MOTOSACU ADMIN

A Project Report Submitted In Partial Fulfillment of the Requirements for The Degree of

MASTER OF COMPUTER APPLICATION

by

RONIKA GUPTA
(Roll no.: 1900290149085)

Under the supervision of

Mr. ANKIT SHARMA

KIET GROUP OF INSTITUTIONS, GHAZIABAD

to the

DEPARTMENT OF COMPUTER APPLICATIONS

DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY
LUCKNOW
(Formerly Uttar Pradesh Technical University, Lucknow)

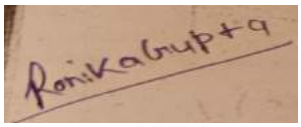
MAY, 2021

DECLARATION

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to substantial extent has been accepted for the award of any degree or diploma of the university or another institute of higher learning except where due acknowledgment has been made in the text.

Name: Ronika Gupta

ROLL NO.: 1900290149085

Signature: 

CERTIFICATE

Certified that **Ronika Gupta (Univ Roll - 1900290149085)** have supported the project work having " **MOTOSACU ADMIN**" for the award of **Master of Computer Applications** from Dr. APJ Abdul Kalam Technical University, Lucknow under my supervision. The thesis embodies results of original work, and studies are carried out by the student himself and the contents of the thesis do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Ms. Ankit Sharma
Assistant Professor
Department of Computer Applications
KIET Group of Institutions, Ghaziabad

External Examiner

Dr. Ajay Kumar Srivastava
Prof. & Head of Department
KIET Group of Institutions,
Ghaziabad

To Whom It May Concern

ZIMOZI
AI | MOBILE APPS | ENTERPRISE SOLUTIONS
Office No. - 727, The Ithum Tower, Sec-62, Noida (U.P), Postal Code - 201301

Date: 08th July 2021

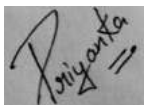
This is to certify that Ronika Gupta a student of KIET group of Institutions successfully completed her training period from 05th January 2021 to 05th July 2021 with reference to the partial fulfillment of the requirements of the MCA of Dr. APJ Abdul Kalam Technical University.

All the necessary guidance and hands on experience were provided by Zimozi for the establishment of this Training.

We wish her the very best in all her future endeavors.

For Zimozi Solutions Pvt. Ltd.

HR Head



Priyanka Bijolia

+91-120-4158818 | hello@zimozi.co | www.zimozi.co

MOTOSACU ADMIN

RONIKA GUPTA

ABSTRACT

The “Motosacu Admin” is a React.js based Admin page designed for management of Consumer and Provider that are register by the Motosacu App . In which we can create new Consumer and Provider also update the details of the Consumer and Provider. We can see all the requests and request created by Consumer and Provider list also. In this admin page we show all the data of the Motosacu app and helps to manage the details of the Consumer and Provider. It makes easy to managing the large amount of data, It contains filters to find the Consumer and Provider in quick time and very easily.

ACKNOWLEDGEMENTS

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I am highly indebted to Ms. Sathi Mandal for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

Words are not enough to express my gratitude to Dr. Ajay Kumar Shrivastava, Professor and Head, Department of Computer Applications, for his insightful comments and administrative help at various occasions.

Ronika Gupta

1900290149085

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LITERATURE REVIEW

1.1 Start Programming Using HTML, CSS, and JavaScript

A Beginner's Guide to Computer Programming Start Using HTML, CSS, and JavaScript is a manual for undergraduate student in engineering and the natural sciences to discover how computer programming work. Using a dialog format between two students and professor , the next teaches student how the mainstream web languages HTML, CSS, And JavaScript.

1.2 Introduction to Web Interaction Design

This book introduces standard and new HTML5 elements and attributes and CSS3 properties commonly used in Web design as well as design guidelines for their effective use. Its approach of explaining every line of code in the examples it uses to show the usage of the HTML elements and CSS properties introduced makes it an invaluable Web design resource for beginners as well as intermediates looking to fill in gaps in their knowledge. In addition, the inclusion of user-centered design process stages and how they are best managed in website development makes the book unique in its area. Also, the book's approach of including challenges after each topic to help refresh readers' knowledge, as well as make them think, ensures that there are ample activities to keep learners motivated and engaged.

1.3 Dynamic Web Programming and HTML5

With organizations and individuals increasingly dependent on the Web, the need for competent, well-trained Web developers and maintainers is growing. Helping readers master Web development, Dynamic Web Programming and HTML5 covers specific Web programming languages, APIs, and coding techniques and provides an in-depth understanding of the underlyin.

1.4 JavaScript for Sound Artists

Learn how to program JavaScript while creating interactive audio applications with **JavaScript for Sound Artists: Learn to Code With the Web Audio API**! William Turner and Steve Leonard showcase the basics of JavaScript language programing so that readers can learn how to build browser based audio applications, such as music synthesizers and drum machines. **T**he companion website offers further opportunity for growth. Web Audio API instruction includes oscillators, audio file loading and playback, basic audio manipulation, panning and time. This book encompasses all of the basic features of JavaScript with aspects of the Web Audio API to heighten the capability of any browser.

Key Features

- Uses the readers existing knowledge of audio technology to facilitate learning how to program using JavaScript. The teaching will be done through a series of annotated examples and explanations.
- Downloadable code examples and links to additional reference material included on the books companion website.
- This book makes learning programming more approachable to nonprofessional programmers
- The context of teaching JavaScript for the creative audio community in this manner does not exist anywhere else in the market and uses example-based teaching

CHAPTER 1

INTRODUCTION

The “Motosacu Admin” is a React.Js based Admin page designed for management of Consumer and Provider that are register by the Motosacu App. In which we can create new Consumer and Provider also update the details of the Consumer and Provider

1.1 MOTOSACU ADMIN

The “Motosacu Admin” is a React.Js based Admin page designed for management of Consumer and Provider that are register by the Motosacu App . In which we can create new consumer and provider also update the details of the consumer and provider. We can see all the requests and request created by consumer and provider list also. In this admin page we show all the data of the Motosacu app and helps to manage the details of the consumer and provider.

In this Project we provide a login screen for Admin, admin can enter in the admin page by using their username and password. Then it will show all the details like number of users and donors, number of requests, we can also check the individual details and perform many operations like add the Consumer, Provider, edit/update consumer, Provider etc. It helps more and easily to make such operations.

1.2 Project Scope

- Manage the records.
- Get all Consumer ❍ Search Consumer
- Get all requests
- Showing individual Provider details and requests
- Performing operations on records like Add/Update request.

1.3 Hardware and Software used in Project

Table 1.1 Hardware

`

Hardware	Configuration
Processor	Intel core i3-6006U @ 2.0 GHz
RAM	8GB DDR4
Monitor	Dell Built-in display
Hard Disk	1 TB

Keyboard	Dell Built-in keyboard

Table 1.2 Software

Software	Configuration
Operating System	Windows 10
Language	React.js & Web Technologies
IDE	Visual Studio Code

1.4 Technologies Description

1.4.1 React.js:

React (pronounced /vju:/, like view) is a progressive framework for building user interfaces. The core library is focused on the view layer only, and is easy to pick up and integrate with other libraries or existing projects.

On the other hand, React is also perfectly capable of powering sophisticated Single-Page Applications Progressive framework is a framework that you can insert into your project as you feel the Motosacu for it. Differently of other JavaScript framework like Angular, that since the beginning, you Motosacu a full project make in Angular, follow the “Angular rules”.

This implies you Motosacu to learn a lot of things to start programming with Angular. React is more simple and flexible. React allows you make just specific parts of your application. You learn just what is necessary for the problem you are dealing with. Or if it is necessary and you have time, you can learn more and make a full complex front-end application 100% React.

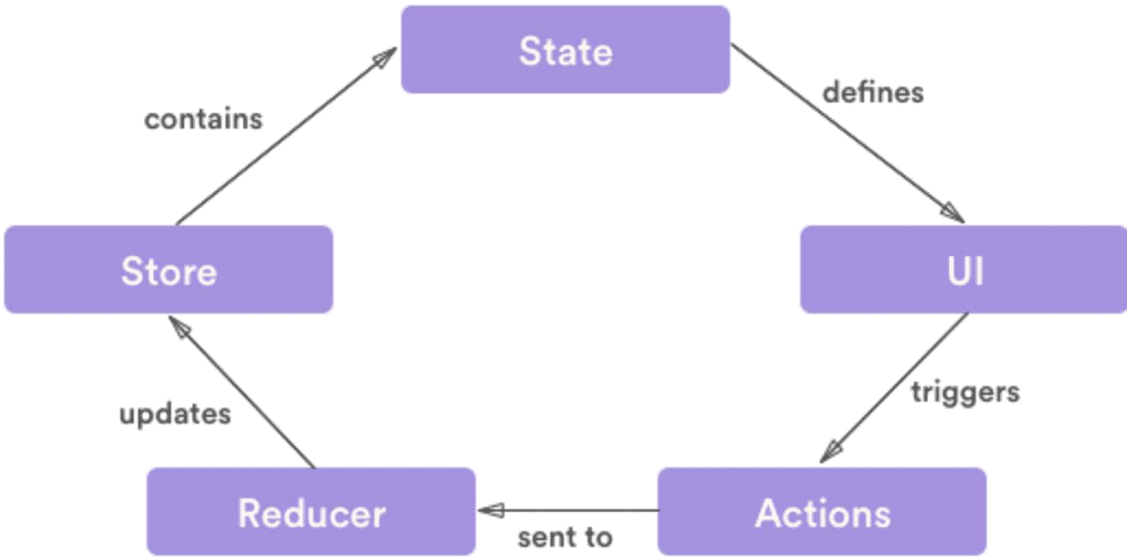


Fig 1.1State Management for React.js

Features

Following are the features available with ReactJS.

Virtual DOM:

React.JS makes the use of virtual DOM, which is also used by other frameworks such as React, Ember, etc. The changes are not made to the DOM, instead a replica of the DOM is created which is present in the form of JavaScript data structures. Whenever any changes are to be made, they are made to the JavaScript data structures and the latter is compared with the original data structure. The final changes are then updated to the real DOM, which the user will see changing. This is good in terms of optimization, it is less expensive and the changes can be made at a faster rate.

Data Binding:

The data binding feature helps manipulate or assign values to HTML attributes, change the style, assign classes with the help of binding directive called **bind** available with React.JS.

Components:

Components are one of the important features of React.JS that helps create custom elements, which can be reused in HTML.

Event Handling:

is the attribute added to the DOM elements to listen to the events in react.JS.

Animation/Transition:

ReactJS provides various ways to apply transition to HTML elements when they are added/updated or removed from the DOM. React.JS has a built-in transition component that Motosacus to be wrapped around the element for transition effect. We can easily add third party animation libraries and also add more interactivity to the interface.

Computed Properties:

This is one of the important features of React.JS. It helps to listen to the changes made to the UI elements and performs the necessary calculations. There is no Motosacu of additional coding for this.

Templates:

React.JS provides HTML-based templates that bind the DOM with the React instance data. React compiles the templates into virtual DOM Render functions. We can make use of the template of the render functions and to do so we have to replace the template with the render function.

Directives:
React.js has built-in directives such as , which are used to perform various actions on the frontend.

Watchers:
Watchers are applied to data that changes. For example, form input elements. Here, we don't have to add any additional events. Watcher takes care of handling any data changes making the code simple and fast.

Routing:
Navigation between pages is performed with the help of react-router.

Lightweight:
React.JS script is very lightweight and the performance is also very fast.

React-:
ReactJS can be installed at the command line using the React-cli command line interface. It helps to build and compile the project easily using React-cli.

1.4.2 HTML:

Hypertext Markup Language (**HTML**) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by *tags*, written using angle brackets. Tags such as <**img** /> and <**input** />directly introduce content into the page. Other tags such as <**p**> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.
"The core to all web design is Hypertext Markup Language (HTML), the code that sits behind every web page and allows users to create stunning web sites. Today's web sites can do amazing things. Can you imagine not being able to use solutions such as Google's Gmail, Microsoft's Bing, or view content on YouTube? Web sites have moved from static pages to complex applications. The core HTML language requires more and more functionality to meet our Motosacu. To this end, a new standard has been introduced-HMTL5." [1]

1.4.3 CSS:

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.
CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device. "Tags are used in HTML5 to place and organize content at a level that is descriptive. This does not mean that the page will look good. Presentation of content on the page is controlled using Cascading Style Sheets Level 3, or CSS3, in HTML5." [3]

1.4.4 JavaScript:

JavaScript often abbreviated as **JS**, is a programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototypebased object-orientation, and first-class functions.

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it for client-side page behavior, and all major web browsers have a dedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM). However, the language itself does not include any input/output (I/O), such as networking, storage, or graphics facilities, as the host environment (usually a web browser) provides those APIs.

Other technologies that also played important role in development of this project are Bootstrap, jQuery etc.

1.4.5 MongoDB:

MongoDB is a document database designed for ease of development and scaling. The Manual introduces key concepts in MongoDB, presents the query language, and provides operational and administrative considerations and procedures as well as a comprehensive reference section.

MongoDB offers both a *Community* and an *Enterprise* version of the database:

- MongoDB Community is the source available and free to use edition of MongoDB.
- MongoDB Enterprise is available as part of the MongoDB Enterprise Advanced subscription and includes comprehensive support for your MongoDB deployment. MongoDB Enterprise also adds enterprise-focused features such as LDAP and Kerberos support, on-disk encryption, and auditing.

1.5 Admin Page Modules

1.5.1 Login:

This is **login** screen which is use to validate the login user is authorized or not. In this page we are using firebase authorization to validate the User.

.

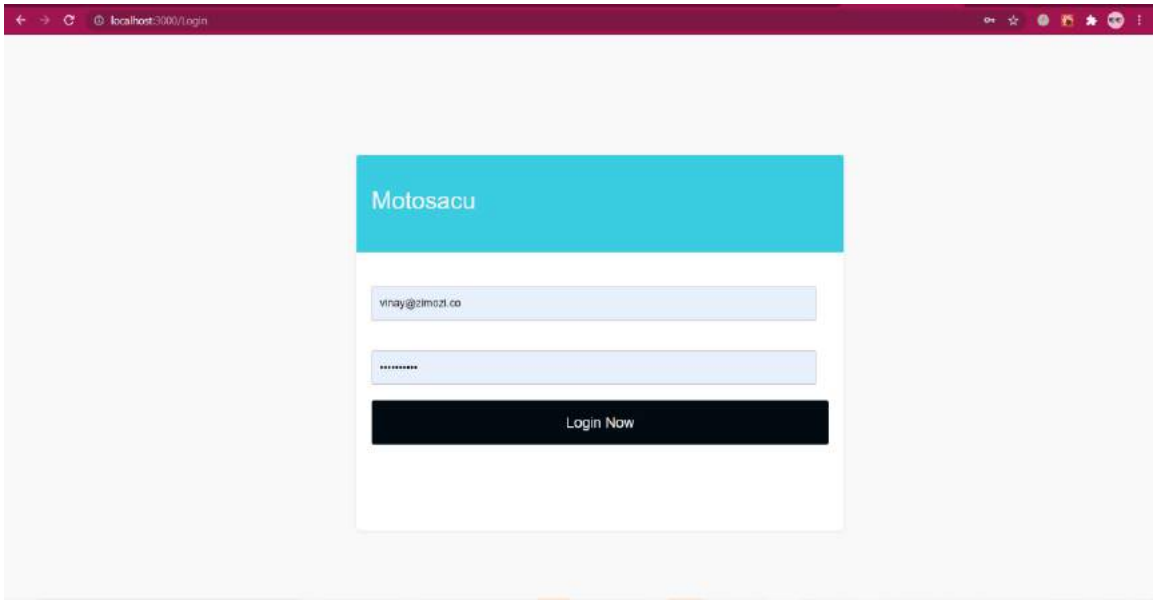


Fig 1.2 Login Page

1.5.2 Consumer Module:

This is another module of Motosacu Admin. In consumer module we can see all the records of consumer, we can see total number of users, individual consumer details and requests. Here we can also perform such operation like add new consumer, update the existing consumer details, and see more information about perticular consumer .

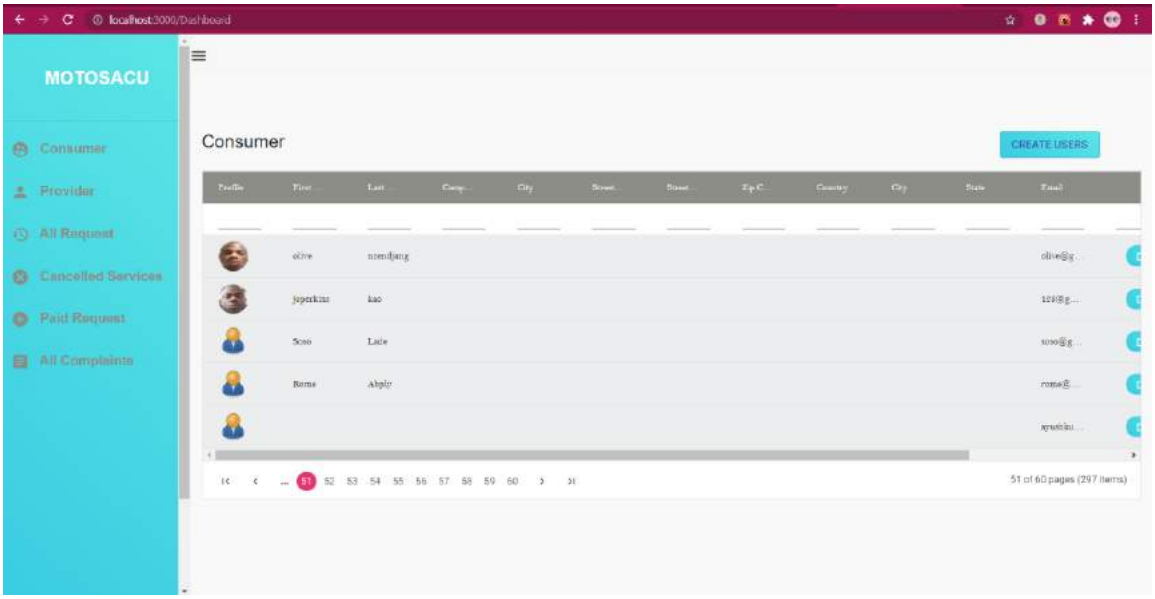


Fig 1.3 Consumer

1.5.3 Details of Consumer:

Here we can see all the Details (details related to request) created by the Consumer.

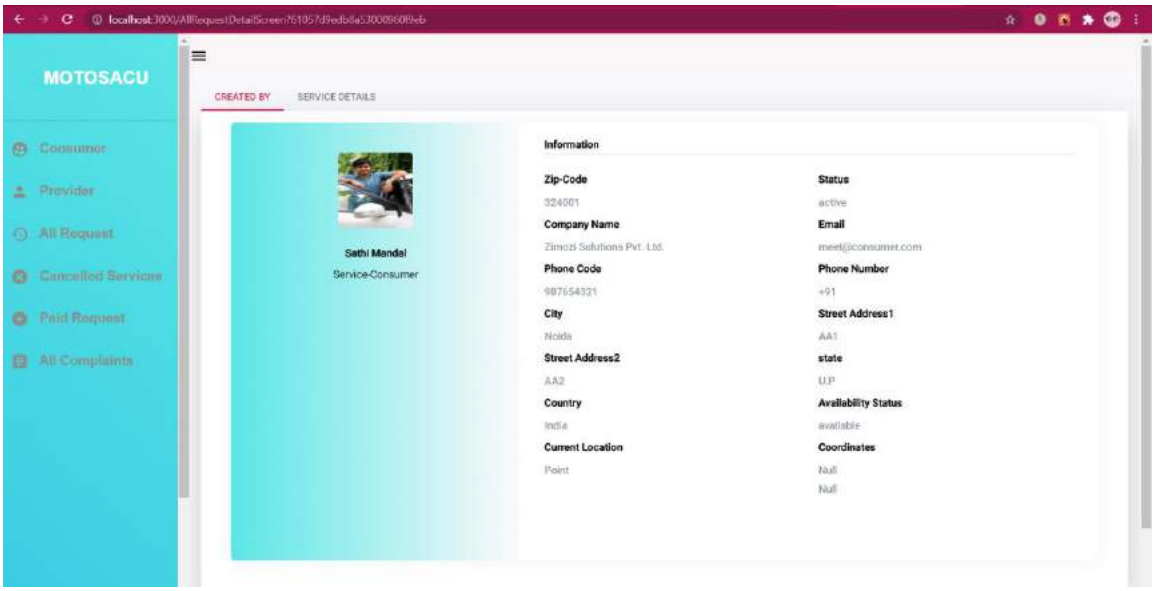


Fig 1.4 Details of Consumer

1.5.3 Create User:

Here we can create users here we have used post API to login the user.

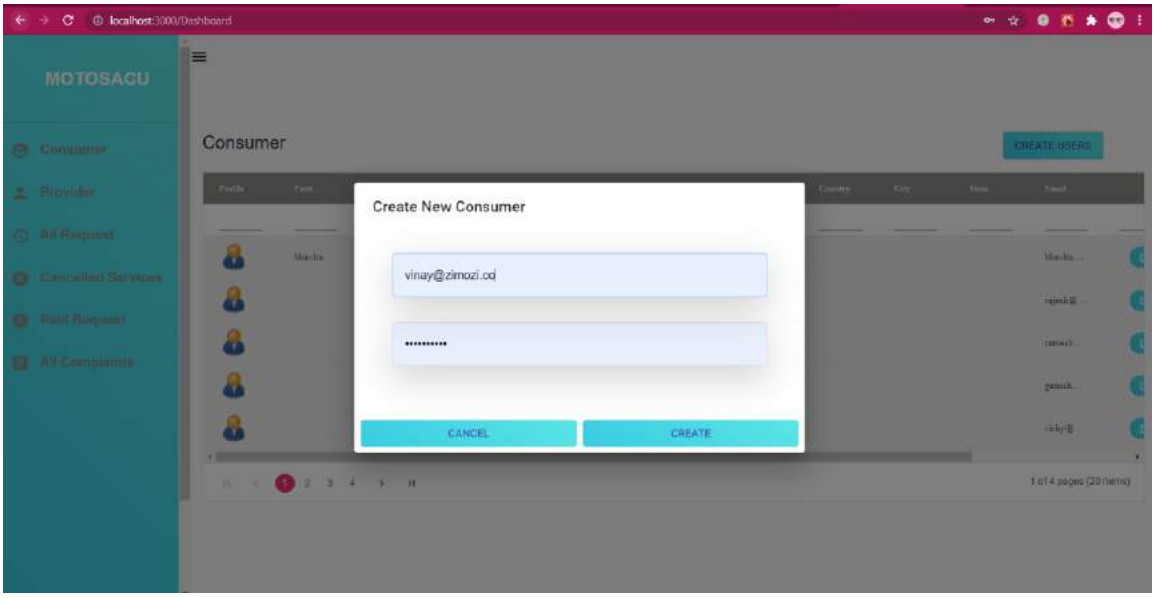


Fig 1.5 Create Consumer

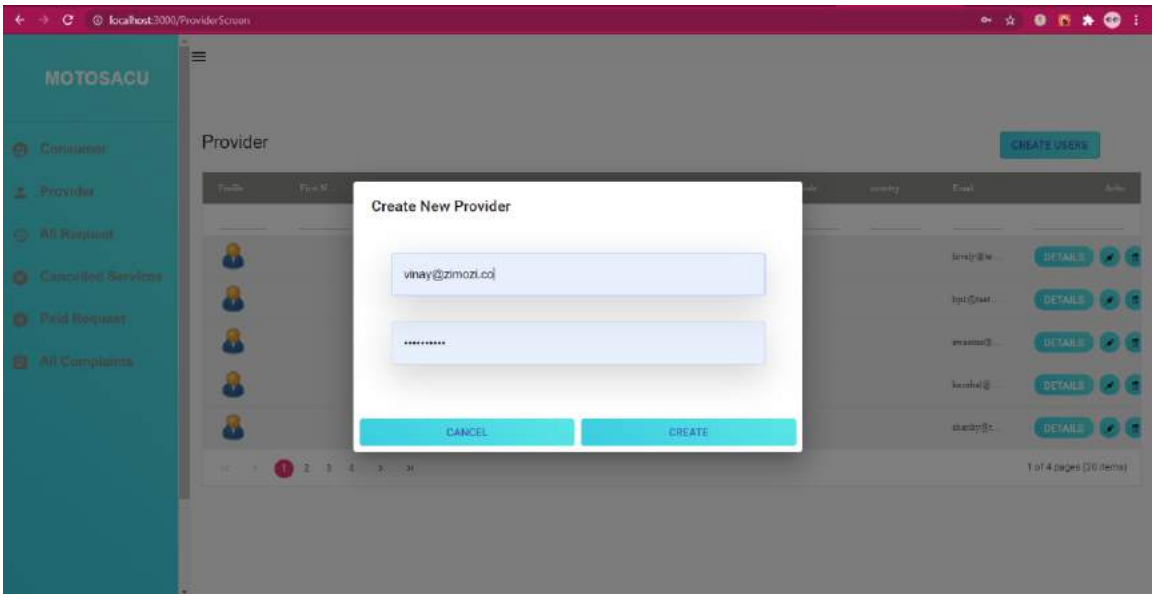


Fig 1.6 Create Provider

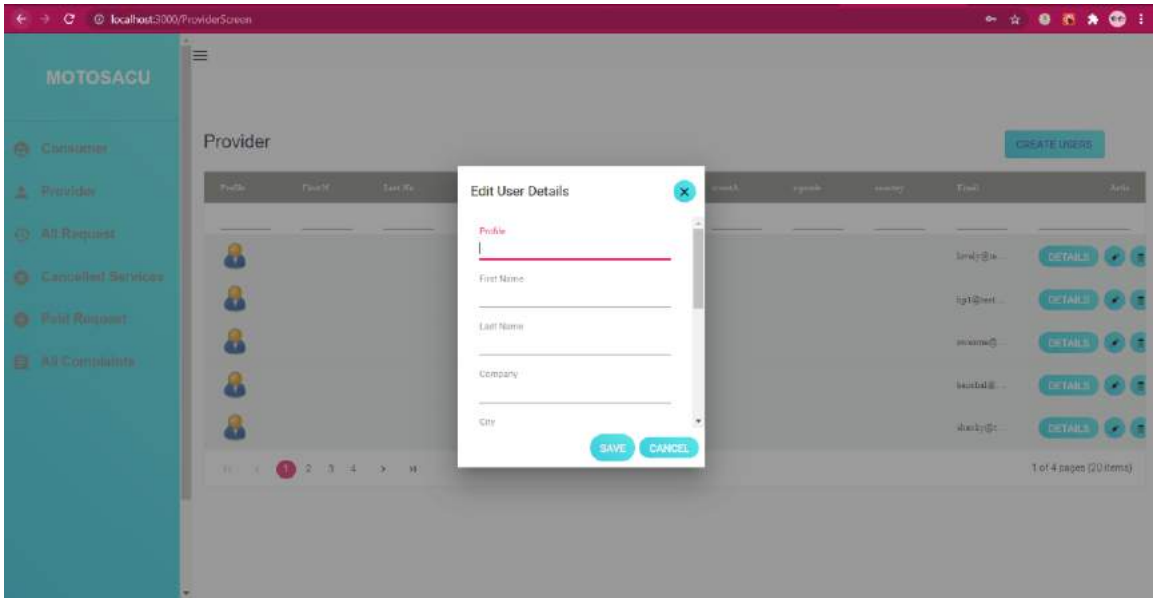


Fig 1.7 Edit User

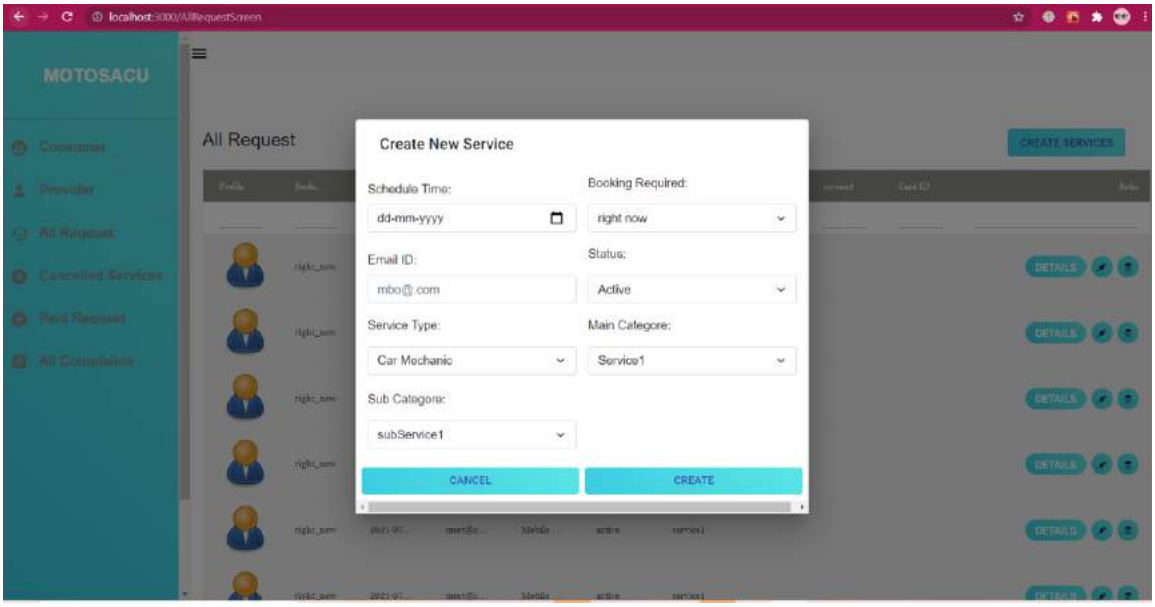


Fig 1.7 Create New Services

CHAPTER 2

FEASIBILITY STUDY

A **feasibility study** is an assessment of the practicality of a proposed project or system. A feasibility study aims to objectively and rationally uncover the strengths and weaknesses of an existing business or proposed venture, opportunities and threats present in the natural environment, the resources required to carry through, and ultimately the prospects for success. In its simplest terms, the two criteria to judge feasibility are cost required and value to be attained.

"This book presents a set of tools that will aid in deciding whether a project should go ahead, be improved, or abandoned altogether by pinpointing its vulnerabilities. It offers a review of project feasibility analysis, and more critically, psychodynamic aspects that are often neglected, including how stakeholders interact. It provides a complement to the common techniques used for analyzing technical, financial, and marketing feasibility. The goal is to identify "hidden truths" and eliminate those gray areas that jeopardize the success of a given project. The focus is on uncovering points of vulnerabilities in four key aspects of a project: People, Power,

Processes, and Plan." [4]

2.1 . Technical Feasibility

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed.
Technical issues raised during the investigation are:

- Does the existing technology sufficient for the suggested one?
- Can the system expand if developed?

The project should be developed such that the necessary functions and performance are achieved within the constraints. The project is developed within latest technology. Through the technology may become obsolete after some period of time, due to the fact that never version of same software supports older versions, the system may still be used. So, there are minimal constraints involved with this project. The system has been developed using Java the project is technically feasible for development.

2.2 Operational Feasibility

Proposed projects are beneficial only if they can be turned out into information system. That will meet the organization's operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following: Ø Is there sufficient support for the management from the users? Will the system be used and work properly if it is being developed and implemented? Will there be any resistance from the user that will undermine the possible application benefits? This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration. So, there is no question of resistance from the users that can undermine the possible application benefits.

2.3 Economical Feasibility

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.

The benefits in the form of reduced costs or fewer costly errors. Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also, all the resources are already available, it gives an indication of the system is economically possible for development.

CHAPTER 3

DESIGN

The design phase of software development deals with transforming the customer requirements as described in the SRS documents into a form implementable using a programming language.

The software design process can be divided into the following three levels of phases of design:

1. Interface Design

2. Architectural Design
3. Detailed Design

3.1 Interface Design:

Interface design is the specification of the interaction between a system and its environment. this phase proceeds at a high level of abstraction with respect to the inner workings of the system that is during interface design, the internal of the systems are completely ignored and the system is treated as a black box. Attention is focused on the dialogue between the target system and the users, devices, and other systems with which it interacts. The design problem statement produced during the problem analysis step should identify the people, other systems, and devices which are collectively called agents.

3.2 Architectural Design:

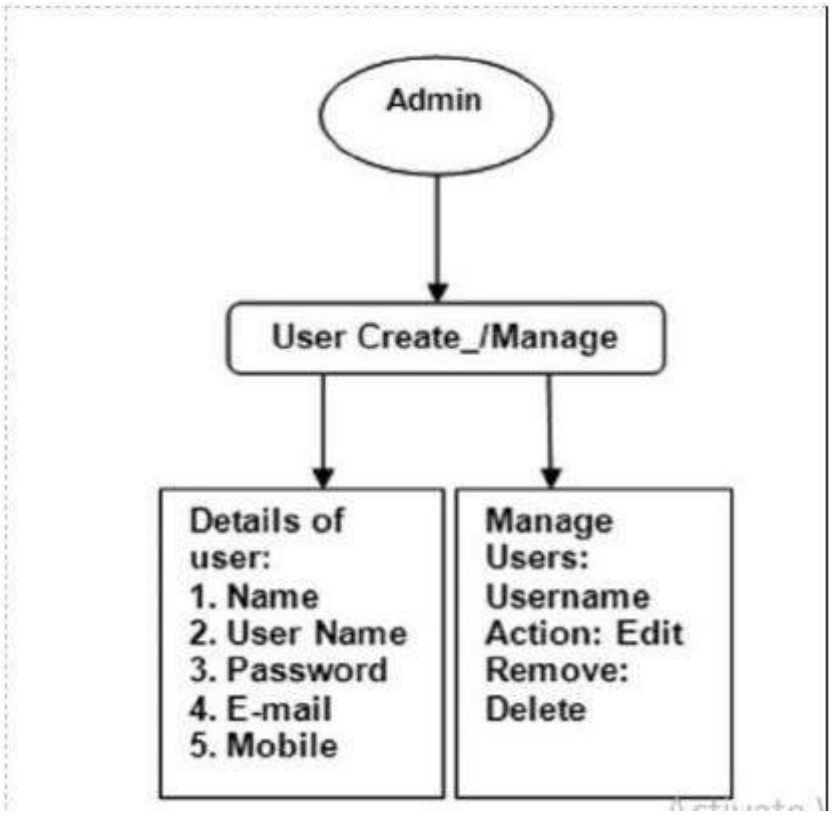
Architectural design is the specification of the major components of a system, the responsibilities, properties, interfaces and the relationships and interactions between them. In architectural design comma the overall structure of the system is chosen, but the internal details of major components are ignored.

3.3 Detailed Design:

Detailed design is the specification of the internal elements of all major system components, their properties, relationships, processing and often their algorithms and data structures.

Detailed design is the phase where the design is refined and plans, specifications and estimates are created. Detailed design will include outputs such as 2D and 3D models, P & ID's, cost build up estimates, procurement plans etc. This phase is where the full cost of the project is identified.

3.1 Data Flow Diagrams



CHAPTER 4

CODING

The coding phase of the software life cycle is concerned with the development of code that will implement the design. This code is written is a formal language called a programming language. Programming languages have evolved over time from sequences of ones and zeros directly interpret-able by a computer, through symbolic machine code, assembly languages, and finally to higher-level languages that are more understandable to humans.

4.1 Sidebar

```
import React from "react";
import "./Css/Sidebar.css";
import DiscFullIcon from '@material-ui/icons/DiscFull';
import { NavLink } from "react-router-dom";
```

```
import HomeSharpIcon from '@material-ui/icons/HomeSharp';
import {
    ProSidebar,
    Menu,
    MenuItem,
    SidebarHeader,
    SidebarFooter,
    SidebarContent,
} from "react-pro-sidebar";


function login() {
    return (
        <div>
  <div className="menu menu--right">

     <ul>
        <li className="menu-list-item menu-list-item--right">
            <img src="
CMCAYAAACuwEE+AAAABHNCSVQICAgIfAhkiAAAG3BJREFUeF7tXQl4TUf7/909izUkSCjFp02p
pbqh1VZVq/adagVB1BpJ7YpEmpCoJSGIJYTYqa20VMunltoJgiQ0CYIkEtnuzd3O/5kTyT/03n
vOufee6PfcOX08bd13Zt75ze+8M/POO++RMAzDgD4UAZ4ISChheCJFxVgEKGEoEQQhQAkjCC4q
TAlDOSAIAUoYQXBRYUoYygFBCFDCCIKLClPCUA4IQoASRhBcVJgShnJAEAKUMILgosKUMJQDgh
CghBEEFxWmhKEcEIQAJYwguKgwJQzlgCAEKGEEwUWFKWEoBwQhQAkjCC4qTAlDOSAIAUoYQXBR
YUoYygFBCFDCCIKLClPCUA4IQoASRhBcVJgShnJAEAKUMILgosKUMJQDghCghBEEFxWmhKEcEI
QAJYwguKgwJQzlgCAEKGEEwUWFKWEoBwQhQAkjCC4qTAlDOSAIAUoYQXCZE2ZAUksajUaQHJPk
vxkwkJB/JBJIpSX/Jn/+1x9KGBtGMO1eBs5dTMClq7eQdv8hCguLYDASwpQkJiX0ICRRKhWoWc
MNbzVvgg/btEbjhq9AKpXa0PLLK/rSCKPWFON+xiM4KVXw8vT41799xVodzl64iuMnL+DvtAd4
nJmNwiJNCTF4Wo5SIlVydUY9r9po804L9OzaAa4uzi+PAQJbrlDC5OUV4Mix0zj253ncTkmFTq
dn1W36ekPMnTYGNdyqCVRfXHGNphgnzlzEyTOXcPFKIgoKNez0Yq+HEEihkKF929bo8eUnaPp6
Y3tVLVo9FUKYgsIi7Pn5KDbv/AXFxVqTb6Tf0L7o17OTaB0VUnFq+gMc+eM09h78A2q1tmRuEf
lhGCM+eL8lhgzsgUav1hO5NeurF5Uw5A1KvpOGSTMiQKYgS6a7X4+O8BvW3/qe2FiS6Jr1JBdr
43bhyLEzkEhezhqDYDR8cA/07t4RKqXSxl7Zv7iohDl99jJmhCyDjMcCz/frnhjU90v795BHjW
R3E7dlHzZt/5ksSHiUEFeEkLdKJRfErQxF5cqu4jYmsHbRCJP7NA+DR01HcbGOl0qB44agc8cP
eMnaUyjxVgqCI1bhcWYO78WrPdu3VJerqzPmThuNVm96V1STnO2IRpjklDSM9A+CTCbjVIIIzJ
zih0/atuYlay+h9fF7sPWnX6HXG+xVpd3rIVvyMcP7o+sXH9m9bmsqFI0wKXfTMGICP8IQExwe
PAmtW7xhTR8ElyEECV20mt0i890SW2zkxQ/C2Hlak8tkGD9yILr8C0gjGmHu3E2H74S5vCwMIc
yyiOnwbtJQ8OALLVBYpMZ33y/E7eQ0/mR5RghGpQLj6grGxRmMiwuYSq5gKlWG0dUZIAtUqQww
6CEt0kCSnw9Jbi6keXmQ5OUDhYUlmy0rySSTSTF/rj9aNnuNv95CweEhLxphyO5o5ER+FsZgMG
Jt1Fw0bFCXh8rWi5At/ZjJP+Dv1Ae8QGeUSjBVqwItmkLWtBnkNdxQXS5BfaUc9Z0VqKuSo4ZC
hsoyGVRSCQwMkG/Q41GxHrfVelwr1CBDawBx7xVnZkF17ASkKSmAWmPVTp34gOJW/IDatWpaD4
KNJUUjTFJKKkb5zwN5M7geg8GATatC4eVZi0vU6t+Js35WSCT+On/Nch1GAyCTA4wRDeZMx7BX
a+HdKs5Q8NjpmauYdS8UFWNBWjauFWkhy8mBass2SLNzBPdHKpFg18ZFqFzp5eyeRCPMzdt3MC
YwFFKehNkeGwH3mm6CAeRbIH77AayL32vaspADQ4UCmp7d0al1U9xauxEPb9/BysWz0KRRA75N
cMoR4uToDTiWW4iotCfQZTyEKn4LJEVqQVPVB23ewtypoznbE0NANMLcuJmMcVPm8zpkIxZmz6
YlqFq1shh9xKWriZgWFAnSzosPo5BD0uFj9O/8EbrWdkMdlRwB8yJx5fw1hMwcg7bvtRJFp6d6
A3Y8forYe08gP34C8pOneU9ThHhTJw5Dpw5tRdHNUqWiEeZ6YhLGT13AizDEcbZ/SyRcXV3sDg
AhycgJc5F2/9HzdZNwhLqeaDd0AL5t3hieKkXZ79OXxOLsH6cwbuRA9Or6qd11Kq2QDPwdtRbz
UrORfOk6VHv3QlKs5dWeW/UqWBsVVOFTk2iEEWJhSPTIga3L4OSk4gWWECHia9m4/efnpyKGgd
```

```
z7NfiOH4Z+XmQafN67G7x+N/675xcM6NUJI336CmnOKlm90Yjgv7Pw+9Xb7NpGUlzMWQ/ZuPXq
8jHGjfqKU9aeAqIR5tqtO5gwOZSXhSE7TUIYlcq+ZyfEcvUZEoD8gqIyzMjiV9bgFSwKDkCzKq
YtWvieozgcuw2fdXgfUycOtyfeFusKT83C/ht3oIxZC4mR4Tz0dHJSYvu6CLhUYHiEaIQ5desO
vp8cCgmP3QUJJjqwNYoNNLLns/fg74hcteU56yKtWgVrokNQz0Vldmu99OgZ7I9ci/ffaY6QWe
PtqZLFuowMgx/Ts7H3ShJUa2M51zRkSmvfphXmTBtTYTqKRpjDSalYEBgMCXFmcTxiEEar1aGP
TwDU6nLmXaXEnKVB+KCWm0U/TPSJi9i9cAXebPofLA6dwqW+XX/XGo2YlPQQ1/YcguLUGc7dk4
uzCvu2RNlVh5ey6N2fch9LJ80BeFgY4p4/sC3Krsf59zMeY+iY78vCJUmg7TvdOiFsRD9OcFef
vIxt4cvRuFE9rFw026y8prgYi5bHIe3eQ6iUCtbx2K9HJ3jW8eBsw5JAkcGIvgnp0ERGQ/rEsq
+GWJkR5KS/Xxeb2uRbWDQLsyftEZaNn8X5hhBFScA0WcM423HRuyZuF7bs+qXMklRyq4bYFSGo
zmOdtO7UZWxesBwNXvHEmqggs1jOX7waR46dLWujJACcwYBen2PIoG5wdnLiOw7/kEsoUGPCb+
cg3RBPPi5usR6ZVIK9WyLhpLL/puHFhsUjTMYTRI2ZztlZopDRaGDNaiU7bavJVrrPkECQSD+W
kAyDfn07Y/Q3vXkN4Po/L2BTxErU86zFkszc4/PtLBBLZur57JP3MGXCcF5HEObqn5LyGOdi4i
C7dZtT75gls0U/WiFKiEaYg1n5+JEQRsvtVyADvDtuEapXr8oJDB+Bh4+zMMh3atnBJyHM5tXz
UcujBp/i2HD8LDYuWg1PT3fErQg1W2ag7xRkZeea/J30acOKELxStw6vNk0Jkamp52/nYYhezU
m83t06YIzvQKvb4ltQNMIQ93fw+NmQ5udz6kLAjV8dBs/ats39pQ1dvX6bDQstDV0gQeZLwqZy
gl5aPva3U4iPiuUkTD+fQOQ8Nd0/sqVfvpCcwDfi7L8lgcj0bOyOXAd5UpLFelRKOfZujoJczr
3JsEUh0QhzMb8YgZN/gOzhQ079CLjEpJL7OvZ4jh7/C2GL15ZVNWJIbwzs/QXvqtcePI4tMfGo
6+mB9dHmp6QeX/ujsJyPp3wDxKotnBeAVs1ti5bL0urQ96f/Qr55q8X1IMFwQ/Q81LPBovEBSD
TCJBdpMXzOUig53ozSNUZEcADeamEbuKUd3rh1PzZs3c/+L1kvhgdNROuWTfngwcqs/OlX7Nyw
C6941ca65cFmy3UeOB46jWmvLCFMyMyxaPNuS97tmhMMu30fv02fB4nBaLGu4Onk7Mv29iw1Ih
phsrV69ApfA+X5i7wAmxk4Ap98+C4vWS6hxdFx+Pnwn6wYibXZvCYMdWq5cxUr+31p/B7s33EQ
DRt4IWbJHJPlSKRyl/5jYdSaj1meEeCLDu3f492uOcHL+WpMXLIeiguXLNbVv+dnGDWU221gi0
KiEUZnMKJz9Dbg6B+89Bvj2x+9u3XkJcslNDcsGn/+dZkVI0FHJV5k/scOC1ZtxpFDx+Dd5FVE
hU832VyukUHfAWOBZ5fxTAkFjv0GnT/7kEtdzt+LjUZ0O3QWxtXrLMp6/6cBoiJmcNZni4BohC
FK9Yr/Bfk7dvPSr0/3T/Ht8AG8ZLmEAmaE4+qNZFbMs7Y74lb+wFXkud+DflyNEyfO4Z2WTRE2
d6LJsqnFOgwfPBESfcntTVMPOe3u2aWDoLbNCUclpmHv1GCLjlCVSoF9WyIh4+Fdt1YpUQkz8t
czuBu9lpfzrs07zTFv5jhr+/FcOd/xs5GaXrLYJrcJyTVcIc/UoCW4cOkGOn78Hqb5+5osmqjW
YewQf0h15qekUT590L/X50KaNiubmFeIcd9MtHjBjuw2d274UdQrx6ISZvaVFJyaFQrwuGpSp1
YNbFwVZhdwy293hwzsiiEDuwuqd/SkeUi6kwafQd3Mlr1UqEXgsABILfiZrGnbnKLZOgP6jAuC
/JH5XSdZr61ear/dpildRCXMivRs7JgcBKmmJMuBpYfEqu7etNhmby95y7oNGg+ttmSqmDZpOD
p+9D5X88/93m9oIJ7k5GFm4Eh0aG96IX6pQIPAEVMs9s2ei1CtkcHnEeshO3XKrMWuiOs6ohJm
X1Y+Fi2MgfzmLc4BI36ERSGBaGmj34JcI+k+aEKZk27RD9+hedMmnO2XChSpNeg5eCJ7uS160S
y83th0TO/lAjUCRk2DlMTjmnm6d/4IE/wG827bsiAD391/IHXDFotT/PQpfvhUxAuBohLmfJ4a
k3YegWrfAV6gfdO/C3y+6sFL1pxQZtYT9B82uexYgJjoV+vzv75yLTEJE0hoqUyKHRuXoHol00
FWiYUajPWbAWlBgVl9P+/QBpMnDLOpP+ULLz+fiN3zFln0WI/1H45eHwuzqEIUFJUwWVo9uh29
BNeVMbwWvu+0egNhc/yF6P8P2bup9zF83OwywsQuD2aT9/B94rbuY51+KldXHIhfAnOXZNI1Wv
iMnglp7lOzVX/UrjW+n+zHt2lOuT/u3ENIQLDFwKphE30x+BPbfT/mlBGVMGRO7XkuGYXz5vNa
+HrUrI7NaxZwAmdJIOFGEiZOCy9L/BO3MkTQGdV33y/C5YSbcPOqje0WvLx5Oj16TgiCNOOF4P
Jyytlz50eqTXqYidGjZ5onDMNgcKAfhrV/2yYMLRUWlTCk4Yi0LBycHwXZgwwenWDYE16vOtZf
aDt97gpmhSxjzTaJItkcE8b7lJoo+I3fDGQ8ykKj5t5YFTzJrM4GoxGdZi2F5EaiWZm3mr+O8O
AAHv3mJ3L/4WPWqpkN9jUaMWiWP3zfbcavQiukRCfMydwiTDt0EsptO3jFqPbt9im+HWH9Mf3R
42cQtrjEI0os3LZ14ahZozovaPR6Pbp/NREkvLN1p4+wYIzlBWun6B0w/nrY7HTr3aQBosLt53
m9fjOZXV+ZTSBgMODr8O8x1Nt+l+9eBE50wjwo1qH3+WS4/rgUEqPlwzOiXC13N8Svns9rgE0J
Hfj1OJasiGd/IjuvXSTOploVXvUd//McgiNiWGJ3HTcc/p+1sViu8+Hz0C1fZZYwnnVIPI0wL7
OlBknat6iYrWYJw0gkGBkVioF1+cX98ALlBSHRCaM3Muh8JRWG6BhIH2dyLn7JXWzirbT27vBP
B45i+ZptbDeJT2bf5kheWZyINRo3ORS3klPZqcw/MgTdXrEcn9P9ShoKg8MgMXGjkrRPArT3xE
faJZEi0S9gZgQSnh15mBpso5sbAiK+R9ca4t27Fp0wpGOxD3KwOiEFLtHm38ZSAEg4wodthLvz
S8uXJwyZYg7tiIazM3dsbbFWi74+gewtA72HB9YtnYOGzpavvfS/fg9ZEZGQ5pqOuiuZEhewOX
ptfTTFWnzZ91tILXjNda1aIsx/KNpWtf8N0lL9K4QwJP1Fn+v3oIyKhjTHNLjlASVRYztiF/Ky
```

```
DC8OxKHDJ/Bj9Eb2r+UyCX7eHs0rR82Fy9cxde5Stpzzh22xJ8AHMo5cLhNuZ+DqiljI7/5tkg
+EMHOm+KF9O9t3LSSz5/Bxcyz6YNS9emBjv45o6CJeMHiFEEZPttcJ6cg7exHK/dyJBwnQvbp8
YtU10EvXbiNwRjgLbLv3WiB4BveBJrFEg0dNQ1b2U3b98s6Y4Qj73PL6hTBk7YMcbNh5CEoSwm
GGXPZy3gXNj8aJMyUhG2bYCe23o/DrZ63hzCNjhrUWr0IIQ5T7/UkBZt3NhNOKGEizsjn1JaRZ
GzkXDep7ccqWF1Dr9Ahaug55DzMxc/IoePEInLqZdBdjvwsteXvlcvguDsKgetwBV9cKNBj110
04Ry1nvytg6qlfrw57ad6Wh+Ta8ZsUYnEtxMhl8Jw9DRubkzBX8TKBVhhhCGA9EtKRc+EylD/t
41z8EvmWzZogbM5EKBT2vUJbfvDICe+MeUtw4fJN9q+177ZG9AQfNKvEve4hl+g7Xk2HdNlKsy
8BSTm7KeYHuNe0buei0+sxI3gp+z0DS4/u7dbo49MXE+pZ1w5fQlcoYVbfe4J1j/PgFLsRsvR0
Th3JAnjooG74ekBX3hH/nJW+IHDqr8uYFboc5LScBADLxvrh546tIeeZIn7AtXt4ePAIFKdOm2
367VbeCJvtb1Uf4rbsxYatByyXZRiox/gh/D1vtKsm3g6JdLBCCUPuOPZKuIfHj7LgHBkN8BgU
duE41Q/t29q+cHxxREnOO3K3qDS7A+PsjK8WzoEvmwKE3zM5+SFO3/obTqvWmI2GI6lnt8WGo1
oVYQmTDv9+EguWruckGnEDFMyYgp9a1H8uzw2/HgiTqmDCAAey8vFDajYb8qDavpPX1KRUyjHr
u5Fo+659s0HNDInEmXMJZccIGp+vEff5+2jkwj/+d19mHsLSsuG0JhayDNPBTYT01atWxsaYUF
7XZ4n8/l+Osf4kMmVafBgGmr694d6yKXY1q1diKUV8KpwwBoaBf9JDXMhTQ/H7cchP8UvVpZDL
MXJIL/Ts+imvnDNcmC1YtBZHjv9/dgSjizMqTfbH7hYNeE9HpI08vQFdE9LB3LgJFUf8css3X8
PsKX6oUrmSWfXyCwqxLGYzfj9xjr0iw/WQ9K/q7/wR6OWGPrX4ebS56rT0e4UThiiTptFi2M0M
qA1GKPcegPzKVV6Whrx5H7drjW99B/A+H3qx80+f5iN08WpcuJRY1iYjlaJ4mA8iPmyBNtWEO7
38kzJwNisfzksiOVOOebi7sZmtyCdvqlcruRpMPNJ3U++xJDl05ATyC8wHZT3XH4aB9otO0L77
Nva+WRe17JxfxxRxXgphiCKnnxYhIPkxuwN02rkb0hs3eW8GSWoNEi9LHGK1PWpyzvGkPbVGg7
/OJ2Dhsg2sN7fsAI9hYHjDG8oBfbD3zXpwssKHsfvxU0Sk50CeeBOqndy3JNgPbUmAWjXd2CRK
WU+eoqhILdhyGj3coR49Eg1Vcmx8w0v06eglLHqf5+zmjBwsyyCeXwkUvxyB/Nx5XtkeSmsxMk
Y0blgPA3t2QlPvJmyOPIVCzgJHUoiQMEu1WoNjJ88hfvvPKCz65y1Fcv6iGeuH0Ffd8bGb+anC
kpnO0enRLeEejHo9nBcshIR8u0DktQSZrYq/GQTm1Vexrakn6vJIY2LLVFRa9qVZGKIAedOWpG
dje2YByBBL7/wNp03xgIBvFbFvK3thTcr+YcepdO6XlNx8JKfWpkICGLkcmnGj0cijBjZ4e/Ky
VCZBZxhEpGfhp6xCSDOz4bRilT3GxmId+tatoP3yC3ipFNjRjH8Iqq2KvVTClJJmXUYO1mTklQ
x2URGUB3+F/PoNUd9Skha+eMQwOHvUxGZvT7iXS7tqDagk0/fXiQ9Y66bauh2yW0mi6W+s7QHN
MB9ArkBMk9poVpnbyWhNn/5Va5gXlTmcnc9uT4uJdWAYSNPvQXHsv5CRgz17mneyZnmlPrR9uo
OpUgUhDWqig5VTUfk+kISGk5Mf4XS+BtDq4BSzFtInT+w1Ts/qYcAoVdCMHgmmWlW0reyEhf/h
H69sD2VeuoUp34k76mIsSH2Cq4WasulBmpoG+bkLkKXcgYTcb7KWPCSdWCVX6Nu2ge59ctdIgn
Fe1fBVbft9mDSjWIde1+6z/kjpgwdQxW3mlVCJ70AySgU0I33B1HCDs0SC9a/XQT1n/j4jvu1Y
kvtXEYbdYjIMNmXkYmtmPnL1ZO3xTH2jEbLEm5BfSYAkOxsS8jmZ0qzZpkhU+skauRzG6tVgfL
MZtG3fZ72xcokEAXXd0MvDvn4Lsp6anPIYJ5+qWb1l12+w52Z8Ig0tDibDwOjiguJRvmCqVmHX
fmO9quNrO5KdL5n+dYQpU5xhsOpBDnZn5SOf9XY+78EkMxf5FpE07R7k6feAnCfsZ2XYx8kJRv
eaMDZuBEOD+s+57N3lUqx6rQ5q27hmsQTwl1fSkPvMQytL+RvKrdtKLu1bYR3ZGdrDHRrfoYBC
wa7nP6jihIjGJFBeXK/uv3oNY0o5sjXWGYHrhRp2N3VTrWMXldbARKCeWNcN3WtWgUzC/+PkfN
+88nJnnhZhUvLjZzs2BpKCAqhI8JjQTwUajdB++jF07dqWJch2lUqw5826cOVxX90a3bnK/Hst
zAual3465n6xDhfzNTiTp8btIh0KyJb52dfpS4lEZiMjGHae/6iaC7rUqARvVxUqi5z/rbzKC1
OzsCubuAuePVot5OcvQnHseImfxtzzbNGvb9kCuvbtwFSvXmZIXCQSbGnqBXelnGtcRfv9f4Yw
phDQGRnk6vUgmQ3y9EZoye6KBF/LpKipkKOWUmbTh7FsQZ1k9Ca7pnMFLzgLC4sgS0lhd3/sJT
hy1ZawysUVZLtsqF8fhkYNwVSp/JxfqIZchpjXaqOOiFMpn/7+TxOGTwdfpkyB3oCZdzJxtuCf
n+wjhoR1JpY7YCRT8IvTLbGszV1VCGvoAbeXaFlKcaSEEZlRJN1Y0N1MHHtKkkwLW30pJUDPGp
Uxvp4bZ0C6yN0oq54SpkKQZnDhqRpT72aiyEjsiHnqEItCFvZ1FDIsf612hZxAC4GAEkYIWjbK
knCO83lFWPkgF0kaHWs1So++yOeRXKQS9hLa4FpV2YUt1zUXG9WxqjgljFWw2VaIWBHiWyLffd
QyAMndXUkmRVU5WaQLm7Zs00R4aUoY4Zg5dAlKGIcefuGdp4QRjplDl6CEcejhF955ShjhmDl0
CUoYhx5+4Z2nhBGOmUOXoIRx6OEX3nlKGOGYOXQJShiHHn7hnaeEEY6ZQ5eghHHo4RfeeUoY4Z
g5dAlKGIcefuGdp4QRjplDl6CEcejhF955ShjhmDl0CUoYhx5+4Z2nhBGOmUOXoIRx6OEX3nlK
GOGYOXQJShiHHn7hnaeEEY6ZQ5eghHHo4RfeeUoY4Zg5dAlKGIcefuGdp4QRjplDl6CEcejhF9
55ShjhmDl0CUoYhx5+4Z2nhBGOmUOXoIRx6OEX3nlKGOGYOXQJShiHHn7hnaeEEY6ZQ5eghHHo
4Rfe+f8DYuOBAlUYXCkAAAAASUVORK5CYII="/>
```

```
        </li>
    </ul>
        <Menu >
            <MenuItem icon={<img src="https://image.flaticon.com/icons/png/512/4395/4395359.png" style={{color:"#FFFFFF",height:"30px",width:"30px"}}/>}>
            <NavLink to="/Dashboard"  activeClassName="active_class" > Consumer</NavLink>
            </MenuItem>

            <MenuItem icon={<img src="https://image.flaticon.com/icons/png/512/2745/2745370.png" style={{color:"#FFFFFF",height:"30px",width:"30px"}}/>}>
            <NavLink to="/ProviderScreen"  activeClassName="active_class" >  Provider </NavLink>
            </MenuItem>

            <MenuItem icon={<img src="https://image.flaticon.com/icons/png/512/2745/2745370.png" style={{color:"#FFFFFF",height:"30px",width:"30px"}}/>}>
            <NavLink to="/AllRequestScreen"  activeClassName="active_class" >  All Request </NavLink>
            </MenuItem>

            <MenuItem icon={<img src="https://image.flaticon.com/icons/png/512/2745/2745370.png" style={{color:"#FFFFFF",height:"30px",width:"30px"}}/>}>
            <NavLink to="/CancledServiceScreen"  activeClassName="active_class" >  Cancelled Services</NavLink>
            </MenuItem>


            <MenuItem icon={<img src="https://image.flaticon.com/icons/png/512/2745/2745370.png" style={{color:"#FFFFFF",height:"30px",width:"30px"}}/>}>
            <NavLink to="/PaidServicesScreen"  activeClassName="active_class" >  Paid Services</NavLink>
            </MenuItem>




            <MenuItem icon={<img src="https://image.flaticon.com/icons/png/512/2745/2745370.png" style={{color:"#FFFFFF",height:"30px",width:"30px"}}/>}>
            <NavLink to="/AllComplaintsScreen"  activeClassName="active_class" >  All Complaints</NavLink>
            </MenuItem>
        </Menu>
```

```
        </div>
  </div>
);
    }
export default login
```

4.2 Consumer Screen

```
import React, { useState, useEffect, Profiler, Component } from 'react';
import './App.css';
import MaterialTable from 'material-table';
import './Css/edit.css';
import ReactDOM from "react-dom"
import CancelIcon from '@material-ui/icons/Cancel';
import ConsumerDetailsButtons from "./ConsumerDetailsButtons"
import ConsumerRejectButton from "./ConsumerRejectButton"
import axios from "axios";
// import DeactivateConsumerButton from "./DeactivateConsumerButton"

import Icon from '@material-ui/core/Icon';
import Button from '@material-ui/core/Button';
import TextField from '@material-ui/core/TextField';
import Dialog from '@material-ui/core/Dialog';
import DialogActions from '@material-ui/core/DialogActions';
import DialogContent from '@material-ui/core/DialogContent';
import DialogContentText from '@material-ui/core/DialogContentText';
import DialogTitle from '@material-ui/core/DialogTitle';
import "./Css/cs.css"
// or
// import { Icon } from '@material-ui/core';




const empList = [
    { id: 1, name: "Snow", email: '', Role: "Service Consumer", city: "Ban
galore", status: "Active" },
    { id: 2, name: "Lannister", email: '', Role: "Service Consumer", city:
 "Chennai", status: "Active" },
    { id: 3, name: "Stark", email: 'Sathi@zimozi.co', Role: "Service Consu
mer", city: "Jaipur", status: "Active" },
    { id: 4, name: "Targaryen", email: '', Role: "Service Consumer", city:
 "Hyderabad", status: "Active" },
    { id: 5, name: "Clifford", email: '', Role: "Service Consumer", city:
"Hyderabad", status: "Active" },
    { id: 6, name: "Frances", email: '', Role: "Service Consumer", city: "
Hyderabad", status: "Active" },
```

```
    { id: 7, name: "Frances", email: '', Role: "Service Consumer", city: "
Hyderabad", status: "Active" },
]



function App() {
    const tableRef = React.createRef();
    const [open, setOpen] = React.useState(false);

    const [userId, setUserId] = useState(null)

    const handleClickOpen = (rowData) => {
        setOpen(true);
        console.log(rowData._id)
        setUserId(rowData._id)
    };

    const handleClose = () => {
        setOpen(false);
    };

    // componentDidMount(){
    //     this.Deactivateuser();
    // }


    const [modalShown, toggleModal] = React.useState(false);
    const [dataLoad, setDataLoad] = useState(true)
    const [PageToken, setPageToken] = useState("")

    // const [data, setData] = useState([])

    useEffect(() => {

        axios.get("https://84e6ux9vhc.execute-api.us-east-
1.amazonaws.com/develop/admin/getAllServiceConsumers", {
            headers: {
                Authorization: localStorage.token,
                Role: "Service-Consumer"

            }

        }).then((result) => {

            console.log("result", result.data.data)
            // const data = setData(result.data.data)
            setDataLoad(false)
        })
```

```
    }, []);


    function Deactivateuser(rowData) {
        // console.log("...", rowData._id);
        // alert("hello")
        console.log(userId,"...")

        axios.post("https://84e6ux9vhc.execute-api.us-east-
1.amazonaws.com/develop/admin/deactivateUser/" + userId, {
            headers: {
                Authorization: localStorage.token,
                Role: "Service-Consumer",
                nextPageToken: "TmV4dF9QYWdlX1Rva2VuMQ=="

            }



        }).then((result) => {

            console.log("Deactivate successfully", result.data.data.status
)
            // const data = setData(result.data.data)
            setDataLoad(false)
            setOpen(false);
            window.location.reload()

        })

    }



    const columns = [
        { title: 'Avatar', field: 'profile.avatar', render: rowData => <im
g src={`${rowData.profile.avatar ? rowData.profile.avatar : "https://uploa
d.wikimedia.org/wikipedia/commons/1/12/User_icon_2.svg"}`} style={{ height
: "70px", width: "70px", borderRadius: '80%' }} />, filtering: false },
        // { title: "ID", field: "id" },
        { title: "Name", field: "profile.firstName" },
        { title: "Email", field: "email" },
        { title: "Role", field: 'role' },
        { title: "Status", field: 'status' },


        // { title: '', field: 'profile.avatar', render: rowData => <butto
n onClick={() => Deactivateuser(rowData)}>{`${rowData.profile.status === "
```

```
activate" ? rowData.profile.status : "Deactivate"}`}</button>, filtering:
false },
        { title: '', field: 'profile.avatar', render: rowData => <Button v
ariant="contained" class="btn btn-
info" onClick={() => handleClickOpen(rowData)}>{`${rowData.profile.status
=== "activate" ? rowData.profile.status : "Deactivate"}`}</Button>, filter
ing: false },
        { title: '', filtering: false, search: false, field: 'imageUrl', r
ender: rowData => <ConsumerDetailsButtons /> },




    ]


    return (
        <div className="App">

            <div>

                <Dialog style={{width:"30%",marginLeft:"40%"}} open={open}
 onClose={handleClose} aria-labelledby="form-dialog-title">
                    <DialogTitle id="form-dialog-
title"> Are u sure you want to Deactivate user?</DialogTitle>



                    <DialogActions>
                        <Button onClick={handleClose} color="primary">
                         No
          </Button>
                        <Button onClick={Deactivateuser} color="primary">
                        Yes
          </Button>
                    </DialogActions>
                </Dialog>
              </div>

              <MaterialTable
                tableRef={tableRef}
                  actions={[
                      {
                          icon: "edit",
                          tooltip: "Edit",
                          onClick: (event, row) => { toggleModal(!modalShown
); }
```

```
                    },


                        {
                          icon: 'refresh',
                          tooltip: 'Refresh Data',
                          isFreeAction: true,
                          onClick: () => tableRef.current && tableRef.curr
ent.onQueryChange(),
                        }

                    // {
                    //     icon: "text",
                    //     tooltip: "Detail",
                    //     onClick: (event, row) => { alert("are u sure wa
nt to delete?"); }
                    // },

                ]}


                style={{ width: "80%", marginLeft: "17%", marginTop: "3%"
}}
                title={""}
                data={query =>
                    new Promise((resolve, reject) => {

                      let url = 'https://84e6ux9vhc.execute-api.us-east-
1.amazonaws.com/develop/admin/getAllServiceConsumers?'
                    //   url += 'size=' + query.pageSize
                      url += 'nextPageToken=' + ( PageToken)
                      axios.get(url,{headers:{Authorization: localStorage.
token, Role: "Service-Consumer" }})
                        // .then(response => response.json())
                        .then(result => {
                            setPageToken(result.data.pagination.nextPageTo
ken)

                            console.log(result.data.pagination.nextPageTok
en)

                            resolve({
                            data: result.data.data,
                            page: result.data.pagination.page,
                            totalCount: result.data.pagination.total,

                          })
                        })
                    })
                  }
                columns={columns}
```

```
                isLoading={dataLoad}


                options={{

                    actionsColumnIndex: -
1, addRowPosition: "first", filtering: true, search: false
                }}


            />
        </div>
    );
}

export default App;
```

4.3 All Request

```
import React, { useState, useEffect, Profiler } from 'react';
import './Css/edit.css';
import MaterialTable from 'material-table';
import ReactDOM from "react-dom"
import CancelIcon from '@material-ui/icons/Cancel';
import ConsumerDetailsButtons from "./ConsumerDetailsButtons"
import ConsumerRejectButton from "./ConsumerRejectButton"
import axios from "axios";

import Icon from '@material-ui/core/Icon';
// or
// import { Icon } from '@material-ui/core';



const empList = [
    { id: 1, name: "Snow", email: '', Role: "Service Consumer", city: "Ban
galore", status: "Active" },
    { id: 2, name: "Lannister", email: '', Role: "Service Consumer", city:
 "Chennai", status: "Active" },
    { id: 3, name: "Stark", email: 'Sathi@zimozi.co', Role: "Service Consu
mer", city: "Jaipur", status: "Active" },
    { id: 4, name: "Targaryen", email: '', Role: "Service Consumer", city:
 "Hyderabad", status: "Active" },
    { id: 5, name: "Clifford", email: '', Role: "Service Consumer", city:
"Hyderabad", status: "Active" },
    { id: 6, name: "Frances", email: '', Role: "Service Consumer", city: "
Hyderabad", status: "Active" },
```

```
    { id: 7, name: "Frances", email: '', Role: "Service Consumer", city: "
Hyderabad", status: "Active" },
]

function Modal({ children, shown, close }) {


    return shown ? (
        <div
            className="modal-backdrop1"
            onClick={() => {
                // close modal when outside of modal is clicked
                close();
            }}
        >
            <div
                className="modal-content1"
                onClick={e => {
                    // do not close modal if anything inside modal content
 is clicked
                    e.stopPropagation();
                }}
            >
                <CancelIcon onClick={close} style={{ marginLeft: "95%", ma
rginBottom: "3%" }} />
                {children}
            </div>
        </div>
    ) : null;
}

function App() {

    const [modalShown, toggleModal] = React.useState(false);
    const [modalShown1, toggleModal1] = React.useState(false);
    const [dataLoad, setDataLoad] = useState(true)
    const [fname, setFname] = useState("")
    const [lname, setLname] = useState("")
    const [city, setCity] = useState("")
    const [role, setRole] = useState("")
    const [address1, setAddress1] = useState("")
    const [address2, setAddress2] = useState("")
    const [status, setStatus] = useState("")
    const [img, setImg] = useState("")
    const [data, setData] = useState([])
    const [userId, setUserId] = useState("")

    console.log(fname,lname,city,role,address1,address2,status,img)
```

```
    function updateCreateService() {
        axios.put("https://84e6ux9vhc.execute-api.us-east-
1.amazonaws.com/develop/admin/AdminProfile/" + userId, {
            "profile": {
                "category": "Individual",
                "firstName": fname,
                "lastName": lname,
                "company": "Zimozi Solutions Pvt. Ltd.",
                "city": city,
                "address": {
                  "streetAddress1": "AA1",
                  "streetAddress2": "AA2",
                  "city": "Noida",
                  "state": "U.P",
                  "zipcode": "324001",
                  "country": "India"
                }
              }
            }, {
             headers: {
                 Authorization: localStorage.token,
                 Role: "Service-Provider",


             }



        }).then((result) => {

            console.log("create user", result)
            // const data = setData(result.data.data)
            // setDataLoad(false)
            // setOpen(false);
            // window.location.reload()

        })
    }


    function DeleteService() {
        axios.delete("https://84e6ux9vhc.execute-api.us-east-
1.amazonaws.com/develop/admin/deleteUser/" + userId, {

            headers: {
                Authorization: localStorage.token,
                Role: "Service-Consumer",
```

```
            }



        }).then((result) => {

            console.log("create user", result)
            // const data = setData(result.data.data)
            // setDataLoad(false)
            // setOpen(false);
            // window.location.reload()

        })
    }




    useEffect(() => {

        axios.get("https://84e6ux9vhc.execute-api.us-east-
1.amazonaws.com/develop/admin/getAllServiceProviders", {
            headers: {
                Authorization: localStorage.token,
                Role: "Service-Provider"
            }

        }).then((result) => {
                console.log("result..", result.data.result.data)
              const  data = setData(result.data.result.data)
             setDataLoad(false)


        })

    }, []);

    const columns = [
        { title: 'Avatar', field: 'profile.avatar', render: rowData => <im
g src={ `${rowData.profile.avatar ? rowData.profile.avatar : "https://uplo
ad.wikimedia.org/wikipedia/commons/1/12/User_icon_2.svg"}`} style={{ heigh
t:"70px",width: "70px", borderRadius: '80%' }} />, filtering: false },
        // { title: "ID", field: "id" },
        { title: "First Name", field: "profile.firstName" },
        { title: "Last Name", field: "profile.lastName" },
        { title: "City", field: "profile.address.city" },
        { title: "Address1", field: "profile.address.streetAddress1" },
```

```
        { title: "Address2", field: "profile.address.streetAddress2" },
        { title: "Role", field: 'role' },
        { title: "Status", field: 'status' },
        // { title: '', filtering: false, search: false, field: 'imageUrl'
, render: rowData => <ConsumerRejectButton /> },
        { title: '', filtering: false, search: false, field: 'imageUrl', r
ender: rowData => <ConsumerDetailsButtons  /> },
        // { title: "State", field: actions, }

    ]



    return (
        <div className="App">
            <Modal
                shown={modalShown}
                close={() => {
                    toggleModal(false);
                }}
            >

                 <div class="row g-2 ">
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid ">First Name</la
bel>
                            <input type="email" class="form-control mt-
2" id="floatingInputGrid" placeholder="name@example.com"  value={fname} on
Change={(e) => setFname(e.target.value)}/>

                        </div>
                    </div>
                    <div class="col-md ">
                        <div class="">
                            <label for="floatingInputGrid">Last Name</labe
l>
                            <input type="email" class="form-control mt-
2" id="floatingInputGrid" placeholder="name@example.com"  value={lname} on
Change={(e) => setLname(e.target.value)}/>
                            {/* <label for="floatingSelectGrid">Works with
 selects</label> */}
                        </div>
                    </div>
                </div>
                <div class="row g-2">
                    <div class="col-md">
                        <div class="">
```

```
                            <label for="floatingInputGrid" className="mt-
2">City</label>
                            <input type="email" class="form-control mt-
2" id="floatingInputGrid" placeholder="name@example.com" value={city} onCh
ange={(e) => setCity(e.target.value)} />

                        </div>
                    </div>
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid"className="mt-
2">Role</label>
                            <select class="form-select mt-
2" id="floatingSelectGrid" aria-
label="Floating label select example" value={role} onChange={(e) => setRol
e(e.target.value)}>
                                <option value>Service Provide</option>
                                <option value="1">Service Provider</option
>
                                <option value="2">Service Consumer</option
>
                                {/* <option value="3">Three</option> */}
                            </select>
                            {/* <label for="floatingSelectGrid">Works with
 selects</label> */}
                        </div>
                    </div>
                </div>
                <div class="row g-2 ">
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid "className="mt-
2">Address 1</label>
                            <input type="email" class="form-control mt-
2" id="floatingInputGrid" placeholder="name@example.com"  value={address1}
 onChange={(e) => setAddress1(e.target.value)}/>

                        </div>
                    </div>
                    <div class="col-md ">
                        <div class="">
                            <label for="floatingInputGrid"className="mt-
2">Address 2</label>
                            <input type="email" class="form-control mt-
2" id="floatingInputGrid" placeholder="name@example.com" value={address2}
onChange={(e) => setAddress2(e.target.value)} />
                            {/* <label for="floatingSelectGrid">Works with
 selects</label> */}
                        </div>
```

```
                    </div>
                </div>
                <div class="row g-2">
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid" className="mt-
2">Profile Picture</label>
                            <input class="form-control mt-
2" type="file" id="formFile" value={img} onChange={(e) => setImg(e.target.
value)}/>

                        </div>
                    </div>
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid"className="mt-
2">Status</label>
                            <select class="form-select mt-
2" id="floatingSelectGrid" aria-
label="Floating label select example"value={status} onChange={(e) => setSt
atus(e.target.value)}>
                                <option value>Active</option>
                                <option value="1">Active</option>
                                <option value="2">Delete</option>
                                {/* <option value="3">Three</option> */}
                            </select>
                            {/* <label for="floatingSelectGrid">Works with
 selects</label> */}
                        </div>
                    </div>
                </div>
                <div className="mt-
3"> <button type="button" class="btn btn-
secondary" onClick={updateCreateService}>Save</button></div>



            </Modal>
            <Modal
                shown={modalShown1}
                close={() => {
                    toggleModal1(false);
                }}
            >
                <p>Are u sure u want to delete this record</p>
                <div className="mt-
3"> <button type="button" class="btn btn-
secondary" onClick={DeleteService}>Yes</button></div>
```

```
                {/* <div className="mt-
3"> <button type="button" class="btn btn-
secondary" onClick={}>No</button></div> */}
                </Modal>
            <MaterialTable
                actions={[
                    {
                        icon: "edit",
                        tooltip: "Edit",
                        onClick: (event, row) => { toggleModal(!modalShown
) ; setUserId(row._id) }

                    },

                    {
                        icon: "delete",
                        tooltip: "Delete",
                        onClick: (event, row) => {  toggleModal1(!modalSho
wn1) ; setUserId(row._id) }
                    },

                ]}
                isLoading={dataLoad}
                style={{ width: "75%", marginLeft: "19%", marginTop: "3%"
}}
                title={""}
                data={data}
                columns={columns}



                options={{

                    actionsColumnIndex: -
1, addRowPosition: "first", filtering: true, search: false
                }}
            />
        </div>
    );
}

export default App;
```

4.4

```
import React, { useState, useEffect, Profiler } from 'react';
import './App.css';
```

```
import MaterialTable from 'material-table';
import ReactDOM from "react-dom"
import CancelIcon from '@material-ui/icons/Cancel';
import ConsumerDetailsButtons from "./ConsumerDetailsButtons"
import ConsumerRejectButton from "./ConsumerRejectButton"
import axios from "axios";

import Icon from '@material-ui/core/Icon';
// or
// import { Icon } from '@material-ui/core';




const empList = [
    { id: 1, name: "Snow", email: '', Role: "Service Consumer", city: "Ban
galore", status: "Active" },
    { id: 2, name: "Lannister", email: '', Role: "Service Consumer", city:
 "Chennai", status: "Active" },
    { id: 3, name: "Stark", email: 'Sathi@zimozi.co', Role: "Service Consu
mer", city: "Jaipur", status: "Active" },
    { id: 4, name: "Targaryen", email: '', Role: "Service Consumer", city:
 "Hyderabad", status: "Active" },
    { id: 5, name: "Clifford", email: '', Role: "Service Consumer", city:
"Hyderabad", status: "Active" },
    { id: 6, name: "Frances", email: '', Role: "Service Consumer", city: "
Hyderabad", status: "Active" },
    { id: 7, name: "Frances", email: '', Role: "Service Consumer", city: "
Hyderabad", status: "Active" },
]

function Modal({ children, shown, close }) {


    return shown ? (
        <div
            className="modal-backdrop1"
            onClick={() => {
                // close modal when outside of modal is clicked
                close();
            }}
        >
            <div
                className="modal-content1"
                onClick={e => {
                    // do not close modal if anything inside modal content
 is clicked
                    e.stopPropagation();
                }}
```

```
            >
                <CancelIcon onClick={close} style={{ marginLeft: "95%", ma
rginBottom: "3%" }} />
                {children}
            </div>
        </div>
    ) : null;
}

function App() {

    const [modalShown, toggleModal] = React.useState(false);
    const [dataLoad, setDataLoad] = useState(true)
    const [uid, setuid] = useState("")
    const [data, setData] = useState([])
    const[time,setTime]=useState("");
    const[booking,setBooking]=useState("");
    const[email1,setEmail1]=useState("");
    const[status,setStatus]=useState("");
    const[mainService,setMainService]=useState("");
    const[subService,setSubService]=useState("");
    const[serviceType,setServiceType]=useState("");

    console.log(time,booking,email1,status,mainService,subService,serviceT
ype)

    function updateService() {
        debugger
        axios.post("https://84e6ux9vhc.execute-api.us-east-
1.amazonaws.com/develop/admin/updateAdminServiceRequest", {
            "serviceId": uid,
            "scheduledLaterSlot": [
                27,
                21
            ],
            "servicesRequired": [
                {
                    "mainCategory": mainService,
                    "subCategory": [
                        "subService1",
                        "subService2"
                    ]
                }
            ],
            "bookingRequired": booking,
            "scheduledTime": time,
            "requiredServiceType": serviceType,
            "coordinates": [
                74.966,
```

```
                57.456
            ]
        }, {
            headers: {
                Authorization: localStorage.token,
                Role: "Service-Consumer",


            }



        }).then((result) => {

            console.log("create user", result)
            // const data = setData(result.data.data)
            // setDataLoad(false)
            // setOpen(false);
            // window.location.reload()

        })
    }



    useEffect(() => {

        axios.get("https://84e6ux9vhc.execute-api.us-east-
1.amazonaws.com/develop/admin/getAllServiceRequests", {
            headers: {
                Authorization: localStorage.token,
                Role: "Service-Consumer"

            }

        }).then((result) => {

            console.log("result", result.data.data)
            const data = setData(result.data.data)
            setDataLoad(false)
        })

    }, []);

    const columns = [
        // { title: 'Avatar', field: 'profile.avatar', render: rowData =>
<img src={ `${rowData.profile.avatar ? rowData.profile.avatar : "https://u
pload.wikimedia.org/wikipedia/commons/1/12/User_icon_2.svg"}`} style={{ he
ight:"70px",width: "70px", borderRadius: '80%' }} />, filtering: false },
```

```
        // { title: "ID", field: "id" },
        { title: "Booking Required", field: "bookingRequired" },
        {
            title: "Schedule Time", field: "scheduledTime", type: 'date',
            dateSetting: {
                format: 'dd/MM/yyyy'
            },
        },
        { title: "Requested By", field: 'requestedByDetails.email' },
        { title: "Status", field: 'status' },
        { title: "Service Type", field: 'requiredServiceType' },
        { title: "Main Categore", field: 'serviceName' },
        // { title: "Main Categore", field: 'mainCategory' },
        { title: "Sub Categore", field: 'subCategory.subService1' },
        { title: "Card ID", field: 'carId' },

        // { title: '', filtering: false, search: false, field: 'imageUrl'
, render: rowData => <ConsumerRejectButton /> },
        { title: '', filtering: false, search: false, field: 'imageUrl', r
ender: rowData => <ConsumerDetailsButtons /> },
        // { title: "State", field: actions, }

    ]


    return (
        <div className="App">
            <Modal
                shown={modalShown}
                close={() => {
                    toggleModal(false);
                }}
            >
                <div class="row g-2">
                    <div class="col-md">
                        <div class=" ms-3">
                            <label for="exampleInputEmail1" class="form-
label">Schedule Time</label>
                            <input type="date" class="form-
control" id="exampleInputEmail1" aria-
describedby="emailHelp" value={time} onChange={(e) => setTime(e.target.val
ue)} />
                            {/* <DatePickerComponent id="datepicker" place
holder="Enter date"value={time} onChange={(e) => setTime(e.target.value)}/
> */}
                        </div>
                    </div>
                    <div class="col-md">
                        <div class=" me-3">
```

```
                            <label for="floatigSelectGrid">Booking Require
d</label>
                            <select class="form-select mt-
2" id="floatingSelectGrid" aria-
label="Floating label select example" value={booking} onChange={(e) => set
Booking(e.target.value)}>
                                <option value="right now">right now </opti
on>
                                <option value="later">later</option>
                                <option value="1">1</option>
                            </select>
                            {/* <DropDownListComponent id="ddlelement" cla
ssName="mt-
2" dataSource={sportsData1}  placeholder="Select a game" value={booking} o
nChange={(e) => setBooking(e.target.value)} /> */}

                        </div>
                    </div>
                </div>

                <div class="row g-2">
                    <div class="col-md">
                        <div class="ms-3 mt-3">
                            <div className="textboxes">
                                <label for="floatingSelectGrid" className=
"form-label">Email ID</label>
                                <input type="email" class="form-
control" id="exampleInputEmail1" aria-
describedby="emailHelp" value={email1} onChange={(e) => setEmail1(e.target
.value)} />
                                {/* <TextBoxComponent placeholder="@mbo.co
m" showClearButton= {true} floatLabelType="Never"value={email1} onChange={
(e) => setEmail1(e.target.value)}/> */}
                            </div>

                        </div>
                    </div>
                    <div class="col-md">
                        <div class=" me-3 mt-3">
                            <label for="floatingSelectGrid">Status</label>
                            {/* <DropDownListComponent id="ddlelement" cla
ssName="mt-
2" dataSource={sportsData2}  placeholder="Select a game" value={status} on
Change={(e) => setStatus(e.target.value)}/> */}
                            <select class="form-select mt-
2" id="floatingSelectGrid" aria-
label="Floating label select example" value={status} onChange={(e) => setS
tatus(e.target.value)}>
                                <option value="1">Active </option>
```

```
                                <option value="....">....</option>
                            </select>

                        </div>
                    </div>
                </div>

                <div class="row g-2">
                    <div class="col-md">
                        <div class="ms-3 mt-3">
                            <label for="floatingSelectGrid" className="for
m-label">Service Type</label>
                            {/* <DropDownListComponent id="ddlelement" cla
ssName="mt-
2" dataSource={sportsData5}  placeholder="Select a game" value={serviceTyp
e} onChange={(e) => setServiceType(e.target.value)}/> */}
                            <select class="form-select mt-
2" id="floatingSelectGrid" aria-
label="Floating label select example" value={serviceType} onChange={(e) =>
 setServiceType(e.target.value)}>
                                <option value="Car Mechanic">Car Mechanic<
/option>
                                <option value="Mobile Mechanic">Mobile Mec
hanic</option>
                            </select>

                        </div>
                    </div>

                    <div class="col-md">
                        <div class=" me-3 mt-3">
                            <label for="floatingSelectGrid" className="for
m-label">Main Categore</label>
                            {/* <DropDownListComponent id="ddlelement" cla
ssName="mt-
0" dataSource={sportsData3}  placeholder="Select a game" value={mainServic
e} onChange={(e) => setMainService(e.target.value)}/> */}
                            <select class="form-select mt-
2" id="floatingSelectGrid" aria-
label="Floating label select example" value={mainService} onChange={(e) =>
 setMainService(e.target.value)}>
                                <option value="Service1">Service1</option>
                                <option value="Service2">Service2</option>
                            </select>
                        </div>
                    </div>
                </div>
```

```
                <div class="row g-2">
                    <div class="col-md">
                        <div class="ms-3 mt-3">
                            <label for="exampleInputEmail1" class="form-
label textboxes mt-1">Sub  Categore </label>
                            {/* <MultiSelectComponent style={{marginTop:"-
5%"}} className="e-control e-textbox e-
lib" id="mtselement" floatLabelType="Auto" dataSource={sportsData4}  place
holder="" value={subService} onChange={(e) => setSubService(e.target.value
)}/> */}
                            <select class="form-select mt-
2" id="floatingSelectGrid" aria-
label="Floating label select example" value={subService} onChange={(e) =>
setSubService(e.target.value)}>
                                <option value="1">subService1</option>
                                <option value="2">subService2</option>
                            </select>
                        </div>
                    </div>

                    <div class="col-md">
                        <div class=" me-3 mt-3">
                        <button type="button" class="btn btn-
dark lg" style={{marginTop:"15%"}} onClick={updateService}>Update</button>
                        </div>
                    </div>
                </div>

                {/* <h1>Look! I'm inside the modal!</h1> */}
            </Modal>
            <MaterialTable
                actions={[
                    {
                        icon: "edit",
                        tooltip: "Edit",
                        onClick: (event, row) => {
                            debugger
                            toggleModal(!modalShown);
                            setuid(row._id);
                            setBooking(row.bookingRequired);
                            setTime(row.scheduledTime)
                         }
                    },

                    // {
                    //     icon: "delete",
                    //     tooltip: "Delete",
                    //     onClick: (event, row) => { alert("are u sure wa
nt to delete?"); }
```

```
                    // },

                ]}

                style={{ width: "75%", marginLeft: "19%", marginTop: "3%"
}}
                title={""}
                data={data}
                columns={columns}
                isLoading={dataLoad}


                options={{

                    actionsColumnIndex: -
1, addRowPosition: "first", filtering: true, search: false
                }}
            />
        </div>
    );
}

export default App;
```

4.4 Cancelled Service

```
import React, { useState, useEffect, Profiler } from 'react';
import './App.css';
import MaterialTable from 'material-table';
import ReactDOM from "react-dom"
import CancelIcon from '@material-ui/icons/Cancel';
import ConsumerDetailsButtons from "./ConsumerDetailsButtons"
import ConsumerRejectButton from "./ConsumerRejectButton"
import axios from "axios";

import Icon from '@material-ui/core/Icon';
// or
// import { Icon } from '@material-ui/core';



const empList = [
    { id: 1, name: "Snow", email: '', Role: "Service Consumer", city: "Ban
galore", status: "Active" },
    { id: 2, name: "Lannister", email: '', Role: "Service Consumer", city:
 "Chennai", status: "Active" },
    { id: 3, name: "Stark", email: 'Sathi@zimozi.co', Role: "Service Consu
mer", city: "Jaipur", status: "Active" },
```

```
    { id: 4, name: "Targaryen", email: '', Role: "Service Consumer", city:
 "Hyderabad", status: "Active" },
    { id: 5, name: "Clifford", email: '', Role: "Service Consumer", city:
"Hyderabad", status: "Active" },
    { id: 6, name: "Frances", email: '', Role: "Service Consumer", city: "
Hyderabad", status: "Active" },
    { id: 7, name: "Frances", email: '', Role: "Service Consumer", city: "
Hyderabad", status: "Active" },
]

function Modal({ children, shown, close }) {


    return shown ? (
        <div
            className="modal-backdrop1"
            onClick={() => {
                // close modal when outside of modal is clicked
                close();
            }}
        >
            <div
                className="modal-content1"
                onClick={e => {
                    // do not close modal if anything inside modal content
 is clicked
                    e.stopPropagation();
                }}
            >
                <CancelIcon onClick={close} style={{ marginLeft: "95%", ma
rginBottom: "3%" }} />
                {children}
            </div>
        </div>
    ) : null;
}

function App() {

    const [modalShown, toggleModal] = React.useState(false);
    const [dataLoad, setDataLoad] = useState(true)
    const [data, setData] = useState([])

    useEffect(() => {

        axios.get("https://84e6ux9vhc.execute-api.us-east-
1.amazonaws.com/develop/admin/getCancelledServices", {
            headers: {
                Authorization: localStorage.token,
```

```
                Role: "Service-Consumer"

            }

        }).then((result) => {

                console.log("result", result.data.data)
              const  data = setData(result.data.data)
             setDataLoad(false)
        })

    }, []);

    const columns = [
        // { title: 'Avatar', field: 'profile.avatar', render: rowData =>
<img src={ `${rowData.profile.avatar ? rowData.profile.avatar : "https://u
pload.wikimedia.org/wikipedia/commons/1/12/User_icon_2.svg"}`} style={{ he
ight:"70px",width: "70px", borderRadius: '80%' }} />, filtering: false },
        // { title: "ID", field: "id" },
        { title: " Service Name", field: "serviceName" },
        { title: "Service Type", field: "serviceType" },
        { title: "Service Status", field: "serviceStatus" },
        { title: "Time Taken", field: 'timeTaken' },
        { title: "Amount Paid", field: 'amountPaid' },
        // { title: '', filtering: false, search: false, field: 'imageUrl'
, render: rowData => <ConsumerRejectButton /> },
        { title: '', filtering: false, search: false, field: 'imageUrl', r
ender: rowData => <ConsumerDetailsButtons  /> },
        // { title: "State", field: actions, }

    ]


    return (
        <div className="App">
            <Modal
                shown={modalShown}
                close={() => {
                    toggleModal(false);
                }}
            >
                <div class="row g-2">
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid">Name</label>
                            <input type="email" class="form-
control" id="floatingInputGrid" placeholder="name@example.com" value="" />

                        </div>
```

```
                    </div>
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid">Role</label>
                            <select class="form-
select" id="floatingSelectGrid" aria-
label="Floating label select example">
                                <option value>one</option>
                                <option value="1">One</option>
                                <option value="2">Two</option>
                                <option value="3">Three</option>
                            </select>
                            {/* <label for="floatingSelectGrid">Works with
 selects</label> */}
                        </div>
                    </div>
                </div>
                <div class="row g-2 mt-3">
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid">state</label>
                            <input type="email" class="form-
control" id="floatingInputGrid" placeholder="name@example.com" value="" />

                        </div>
                    </div>
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid">Status</label>
                            <select class="form-
select" id="floatingSelectGrid" aria-
label="Floating label select example">
                                <option value>1</option>
                                <option value="1">One</option>
                                <option value="2">Two</option>
                                <option value="3">Three</option>
                            </select>
                            {/* <label for="floatingSelectGrid">Works with
 selects</label> */}
                        </div>
                    </div>
                </div>
                <div class="mb-3 mt-3">
                    {/* <label for="formFile" class="form-
label">Default file input example</label> */}
                    <input class="form-
control" type="file" id="formFile" />
                </div>
                {/* <h1>Look! I'm inside the modal!</h1> */}
```

```
            </Modal>
            <MaterialTable
                actions={[
                    {
                        icon: "edit",
                        tooltip: "Edit",
                        onClick: (event, row) => { toggleModal(!modalShown
); }
                    },

                    // {
                    //     icon: "delete",
                    //     tooltip: "Delete",
                    //     onClick: (event, row) => { alert("are u sure wa
nt to delete?"); }
                    // },

                ]}

                style={{ width: "80%", marginLeft: "17%", marginTop: "3%"
}}
                title={""}
                data={data}
                columns={columns}
                isLoading={dataLoad}


                options={{

                    actionsColumnIndex: -
1, addRowPosition: "first", filtering: true, search: false
                }}
            />
        </div>
    );
}

export default App;
```

4.5 Paid Service

```
import React, { useState, useEffect, Profiler } from 'react';
import './App.css';
import MaterialTable from 'material-table';
import ReactDOM from "react-dom"
import CancelIcon from '@material-ui/icons/Cancel';
import ConsumerDetailsButtons from "./ConsumerDetailsButtons"
import ConsumerRejectButton from "./ConsumerRejectButton"
```

```
import axios from "axios";

import Icon from '@material-ui/core/Icon';
// or
// import { Icon } from '@material-ui/core';



const empList = [
    { id: 1, name: "Snow", email: '', Role: "Service Consumer", city: "Ban
galore", status: "Active" },
    { id: 2, name: "Lannister", email: '', Role: "Service Consumer", city:
 "Chennai", status: "Active" },
    { id: 3, name: "Stark", email: 'Sathi@zimozi.co', Role: "Service Consu
mer", city: "Jaipur", status: "Active" },
    { id: 4, name: "Targaryen", email: '', Role: "Service Consumer", city:
 "Hyderabad", status: "Active" },
    { id: 5, name: "Clifford", email: '', Role: "Service Consumer", city:
"Hyderabad", status: "Active" },
    { id: 6, name: "Frances", email: '', Role: "Service Consumer", city: "
Hyderabad", status: "Active" },
    { id: 7, name: "Frances", email: '', Role: "Service Consumer", city: "
Hyderabad", status: "Active" },
]

function Modal({ children, shown, close }) {


    return shown ? (
        <div
            className="modal-backdrop1"
            onClick={() => {
                // close modal when outside of modal is clicked
                close();
            }}
        >
            <div
                className="modal-content1"
                onClick={e => {
                    // do not close modal if anything inside modal content
 is clicked
                    e.stopPropagation();
                }}
            >
                <CancelIcon onClick={close} style={{ marginLeft: "95%", ma
rginBottom: "3%" }} />
                {children}
            </div>
        </div>
```

```
    ) : null;
}

function App() {

    const [modalShown, toggleModal] = React.useState(false);
    const [dataLoad, setDataLoad] = useState(true)
    const [data, setData] = useState([])

    useEffect(() => {

        axios.get("https://84e6ux9vhc.execute-api.us-east-
1.amazonaws.com/develop/admin/getAllPaidServices", {
            headers: {
                Authorization: localStorage.token,
                Role: "Service-Consumer"

            }

        }).then((result) => {

                console.log("result", result.data.data)
              const  data = setData(result.data.data)
             setDataLoad(false)
        })

    }, []);

    const columns = [
        // { title: 'Avatar', field: 'profile.avatar', render: rowData =>
<img src={ `${rowData.profile.avatar ? rowData.profile.avatar : "https://u
pload.wikimedia.org/wikipedia/commons/1/12/User_icon_2.svg"}`} style={{ he
ight:"70px",width: "70px", borderRadius: '80%' }} />, filtering: false },
        // { title: "ID", field: "id" },
        { title: " Service Name", field: "serviceName" },
        { title: "Service Type", field: "serviceType" },
        { title: "Service Status", field: "serviceStatus" },
        { title: "Time Taken", field: 'timeTaken' },
        { title: "Amount Paid", field: 'amountPaid' },
        { title: "Requested Type", field: 'requiredServiceType' },
        { title: "Requested To", field: 'requiredServiceTo' },
        // { title: '', filtering: false, search: false, field: 'imageUrl'
, render: rowData => <ConsumerRejectButton /> },
        { title: '', filtering: false, search: false, field: 'imageUrl', r
ender: rowData => <ConsumerDetailsButtons  /> },
        // { title: "State", field: actions, }

    ]
```

```
    return (
        <div className="App">
            <Modal
                shown={modalShown}
                close={() => {
                    toggleModal(false);
                }}
            >
                <div class="row g-2">
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid">Name</label>
                            <input type="email" class="form-
control" id="floatingInputGrid" placeholder="name@example.com" value="" />

                        </div>
                    </div>
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid">Role</label>
                            <select class="form-
select" id="floatingSelectGrid" aria-
label="Floating label select example">
                                <option value>one</option>
                                <option value="1">One</option>
                                <option value="2">Two</option>
                                <option value="3">Three</option>
                            </select>
                            {/* <label for="floatingSelectGrid">Works with
 selects</label> */}
                        </div>
                    </div>
                </div>
                <div class="row g-2 mt-3">
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid">state</label>
                            <input type="email" class="form-
control" id="floatingInputGrid" placeholder="name@example.com" value="" />

                        </div>
                    </div>
                    <div class="col-md">
                        <div class="">
                            <label for="floatingInputGrid">Status</label>
                            <select class="form-
select" id="floatingSelectGrid" aria-
label="Floating label select example">
                                <option value>1</option>
```

```
                                <option value="1">One</option>
                                <option value="2">Two</option>
                                <option value="3">Three</option>
                            </select>
                            {/* <label for="floatingSelectGrid">Works with
 selects</label> */}
                        </div>
                    </div>
                </div>
                <div class="mb-3 mt-3">
                    {/* <label for="formFile" class="form-
label">Default file input example</label> */}
                    <input class="form-
control" type="file" id="formFile" />
                </div>
                {/* <h1>Look! I'm inside the modal!</h1> */}
            </Modal>
            <MaterialTable
                actions={[
                    {
                        icon: "edit",
                        tooltip: "Edit",
                        onClick: (event, row) => { toggleModal(!modalShown
); }
                    },

                    // {
                    //     icon: "delete",
                    //     tooltip: "Delete",
                    //     onClick: (event, row) => { alert("are u sure wa
nt to delete?"); }
                    // },

                ]}

                style={{ width: "80%", marginLeft: "17%", marginTop: "3%"
}}
                title={""}
                data={data}
                columns={columns}
                isLoading={dataLoad}


                options={{

                    actionsColumnIndex: -
1, addRowPosition: "first", filtering: true, search: false
                }}
            />
```

```
        </div>
    );
}

export default App;
```

CHAPTER 5

TESTING

The main purpose of the test plan for this Donation Website is to discuss the testing details of the use cases of the Online Donation System. The software project test plan also describes the objective, scope and approach of the software testing effort for the Together we can project. The test plan for this project also indicates the personnel responsible for each task and also specifies the risks associated with the test plan.

"Software testing is at a very important crossroad, where it is going back to the roots on certain fronts while moving inexorably forward. For instance, test automation is growing in prominence, but manual testing is becoming a niche; we are increasingly collaborating with the developers, breaking the bounds of unrealistic independence in testing, and bringing in true conscious quality. At such an important stage, it is important to take stock of the past, present, and future to define both the direction the discipline will take as well as the careers it will entail for testers." [3]

5.1Unit Testing

The Unit Testing is a test that tests each single module of the software to check for errors. This is mainly done to discover errors in the code of this Project. The main goal of the unit testing would be to isolate each part of the program and to check the correctness of the code. In the case of this project, all the web forms and the php , JavaScript code will be tested. There are many benefits for this unit testing:

• The unit testing facilitates change in the code.

• It allows testing to be done in a bottom up fashion.

At the same time, unit testing has some disadvantages such as, it might not identify each and every error in the system.

5.2 Integration Testing

In Integration Testing, the individual software modules are combined and tested as a whole unit. The integration testing generally follows unit testing where each module is tested as a separate unit. The main purpose of the integration testing is to test the functional and performance requirements on the major items of the project. All the modules of the project developed individually would be combined together and tested as a whole system in the integration testing.

5.3Regression Testing

The Regression Testing is generally done whenever modifications are made to the source code of a project. The Regression Testing can also be defined as the process of testing changes made to the computer program and also makes sure that the older programming still works with the new changes. So, before any new version of a software product is released, the old test cases for the project will be run against the software with the changes made, to make sure that the old functionalities of the project still work.

5.4 Acceptance Testing

This testing is generally performed when the project is nearing its end. This test mainly qualifies the project and decides if it will be accepted by the users of the system. The users or the customers of the project are responsible for the test.

5.5 System Testing

The system testing is mainly done on the whole integrated system to make sure that the project that has been developed meets all the requirements. The test cases for the system testing will be the combination of unit and integration tests.

5.6 Test Cases

5.6.1 Test Case 1 – User/Admin Login

- **Fail Criteria**: Unexpected error while using correct username and password.
- **Pass Criteria**: User is logged in successfully using correct credentials.

5.6.2 Test Case 2 – User/Admin Log out

- **Fail Criteria**: Unexpected error while logging out.
- **Pass Criteria**: User/Admin log out successfully.

5.6.3 Test Case 3 – Searching Content

- **Fail Criteria**: Existed content not found using search bar.
- **Pass Criteria**: Content related to searched text is listed on page.

5.6.4 Test Case 4 – Admin toolbar

• **Fail Criteria**: Drupal toolbar menu are not showing.

• **Pass Criteria**: Toolbar menu is shown with roles of logged in user.

5.7 Test Cases Result Summary

Table 5.1 Test case result summary

Test Case #	Description	Result
TC #1	User/Admin Login	Passed
TC #2	User/Admin Log out	Passed
TC #3	Searching Content	Passed
TC #4	Admin Toolbar	Passed

REFERENCES

eBooks:

1. Matthew David. “HTML5.” Routledge New York, Abstract (2013).
2. Mukesh Sharma. “Software Testing 2020.” Auerbach Publications, Abstract (2016).
3. Oliver Mesly. “Project Feasibility.” CRC Press (2017).
4. Macaulay, M. (2017). Introduction to Web Interaction Design: With HTML and CSS (1st ed.). Chapman and Hall/CRC.
5. Fajfar, I. (2015). Start Programming Using HTML, CSS, and JavaScript (1st ed.). Chapman and Hall/CRC. https://doi.org/10.1201/b19402
6. Wang, P.S. (2013). Dynamic Web Programming and HTML5 (1st ed.). Chapman and Hall/CRC.
7. Turner, W., & Leonard, S. (2017). JavaScript for Sound Artists: Learn to Code with the Web Audio API (1st ed.). Routledge. https://doi.org/10.1201/9781315659732

Journal Research Papers:

1. Arutyun I. Avetisyan. “Programming and Computer Software” Vol. 47, Issue 2 (2021).
2. Marek Rusinkiewicz, Yanchun Zhang. “World Wide Web” Vol. 23, Issue 2 (2021).

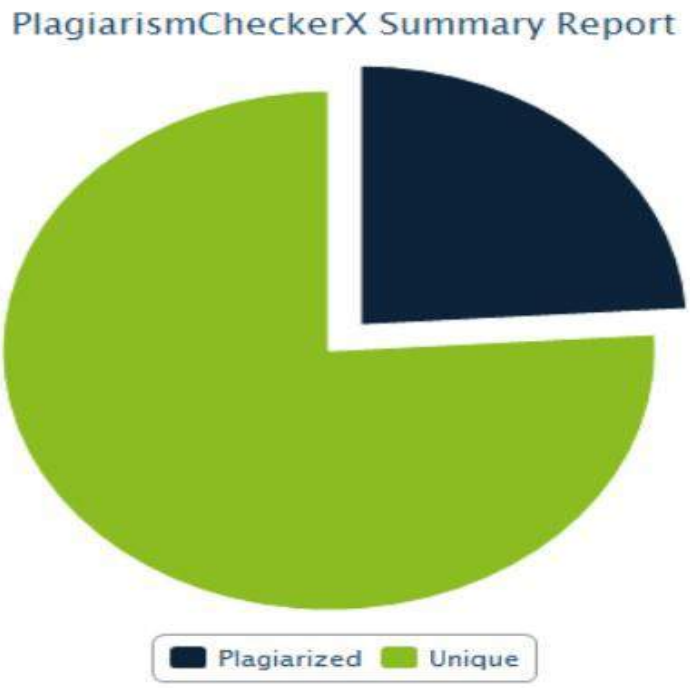
eBook Chapter:

1. Elvis C. Foster. “Software Economics” In: Software Engineering, Apress, Berkeley, pp 271-288 (2014).
2. Gerard O’Regan. “Test Case Analysis and Design”In: Concise Guide to Software Testing, Undergraduate Topics in Computer Science. Springer,Cham, pp 117-132 (2019).
3. Matthew David. “Picture CSS3.” In: HTML5, Chapter 1, Section 2, Routledge New York (2013).
4. Murali Chemuturi. “Product Architecture Design” In: Software Design, Chapter 8, Chapman and Hall/CRC (2018).
5. Preston Zhang. “Introduction to SQL and Relational Databases” In: Practical Guide for Oracle SQL, T-SQL and MySQL, Chapter 1,CRC Press (2017). 6. Ronald J. Leach. “Software Design” In: Introduction to Software Engineering, Chapter 4, Chapman and Hall/CRC (2018).
7. Rudiger Heimgartner. “User Interface Design” In: Intercultural User Interface Design.
8. Human–Computer Interaction Series. Springer, Cham, pp 121-166 (2019) 9. William Turner, Steve Leonard. “JavaScript for Sound Artist” Chapter 9, Routledge New York (2017).

Internet Resources:

1. https://react.js.org/
2. https://www.syncfusion.com/
3. https://www.youtube.com/watch?v=ZCPzScnc0SQ&t=616s
4. *https://www.w3schools.com/*
5. https://app.pluralsight.com/
   https://doi.org/10.1201/9781315692333
   https://doi.org/10.1201/b13928
   https://doi.org/10.1201/9781315659732

PLAGIARISM REPORT



Plagiarism Checker X Originality Report

Plagiarism Quantity: 24% Duplicate

Date	**Wednesday, June 23, 2021**
Words	**1421 Plagiarized Words/ Total 5923 Words.**
Sources	**More than 117 Sources Identified.**
Remarks	**Medium Plagiarism Detected- Your Document Motosacus Selective Improvement.**