right now, a lot more manual labour is necessary, and it takes a long time.

In order to make it simple for customers to readily buy any programme through a website that suited their needs, the concept of a "Software Centre" was established.

Customers can easily read the product's comprehensive description and order the one that best suits their requirements. This expedites and streamlines the client's software acquisition process. It also improves how products are marketed in this market.

For this project, the entire software procurement procedure was completed online with almost any assistance from human users. When the customer orders the product they need and finishes the payment process, the selected support staff can make additional changes, such as putting the item in the user's environment.

The process of buying electronic software, often known as online software procurement, has finished.

FUTURE SCOPE

Future improvements to the project will involve offering users online technical support. A technical assistance or assistance team member will get in touch with the user to address any problems they are experiencing with the software products they have bought and set up in their environment when the user reports those problems online. The project is not yet accessible worldwide because it was created for a particular area. A global accessibility initiative for this project may be implemented in the future, and transactions may potentially be supported for a variety of countries and currencies.

REFERENCES

[1] RESEARCH & DEVELOPMENT OF E-COMMERCE WEBSITE USING ASP. NET. (2023, April 21). *International Journal of Progressive Research in Engineering Management and Science*. https://doi.org/10.58257/ijprems30933

[2] Al-Farsi, A., Al-Mahruqi, A., & Vrindavanam, J. (2014, July 4). Application System for File Uploading using ASP. NET. *International Journal of Applied Information Systems*, *7*(5), 11–15. https://doi.org/10.5120/ijais14-451196

[3] Al-Farsi, A., Al-Mahruqi, A., & Vrindavanam, J. (2014, July 4). Application System for File Uploading using ASP. NET. *International Journal of Applied Information Systems*, *7*(5), 11–15. https://doi.org/10.5120/ijais14-451196