

1. INTRODUCTION TO TOOLHUB

1.1.Overview

Tool Hub is a web based B2C based system where user can download or buy software and tools at one platform. It is a form of electronic-commerce. In traditional way, user has to visit to store physically. Tool Hub is an attempt where user get all types of tools on a daily basis. We also aim to provide functionality like Modern UI, Different Themes, Shortcut Keys, and the system recommends the services for 24x7. Since the application is available in the smartphone, it is easily accessible and always available to smaller devices.

1.2.Background

Software and Tool based store was introduced about 40 years ago in its earliest form. Since this has helped countless business grows with the help of new technologies, improvements in internet. Many MNC company provided copies of their tools in their store. This was hard to manageable for Start-up Company at their earlier state.

1.3.Motivation

It gives a good exposure to your business and helps you to reach out to potential customers. Since most of the people prefer to shop online due to paucity of time, you can easily make more revenue. It is easy for users to manage for company. It will be a win-to-win situation for both parties.

1.4.Existing System

The current system has following feature below:-

- User visits the store.
- User has to wait for his/her turn.
- Description of items limited.
- Time consuming process.
- No feedback system.

1.5.Problems in existing system

The current system has the following problems: -

- Slow Response to Trends and Time-To-Market.
- Limited Customization Options.
- Flawed Security.
- Dependence on Default Technology.
- Less User-friendly.

1.6.Proposed system

In the proposed system, user will visit our website to download and buy software by using computer and mobiles, The Admin can insert products (software or tool). The website will be light weighted so that any low device can access it.

2. OBJECTIVES OF THE PROJECT

The main objective of Tool Hub are:

1. Design and develop an automated and user-friendly service for downloading the tools.
2. To purpose a modern technique for switching the theme using Tailwindcss.
3. To analysis the system using Mobile-first Approach.

3. SYSTEM STUDY

3.1. Feasibility Study

At this stage, the analyst estimates the urgency of the project and estimates the development cost. In feasibility analysis, we have to study the following:

1) **Technical Feasibility:** Technical feasibility is concerned with the availability of hardware and software required for the development of the system, to see compatibility and maturity of the technology proposed to be used and to see the availability of the required technical manpower to develop the system.

After the study we came to conclusion that we proceed further with the tools and development environment chosen by us. This was important in our case as we were working on two various phases of the department that will need to be integrated in future to make an extended system.

2) **Operational Feasibility:** Operational feasibility is all about problems that may arise during operations. There are two aspects related with this issue:

- What is the probability that the solution developed may not be put to use or may not work?

What is the inclination of the management and end users towards the solution? Though, there is very least possibility of management being averse to the solution, there a significant probability that the end users may not be interested in using the solution due to lack of training, insight etc.

3) **Economic Feasibility:** It is the measure of cost effectiveness of the project. The economic feasibility is nothing but judging whether the possible benefit of solving the problems is worthwhile or not at the feasibility study level, it is impossible to estimate the cost because member's requirements and alternative solutions have not been identified at this stage. However, when the specific requirements and solutions have been identified, the analyst weighs the cost and benefits of all solutions, this is called "cost benefit analysis.

Software Requirement Specification

The Software Requirements Specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional and behavioral description, an indication of performance requirements and design constraints, appropriate validation criteria, and other data pertinent to requirements. The proposed system has the following requirements:

- System needs store the new user information.

- System needs to keep information update and find them as per various queries.
- System needs to maintain record.
- System needs to keep the record of User and Products.
- System should be able to update and delete the products easily.

4. SYSTEM ANALYSIS

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information about the Tool hub Website to recommend improvements on the system. It is a problem-solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analyzed.

The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action. A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals.

The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with proposal. Preliminary study is the process of gathering and interpreting facts, using the information for further studies on the system. Preliminary study is problem solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies. In these studies, a rough figure of the system activities can be obtained, from which the decision about the strategies to be followed for effective system study and analysis can be taken.

4.1.Importance of the project

It may help collecting perfect management in details. In a very short time, the collection will be obvious, simple and sensible. It will help a person to know the management of passed year perfectly and vividly. It also helps in current all works relative to Tool hub Website. It will be also reduced the cost of collecting the management & collection procedure will go on smoothly.

Our project aims at Business process automation, le we have tried to computerize various processes of Tool hub Website.

- In computer system the person has to fill the various forms & number of copies of the forms can be easily generated at a time
- The system should allow new users to register online and generate membership card
- In system, user can download various tools at ease.
- To utilize resources in an efficient manner by increasing their productivity through automation.
- The system generates types of information that can be used various purpose
- It satisfies the user requirement.
- Be easy to understand by the user and operator.
- Be easy to operate.
- Have a good user interface.
- Be expandable.

4.2.Functional Requirements

Requirement analysis is a software engineering technique that is composed of the various tasks. that determine the needs or conditions that are to be met for a new or altered product, taking into consideration the possible conflicting requirements of the various users. Functional requirements are those requirements that are used to illustrate the internal working nature of the system, the description of the system, and explanation of each subsystem. It consists of what task the system should perform. the processes involved, which data should the system holds and the interfaces with the user. The functional requirements identified are:

Hardware requirements

Processor: Intel core i3 (Min.)

Ram: 512 MB and above.

Hard disk: 80 GB and above

Keyboard: Normal or Multimedia

Mouse: Compatible mouse

Software requirements

Front-end: HTML5, Tailwindcss, JS

Back-end : PHP, MySQL

Operating system: All version of windows

5. SYSTEM DESIGN

In this phase, a logical system is built which fulfils the given requirements. Design phase of software development deals with transforming the client's requirements into a logically working system. Normally, design is performed in the following in the following two steps:

Primary Design Phase:

In this phase, the system is designed at block level. The blocks are created on the basis of analysis done in the problem identification phase. Different blocks are created for different functions emphasis is put on minimizing the information flow between blocks. Thus, all activities which require more interaction are kept in one block.

Secondary Design Phase:

In the secondary phase the detailed design of every block is performed.

The general tasks involved in the design process are the following:

1. Design various blocks for overall system processes.
2. Design smaller, compact and workable modules in each block.
3. Design various database structures.
4. Specify details of programs to achieve desired functionality.
5. Design the form of inputs, and outputs of the system.
6. Perform documentation of the design.
7. System reviews.

User Interface Design

User Interface Design is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventually presentation of desired inputs and outputs. The overall flow of screens and messages is called a dialogue.

The following steps are various guidelines for User Interface Design:

1. The system user should always be aware of what to do next.
2. The screen should be formatted so that villous types of information, instructions and messages always appear in the same general display area.
3. Message, instructions or information should be displayed long enough to allow the system user to read them.
4. Use display attributes sparingly.

5. Default values for fields and answers to be entered by the user should be specified.
6. A user should not be allowed to proceed without correcting an error.
7. The system user should never get an operating system message or fatal error.

Preliminary Product Description:

The first step in the system development life cycle is the preliminary investigation to determine the feasibility of the system. The purpose of the preliminary investigation is to evaluate project requests. It is not a design study nor does it include the collection of details to describe the business system in all respect. Rather, it is the collecting of information that helps committee members to evaluate the merits of the project request and make an informed judgment about the feasibility of the proposed project.

Analysts working on the preliminary investigation should accomplish the following objectives:

- Clarify and understand the project request
- Determine the size of the project.
- Assess costs and benefits of alternative approaches.
- Determine the technical and operational feasibility of alternative approaches

5.1.SDLC

A software life cycle model (also termed process model) is a pictorial and diagrammatic representation of the software life cycle. A life cycle model represents all the methods required to make a software product transit through its life cycle stages. It also captures the structure in which these methods are to be undertaken.

In other words, a life cycle model maps the various activities performed on a software product from its inception to retirement. Different life cycle models may plan the necessary development activities to phases in different ways. Thus, no element which life cycle model is followed, the essential activities are contained in all life cycle models though the action may be carried out in distinct orders in different life cycle models. During any life cycle stage, more than one activity may also be carried out.

Stage1: Planning and requirement analysis

Requirement Analysis is the most important and necessary stage in SDLC.

The senior members of the team perform it with inputs from all the stakeholders and domain experts or SMEs in the industry.

Planning for the quality assurance requirements and identifications of the risks associated with the projects is also done at this stage.

Business analyst and Project organizer set up a meeting with the client to gather all the data like what the customer wants to build, who will be the end user, what is the objective of the product. Before creating a product, a core understanding or knowledge of the product is very necessary.

For Example, A client wants to have an application which concerns money transactions. In this method, the requirement has to be precise like what kind of operations will be done, how it will be done, in which currency it will be done, etc.

Stage2: Defining Requirements

Once the requirement analysis is done, the next stage is to certainly represent and document the software requirements and get them accepted from the project stakeholders.

This is accomplished through "SRS"- Software Requirement Specification document which contains all the product requirements to be constructed and developed during the project life cycle.

Stage3: Designing the Software

The next phase is about to bring down all the knowledge of requirements, analysis, and design of the software project. This phase is the product of the last two, like inputs from the customer and requirement gathering.

Stage4: Developing the project

In this phase of SDLC, the actual development begins, and the programming is built. The implementation of design begins concerning writing code. Developers have to follow the coding guidelines described by their management and programming tools like compilers, interpreters, debuggers, etc. are used to develop and implement the code.

Stage5: Testing

After the code is generated, it is tested against the requirements to make sure that the products are solving the needs addressed and gathered during the requirements stage.

During this stage, unit testing, integration testing, system testing, acceptance testing are done.

Stage6: Deployment

Once the software is certified, and no bugs or errors are stated, then it is deployed.

Then based on the assessment, the software may be released as it is or with suggested enhancement in the object segment. After the software is deployed, then its maintenance begins.

Stage7: Maintenance

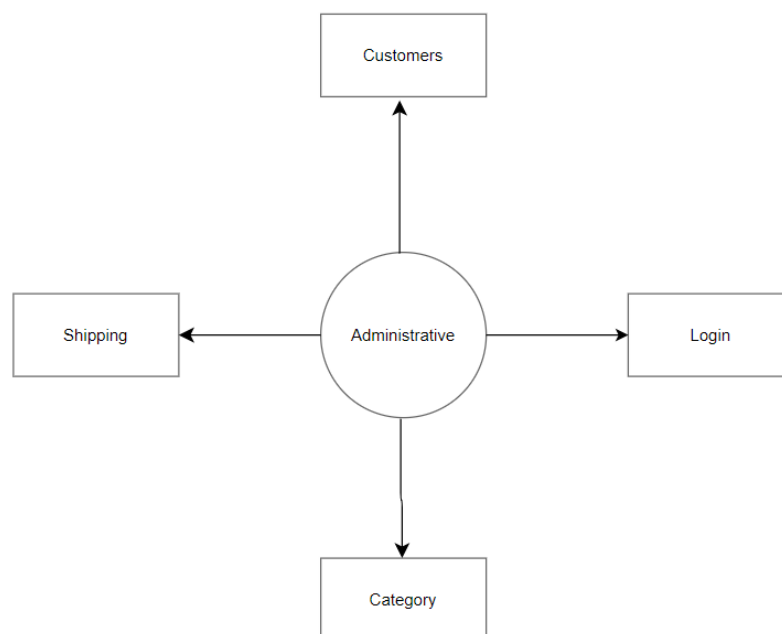
Once when the client starts using the developed systems, then the real issues come up and requirements to be solved from time to time.

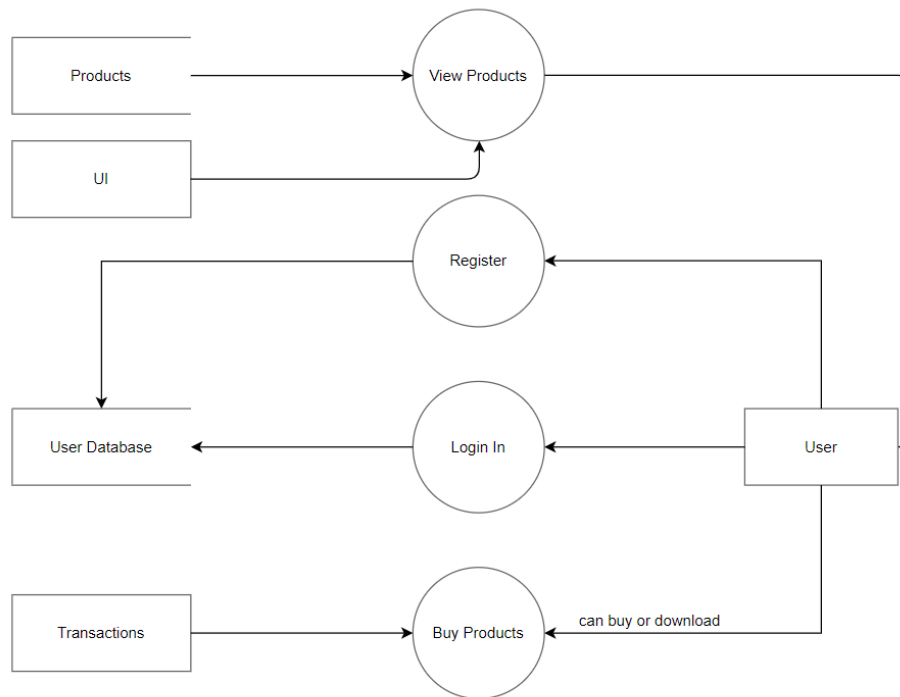
This procedure where the care is taken for the developed product is known as maintenance.

5.2.DFD

Data flow diagram is the starting point of the design phase that functionally decomposes the requirements specification. A DFD consists of a series of bubbles joined by lines. The bubbles represent data transformation and the lines represent data flows in the system. A DFD describes what data flow rather than how they are processed, so it does not hardware, software and data structure.

A data-flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFDS can also be used for the visualization of data processing (structured design). A data flow diagram (DFD) is a significant modeling technique for analyzing and constructing information processes DFD literally means an illustration that explains the course or movement of information in a process. DFD illustrates this flow of information in a process based on the inputs and outputs.





5.3.Flowchart

Flowchart is a graphical representation of an algorithm. Programmers often use it as a program-planning tool to solve a problem. It makes use of symbols which are connected among them to indicate the flow of information and processing. The process of drawing a flowchart for an algorithm is known as “flowcharting”.

Basic Symbols used in Flowchart Designs

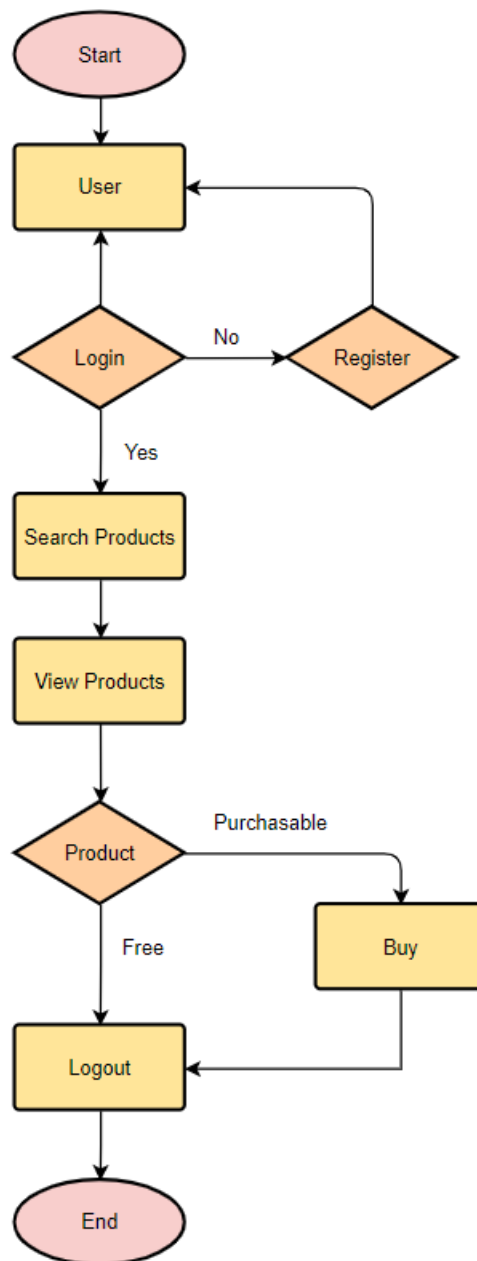
Terminal: The oval symbol indicates Start, Stop and Halt in a program’s logic flow. A pause/halt is generally used in a program logic under some error conditions. Terminal is the first and last symbols in the flowchart.

Input/Output: A parallelogram denotes any function of input/output type. Program instructions that take input from input devices and display output on output devices are indicated with parallelogram in a flowchart.

Processing: A box represents arithmetic instructions. All arithmetic processes such as adding, subtracting, multiplication and division are indicated by action or process symbol.

Connectors: Whenever flowchart becomes complex or it spreads over more than one page, it is useful to use connectors to avoid any confusions. It is represented by a circle.

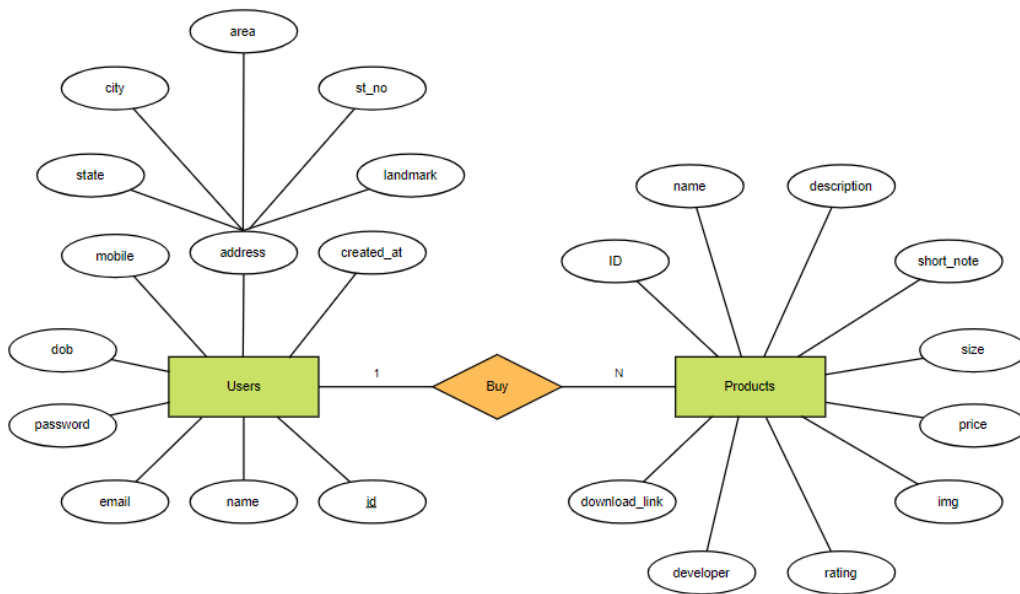
Flow lines: Flow lines indicate the exact sequence in which instructions are executed. Arrows represent the direction of flow of control and relationship among different symbols of flowchart.



5.4.E-R diagram

ER Model is used to model the logical view of the system from data perspective which consists of these components: Entity, Entity Type, Entity Set.

An Entity may be an object with a physical existence a particular person, car, house, or employee or it may be an object with a conceptual existence a company, a job, or a university course.

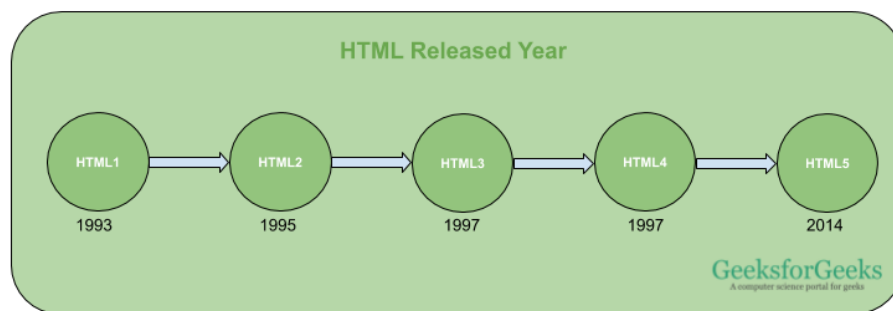


6. DATA DICTIONARY

6.1.Introduction to Languages

HTML

HTML stands for Hyper Text Markup Language. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. A markup language is used to define the text document within tag which defines the structure of web pages. This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly. Most markup languages (e.g., HTML) are human-readable. The language uses tags to define what manipulation has to be done on the text.



Elements and Tags: HTML uses predefined tags and elements which tell the browser how to properly display the content. Remember to include closing tags. If omitted, the browser applies the effect of the opening tag until the end of the page.

HTML page structure: The basic structure of an HTML page is laid out below. It contains the essential building-block elements (i.e., doctype declaration, HTML, head, title, and body elements) upon which all web pages are created.

```
<!DOCTYPE html>  ← Tells version of HTML
<html>           ← HTML Root Element

<head>           ← Used to contain page HTML metadata
  <title>Page Title</title> ← Title of HTML page
</head>

<body>           ← Hold content of HTML
  <h2>Heading Content</h2> ← HTML heading tag
  <p>Paragraph Content</p> ← HTML paragraph tag
</body>

</html>
```

HTML Page Structure

`<DOCTYPE! html>`: This is the document type declaration (not technically a tag). It declares a document as being an HTML document. The doctype declaration is not case-sensitive.

`<html>`: This is called the HTML root element. All other elements are contained within it.

`<head>`: The head tag contains the “behind the scenes” elements for a webpage. Elements within the head aren’t visible on the front-end of a webpage. HTML elements used inside the `<head>` element include:

- `<style>`
- `<title>`
- `<base>`
- `<noscript>`
- `<script>`
- `<meta>`
- `<link>`

`<body>`: The body tag is used to enclose all the visible content of a webpage. In other words, the body content is what the browser will show on the front-end.

An HTML document can be created using any text editor. Save the text file using .html or .htm. Once saved as an HTML document, the file can be opened as a webpage in the browser.

Note: Basic/built-in text editors are Notepad (Windows) and TextEdit (Macs). Basic text editors are entirely sufficient for when you’re just getting started. As you progress, there are many feature-rich text editors available which allow for greater function and flexibility.

Introduction to JavaScript

JavaScript is a lightweight, cross-platform, and interpreted scripting language. It is well-known for the development of web pages; many non-browser environments also use it. JavaScript can be used for Client-side developments as well as Server-side developments. JavaScript contains a standard library of objects, like Array, Date, and Math, and a core set of language elements like operators, control structures, and statements.

Syntax:

```
<script>
```

```
// JAVASCRIPT CODE
```

```
</script>
```

- ◆ JavaScript was created in the first place for DOM manipulation. Earlier websites were mostly static, after JS was created dynamic Web sites were made.

- ◆ Functions in JS are objects. They may have properties and methods just like another object. They can be passed as arguments in other functions.
- ◆ Can handle date and time.
- ◆ Performs Form Validation although the forms are created using HTML.
- ◆ No compiler is needed.

Advantages of JS:

- ◆ JavaScript executed on the user's browsers not on the webserver so it saves bandwidth and load on the webserver.
- ◆ The JavaScript language is easy to learn it offers syntax that is close to English language.
- ◆ In JavaScript, if you ever need any certain feature then you can write it by yourself and use an add-on like Greasemonkey to implement it on the web page.
- ◆ It does not require a compilation process so no compiler is needed user's browsers do the task.
- ◆ JavaScript is easy to debug, and there are lots of frameworks available that you can use and become master on that.

Disadvantages of JS:

- ◆ JavaScript codes are visible to the user so user can place some code into the site that compromises the security of data over the website. That will be security issue.
- ◆ All browsers interpret JavaScript that is correct, but they interpret differently to each other.
- ◆ It only supports single inheritance, so in few cases may require the object-oriented language characteristic.
- ◆ A single error in code can totally stop the website's code rendering on the website.

CSS

Cascading Style Sheets, fondly referred to as CSS, is a simply designed language intended to simplify the process of making web pages presentable. CSS allows you to apply styles to web pages. More importantly, CSS enables you to do this independent of the HTML that makes up each web page. CSS is easy to learn and understand, but it provides powerful control over the presentation of an HTML document.

CSS saves time: You can write CSS once and reuse the same sheet in multiple HTML pages.

Easy Maintenance: To make a global change simply change the style, and all elements in all the webpages will be updated automatically.

Search Engines: CSS is considered a clean coding technique, which means search engines won't have to struggle to "read" its content.

Superior styles to HTML: CSS have a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.

Offline Browsing: CSS can store web applications locally with the help of an offline cache. Using this we can view offline websites.

CSS Syntax:

A CSS comprises style rules that are interpreted by the browser and then applied to the corresponding elements in your document.

A style rule set consists of a selector and declaration block.

```
p {  
  color: blue;  
  text-align: center;  
}
```

TailwindCSS

Tailwind CSS is basically a utility-first CSS framework for rapidly building custom user interfaces. It is a highly customizable, low-level CSS framework that gives you all of the building blocks you need to build bespoke designs without any annoying opinionated styles you have to fight to override. The beauty of this thing called tailwind is it doesn't impose design specification or how your site should look like, you simply bring tiny components together to construct a user interface that is unique. What Tailwind simply does is take a 'raw' CSS file, processes this CSS file over a configuration file, and produces an output.

- Faster UI building process
- It is a utility-first CSS framework which means we can use utility classes to build custom designs without writing CSS as in traditional approach.

With Tailwind

Code:

```
index.html  
<button class="bg-blue-500 text-white py-2 px-2 rounded">  
  Primary  
</button>
```

Output:

Primary

Without Tailwind

Code:

```
index.html  
<button class="button">Primary</button>  
  
style.css  
button {  
  background-color: #4299e1;  
  color: #fff;  
  border-radius: 0.25rem;  
  padding-top: 0.5rem;  
  padding-left: 0.5rem;  
  padding-bottom: 0.5rem;  
  padding-right: 0.5rem;  
}
```

Output:

Primary

MySQL

MySQL is an open-source relational database management system (RDBMS). It is the most popular database system used with PHP. MySQL is developed, distributed, and supported by Oracle Corporation.

- The data in a MySQL database are stored in tables which consists of columns and rows.
- MySQL is a database system that runs on a server.
- MySQL is ideal for both small and large applications.
- MySQL is very fast, reliable, and easy to use database system. It uses standard SQL
- MySQL compiles on a number of platforms.

Using **MySQLi procedural** procedure: There is also a procedural approach of MySQLi to establish a connection to MySQL database from a PHP script as described below.

Syntax:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";

// Creating connection
$conn = mysqli_connect($servername, $username, $password);

// Checking connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
echo "Connected successfully";
?>
```

PHP

The term PHP is an acronym for *PHP: Hypertext Pre-processor*. PHP is a server-side scripting language designed specifically for web development. It is open-source which means it is free to download and use. It is very simple to learn and use. The files have the extension “.php”. Rasmus Lerdorf inspired the first version of PHP and participating in the later versions. It is an interpreted language and it does not require a compiler. PHP code is executed in the server. It can be integrated with many databases such as Oracle, Microsoft SQL Server, MySQL, PostgreSQL, Sybase, Informix.

It is powerful to hold a content management system like WordPress and can be used to control user access. It supports main protocols like HTTP Basic, HTTP Digest, IMAP, FTP, and others.

Other characteristics of PHP are as follows.

- Simple and fast
- Efficient
- Secured
- Flexible
- Cross-platform,

Syntax:

```
<?php
    PHP code goes here
?>
```

6.2.Physical Design

The physical design is a kind of system design. It is a graphical illustration of the system, representing external and internal entities of the system with to and fro data flow. An external entity is an entity exterior to the system and internal entity is an entity inside the system where both can convert the data. In fact, the physical design is related to the concrete input and output of the system. This deals with how the input data is provided, how the input is processed, and how the output is displayed.

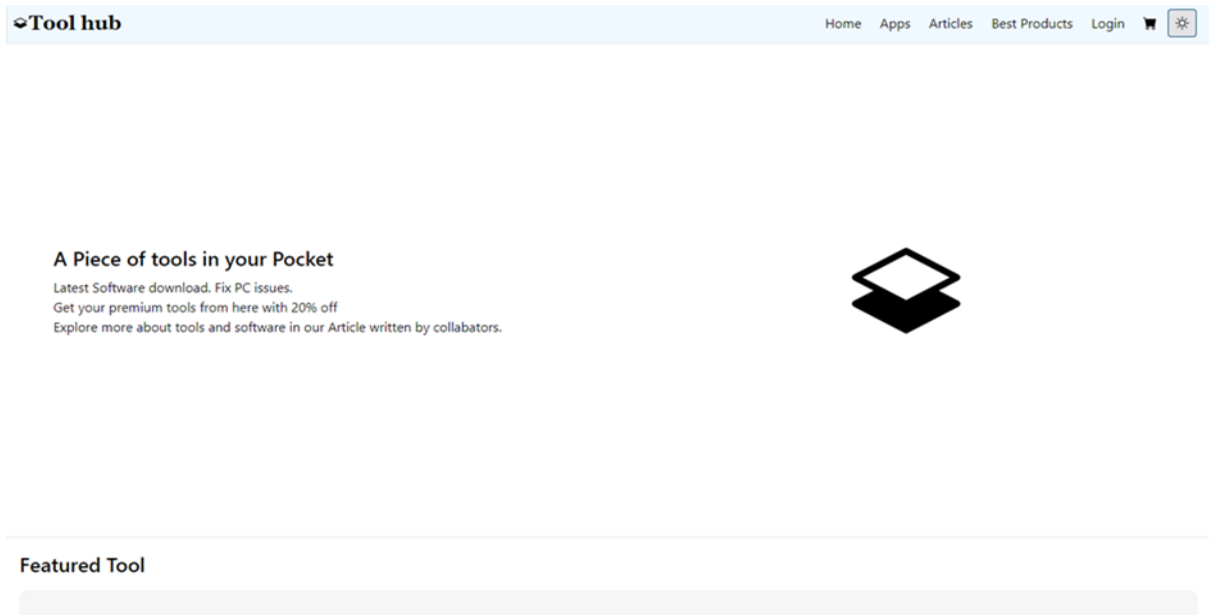


Figure 1 Home page

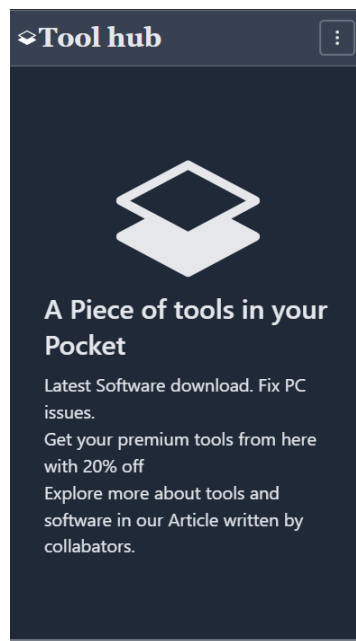


Figure 2 Featured tools

Featured Tool

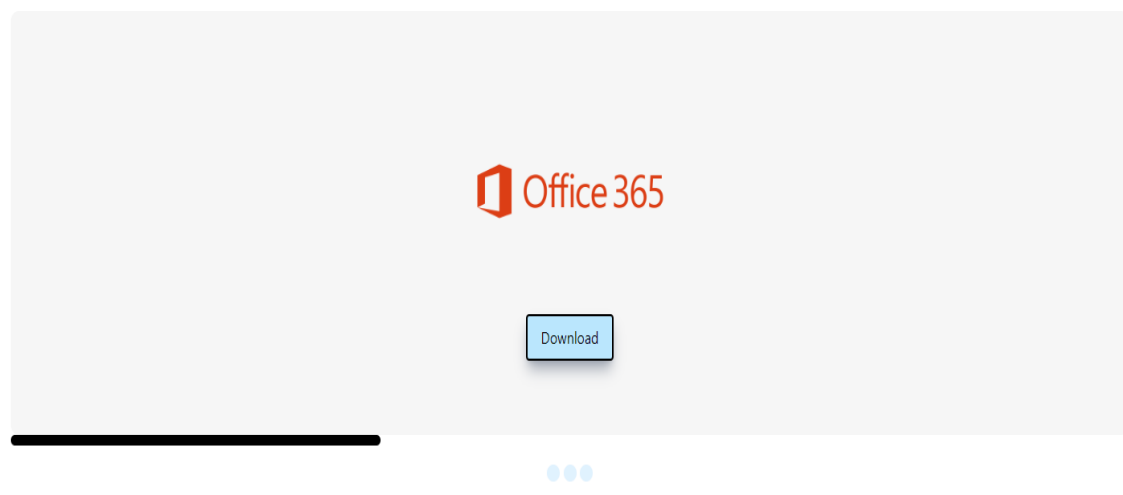
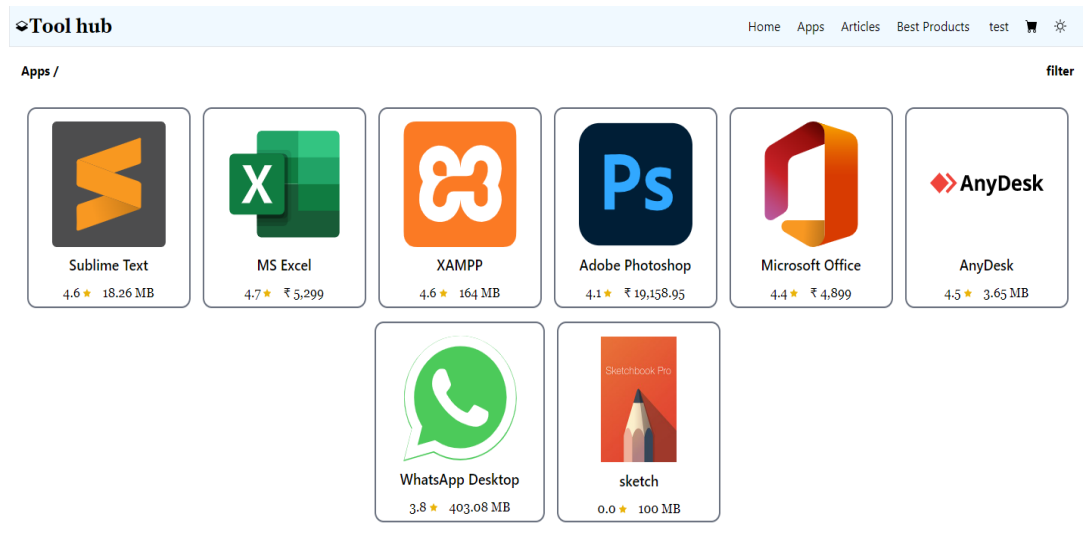


Figure 3 Featured tools



Lets take your digital experience to the next level

With over ten years of experience and many successful projects completed, we know which softwares suits you better.

Figure 3 Apps

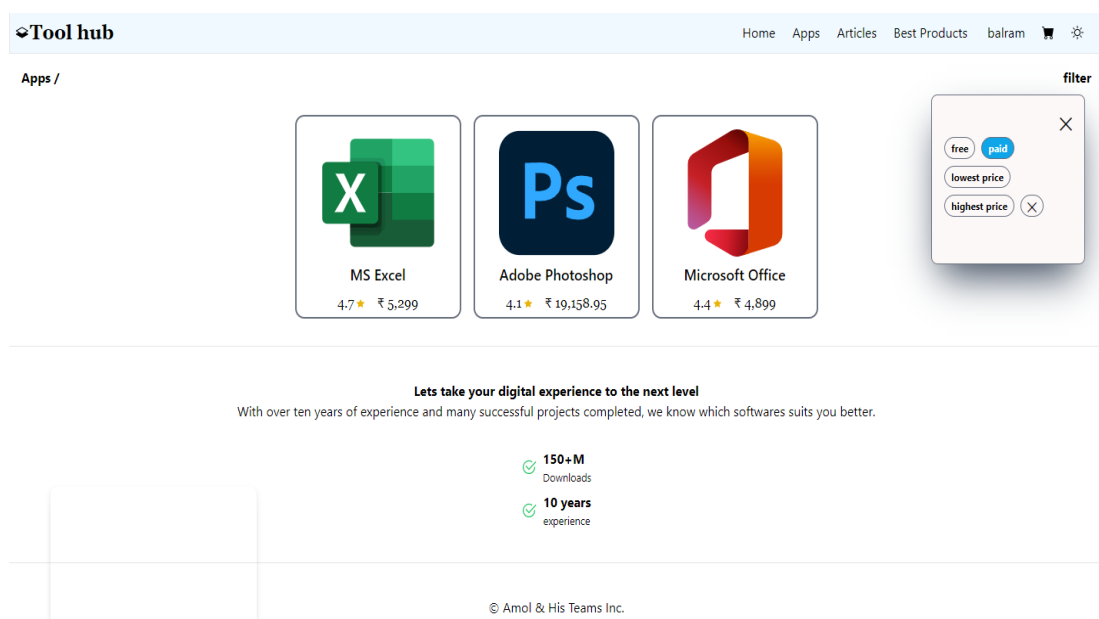


Figure 4 Working with Filter

Login

User

test

Password

....

Confirm

dont have a account? [Register](#)

Lets take your digital experience to the next level

With over ten years of experience and many successful projects completed, we know which softwares suits you better.

- ✔️ **150+ M**
Downloads
- ✔️ **10 years**
experience

Figure 5 Login

Login

No Account exist with this name

User

Your name

Password

Password

Confirm

dont have a account? [Register](#)

Lets take your digital experience to the next level

With over ten years of experience and many successful projects completed, we know which softwares suits you better.

- ✔️ **150+ M**
Downloads

Figure 6 Login with Validation

Register


User	Email
<input type="text" value="Your name"/>	<input type="text" value="Your Email"/>
Password	Re-Password
<input type="text" value="Password"/>	<input type="text" value="Enter password again"/>
Date of Birth	Mobile No.
<input type="text" value="dd-mm-yyyy"/> 	<input type="text" value="111-222-3333"/>
State	City
<input type="text" value="eg. punjab"/>	<input type="text" value="eg. ludhiana"/>
Area	Block No.
<input type="text" value="eg. model town"/>	<input type="text"/>
Street No.	House No.
<input type="text"/>	<input type="text"/>
Landmark	
<input type="text"/>	

Figure 7 Register

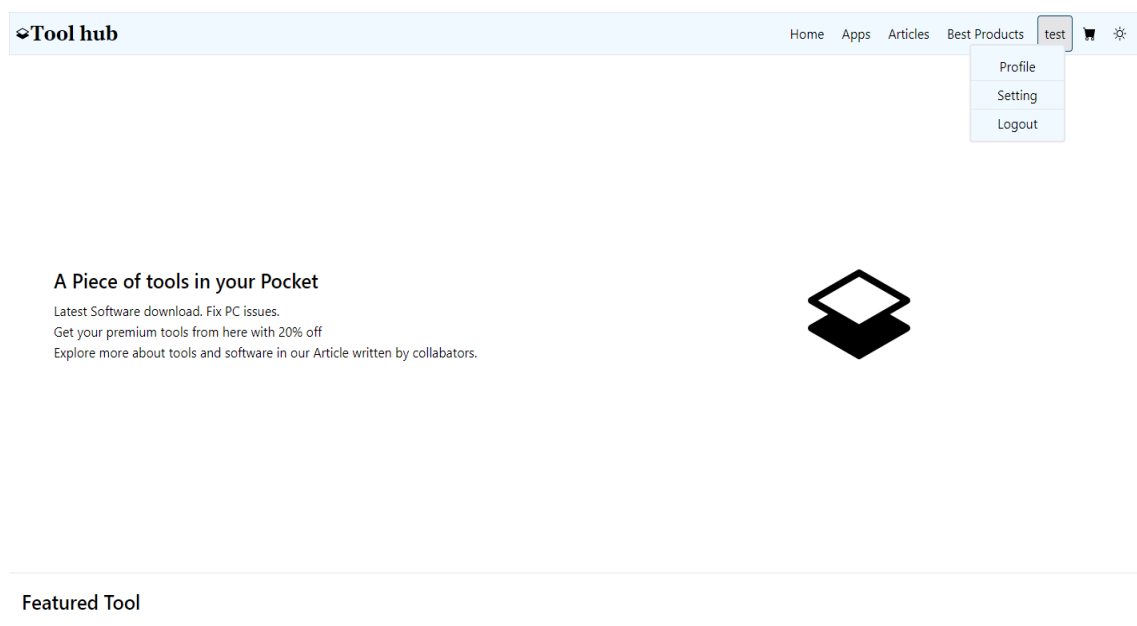


Figure 8 User Info

Test

Contact Information

Mobile No. 4564564560
Address 537-D street no.1000 Block-no near Model town Nice leh Assam
E-mail test@gmail.com

Lets take your digital experience to the next level

With over ten years of experience and many successful projects completed, we know which softwares suits you better.

 **150+M**
Downloads

 **10 years**
experience

© Amol & His Teams Inc.
All rights reserved.
[Privacy Policy](#)

Figure 9 Profile



Test

Update Info

Change Password

Delete Account?

Lets take your digital experience to the next level

With over ten years of experience and many successful projects completed, we know which softwares suits you better.

 **150+M**
Downloads

 **10 years**
experience

Figure 10 Setting

Update Information


Username	<input type="text" value="test"/>
Date of Birth	<input type="text" value="30-04-2022"/> 
E-mail	<input type="text" value="test@gmail.com"/>
Mobile No.	<input type="text" value="4564564560"/>
State	<input type="text" value="assam"/>
City	<input type="text" value="leh"/>
Area	<input type="text" value="nice"/>
Block No.	<input type="text" value="no"/>
Street No.	<input type="text" value="1000"/>
House No.	<input type="text" value="537-D"/>

Figure 11 Update User



Test

Password	<input type="password" value="Password"/>
Create New Password	<input type="text" value="Enter"/>
	<input type="text" value="Re-Enter"/>
	<input type="button" value="Confirm"/>

Lets take your digital experience to the next level

With over ten years of experience and many successful projects completed, we know which softwares suits you better.

 **150+M**
Downloads

 **10 years**
experience

Figure 12 Change Password

Tool hub

HomeAppsArticlesBest Productsbalram

B

Balram

Incorrect Password

Password

wrong

Create New Password

...

...

Confirm

Lets take your digital experience to the next level

With over ten years of experience and many successful projects completed, we know which softwares suits you better.

✓ 150+M

Downloads

✓ 10 years

experience

Figure 13 Validation on Change password



Test

Update Info

Change Password

Delete Account?

Lets take your digital experience to the next level

With over ten years of experience and many successful projects completed, we know which softwares suits you better.

 **150+M**
Downloads

 **10 years**
experience

Figure 14 Deleting User

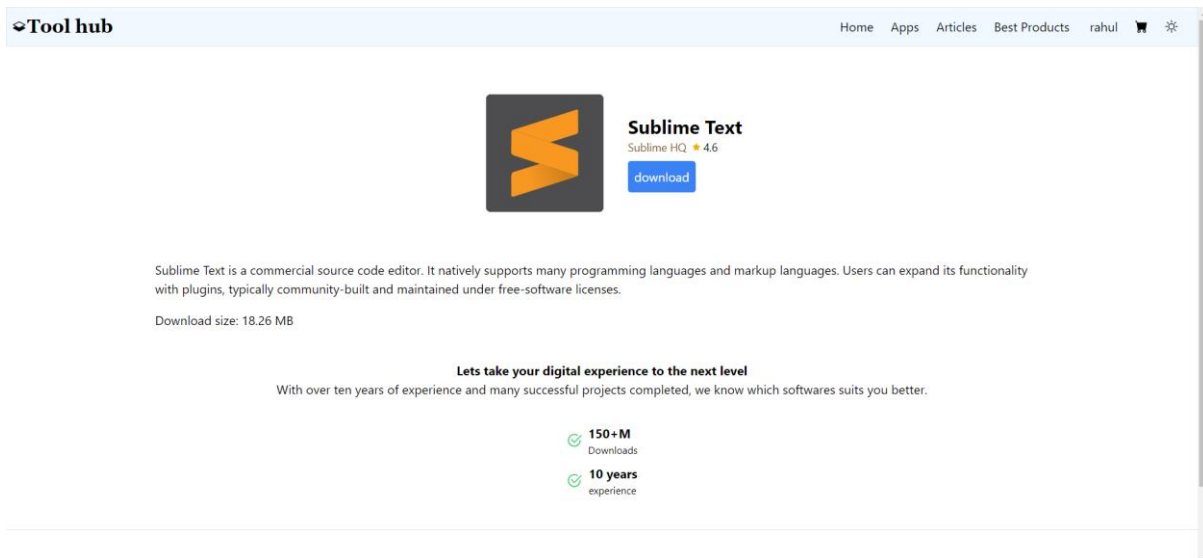


Figure 15 Product template

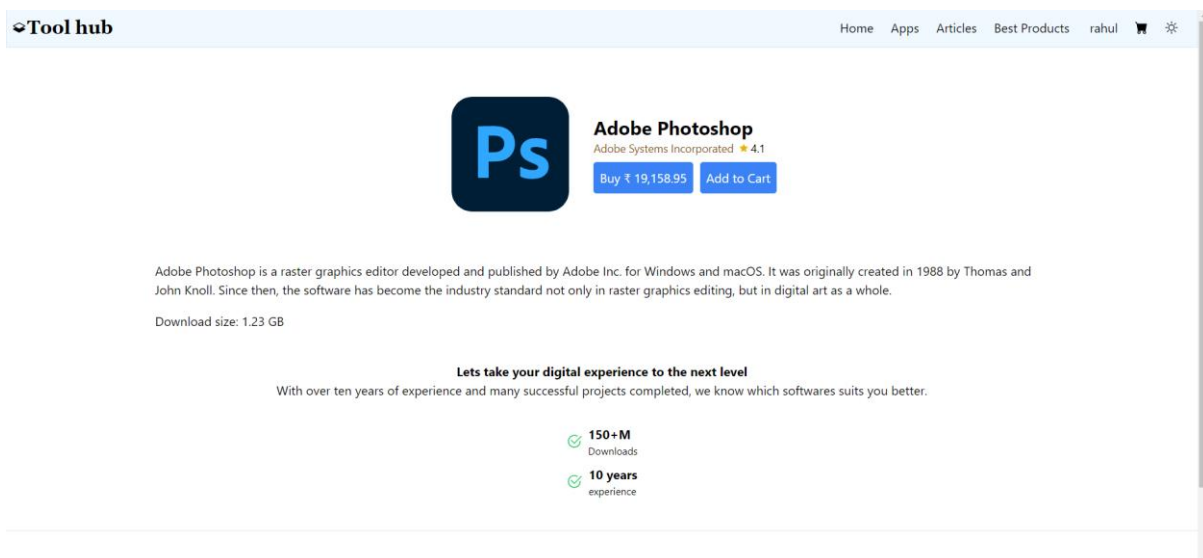
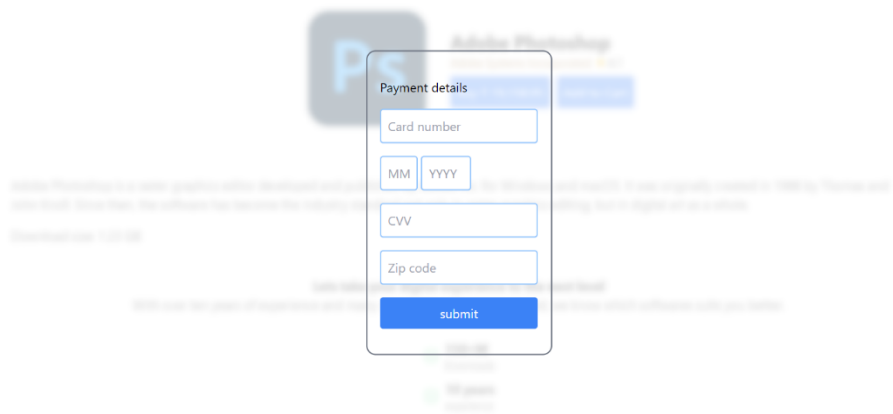


Figure 16 Paid Products template



Payment details

Card number

MM YYYY

CVV

Zip code

submit

Figure 17 Payment detail

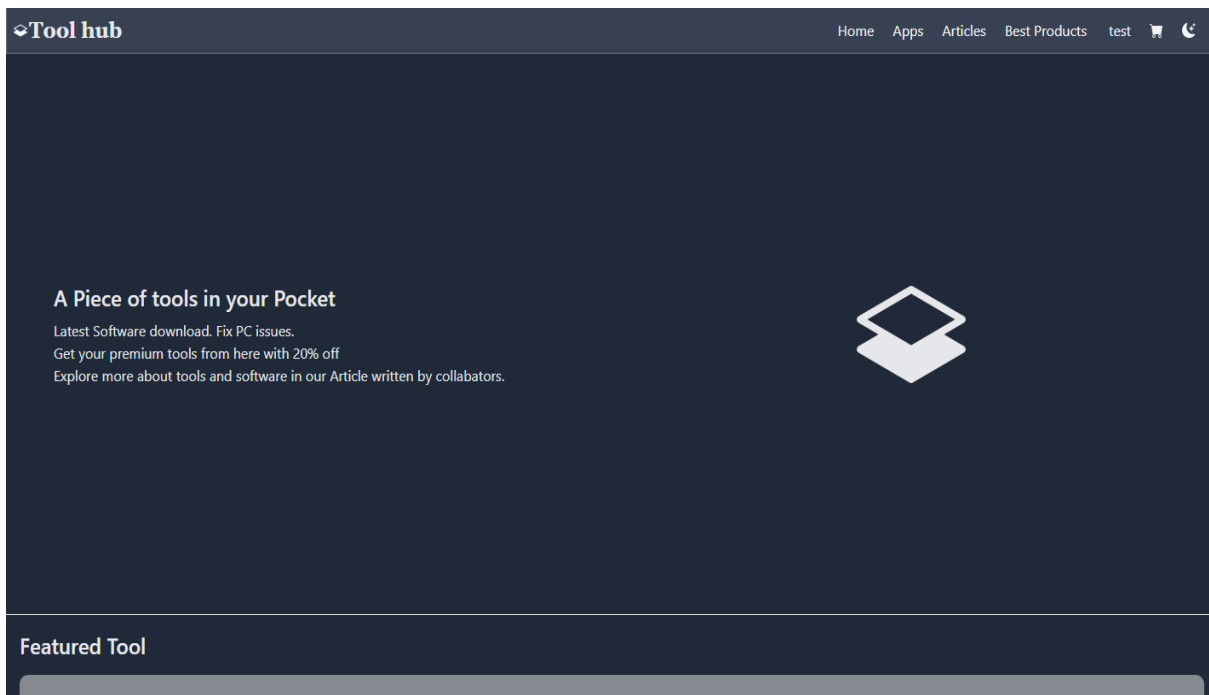


Figure 18 Home page in dark mode

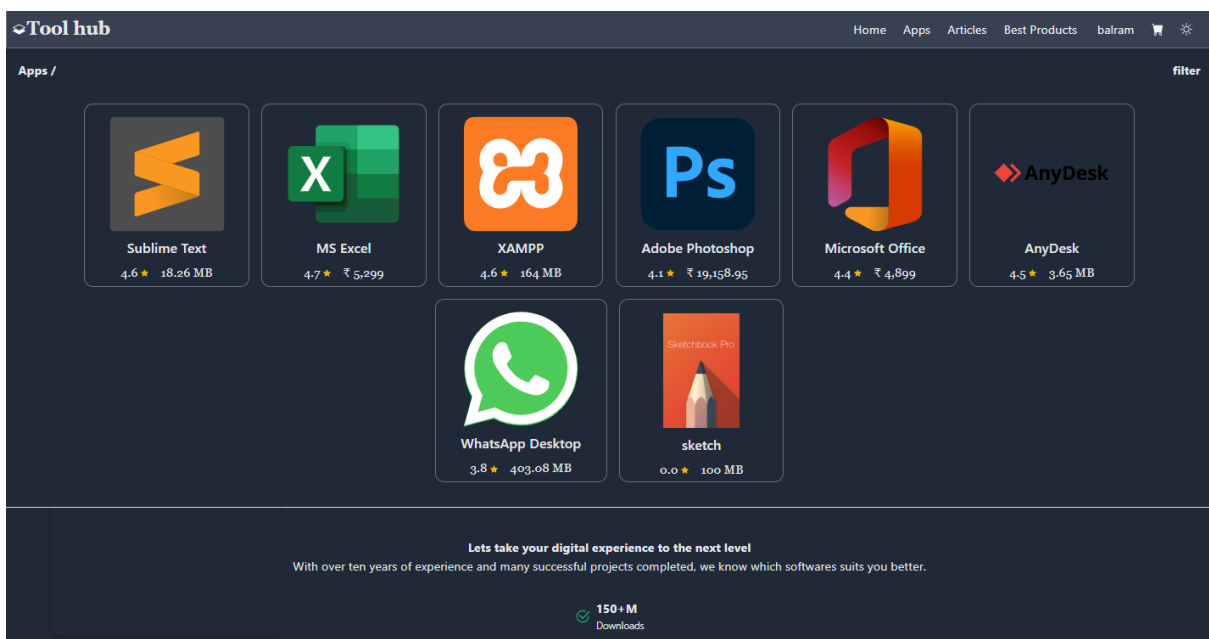


Figure 19 dark mode

Server: 127.0.0.1 » Database: toolhub » Table: users

Browse

Structure

SQL

Search

Insert

Export

Import

Privileges

Operations

Tracking

Triggers

Showing rows 0 - 5 (6 total, Query took 0.0008 seconds.)

SELECT * FROM `users`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all

Number of rows: 25

Filter rows: Search this table

Sort by key: None

Options

		id	name	email	password	bod	mobile	state	city	area	block	st_no	h_no	landmark	created_at		
<input type="checkbox"/>				7	mohit	mohit@amol.com	\$2y\$10\$esqGepqLde42f6sTt0TYh Ug0sqXitQSpJetrRNP9ul...	2022-04-07	1112223333	punjab	ludhiana	mdel town	D	2	123	clock town	2022-04-07 01:23:16
<input type="checkbox"/>				8	test	test@gmail.com	\$2y\$10\$JsZoeTsMRkbjZnZFFyle.z6YLQ3VQC5pnw3tbbJxAq...	2022-04-30	4564564560	assam	leh	nice	no	1000	537-D	model town	2022-04-07 01:35:56
<input type="checkbox"/>				10			\$2y\$10\$KAJ3KvFOVasCdTvcBMJOGex4HMDXongtIBTI9Y0Y9Wi...	0000-00-00	0								2022-04-08 21:40:14
<input type="checkbox"/>				13	mayank	mayank@amol.com	\$2y\$10\$lgvnuSjRLgl5KQ7cmckAZeDFTFOCURu5IT.0d7U5iZ...	2022-04-04	1112223333	punjab	ludhiana	arya	IT	1	lab3	none	2022-04-09 00:12:17
<input type="checkbox"/>				14	balram	balram@amol.com	\$2y\$10\$Lbm5Ug1uFJlp2SLc8h8euWUdx8pH5JVkP6Z2CI4T...	2022-04-08	1234567890	punjab	ludhiana	model toem	D	12	240	krishna mandir	2022-04-18 09:55:44

Check all

With selected:

Edit

Copy

Delete

Export

Show all

Number of rows: 25

Filter rows: Search this table

Sort by key: None

Query results operations

Print

Copy to clipboard

Export

Display chart

Create view

Bookmark this SQL query

Label:

☐ Let every user access this bookmark

Console

Press Ctrl+Enter to execute query

> SELECT * FROM `users`

> DELETE FROM `users` WHERE `users`.`id` = 1

Bookmarks Options History Clear

Figure 20 User table

Source code

```
<?php
spl_autoload_register('auto');
function auto($class){
    include($class . '.php');
}
?>
```

Figure 23 autoload function

```
<?php
namespace App;

class Db{
    public $conn;
    public function __construct(){
        $this->conn = mysqli_connect(HOST,USERNAME,PASSWORD,DATABASE);
        if(!$this->conn)
        {
            die('connection erupted;'. mysqli_connect_error());
        }
    }
}
?>
```

Figure 24 database class

```

<?php
namespace App\Models;

class Model{
    protected $table;
    protected $conn;
    function __construct($db){
        $this->conn = $db->conn;
    }
}

```

Figure 25 Modal class

```

<?php
namespace App\Models;
use App\Models\Model;
class Product extends Model{
    protected $table='products';
}
?>

```

Figure 26 Product class

```

<?php
namespace App\Models;
use App\Models\Model;
class User extends Model{
    protected $table = 'users';
    use \Traits\insertUser;
}

```

Figure 27 Users class

```

<?php
namespace App\Models;
use App\Models\Model;
class Dist extends Model{
    protected $table = 'auth_dist';
    use \Traits\insertDist;
}

```

Figure 28 Dist class

```

<?php
namespace Helper\Buyable;
use Helper\Unit;
class Buyable{
    /**
     * used to find whether product is free or premium
     * @method is
     * @param int price of the product
     * @return bool TRUE if product is premium
     */
    static function is($price){
        if($price==0){
            return false;
        }
        return true;
    }
}

```

Figure 29 is() function

```

function where($col,$oper,$val){
    $stmt = mysqli_prepare($this->conn,"select * from ". $this->table ." where ` $col ` ". $oper. " ? ");
    mysqli_stmt_bind_param($stmt,"s",$val);
    if(!$stmt){
        echo "error in stmt :". mysqli_error($this->conn);
    }
    mysqli_stmt_execute($stmt);
    $result = mysqli_stmt_get_result($stmt);
    if(!$result){
        die("Error in Model : ". mysqli_error($this->conn));
    }
    if(mysqli_num_rows($result)>0){
        return $result;
    }
    return false;
}

```

Figure 30 where command

```

function orderBy($col,$opt='ASC'){
    $result = mysqli_query($this->conn,"select * from ". $this->table ." order by ". $col ." ". $opt .""");
    if(!$result){
        die("Error in Model : ". mysqli_error($this->conn));
    }
    if(mysqli_num_rows($result)>0){
        return $result;
    }
    return false;
}

```

Figure 31 orderby command


```

function update($col,$val){
    $stmt = mysqli_prepare( $this->conn,"update ". $this->table ." set ` $col ` = ? where id = ? ");
    if(!$stmt){
        echo "error in stmt1 :". mysqli_error($this->conn);
        die();
    }
    mysqli_stmt_bind_param($stmt,'ss',$val,$_SESSION['id']);
    if(!$stmt){
        echo "error in stmt2 :". mysqli_error($this->conn);
        die();
    }
    $result = mysqli_stmt_execute($stmt);
    if(!$result){
        die("Error in Model : ". mysqli_error($this->conn));
    }
    return true;
}

```

Figure 32 update command

```

function delete($col,$val){
    $result = mysqli_query($this->conn,"DELETE FROM ". $this->table ." WHERE ". $col ." = ". $val);
    if(!$result){
        die("Error in Model : ". mysqli_error($this->conn));
    }
    return true;
}

```

Figure 33 delete command

```

<?php
require_once("../init.php");
use App\Controllers\Users;
use Helper\FILTER\FILTER;
use Helper\Msg;

//Register
if(isset($_POST['reg'])){
    $data = FILTER::data($_POST);
    $cnt = Users::where('name','=', $data['name']);
    if(!$cnt){
        Users::insert($data);
    }
    else{
        Msg::set('user_err', "Username Already Exists.");
        header("Location:../register.php");
        die();
    }
}
header("Location:../index.php");
?>

```

Figure 34 Registration

```

<?php
require_once("../init.php");
use App\Controllers\Users;
use Helper\FILTER\FILTER;
use Helper\Msg;

//logging in
if(isset($_POST['log'])){
    $data=FILTER::data($_POST);
    $cnt = Users::where('name','=', $data['name']);
    if($cnt){
        if(password_verify($data['password'], $cnt['password'][0])){
            $_SESSION['name']=$cnt['name'][0];
            $_SESSION['id']=$cnt['id'][0];
        }else{
            Msg::set('user_err','Password do not match');
            header("Location:../login.php");
            die();
        }
    }else{
        Msg::set('user_err','No Account exist with this name');
        header("Location:../login.php");
        die();
    }
}
header("Location:../index.php");
?>

```

Figure 35 Login

```
<?php
require_once('init.php');
unset($_SESSION['name']);
unset($_SESSION['id']);
header("Location:index.php");
?>
```

Figure 36 Logout

```

if(localStorage.theme === 'dark' || (!(theme in localStorage) && window.matchMedia('(prefers-color-scheme: dark)').matches)) {
  document.documentElement.classList.add('dark')
}else{
  document.documentElement.classList.remove('dark')
}
document.querySelector(".dark-mode-icon").addEventListener('click',() =>{
  Array.from(document.getElementsByClassName('icon')).forEach( i => {
    i.classList.toggle('icon-hidden');
  });
  document.documentElement.classList.toggle('dark');
  localStorage.theme = document.documentElement.classList.contains('dark')?'dark':'light';
});

```

Figure 37 Darkmode Module

```

if('Notification' in window){
  if(Notification.permission==='granted'){
    var notification = new Notification('Watch Latest App',{
      body:'Go to Products page',
      icon:'assets/favicon/favicon-32x32.png',
      badge:'assets/favicon/favicon-32x32.png',
    });
  }
  else if(Notification.permission !== 'denied'){
    Notification.requestPermission().then( function(per){
      if(per==='granted'){
        var notification = new Notification('hi there');
      }
    });
  }
  notification.onclick = (e)=>{
    e.preventDefault();
    window.location.href = 'products.php';
  }
}

```

Figure 38 Notification API

7. TESTING

7.1. Testing Phases

The testing phase of the software development lifecycle (SDLC) is where you focus on investigation and discovery. During the testing phase, developers find out whether their code and programming work according to customer requirements.

The steps involved during Unit testing are as follows:

- a. Preparation of the test cases.
- b. Preparation of the possible test data with all the validation checks.
- c. Complete code review of the module.
- d. Actual testing done manually.
- e. Modifications done for the errors found during testing.
- f. Prepared the test result scripts.

Unit Testing

In computer programming, unit testing is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures are tested to determine whether they are fit for use.

The unit testing done included the testing of the following items:

1. Functionality of the entire module/forms.
2. Validations for user input.
3. Checking of the coding standards to be maintained during coding.
4. Testing the module with all the possible test data.
5. Testing of the functionality involving all type of calculations etc.
6. Commenting standard in the source files.

After completing the Unit testing of all the modules, the whole system is integrated with all its dependencies in that module. While System Integration, we integrated the modules one by one and tested the system at each step. This helped in reduction of errors at the time of the system testing.

The steps involved during System testing are as follows:

- Integration of all the modules/forms in the system.
- Preparation of the test cases.
- Preparation of the possible test data with all the validation checks.
- Actual testing done manually.
- Recording of all the reproduced errors.

- Modifications done for the errors found during testing.
- Prepared the test result scripts after rectification of the errors.

System Testing

System Testing is a type of software testing that is performed on a complete integrated system to evaluate the compliance of the system with the corresponding requirements.

The System Testing done included the testing of the following items:

1. Functionality of the entire system as a whole.
2. User Interface of the system.
3. Testing the dependent modules together with all the possible test data scripts.
4. Verification and Validation testing.
5. Testing the reports with all its functionality.

After the completion of system testing, the next following phase was the Acceptance Testing Clients at their end did this and accepted the system with appreciation. Thus, we reached the final phase of the project delivery.

There is other six tests, which fall under special category. They are described below:

Peak Load Test: It determines whether the system will handle the volume of activities that occur when the system is at the peak of its processing demand. For example, test the system by activating all terminals at the same time.

Storage Testing: It determines the capacity of the system to store transaction data on a disk or in other files.

Performance Time Testing: it determines the length of time system used by the system to process transaction data. This test is conducted prior to implementation to determine how long it takes to get a response to an inquiry, make a backup copy of a file, or send a transmission and get a response.

Recovery Testing: This testing determines the ability of user to recover data or re-start system after failure. For example, load backup copy of data and resume processing without data or integrity loss.

Procedure Testing: It determines the clarity of documentation on operation and uses of system by having users do exactly what manuals request. For example, powering down system at the end of week or responding to paper-out light on printer.

Human Factors Testing: It determines how users will use the system when processing data or preparing reports.

Testing the module with all the possible test data

a. Security: The subsystem should provide a high level of security and integrity of the data held by the system, only authorized personnel of the company can gain access to the company's secured page on the system, and only users with valid password and username can login to view user's page.

b. Performance and Response time: The system should have high performance rate when executing user's input and should be able to provide feedback or response within a short time span usually 50 seconds for highly complicated task and 20 to 25 seconds for less complicated task.

c. Error handling: Error should be considerably minimized and an appropriate error message that guides the user to recover from an error should be provided Validation of user's input is highly essential. Also, the standard time taken to recover from an error should be 15 to 20. seconds.

d. Availability: This system should always be available for access at 24 hours, 7 days a week Also in the occurrence of any major system malfunctioning the system should be available

7.2.Verification and Validation

- All the fields such as Name, Email are validated and does not take invalid values
- Each form for Customer cannot accept blank value fields
- Avoiding errors in data
- Controlling amount of input
- Integration of all the modules/forms in the system.
- Preparation of the test cases.
- Preparation of the possible test data with all the validation checks.
- Actual testing done manually.
- Recording of all the reproduced errors.
- Modifications done for the errors found during testing.
- Prepared the test result scripts after rectification of the errors
- Functionality of the entire module/forms
- Validations for user input.
- Checking of the Coding standards to be maintained during coding.

8. IMPLEMENTATION

This phase of the systems development life cycle refines hardware and software specifications, establishes programming plans, trains users and implements extensive testing procedures, to evaluate design and operating specifications and/or provide the basis for further modification.

Technical Design

This activity builds upon specifications produced during new system design, adding detailed technical specifications and documentation.

Test Specifications and Planning

This activity prepares detailed test specifications for individual modules and programs. job streams. subsystems, and for the system as a whole.

Programming and Testing

This activity encompasses actual development, writing, and testing of program units or modules.

User Training

This activity encompasses writing user procedure manuals, preparation of user training materials, conducting training programs, and testing procedures.

Acceptance Test

A final procedural review to demonstrate a system and secure user approval before a system becomes operational.

Installation Phase

In this phase the new Computerized system is installed, the conversion to new procedures is fully implemented, and the potential of the new system is explored.

System Installation

The process of starting the actual use of a system and training user personnel in its operation.

Review Phase

This phase evaluates the successes and failures during a systems development project, and to measure the results of a new Computerized Tran system in terms of benefits and savings projected at the start of the project.

Development Recap

A review of a project immediately after completion to find successes and potential problems in future work.

Post-Implementation Review

A review, conducted after a new system has been in operation for some time, to evaluate actual system performance against original expectations and projections for cost-benefit improvements. Also identifies maintenance projects to enhance or improve the system.

9. BIBLIOGRAPHY

- A. <https://www.php.net/>
- B. <https://www.phptutorial.net/>
- C. https://en.wikipedia.org/wiki/Systems_development_life_cycle
- D. <https://www.geeksforgeeks.org/software-testing-life-cycle-stlc/>
- E. <https://www.javatpoint.com/php-mvc-architecture>
- F. <https://developer.mozilla.org/en-US/docs/Web/API/notification>
- G. <https://developer.mozilla.org/en-US/docs/>
- H. <https://en.wikipedia.org/wiki/SQL>