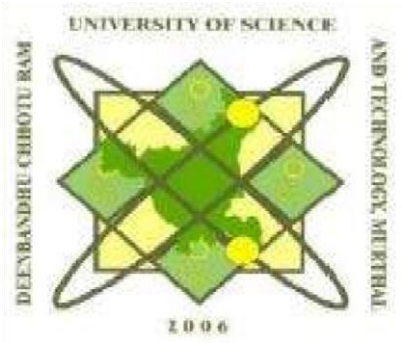
PROJECT REPORT
ON

Forms World

Submitted in the partial fulfillment of requirement for the award of degree of Bachelor of Computers Applications
Session (_2020-23_)

Under supervision of
Ms. Parull Bhardwaj
Assist. Professor

Submitted by:-
Shubham Dahiya
20012041042



DEPARTMENT OF COMPUTER SCIENCE
DEENBANDHU CHHOTU RAM UNIVERSITY OF SCIENCE AND TECHNOLOGY
MURTHAL, SONIPAT

Title of the Project

Forms World

Abstract

When in final semester we were told to build a Minor-Project, the first thing which into my mind is to build an application that people could use to solve their problems based on some solution for real-world. Like there's Zomato for solving the problem of going outside for food. Uber for bringing up the gap between travelers and cab drivers. And I always use them and wandered about their whole system works, like so many people use them. So I decided to build one.

"Forms World" is an Application where users from all over India can apply for various Job and Govt. form applications to save them time. Forms World is a service provide platform that provide to people to fill their forms with the data which once they were given during KYC in the start. It makes users to directly apply to the form application if he is eligible without need of any hard-work.

Acknowledgement

It brings me great pleasure to finally complete the minor project in partial fulfillment of the prerequisite for the award of Bachelor computer application. I would like to thank Ms. Parull mam (project in-charge, BCA department) who was involved with me in this project to make it a success. I extend my heartfelt and profound gratitude to my respected faculty members for providing me the impulse to get this work on the ground.

Date:

Mr. Shubham Dahiya
(20012041042)

Certificate

This Is to certify that Project Report entitled “Forms World“ which is submitted by Shubham (20012041042) has been successfully completed and submitted In partial fulfillment of the requirement for the requirement for the award of Degree Bachelor of Computer Applications In Computer Science, Hindu Institute Of Management, DCRUST (Deenbandhu Chotu Ram University Of Science And Technology), Murthal, Sonipat. Is a record of the candidate’s work carried out by him under the supervision of undersigned. The matter embodied in this report is original and has not been submitted for the award of any other degree.

This Performance was good during the tenure.

Date: **Ms. Parull Bhardwaj**

Declaration

I here declare that all the work presented in this project report entitled 'Forms World' in the partial fulfillment of the requirement for the award of the degree of BCA (Bachelor of Computer Application), Hindu institute of management, DCRUST (Deenbandhu Chotu Ram University of Science and Technology), Murthal is and authentic record of our own work carried out under the guidance of Ms. Parull Bhardwaj Mam.

Date:

Mr.Shubham Dahiya
(20012041042)

Table of Contents

S. No	Contents	Page No.
1	**Introduction to Technologies Used**	
	1.1 Introduction to HTML	1
	1.2 Introduction to CSS	2
	1.3 Introduction to React JS	3
	1.4 Introduction to Node JS	4
	1.5 Introduction to Express JS	5
	1.6 Introduction to Mongo DB	5
	1.7 introduction to Firebase	6
2	**System Analysis**	
	2.1 Introduction	8
	2.2 Identification of Need	9
	2.3 Preliminary Investigation	10
3	**Feasibility Study**	
	3.1 Introduction	12
	3.2 Goals & Objectives	14
	3.3 Technical Feasibility	15
	3.4 Potential Risks	16
	3.5 Economic Feasibility	17
4	**System Analysis**	
	4.1 Introduction	19
	4.2 Functional Requirements	20
	4.3 Non-Functional Requirements	22
5	**Project Monitoring System - Planning**	
	5.1 Introduction	24
	5.2 Gantt Chart	25
6	**System Requirement and Specification**	
	6.1 Introduction	27
	6.2 Hardware Requirements	28
	6.2 Software Requirements	28
7	**System Design**	
	7.1 Introduction	29
	7.2 Architecture	30
	7.3 Components	30
8	**Flow Chart**	33

9	**Data Modelling Description** 9.1 Introduction 9.2 List of Tables 9.3 ER Diagram 9.4 Entity Relationship Dictionary	 35 35 38 42
10	**Data Flow Diagram**	45
11	**Project Design** 11.1 Database Design 11.2 Screen Design	 49 52
12	**Testing**	65
13	**System Implementation** 13.1 Introduction 13.2 System Architecture 13.3 Front-End Implementation 13.4 Backend Implementation 13.5 Database Implementation 13.6 Deployment and Hosting	 100 101 102 104 105 107
14	**Conclusion**	108
15	**Future Enhancements**	109

List of Tables

5.1 Gantt Chart of Forms World……26

9.1 Users Database Model……35

9.2 Users KYC Model……36

9.3 Forms Model……36

9.4 Orders Model……37

9.5 Payments Model……37

12.1 Signup and Login Testing……68

12.2 User kyc Testing……75

12.3 Updating Password Testing……85

12.4 Signout Testing……86

12.5 Services Testing……86

12.6 Admin/Active Testing……90

12.7 Admin/Orders Testing……96

List of Figures

3.1 Feasibility Study Phases……..12

7.1 Flow Chart of Forms World……..34

6.1 Entity Relationship diagram……39

6.2 Entity Relationship diagram Continued……40

9.1 0 level – DFD……46

9.2 1st level – DFD.……..47

9.3 2nd level – DFD.……48

12.1 Types of Testing……65

12.2 Software Testing Strategies……67

Chapter 1

INTRODUCTION TO TECHNOLOGIES USED

(Tech-Stack)

The following technologies were employed in the development of the "Forms World" web application:

HTML, CSS, Node JS, React JS, Express JS, and Mongo DB are all included.

1.1 Introduction to HTML

HTML, or Hypertext Markup Language, is the standard markup language used to create web pages. It is the foundation of the web, and all web pages are written in HTML. In Forms World, HTML is used to create the structure and content of the web pages, providing a clear and organized layout for users to fill out forms and submit job applications.

One of the main advantages of using HTML is its simplicity. HTML is a straightforward and easy-to-learn language, with a basic syntax that consists of opening and closing tags. This makes it accessible to beginners and experts alike, and allows for quick and efficient creation of web pages.

Another advantage of HTML is its flexibility. HTML allows for the creation of a wide variety of web pages, from simple text-based pages to complex multimedia pages that incorporate images, videos, and interactive features. This flexibility allows developers to create custom and engaging user experiences, which is important for web applications like Forms World.

HTML also allows for the creation of accessible web pages that are easily navigable by users with disabilities. By using semantic HTML tags and properly

labeling form elements, developers can ensure that users with screen readers and other assistive technologies can easily navigate and interact with the forms on the website.

Overall, HTML is a crucial component of the web development process, providing the foundation for web pages and ensuring that they are accessible and user-friendly. In Forms World, HTML is used to create the structure and content of the web pages, providing a clear and organized layout for users to fill out forms and submit job applications.

1.2 Introduction to CSS

CSS, or Cascading Style Sheets, is a style sheet language used to describe the visual appearance of web pages. It allows developers to control the layout, colors, fonts, and other visual elements of a web page, ensuring that the web page is visually appealing and consistent across different devices and browsers.

One of the main advantages of using CSS is its ability to separate the presentation of a web page from its content. This allows developers to create a consistent visual style across a website, and to make changes to the visual style without having to edit each individual web page. Additionally, by separating the presentation from the content, web pages can be loaded more quickly, improving the overall performance of the website.

Another advantage of CSS is its flexibility. CSS allows developers to create responsive web pages that adapt to different screen sizes and devices. This is important for web applications like Forms World, which need to be accessible and usable on a wide variety of devices, including desktop computers, laptops, tablets, and smartphones.

CSS also allows for the creation of accessible web pages that are easily navigable by users with disabilities. By using proper color contrast, font sizes, and layout techniques, developers can ensure that users with visual impairments and other disabilities can easily navigate and interact with the forms on the website.

Overall, CSS is a crucial component of the web development process, providing the visual style and layout of web pages and ensuring that they are accessible and responsive across different devices and browsers. In Forms World, CSS is used to create a consistent and visually appealing style across the website, and to ensure that the website is accessible and usable for all users, including those with disabilities.

1.3 Introduction to Node.js

Node.js is a server-side JavaScript runtime environment that allows developers to build scalable and efficient backend systems. It is used in Forms World to handle HTTP requests, manage sessions, and integrate with MongoDB.

One of the main advantages of using Node.js is its performance. Node.js uses an event-driven, non-blocking I/O model, which allows it to handle a large number of simultaneous connections without blocking the event loop. This makes it a great choice for building scalable and efficient backend systems, especially for web applications.

Another advantage of Node.js is its ease of use. Node.js has a simple and intuitive API that allows developers to quickly build and deploy backend systems. Additionally, Node.js has a large and active community, which means that there are many libraries and modules available that can help developers build web applications faster and more efficiently.

In Forms World, Node.js is used to handle HTTP requests, manage sessions, and integrate with MongoDB. By using Node.js, Forms World is able to provide a reliable and efficient backend system for users to fill out forms and submit job applications.

1.4 Introduction to React

Node.js is a server-side JavaScript runtime environment that allows developers to build scalable and efficient backend systems. It is used in Forms World to handle HTTP requests, manage sessions, and integrate with MongoDB.

One of the main advantages of using Node.js is its performance. Node.js uses an event-driven, non-blocking I/O model, which allows it to handle a large number of simultaneous connections without blocking the event loop. This makes it a great choice for building scalable and efficient backend systems, especially for web applications.

Another advantage of Node.js is its ease of use. Node.js has a simple and intuitive API that allows developers to quickly build and deploy backend systems. Additionally, Node.js has a large and active community, which means that there are many libraries and modules available that can help developers build web applications faster and more efficiently.

In Forms World, Node.js is used to handle HTTP requests, manage sessions, and integrate with MongoDB. By using Node.js, Forms World is able to provide a reliable and efficient backend system for users to fill out forms and submit job applications.

1.5 Introduction to Express

Express is a web application framework for Node.js that provides a set of tools and utilities for building web applications. It simplifies the process of handling HTTP requests, managing sessions, and integrating with MongoDB, making it an ideal choice for building the backend of Forms World.

One of the main advantages of using Express is its simplicity. Express provides a minimal and flexible set of tools and utilities, which allows developers to build web applications quickly and efficiently. Additionally, Express is modular, which means that developers can choose which components to use based on their specific needs.

Another advantage of Express is its compatibility with Node.js. Because Express is built on top of Node.js, it is able to take advantage of Node.js' non-blocking I/O model, which allows it to handle a large number of simultaneous connections without blocking the event loop. This makes Express a great choice for building scalable and efficient web applications.

In Forms World, Express is used to handle HTTP requests, manage sessions, and integrate with MongoDB. By using Express, Forms World is able to provide a reliable and efficient backend for users to fill out forms and submit job applications.

1.6 Introduction to MongoDB

MongoDB is a popular NoSQL database that stores data in JSON-like documents, making it easy to work with data in a flexible and scalable way. Unlike traditional relational databases, MongoDB does not use tables, rows, or columns to store data. Instead, it uses a collection of documents that can have any number of

fields and be of any shape. This makes it easier to work with complex and unstructured data.

One of the main advantages of using MongoDB is its scalability. MongoDB is designed to scale horizontally, meaning that it can handle large amounts of data and traffic by adding more servers to the database cluster. This makes it a great choice for applications that need to handle a lot of data, like Forms World. Additionally, MongoDB has a built-in sharding feature, which allows data to be distributed across multiple servers, further improving performance and scalability.

Another advantage of MongoDB is its flexibility. Because documents can have any number of fields and be of any shape, it is easy to add new fields or change the structure of existing documents as needed. This makes it a great choice for applications that need to be agile and adapt to changing requirements.

In Forms World, MongoDB is used to store data related to user accounts, job applications, and other forms. By using MongoDB, Forms World is able to handle large amounts of data and traffic while still providing a flexible and scalable platform for users to fill out forms and submit job applications.

1.7 Firebase:

Firebase is a cloud-based platform provided by Google that offers a number of features, including real-time database, cloud storage, authentication, and hosting. It is used in Forms World to deploy and host the website, making it scalable, reliable, and secure.

One of the main advantages of using Firebase is its ease of use. Firebase has a simple and intuitive interface that allows developers to quickly deploy and host

web applications. Additionally, Firebase provides a number of features, such as real-time database and cloud storage, that make it easy to store and manage data.

Another advantage of Firebase is its scalability. Firebase is designed to handle large amounts of traffic and data, making it a great choice for web applications that need to scale quickly. Additionally, Firebase provides automatic scaling, which means that resources are allocated based on the current traffic and usage patterns.

In Forms World, Firebase is used to deploy and host the website. By using Firebase, Forms World is able to provide a scalable, reliable, and secure platform for users to fill out forms and submit job applications.

Overall, the MERN stack and Firebase provide a powerful and flexible platform for building web applications. By using MongoDB, Express, React, Node.js, and Firebase, Forms World is able to handle large amounts of data and traffic while still providing a fast, responsive, and secure platform for users to fill out forms and submit job applications.

Chapter 2

System Analysis

2.1 Introduction:

The internet has revolutionized the way we work, communicate, and conduct business. In today's fast-paced world, people are always looking for ways to simplify their lives and save time. This is where our website comes in – we provide form filling services for various jobs, making it easy for people to complete important documents and applications quickly and efficiently.

The purpose of this system analysis report is to provide a detailed overview of our website and the system requirements necessary to support its functionality. This report will cover the scope of the project, the system requirements, the functional and non-functional requirements, use cases, data flow diagrams, system architecture, and risk mitigation strategies.

Our website is designed to be user-friendly and easy to navigate, with a simple and intuitive interface that allows users to quickly fill out forms and submit them with ease. By providing this valuable service, we aim to simplify people's lives and help them achieve their goals more efficiently.

We hope that this system analysis report will serve as a valuable resource for anyone interested in our website and the technology behind it. By providing a comprehensive overview of our system, we hope to inspire confidence in our users and stakeholders and help them understand the value of our services.

2.2 Identification of Need:

The need for a website that provides form filling services for various jobs arises from the growing demand for digital services and the need for people to complete important documents and applications quickly and efficiently. With the rise of the internet and the digitalization of many services, people are increasingly looking for ways to complete their tasks online without having to visit physical offices or mail in documents.

Furthermore, the process of filling out forms and applications can often be time-consuming and confusing, requiring users to navigate complex processes and requirements. This can be especially challenging for people who are not familiar with the process or who have limited access to resources such as computers, printers, and scanners.

Our website addresses these needs by providing a simple and intuitive interface that allows users to quickly fill out forms and applications online. By eliminating the need for physical forms and paper documents, we help users save time, reduce frustration, and streamline the process of completing important tasks.

In addition, our website offers a range of services that are tailored to the needs of different users, including job seekers, business owners, and individuals applying for government services. By offering a one-stop-shop for form filling services, we make it easy for users to find the forms they need and complete them quickly and efficiently.

Overall, the need for our website arises from the growing demand for digital services and the need for a simple and user-friendly platform that can help users complete important tasks more efficiently.

2.3 Preliminary Investigation:

Before beginning the development of our website, we conducted a preliminary investigation to determine the feasibility of the project and identify any potential challenges or roadblocks. This investigation involved a thorough analysis of the market demand, the target audience, and the technology requirements necessary to support the website's functionality.

Market Demand: We conducted market research to determine the demand for form filling services and identify any potential competitors in the market. Our research revealed that there is a growing demand for digital services and that many users are looking for a simple and user-friendly platform that can help them complete important forms and applications quickly and efficiently.

Target Audience: We identified the target audience for our website as individuals, job seekers, business owners, and government agencies who need to complete various forms and applications. By targeting this specific audience, we can tailor our services to their needs and provide a platform that meets their requirements.

Technology Requirements: We conducted an analysis of the technology requirements necessary to support our website's functionality, including hardware, software, and network requirements. We determined that we would need a reliable web hosting service, a database management system, and a variety of programming languages and tools to develop the website.

Overall, our preliminary investigation revealed that there is a strong market demand for form filling services and that our target audience is in need of a simple and user-friendly platform to complete important forms and applications. We also identified the technology requirements necessary to

support the website's functionality, which will be crucial to the successful development and deployment of our website.

Chapter 3

Feasibility Study

3.1 Introduction:

After choosing a project, a feasibility investigation is the next phase in the project development process.

Phases of Feasibilitiy Study:

Fig.3.1 Feasibility Study Phases

1. Plan: we plan each module in this step like signup login logout and post adding, post saving.
2. Do: We analyze and collect data to implement the modules.
3. Study: We study the prosed change with the current solution. Like updating the signup details.
4. Act: if the change was good then we implement the change and publish it on service.

The development of a website that provides form filling services for various jobs requires a feasibility study to assess the viability of the project. The website is developed using the MERN stack, which includes MongoDB, Express, React, and Node.js. The feasibility study will evaluate the technical, economic, social, legal, and organizational aspects of the project.

The objective of the feasibility study is to determine whether the project is feasible, taking into consideration the resources, timelines, and risks associated with its development and operation. The study will provide a comprehensive analysis of the project and will help in identifying potential challenges and risks, as well as their mitigation strategies.

This feasibility study will be divided into various sections that will examine the different aspects of the project, including technical feasibility, economic feasibility, social feasibility, legal feasibility, and organizational feasibility. The study will also provide a recommendation on the viability of the project and suggest any further steps that need to be taken before proceeding with the development of the website.

Overall, the feasibility study will be an important tool for making informed decisions regarding the development of the website. It will help in identifying potential challenges, assessing the feasibility of the project, and developing strategies to mitigate any risks associated with the project. With this feasibility study, the project team will be able to make an informed decision about the feasibility of the project and ensure that it is developed in a manner that is both cost-effective and sustainable.

3.2 Goals & Objectives

1. To provide a user-friendly interface that allows users to easily access and complete job application forms.
2. To ensure that the website is secure and user data is protected.
3. To provide accurate and up-to-date information about various job application processes.
4. To ensure that the website is accessible to a diverse range of users, including those with disabilities.
5. To optimize the website for search engines to increase visibility and attract more users.
6. To develop a sustainable business model that generates revenue from the form filling services provided.
7. To establish partnerships with employers and organizations that can help promote the website and its services.
8. To continuously improve the website and its services based on user feedback and analytics.
9. To maintain high levels of customer satisfaction and retention.
10. To expand the range of services offered on the website over time to include additional form filling services for various types of applications.

3.3 Technical Feasibility:

The technical feasibility of the project involves assessing the availability of technology, infrastructure, and resources required to develop and maintain the website. This section will examine the technical requirements of the website and assess the feasibility of meeting them within the project timeline and budget.

Technical Requirements:

The website is developed using the MERN stack technology, which includes MongoDB, Express, React, and Node.js. The technical requirements for the website include the following:

1. Front-end development using React.js
2. Back-end development using Node.js and Express.js
3. Integration with MongoDB as the database for storing user data and other information
4. Implementation of security measures such as HTTPS, encryption, and user authentication to protect user data
5. Integration with third-party services such as payment gateways and email providers
6. Optimization of website performance and scalability to accommodate increasing user traffic and data volume.

Feasibility Assessment:

To assess the technical feasibility of the project, the availability of the technology, infrastructure, and resources needed to develop and maintain the website will be evaluated. The assessment will consider the following factors:

1. Availability of qualified technical staff with expertise in the MERN stack technology
2. Availability of technology infrastructure such as servers, storage, and networking
3. Availability of third-party services and APIs that can be integrated into the website
4. Feasibility of meeting technical requirements within the project timeline and budget.

3.4 Potential Risks:

The technical feasibility assessment will also identify potential risks associated with the development and operation of the website. Risks may include technical challenges such as software bugs, system failures, and data breaches. To mitigate these risks, the following strategies may be implemented:

1. Regular testing and quality assurance to identify and fix software bugs.
2. Implementation of disaster recovery and backup procedures to mitigate system failures
3. Implementation of security measures such as encryption, firewalls, and user authentication to protect user data
4. Regular monitoring and updating of the website to ensure that it is optimized for performance and scalability.

Overall, the technical feasibility assessment will provide valuable insights into the feasibility of developing and maintaining the website. The assessment will help identify potential challenges and risks, as well as their mitigation strategies, to ensure that the website is developed in a technically feasible manner.

3.5 Economic Feasibility:

The economic feasibility of the project involves evaluating the costs and benefits associated with the development and maintenance of the website. This section will assess the viability of the project from a financial perspective by estimating the initial and ongoing costs of development, marketing, and maintenance of the website, as well as the potential benefits that can be derived from it.

Cost-Benefit Analysis:

A cost-benefit analysis will be conducted to evaluate the economic feasibility of the project. The analysis will compare the expected costs of development, marketing, and maintenance of the website against the expected benefits that can be derived from it. The benefits may include revenue generated from form filling services, cost savings for users who utilize the website, and other potential benefits such as increased user engagement and satisfaction.

Cost Estimation:

The cost estimation for the website will consider the following factors:

1. Development costs: including the cost of hiring technical staff, software licenses, and hardware infrastructure required for development.
2. Marketing costs: including the cost of advertising and promoting the website to potential users.
3. Maintenance costs: including the cost of ongoing maintenance and updates to the website to ensure that it remains secure, up-to-date, and optimized for performance.

Revenue Generation:

The revenue generation potential of the website will depend on the pricing model and user adoption rate. The pricing model for the form filling services can

be structured on a per-form basis or a subscription basis. Revenue can also be generated through partnerships with employers and organizations that require form filling services for their job applications.

Risk Assessment:

To assess the economic feasibility of the project, potential risks associated with the development and operation of the website will be identified. Risks may include lower than expected user adoption rate, unexpected costs, and competition from other similar websites. The mitigation strategies for these risks can be included in the risk assessment plan.

Conclusion:

The economic feasibility analysis will provide valuable insights into the viability of the project. It will help identify potential challenges and risks and their mitigation strategies, and provide a framework for decision-making regarding the financial aspects of the project.

Chapter 4

Software Requirement Specification - SRS

4.1 Introduction

In today's digital age, the need for efficient and reliable online services has become more crucial than ever. The internet has become an integral part of our lives, and people rely on it for various tasks, from socializing to conducting business. With the rise of online transactions, the need for secure and user-friendly form filling services has also increased.

Our project aims to provide a reliable and efficient solution to this problem. We are developing a website that offers form filling services for various job applications. Our website will simplify the application process by providing users with pre-filled forms that they can customize to their specific requirements. The website will also provide users with a platform to submit their applications online, eliminating the need for paper-based applications.

This project report aims to provide a comprehensive overview of the software requirements for our website. It will cover all the functional and non-functional requirements, system architecture, data model, user interface design, system testing, and project management. The report will also outline the benefits and key features of our website and provide insights into the development process.

We believe that our website will be a game-changer in the online job application process. By simplifying the application process and providing a secure platform for users, we aim to make the job application process more accessible and convenient. We hope that this project report will provide valuable insights into

the development of our website and help us achieve our goal of making the job application process more efficient and user-friendly.

4.2 Functional Requirements:

1. User Authentication:

- Users should be able to create an account and log in to the website.
- Users should be able to reset their passwords in case they forget them.

2. Form Pre-filling:

- Users should be able to select the job application they want to fill out.
- Users should be able to upload their resume or CV.
- The system should automatically pre-fill the application form with the information from the user's resume or CV.

3. Form Customization:

- Users should be able to edit the pre-filled form.
- Users should be able to add or remove sections or fields from the form.

4. Form Submission:

- Users should be able to review and finalize their application before submitting it.
- Users should be able to submit their application online through the website.

- The system should generate a confirmation message after the application is submitted.

5. Application Tracking:

- Users should be able to track the status of their application.
- Users should receive notifications when their application is received, reviewed, or accepted/rejected.

6. User Feedback:

- Users should be able to provide feedback on the application process.
- The system should provide a feedback mechanism for users to report any issues or errors they encounter.

7. Administration:

- Administrators should be able to manage user accounts and permissions.
- Administrators should be able to view and download submitted applications.
- Administrators should be able to generate reports on the application process.

These are some basic functional requirements for your form filling website. You can further refine and expand these requirements based on your specific needs and goals.

4.3 Non-Functional Requirements:

1. Performance:

- The website should be fast and responsive, with minimal lag time.
- The website should be able to handle a large number of users simultaneously without crashing.

2. Security:

- The website should be secure and protect user data from unauthorized access.
- The website should encrypt sensitive user data, such as passwords and personal information.
- The website should implement secure authentication and authorization mechanisms to prevent unauthorized access.

3. Usability:

- The website should be easy to use and navigate for users with different levels of computer proficiency.
- The website should have clear instructions and guidance on how to use the form filling services.
- The website should be accessible for users with disabilities, such as visually impaired users.

4. Compatibility:

- The website should be compatible with different web browsers and operating systems.

- The website should be designed to work on different screen sizes, including mobile devices.

5. Reliability:

- The website should be reliable and available 24/7, with minimal downtime for maintenance.
- The website should be able to recover quickly from any technical issues or errors.

6. Scalability:

- The website should be scalable and able to handle an increase in users or traffic over time.
- The website should be designed to accommodate future enhancements and upgrades.

These are some basic non-functional requirements for your form filling website. You can further refine and expand these requirements based on your specific needs and goals.

Chapter 5

Project Monitoring System - Planning

5.1 Introduction:

As the owner of Forms World, it's essential to have a project monitoring and planning system in place to ensure the smooth and efficient operation of your website. A project report is a critical tool for tracking the progress of your projects and keeping stakeholders informed of your progress. With a project monitoring and planning system, you can easily generate detailed project reports that highlight key metrics, milestones, and deliverables.

To create an effective project monitoring and planning system for your project report, consider the following steps:

1. Define your project goals and objectives: Before you can start tracking your progress, you need to establish clear goals and objectives for your project. These objectives ought to be SMART, or specific, measurable, achievable, relevant, and time-limited.
2. Break down your project into smaller tasks: Once you have established your project goals, you need to break down your project into smaller, more manageable tasks. Each task should be assigned to a specific team member or contractor, with a deadline for completion.
3. Set up a system for tracking progress: To track progress, you need to establish a system for monitoring and reporting on each task. This can be done using project management software or a spreadsheet.
4. Monitor your progress: As your project progresses, it's important to regularly monitor your progress and make adjustments as necessary.

5. Generate regular project reports: To keep stakeholders informed of your progress, you should generate regular project reports that highlight key metrics, milestones, and deliverables.

By implementing a project monitoring and planning system for your project report, you can ensure that your projects stay on track and achieve their goals. With detailed progress reports, you can keep stakeholders informed and make data-driven decisions to improve your project's success.

5.2 Gantt Chart

A Gantt chart is an effective tool for visualising a project's timeline and tracking its advancement towards completion. It offers a graphical depiction of the project schedule, with each task represented by a horizontal bar that spans the time needed to complete it. The chart makes it simple for team members and project managers to understand the order of tasks and their interdependencies, as well as to monitor progress and modify timetables as necessary.

Henry Gantt invented the Gantt chart in the early 20th century in an effort to boost manufacturing productivity.

The Gantt chart helps project managers and team members keep organised and motivated by offering a clear, simple-to-read visual representation of project timeframes and dependencies, lowering the risk of missed deadlines or unfinished activities.

Forms World Gantt Chart:

	February				March				April			
Requirements Gathering												
Analysis												
Design												
Coding												
Testing												
Implementation												
	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12

Table 5.1 Gantt chart of Forms World

Chapter 6

System Requirements & Specifications

6.1 Introduction:

When we are about to build some software project, then we have some requirements. We write down them in a document. That document is known as SRS i.e. software requirement and specification document.

Basically, SRS describe the requirements associated with project. And specification of the software functionality.

In general words, SRS is the picture of our entire project. SRS have in-depth description of the project.

SRS describe the requirements and describe how to overcome from the requirements. It describes all resources associated with project and how to overcome out of them.

Functions of Software Requirement and specification

- SRS defines the internal calculations
- It describes the internal details
- It works as an agreement between client and company.

6.2 Hardware Requirements are as follows

RAM	:	4 GB or More
Processor	:	Intel i3 or Later
Hard disk	:	51GB or more
Monitor	:	Any Compatible Monitor
Keyboard & Mouse	:	Any QWERTY Keyboard and compatible Mouse

6.3 Software Requirements are as follows

Front-End	:	HTML, CSS, JavaScript, Bootstrap, React JS
Backend	:	Express JS, Node JS
Database	:	Mongo DB
IDE	:	VS Code
Web-Browser	:	Mozilla Firefox, Google Chrome, Microsoft Edge
Window	:	Linux, Windows 7 & later supporting Node JS.

Chapter 7

System Design

7.1 Introduction

In today's digital age, job applications are increasingly done online, and filling out these forms can be a daunting task for many job seekers. As such, there is a growing demand for services that assist with filling out job applications. This is where our website comes in - we provide form filling services for various jobs, making the job application process easier and more convenient for users.

Our website is built using the MERN stack, which includes MongoDB, Express.js, React.js, and Node.js, and is deployed on Firebase. The MERN stack provides a modern and robust set of tools for building web applications, and Firebase provides an easy and reliable platform for hosting and deploying web applications.

In this system design, we will outline the key components and architecture of our website, including how user data is stored and processed, how payments are handled, and how website content is managed. We will also discuss how Firebase's built-in scalability features will allow our website to handle increased traffic and user demand.

By providing form filling services for various jobs, our website aims to make the job application process less stressful and more accessible to all job seekers. Our system design will outline the technical details of how we achieve this goal, ensuring that our website is reliable, scalable, and secure for our users.

7.2 Architecture

Your website will follow a typical client-server architecture, where the client will be the user's web browser and the server will handle requests and responses. The server-side will be built using the MERN stack, which includes:

- MongoDB: A NoSQL document-oriented database to store user data and website content.
- Express.js: A web application framework for Node.js that will handle routing and server-side logic.
- React.js: A JavaScript library for building user interfaces that will handle the client-side.
- Node.js: A JavaScript runtime that will power the server.

Firebase will be used as the hosting and deployment platform for your application.

7.3 Components

The main components of your website will include:

User Interface

The user interface will be built using React.js and will provide a clean and intuitive experience for users. It will include pages for users to view available jobs, fill out job applications, and submit payment for services rendered. Users will also be able to create an account and manage their profile information.

Application Processing

When a user submits a job application, the application data will be stored in MongoDB. A server-side endpoint will handle the processing of the application

and will perform any necessary validation and verification steps. If the application is valid, the server will generate an invoice for the user and provide a payment link.

Payment Processing

When a user submits payment for services rendered, the payment data will be securely transmitted to a payment processor such as Stripe. Once payment is processed, the server will mark the job application as complete and notify the user.

Content Management

Your website will require a content management system (CMS) to allow you to manage website content such as job descriptions and pricing information. The CMS will be built using Express.js and will allow you to make changes to website content without requiring any code changes.

Authentication and Security

To ensure the security of user data, your website will use Firebase Authentication to handle user authentication. All sensitive data will be encrypted using a secure encryption algorithm such as AES.

Scaling

Firebase provides built-in scalability features that can handle increased traffic and user demand. These features include:

- Auto-scaling: Automatically scale your application resources to match demand.
- Global CDN: Serve your application content from a global content delivery network for improved performance.

- Real-time database: Use Firebase's real-time database to handle real-time data synchronization and collaboration.

Conclusion

In conclusion, your MERN stack based website hosted on Firebase will provide a clean and intuitive user experience for job seekers. By leveraging modern web development techniques and technologies, your website will be able to handle large volumes of traffic and user demand, ensuring that your users have a smooth and reliable experience. Firebase's built-in scalability features will allow your website to handle growth and continue to provide a high-quality experience for users.

Chapter 8

Flow Chart

A flowchart is a visual representation of the steps in a process or system. It provides a step-by-step illustration of the various stages involved in completing a task or achieving a goal. In the context of a project report, a flowchart can be a powerful tool for communicating the workflow of the project and identifying areas for improvement.

The use of a flowchart in a project report can help to make complex processes more accessible and easy to understand. By breaking down the project into its individual steps and visually representing the flow of tasks, a flowchart can help stakeholders and team members quickly grasp the project's objectives and requirements. This can help to ensure that everyone involved in the project is on the same page and working towards the same goals.

In addition to providing clarity, a flowchart can also help to identify potential bottlenecks or inefficiencies in the project. By examining the flow of tasks and identifying areas where tasks are taking longer than expected or where there is a backlog, project managers can make data-driven decisions about how to optimize the workflow and improve the project's efficiency.

When creating a flowchart for a project report, it's important to keep the chart clear and concise. Use clear and concise labels for each step, and use arrows to show the flow of tasks. Avoid cluttering the chart with unnecessary details, and focus on the key steps in the project workflow.

Overall, a flowchart can be an invaluable tool for project managers and team members seeking to optimize their project workflow and achieve their

objectives in a timely and efficient manner. A flowchart can help to ensure the success of the project and the satisfaction of stakeholders.

Here's Forms World Flowchart:

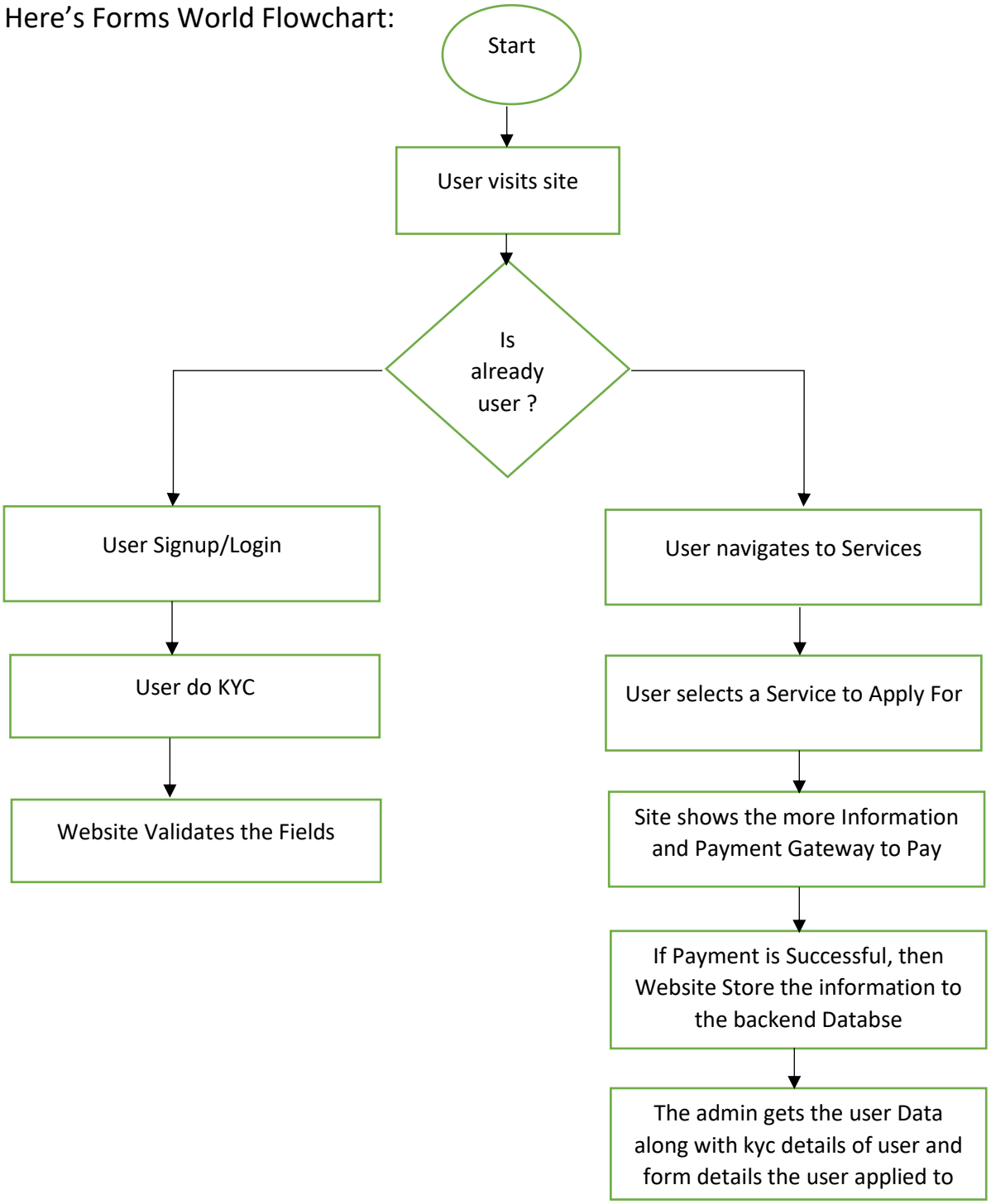


Fig. 7.1 Flow Chart of Forms World

Chapter 9

Data Modelling Description

9.1 Introduction

A data model is the conceptual representation of the data structures of that are required by a database. It defines primary data objects, composition, of each data object and attributes of the project, relationships between each object and other object and between objects and processes.

9.2 List of Tables

1. Users
2. KYC
3. Forms
4. Orders
5. Payments

1. Users

Name	Type
Uid	Varchar(255)
Name	Varchar(255)
Email	Varchar(255)
Phone	Varchar(255)
Password	Varchar(255)

Table 9.1 Users Database Model

2. **KYCs**

Name	**Type**
KID	Varchar(255)
Uid	Varchar(255)
Full Name	Varchar(255)
Aadhar No.	Varchar(255)
Mothers Name	Varchar(255)
Passport Image	Varchar(255)
Signature Image	Varchar(255)
Marksheet 10^{th}	Varchar(255)
Marksheet 12^{th}	Varchar(255)
Domicile Image	Varchar(255)
Caste Certificate	Varchar(255)
Left thumb	Varchar(255)
Right thumb	Varchar(255)

Table 9.2 Users KYC Model

3. **Forms**

Name	**Type**
Fid	Varchar(255)
Title	Varchar(255)
Last Date	DateTime
Total Post	Varchar(255)
Age	Varchar(255)
Qualification	Varchar(255)
Price	Number

Table 9.3 Forms Model

4. **Orders**

Name	Type
Order Id	Varchar(255)
Form Id	Varchar(255)
Uid	Varchar(255)
DateTime	DateTime
Payment Done	Boolean

Table 9.4 Orders Model

5. **Payments**

Name	Type
Pid	Varchar(255)
Order Id	Varchar(255)
Amount	Varchar(255)
Currency	Varchar(255)
Razorpay Order Id	Varchar(255)
Razorpay Payment Id	Varchar(255)
Razorpay Signature	Varchar(255)
Created At	DateTime

Table 9.4 Payments Model

Main Description of the module

All the above mentioned data are stored in the backend and will be used by the designed algorithms.

9.3 ER Diagram

Entity-relationship (ER) diagrams are an essential tool for designing databases and understanding the relationships between entities. An ER diagram is a graphical representation of entities, their attributes, and the relationships between them. ER diagrams are commonly used in database design to illustrate the logical structure of a database and to help ensure that the database is structured in a way that meets the needs of its users.

In this project, we have created an ER diagram for a website that provides form filling services for various jobs. The website allows users to request form filling services for a range of jobs, from visa applications to government forms. The ER diagram shows the entities involved in the system, including users, jobs, forms, and fields, as well as the relationships between these entities.

The goal of the ER diagram is to provide a clear and concise representation of the database schema and to help ensure that the database is structured in a way that is efficient, accurate, and easy to use. By using an ER diagram, we can better understand the relationships between the entities involved in the system and ensure that the database is designed in a way that meets the needs of its users.

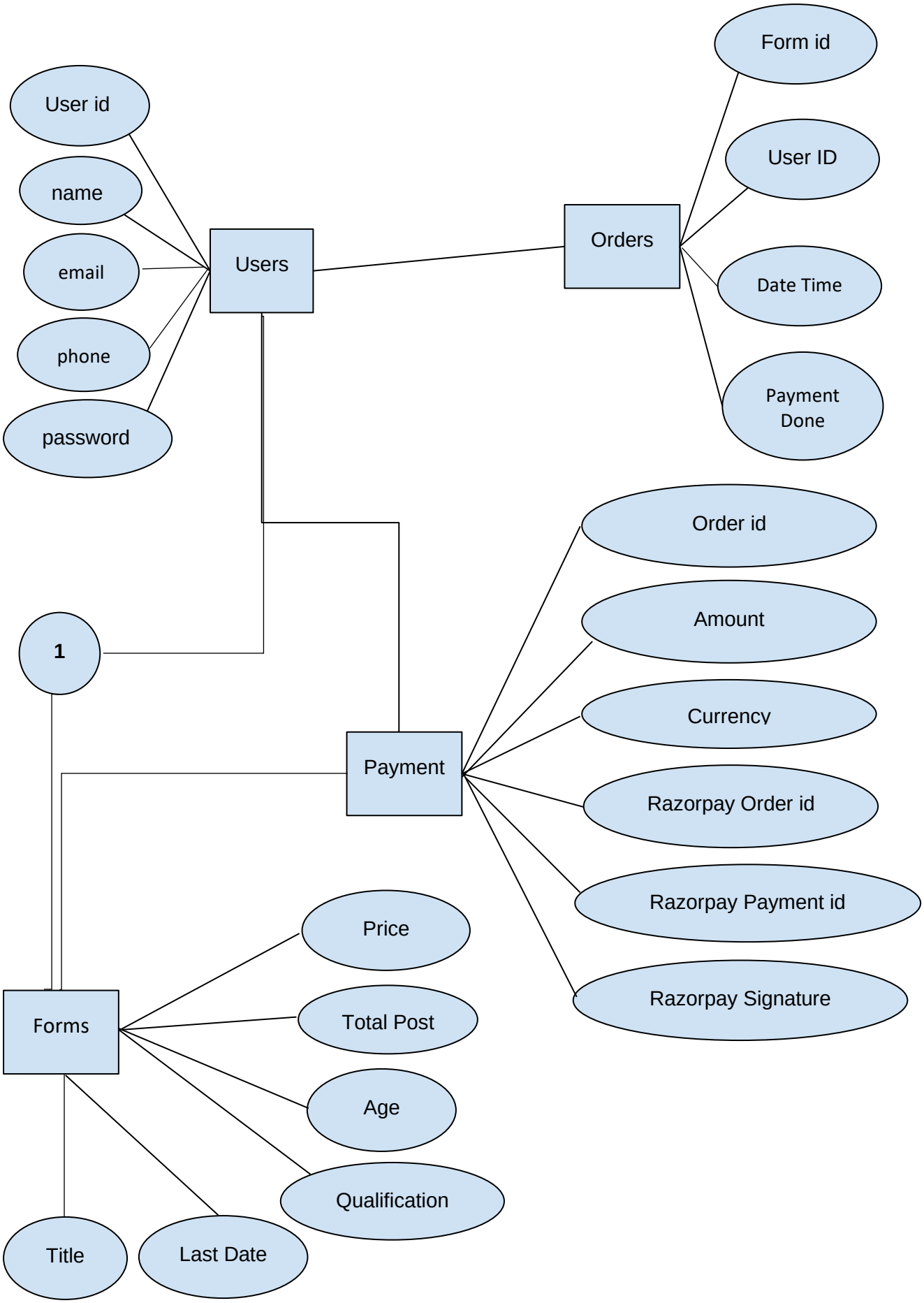


Fig. 6.1 Entity Relationship diagram

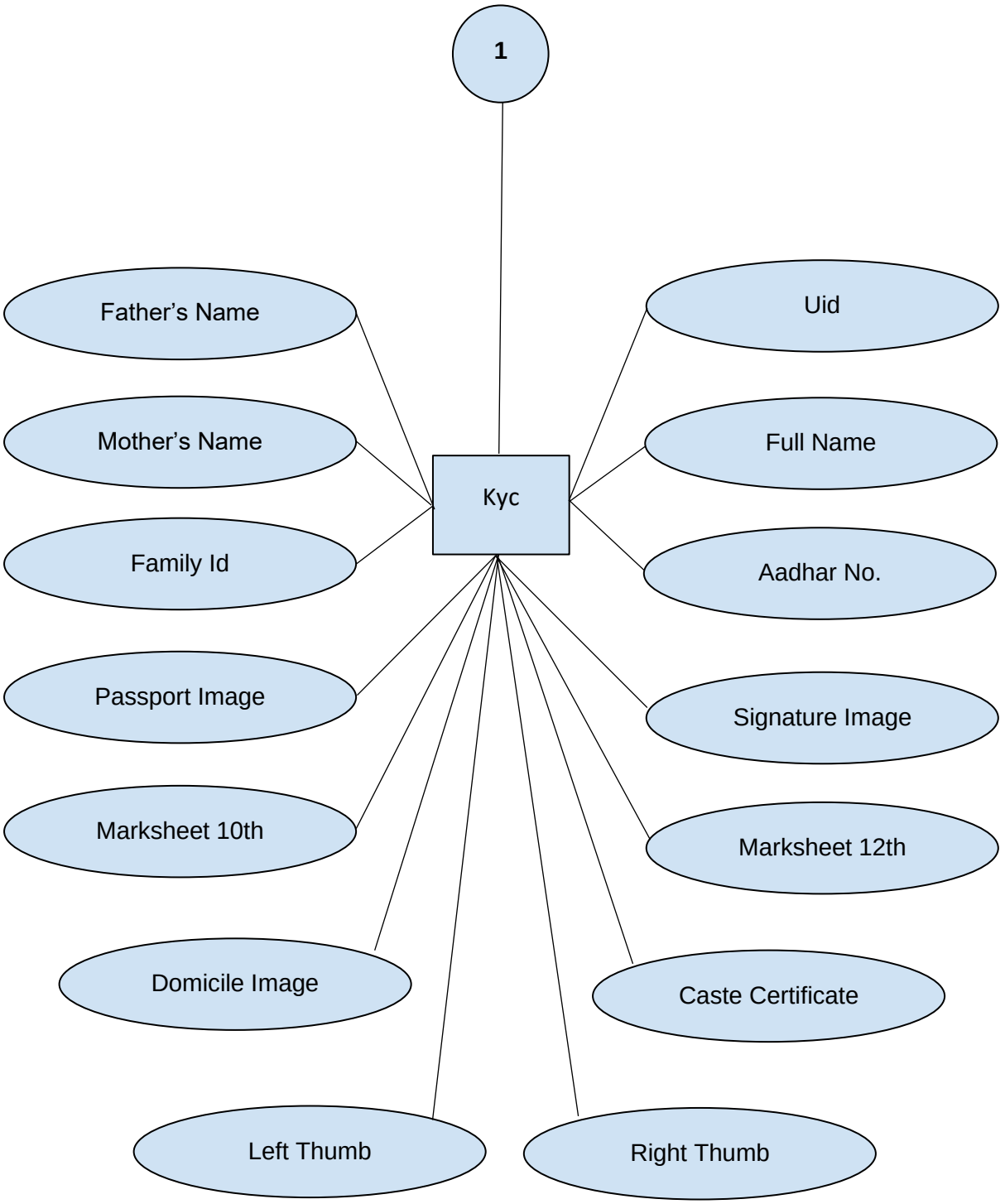


Fig. 6.2 Entity Relationship diagram Continued

The ER diagram shows that:

- The forms model has a one-to-many relationship with the orders model, as one form can have multiple orders.
- The orders model has a many-to-one relationship with both the forms and users models, as an order belongs to one form and one user, but one user and one form can have multiple orders.
- The payment model has a one-to-one relationship with the orders model, as each order can have only one payment.
- The users model has a many-to-one relationship with the orders model, as one user can have multiple orders, but each order belongs to only one user.

9.4 Entity Relationship Dictionary

1. User

A user can create account in Forms World and can login and access the services we provide there.

Attributes:

Name:	name of the user
Email:	email of the user
Phone:	phone number of the users
Password:	password of the user
User id (uid):	user id of the user

2. KYC

A user have to do KYC in order to successfully apply to a service.

Attributes:

uid:	user_id of user from users database
full_name:	password filled when signing-up
aadhar_no:	user's aadhar no.
fathers_name:	user's father's name
mothers_name:	user's mother's name
passport_image:	user's passport size image
signature_image:	user's signature image
marksheet_10th:	user's 10^{th} marksheet image
marksheet_12th:	user's 12^{th} marksheet image
domicile_image:	domicile image of user

caste_certificate: caste certificate image of user

left_thumb: left thumb impression image

right_thumb: right thumb impression image

3. Form

The admin will add the form to the database and it will be displayed on the service nav for the user.

Attributes:

title: title of the form

last_date: last date of form

total_post: posts for form applying

age: required age group for applying

qualification: minimum qualification for form

price: price of the form service

4. Orders

When user successfully apply & pay for the service from the service tab, the order then will come to Orders db.

Attributes:

form_id: reference form id

uid: reference user id

dateTime: date-time of order

payment_done: if payment is successful for order

5. Payment

This contains the details for payment of order.

Attributes:

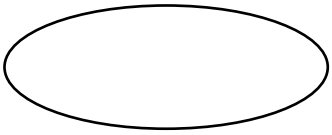
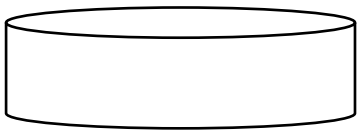
order_id:	reference order_id
amount:	amount of transaction
currency:	currency of transaction
razorpay_order_id:	transaction order id from razorpay
razorpay_payment_id:	transaction payment id from razorpay
razorpay_signature:	transaction signature from razorpay
created_at:	date-time information of payment

Chapter 10

Data Flow Diagram

Data flow diagrams (DFD) are part of a structed model in the development of software. They are a graphical technique that depicts information flow and the transforms that are applied as data move from input to output. Basically, the function of DFDs is to show the user a graphical analysis of a software system. It is like a flowchart, except DFDs show the flow of data throughout the system.

Data Flow Diagram Symbols

- : Data Flow
-  : Process
-  : Entity
-  : Data

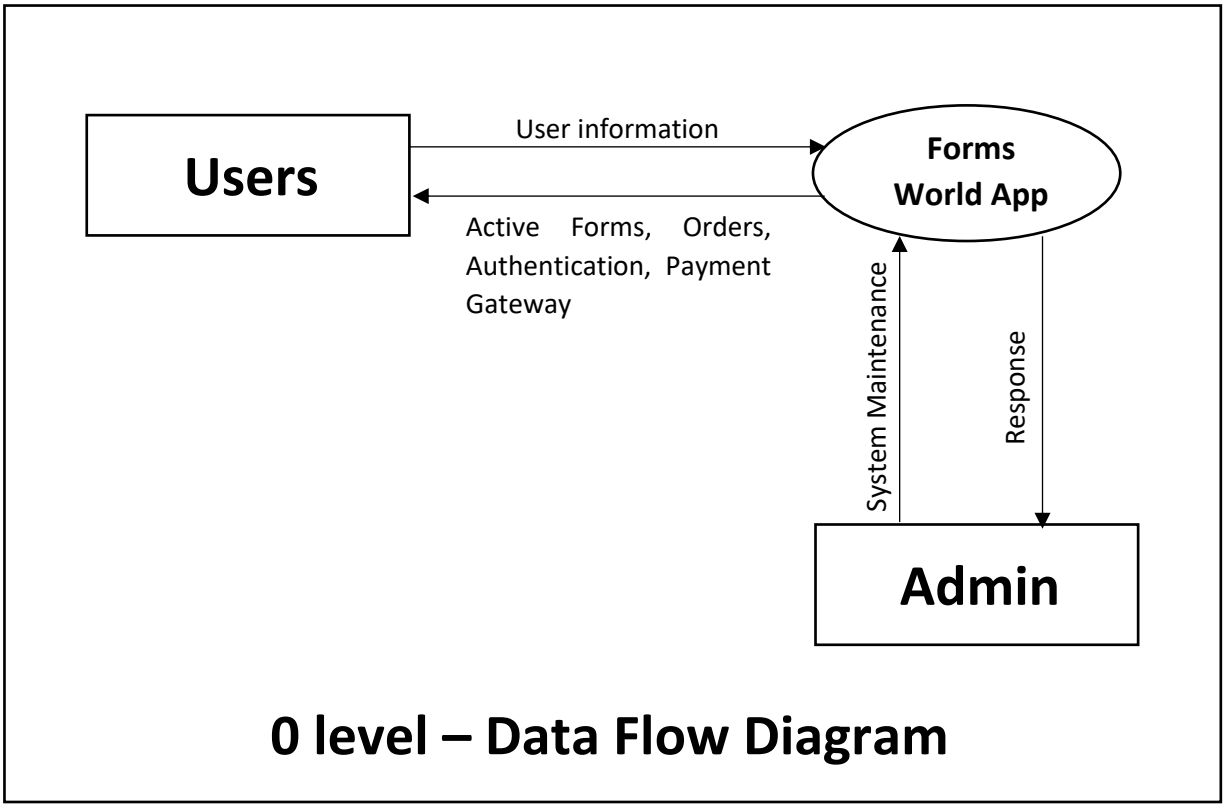


Fig. 9.1 0^{th} level – Data Flow Diagram

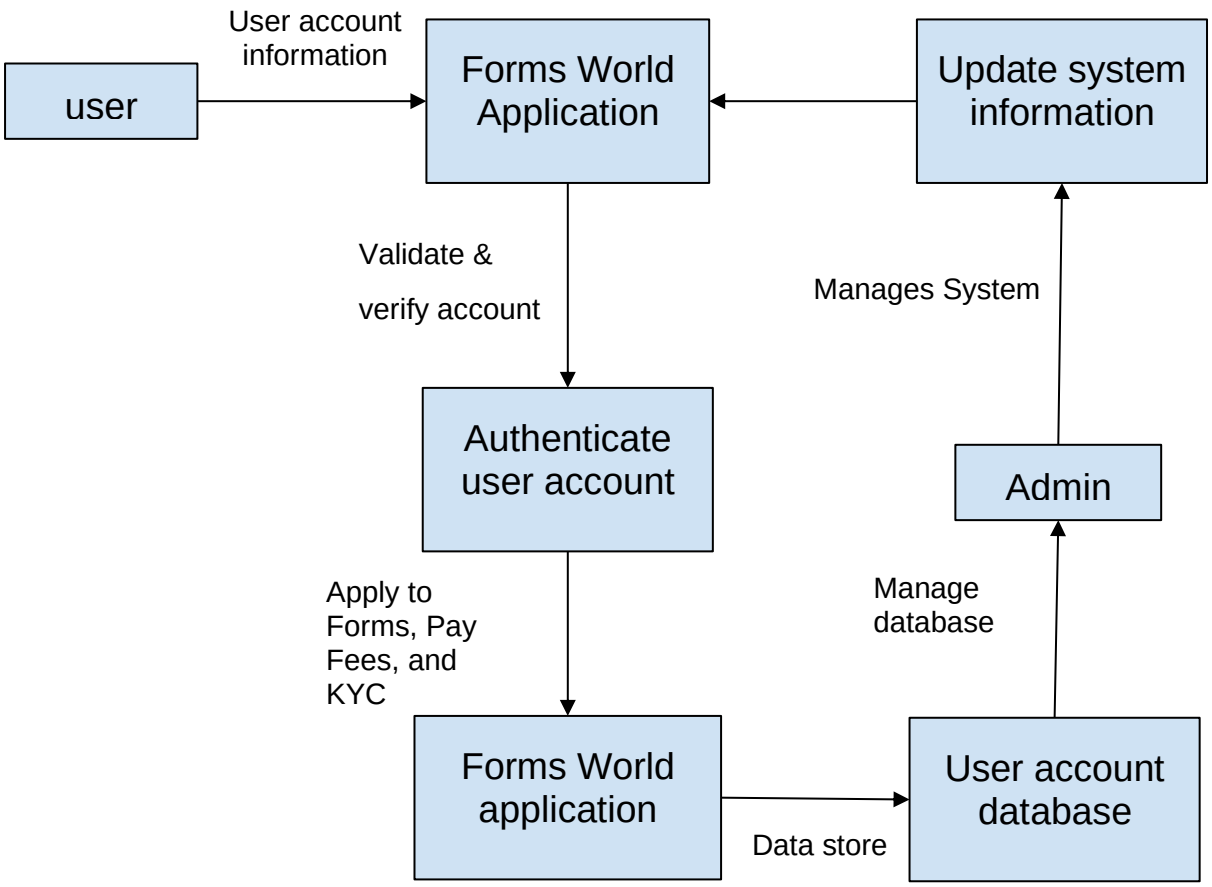
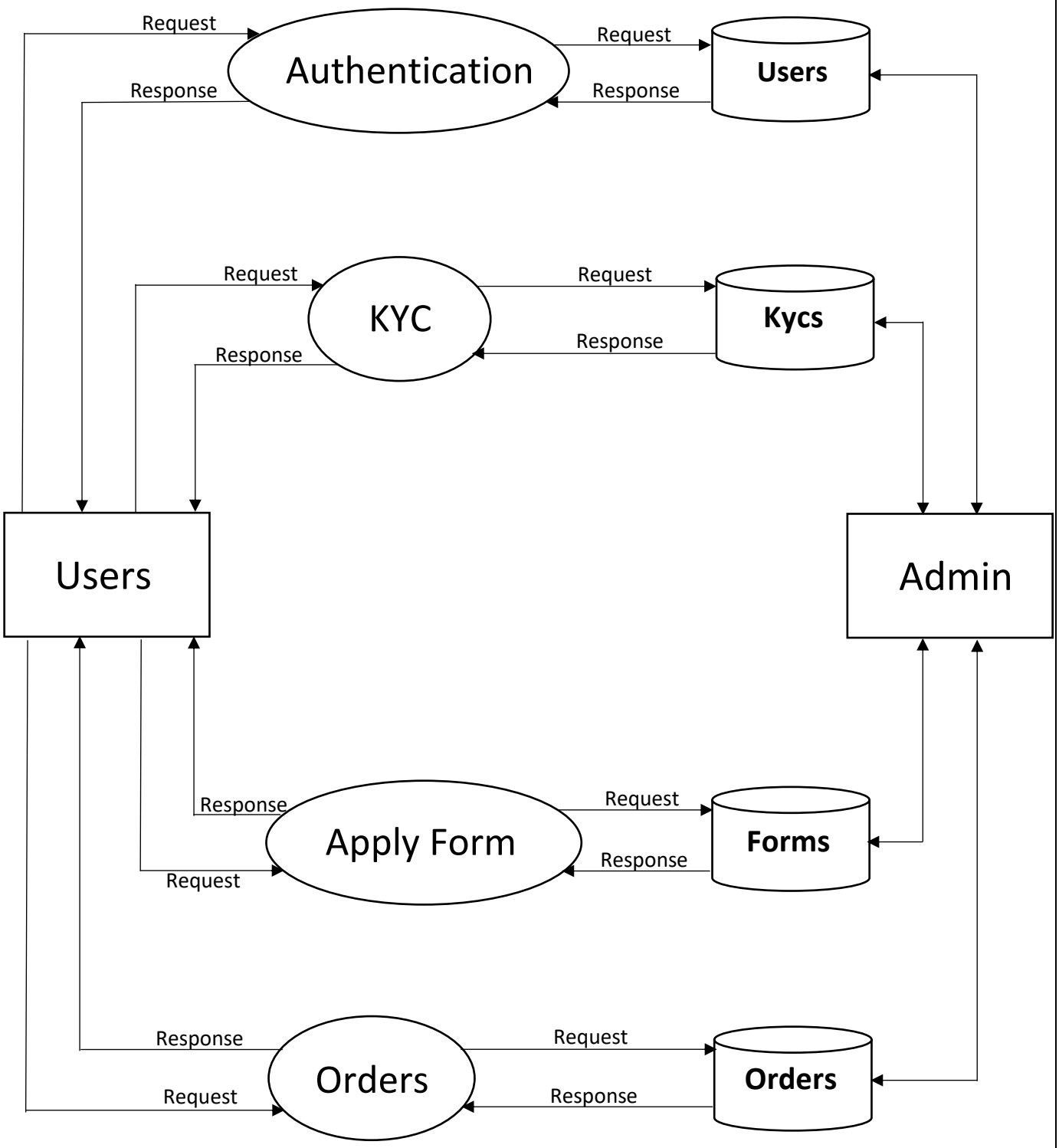


Fig. 9.2 1st level – Data Flow Diagram



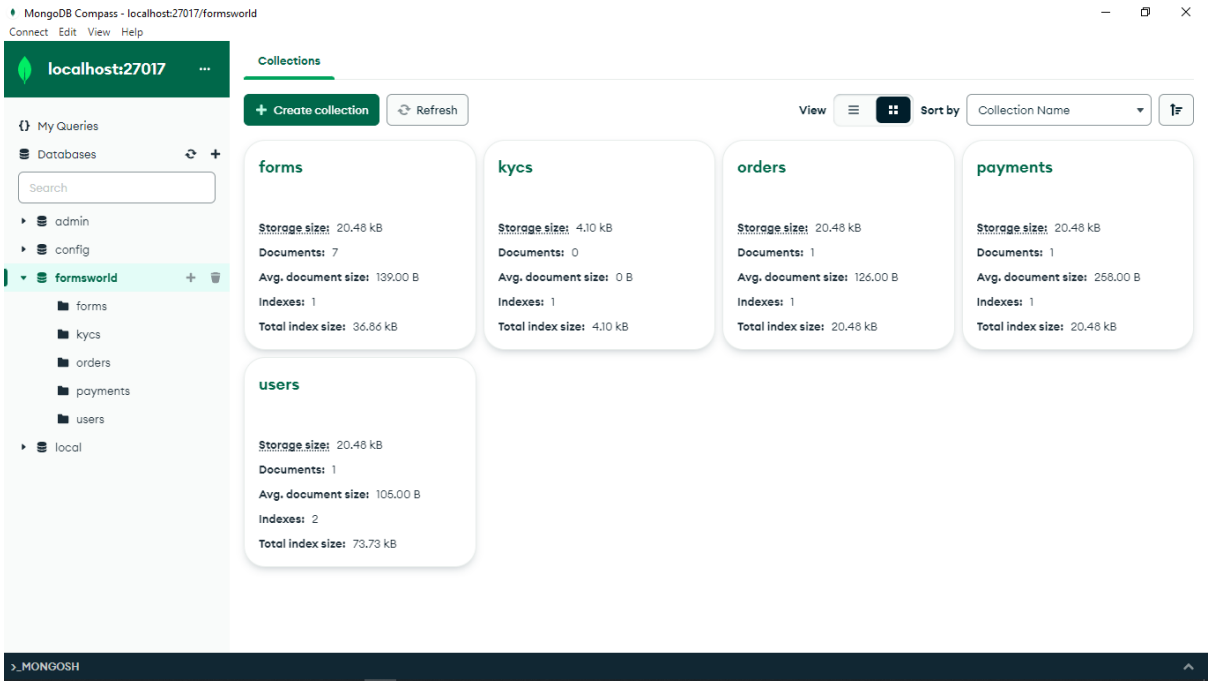
2nd level – Data Flow

Chapter 11

Project Design

11.1 Database Design:

1. Database design and collections



2. Forms stored in database


```
_id: ObjectId('644e3bbb2672e34ca3d71058')
title: "Aadhar Card"
last_date: 2023-04-30T16:00:00.000+00:00
total_post: null
age: "0-200"
qualification: "Everyone"
price: 150
__v: 0
```

```
_id: ObjectId('644e3bdd2672e34ca3d7105b')
title: "Driver Licence"
last_date: 2023-04-30T16:00:00.000+00:00
total_post: null
age: "0-200"
qualification: "Everyone"
price: 2000
__v: 0
```

3. Data stored in database kyc


```
_id: ObjectId('644e2893dd5c35895a090f30')
__v: 0
aadhar_no: 1234567890123456
caste_certificate: "/9j/4AAQSkZJRgABAQAAAQABAAD/4gHYSUNDX1BST0ZJTEUAAQEAAAHIAAAAAAQwAABtbn…"
domicile_image: "/9j/4AAQSkZJRgABAQAAAQABAAD/4gHYSUNDX1BST0ZJTEUAAQEAAAHIAAAAAAQwAABtbn…"
family_id: "abc123def"
fathers_name: "Dilbag Singh Dahiya"
full_name: "Shubham Dahiya"
left_thumb: "/9j/4AAQSkZJRgABAQAAAQABAAD/4gHYSUNDX1BST0ZJTEUAAQEAAAHIAAAAAAQwAABtbn…"
marksheet_10th: "/9j/4AAQSkZJRgABAQAAAQABAAD/4gHYSUNDX1BST0ZJTEUAAQEAAAHIAAAAAAQwAABtbn…"
marksheet_12th: "/9j/4AAQSkZJRgABAQAAAQABAAD/4gHYSUNDX1BST0ZJTEUAAQEAAAHIAAAAAAQwAABtbn…"
mothers_name: "Susheela Dahiya"
passport_image: "/9j/4AAQSkZJRgABAQAAAQABAAD/4gHYSUNDX1BST0ZJTEUAAQEAAAHIAAAAAAQwAABtbn…"
right_thumb: "/9j/4AAQSkZJRgABAQAAAQABAAD/4gHYSUNDX1BST0ZJTEUAAQEAAAHIAAAAAAQwAABtbn…"
signature_image: "/9j/4AAQSkZJRgABAQAAAQABAAD/4gHYSUNDX1BST0ZJTEUAAQEAAAHIAAAAAAQwAABtbn…"
```


4. Orders stored in database


```
_id: ObjectId('644e3ddb2672e34ca3d71070')
form_id: ObjectId('644e3bbb2672e34ca3d71058')
uid: ObjectId('644e2893dd5c35895a090f30')
payment_done: true
payment_id: ObjectId('644e3ddb2672e34ca3d7106e')
dateTime: 2023-04-30T10:07:23.282+00:00
__v: 0
```

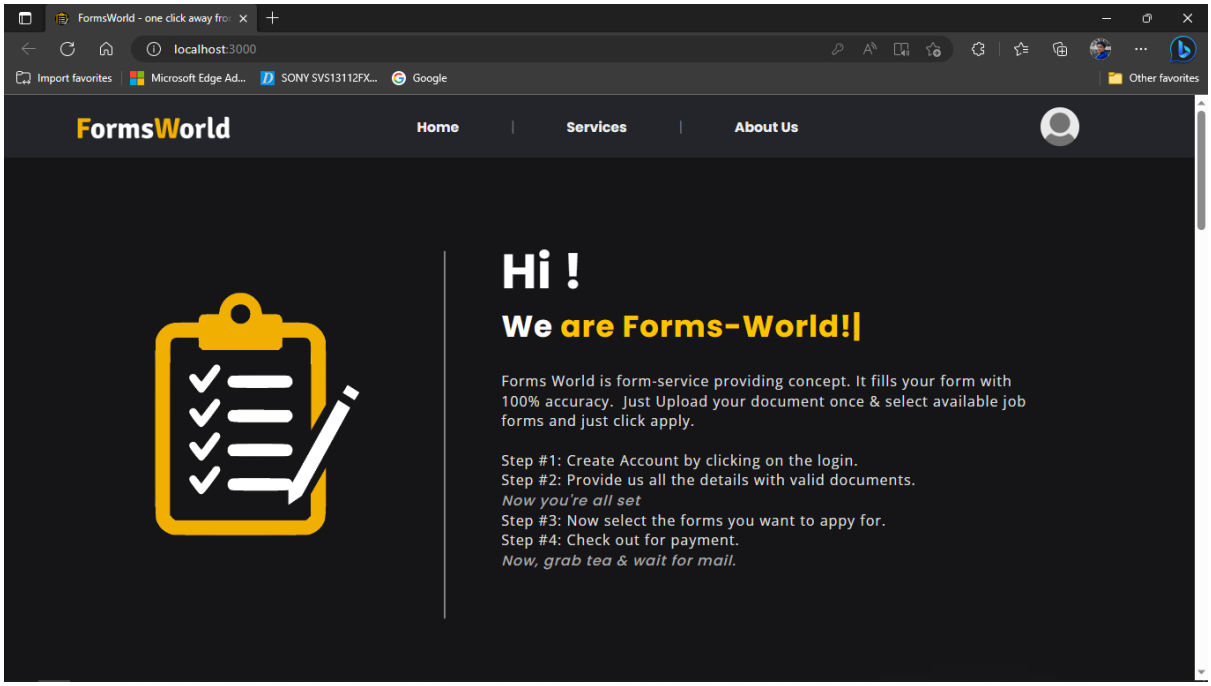
```
_id: ObjectId('644e454d2b21186acc13115b')
form_id: ObjectId('644e3c812672e34ca3d71066')
uid: ObjectId('644e2893dd5c35895a090f30')
payment_done: true
payment_id: ObjectId('644e454d2b21186acc131159')
dateTime: 2023-04-30T10:39:09.792+00:00
__v: 0
```

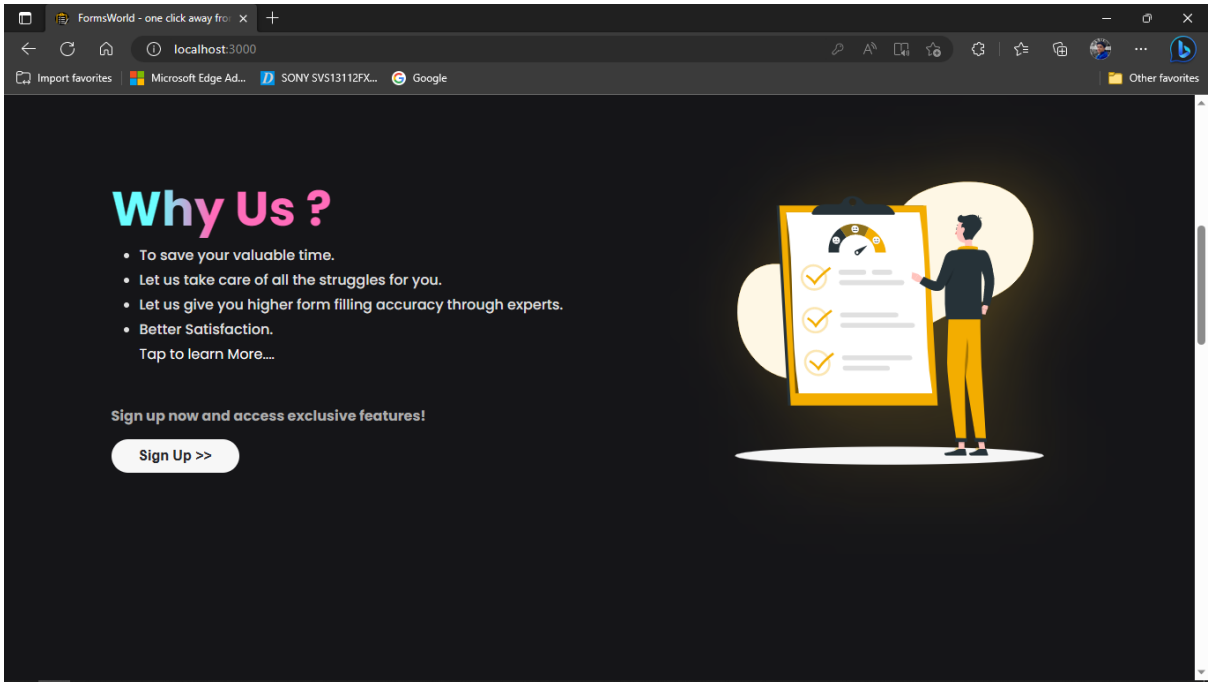
5. Payments stored in database

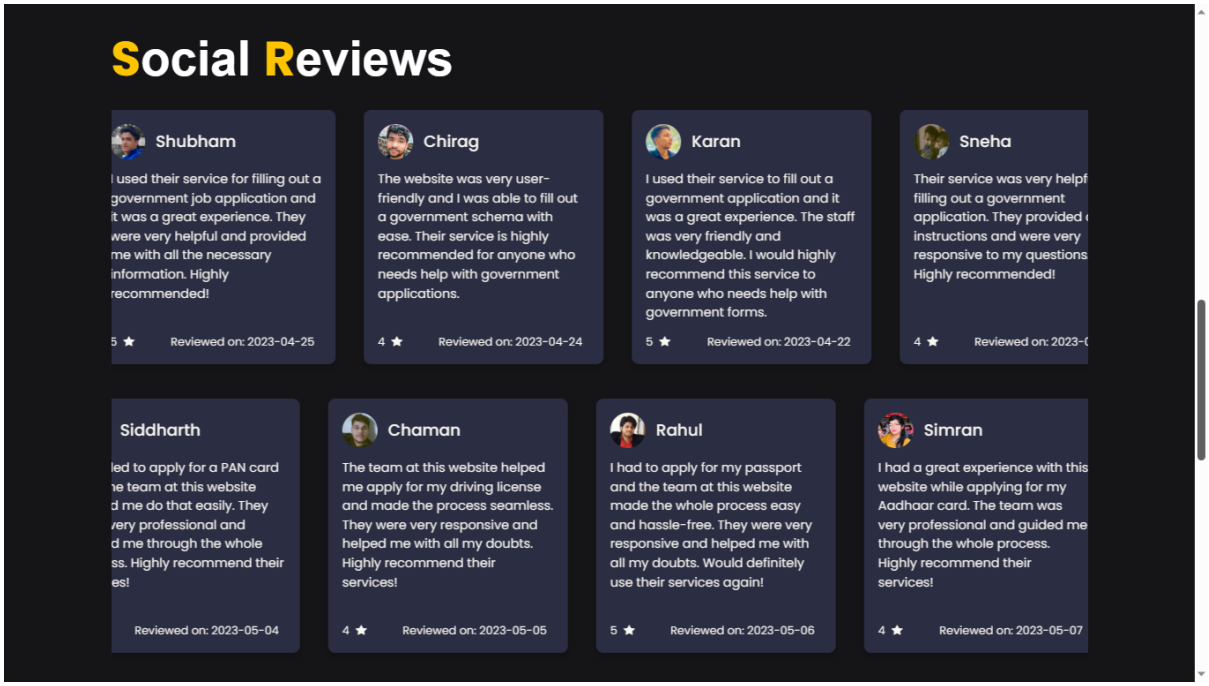
6. Users likes stored in database

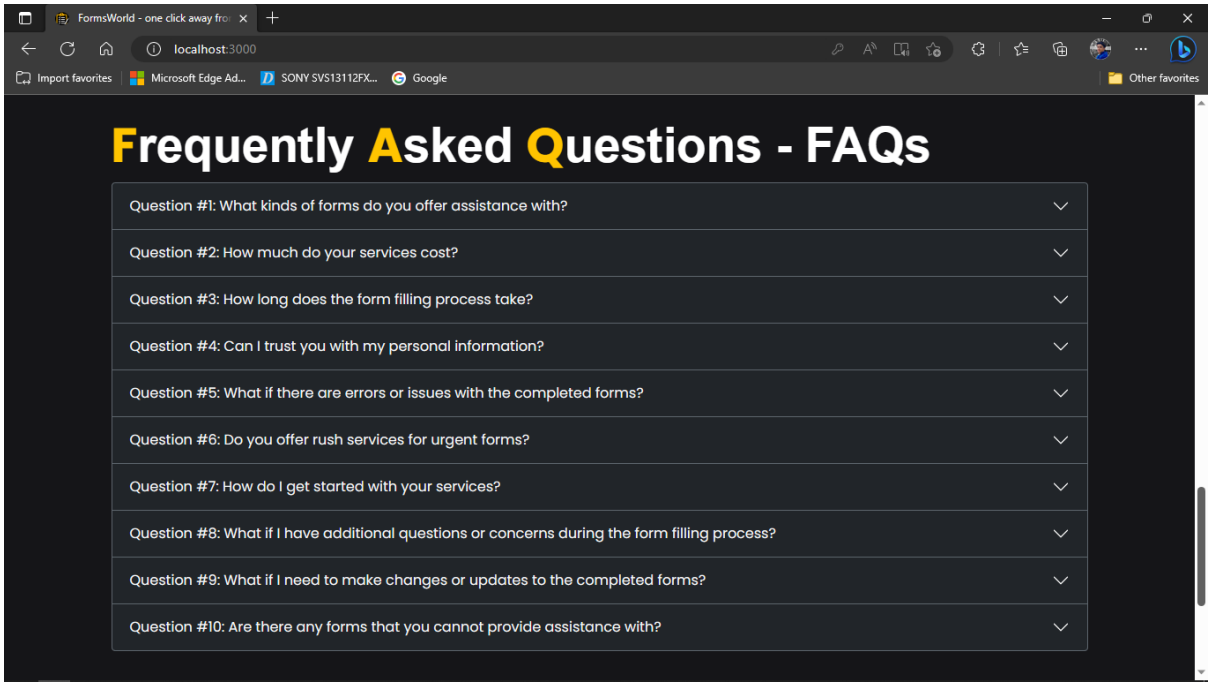
11.2 Screen Designs:

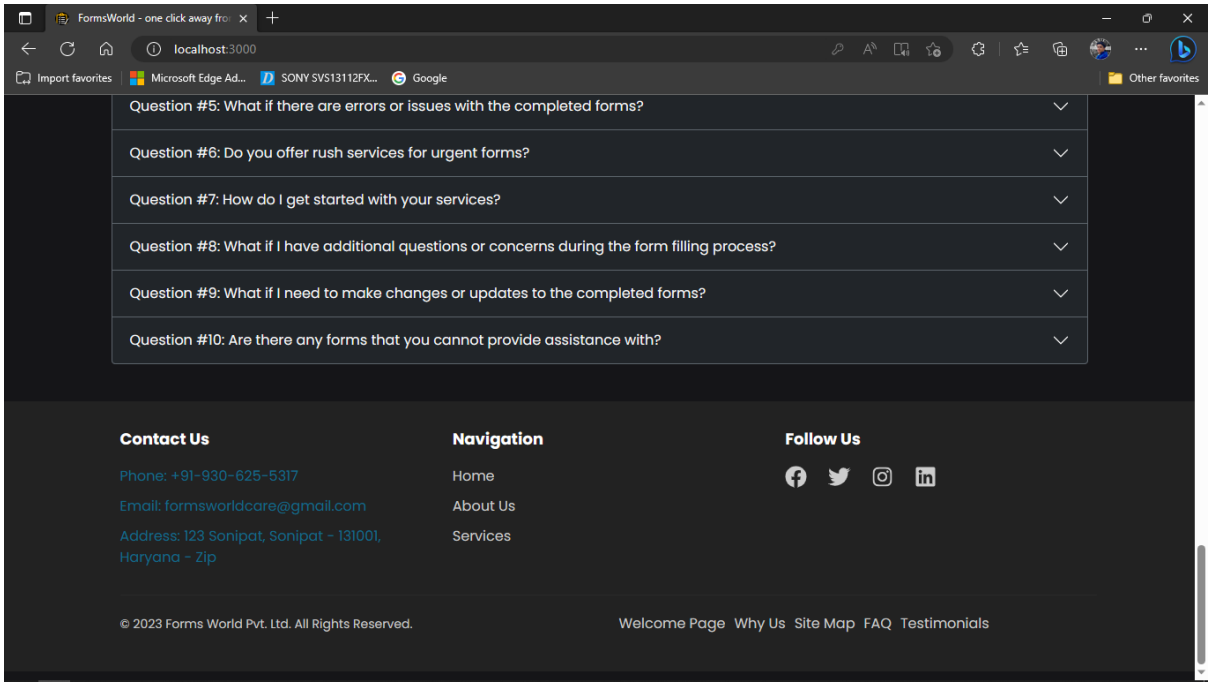
1. Landing page of Forms World



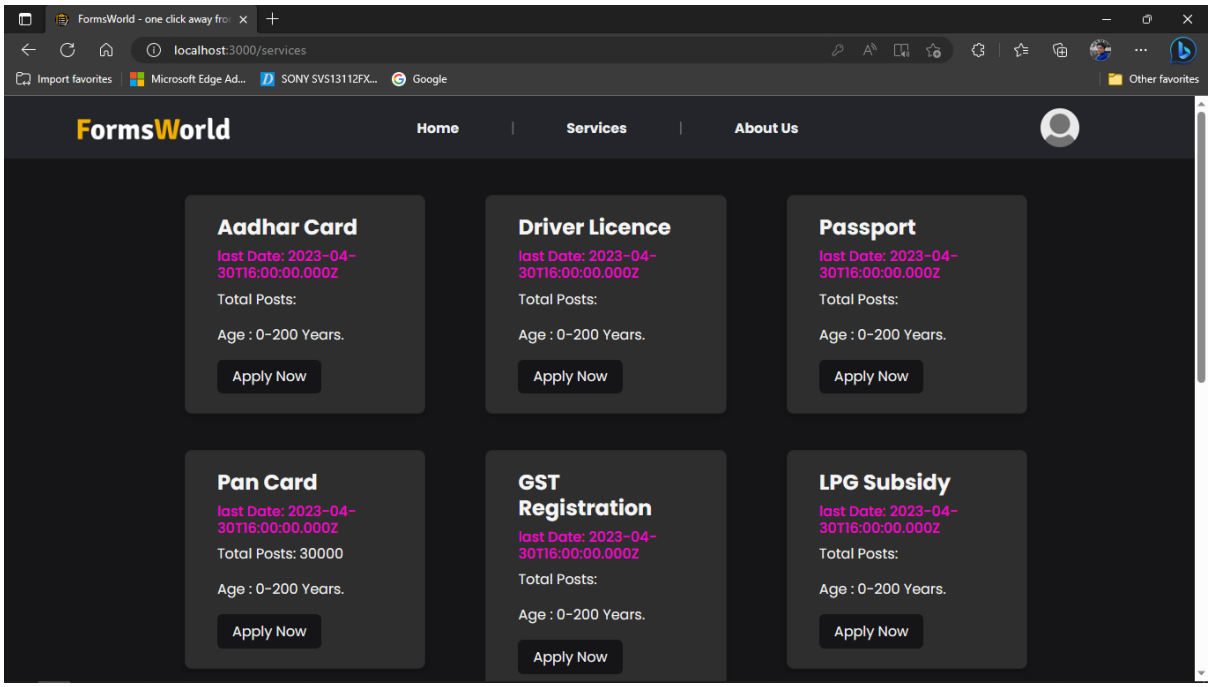


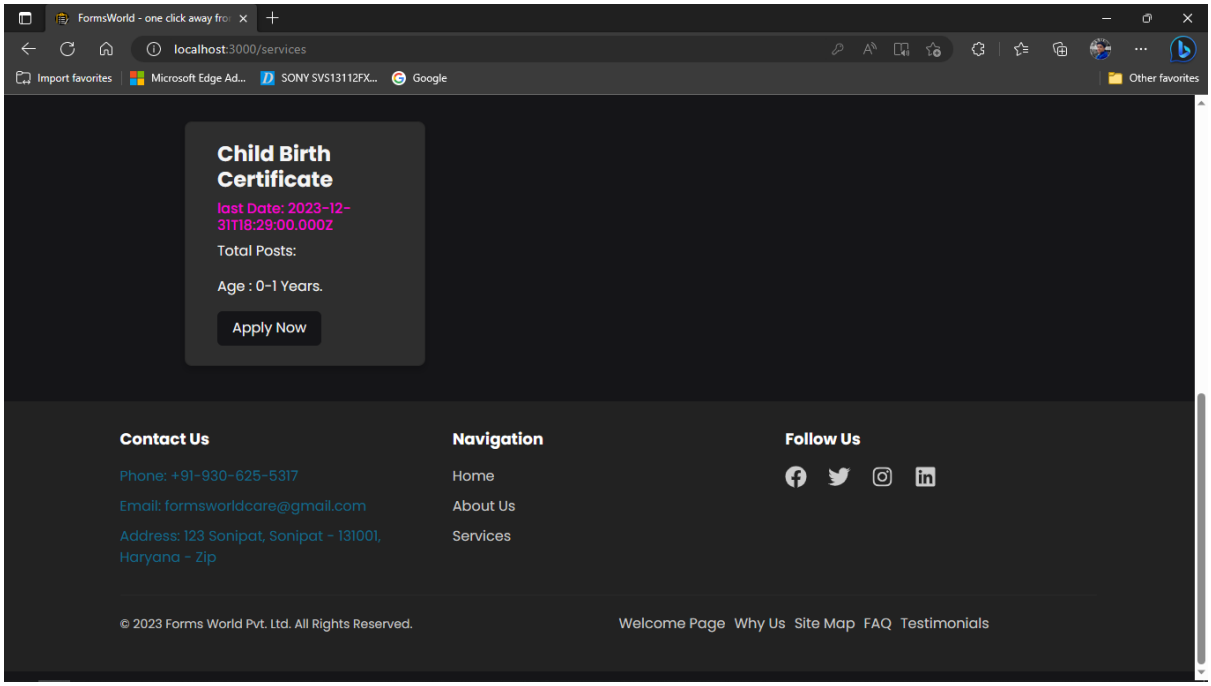

Social Reviews
Shubham
I used their service for filling out a government job application and it was a great experience. They were very helpful and provided me with all the necessary information. Highly recommended!
5 ★
Reviewed on: 2023-04-25
Chirag
The website was very user-friendly and I was able to fill out a government schema with ease. Their service is highly recommended for anyone who needs help with government applications.
4 ★
Reviewed on: 2023-04-24
Karan
I used their service to fill out a government application and it was a great experience. The staff was very friendly and knowledgeable. I would highly recommend this service to anyone who needs help with government forms.
5 ★
Reviewed on: 2023-04-22
Sneha
Their service was very helpf filling out a government application. They provided instructions and were very responsive to my questions Highly recommended!
4 ★
Reviewed on: 2023-
Siddharth
led to apply for a PAN card he team at this website d me do that easily. They very professional and d me through the whole ss. Highly recommend their es!
Reviewed on: 2023-05-04
Chaman
The team at this website helped me apply for my driving license and made the process seamless. They were very responsive and helped me with all my doubts. Highly recommend their services!
4 ★
Reviewed on: 2023-05-05
Rahul
I had to apply for my passport and the team at this website made the whole process easy and hassle-free. They were very responsive and helped me with all my doubts. Would definitely use their services again!
5 ★
Reviewed on: 2023-05-06
Simran
I had a great experience with this website while applying for my Aadhaar card. The team was very professional and guided me through the whole process. Highly recommend their services!
4 ★
Reviewed on: 2023-05-07


FormsWorld - one click away fro
localhost:3000
Import favorites
Microsoft Edge Ad...
SONY SVS13112FX...
Google
Other favorites
Frequently Asked Questions - FAQs
Question #1: What kinds of forms do you offer assistance with?
Question #2: How much do your services cost?
Question #3: How long does the form filling process take?
Question #4: Can I trust you with my personal information?
Question #5: What if there are errors or issues with the completed forms?
Question #6: Do you offer rush services for urgent forms?
Question #7: How do I get started with your services?
Question #8: What if I have additional questions or concerns during the form filling process?
Question #9: What if I need to make changes or updates to the completed forms?
Question #10: Are there any forms that you cannot provide assistance with?

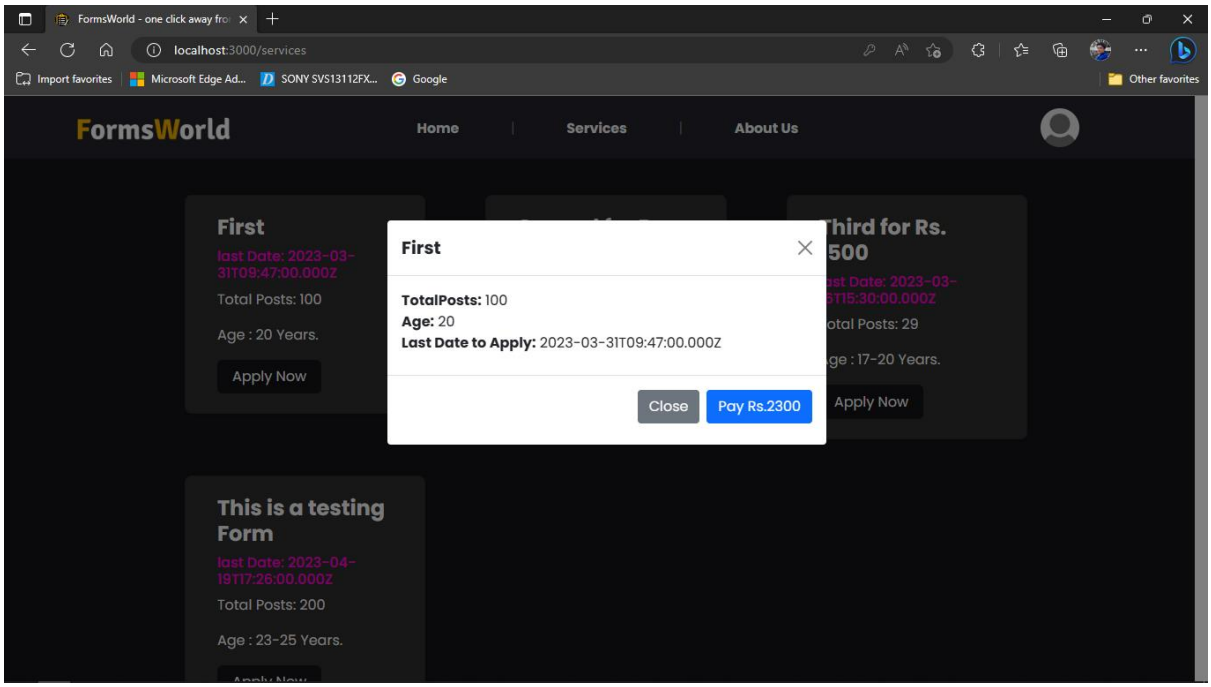


2. Services Page of Forms World

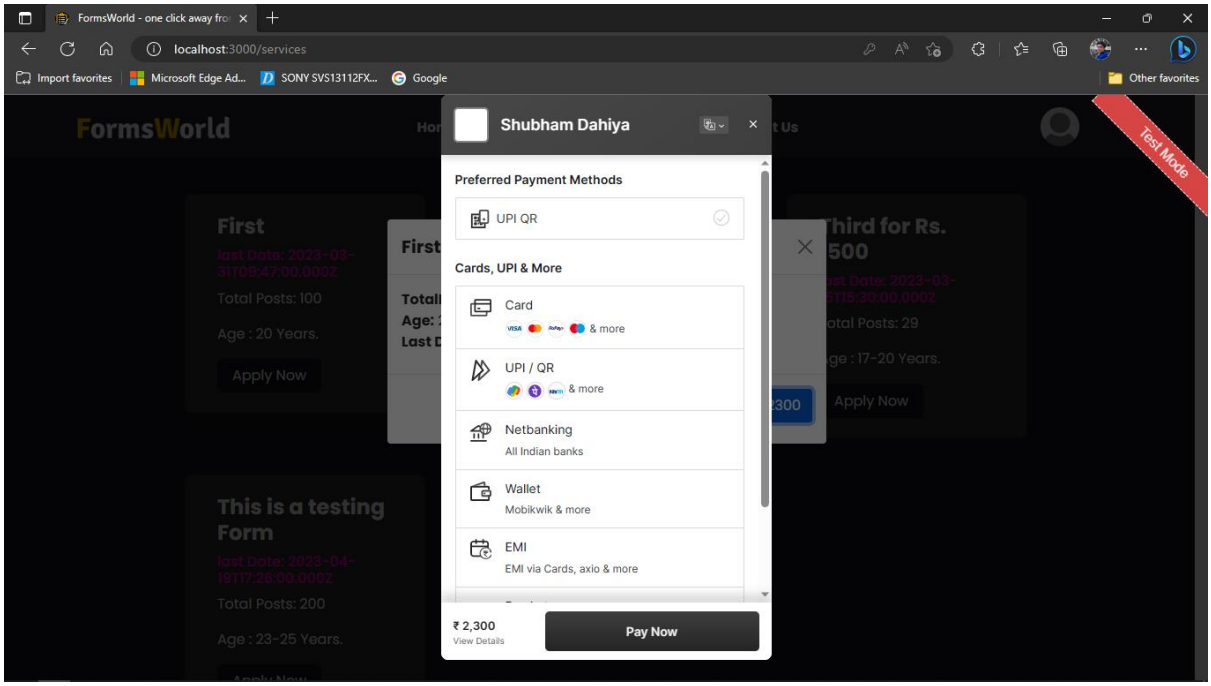




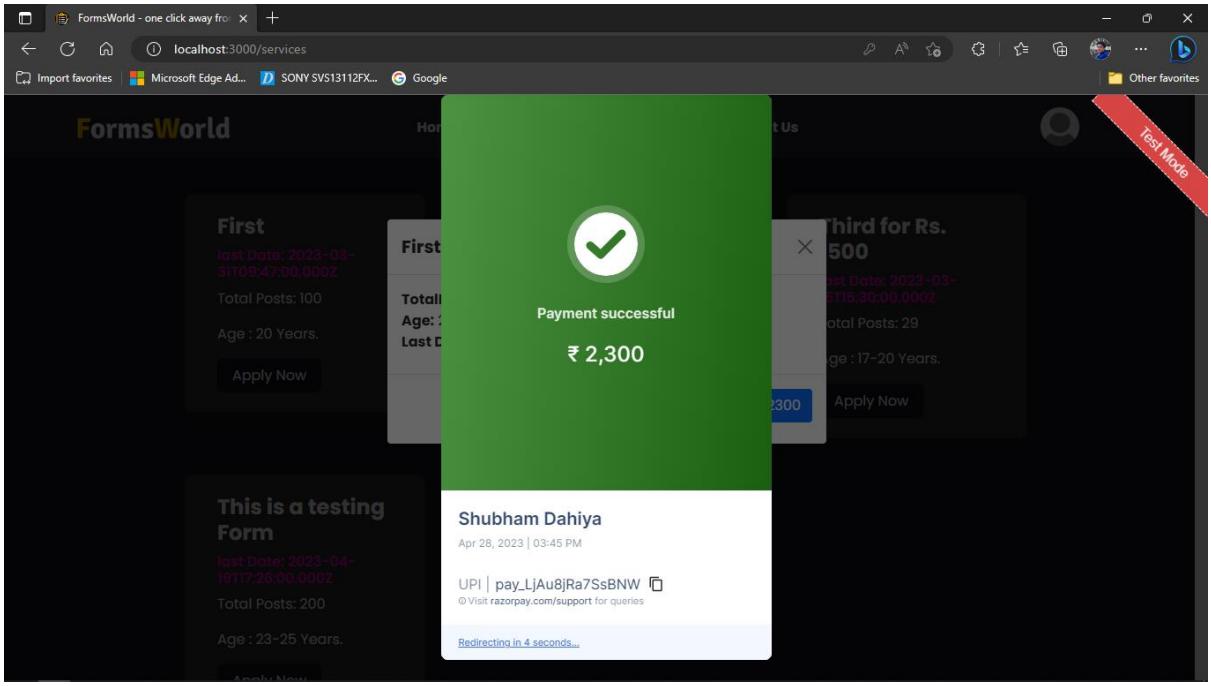
3. Form Details Modal



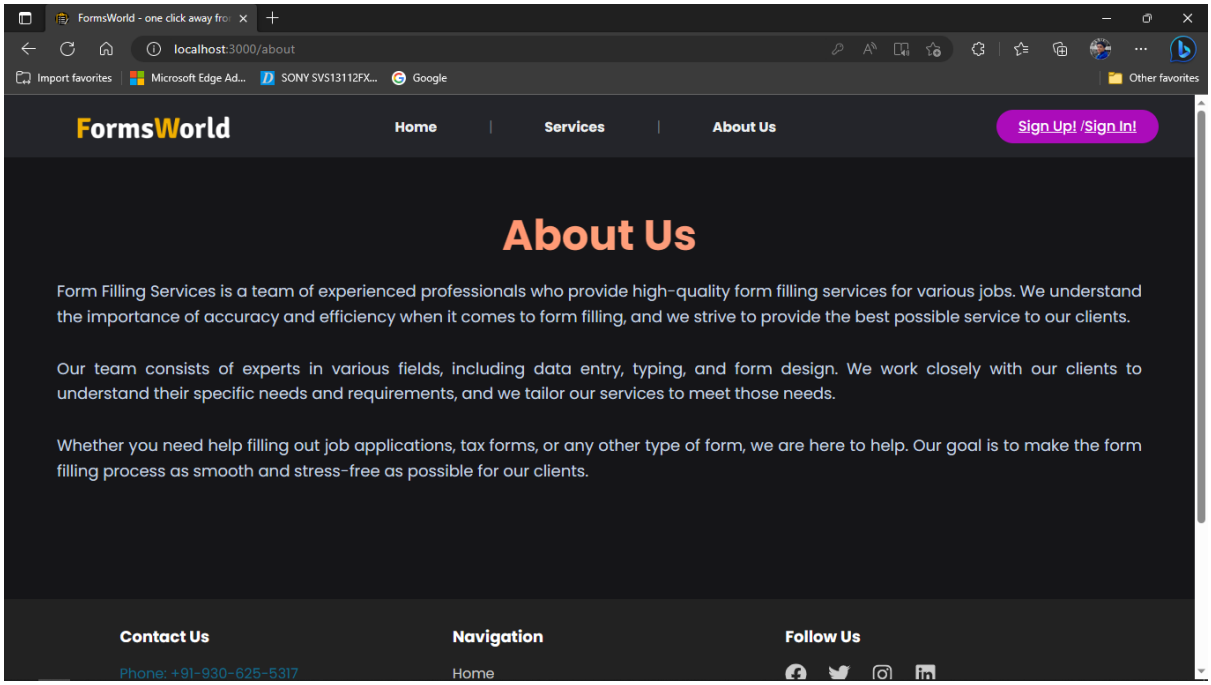
4. Payment Gateway



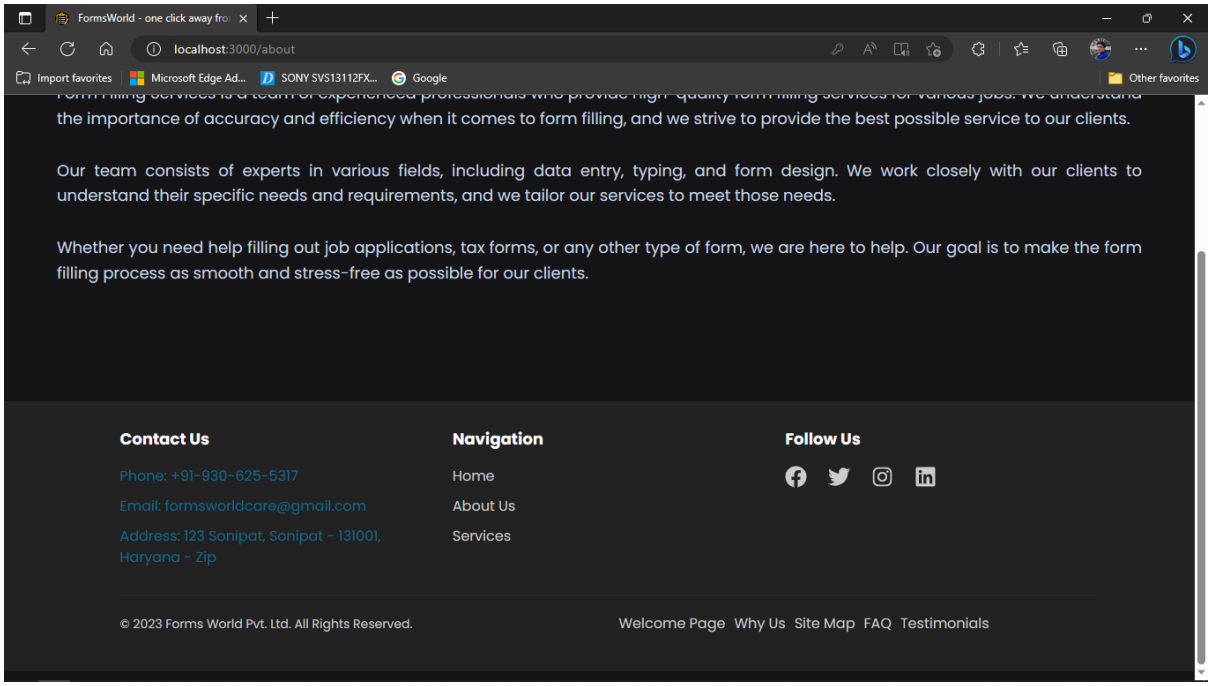
5. Payment



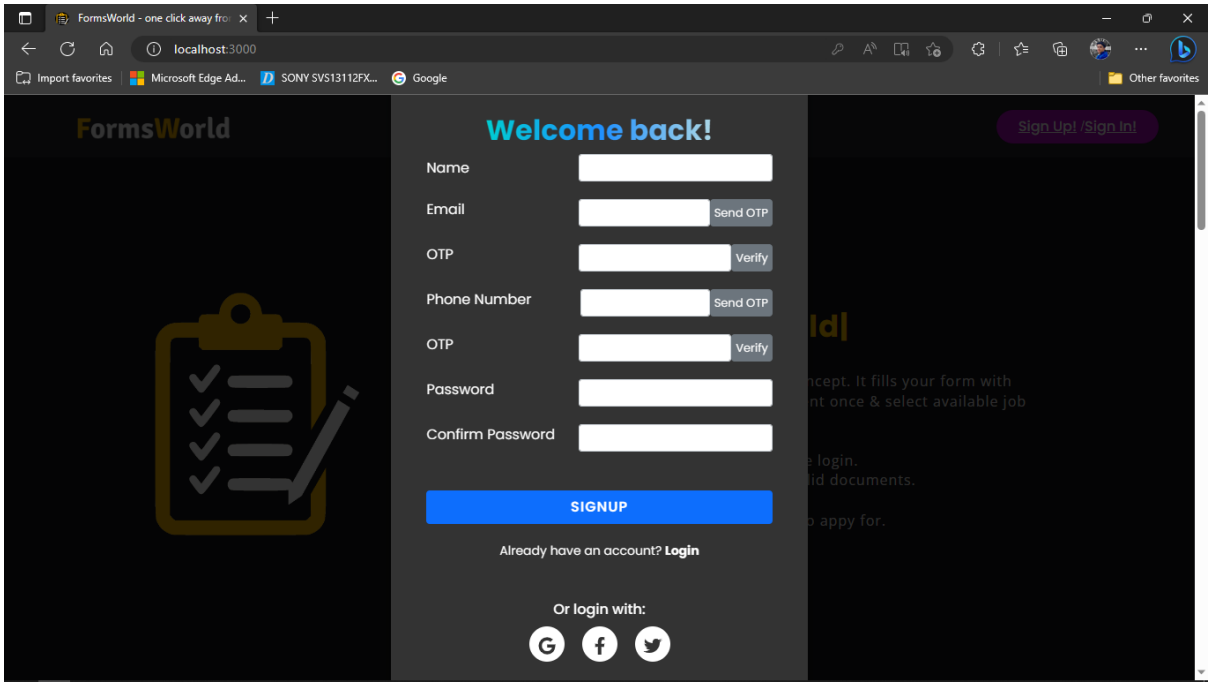
6. About us Page of Forms World



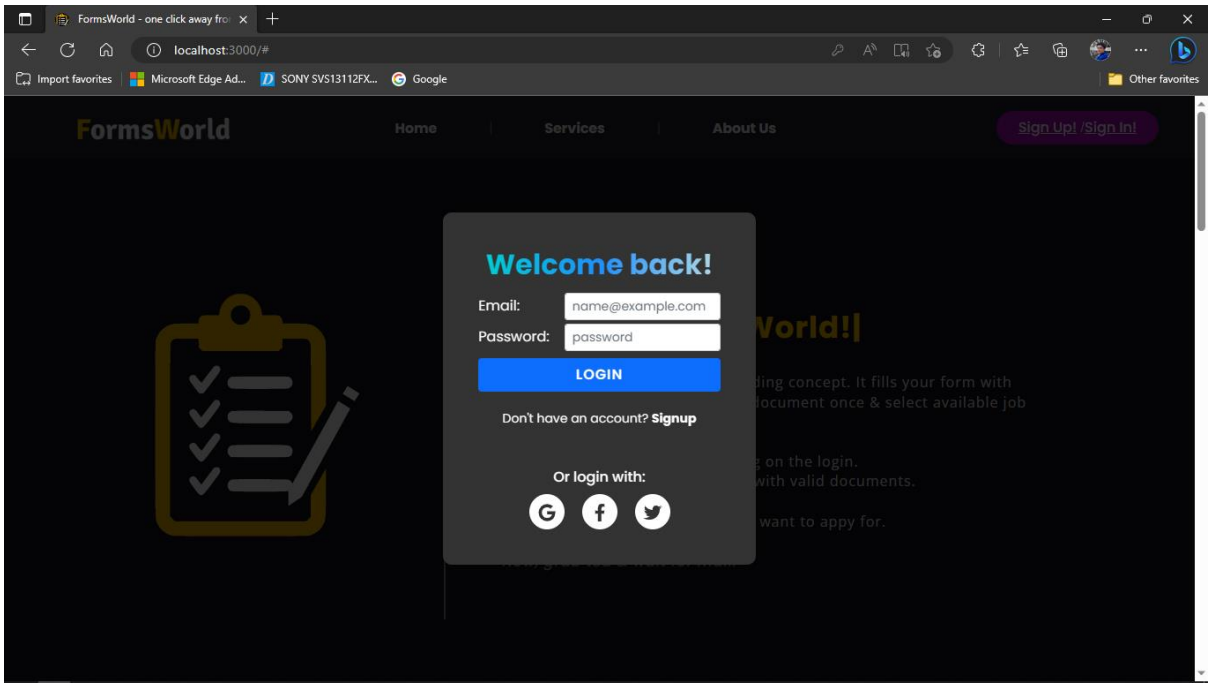
Footer



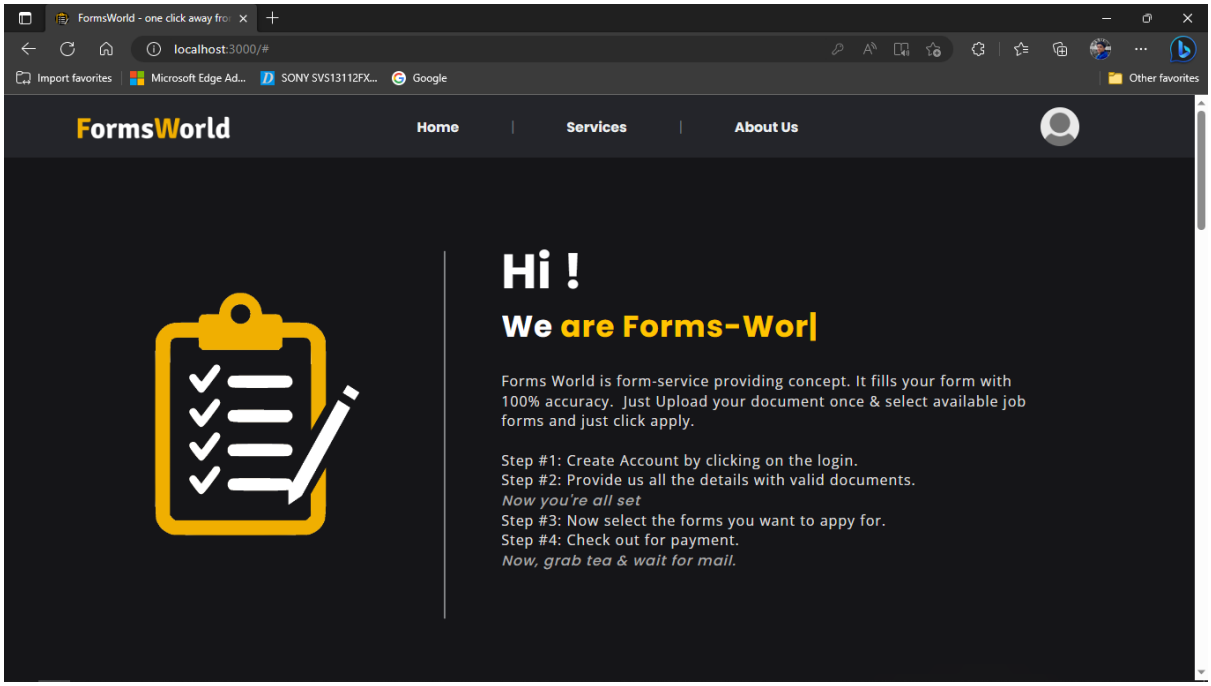
7. Sign up popup on Forms World



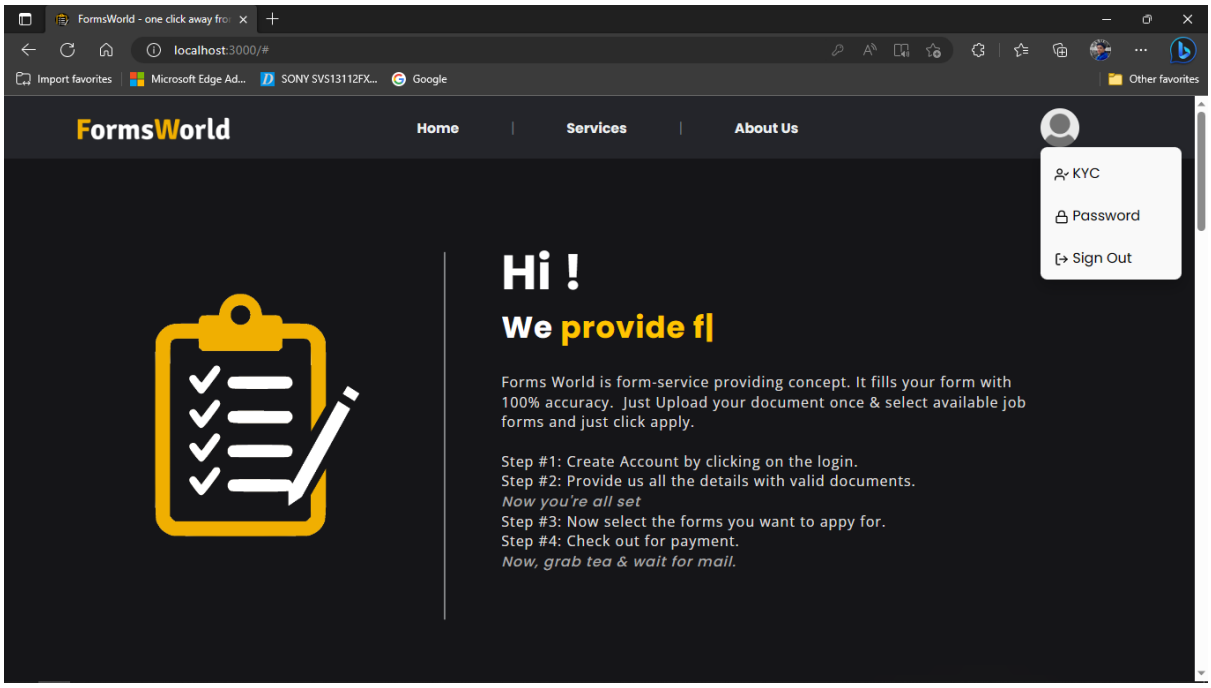
8. Sign in popup on Forms World:



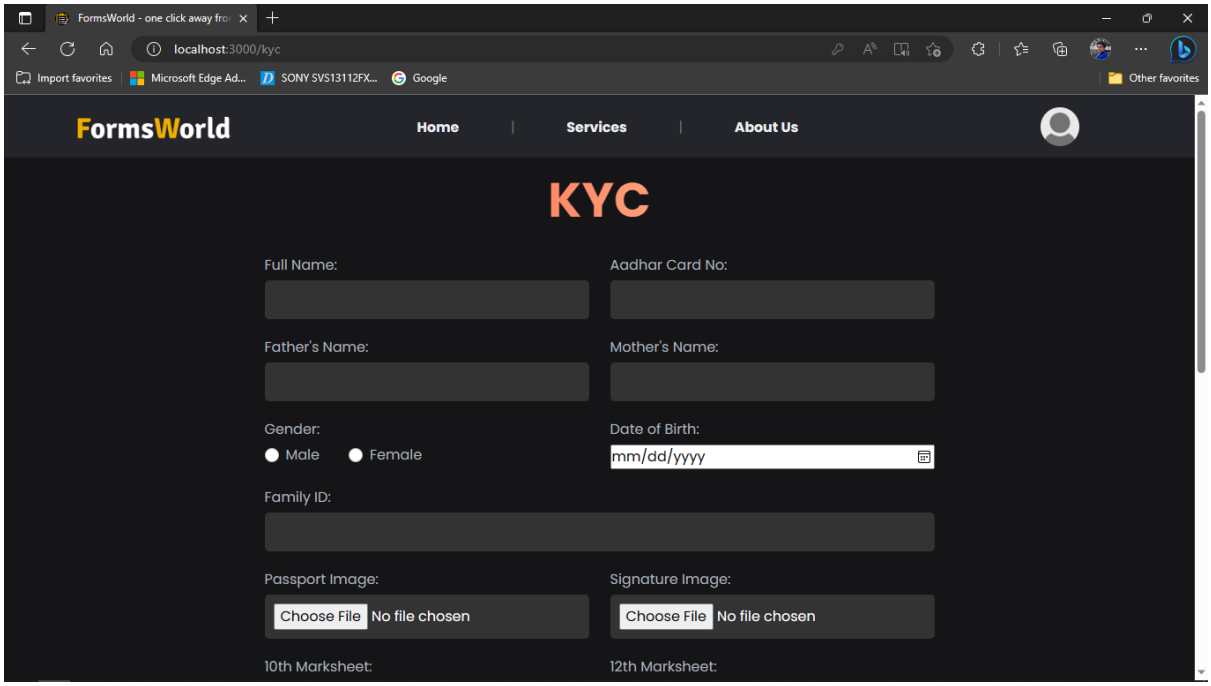
9. Signed in user page



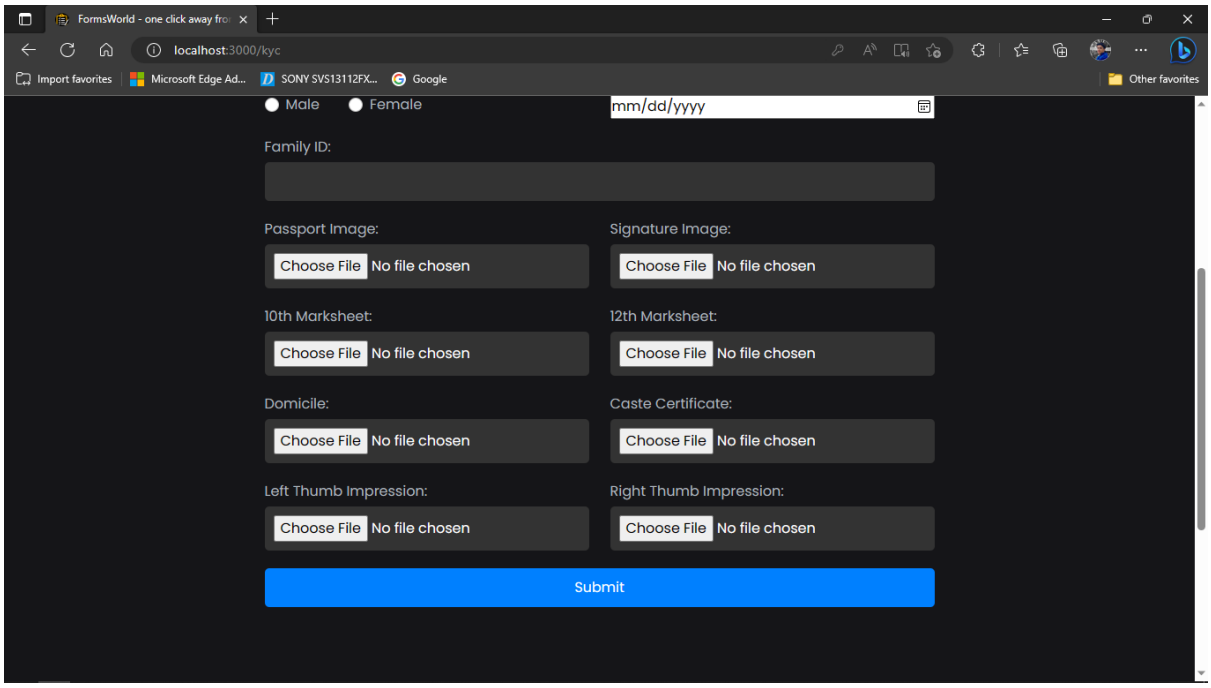
10. Signed user options



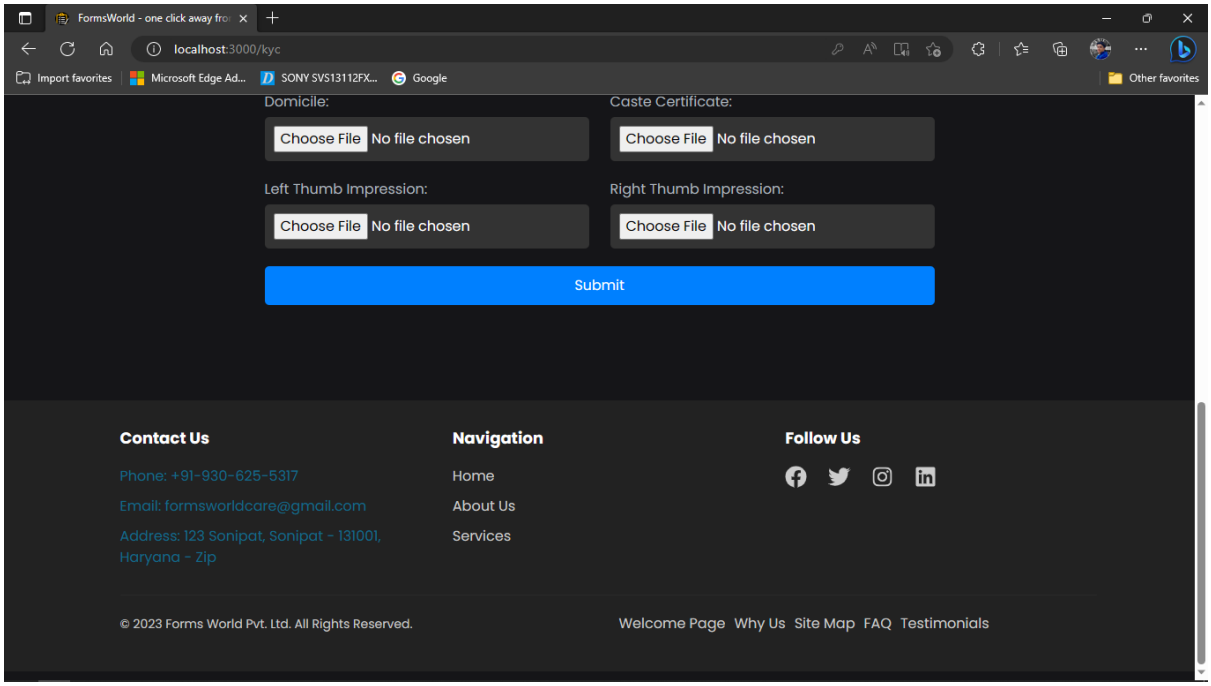
11. KYC page for user on Forms World



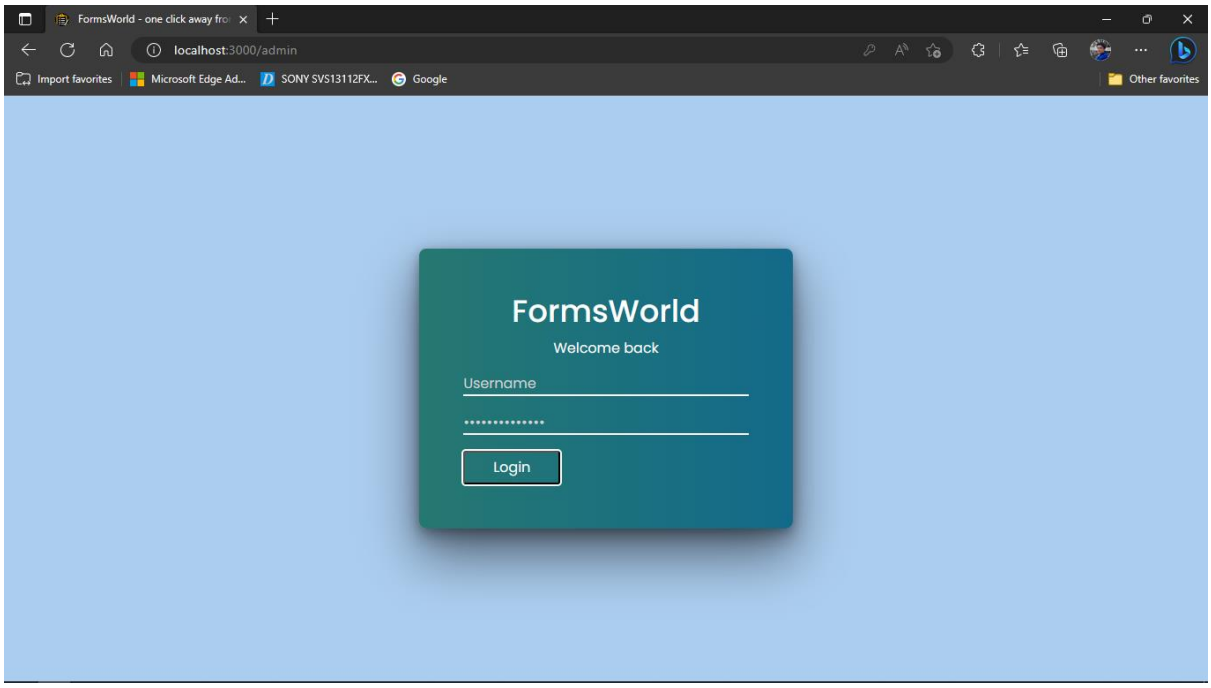
scrolled



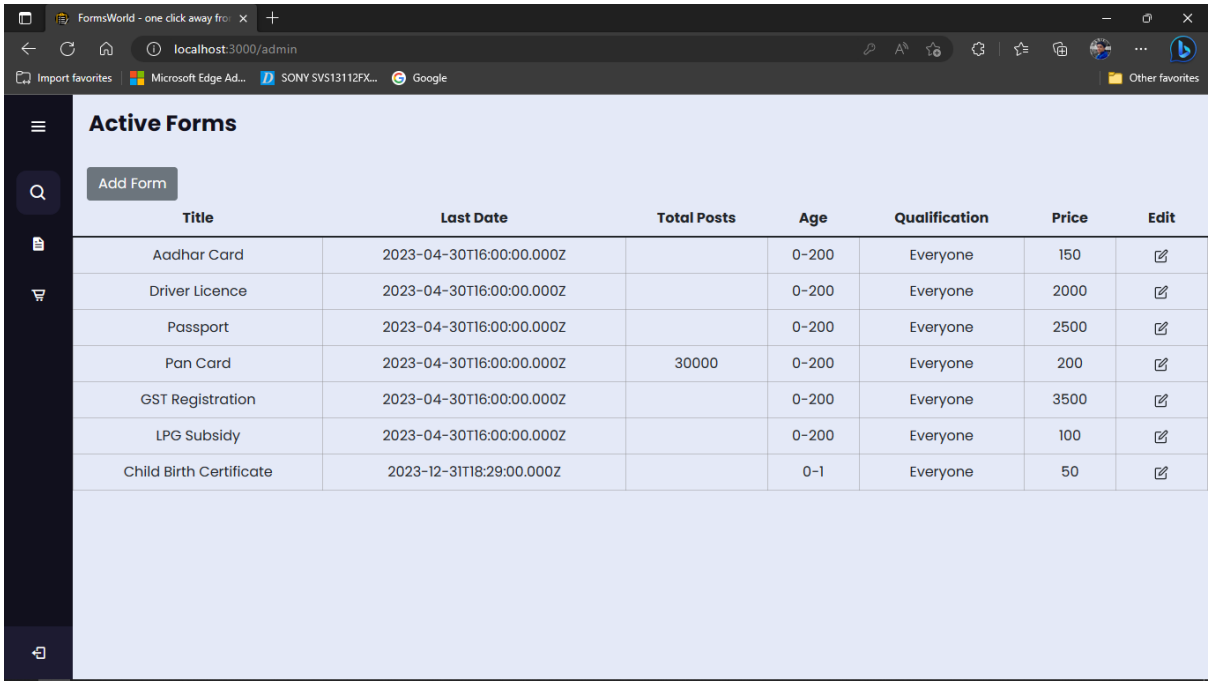
Scrolled



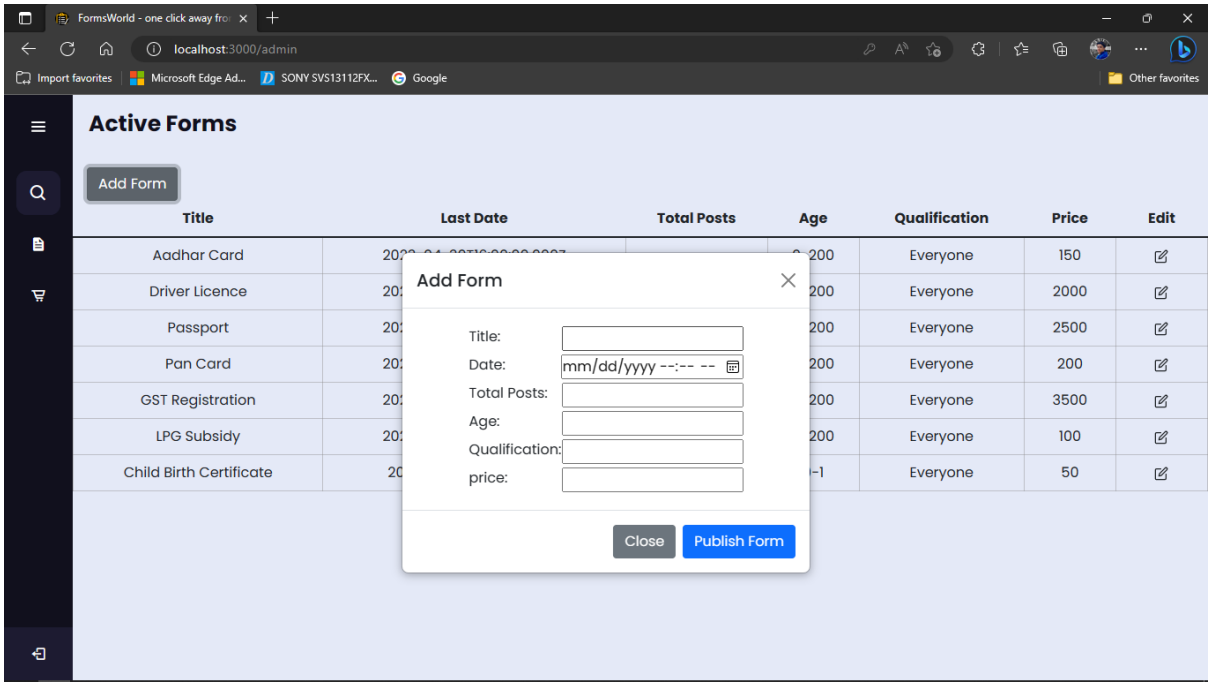
12. Admin page on Forms World



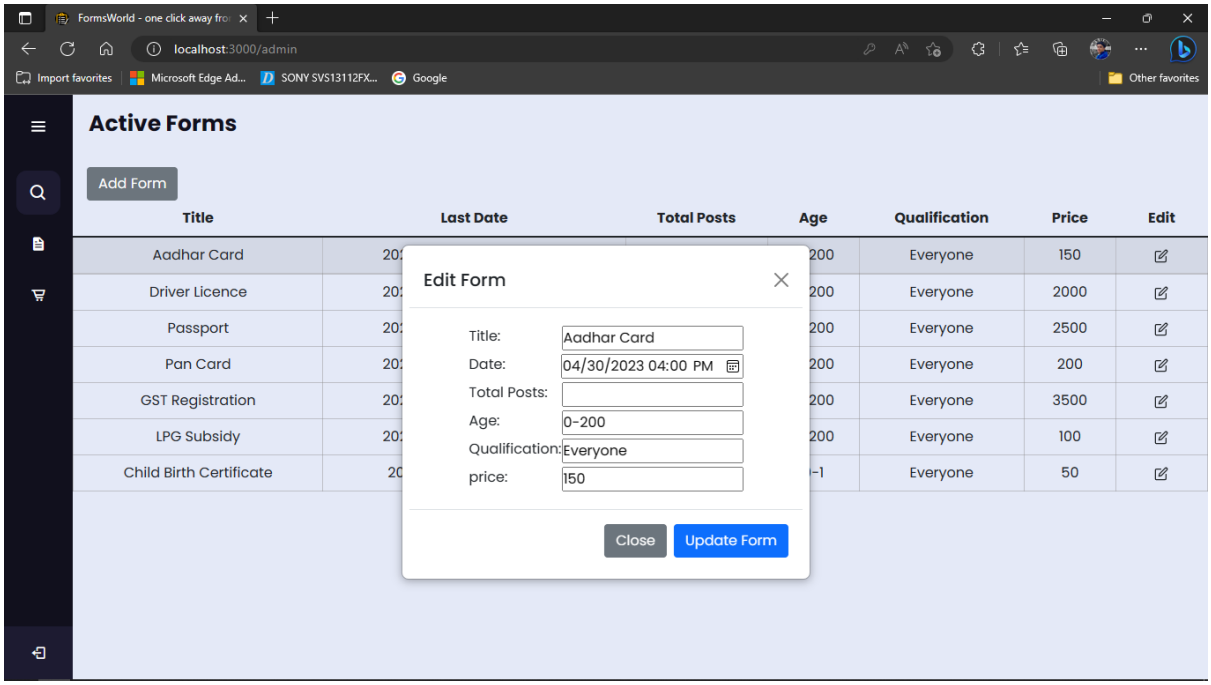
13. Admin logged in showing Active Forms



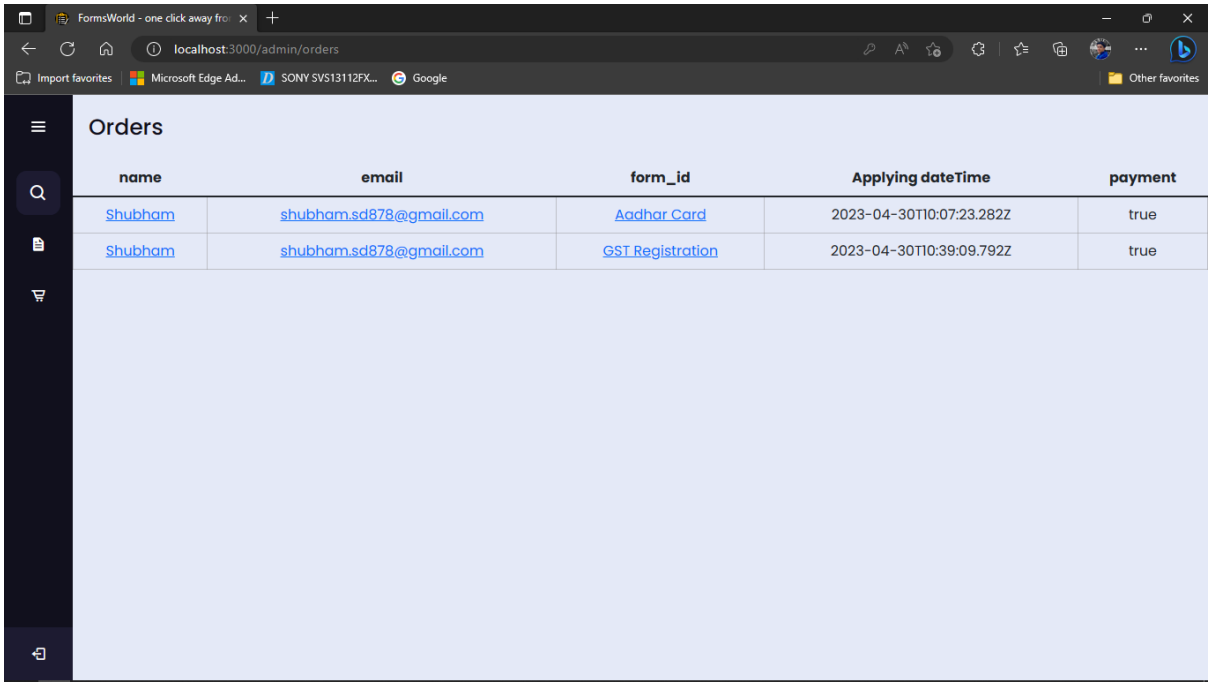
14. Add Form popup Modal



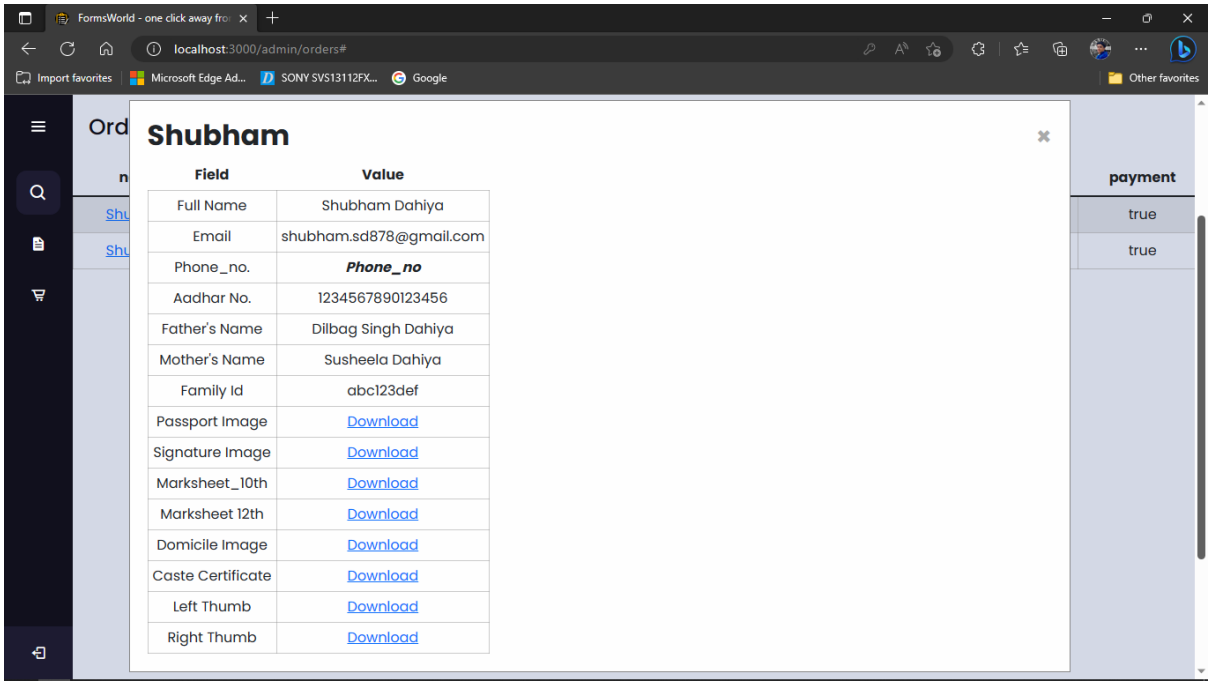
15.Edit Form popup modal



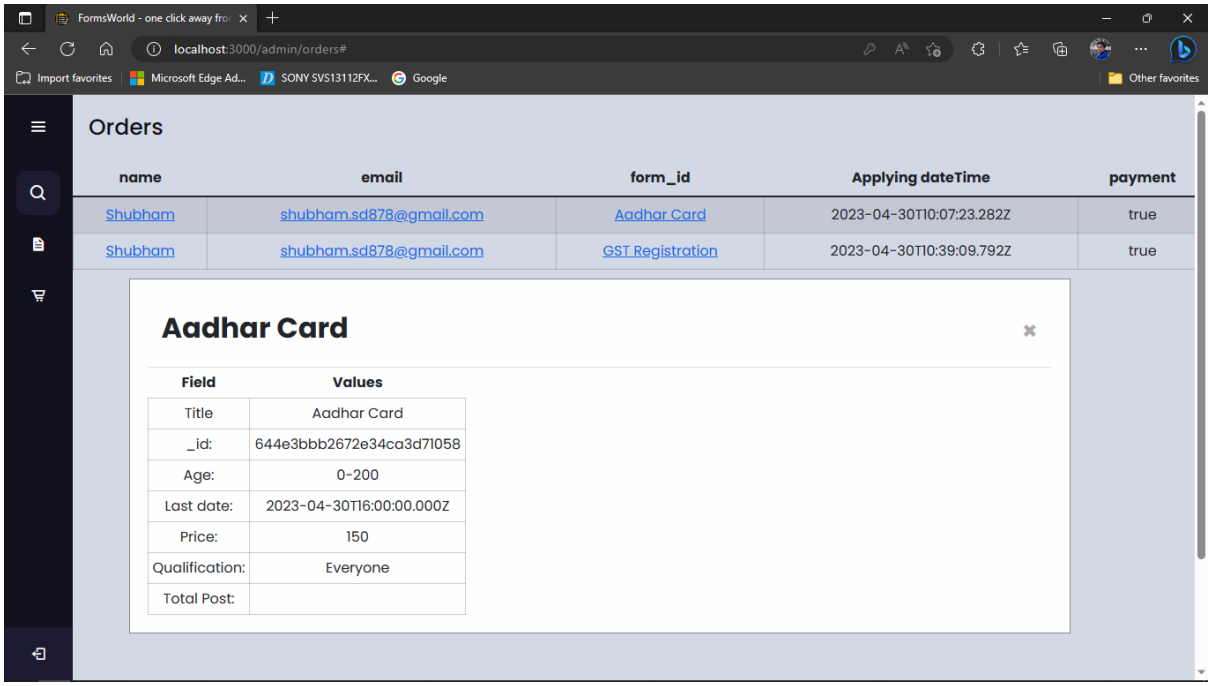
16. Orders Nav on Admin Screen



17. User Details modal on Orders nav on Admin page



18. Form Details on Orders nav on Admin page



Chapter 12

Testing

Testing is an integral part of any software development process, and it ensures that the software being developed is of high quality, reliable, and free from any errors or bugs.

In the case of the MERN stack-based website that provides form filling services for various jobs, testing is particularly crucial to ensure that the website works as expected and meets the requirements of the end-users.

The testing process for this project can be broken down into several stages, including unit testing, integration testing, and user acceptance testing (UAT).

Testing Process:

1) Unit testing
2) Integration testing
3) User Acceptance testing

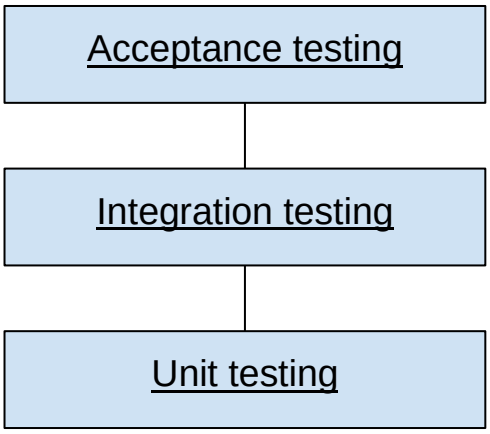


Fig. 12.1 Types of Testing

Unit testing involves testing individual units or components of the website in isolation to ensure that they function correctly. The unit tests can cover various aspects of the website, including the user interface, database operations, and server-side logic.

Integration testing involves testing the website's interaction with other external systems or services, such as APIs, payment gateways, and email services. This can be done using tools such as Postman, which allows developers to test the integration of various APIs with the website.

User acceptance testing (UAT) is the final stage of testing, where end-users are invited to test the website to ensure that it meets their needs and requirements. This can be done by creating test scenarios that simulate real-world scenarios and testing the website's response to these scenarios. The feedback obtained from UAT can be used to improve the website and ensure that it meets the users' expectations.

In addition to the above testing stages, it is also essential to perform security testing to ensure that the website is free from any vulnerabilities or security threats.

Overall, testing process is very important for the success of the MERN stack-based website that provides form filling services for various jobs. By ensuring that the website is thoroughly tested and free from errors or bugs, we can guarantee that the end-users have a positive experience when using the website, which may ultimately end in higher satisfaction with customers and business achievement.

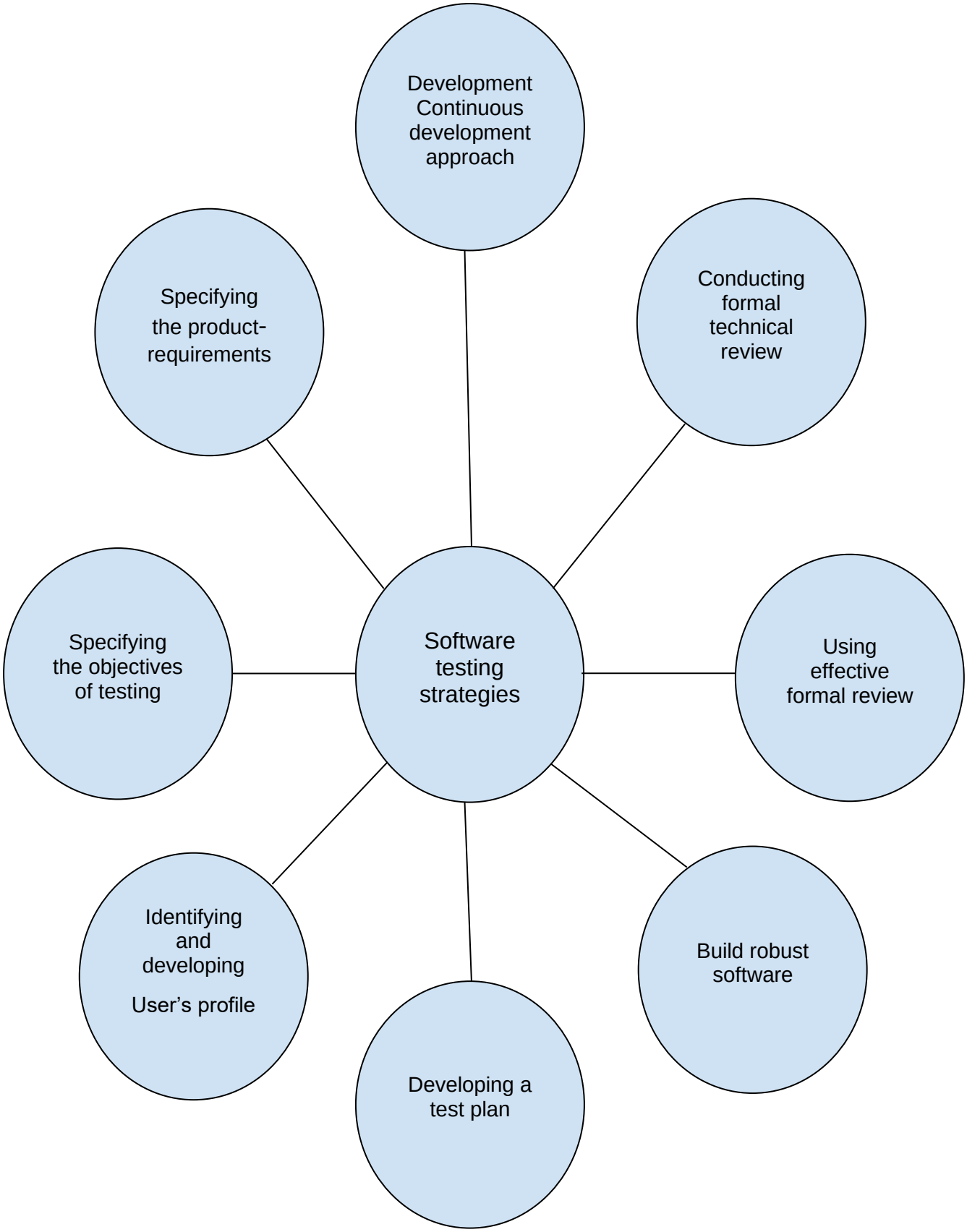


Fig. 12.2 Software Testing Strategies

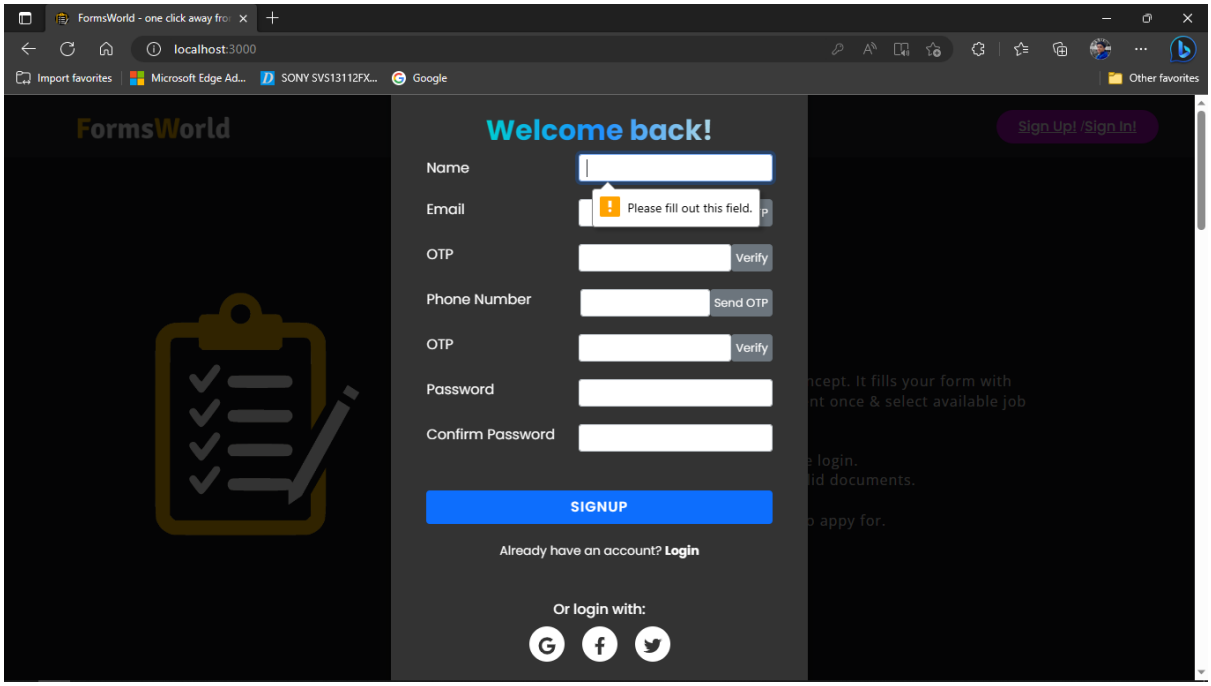
1. **Signup and login testing**

TEST	CASE	EXPECTED	RESULT
Signup	Insufficient details	Error fill the details	PASS
Signup	Wrong email entered	@ is required	PASS
Signup	Send otp to mail.	User receive otp to mail	PASS
Signup	Try to submit without otp	Show alert.	PASS
Signup	Verify mail otp	Verify the otp else show alert.	PASS
Signup	Send otp to number	User receive otp on number	PASS
Signup	Verify number otp	Verify the otp else show alert.	PASS
Signup	Try to submit without number otp	Show alert	PASS
Signup	All details correctly filled	Popup for Successful Registration	PASS
Sign In	Enter incorrect details	Show invalid details	PASS
Sign In	Enter correct details	Login	PASS
Sign In	After login redirect to home page	Redirection to home page	PASS

Table 12.1 Signup and Login Testing

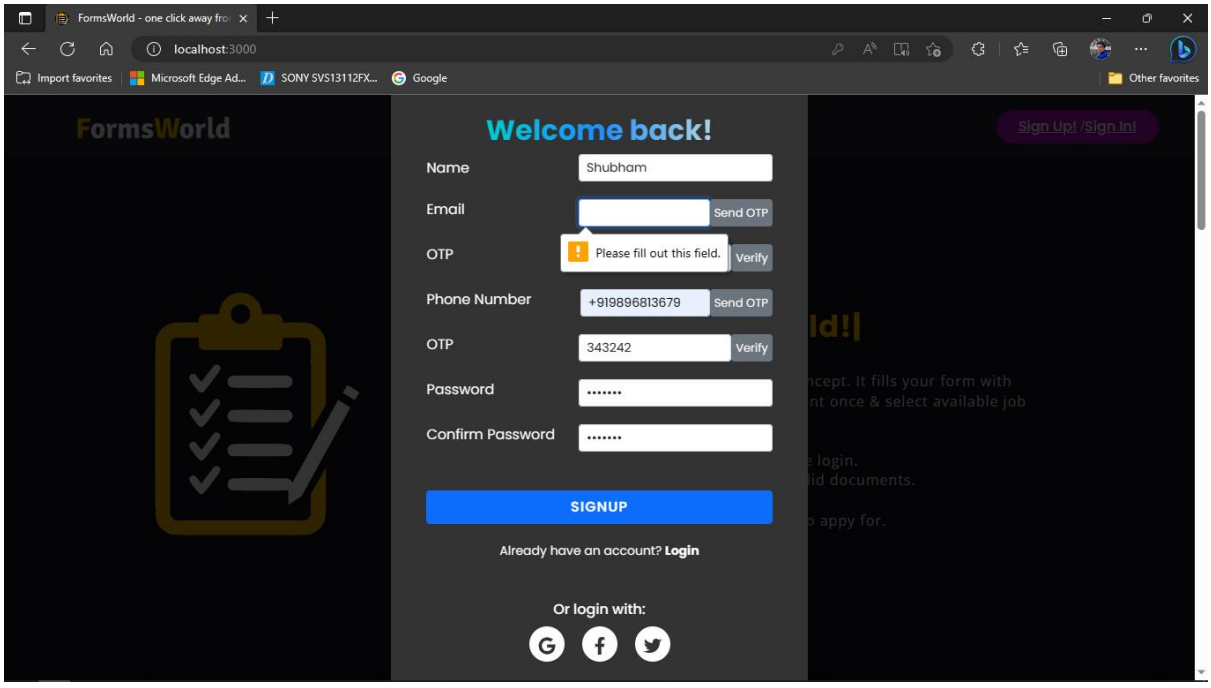
SignUp

1. Insufficient details

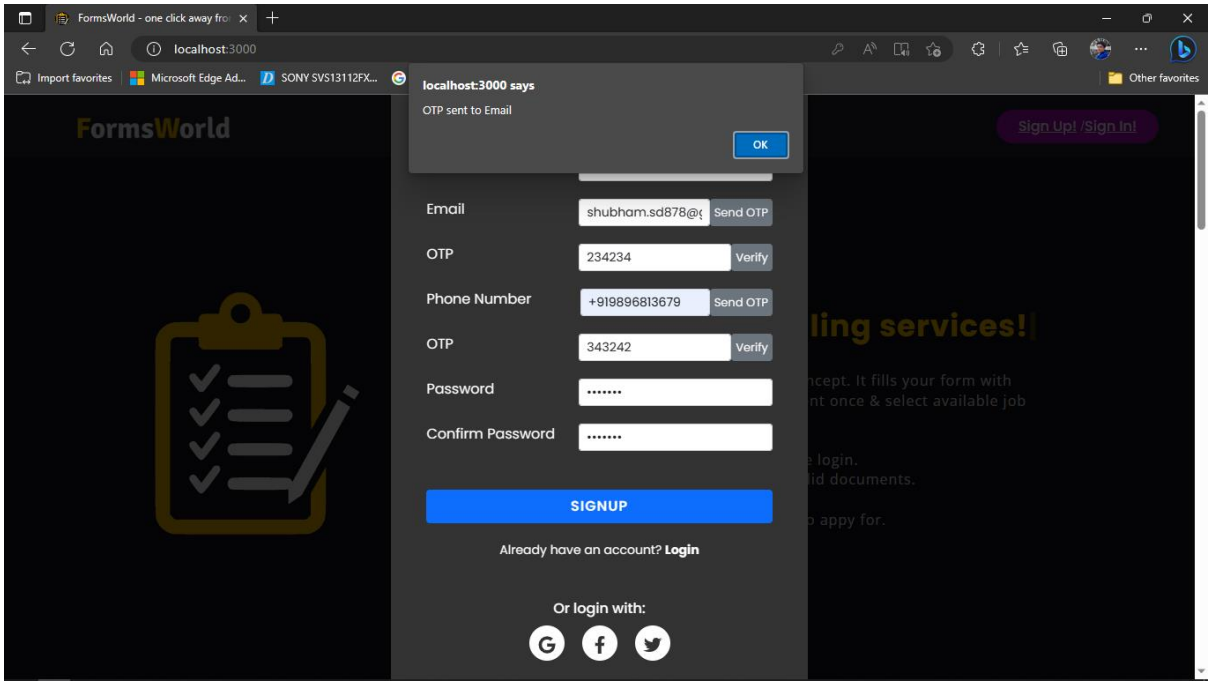


If all the details are not filled up then sign up will not be completed. Prompt will be showing one by one to fill up the details.

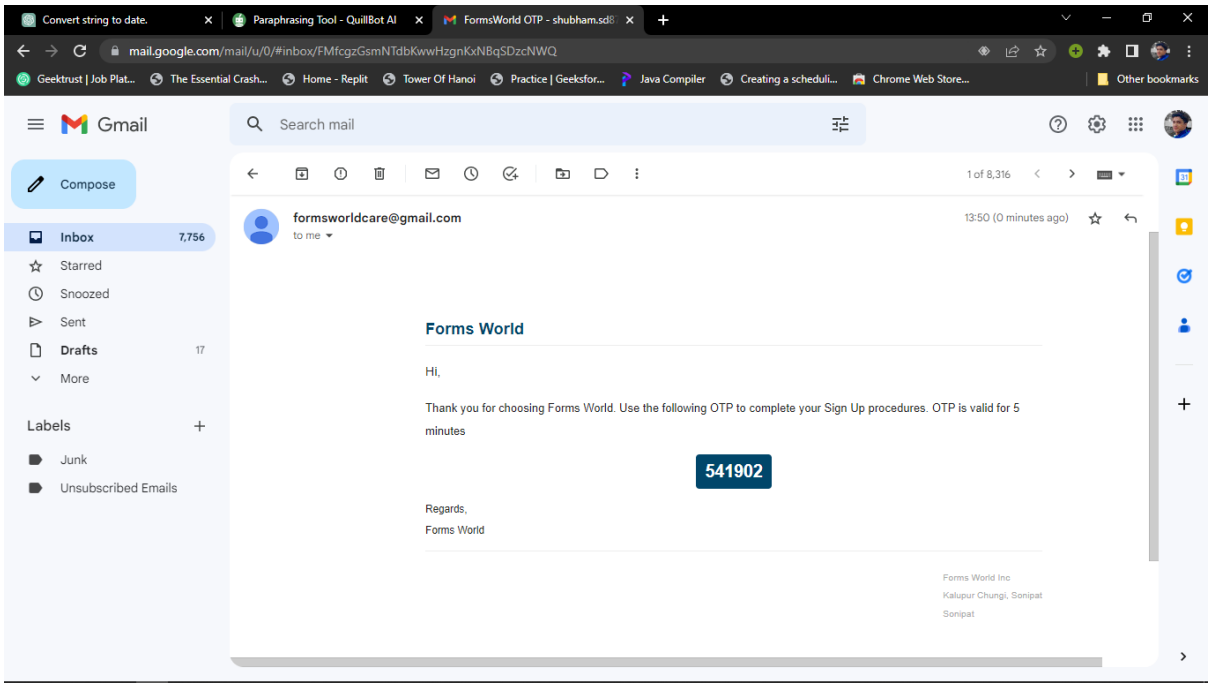
2. No Email or improper email format



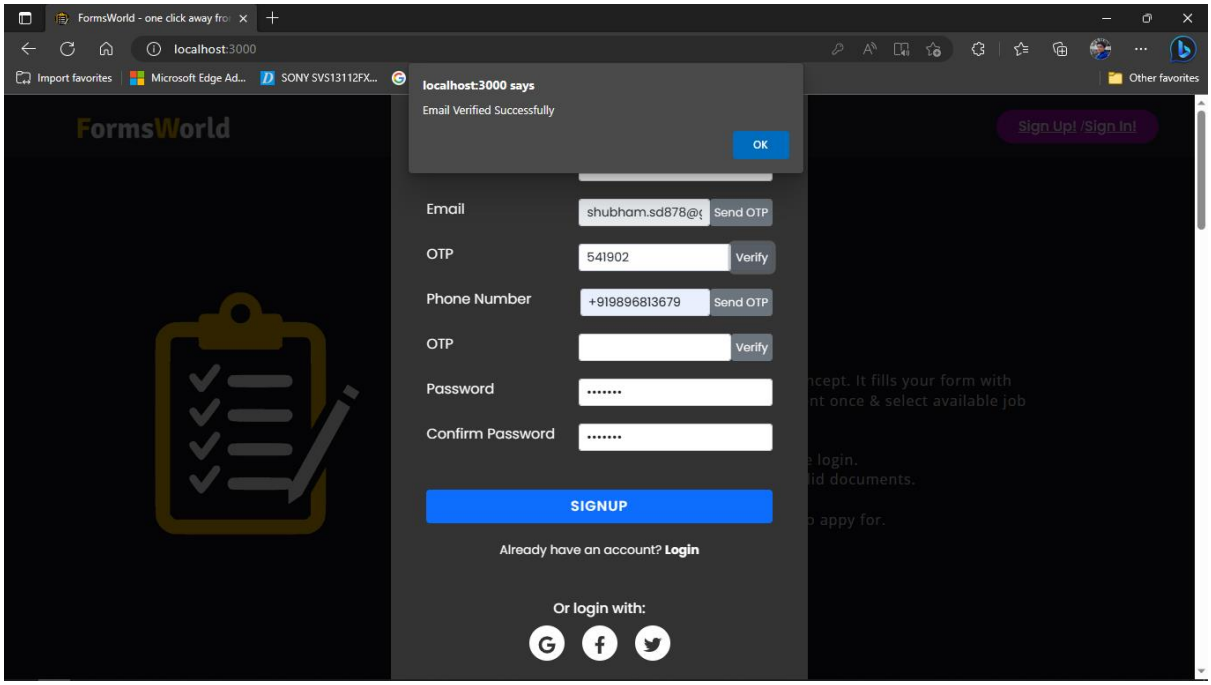
3. Send otp to mail



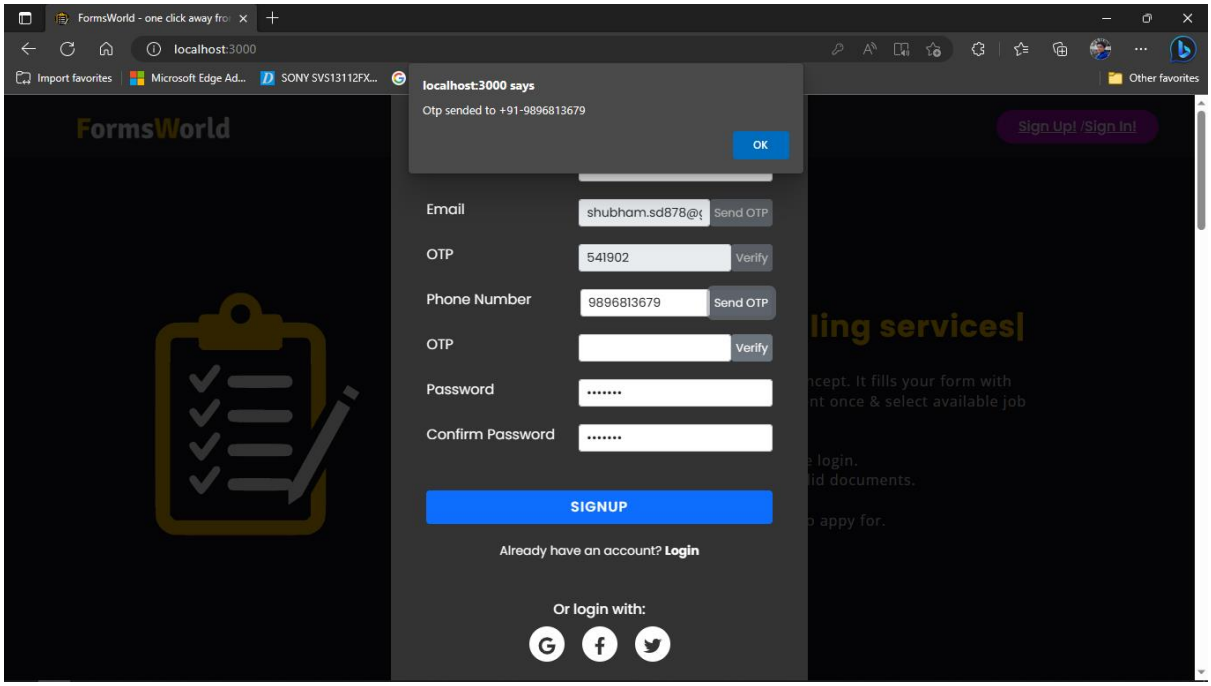
Otp on mail



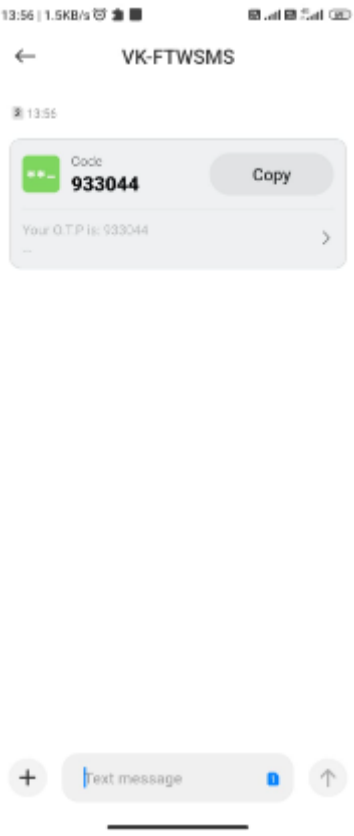
4. Verify Email otp



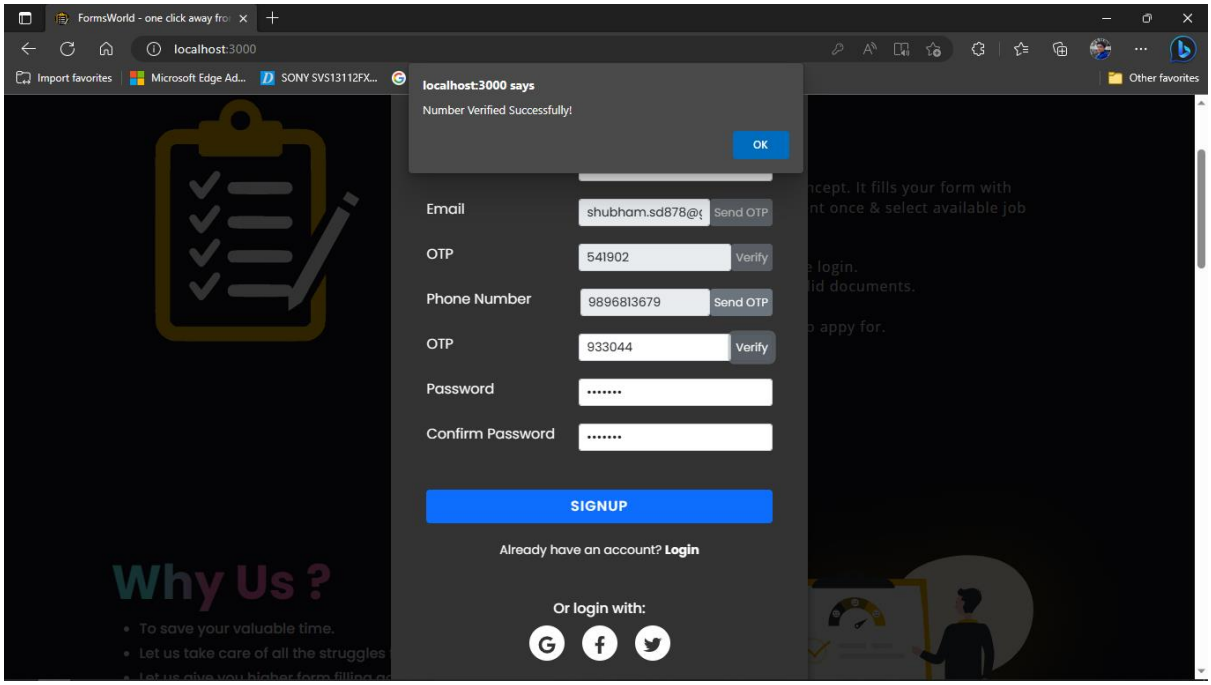
5. Sent otp to number



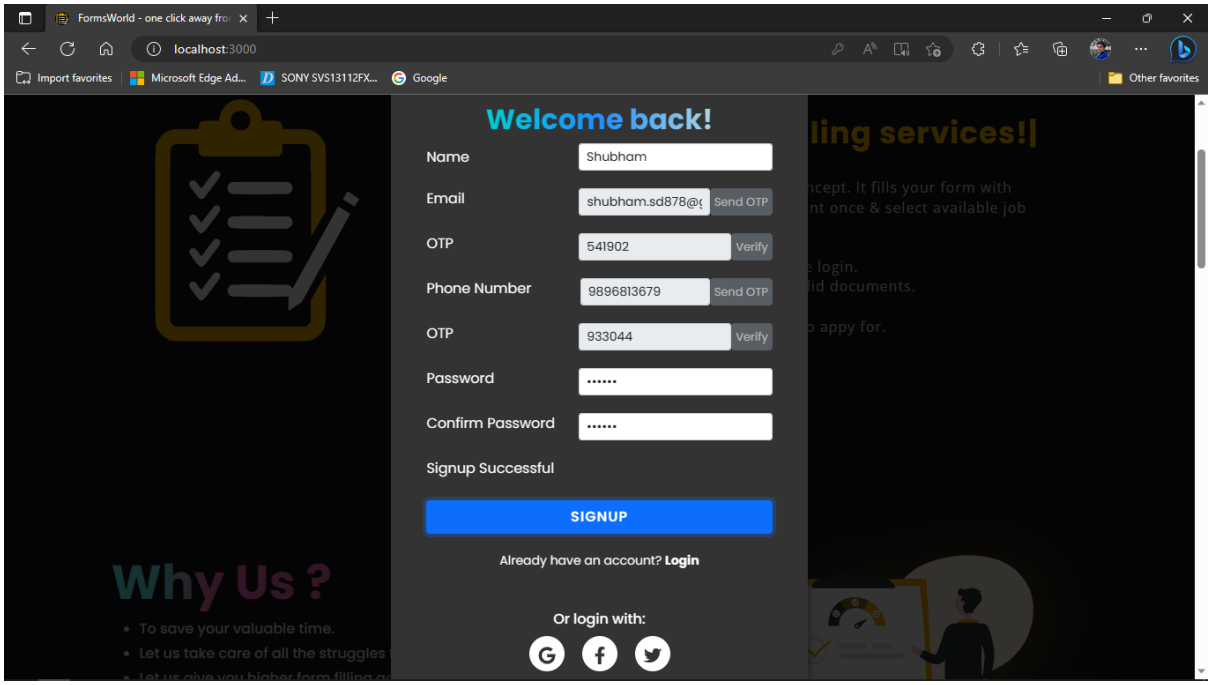
Phone otp:



6. Verify Number otp

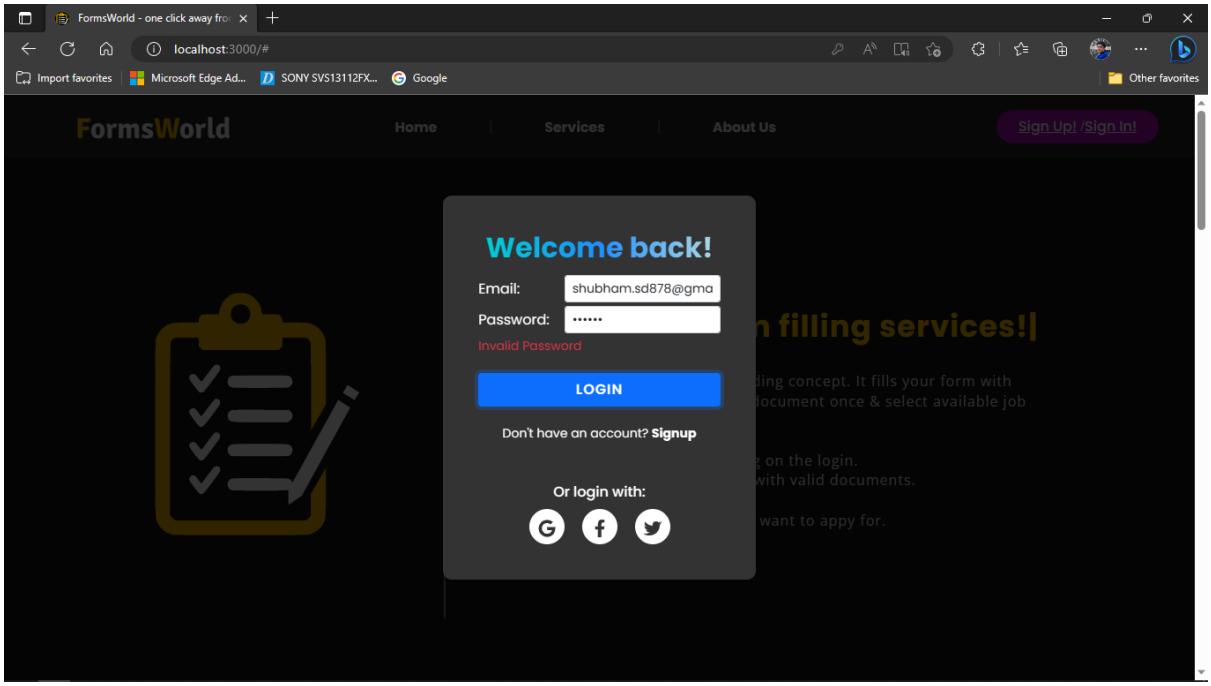


7. All Details correctly filled

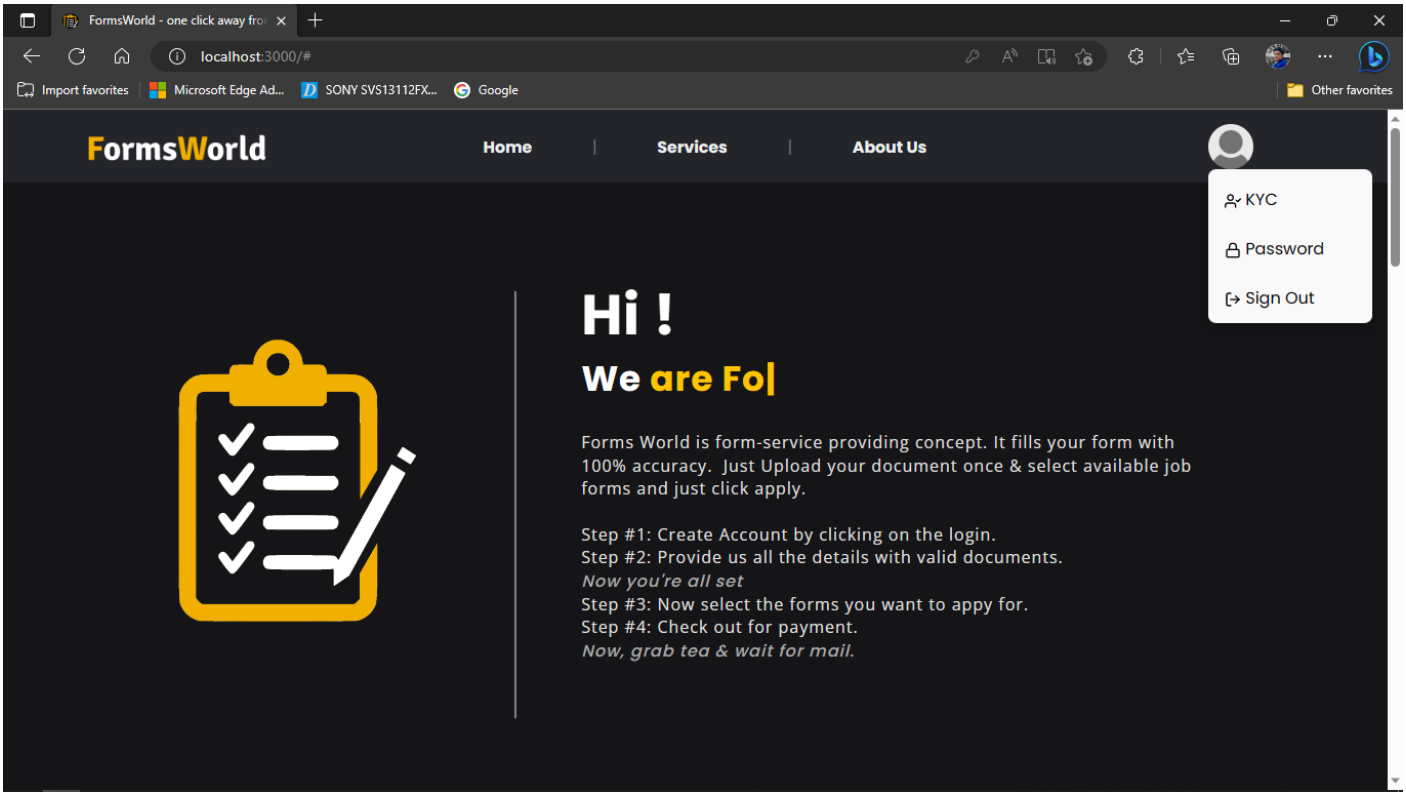


Login

8. Enter incorrect details



9. After login redirect to home page



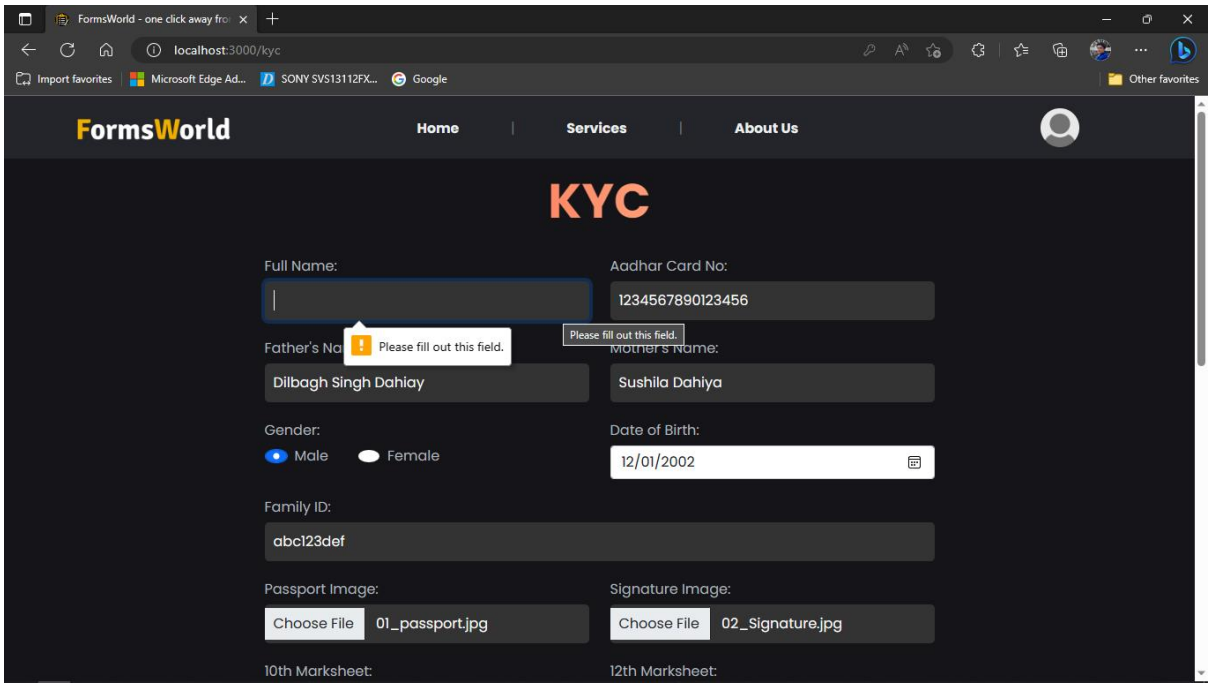
If a user filles email and password correct then user will be redirected to home page of swagZinn showing posts of following users.

2. User KYC

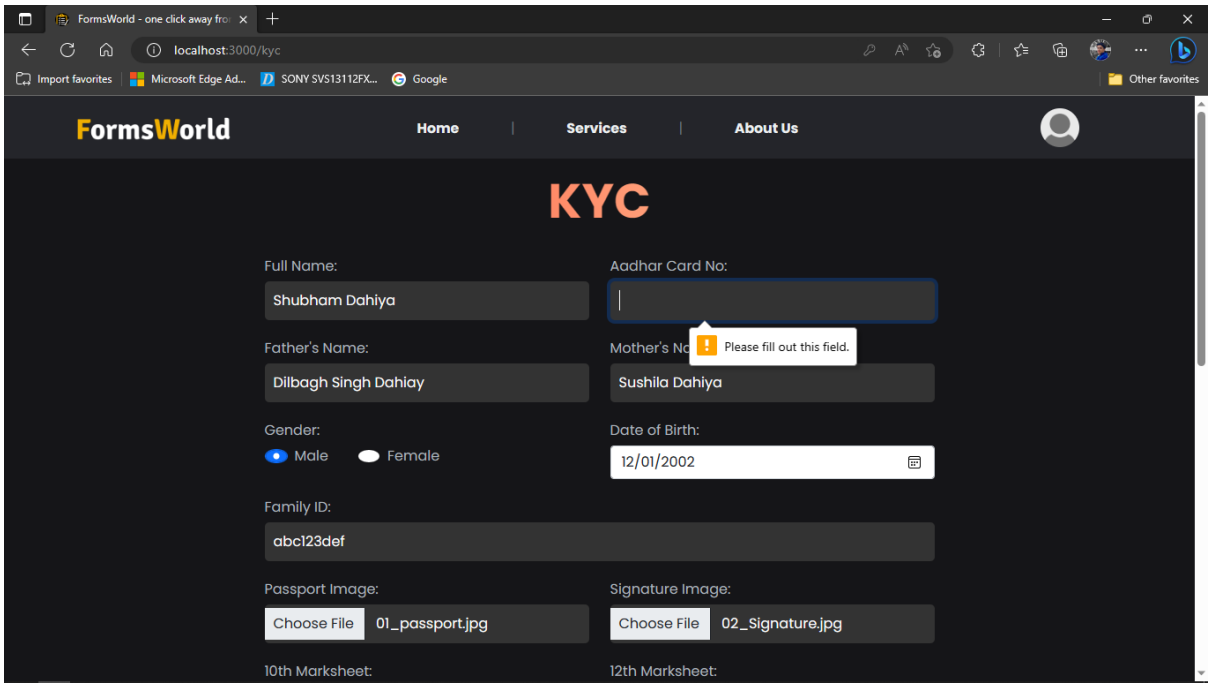
TEST	CASE	EXPECTED	RESULT
Incomplete KYC	KYC without Full Name	Error to fill the details	PASS
Incomplete KYC	KYC without Full Aadhar No.	Error to fill the details	PASS
Incomplete KYC	KYC without Father's Name	Error to fill the details	PASS
Incomplete KYC	KYC without Mother's Name	Error to fill the details	PASS
Incomplete KYC	KYC without Gender	Error to fill the details	PASS
Incomplete KYC	KYC without DOB	Error to fill the details	PASS
Incomplete KYC	KYC without Family Id	Error to fill the details	PASS
Incomplete KYC	KYC without Passport Image	File upload	PASS
Incomplete KYC	KYC without Signature Image	File upload failed	PASS
Incomplete KYC	KYC without 10^{th} Marksheet Image	File upload failed	PASS
Incomplete KYC	KYC without 12^{th} Marksheet Image	File upload failed	PASS
Incomplete KYC	KYC without Domicile Image	File upload failed	PASS
Incomplete KYC	KYC without Full Caste Certificate Image	File upload failed	PASS
Incomplete KYC	KYC without Left Thumb Impression Image	File upload failed	PASS
Incomplete KYC	KYC without Right Thumb Impression Image	File upload failed	PASS
Complete KYC	KYC with all details	Alert to notify user that KYC is Successful	PASS

Table 12.2 User kyc Testing

1. No Name

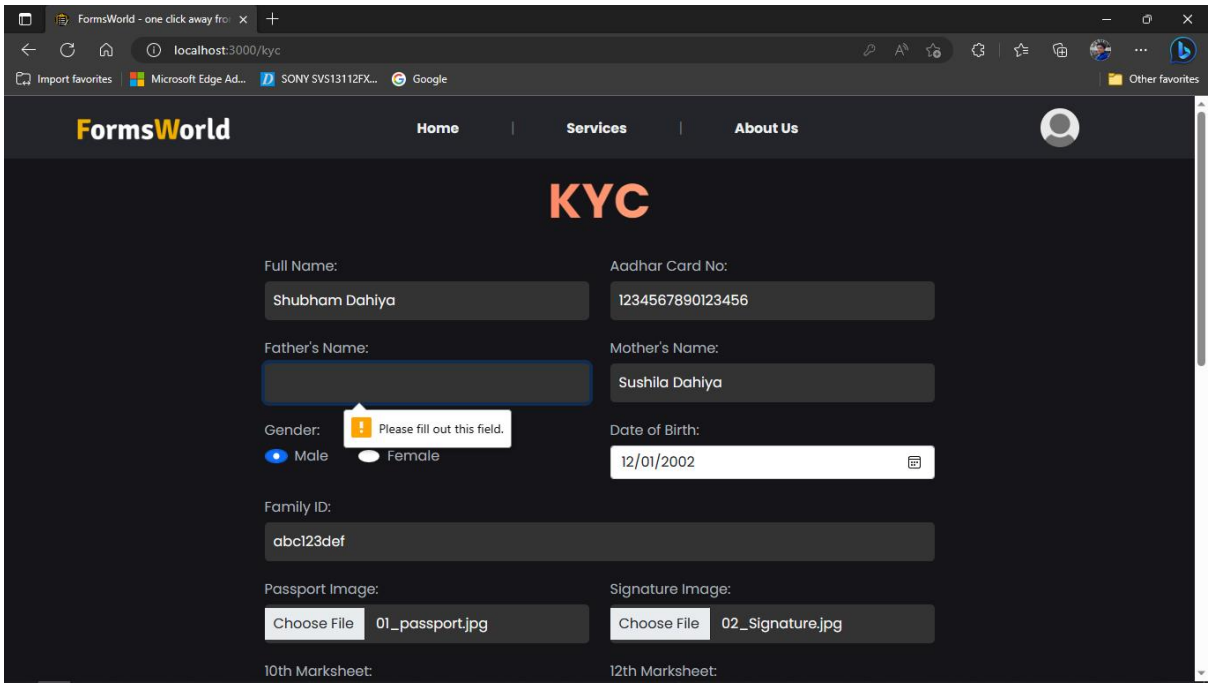


2. No Aadhar No.

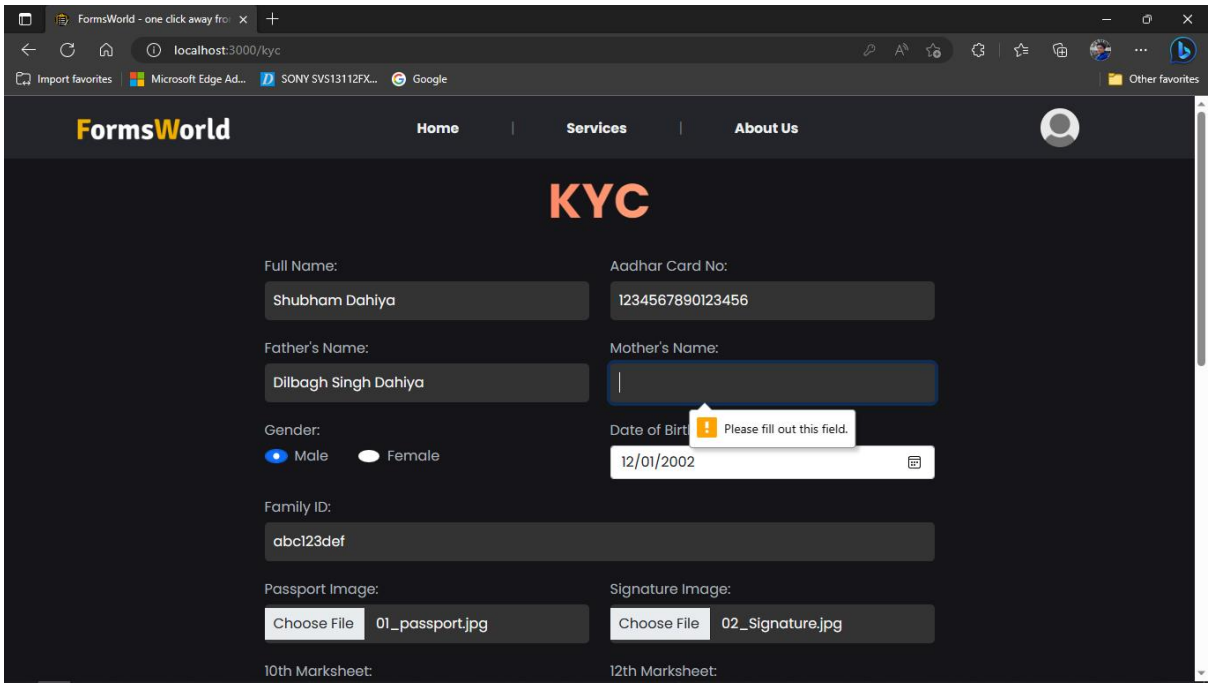


If user tries to upload post without file then operation will be rejected by Forms World.

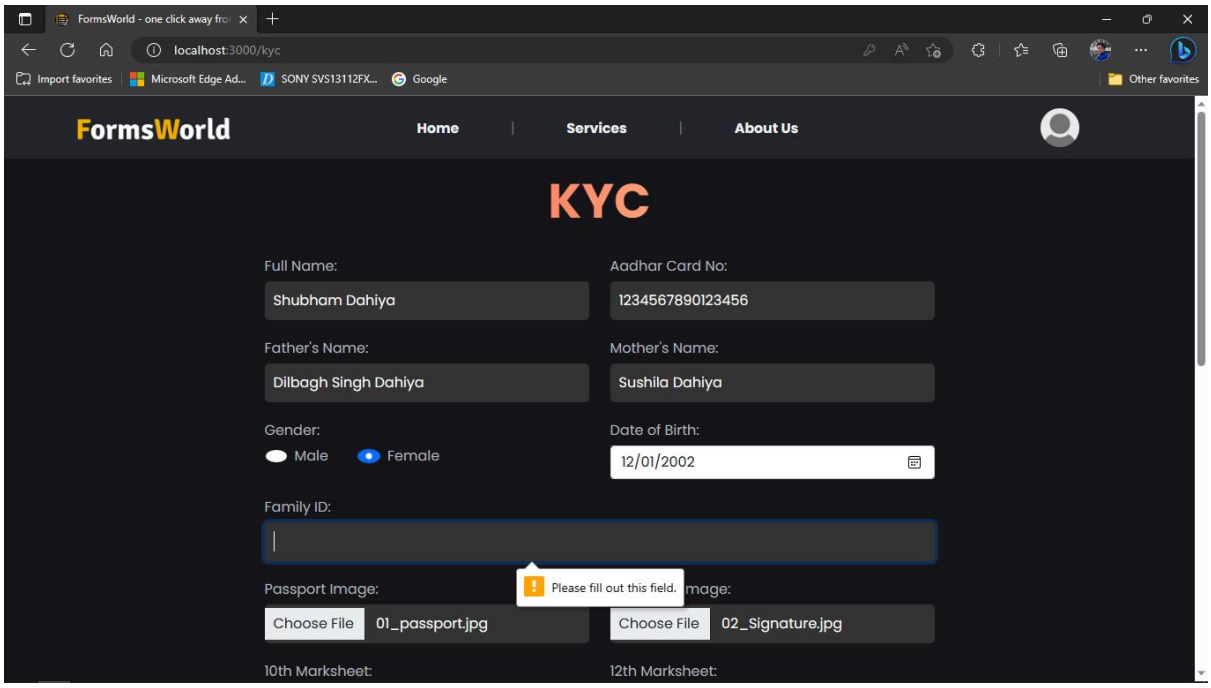
3. Without Father’s Name



4. No Mother’s Name

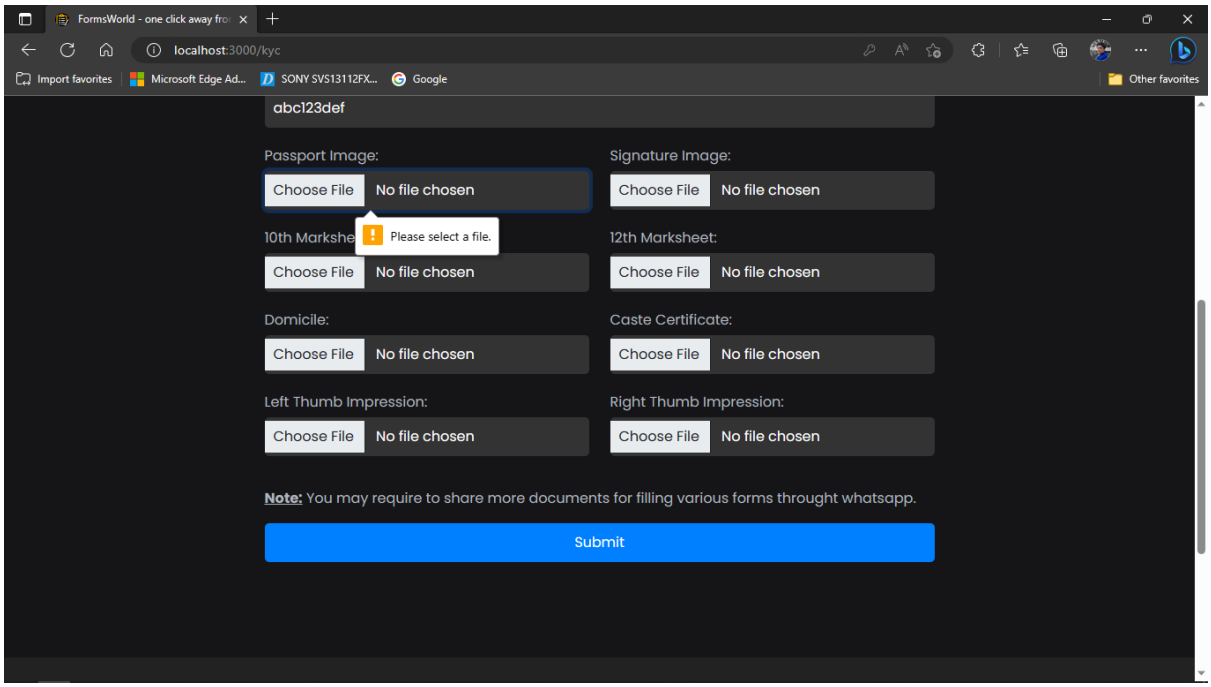


5. No family Id

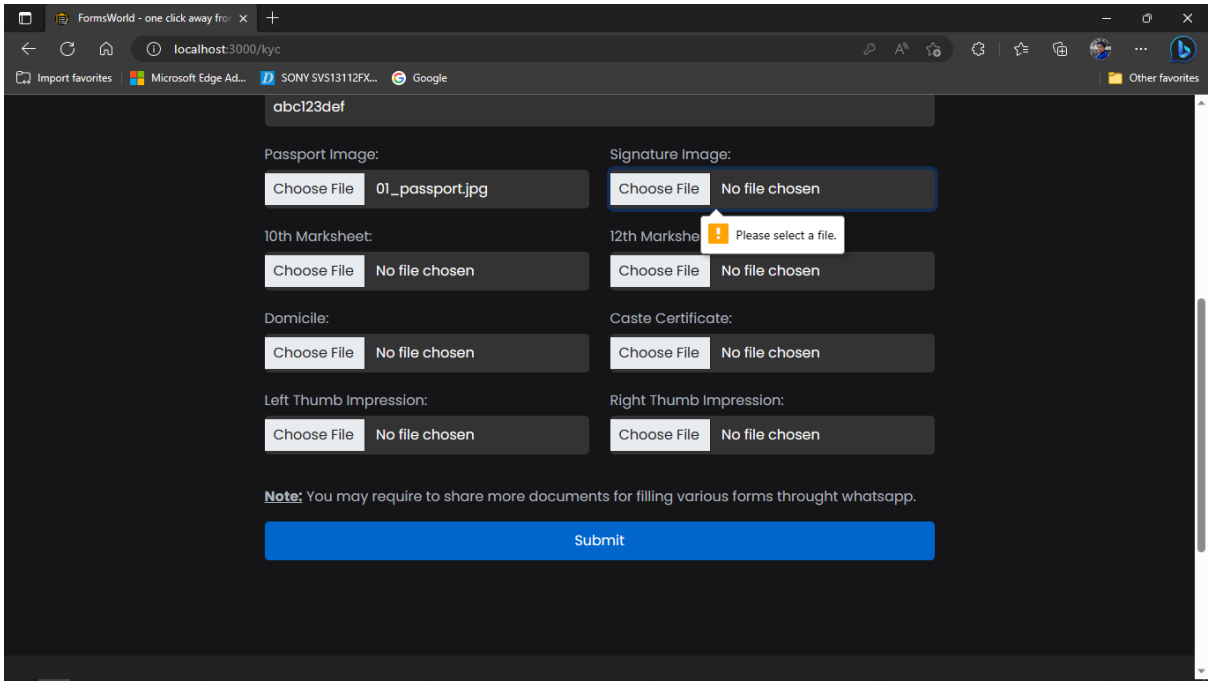


If user tries to upload empty post, then it will be rejected by Forms World.

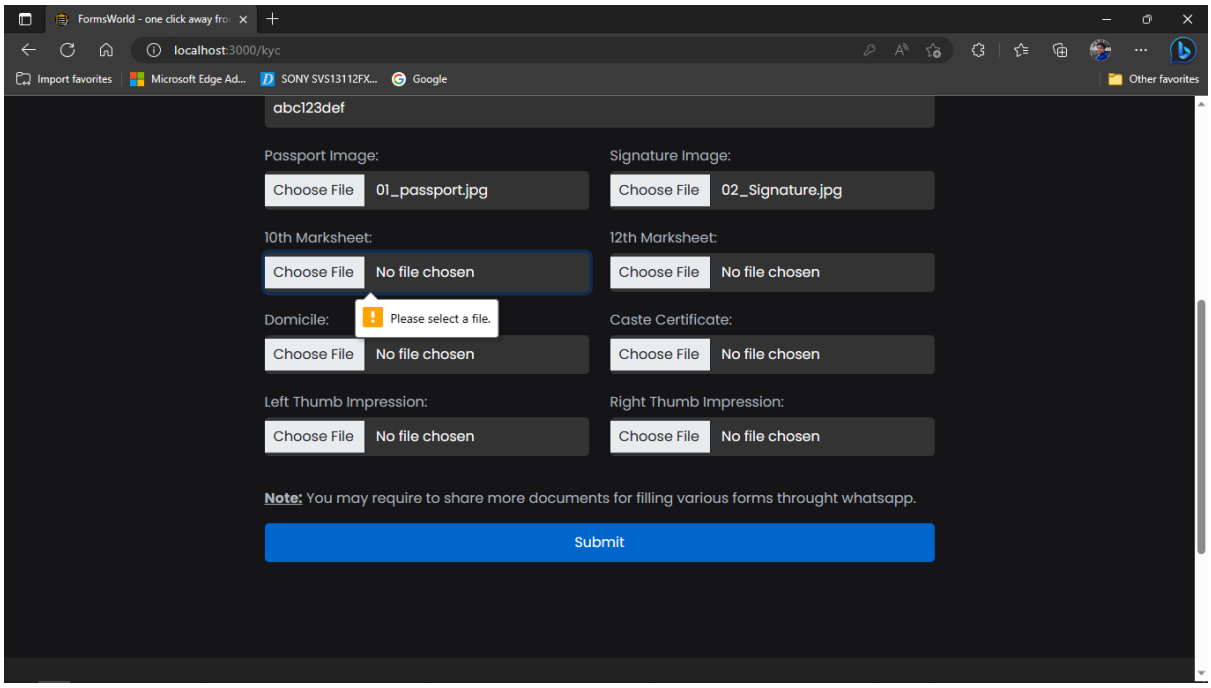
6. Without Passport Image



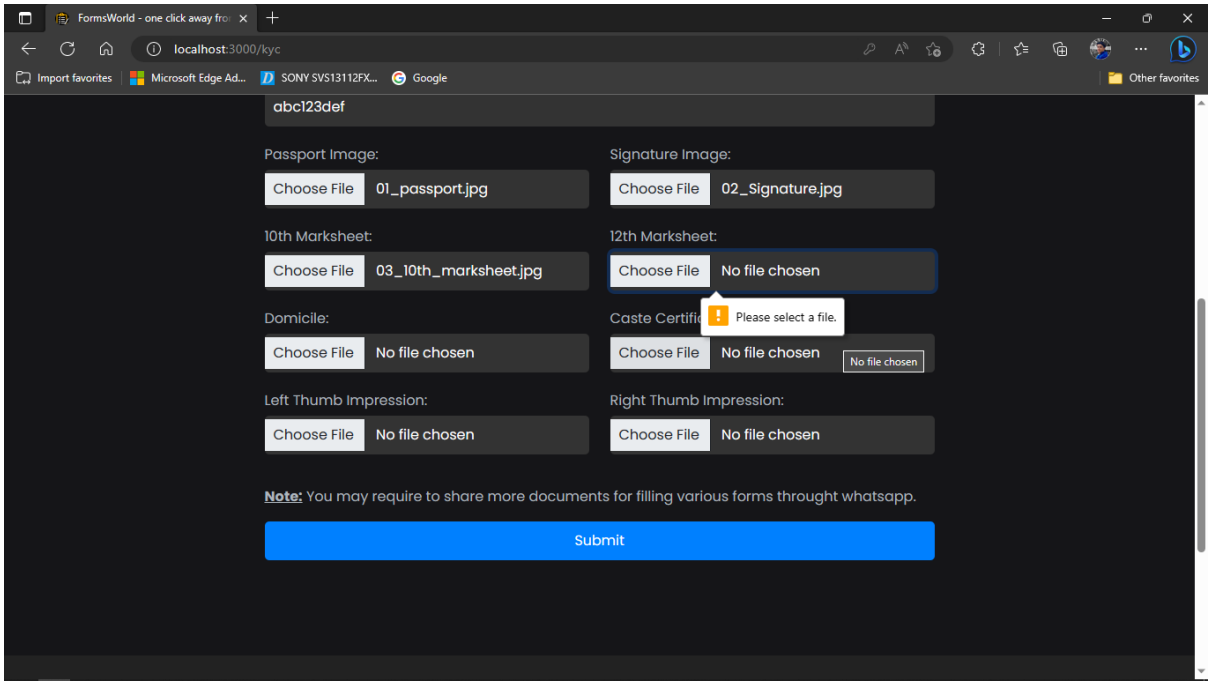
7. Without Signature Image



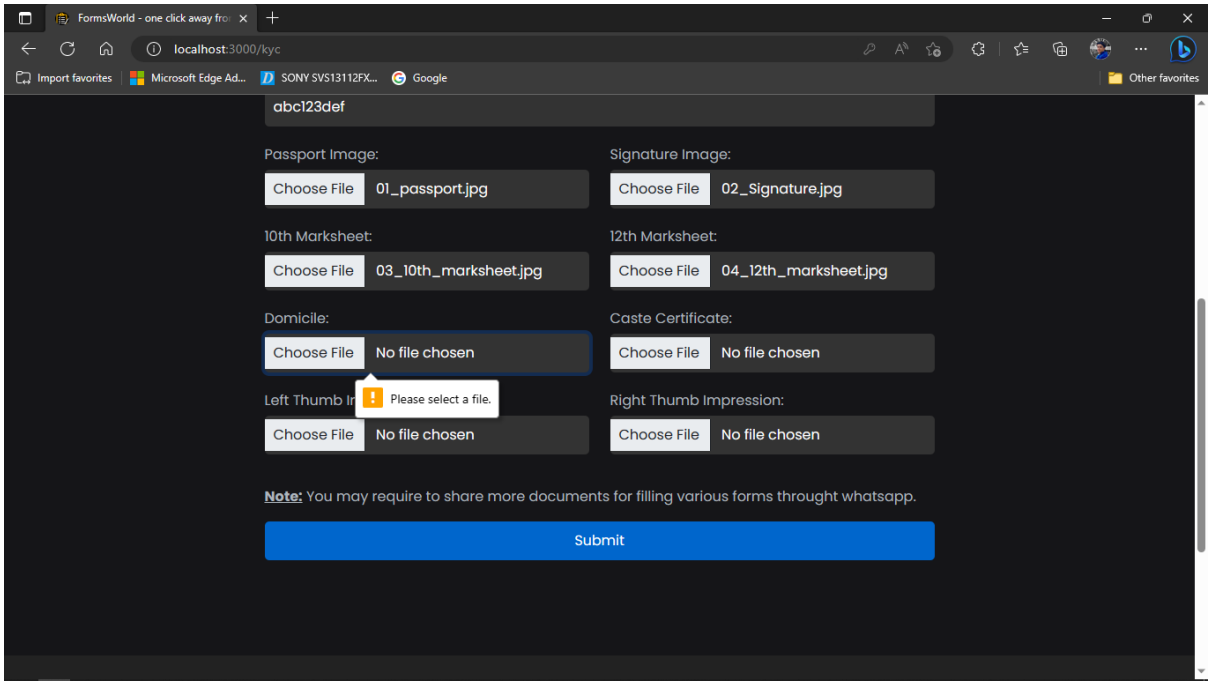
8. Without 10th Marksheet



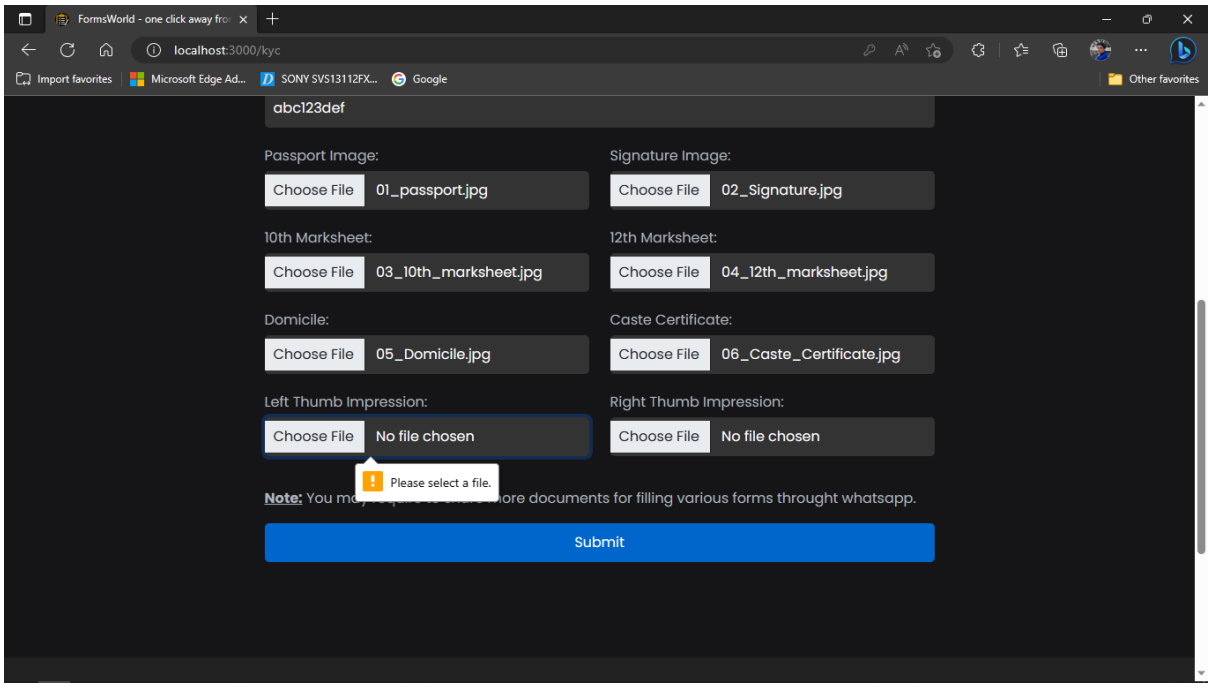
9. Without 12th Marksheet



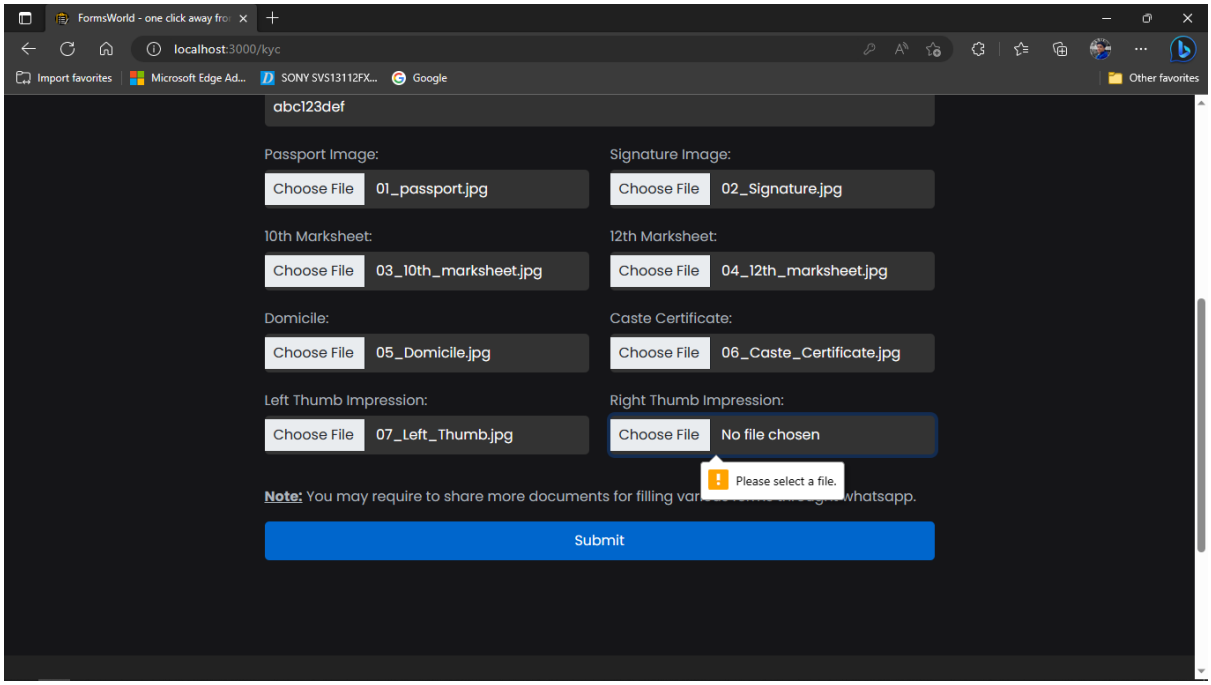
10. Without Domicile



11. Without Left Thumb Impression



12.Without Right Thumb Impression

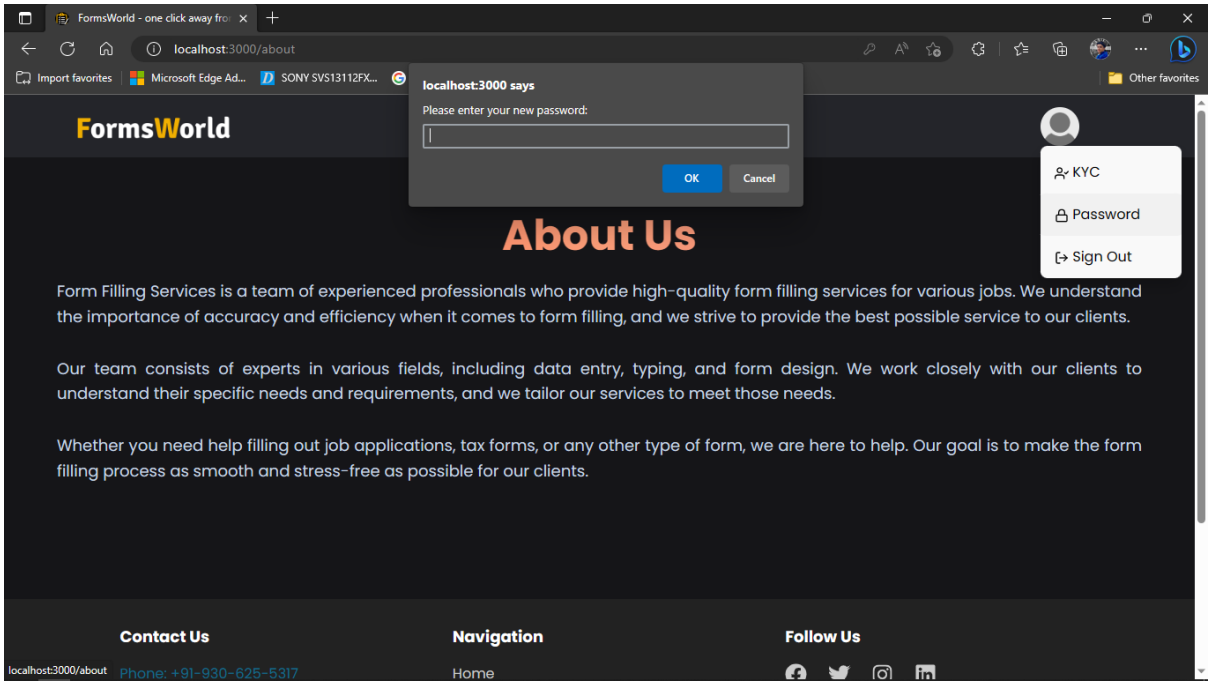


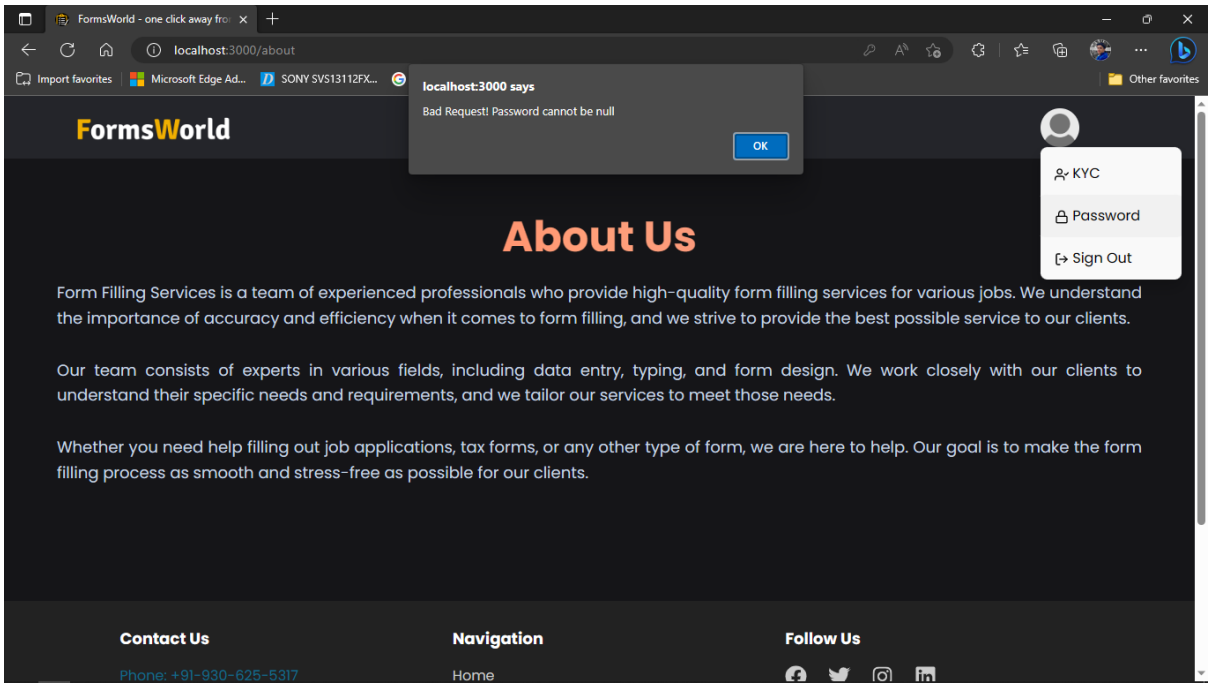
3. Updating user password

TEST	CASE	EXPECTED	RESULT
Updating Password	Blank Password	Alert to show Bad Request	PASS
Updating Password	New password	Account found	PASS

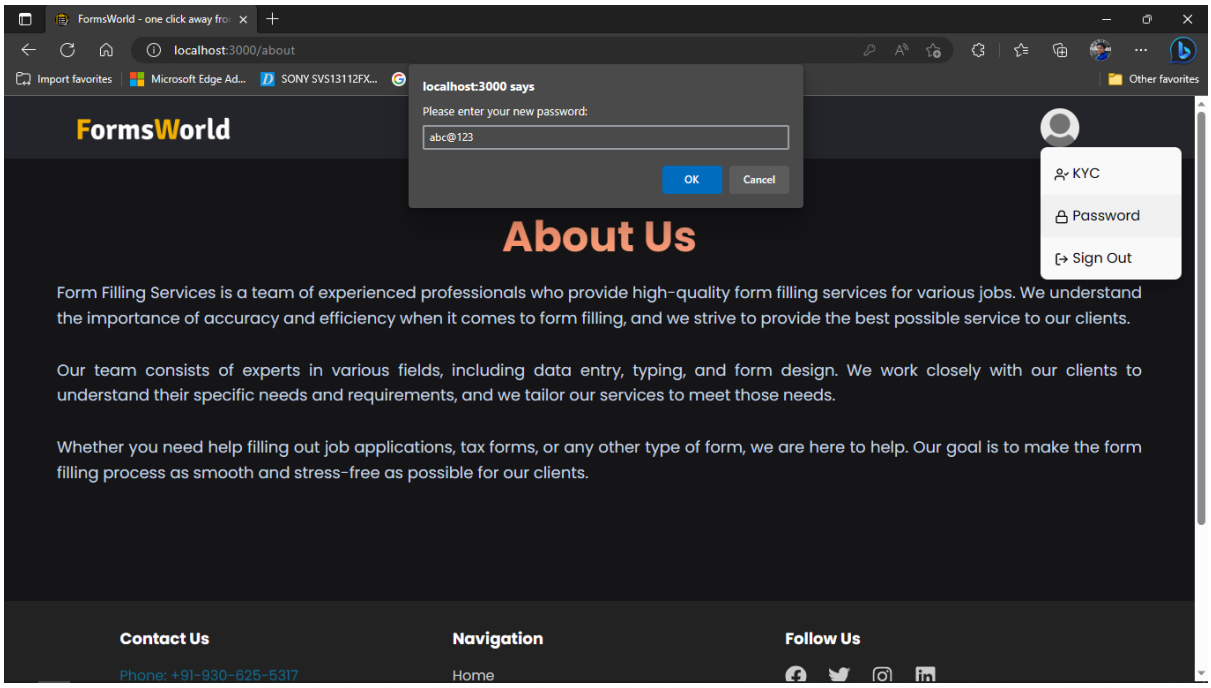
Table 12.3 Updating Password Testing

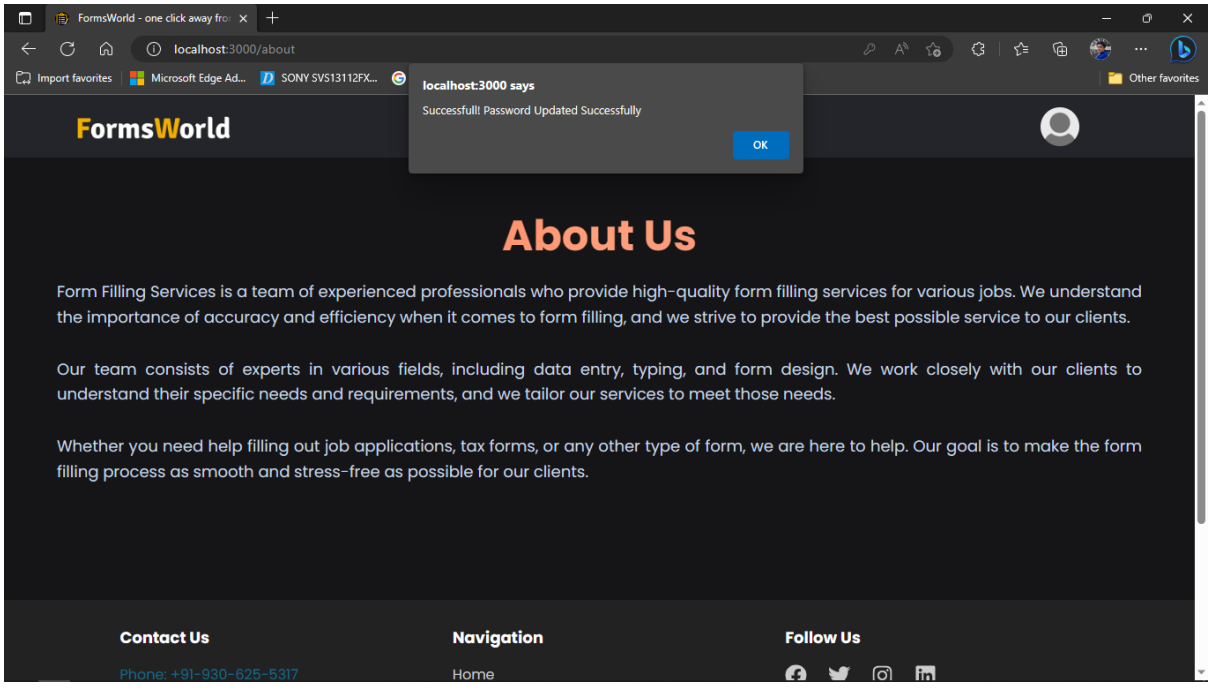
1. Blank Password





2. New Password



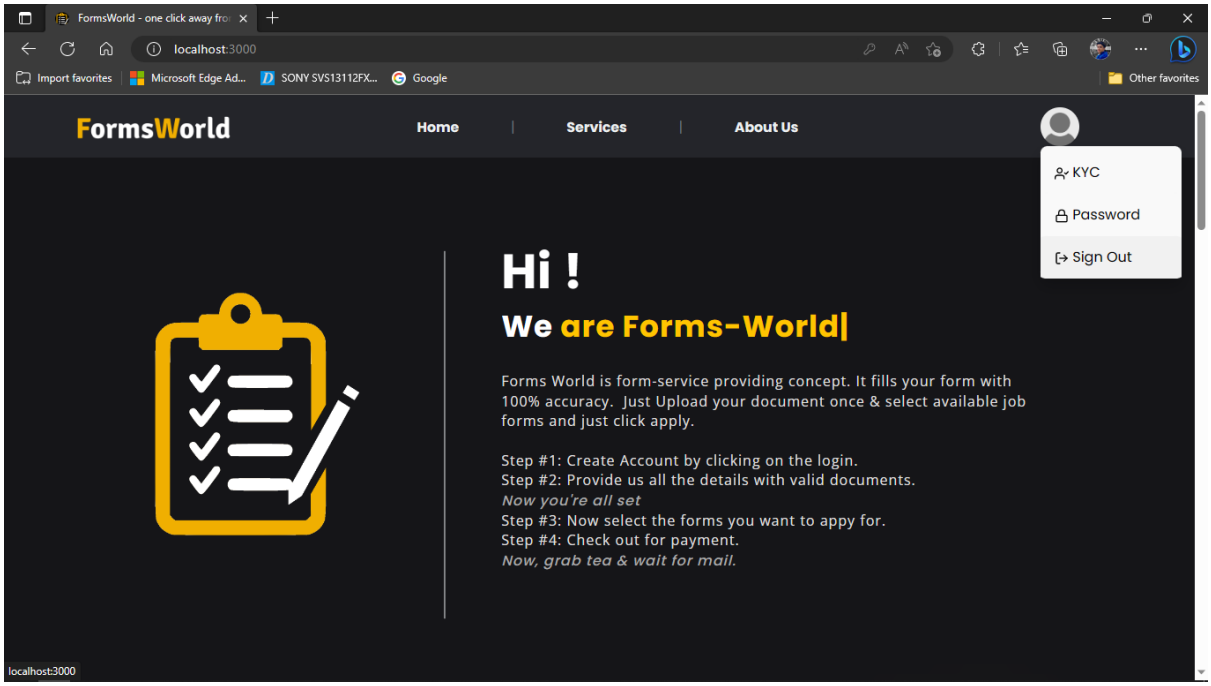


4. Signout

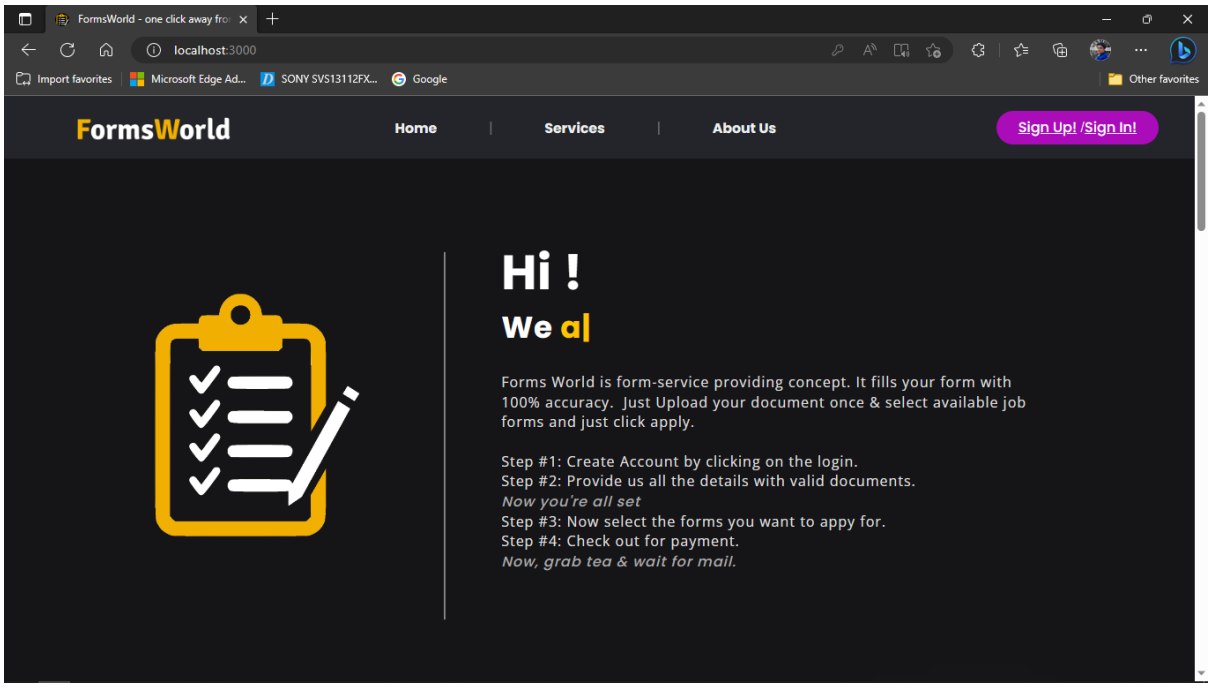
TEST	CASE	EXPECTED	RESULT
Signout	Signout User	Signout the user from signin state	PASS

Table 12.4 Sign out Testing.

1. Signout



Successful Signout

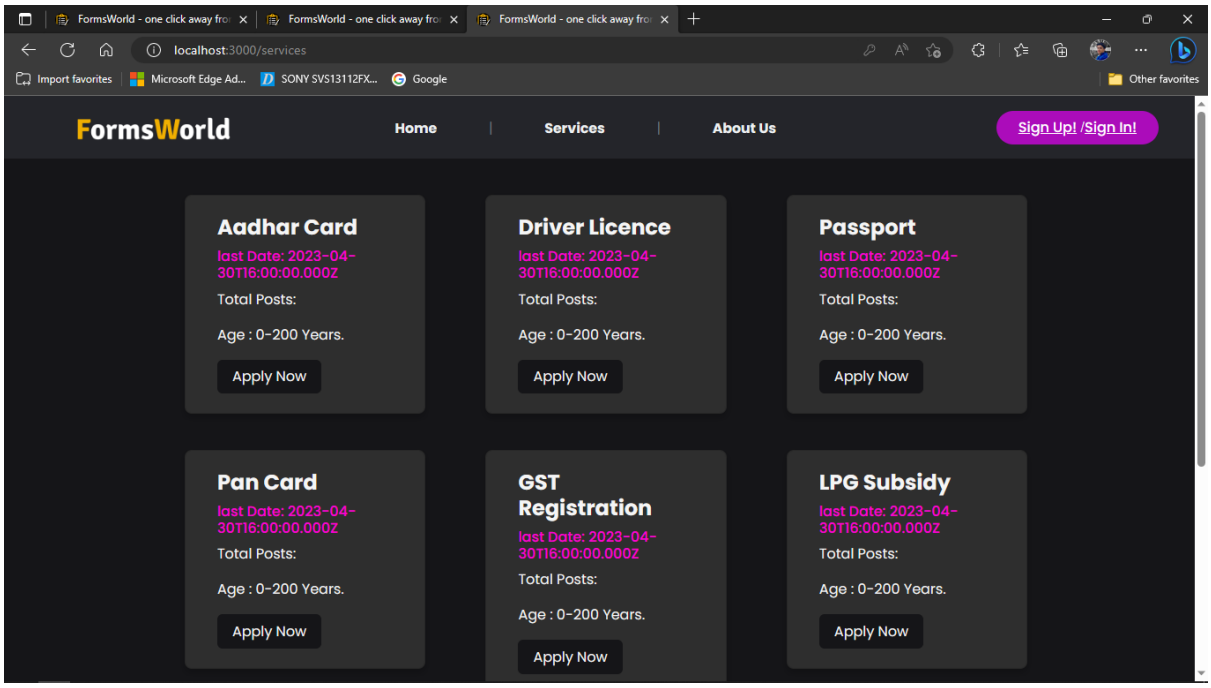


5. Services

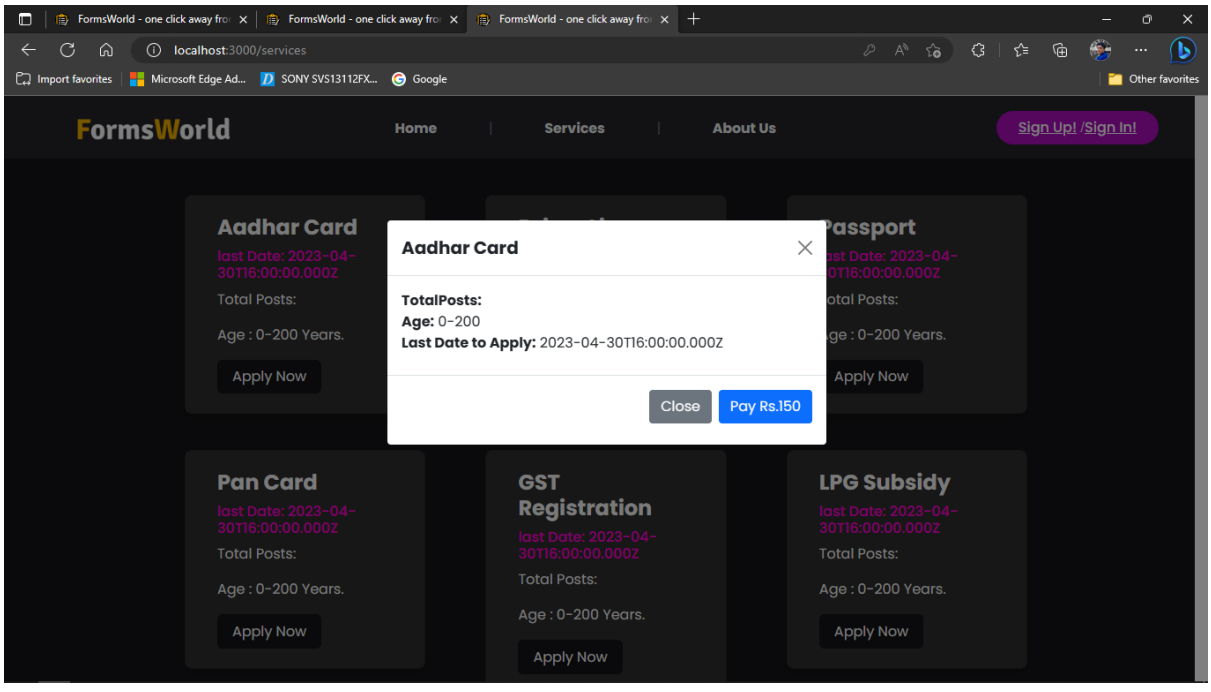
TEST	CASE	EXPECTED	RESULT
Services	Click Apply Now	Popup Modal containing Form Details	PASS
Services	Click Pay Rs. X without Signup	Alert to Signup	PASS
Services	Click Pay Rs. X	Open Payment Gateway for Payment of X	PASS
Services	Payment Success	Show Applied on Form Details Modal	PASS

Table 12.5 Services Testing

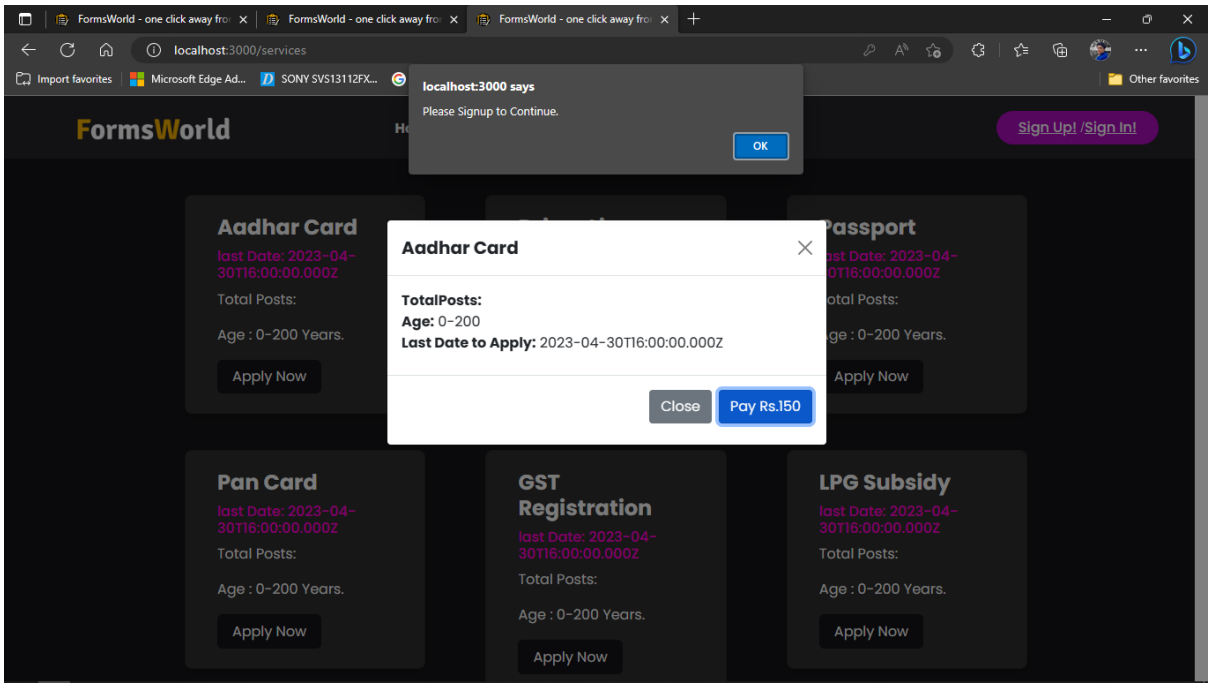
1. Click Apply Now



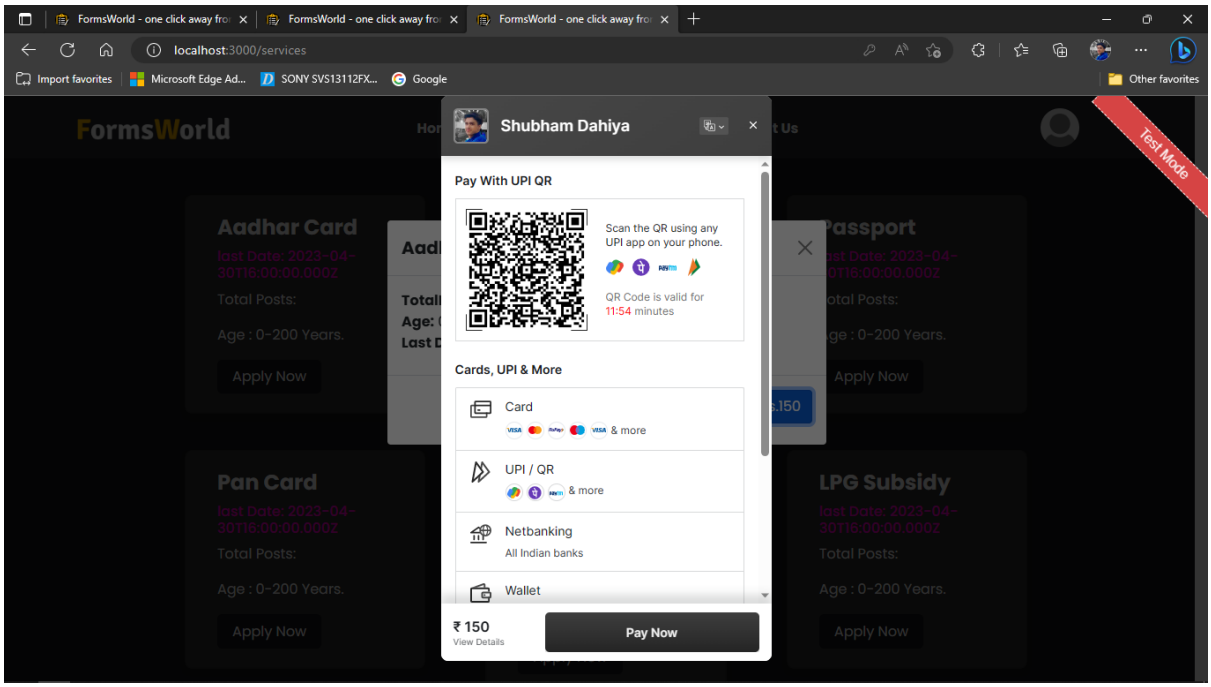
Apply Now clicked:



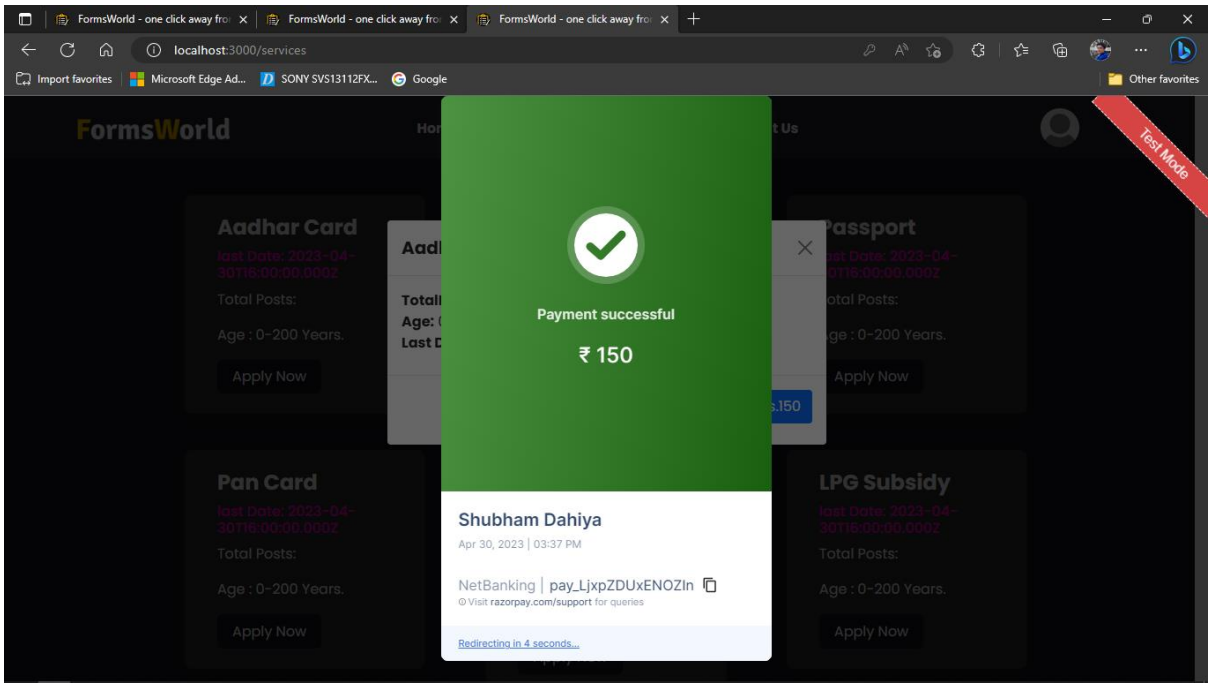
2. Click Pay Rs. X without Signup

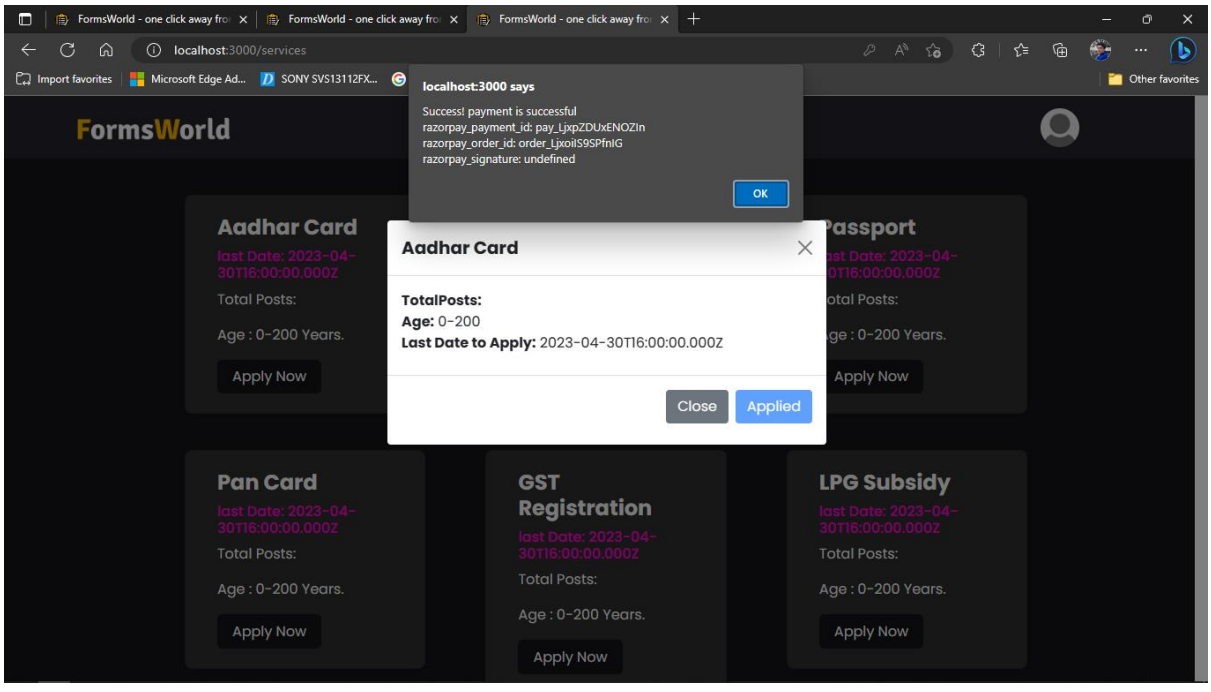


3. Click Pay Rs. X



4. Payment Success



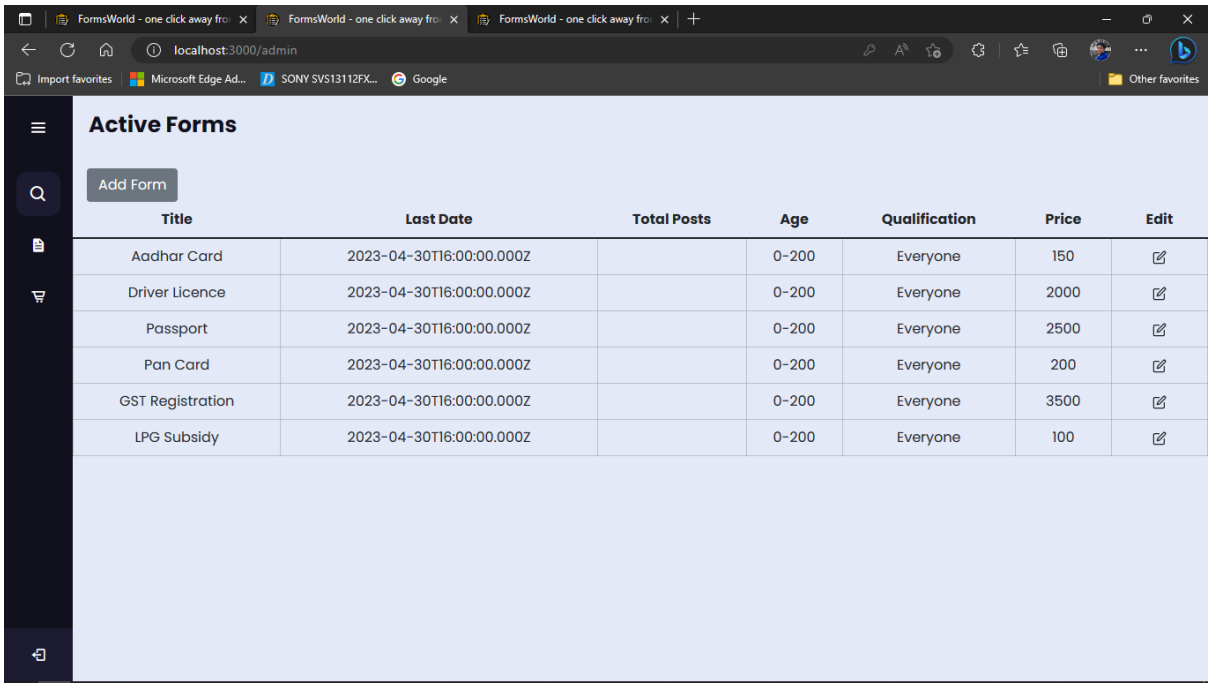


6. Admin/Active Forms

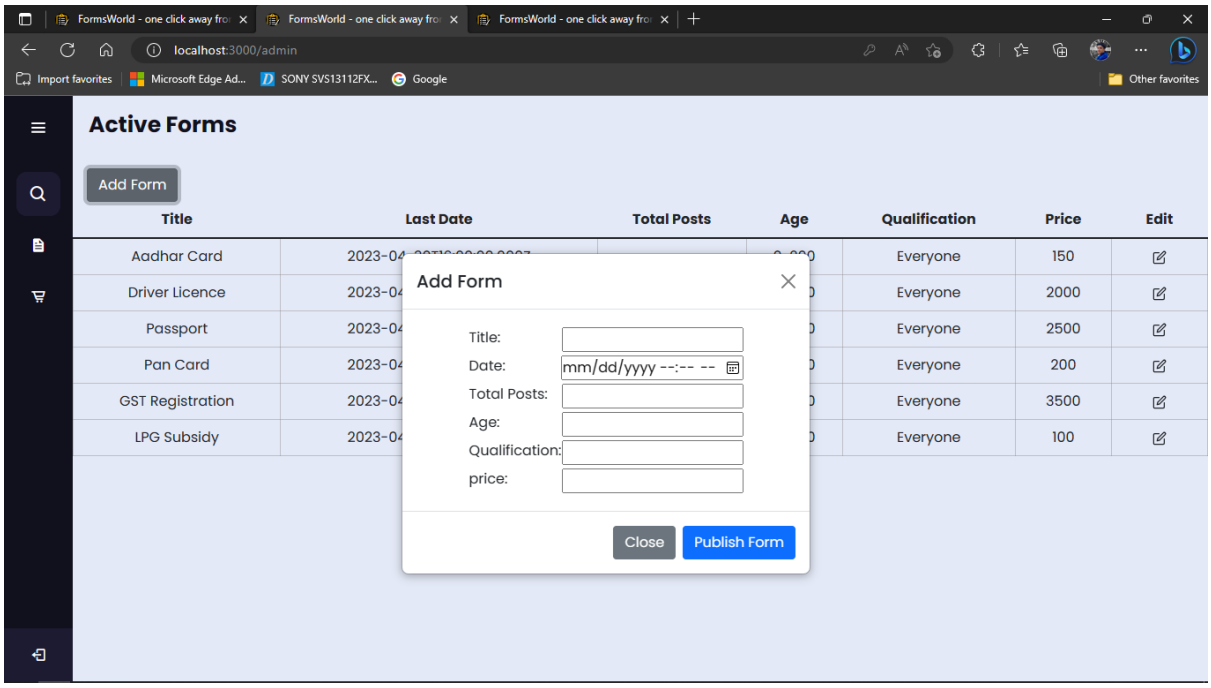
TEST	CASE	EXPECTED	RESULT
Add Form	clicking button	should display modal	PASS
Add Form	Clicking Publish From	Publish the newly Pusblished Form and shows to the Services page	PASS
Form Details	On Load page	Should fetch all the forms, active on the application.	PASS
Form Details	clicking edit icon	Opens the poup modal for modifying the previous details of form.	PASS
Update Form	Clicking update Form	Show alert of update, and update to the database and services tab across the application.	PASS

Table 12.6 Admin/Active Forms Testing

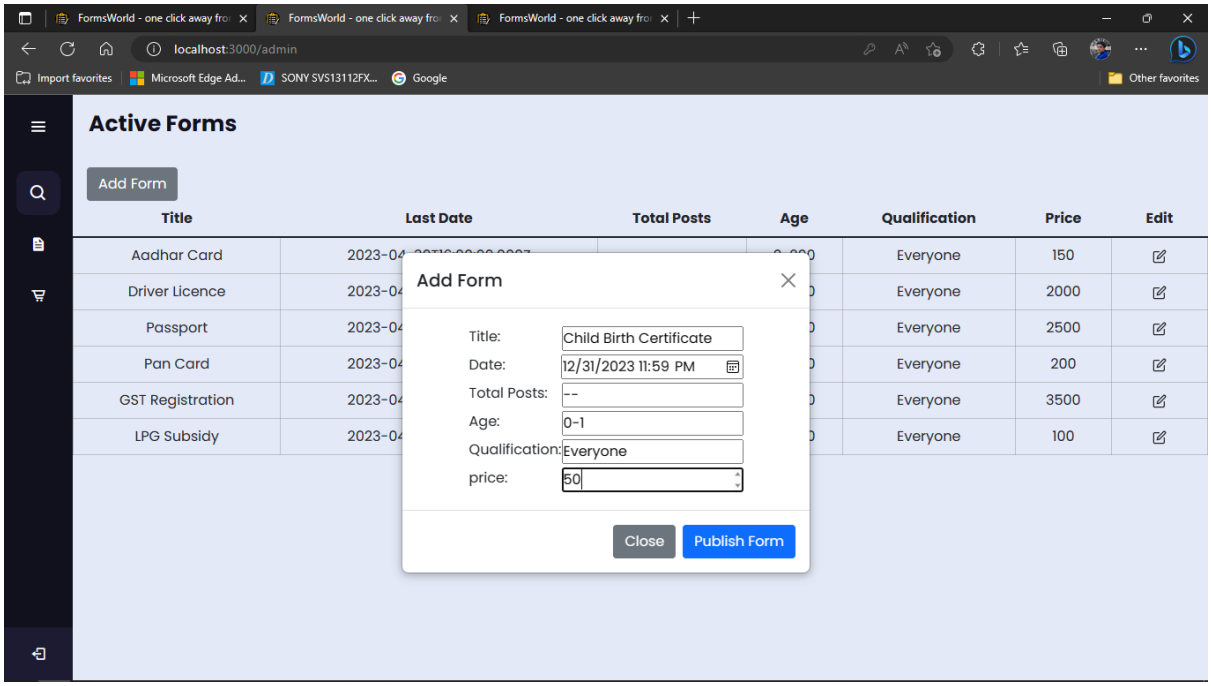
1. Add Form



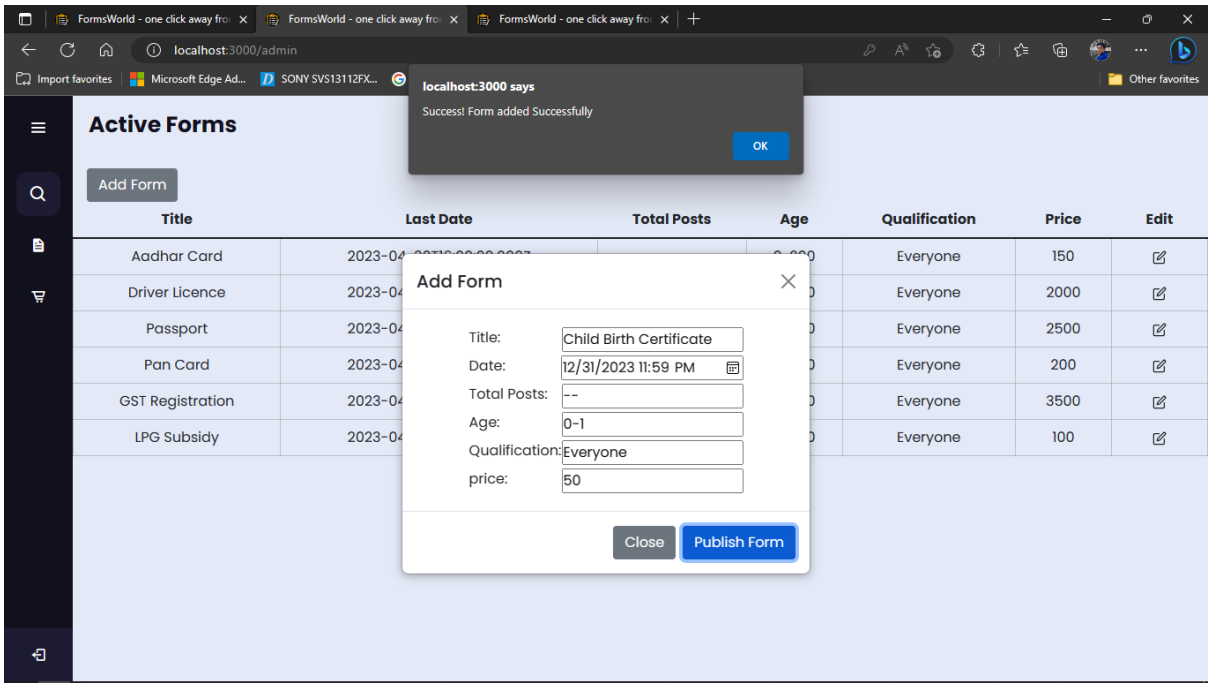
Add Form Clicked:



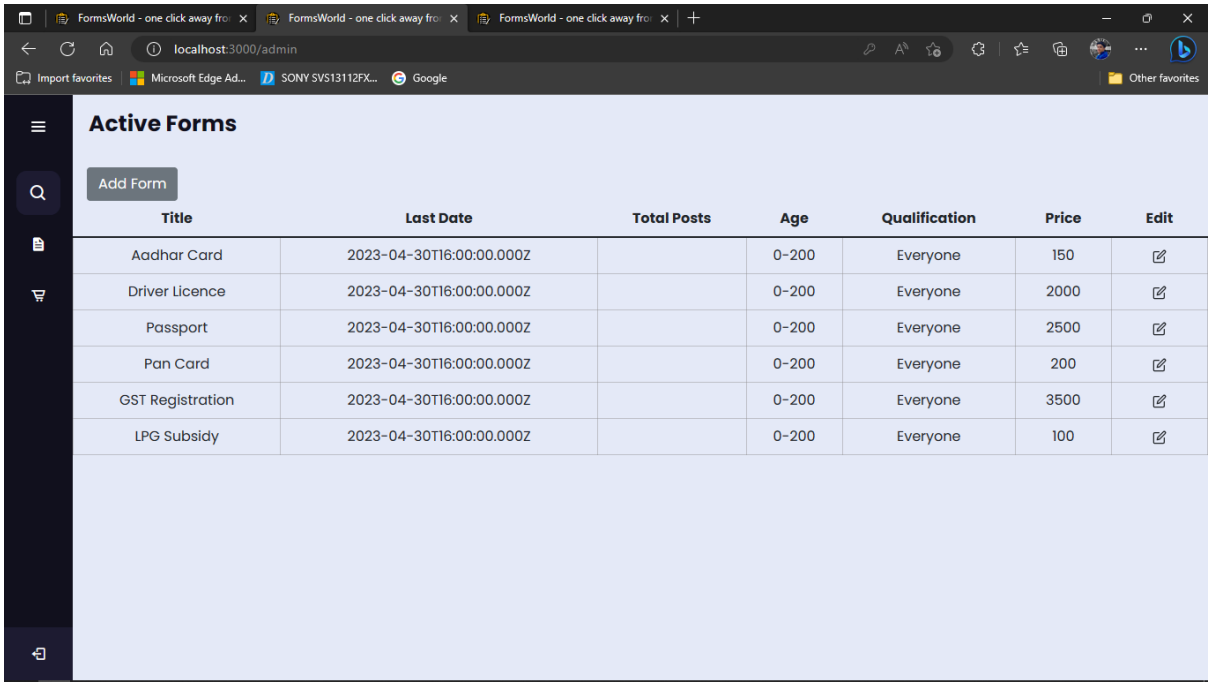
Adding Details



2. Click Publishing Form

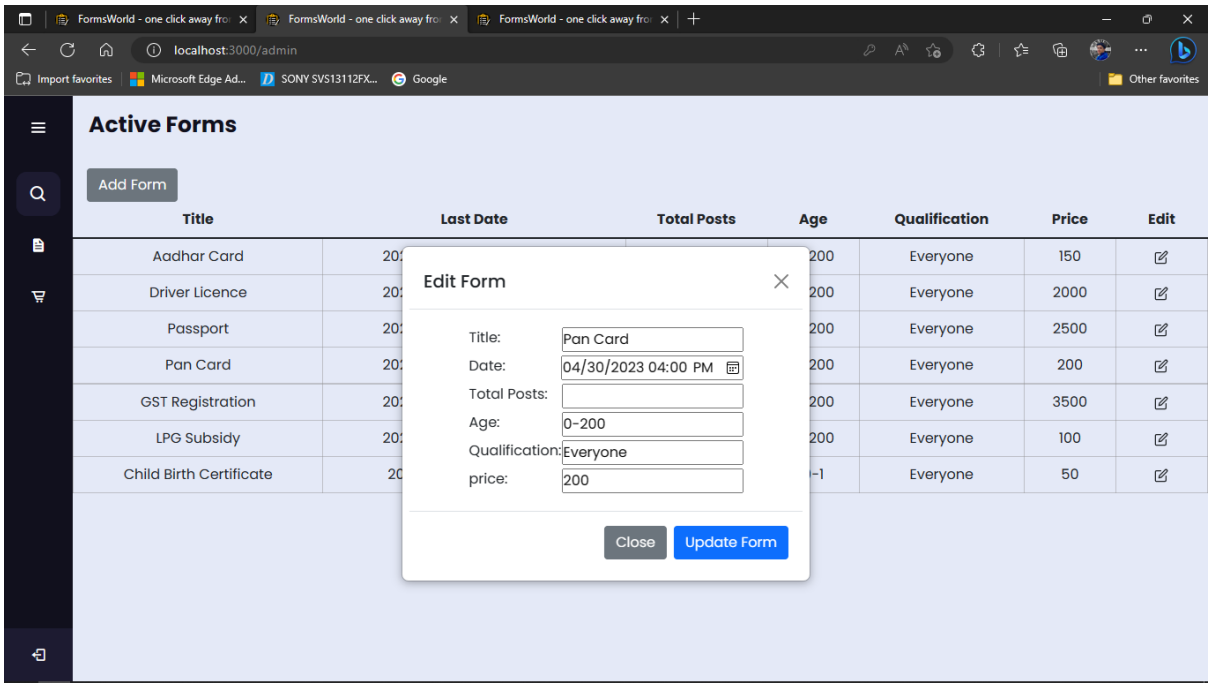


3. Loading all Forms on Page load

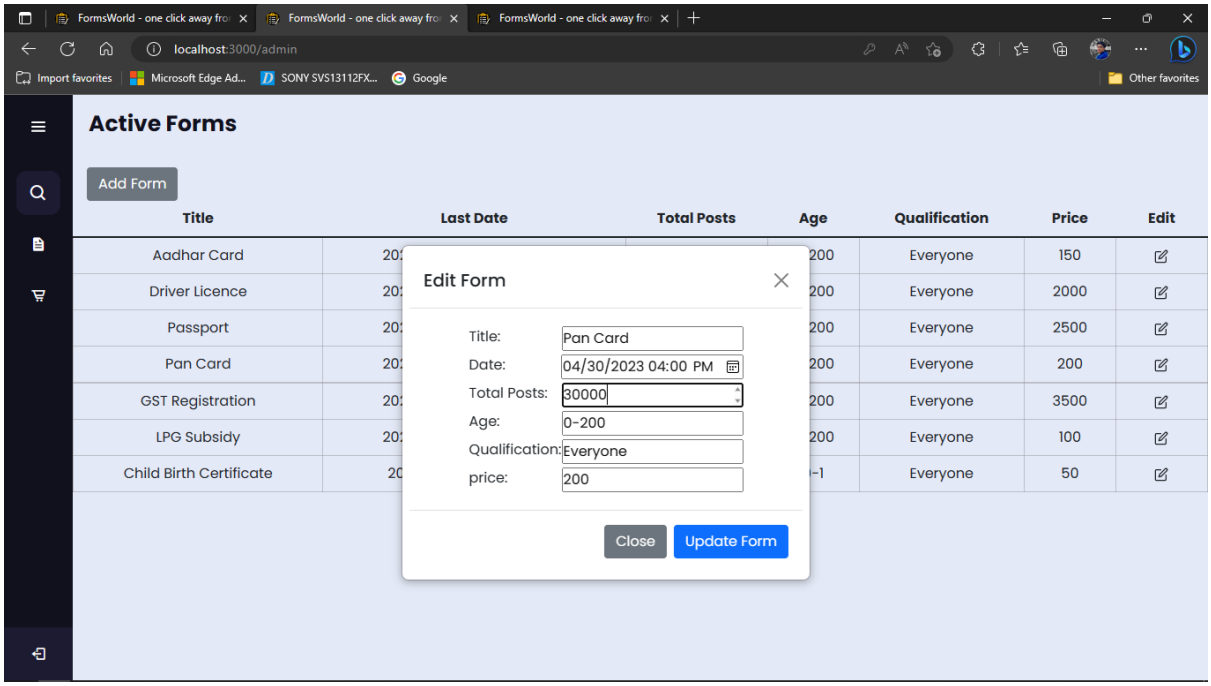


Total Post is empty, because these forms were available for everyone.

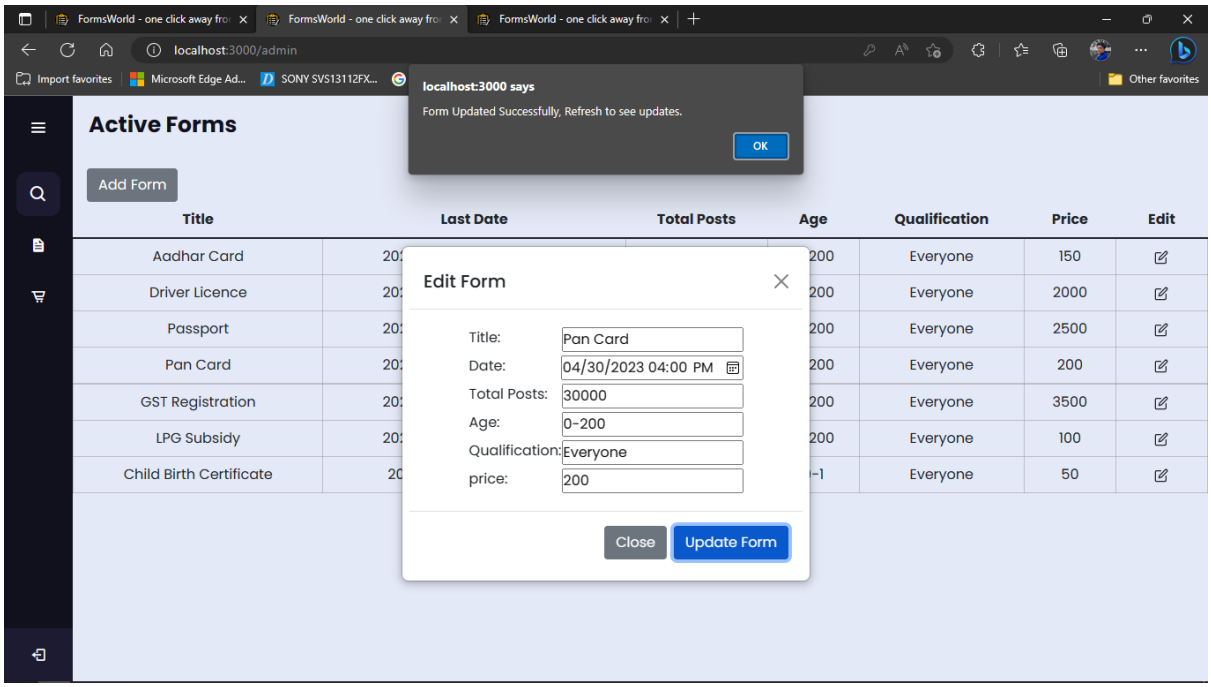
4. Click Editing Form



5. Updating form

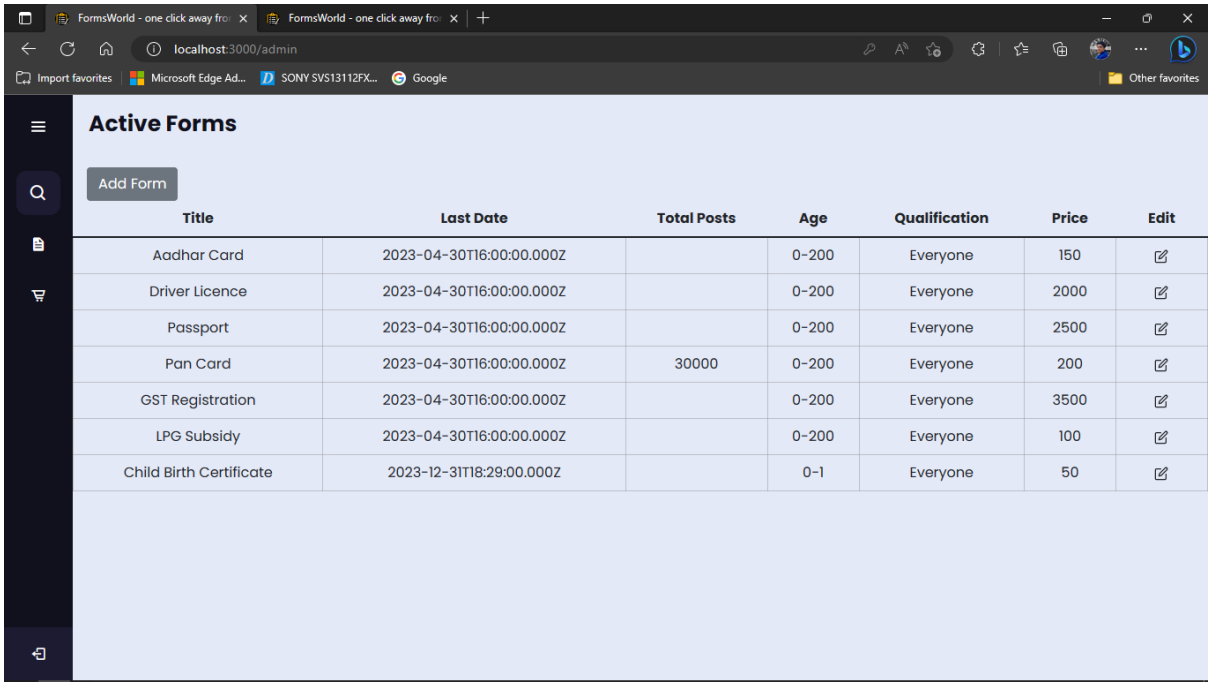


Clicking updating form:

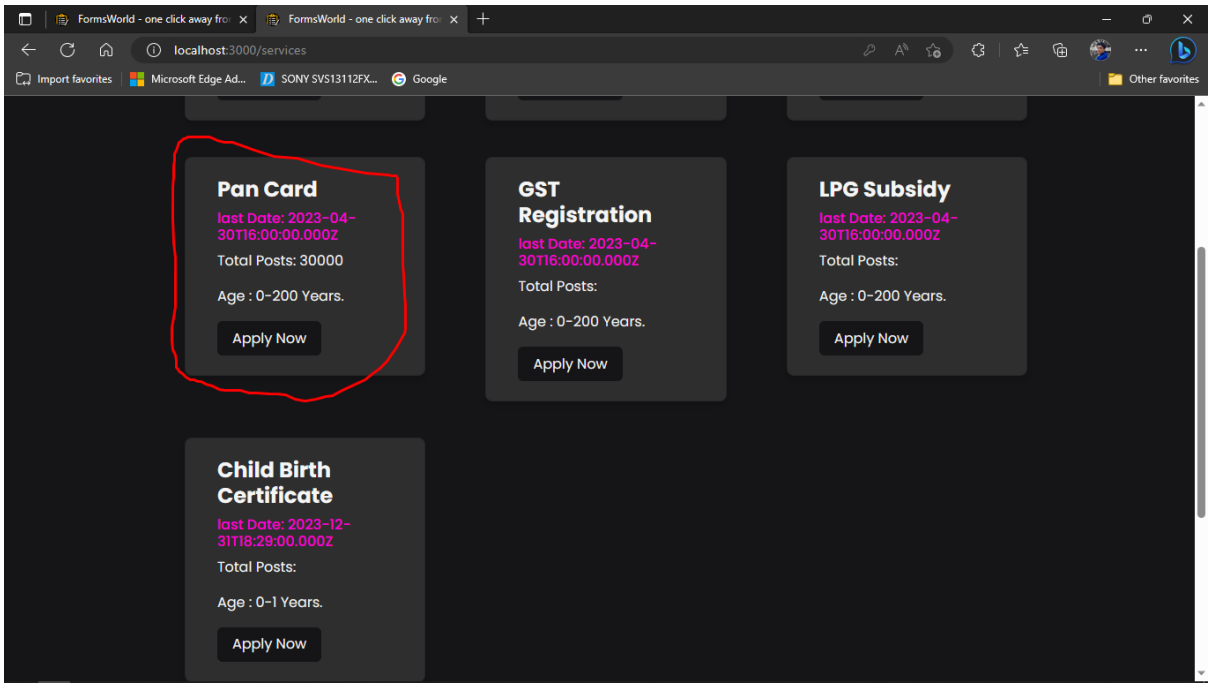


6. Viewing Updates

On admin tab:



On user Servies tab:

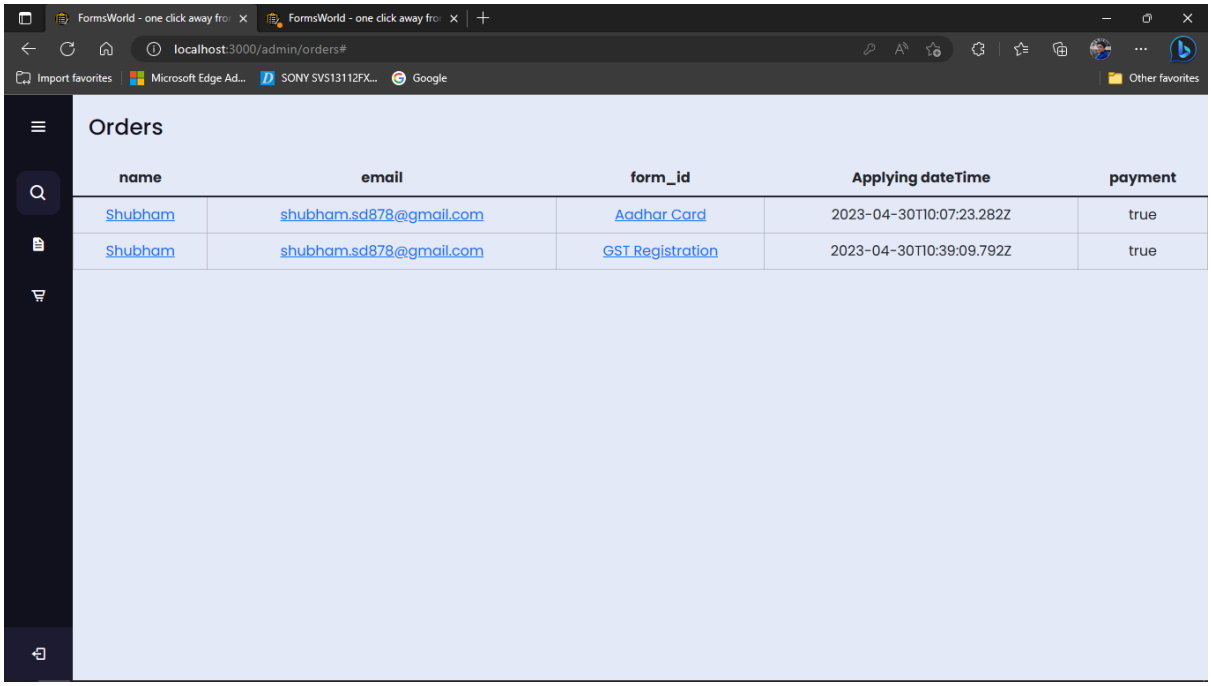


7. Admin/Orders

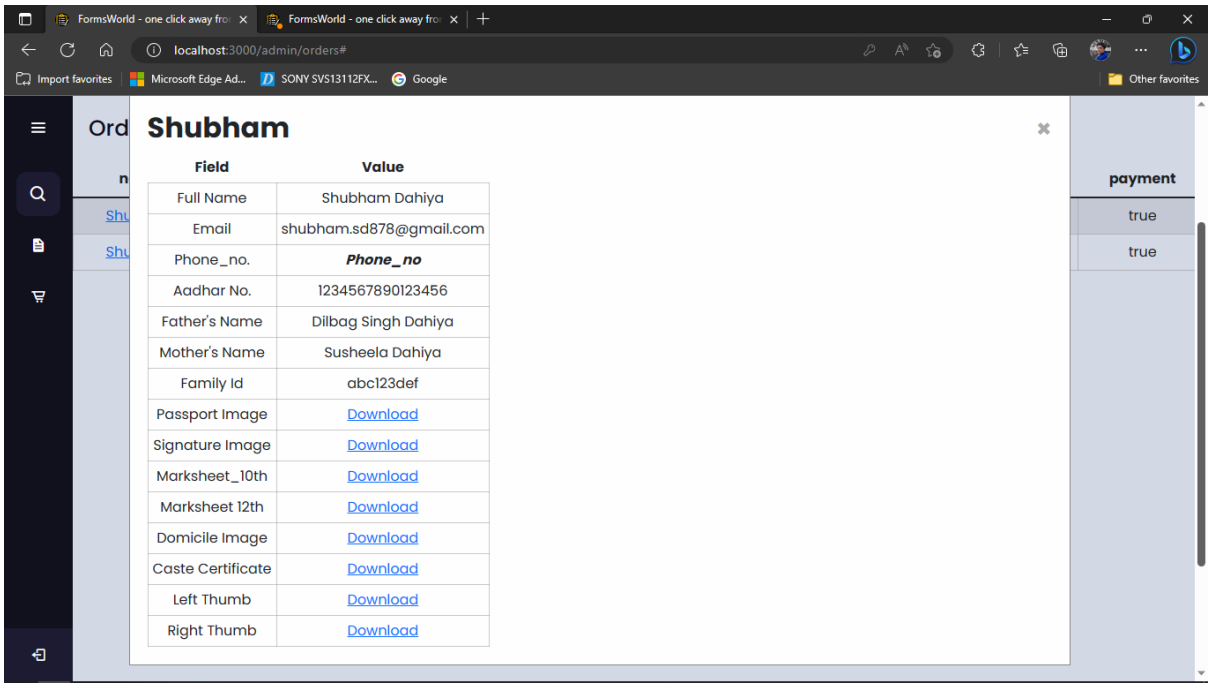
TEST	CASE	EXPECTED	RESULT
Orders	On load	Should display all the orders with payment status.	PASS
Orders	Clicking on name or email	Should popup the modal of user details to the admin.	PASS
Orders	Try to download Documents	Download document	PASS
Orders	Clicking Form Details	Should popup the modal for showing the form details to the admin.	PASS

Table 12.7 Admin/Orders Testing

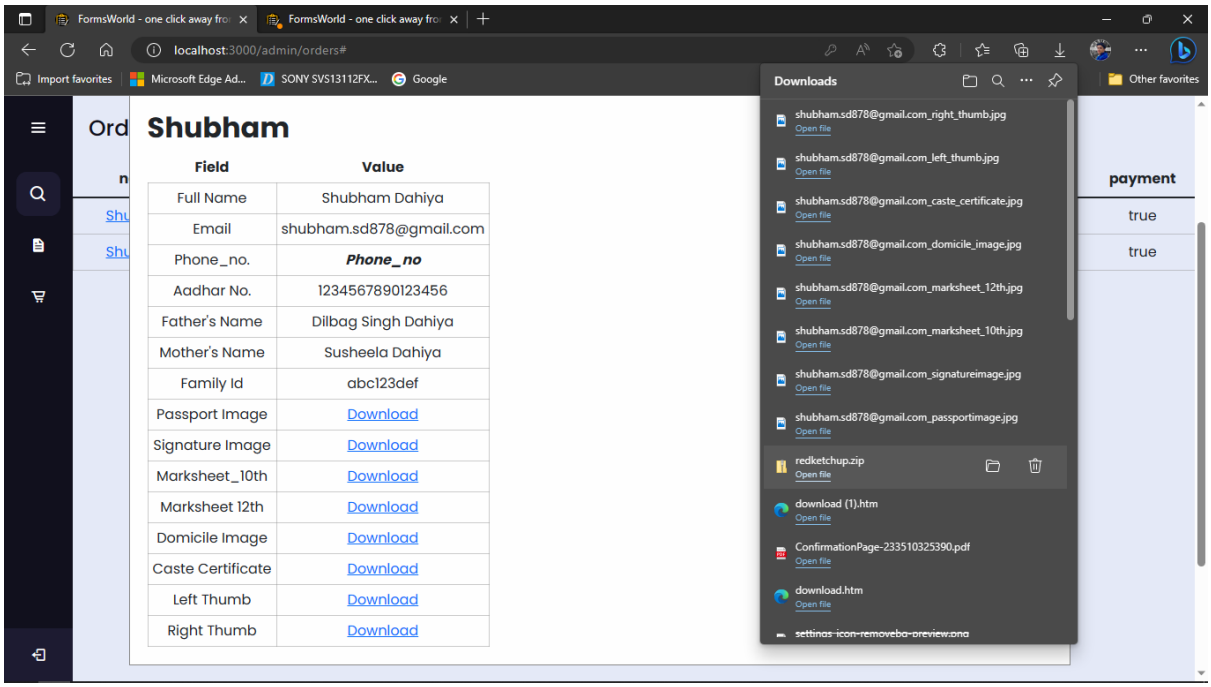
1. Fetching all Orders on load



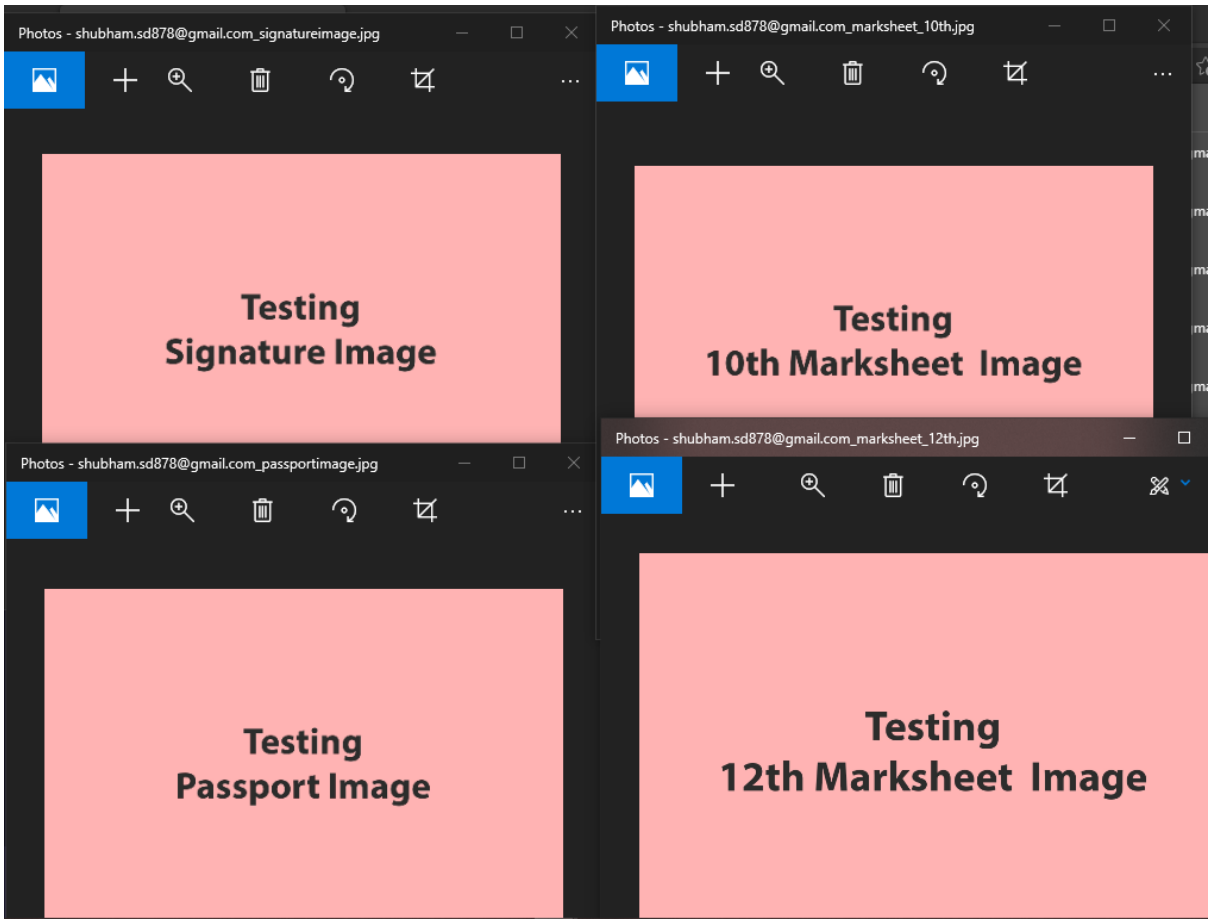
2. Clicking on name or email



3. Downloading Documents

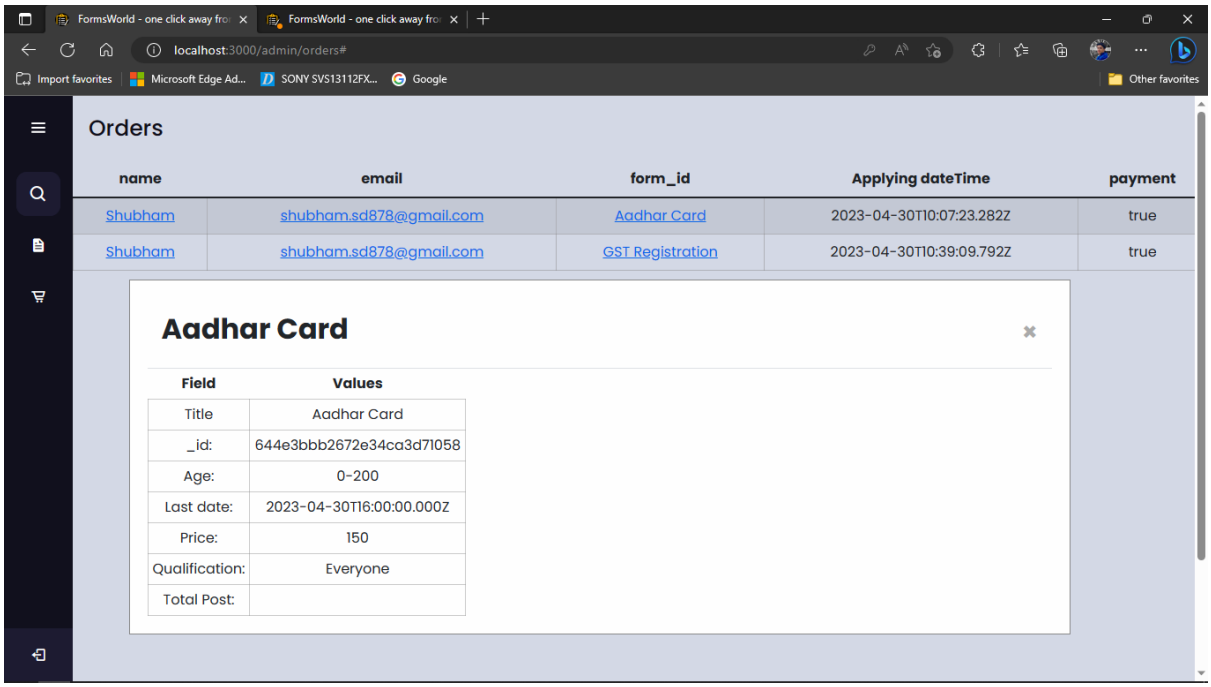


Trying to open some downloads



These were the files I uploaded while for KYC.

4. Clicking Form Details Modal



Chapter 13

System Implementation

13.1 Introduction

In today's fast-paced world, time is a valuable commodity, and people are always looking for ways to save time and increase efficiency. One such way is by outsourcing mundane and repetitive tasks to specialized service providers. The goal of this project is to provide a solution for individuals and businesses who require assistance with form filling by creating a website that offers form filling services for various jobs.

The website was developed using the MERN stack, which stands for MongoDB, Express, React, and Node.js. MongoDB was used as the database to store user information, while Express and Node.js were used to create a back-end API that handles requests and responses. React was used for the front-end, providing a user-friendly interface for users to interact with the website.

The website was deployed on Firebase, a platform that offers a range of tools and services for hosting and managing web applications. The project aims to provide a hassle-free, secure, and efficient solution to the form filling needs of its users.

In this report, we will discuss the implementation of the website, including its system architecture, front-end and back-end implementation, database implementation, deployment and hosting, performance and scalability, and security and privacy measures.

13.2 System Architecture:

The system architecture is a fundamental aspect of the website that determines its functionality and performance. The website was designed using the MERN stack, which consists of MongoDB, Express, React, and Node.js. Each component of the stack plays a critical role in the system architecture, as described below.

1. Front-End:

The front-end of the website was built using React, a popular JavaScript library used for building user interfaces. The user interface includes various components, such as forms, buttons, and input fields, that allow users to interact with the website. The front-end also handles the display of data fetched from the back-end.

2. Back-End:

The back-end of the website was built using Express and Node.js. Express is a minimalist web framework that provides a set of tools for creating APIs, while Node.js is a JavaScript runtime environment that enables server-side programming. The back-end handles requests and responses from the front-end and performs various tasks, such as data processing, validation, and authentication.

3. Database:

The database used in the system architecture is MongoDB, a NoSQL document database that provides flexibility and scalability. MongoDB stores user information, such as login credentials, personal details, and form data. The database is accessed via the back-end API and provides fast and efficient data retrieval and storage.

4. Deployment and Hosting:

The website is hosted on Firebase, a platform that provides a range of tools and services for hosting and managing web applications. Firebase provides a secure and scalable environment for the website, including features such as automatic scaling, real-time database, and hosting.

5. Third-Party Services:

The system architecture includes various third-party services, such as payment gateways, email services, and analytics tools. These services are integrated into the website to enhance its functionality and provide a better user experience.

13.3 Front-End Implementation:

The front-end of the website was built using React, a popular JavaScript library used for building user interfaces. React provides a flexible and modular structure for designing user interfaces, which makes it easy to create complex and dynamic web applications. The front-end of the website includes the following components:

1. User Interface Design:

The user interface design of the website was based on a modern and minimalist approach, with a focus on ease of use and simplicity. The website features a clean and intuitive interface, with prominent call-to-action buttons, clear instructions, and well-designed forms.

2. Forms:

Forms are a critical component of the website, as they allow users to input their information and receive customized form filling services. The website includes

various types of forms, such as personal information forms, job application forms, and legal document forms. The forms are designed using React components, which provide a modular and reusable approach to form design.

3. Input Validation:

Input validation is an essential part of the front-end implementation, as it ensures that user input is correct and conforms to specific standards. The website includes various input validation techniques, such as regular expressions, conditional rendering, and error messages.

4. Data Display:

The front-end is responsible for displaying data fetched from the back-end. The website includes various components for displaying data, such as tables, charts, and graphs. The data display components are designed using React components and are customizable and reusable.

5. Navigation:

Navigation is an essential aspect of the website, as it allows users to move between different pages and sections. The website includes a navigation bar that provides links to different pages, such as the home page, services page, and about page. The navigation bar is designed using React components and is customizable.

13.4 Backend Implementation:

The back-end of the website was built using Express and Node.js. The back-end provides a range of functionalities that support the website's form filling services. The back-end implementation includes the following components:

1. API Design:

The API design is a critical part of the back-end implementation, as it defines the endpoints and methods used to communicate with the front-end. The website includes various APIs for handling user authentication, data retrieval, and form submission. The APIs are designed using Express and are RESTful, which ensures that they are scalable and maintainable.

2. Authentication and Authorization:

Authentication and authorization are essential aspects of the website, as they ensure that user data is secure and protected. The website includes various authentication and authorization techniques, such as password hashing, token-based authentication, and role-based access control.

3. Database Integration:

The back-end integrates with MongoDB, a NoSQL document database that provides flexibility and scalability. The back-end uses Mongoose, a popular object data modeling (ODM) library for MongoDB, to interact with the database. The back-end stores and retrieves user data, such as login credentials, personal details, and form data, using the database.

4. Form Submission and Processing:

Form submission and processing are the core functionalities of the website, as they provide customized form filling services to users. The back-end includes

various modules for processing form data, such as data validation, formatting, and storage. The back-end uses middleware functions to handle form submission and processing, which ensures that the data is processed efficiently and accurately.

5. Third-Party Integration:

The back-end integrates with various third-party services, such as payment gateways to enhance the website's functionality. The back-end uses APIs provided by these services to interact with them and to provide a seamless user experience.

13.5 Database Implementation:

The database implementation of the website was built using MongoDB, a NoSQL document database. MongoDB provides flexibility, scalability, and performance, which makes it an excellent choice for web applications. The database implementation includes the following components:

1. Data Modeling:

Data modeling is a critical part of the database implementation, as it defines the structure and relationships between the data. The website uses Mongoose, a popular object data modeling (ODM) library for MongoDB, to define the data models. The data models include user profiles, form data, and other related data.

2. Data Storage:

The website stores various types of data in the database, such as user profiles, form data, and session data. The data is stored in collections, which are similar

to tables in a relational database. The database implementation uses the CRUD (create, read, update, delete) operations to interact with the data, which ensures that the data is managed efficiently and accurately.

3. Data Retrieval:

Data retrieval is a crucial aspect of the database implementation, as it enables the website to fetch and display user data. The website includes various APIs for retrieving user data, such as form data and user profiles. The database implementation uses queries to retrieve the data, which ensures that the data is retrieved accurately and efficiently.

4. Indexing:

Indexing is a technique used to optimize database performance by creating indexes on the database fields. The website includes various indexes, such as indexes on the user ID field and the form submission date field. The indexing ensures that the database queries are executed efficiently, which enhances the website's performance.

5. Security:

Security is an essential aspect of the database implementation, as it ensures that the data is protected and secure. The website includes various security measures, such as data encryption and access control. The database implementation uses MongoDB's built-in security features, such as authentication and authorization, to ensure that the data is secure and protected.

13.6 Deployment and Hosting:

The website was deployed and hosted on Firebase, a platform that provides various services for building, testing, and deploying web applications. The deployment and hosting implementation includes the following components:

1. Deployment:

The website was deployed on Firebase using the Firebase CLI (Command-Line Interface), which is a set of tools for managing Firebase projects. The deployment process involved the following steps:

- Creating a Firebase project and configuring the project settings
- Installing the Firebase CLI on the development machine
- Creating a build of the website using the npm build command
- Deploying the build to Firebase using the Firebase deploy command

2. Hosting:

The website was hosted on Firebase Hosting, which is a service that provides fast and secure hosting for web applications. The hosting implementation includes the following features:

- SSL/TLS encryption for secure communication between the client and server
- Content Delivery Network (CDN) for faster content delivery
- Custom domain support for branding and SEO.

Chapter 14

Conclusion

After successfully building and deploying "Forms World" using the MERN stack and Firebase, it can be concluded that the project has achieved its objectives of providing a platform for form filling services for various jobs.

The website offers a user-friendly interface that allows users to easily access and fill out the required forms. The MERN stack provided a robust and efficient framework for developing the website, while Firebase enabled seamless deployment and hosting. This combination of technologies helped in creating a responsive and interactive website that could handle complex applications.

The project faced various challenges, including ensuring data security and implementing reliable payment options. These challenges were addressed through careful planning and implementation of appropriate security measures and payment gateways.

Overall, the "Forms World" website has been designed to cater to the needs of users seeking form filling services, and the use of modern technologies has resulted in a highly functional and efficient website. Future improvements could include additional features such as integration with various job portals and the inclusion of more form templates to increase the website's usability.

In conclusion, the "Forms World" project has been a success, and it has the potential to become a valuable resource for job seekers and employers alike.

Chapter 15

Future Enhancements

Technology is changing every day. Everyday a new system comes out that makes other things outdated. It is not possible to develop an system that can be fulfill all users requirements.

We will upgrade “Forms World” to provide reach user experience:

- We will implement status seeing and feature on both user and admin side.
- We will also add the ability to fetch the category of user (like General, SC, BC, ST) and show them the price according them.
- We will add infinite scroll in services tab.
- We will improve the user experience.
- We will upgrade our project to mobile application platform.
- We will add functionality to notifications ability too.