SYNOPSIS

Minor Project (BCA-508)

Dressingity Ecommerce Website

(*SUBMITTED FOR PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE)*

BATCHELOR OF COMPUTER APPLICATION

SUBMITTED BY
Sonali Rawat
(ROLL NO.: 205049)
Batch (2020-23)
Session (Odd, 2022-23)

Ch. Charan Singh University, Meerut
Uttar Pradesh, INDIA

Under the Supervision of

(Assistant / Associate Professor)

INTEGRATED ACADEMY OF MANAGEMENET OF TECHNOLOGY
GHAZIABAD, UTTAR PRADESH INDIA

Contents

INTRODUCTION TO TOPIC

This, Dressingnity Dummy E-Commerce website project aims to make a Dummy website for demonstration of Technology learned by students of BCA Semester 5. This is a dummy E-Shopping website which aim to show how basic technologies of web development can be used to create an eye-soothing website. For this, we need a website that is user-friendly.

This website will be made using basic technologies such HTML, CSS, JavaScript and their framework such a Bootstrap and JQuery.

PROBLEM STATEMENT

In this Problem statement we are going to describe What you are going to design/Develop/Study, as required by format provided to us by our project mentor.

We will be making a basic dummy E-commerce website which we(Team-members) have collectively agreed to call “Dressingnity”. E-commerce provides an easy way to sell products to a large customer base. However, there is a lot of competition among multiple e-commerce sites. When users land on an e-commerce site, they expect to find what they are looking for quickly and easily. Also, users are not sure about the brands or the actual products they want to purchase.

So make this process easy for them, we aim to make this website to have a mixture of fluid and material UI, which Is easy to navigate and use.

METHODOLOGY

While discussing the methodology to follow, among our team we were really confused until we looked back and thought what methodology does most successful software company uses.

Although we know a website is vastly different from a full-fledged cross-platform FOSS (Free and Open-Sourced) software, we realized the basic principle were same for both of them.

We found that all of these follow some variation of KISS principle except Arch Linux, which completely focuses on it. So in the end we decided to take this tested path followed by many.

WHAT IS “KISS” Methodology?

KISS (Keep it Simple, Stupid) - A Design Principle:

It was Albert Einstein who said; “If you can’t explain it, you don’t understand it well enough.”

The phrase; “keep it simple, stupid” is thought to have been coined by the late Kelly Johnson, who was the lead engineer at the Lockheed Skunk Works

Today the KISS principle is celebrated in many engineering professions (including software engineering) and is often brought to bear by managers in many professions, as well as by trainers and educators.

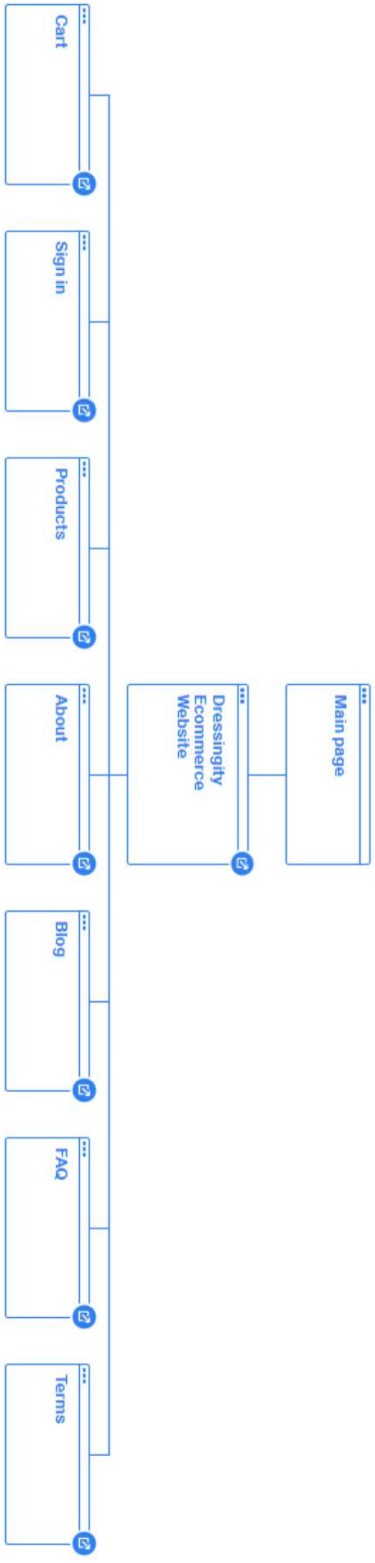
Variants of KISS

The KISS principle is also offered in two other forms (for those who feel delicate about the inclusion of the word “stupid”):

- Keep it short and simple
- Keep it simple and straightforward

Though both phrases technically introduce an “a” into the acronym – they both deliver the same message as “keep it simple, stupid”. The objective of any process is to deliver the simplest possible outcome.

Website Structure Diagram



(SRS)

(Software Requirement Specifications)

Introduction

The purpose of this document is to define and describe the requirements of the project and to spell out the system's functionality and its constraints.

General Description

1. Product Functions

The product should make entire shopping process easier and streamlined for the users and time efficient by simplifying the UI using, KISS methodology.

2. User Objectives

The user wants a website that will swift and easy to use. The website must facilitate the speed and ease of use.

3. General Constraints

Constraints include an easy to use interface for the website, a Windows platform or, at bare minimum, a Mac with Access internet. Also, it must be Fast and ad-free, and easy to navigate and use.

Functional Requirements

Items provided on Dressingnity shall be easily bought.

1. Items shall be stored on the laptop machine or server.
2. Very high up time.
3. Limited network / Wi-Fi availability could present a technical challenge

4. The above stated factor is a risk we have encountered. Eliminate it by reducing the dependency of our program on these things.

5. users should be able to add item to cart

6. user can checkout multiple item

7. User can login to access his/her/their accounts

8. This requirement is the basis of the project; all other aspects depend on it.

Interface Requirements

1. User Interfaces

- GUI – The user interface for this program is the interface is highly inspired by Zorin OS 16 Gnome version. It include front page, product page, cart page, about section for site, about section for team and a FAQ for help and support. User can suggest improvement and bug by using contact us info.
- CLI – There is no command line interface
- API – There is no API for the product

2. Hardware Interfaces

The website uses the GitHub.io for hosting the site. Access to the data and other hardware is managed by the GitHub and web browser.

3. Software Interfaces

The Website's Software interface can be accessed by using a modern computer or a smartphone device although website is designed especially for Desktop mode.

Cost Feasibility

There is no particular budget for product as the project is done be some college students. And it is a known fact that college student are always on the verge of start begging.

Software Requirements

1. Development Requirements.

- Any Modern OS
- Visual Studio Code/Codium
- Git and GitHub
- Modern Web browser (Microsoft Edge, Google Chrome etc.)
    - Pesticide Extension
    - Developer tools
    -

2. Consumer Requirements.

- Any modern Web browser (Microsoft Edge, Google Chrome etc.)
- Active Internet Connection

Hardware Requirements

1. Computer Requirements

- 1.5 GHz processor or higher
- 4 GB RAM or higher
- 4 GB Available Hard Drive Space(for Browser)
- Windows 7 SP2 or later operating system OR Any LINUX distribution.
- It is also Accessible on Mac OS
- An active Internet Connection

OR

2. Smartphone

- Android 5.0+(for Web-view and browser Support)

Technical Requirements

- HTML (HyperText Markup Language)
- CSS (Cascading Style Sheets)
- JavaScript (ES6 and Later)
- Bootstrap (HTML, CSS, JavaScript Framework, V5.2.0)
- JQuery (JavaScript library, V3.5)
- Git (Distributed version control system)
- GitHub (Git repository Hosting as well as Free basic website Hosting)
- API (Application Programming Interfaces)
- PHP
- MySQL

References

Technology References:

HTML(https://html.spec.whatwg.org/multipage/) (For Website Structure)

CSS (https://www.w3.org/Style/CSS/Overview.en.html) (For Designing and Styling)

JavaScript(https://www.javascript.com/) (For minor Functionality, Part of Bootstrap CDN)

Bootstrap(https://getbootstrap.com/) (HTML, CSS and JS Framework)

JQuery(https://jquery.com/) (Part of Bootstrap CDN)

LibreOffice(https://www.libreoffice.org/discover/libreoffice/)(For making this synopsis)

Documentation References:

MDN(https://developer.mozilla.org/en-US/)

Bootstrap Docs (https://getbootstrap.com/docs/5.2/getting-started/introduction/)

Others:

Zorin OS 16 Pro (https://zorin.com/os/) (Took UI inspiration)

GitHub(https://github.com/) (Hosting of project files as well as web hosting)

Arch Linux

(https://www.garron.me/en/linux/kiss-simplicity-arch-linux-review.html)

KISS Methodology)

Composite reference:

Google (https://www.google.co.in/)

Team-members

Prakash(205028)

Sonali Rawat(205049)

Sagar Goswami(205040)

Teesha Singhal(205041)