Dressingnity

Dressingity - Dress with dignity

INDEX

ACKNOWLEDGMENT

I am very grateful to my minor project (505P) for semester 5 mentor Dr. Surendra Gautam, for giving his valuable time and constructive guidance in preparing this Synopsis and Project (505P). It would not have been possible to complete this Project (505P) in the short period of time without his kind encouragement and valuable guidance.

DATE: SIGNATURE:

CERTIFICATE OF ORIGINALITY

This is to certify that the project synopsis entitled ____________________ submitted to Integrated Academy of Management and Technology (INMANTEC) in partial fulfillment of the requirement for the award of the degree of BACHELOR'S OF COMPUTER APPLICATIONS (BCA), is an authentic and original work carried out by Mr. / Ms.__________________ with roll no. _______________________under my guidance.

The matter embodied in this project is genuine work done by the student and has not been submitted whether to this University or to any other University / Institute for the fulfillment of the requirements of any course of study.

……………………….	...…………………….
Signature of the Student:	Signature of the Guide
Date: ………………..	Date: …………………

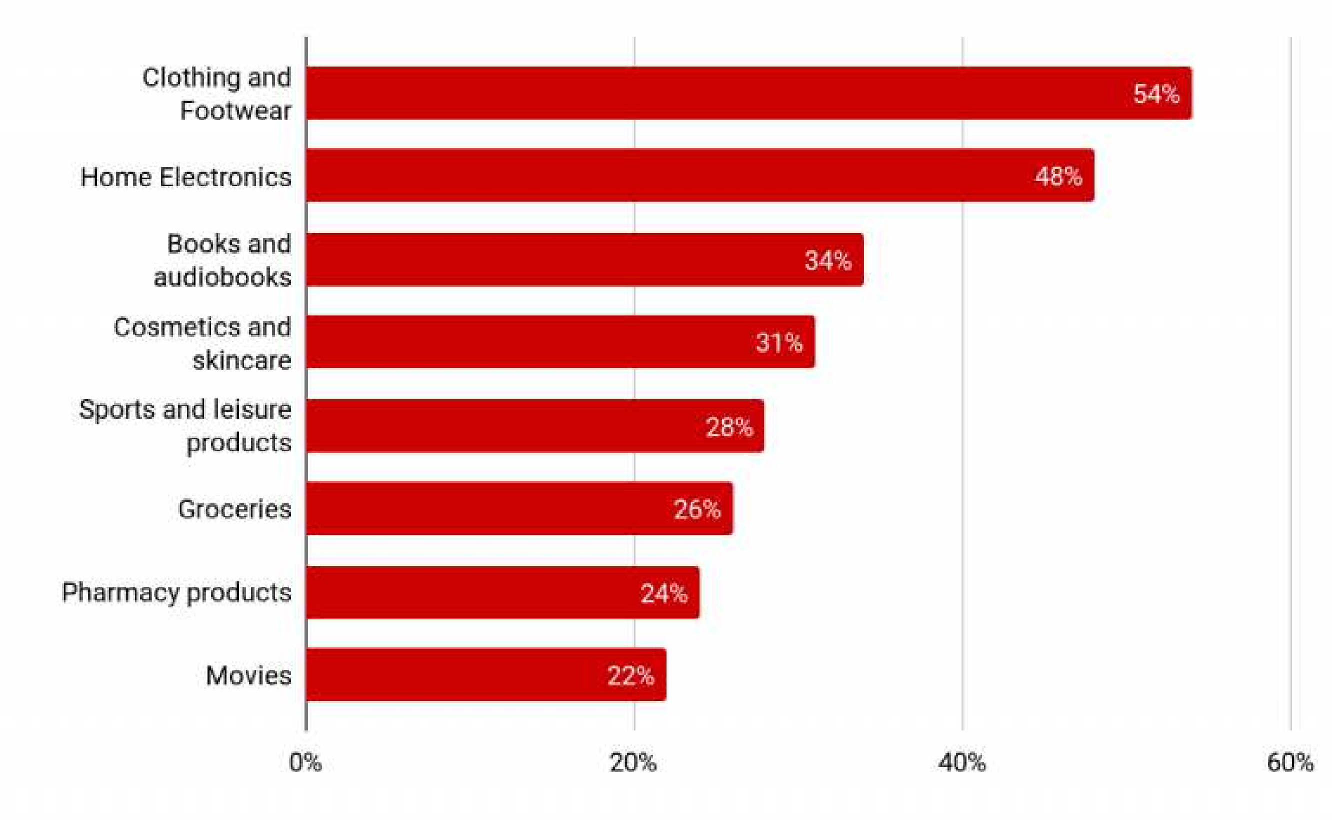
INTRODUCTION

This, Dressingity E-Commerce website project aims to make a website for masses. This is an E-Shopping website where Gen Z can buy clothing that are in trends at a reasonable price. For this, we need a website that is user-friendly and encourage user to do what they want, to shop without getting in their way with ads or unnecessary promotions. We believe the needs of Gen Z should be covered with out-most respect and care.

Our website “Dressingity” is to automate the existing manual system of buying clothing by going to shops, fulfilling the Generation Z’s requirements of fashion at short notice so they can be confident in themselves. This means they don’t need to worry about finding what is in current trend as we are always updated with it. So, they can utilize their time to maximize their awesomeness. It’s time to switch from dressing to Dressingity.

MARKET SURVEY

Before deciding on making an E-Commerce site, which will help Gen Z we researched that as to what was the need of Gen Z and instead of writing about it, we believe that a graph would show the result much more clearly.



Most commonly bought products online in age group of 14-28

This section of the E-Commerce market research is where we look closely at the factors that matter to our client as an E-Commerce business.

By seeing this trend of Gen Z, Our team decided to bask on an adventure to bring a truly Gen Z shopping experience to the of-course, Gen Z.

PROBLEM STATEMENT

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.

They have a very broad idea about what they want to buy. Many customers nowadays, search for their products on Google rather than visiting specific e-commerce sites. They believe that Google will take them to the e-commerce sites that have their product.

The purpose of any e-commerce website is to help customers narrow down their broad ideas and enable them to finalize the products.

Most know E-Commerce website (Amazon, Flipkart etc.) have an abnormally large no. of products. Which makes it hard for user to find what they actually needs. Apart from that, they have an obnoxiously large no. of ads.

Although there exist few sites that focus only on fashion like (Myntra, Ajio etc.) they generally do not have a price range that everyone can afford.

That’s where Dressingity comes in, we aim to provide trending fashion at a reasonable cost, while also provide a soothing shopping experience.

OBJECTIVES

The main objectives of creating an E-Commerce Website, We will provide all the fashion needs of the Gen Z which will help empower them by making them feel confident in themselves

To build a responsive E-Commerce website for masses to shop and explore fashion, the current trends in the fashion world. And build a portal that facilitates continuous growth.

The main objectives are:

1. To provide a user-friendly shopping experience.
2. To provide up-to-date Fashion trends.
3. To remain Ad-free, so you can focus on your shopping.
4. All of the above at a reasonable price.

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 both aim to target a large no. of customers while providing the users the tool and user experience to do what they do best, without getting in their way.

Since we swear by FOSS, we obviously thought of the big giants of FOSS Communities, such as Arch Linux, The GNU project and obviously biggest company in FOSS domain that is The Red Hat, Inc.

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.”

Though, it is often miss-reported as being; “If you can’t explain it to a six-year-old, you don’t understand it well enough.” What Einstein was driving at was a particular application of “keep it simple, stupid”.

From scientific concepts to products, the end-user doesn’t care how clever the creator or designer of something is. They care about being able to take that person’s output and make it useful to their own lives. The simpler the explanation and the simpler the product, the more likely it is that the output will be useful to others.

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

Kelly explained the idea to others with a simple story. He told the designers at Lockheed that whatever they made had to be something that could be repaired by a man in a field with some basic mechanic’s training and simple tools. The theater of war (for which Lockheed’s products were designed) would not allow for more than that. If their products weren’t simple and easy to understand – they would quickly become obsolete in combat conditions and thus worthless.

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.

The First Usability Principle?

KISS may have been the first usability principle for product design – though, it was never formally presented as a usability principle. It focuses on the idea that if we can’t understand a product, we can’t use it properly and that the widest possible audience must be able to understand it, if the product is to gain maximum market share. This is as true for mobile applications as it is for fighter planes.

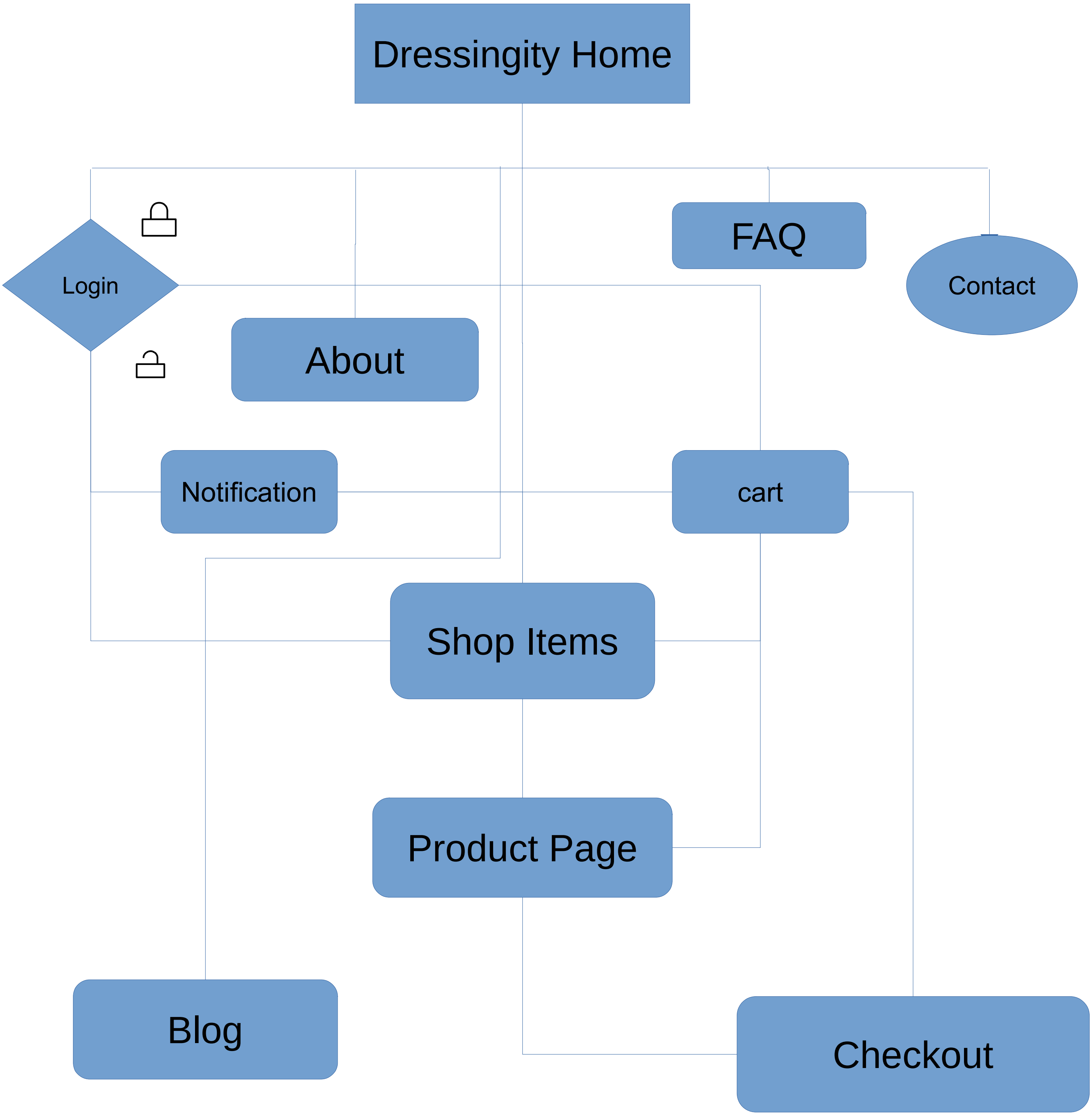
Variants of KISS

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.

1. Scope of this Document

The customer and the user for the system are the Gen Z, and the developers of the system is the Dressingity Team. Our constraints for this section includes our deadline for the document which is due ________

2. Overview

The product is a E-Commerce website for buying clothing mainly focused on Gen Z it capabilities are buying trending clothing, reasonable pricing policy, friendly and soothing UI/UX, while being ad-free. The website’s design is mainly inspired by Zorin OS 16 Gnome.

General Description

1. Product Functions

The product should make entire shopping process easier and streamlined for the users (gen z) and time efficient.

2. User Characteristics

The users include the youngsters of the Gen Z as they shop for clothing. For this system, the user is required to know the basic usability of Internet as well as a very base level understanding of UI and payment methods, which hopefully will be facilitated by the KISS methodology.

3. User Problem Statement

The users system, currently, is slow and inefficient as it full of ads and unnecessary promotions. users must do a lot of searching to find the product of their need. Too many hours have been needed to find the desired product.

4. User Objectives

The user wants a website that will swift and easy to use. The website must facilitate the speed and ease of use. It also must provide up to date fashion trends.

5. 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 Dressingity 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.

Performance Requirements

The database is designed to be operated through a normal Computer or Smartphone, thus no additional system requirements exist beyond those, except for a negligible amount of network bandwidth to access the website through internet.

Other non-functional attributes

1. Security

The system shall be designed with a level of security appropriate for the sensitivity of information enclosed in the transaction and privacy. Since there is obvious information that is of a high security level such as credit card information, the only requirements that could be implemented are encryption and usage of HTTPS.

2. Compatibility

This website will be compatible with any computer that has a modern browser installed (whether PC or Mac), and will be designed with more than one computer in mind.

3. Reliability

Reliability is one of the key attributes of the website. Back-ups will be made regularly so that restoration with minimal data loss is possible in the event of unforeseen events. The backup will be done by GIT’s own Version control System

4. Maintainability

The system shall be maintained by Current and Future Dressingity Team.

5. Portability

The system shall be designed in a way that shall allow it to be run on multiple computers with a modern browser installed.

6. Extensibility

The system shall be designed and documented in such a way that anybody with an understanding of basic web development, be able to extend the system to fit their needs with the team’s basic instructions.

7. Reusability

The system should be designed in a way that allows the website’s component to be re-used regularly for the various other projects.

8. Resource Utilization

The resources used website include: Dressingity Team, the computers in INMANTEC, and the internet.

Operational Scenarios

Scenario A: Shopping/Exploring Fashion

The user shall be able to browse and products for shopping or just exploring the current fashion trends.

Scenario B: Customer Check-out

The user shall be able to enter information about the purchasing a particular item, and be able to checkout for the purchase.

Scenario C: Contribution to website

Since website is hosted on GitHub it is completely Open soured and can be improved by anyone as they require or see fit by helping the Dressingity Team.

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 GHz processor or higher
- 1GB RAM or higher
- 1.5GB 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)

References

Technology References:

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

CSS (https://www.w3.org/Style/CSS/Overview.en.html)

JavaScript(https://www.javascript.com/)

Bootstrap(https://getbootstrap.com/)

JQuery(https://jquery.com/)

Documentation References:

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

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

Others:

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

GitHub(https://github.com/) (Hosting)

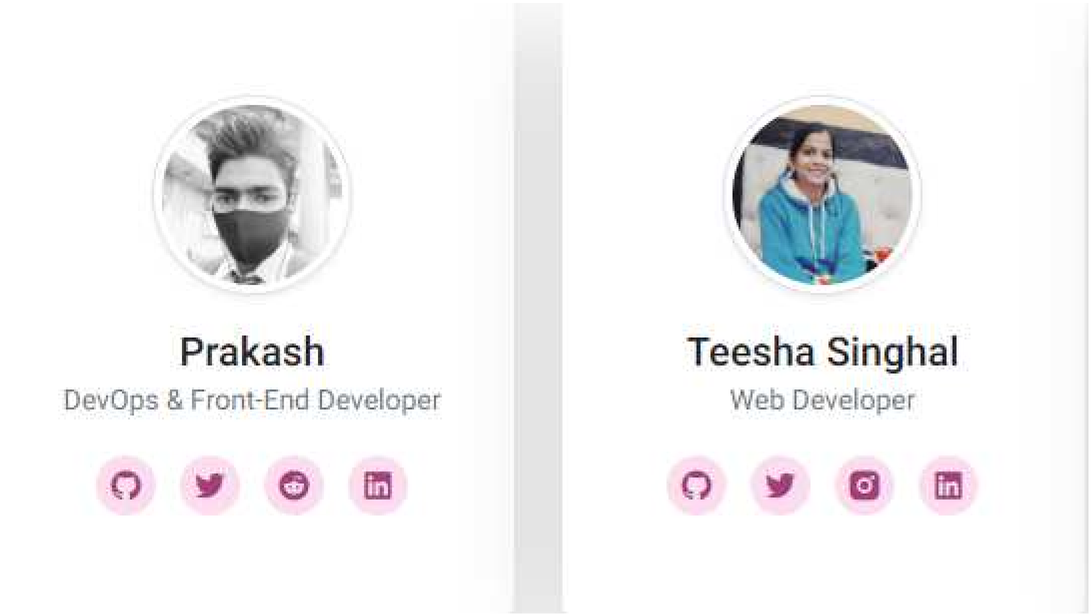
Arch Linux

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

KISS Methodology)

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

Credits

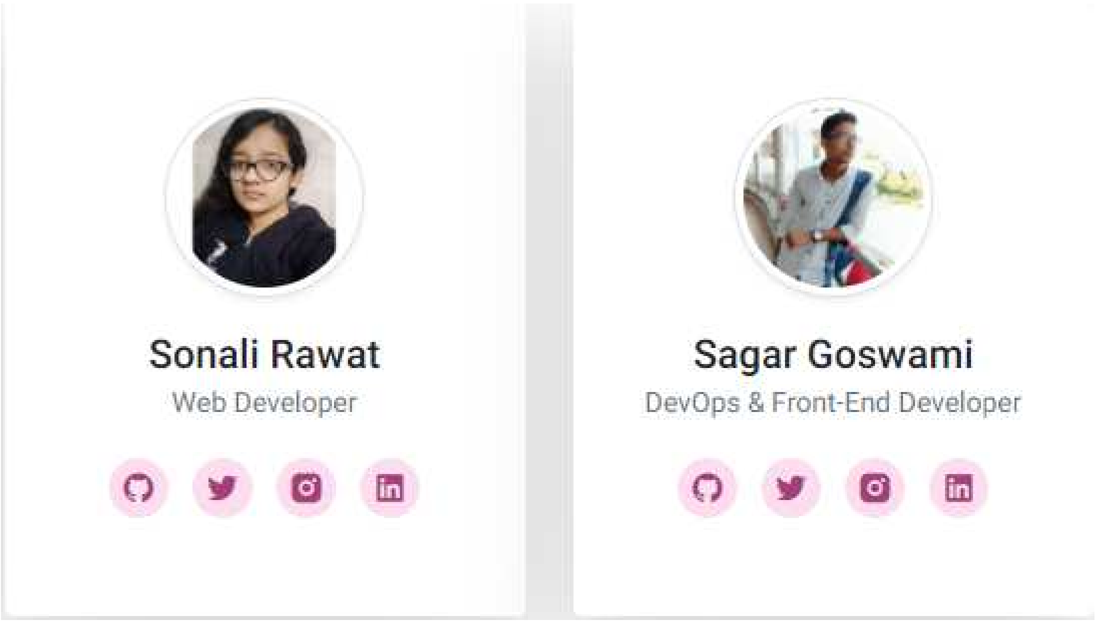


Prakash

DevOps & Front-End Developer

Teesha Singhal

Web Developer



Sonali Rawat

Web Developer

Sagar Goswami

DevOps & Front-End Developer