

**FINAL REPORT**

**Online News Channel**

**(**WEB-BASED**)**

**Submitted By**

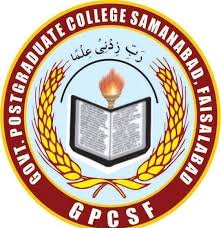
Usman Maqbool

(**2015-GCUF-055894)**

Hamza Ilyas

**(2015-GCUF-056017)**

**Department of Computer Science**

****

**Government Postgraduate College Samanabad Faisalabad**

**2019**

**CERTIFICATE**

This is to certify that **Usman Maqbool** and **Hamza Ilyas** bearing Registration No.**2015-GCUF-055894 and 2015-GCUF-056017** havecompletedthe final project titled as “**Online News Channel**” at the **Department of Computer Science, Govt. Postgraduate College Samanabad Faisalabad** to fulfill the partial requirement for the degree of **BSCS**.

**Supervisor**

Mudassar Maqsood Signature:

Hamid Mahmood Signature:

**Internal Panel**

Member 1: Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature: \_\_\_\_\_\_\_\_\_\_\_\_

Member 2: Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature: \_\_\_\_\_\_\_\_\_\_\_\_

Member 3: Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature: \_\_\_\_\_\_\_\_\_\_\_\_

Member 4: Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature: \_\_\_\_\_\_\_\_\_\_\_\_

**Project Coordinator Head of Department**

**Principal**

**DECLARATION**

The work reported in this project was carried by under the supervision of **Mudassar Maqsood and Hamid Mahmood** at Government Postgraduate College Samanabad Faisalabad.

We hereby declare that this project and the contents of project are the product of our own research and no part has been copied from any other written or published source (accept the references, standard mathematical or genetics models / equation / formulas / protocol etc.).s

We further declare that this work has not been submitted for award of any other degree / diploma.

The university may take action if the provided information is found inaccurate at any stage.

**Name: Usman Maqbool : Name: Hamza Ilyas**

**Registration No: 2015-GCUF-055894 : Registration No: 2015-GCUF-056017**

**ACKNOWLEDGEMENTS**

In completing this graduate project I have been fortunate to have help, support and

Encouragement from many people. I would like to acknowledge them for their cooperation.

First and foremost deeply thankful to Professor Dr. Soon-Ok Park, for her wonderful guidance

during this project work in field of Computer Science, at Governors State University. I am also

Thankful for her continuous feedback and encouragement throughout this project work. Her

broad knowledge and hardworking attitude has left me with very deep impressions and they will

greatly benefit me throughout my life.

I would like to thank my project Readers Dr. Kong-Cheng Wong and Dr. Steve Hyzny for their

support throughout this project work.

**ABSTRACT**

In today’s fast-changing business environment, it’s extremely important to be able to respond

to client needs in the most effective and timely manner. If your customers wish to see your

business online and have instant access to your products or services.

Online Shopping is a lifestyle e-commerce web application, which retails various fashion and

lifestyle products (Currently Men’s Wear). This project allows viewing various products

available enables registered users to purchase desired products instantly using PayPal payment

processor (Instant Pay) and also can place order by using Cash on Delivery (Pay Later) option.

This project provides an easy access to Administrators and Managers to view orders placed

using Pay Later and Instant Pay options.

In order to develop an e-commerce website, a number of Technologies must be studied and

understood. These include multi-tiered architecture, server and client side scripting techniques,

implementation technologies such as ASP.NET, programming language (such as C#) and

relational databases. This is a project with the objective to develop a basic website where a

consumer is provided with a shopping cart application and also to know about the technologies

used to develop such an application.

This document will discuss each of the underlying technologies to create and implement an ecommerce website.

**Introduction**

E-commerce is fast gaining ground as an accepted and used business paradigm. More and more business houses are implementing web sites providing functionality for performing commercial transactions over the web. It is reasonable to say that the process of shopping on the web is becoming commonplace.

The e-commerce concept relates to business or financial transactions that facilitate electronic payments of items purchased from online stores and service vendors. E-commerce covers a broad range of business activities, from digital content used for online consumption to conventional orders of online merchandise. Online banking is another form of e-commerce. E-commerce transactions are conducted between businesses, businesses and consumers, businesses and government, businesses and employees and consumers and businesses.

Online shopping is an e-commerce format, in which real-time sales transactions occur as a consumer purchases an item or service from an online store. This may be explained as an interactive collaboration between a consumer and merchant. In online shopping, there is no intermediary - just the interaction between the online buyer and store/service provider. Here, electronic financial transactions are conducted securely. E-commerce also describes the exchange of data between the financing, billing and payment aspects of electronic business transactions.

* 1. **Scope**

Electronic commerce (e-commerce) is the marketing, buying and selling of merchandise or services over the Internet. It encompasses the entire scope of online product and service sales from start to finish. E-commerce tools include computer platforms, applications, solutions, servers and various software formats manufactured by e-commerce service providers and purchased by merchants to increase online sales.

E-commerce facilitates the growth of online business. It is categorized as follows:

Online marketing

`Online advertising

Online sales

Product delivery

Product service

Online billing

Online payments

* 1. **Objective**

The objective of this project is to develop a general purpose e-commerce store where product can be bought from the comfort of home through the Internet. However, for implementation purposes, this paper will deal with an online shopping for clothes , electronics , footwear and many more.

An online store is a virtual store on the Internet where customers can browse the catalog and select products of interest. The selected items may be collected in a shopping cart. At checkout time, the items in the shopping cart will be presented as an order. At that time, more information will be needed to complete the transaction. Usually, the customer will be asked to fill or select a billing address, a shipping address, a shipping option, and payment information such as credit card number. An e-mail notification is sent to the customer as soon as the order is placed.

* 1. **Goals**

This year, here are a few good goals to consider setting for your website in 2019.

**Increase relevant traffic**

Increasing traffic is probably a familiar goal for you. For pretty much every website, accomplishing whatever it’s meant to do requires people visiting the site. But not all traffic is created equal. If your marketing is bringing in traffic, but it’s mostly made up of people who will never have a need for your products or services, then it’s not doing you much good. For instance, a B2B software company probably wouldn’t benefit from a huge amount of traffic if most of it was coming from teenagers

In addition to continuing your marketing efforts for raising traffic, this year take some time to analyze if the traffic you’re getting seems to match up with the target audience you’re trying to reach. In the Audience section of Google Analytics, you can find a lot of information about the kinds of people visiting your website, including:

Demographic information like age and gender

Interests they have, based on their other browsing information

Geographic information

Tech information like the device and browser they use goggle analytics traffic visitor interests

That can go a long way to helping you figure out if you’re reaching the people you’re trying to. If you find that you’re not, then it’s time to rethink and refine your marketing efforts to make sure they’re more targeted moving forward.

Once you increase your site traffic, it’s time to upgrade your hosting to meet demand. Learn more about your web hosting options to choose the best fit for your website needs.

**Grow your email list**

Getting people to your website is a big part of being successful, but getting them to come back so you can form an ongoing relationship is even more important. One of the best tools you have to do that is your email list. When someone opts into your email list, they’re opening the door to ongoing communication with your brand – which is a big deal! It’s no surprise that email marketing has one of the highest conversion rates of the main online marketing activities.

Email marketing ROI

That means that one of the best ways to improve your website’s success in the coming year is to put more of an emphasis on growing your email list. You can do that by offering deals to your customers for signing up, promoting it on social media, and including CTAs throughout your website urging people to subscribe.

And make sure you keep the emails you send relevant and useful so you keep those subscribers once you get them. They’re too valuable to lose lightly.

If you haven’t set up email marketing for your website yet, get started today with Constant Contact.

**Provide visitors a personalized experience**

Each of your visitors is a unique person who interacts with your website in their own way. Technology now makes it possible for you to track how people interact with your website and customize their experience on the site as they go. You can suggest items that are similar or complementary to what they’re looking at, or deliver up content on topics that are related to the piece they clicked on.

One study found that providing a personalized web experience for visitors can cause an 18% uptick in sales. When you make it easier for people to see the parts of your website that are most relevant to them, it increases their chances of having a good experience and continuing their relationship with your brand.

**Make your website responsive**

By this point, the reasons to make your website mobile friendly are well known and if you’re reading this, you’ve probably already taken that step (you have, right?). In the coming year, Google’s planning to roll out mobile-first indexing, meaning that their algorithm will give greater priority to your mobile website in determining rankings.

That means you want your mobile website to have the same content and information on it as your desktop website, just visually optimized for a mobile screen. Making your website responsive is one of the best and easiest ways to do that. And in general, make it a goal to make an effort to pay attention to your mobile experience and make sure your website is easy and intuitive for people on mobile to use. Mobile’s not going away and is likely to continue to become more important with every passing year, so you have to treat it like the priority it is.

**Increase sales**

For e-commerce businesses, this goal will remain consistent in all years. For your website and business to thrive, you have to be making sales. Everything else on this list is designed to help make that happen, but you should also look to your own analytics from past years. Figure out what tactics have worked the best for getting you sales – not just traffic or email signups (which matter too), but marketing activities specifically led to sales.

While there are general trends each year that should guide what you try, the most important thing you should look to is what’s working best for your particular business. At the end of the day, your business is unique. Only you can work out what makes the most sense for you and your audience. Analyze your metrics and design your marketing plan to optimize doing more of whatever is getting you sales now.

**Chapter No 2**

**Background**

By definition, ecommerce or electronic commerce, is the buying and selling of products or services via the Internet. For many Americans, ecommerce is something we participate in on a daily basis, like [online bill payment or purchasing from an e-trailer.](https://www.miva.com/mivapay?utm_source=organic&utm_medium=blog&utm_content=the-history-of-ecommerce-how-did-it-all-begin)

Nowadays the thought of living without ecommerce seems unfathomable, complicated and an inconvenience to many. It wasn’t until only a few decades ago that the idea of ecommerce had even appeared.

The history of ecommerce started 40 years ago and, to this day, continues to grow with new technologies, innovations, and thousands of businesses entering the online market each year. Electronic Data Interchanges and teleshopping in the 1970s paved the way for the modern day ecommerce store. The history of ecommerce is closely intertwined with the history of the internet. Online shopping only became possible when the internet was opened to the public in 1991. [Amazon.com](https://amazon.com/) was one of the first ecommerce sites in the US to start selling products online and thousands of businesses have followed since. The convenience, safety, and user experience of ecommerce have improved exponentially since its inception.  This article will address some of the key players and milestones of ecommerce.

**When was online shopping invented?**

Online shopping was invented and pioneered in 1979 by Michael Aldrich in the United Kingdom. He connected a modified domestic television via a telephone line to a real-time multi-user transaction processing computer. The system was marketed beginning in 1980 and offered mainly business-to-business systems that were sold in the UK, Ireland, and Spain. One the earliest consumer shopping experiences was Book Stacks Unlimited, an online bookstore created by Charles M. Stack in 1992. Stack’s store began as a dial-up bulletin board two years before Amazon was founded by Jeff Bezos. In 1994, Book Stacks Unlimited moved to the Internet as Books.com and was eventually acquired by Barnes & Noble.

**When was the first online transaction?**

The first online transaction was, by some reports, marijuana sold by Stanford students to MIT students via the Arpanet account at their artificial intelligence lab in 1972. However, the first online shopping transaction on the Internet took place some 22 years later. With the headline “The Internet is Open”, the August 12, 1994, issue of New York Times chronicled the sale between two friends of a Sting CD. The Times said, “The team of young cyberspace entrepreneurs celebrated what was apparently the first retail transaction on the Internet using a readily available version of powerful data encryption software designed to guarantee privacy.”

**Ecommerce Timeline**

**1960 – 1982**

Paving the way for electric commerce was the development of the [Electronic Data Interchange](https://www.covalentworks.com/what-is-edi.asp)(EDI). EDI replaced traditional mailing and faxing of documents with a digital transfer of data from one computer to another.

Trading partners could transfer orders, invoices and other business transactions using a data format that met the ANSI ASC X12, the predominant set of standards in North America.

Once an order is sent, it is then examined by a VAN (Value-Added Network) and finally directed to the recipient’s order processing system. EDI allowed the transfer of data seamlessly without any human intervention.

[Michael Aldrich](https://www.aldricharchive.com/inventors_story.html), an English inventor, innovator and entrepreneur is credited with developing the predecessor to online shopping. The idea came about during a stroll with his wife and Labrador when Aldrich lamented about their weekly supermarket shopping expedition. This conversation sparked an idea to hook a television to their supermarket to deliver the groceries. Immediately after the discussion Aldrich quickly planned and implemented his idea.

In 1979 Aldrich connected a television set to a transaction processing computer with a telephone line and created what he coined, “teleshopping,” meaning shopping at a distance.

**1982 – 1990**

It was apparent from the beginning that B2B online shopping would be commercially lucrative but B2C would not be successful until the later widespread use of PC’s and the World Wide Web, also known as, the Internet. In 1982, France launched the precursor to the Internet called, [Mintel.](https://www.teleread.com/net-related-tooks-from-search-engines-to-blogware/olpc-lessons-from-minitel/)

The online service used a Videotext terminal machine that was accessed through telephone lines. The Mintel was free to telephone subscribers and connected millions of users to a computing network.

By 1999, over 9 million Mintel terminals had been distributed and were connecting approximately 25 million users in this interconnected network of machines. The Mintel system peaked in 1991 and slowly met its demise after the success of the Internet 3 years later. Eventually, in 2011, France Telecom announced its [shutdown of the Mintel service system](https://www.pcmag.com/article2/0,2817,2389164,00.asp#fbid=-1XmaaLOFxc). Sadly, it had not become what it had hoped to be, the Internet.

**The World Wide Web Arrives, the early 90’s**

In 1990 Tim Berners Lee, along with his friend Robert Cailliau, published a proposal to build a “Hypertext project” called, “Worldwide Web.” The inspiration for this project was modeled after the Dynatex SGML reader licensed by CERN.

That same year, Lee, using a Next computer created the first web server and wrote the first web browser. Shortly thereafter, he went on to debut the web on Aug. 6, 1991 as a publicly available service on the Internet. When Berner’s Lee decided he would take on the task of marrying hypertext to the Internet, in doing that, the process led to him developing URL, HTML and HTTP.

When the National Science Foundation lifted its restrictions on commercial use of the NET in 1991, the Internet and online shopping saw remarkable growth. In September 1995, the NSF began charging a fee for registering domain names. 120,000 registered domain names were present at that time and within 3 years that number grew to beyond 2 million. By this time, NSF’s role in the Internet came to an end and a lot of the oversight shifted to the commercial sector.

The 1992 book, [Future Shop: How Technologies Will Change The Way We Shop And What We Buy](https://bookstore.iuniverse.com/Products/SKU-000061062/FUTURE-SHOP.aspx), provided insight and predictions on the future of consumerism. An overview of the book explains:

*“*For hundreds of years the marketplace has been growing more complex and more confusing for consumers to navigate. Published in 1992, long before the Internet became a household word. Future Shop argued that new information technologies, combined with innovative public policies, could help consumers overcome that confusion. A prescient manifesto of the coming revolution in e-commerce, Future Shop’s vision of consumer empowerment still resonates today.”

From the beginning, there were many hesitations and concerns with online shopping but the development of a security protocol – the [Secure Socket Layers](https://www.evsslcertificate.com/ssl/ssl-history.html) (SSL) – encryption certificate by Netscape in 1994 provided a safe means to transmit data over the Internet. Web browsers were able to check and identify whether a site had an authenticated SSL certificate and based on that, could determine whether or not a site could be trusted.

Now, SSL encryption protocol is a vital part of web security and version 3.0 has become the standard for most web servers today.

**Marketplaces, Payments and Mobile, Mid ‘90s to Present**

**Marketplaces Emerge**

From the mid-nineties there were major advancements in the commercial use of the Internet. One of the first ecommerce sites was Amazon which launched in 1995 as an online bookstore and has grown to be the largest online retailer in the world. Brick-and-mortar bookstores were limited to about 200,000 titles and Amazon, being an online only store, without physical limitations was able to offer exponentially more products to the shopper.

Amazon’s range now includes not only books but music and video downloads, electronics, apparel, furniture, food, and toys.

Amazon was one of the first online retailers to add user reviews with a rating scale to for products. Customer reviews are now considered the most effective social media tactic for driving sales.

Other ecommerce marketplace success stories include EBay, an online auction site that debuted in 1995 and Etsy, which launched in 2005 and by Q2 2018 saw gross merchandise sales total $901.7 million globally.

The late 1990s also saw new ecommerce platform options for merchants. Miva’s first catalogue-based ecommerce product was launched in 1997, achieving wide distribution in the late 1990s.

In 2005, Amazon announces the creation of Amazon Prime, a membership offering free two-day shipping within the contiguous United States on all eligible purchases for a flat annual fee. The membership quickly became popular, putting pressure on other merchants to offer fast and inexpensive shipping options. In 2016, Stamps.com acquires Shipping Easy for $50 million, after previously purchasing Ship Station and Ship Works in 2014.

**Online Payments Evolve**

Global ecommerce company, PayPal, began its services in 1998 and currently operates in 202 markets. The company is an acquired bank that performs payment processing for online vendors, auction sites, and other commercial users. They allow their customers to send, receive and hold funds in 24 currencies worldwide. Currently, PayPal manages more than 244 million accounts, more than 100 million of them active.

As more and more people began doing business online, a need for secure communication and transactions became apparent. In 2004, the [Payment Card Industry Security Standards Council](https://www.pcisecuritystandards.org/) (PCI) was formed to ensure businesses were meeting compliance with various security requirements.

The organization was created for the development, enhancement, storage, dissemination and implementation of security standards for account data protection.

In 2010, payments platform Square allowed small businesses to accept debit and credit cards on mobile devices. By Q4 2017, Square’s gross payment volume was $17.9 billion.

**Mobile Expands**

In 2001, Amazon.com launched its first mobile commerce site. Mobile commerce gained speed over the next two decades, as more users purchase from the palm of their hand. More than a third of U.S. e-commerce sales were made on a mobile device in 2017, according to [E-Marketer Inc.](https://www.emarketer.com/) The research firm expects mobile sales to increase by a third in 2018 – reaching more than $200 billion – and estimates that by 2020 mobile sales will top 50%. Both consumers and business buyers turn to mobile devices for product research and coupons, with engagement through social media becoming increasingly popular. While business buyers expect consumer features such as a responsive design that travels from desktop to laptop to tablet to phone with consistent features, [B2B shoppers demand an even stronger focus on finding products details, pricing, and help fast.](https://www.miva.com/blog/b2b-buyers-expect-mobile-experience/)

**Step Into the Future of Ecommerce**

From the ’60s until today, the Internet, ecommerce, and consumer trends continue to evolve. By Q1 2018, total ecommerce retails sales in the US were $123.7 billion, according to the U.S. Commerce Department.

### ****How to Compete – And Thrive – In the Modern Marketplaces****

Yes, merchants can thrive alongside Amazon. After looking the giant squarely in the face, independent merchants found there were [many things they could do that Amazon couldn’t](https://www.miva.com/blog/10-things-you-can-do-that-amazon-cant/), including building a memorable brand experience and gaining customers for life. But Amazon has [led by example](https://www.miva.com/blog/10-things-you-can-do-that-amazon-cant/) in many ways, including customer experience and [innovation](https://www.miva.com/blog/amazon-effect-innovation/).

### ****Buyer Preferences Driving Rapid Disruption****

What will shoppers want next? That’s what merchants online and off is trying to figure out. For online merchants, the Holy Grail is to find where they have a genuine advantage over brick-and-mortar stores. For example, new online product visualizers can showcase products in a variety of real-life situations, [at home and in offices](https://www.miva.com/blog/beating-brick-mortar-retail-experience-product-visualizers/).

And then there are Millennia’s. In the digital age, there are not many places left to hide. A site that lags or struggles to load loses millennial visitors faster than you can say ‘avocado toast’. As part of the self-directed buying process, Millennia’s will typically shop around a bit and read reviews before making a purchase decision. To be successful, sellers need to monitor comments and listen to what shoppers say about their products and customer service. A site with an outdated design communicates outdated processes, products, and prices to millennial buyers.

So, what’s next? The only thing that is certain is that more change is on the horizon. Your online store needs to be flexible enough to adapt to the changes

**Tools/Technology:**

**Html:**

Hypertext Markup Language (HTML) is the standard [markup language](https://en.wikipedia.org/wiki/Markup_language) for documents designed to be displayed in a [web browser](https://en.wikipedia.org/wiki/Web_browser). It can be assisted by technologies such as [Cascading Style Sheets](https://en.wikipedia.org/wiki/Cascading_Style_Sheets) (CSS) and [scripting languages](https://en.wikipedia.org/wiki/Scripting_language) such as [JavaScript](https://en.wikipedia.org/wiki/JavaScript).

[Web browsers](https://en.wikipedia.org/wiki/Web_browser) receive HTML documents from a [web server](https://en.wikipedia.org/wiki/Web_server) or from local storage and [render](https://en.wikipedia.org/wiki/Browser_engine) the documents into multimedia web pages. HTML describes the structure of a web page [semantically](https://en.wikipedia.org/wiki/Semantic_Web) and originally included cues for the appearance of the document.

[HTML elements](https://en.wikipedia.org/wiki/HTML_element) are the building blocks of HTML pages. With HTML constructs, [images](https://en.wikipedia.org/wiki/HTML_element#Images_and_objects) and other objects such as [interactive forms](https://en.wikipedia.org/wiki/Fieldset) may be embedded into the rendered page. HTML provides a means to create [structured documents](https://en.wikipedia.org/wiki/Structured_document) by denoting structural [semantics](https://en.wikipedia.org/wiki/Semantics) for text such as headings, paragraphs, lists, [links](https://en.wikipedia.org/wiki/Hyperlink), quotes and other items. HTML elements are delineated by *tags*, written using [angle brackets](https://en.wikipedia.org/wiki/Bracket#Angle_brackets). Tags such as <img /> and <input /> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

**CSS:**

Cascading Style Sheets (CSS) is a [style sheet language](https://en.wikipedia.org/wiki/Style_sheet_language) used for describing the [presentation](https://en.wikipedia.org/wiki/Presentation_semantics) of a document written in a [markup language](https://en.wikipedia.org/wiki/Markup_language) like [HTML](https://en.wikipedia.org/wiki/HTML). CSS is a cornerstone technology of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web), alongside HTML and [JavaScript](https://en.wikipedia.org/wiki/JavaScript).

CSS is designed to enable the separation of presentation and content, including [layout](https://en.wikipedia.org/wiki/Page_layout), [colors](https://en.wikipedia.org/wiki/Color), and [fonts](https://en.wikipedia.org/wiki/Typeface). This separation can improve content [accessibility](https://en.wikipedia.org/wiki/Accessibility), provide more flexibility and control in the specification of presentation characteristics, enable multiple [web pages](https://en.wikipedia.org/wiki/Web_page) to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or [screen reader](https://en.wikipedia.org/wiki/Screen_reader)), and on [Braille-based](https://en.wikipedia.org/wiki/Braille_display) tactile devices. CSS also has rules for alternate formatting if the content is accessed on a [mobile device](https://en.wikipedia.org/wiki/Mobile_device).

The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

The CSS specifications are maintained by the [World Wide Web Consortium](https://en.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C).

**JavaScript:**

JavaScript  often abbreviated as JS, is a [high-level](https://en.wikipedia.org/wiki/High-level_programming_language), [interpreted](https://en.wikipedia.org/wiki/Interpreted_language) [scripting language](https://en.wikipedia.org/wiki/Scripting_language) that conforms to the [ECMA Script](https://en.wikipedia.org/wiki/ECMAScript) specification. JavaScript has [curly-bracket syntax](https://en.wikipedia.org/wiki/List_of_programming_languages_by_type#Curly-bracket_languages), [dynamic typing](https://en.wikipedia.org/wiki/Dynamic_typing), [prototype-based](https://en.wikipedia.org/wiki/Prototype-based_programming) [object-orientation](https://en.wikipedia.org/wiki/Object-oriented_programming), and [first-class functions](https://en.wikipedia.org/wiki/First-class_function).

Alongside [HTML](https://en.wikipedia.org/wiki/HTML) and [CSS](https://en.wikipedia.org/wiki/CSS), JavaScript is one of the core technologies of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web). JavaScript enables interactive [web pages](https://en.wikipedia.org/wiki/Web_page) and is an essential part of [web applications](https://en.wikipedia.org/wiki/Web_application). The vast majority of [websites](https://en.wikipedia.org/wiki/Website) use it, and major [web browsers](https://en.wikipedia.org/wiki/Web_browser) have a dedicated [JavaScript engine](https://en.wikipedia.org/wiki/JavaScript_engine) to execute it.

As a multi-paradigm language, JavaScript supports [event-driven](https://en.wikipedia.org/wiki/Event-driven_programming), [functional](https://en.wikipedia.org/wiki/Functional_programming), and [imperative](https://en.wikipedia.org/wiki/Imperative_programming) (including [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) and [prototype-based](https://en.wikipedia.org/wiki/Prototype-based_programming)) [programming styles](https://en.wikipedia.org/wiki/Programming_paradigm). It has [APIs](https://en.wikipedia.org/wiki/Application_programming_interface) for working with text, [arrays](https://en.wikipedia.org/wiki/Array_data_type), dates, [regular expressions](https://en.wikipedia.org/wiki/Regular_expression), and the [DOM](https://en.wikipedia.org/wiki/Document_Object_Model),

Initially only implemented [client-side](https://en.wikipedia.org/wiki/Client-side) in web browsers, JavaScript engines are now embedded in many other types of host software, including [server-side](https://en.wikipedia.org/wiki/Server-side) in web servers and databases, and in non-web programs such as word processors and [PDF](https://en.wikipedia.org/wiki/Portable_Document_Format) software, and in runtime environments that make JavaScript available for writing mobile and desktop applications, including desktop widgets.

The terms [*Vanilla*](https://en.wikipedia.org/wiki/Vanilla_software)*JavaScript* and *Vanilla JS* refer to JavaScript not extended by any frameworks or additional libraries. Scripts written in Vanilla JS are plain JavaScript code.

**Firebase:**

The Firebase Real-time Database is a cloud-hosted No SQL database that lets you store and sync between your users in real-time. ... Real-time syncing makes it easy for your users to access their data from any device, be it web or mobile. Real-time Database also helps your users collaborate with one another.

**Mern Stack:**

**Mongo DB:**

Mongo DB is a document database with the scalability and flexibility that you want with the querying and indexing that you need.

Mongo DB **stores data in flexible, JSON-like documents**, meaning fields can vary from document to document and data structure can be changed over time

The document model **maps to the objects in your application code**, making data easy to work with

**Ad hoc queries, indexing, and real time aggregation** provide powerful ways to access and analyze your data

Mongo DB is a **distributed database at its core**, so high availability, horizontal scaling, and geographic distribution are built in and easy to use

Mongo DB is **free to use**. Versions released prior to October 16, 2018 are published under the AGPL. All versions released after October 16, 2018, including patch fixes for prior versions, are published under the [Server Side Public License (SSPL) v1](https://www.mongodb.com/licensing/server-side-public-license).

**Express:**

Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.

Let's see some of the core features of Express framework:

* It can be used to design single-page, multi-page and hybrid web applications.
* It allows to setup middleware’s to respond to HTTP Requests.
* It defines a routing table which is used to perform different actions based on HTTP method and URL.
* It allows to dynamically rendering HTML Pages based on passing arguments to templates.

**Why use Express**

* Ultra fast I/O
* Asynchronous and single threaded
* MVC like structure
* Robust API makes routing easy

**React:**

React is a declarative, efficient, and flexible JavaScript library for building user interfaces. It lets you compose complex UIs from small and isolated pieces of code called “components”.

* **Jsx:**

JSX is an XML/HTML-like syntax used by React that extends ECMA Script so that XML/HTML-like text can co-exist with JavaScript/React code. Unlike the past, instead of putting JavaScript into HTML, JSX allows us to put HTML into JavaScript.

* **Virtual Dom**

The virtual DOM (VDOM) is a programming concept where an ideal, or “virtual”, representation of a UI is kept in memory and synced with the “real” DOM by a library such as React DOM. This process is called [reconciliation](https://reactjs.org/docs/reconciliation.html).

This approach enables the declarative API of React: You tell React what state you want the UI to be in, and it makes sure the DOM matches that state. This abstracts out the attribute manipulation, event handling, and manual DOM updating that you would otherwise have to use to build your app.

**Node Js:**

Node.js (Node) is an open source development platform for executing JavaScript code server-side. Node.js is intended to run on a dedicated HTTP server and to employ a single thread with one process at a time. Node.js applications are event-based and run asynchronously

**Features**

* Node.js can generate dynamic page content
* Node.js can create, open, read, write, delete, and close files on the server
* Node.js can collect form data
* Node.js can add, delete, modify data in your database

**Chapter No 3**

**System Requirement Analysis**

This chapter provides a brief overview of functional requirements and non-functional requirement. Functional requirements are those which are related to the technical functionality of the system. Non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system in particular conditions, rather than specific behaviors.

**Functional Requirements**

**Responsive Web Design**

Mobile devices are an important part of everyday life, especially since consumers spend [more than five hours a day](https://www.bluecorona.com/blog/mobile-marketing-statistics) on their smartphone. People want to be able to browse the internet and access the information they need from their favorite devices without any headaches or second-guessing. Responsive web design automatically adapts a web store to the mobile device that’s being used without distorting the view.

**Attractive & Easy-to-Navigate**

Attraction equals persuasion. A good-looking online store supports customers throughout each step of the buying journey: from enticing a shopper to visit the webpage, to inspiring them to explore the store and its products, to finally charming them into purchasing.

According to [Blue Corona](https://www.bluecorona.com/blog/20-web-design-facts-small-business-owners/), you have 10 seconds to leave an impression on an online visitor and tell them what they’ll get out of your website and company. If you can’t capture their attention within that time span, then more often than not they’ll leave and won’t return.

**Speedy Page Loading**

Online shoppers want convenience, and they won’t stick around if a page takes forever to load. The online shopping experience should be seamless and easy, especially since [40% of people will abandon a website](https://neilpatel.com/blog/loading-time/) that takes more than three seconds to load. In fact, a one second delay in page response can lead to a 7%decrease in conversions.

**Product Details**

Online shopping can be a leap of faith — a shopper can’t physically pick up and feel merchandise before they buy, like they would in a brick-and-mortar store. In fact, a study found that [62% of shoppers](https://www.retaildive.com/news/why-most-shoppers-still-choose-brick-and-mortar-stores-over-e-commerce/436068/) choose brick-and-mortar stores over online options solely because they want to be able to see, touch, and feel the product.

But shoppers still want to familiarize themselves with what your ecommerce platform has to offer. Include informative features on product pages such as high-res photos; in-depth product descriptions; zoom-in and 360-degree rotation capabilities; the ability to view the product in different colors; and product videos. By providing as much product detail and information as possible, you decrease the chance of returned items and increase the likelihood of satisfied customers.

**Shopping Cart**

A shopping cart isn’t just a place for customers to store items pending checkout — it’s another important touch point that can deliver additional conversion opportunities. Cross-selling at checkout helps target customers more precisely with products they are likely to be interested in based on the items already in their cart — which can in turn increase the purchase total.

Total visibility into the products awaiting purchase and their costs (including shipping and sales tax) keeps customers aware and decreases the likelihood of abandonment. [Bayard](https://baymard.com/lists/cart-abandonment-rate) states that on average approximately 70% of online shopping carts are abandoned.

**Simple Checkout**

Don’t let a complicated or lengthy checkout process make customers abandon their shopping cart at the last minute. [Eliminate friction with an easy checkout](https://home.bluesnap.com/snap-center/blog/13-steps-smooth-checkout-process/) that allows customers to quickly fill out only the most necessary information. Fields that the customers can fill out by choice on the form (such as signing up for a newsletter); pre-filling forms; a progress bar for multiple steps/pages; and the ability to store credit card information for repeat shoppers helps to minimize frustration.

**Non Functional Requirements**

### Performance Requirements

Any page of the application should not take more than 6 seconds to load on a DSL broadband connection.

The system may be throttled or slowed down on heavy loads to ensure service for everybody. By throttling is meant that certain functionality may be unavailable during heavy server load.

The application should be able to support 100 concurrent users without any performance degradation.

Although striving to have a 100% uptime, unless during a scheduled maintenance period, which will be relayed to the users of the site well in advance, problems may occur. No hesitation will be present when the need arises to move over to another server, should the current server be insufficient to provide a decent level of service consistently.

### Scalability Requirements

The system should be able to scale up to 500 concurrent users (if there is a need in the future) by installing additional hardware components.

### 

### Reliability/Availability Requirements

The system has to be online 24 hours a day, 7 days a week. There is no place for an extended downtime, especially when the project goes International, and time zones will control the traffic load.

The MTBF (if any) should not be less than 2 months.

In case of a failure that leads to a system outrage, the MTTR should not be more than 2 hours.

### Security Requirements

There needs to be clearly defined roles of the users. These roles are 'customer’ and 'administrator'. Each person that goes to the system's website will be required to register if they want to do more than just read / browse site content.

A secure server will be required to ensure confidentiality of customer’s credit card and other details

Because of the different roles, passwords and user accounts must be implemented properly. It should be difficult to gain access to the site in an illegal manner.

### Usability

The user interface of the system should be very user friendly.

It should not take more than 120 seconds for a new user to register for an account.

It should not take more than 90 seconds for a registered user to place an order.

**Feasibility Studies**

Feasibility study is, to study about the problem and carries out the best system which meets all the requirements which is needed or necessary. Feasibility study is determining the financially and technically study to develop the system. They involve in analysis the problem and collect all the information about it.

**Technical Feasibility**

Specifying equipment and system that will successfully satisfy the requirement.

The facility to produce outputs in a given time.

Response time under certain conditions.

Ability to process a generate repots at a particular speed.

In technical feasibility, configuration of the system is given more importance than the actual makes of hardware. What speeds of input and output should be achieved at particular quality of printing.

**Operational Feasibility**

* What changes will be brought with the system?
* What organizational structures are disturbed?
* What new skills will be required? Do the existing staff members have these skills? If not, can they be trained in due course of time?

In this operational feasibility is operate the system a person should able to knowing about the system and easily understand to it they should be skilled full if system generate any error the person how controls it should be able to remove it or manage.

**Methodologies**

Software project management begins with a set of activities that are collectively called software project planning. Before the project can begin, the manager and the software team must estimate the work to be done, the resources that will be required, and the time that will elapse from start to finish. Whenever estimates are made, we look into the future that accepts some degree of uncertainty as a matter of course.

There are many methodologies available to develop a system. The methodologies have the common phases like:

* Requirements
* Analysis
* Design
* Coding
* Testing
* Implementation

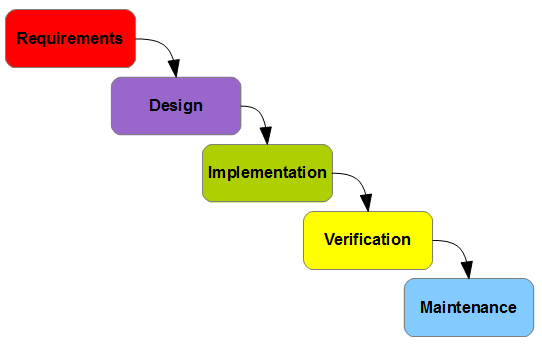
**Software Development Models**

The software development models are the various processes or methodologies that are being selected for the development of the project depending on the project's aims and goals. There are many development life cycle models that have been developed in order to achieve different required objectives.

* Waterfall model
* Spiral Model
* Iterative Model

**Waterfall model**

Waterfall Model. The Waterfall model is the earliest SDLC approach that was used for software development. The waterfall Model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete.



The five stages above are as follows:

1. **Requirement Analysis:**

What - The systems services, constraints and goals are established by consultation with system users. They are then defined in detail and serve as a system specification.

1. **Design:**

How The system design process partitions the requirements to either hardware of software systems. It establishes and overall system architecture. Software design involves fundamental system abstractions and their relationships.

1. **Implementation:**

How – During this stage the software design is realized as a set of programs or program units. Unit testing involves verifying that each unit meets its specifications.

1. **Verification or Integration:**

The individual program unit or programs are integrated and tested as a complete system to ensure that the software requirements have been met. After testing, the software system is delivered to the customer.

1. **Maintenance:**

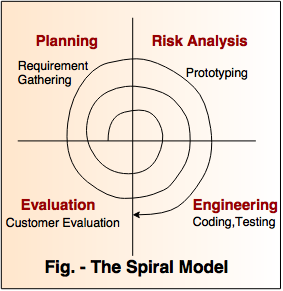
Normally this is the longest phase of the software life cycle. The system is installed and put into practical use. Maintenance involves correcting errors which were not discovered in earlier stages of the life-cycle, improving the implementation of system units and enhancing the system’s services as new requirements are discovered.

## Spiral Model

In principle, the result of each phase is one or more documents which are approved. No phase is complete until the documentation for that phase has been completed and products of that phase have been approved. The following phase should not start until the previous phase has finished.

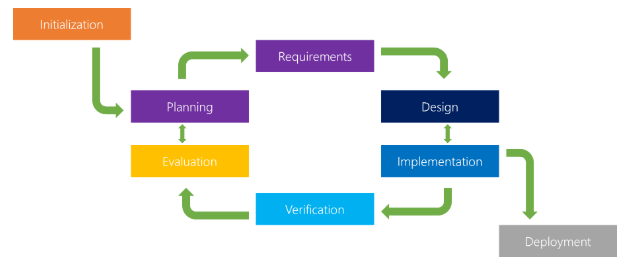
This model was developed by Barry Boehm. The main idea of this model is to avert risk as there is always an element of risk in development of software. For example, key personnel may resign at a critical juncture, the manufacturer of the software development may go bankrupt, etc.

In its simplified form, the Spiral Model is Waterfall model plus risk analysis. In this case each stage is preceded by identification of alternatives and risk analysis and is then followed by evaluation and planning for the next phase. If risks cannot be resolved, project is immediately terminated. This is depicted in the following diagram.



**Iterative Model**

A process for arriving at a decision or desired result by repeating rounds of analysis or a cycle of operations. The objective is to bring the desired decision or result closer to discovery with each repetition (iteration). The iterative process can be used where the decision is not easily recoverable or where the consequences of revocation could be costly.



**Reasons For Choosing Iterative Model**

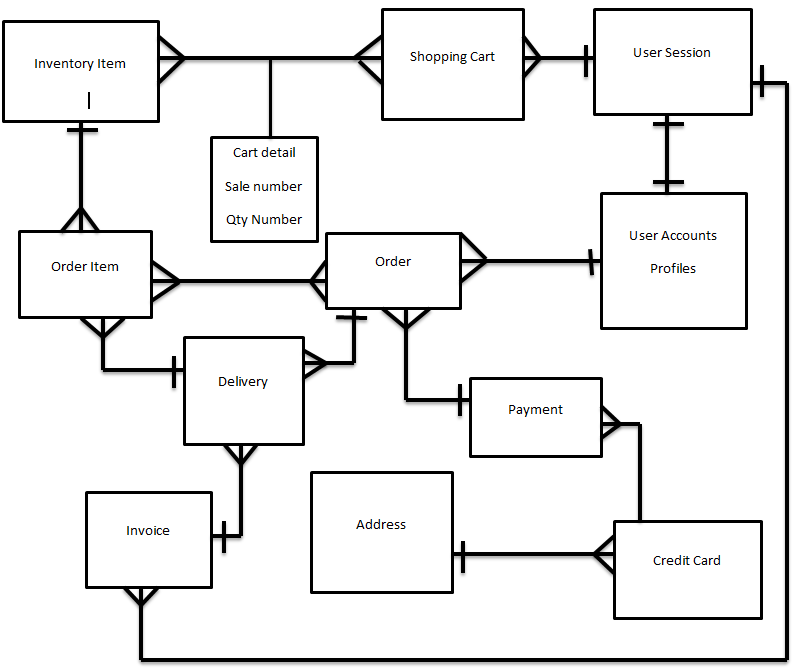
I use this methodology because any change in requirements at any condition then I should able to improve the product as according to new requirements.

* In iterative model we are building and improving the product step by step.
* In iterative model we can get the reliable Customer feedback.
* In iterative model Major requirements must be defined however, some details can evolve with time.
* It allows developers to break down the task of developing a system into a series of smaller tasks.
* In iterative model we are building and improving the product step by step. Hence we can track the defects at early stages. This avoids the downward flow of the defects.
* In iterative model we can get the reliable user feedback. When presenting sketches and blueprints of the product to users for their feedback, we are effectively asking them to imagine how the product will work.

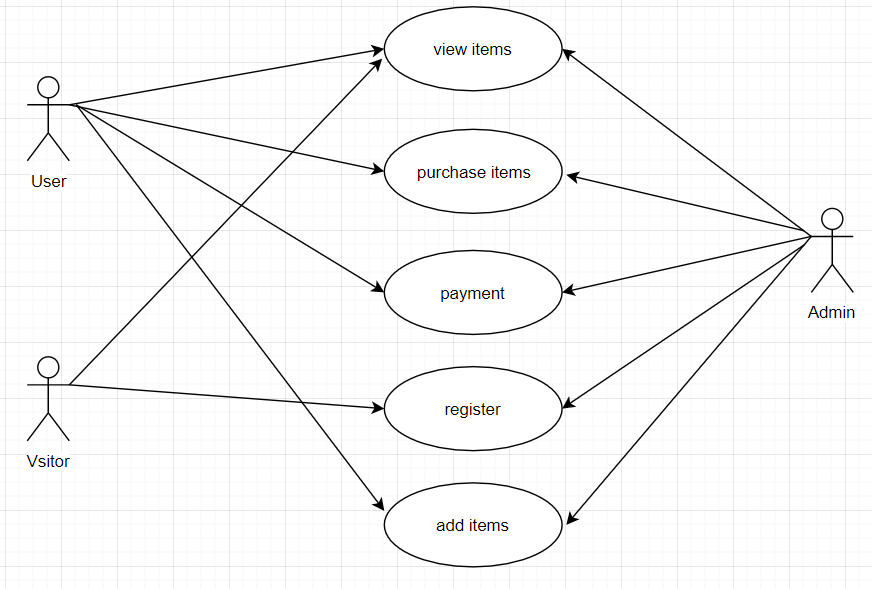
**Chapter No 4**

**System Design**

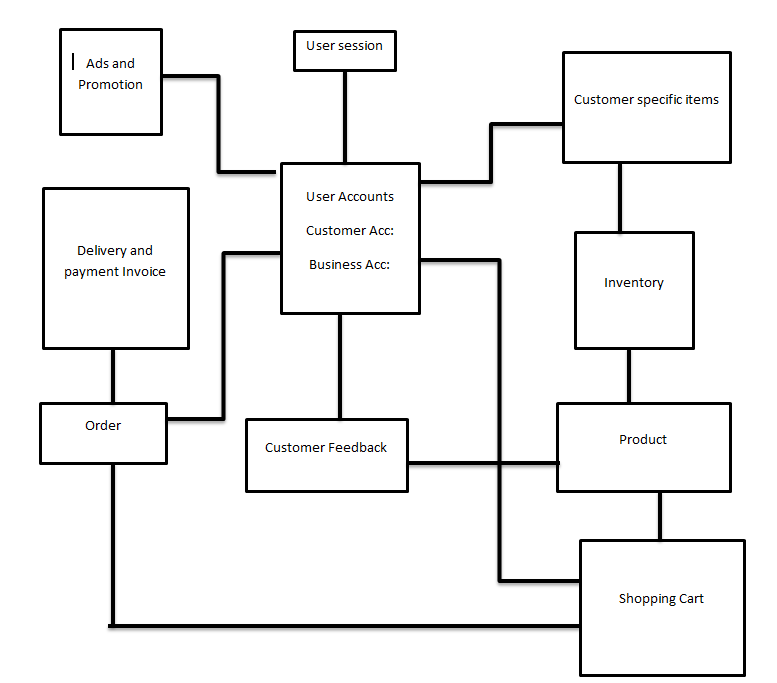
**Relational Diagram**



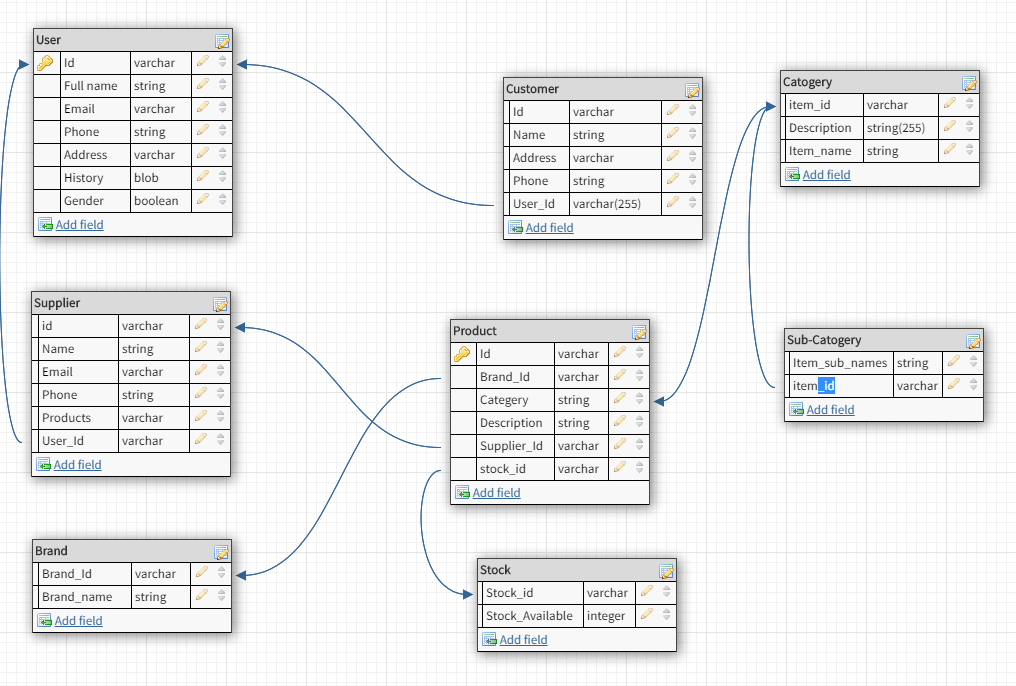
**Use Case Diagram**



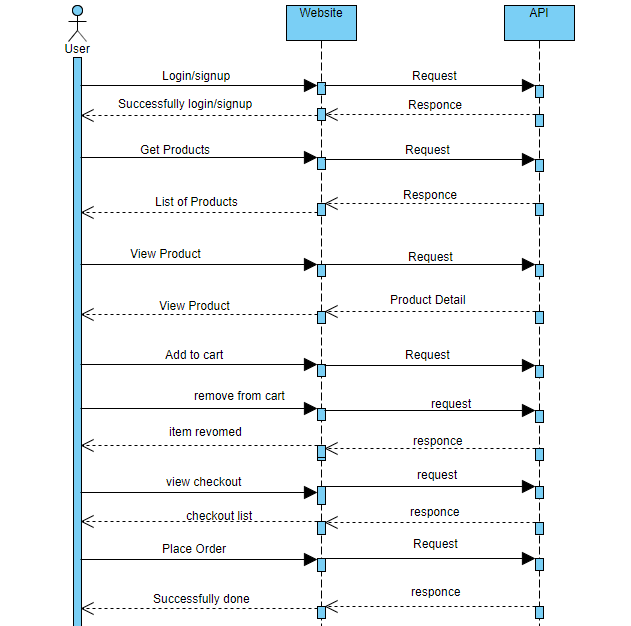
**Schema:-**



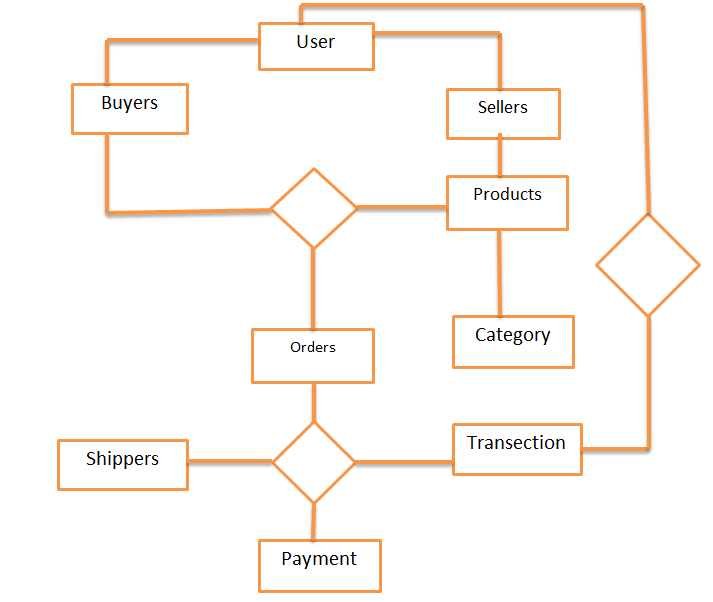
**Database Design:-**

****

**Sequence Diagram**



**Entity Relation Diagram (ERD)**



**Chapter No 5**

**Testing & Implementation**

**Overview**

Software Testing is evaluation of the software against requirements gathered from users and system specifications. Testing is conducted at the phase level in software development life cycle or at module level in program code. Software testing comprises of Validation and Verification.

## Software Validation

Validation is process of examining whether or not the software satisfies the user requirements. It is carried out at the end of the SDLC. If the software matches requirements for which it was made, it is validated.

* Validation ensures the product under development is as per the user requirements.
* Validation answers the question – "Are we developing the product which attempts all that user needs from this software?"
* Validation emphasizes on user requirements.

## Software Verification

Verification is the process of confirming if the software is meeting the business requirements, and is developed adhering to the proper specifications and methodologies.

* Verification ensures the product being developed is according to design specifications.
* Verification answers the question– "Are we developing this product by firmly following all design specifications?"

Verifications concentrates on the design and system specification.

## Manual vs. Automated Testing

Testing can either be done manually or using an automated testing tool:

**Manual**

This testing is performed without taking help of automated testing tools. The software tester prepares test cases for different sections and levels of the code, executes the tests and reports the result to the manager.

Manual testing is time and resource consuming. The tester needs to confirm whether or not right test cases are used. Major portion of testing involves manual testing.

**Automated**

This testing is a testing procedure done with aid of automated testing tools. The limitations with manual testing can be overcome using automated test tools.

A test needs to check if a webpage can be opened in Internet Explorer. This can be easily done with manual testing. But to check if the web-server can take the load of 1 million users, it is quite impossible to test manually.

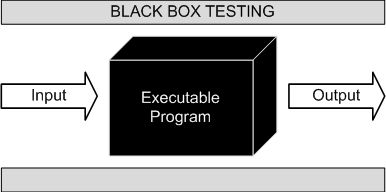
There are software and hardware tools which helps tester in conducting load testing, stress testing, regression testing.

**Testing Methods**

There are different methods which can be used for system testing. This chapter briefly describes those methods.

**Black-Box Testing**

It is carried out to test functionality of the program. It is also called ‘Behavioral’ testing. The tester in this case, has a set of input values and respective desired results. On providing input, if the output matches with the desired results, the program is tested ‘ok’, and problematic otherwise.



In this testing method, the design and structure of the code are not known to the tester, and testing engineers and end users conduct this test on the software.

**Black-Box Testing Techniques:**

* **Equivalence class**

The input is divided into similar classes. If one element of a class passes the test, it is assumed that all the class is passed.

* **Boundary values**

The input is divided into higher and lower end values. If these values pass the test, it is assumed that all values in between may pass too.

* **Cause-effect graphing**

In both previous methods, only one input value at a time is tested. Cause (input) – Effect (output) is a testing technique where combinations of input values are tested in a systematic way.

* **Pair-wise Testing**

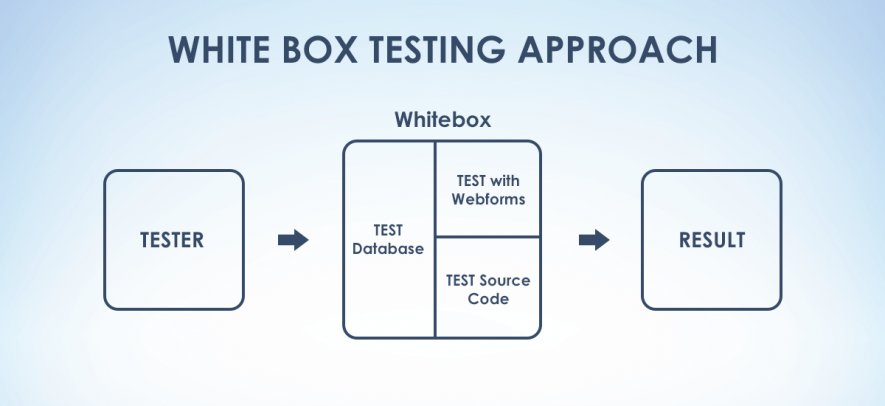
The behavior of software depends on multiple parameters. In pairwise testing, the multiple parameters are tested pair-wise for their different values.

* **State-based testing**

The system changes state on provision of input. These systems are tested based on their states and input.

### White-Box Testing

### It is conducted to test program and its implementation, in order to improve code efficiency or structure. It is also known as ‘Structural’ testing.



In this testing method, the design and structure of the code are known to the tester. Programmers of the code conduct this test on the code.

**The below are some White-box testing techniques:**

* **Control-flow testing**

The purpose of the control-flow testing to set up test cases which covers all statements and branch conditions. The branch conditions are tested for both being true and false, so that all statements can be covered.

* **Data-flow testing**

This testing technique emphasis to cover all the data variables included in the program. It tests where the variables were declared and defined and where they were used or changed.

**Grey Box Testing**

Grey Box testing is a technique to test the application with limited knowledge of the internal workings of an application. In software testing, the term the more you know the better carries a lot of weight when testing an application.

Mastering the domain of a system always gives the tester an edge over someone with limited domain knowledge. Unlike black box testing, where the tester only tests the application's Customer interface, in grey box testing, the tester has access to design documents and the database.



## Testing Levels

Testing itself may be defined at various levels of SDLC. The testing process runs parallel to software development. Before jumping on the next stage, a stage is tested, validated and verified.

Testing separately is done just to make sure that there are no hidden bugs or issues left in the software. Software is tested on various levels –

### Unit Testing

While coding, the programmer performs some tests on that unit of program to know if it is error free. Testing is performed under white-box testing approach. Unit testing helps developers decide that individual units of the program are working as per requirement and are error free.

### Integration Testing

Even if the units of software are working fine individually, there is a need to find out if the units if integrated together would also work without errors. For example, argument passing and data updating etc.

### System Testing

The software is compiled as product and then it is tested as a whole. This can be accomplished using one or more of the following tests:

* **Functionality testing**

Tests all functionalities of the software against the requirement.

* **Performance testing**

This test proves how efficient the software is. It tests the effectiveness and average time taken by the software to do desired task. Performance testing is done by means of load testing and stress testing where the software is put under high user and data load under various environment conditions.

* **Security & Portability**

These tests are done when the software is meant to work on various platforms and accessed by number of persons.

**Acceptance Testing**

When the software is ready to hand over to the customer it has to go through last phase of testing where it is tested for user-interaction and response. This is important because even if the software matches all user requirements and if user does not like the way it appears or works, it may be rejected.

* **Alpha testing**

The team of developer themselves perform alpha testing by using the system as if it is being used in work environment. They try to find out how user would react to some action in software and how the system should respond to inputs.

* **Beta testing**

After the software is tested internally, it is handed over to the users to use it under their production environment only for testing purpose. This is not as yet the delivered product. Developers expect that users at this stage will bring minute problems, which were skipped to attend.

### Regression Testing

Whenever a software product is updated with new code, feature or functionality, it is tested thoroughly to detect if there is any negative impact of the added code. This is known as regression testing.

## Testing Documentation

Testing documents are prepared at different stages -

### Before Testing

Testing starts with test cases generation. Following documents are needed for reference

* **SRS document**

Functional Requirements document

* **Test Policy document**

This describes how far testing should take place before releasing the product.

* **Test Strategy document**

This mentions detail aspects of test team, responsibility matrix and rights/responsibility of test manager and test engineer.

* **Traceability Matrix document**

This is SDLC document, which is related to requirement gathering process. As new requirements come, they are added to this matrix. These matrices help testers know the source of requirement. They can be traced forward and backward.

### While Being Tested

The following documents may be required while testing is started and is being done:

* **Test Case document**

This document contains list of tests required to be conducted. It includes Unit test plan, Integration test plan, System test plan and Acceptance test plan.

* **Test description**

This document is a detailed description of all test cases and procedures to execute them.

* **Test case report**

This document contains test case report as a result of the test.

* **Test logs**

This document contains test logs for every test case report.

### After Testing

The following documents may be generated after testing:

* **Test summary**

This test summary is collective analysis of all test reports and logs. It summarizes and concludes if the software is ready to be launched. The software is released under version control system if it is ready to launch.

## 5.10 Testing vs. Quality Control, Quality Assurance and Audit

We need to understand that software testing is different from software quality assurance, software quality control and software auditing.

* **Software quality assurance**

These are software development process monitoring means, by which it is assured that all the measures are taken as per the standards of organization. This monitoring is done to make sure that proper software development methods were followed.

* **Software quality control**

This is a system to maintain the quality of software product. It may include functional and non-functional aspects of software product, which enhance the goodwill of the organization. This system makes sure that the customer is receiving quality product for their requirement and the product certified as ‘fit for use’.

* **Software audit**

This is a review of procedure used by the organization to develop the software. A team of auditors, independent of development team examines the software process, procedure, requirements and other aspects of SDLC. The purpose of software audit is to check that software and its development process, both conform standards, rules and regulations.