

A Project Report on

**“BROADBAND CONNECTION PORTAL”**

Submitted in Partial Fulfillment of the Requirements For the award of the degree

### **Bachelor of Computer Applications**

**OF**

**BANGALORE UNIVERSITY**

**SUBMITTED BY**

**P.Sai darshan Reg No-17KUSB7058**

### **DEPARTMENT OF COMPUTER SCIENCE**

**Bachelor of Computer Applications**

### MS RAMAIAH COLLEGE Of ARTS, SCIENCE AND COMMERCE

M S R Nagar, Mathikere

**Bangalore-560 054**

### Screenshot (63)

### 

### **DEPARTMENT OF COMPUTER SCIENCE**

**Bachelor of Computer Applications**

**(Affiliated to Bangalore University)**

**M S R Nagar, Mathikere, Bangalore-560024**

## CERTIFICATE

#### This is to certify that P.SAI DARSHAN, has successfully completed the Fifth Semester BCA Project titled as “BROADBAND CONNECTION PORTAL” a partial fulfillment of the requirements for the award of the Degree in Bachelor of Computer Applications, Bangalore University, during the Academic Year 2019– 20.

**Project guide Head of Department**

(Department of computer Application)

**Dr. A.Malathi Palani**  **Dr. Prathiba V Kalburgi**

###### **Examiners**

1. -------------------------------- **Examination Center: ----------**

2. -------------------------------- **Date of the exam: ---------------**

# **ACKNOWLEDGEMENT**

The development of software is generally bit complex and time consuming task. The goal of developing the project **“BROADBAND CONNECTION PORTAL”** could not be archived without the encouragements of kindly helpful and supportive people. Here by we convey our sincere thanks for all of them.

I take this opportunity to express my gratitude to people who had been instrumental in the successful completion of this project.

I am thankful to our management trustee for providing us an opportunity to work and complete the project successfully.

I wish to express my thanks to our Principal for her support to the project work. I would like to acknowledge my gratitude to our HOD of Bachelor of Computer Applications **Dr. Prathiba V Kalburgi** for her encouragement and support. Without her encouragement and guidance this project would not have materialized.

The guidance and support received from our Internal

**Dr. A.Malathi Palani** who contributed to this project, was vital for the success of the project. We are grateful for their constant support and help.

# **SYNOPSIS**

This is a web application developed on the windows platform for requesting, managing a broadband connection and making corresponding bill payment for existing and new connection.

This system contains two modules, one being the admin and the other being the user.

1. The admin has the following functionalities:

* Approval of new connection
* Resolving support queries
* Managing client information

1. After login the user can make use of the following functions:

* Raising support queries
* Requesting new connection
* Making payment
* Managing personal information

This web application makes managing of a

broadband connection convenient and quick, because approval of new connection, resolving support queries, raising support queries, making payment, etc. is online.

The work of the service provider is made easier and it

amplifies the user experience for the client.

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| Serial No. | Name of Topic | Page No. |
| **1.** | **HARDWARE SPECIFICATION** | **1** |
| **2.** | **About the software** | **2** |
| **3.** | **System analysis** | **8** |
| **4.** | **Database Design** | **9** |
| **5.** | **Code** | **11** |
| **6.** | **ER-Diagram** | **81** |
| **7.** | **Data-flow diagram** | **82** |
| **8.** | **Screenshots** | **85** |
| **9.** | **Conclusion** | **105** |
| **10.** | **Bibliography** | **106** |

## HARDWARE SPECIFICATION

Processor : Intel Core 2 Duo 2.00 GHz processor

RAM : 1 GB

Hard Disk : 160 GB

Mother Board : Intel 945 chipset Mother Board

Keyboard : 116 Keys

Mouse : Optical scrolling Mouse

Monitor : 17” Monitor

**SOFTWARE SPECIFICATION**

Front End : HTML5,CSS3,Javascript

Server-Side Language : NodeJs

Data Base : MongoDB

**About The Software**

1. **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.

HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

**2. CSS:**

**Cascading Style Sheets** (**CSS**) is a style sheet language used for describing the presentation of a document written in a markup language such as 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 which reduces complexity and repetition in the structural content as well as enabling the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

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.

The name *cascading* comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS validation service for CSS documents.[[5]](https://en.wikipedia.org/wiki/CSS#cite_note-5)

In addition to HTML, other markup languages support the use of CSS including XHTML, plain XML, SVG, and XUL.

**3. Java Script:**

**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, prototype-based 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.

JavaScript engines were originally used only in web browsers, but they are now embedded in some servers, usually via Node.js. They are also embedded in a variety of applications created with frameworks such as Electron and Cordova.

Although there are similarities between JavaScript and Java, including language name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design.

**4. jQuery:**

jQuery is a javascript library designed to simplify HTML DOM tree traversal and manipulation, as well as event handling, CSS animation, and Ajax. It is free, open-source software using the permissive MIT License. As of May 2019, jQuery is used by 73% of the 10 million most popular websites. Web analysis indicates that it is the most widely deployed JavaScript library by a large margin, having at least 3 to 4 times more usage than any other JavaScript library.

jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery also provides capabilities for developers to create plug-ins on top of the JavaScript library. This enables developers to create abstractions for low-level interaction and animation, advanced effects and high-level, themeable widgets. The modular approach to the jQuery library allows the creation of powerful dynamic web pages and Web applications.

The set of jQuery core features—DOM element selections, traversal and manipulation—enabled by its *selector engine* (named "Sizzle" from v1.3), created a new "programming style", fusing algorithms and DOM data structures. This style influenced the architecture of other JavaScript frameworks like YUI v3 and Dojo, later stimulating the creation of the standard *Selectors API*. Later, this style has been enhanced with a deeper algorithm-data fusion in an heir of jQuery, the D3.js framework.

Microsoft and Nokia bundle jQuery on their platforms. Microsoft includes it with Visual Studio for use within Microsoft's ASP.NET AJAX and ASP.NET MVC frameworks while Nokia has integrated it into the Web Run-Time widget development platform.

**5.Nodejs:**

**Node.js** is an open-source, cross-platform, back end, JavaScript runtime environment that executes JavaScript code outside a web browser. Node.js lets developers use JavaScript to write command line tools and for server-side scripting—running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying web-application development around a single programming language, rather than different languages for server- and client-side scripts.

Though .js is the standard filename extension for JavaScript code, the name "Node.js" doesn't refer to a particular file in this context and is merely the name of the product. Node.js has an event-driven architecture capable of asynchronous I/O. These design choices aim to optimize throughput and scalability in web applications with many input/output operations, as well as for real-time Web applications (e.g., real-time communication programs and browser games).[[7]](https://en.wikipedia.org/wiki/Node.js#cite_note-readwrite-7)

The Node.js distributed development project was previously governed by the Node.js Foundation,] and has now merged with the JS Foundation to form the OpenJS Foundation, which is facilitated by the Linux Foundation's Collaborative Projects program.

Node.js allows the creation of Web servers and networking tools using JavaScript and a collection of "modules" that handle various core functionalities.Modules are provided for file system I/O, networking (DNS, HTTP, TCP, TLS/SSL, or UDP), binary data (buffers), cryptography functions, data streams, and other core functions.[31][45][47] Node.js's modules use an API designed to reduce the complexity of writing server applications.[31][45]

JavaScript is the only language that Node.js supports natively, but many compile-to-JS languages are available.As a result, Node.js applications can be written in CoffeeScript,Dart, TypeScript, ClojureScript and others.

**6.MongoDB:**

MongoDB is a cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas. MongoDB is developed by MongoDB Inc. and licensed under the Server Side Public License (SSPL).

MongoDB supports field, range query, and regular-expression searches.Queries can return specific fields of documents and also include user-defined JavaScript functions. Queries can also be configured to return a random sample of results of a given size.

Fields in a MongoDB document can be indexed with primary and secondary indices.

MongoDB provides high availability with replica sets.A replica set consists of two or more copies of the data. Each replica-set member may act in the role of primary or secondary replica at any time. All writes and reads are done on the primary replica by default. Secondary replicas maintain a copy of the data of the primary using built-in replication. When a primary replica fails, the replica set automatically conducts an election process to determine which secondary should become the primary. Secondaries can optionally serve read operations, but that data is only eventually consistent by default.

MongoDB scales horizontally using sharding.The user chooses a shard key, which determines how the data in a collection will be distributed. The data is split into ranges (based on the shard key) and distributed across multiple shards. (A shard is a master with one or more replicas.). Alternatively, the shard key can be hashed to map to a shard – enabling an even data distribution.

MongoDB can run over multiple servers, balancing the load or duplicating data to keep the system up and running in case of hardware failure.

# **SYSTEM ANALYSIS**

**EXISTING SYSTEM**

The existing system is not user friendly. The records are made in paper and the book can be managed by huge number of peoples. The records have to be stored in a very large spacious room.

Limitations of Existing system

* Time Consuming
* Human error is liable
* Difficulty faced at the time of organizing the whole data
* The records have to be recorded in notebook
* Notebooks can be altered by any persons at any time

**PROPOSED SYSTEM**

The proposed system is going to replace the existing system with its disadvantages removed along with improved performance. The proposed system is designed such that it will determine proper solution for the problems in the existing system. The proposed system is GUI (Graphical User Interface) oriented. It contains many database activities like adding user data, editing user data, viewing user data.

**Features of Proposed System**

* All user details are recorded in the database easily. The records can be managed by the administrator.
* The proposed system doesn’t need huge human power to manage the system.
* The reports can be easily searched using the proposed system

**DATABASE DESIGN**

**Database Name: Broadband Connection Portal**

### **Table Name: New Connection requests**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | Description |
| Name | String | Stores the username |
| Phone | Number | Stores the phone number |
| Address | String | Stores the address |
| Plan | String | Stores the plan |
| Email | String | Stores the email |
| Locality | String | Stores the locality |
| Lat | Number | Stores the latitude |
| Lng | Number | Stores the longitude |
| Date | Date | Stores the applied date |

### 

**Table Name:Payments**

|  |  |  |
| --- | --- | --- |
| Field Name | Data Type | **Description** |
| Payment\_id | String | Stores the payment id |
| Amount | Number | Stores the amount |
| Status | String | Stores the payment status |
| Method | String | Stores the payment method |
| Author | String | Stores the user |
| Purpose | String | Stores the payment purpose |

**Table Name: Plans**

|  |  |  |
| --- | --- | --- |
| Field Name | Data Type | Description |
| Name | String | Stores the plan name |
| Speed | String | Stores the speed |
| Data | String | Stores the data |
| Post\_Fup | String | Stores the post-fup |
| Rental | Number | Stores the monthly rental |

**Table Name: Router**

|  |  |  |
| --- | --- | --- |
| Field Name | Data Type | Description |
| Name | String | Stores the router name |
| Price | Number | Stores the router price |
| Image | String | Stores the image url |

**Table Name: Support**

|  |  |  |
| --- | --- | --- |
| Field Name | Data Type | Description |
| Type | String | Stores the query type |
| Comment | String | Stores the complaint comment |
| Status | String | Stores the query status |
| Date | Date | Stores the date |

**Table Name: Users**

|  |  |  |
| --- | --- | --- |
| Field Name | Data Type | Description |
| isConUser | Boolean | If user is a connection user |
| Username | String | Stores the username |
| Password | Hash | Stores the hash of the password |
| Address | String | Stores the address |
| Data | String | Stores the data remaining |
| Lat | Number | Stores the latitude |
| Lng | Number | Stores the longitude |
| Phone | Number | Stores the phone number |
| Plan | String | Stores the plan |
| Rental | Number | Stores the monthly rental |

## Code

## NodeJS

### **Server :**

require("dotenv").config();

const express = require('express');

const app = express();

const methodOverride=require("method-override");

const bodyParser= require('body-parser');

const mongoose = require('mongoose');

const CronJob=require("cron").CronJob;

const passport =require("passport");

const LocalStrategy=require("passport-local");

var FacebookStrategy = require('passport-facebook').Strategy;

const multer=require("multer");

const async=require("async");

const nodemailer=require("nodemailer");

const crypto=require("crypto");

const flash = require("connect-flash");

const Insta = require("instamojo-nodejs");

var Plan=require("./models/plans");

const Router=require("./models/routers");

const NewCon=require("./models/newconreq");

const NewRouter=require("./models/Nroutereq");

const User=require("./models/user");

const Support=require("./models/newsupport");

const NodeGeocoder = require('node-geocoder');

const moment = require("moment-timezone");

let Razorpay=require("razorpay");

const adminRoutes = require("./routes/adminroutes"),

cloudinaryRoute = require("./routes/cloudinary"),

cronjobs = require("./config/cronjobs"),

customerRoutes = require("./routes/customeroutes"),

homeRoutes = require("./routes/homeroutes"),

indexRoutes = require("./routes/index"),

newconRoutes = require("./routes/newconncetionroute"),

routerRoutes = require("./routes/routeroutes"),

supportRoutes = require("./routes/supportroutes");

cron = require("./config/cronjobs");

app.set('view engine', 'ejs');

app.use(bodyParser.urlencoded({extended:true}));

app.use(bodyParser.json());

app.use(express.static(\_\_dirname + "/public"));

app.locals.moment=moment;

moment.tz.setDefault("Asia/Kolkata");

app.use(methodOverride("\_method"));

app.use(flash());

//PASSPORT CONFIGURATION

app.use(require("express-session")({

secret:"CSGO is the best game",

resave:false,

saveUninitialized:false

}));

app.use(passport.initialize());

app.use(passport.session());

passport.use(new LocalStrategy(User.authenticate()));

passport.use(new LocalStrategy(

function(username, password, done) {

User.findOne({ username: username }, function (err, user) {

if (err) { return done(err); }

if (!user) {

return done(null, false, { message: 'Incorrect username or password' });

}

return done(null, user);

});

}

));

passport.serializeUser(function(user, done) {

done(null, user);

});

passport.deserializeUser(function(user, done) {

User.findById(user.\_id,function(err,user){

done(null, user);

})

});

app.use(function(req,res,next){

res.locals.currentUser=req.user;

res.locals.error=req.flash("error");

res.locals.success=req.flash("success");

next();

})

const mongoURI="mongodb+srv://saidarshan:R@mb02501@cluster0-wjhf4.mongodb.net/project?retryWrites=true&w=majority"

const promise=mongoose.connect(mongoURI, {

useNewUrlParser: true ,

useCreateIndex: true,

useFindAndModify: false,

useUnifiedTopology: true

},function(err){

if(err){

console.log(err);

}else{

console.log('Connected to DB');

}

})

app.use(adminRoutes);

app.use(cloudinaryRoute);

app.use(customerRoutes);

app.use(homeRoutes);

app.use(indexRoutes);

app.use(newconRoutes);

app.use(routerRoutes);

app.use(supportRoutes);

app.listen(process.env.PORT || 3000, process.env.IP, function() {

console.log('Project Started');

});

**ADMIN ROUTES:**

require("dotenv").config();

const express = require("express");

router = express.Router();

NodeGeocoder= require("node-geocoder");

moment = require("moment-timezone");

Support = require("../models/newsupport");

User = require("../models/user");

NewCon = require("../models/newconreq");

NewRouter= require("../models/Nroutereq");

middleware = require("../middleware/index");

var options = {

provider: 'google',

httpAdapter: 'https',

apiKey: process.env.GEOCODER\_API\_KEY,

formatter: null

};

var geocoder = NodeGeocoder(options);

//==========================

//ADMIN ROUTES

//==========================

//dashboard getting all counts

router.get("/adminlogin",function(req,res){

res.render("admin/adminLogin");

})

router.get("/admin",function(req,res){

Support.countDocuments({},function(err,supportcount){

if(err){

console.log(err);

}else{

NewCon.countDocuments({},function(err,concount){

if(err){

console.log(err);

}else{

User.find().where("isConUser").equals("true").exec(function(err,allusers){

const userscount=allusers.length;

res.render("admin/admin",{supportcount:supportcount,concount:concount,userscount:userscount});

})

}

})

}

})

})

//new connection adminside show

router.get("/admin/request/new-connection/:id",function(req,res){

NewCon.findById(req.params.id,function(err,foundnewcon){

if(err){

console.log(err);

}else{

res.render("admin/newconreqshow",{newcon:foundnewcon});

}

})

})

//index for newcon reqs

router.get("/admin/request/new-connection",function(req,res){

NewCon.find({},function(err,newcons){

if(err){

console.log(err);

}else{

res.render("admin/adminnewcon",{newcons:newcons})

}

})

})

//router admin(still pending)

router.get("/admin/request/new-router",function(req,res){

res.send("Routers admin");

})

//index for support tickets (admin)

router.get("/admin/support",function(req,res){

Support.find({},function(err,allsupport){

if(err){

console.log(err);

}else{

res.render("admin/adminsupport",{allsupport:allsupport});

}

})

})

//posting support to admin

router.post("/admin/support",function(req,res){

console.log(req.body);

const data={

type:req.body.type,

comment:req.body.comment,

status:"Pending",

date:moment().format(),

author: {

id:req.user.\_id,

name:req.user.name

}

}

Support.create(data,function(err,newsupport){

if(err){

console.log(err);

}else{

res.redirect("/home");

}

})

})

//show page for tickets admin

router.get("/admin/support/:id",function(req,res){

Support.findById(req.params.id,function(err,ticket){

if(err){

console.log(err);

}else{

res.render("admin/ticketshow",{ticket:ticket});

}

})

})

//update route for ticket status

router.put("/admin/support/:id",function(req,res){

Support.findByIdAndUpdate(req.params.id,{status:"Solved"},function(err,updatedticket){

if(err){

console.log(err);

}else{

res.redirect("/admin/support");

}

})

})

//posting router req to admin

router.post("/admin/request/new-router",middleware.isLoggedIn,function(req,res){

const data={

routername:req.body.routername,

price:req.body.price

}

NewRouter.create(data,function(err,newrouter){

if(err){

console.log(err);

}else{

console.log(newrouter);

res.redirect("/home");

}

})

})

//post route for approving new connection

router.post("/admin/request/new-connection",middleware.isLoggedIn,function(req,res){

var name=req.body.name;

var phone=req.body.phone;

var plan=req.body.plan;

var address=req.body.address;

var email=req.body.email;

var locality=req.body.locality;

var geoadress=address+","+locality;

var date=moment().format();

geocoder.geocode(geoadress,function(err,data){

if(err){

console.log(err);

}else{

const lat=data[0].latitude;

const lng=data[0].longitude;

const newcondata={

name:name,

phone:phone,

address:address,

plan:plan,

email:email,

locality:locality,

lat:lat,

lng:lng,

author:{

id:req.user.\_id,

name:req.user.name

},

date:date

}

NewCon.create(newcondata,function(err,newcon){

if(err){

console.log(err);

}else{

console.log(newcon);

res.redirect("/home");

}

})

}

})

})

module.exports=router;

**CUSTOMER ROUTES:**

require("dotenv").config();

const express = require("express");

router = express.Router();

User = require("../models/user");

Razorpay = require('razorpay');

CryptoJS = require('crypto-js');

sha256 = require('crypto-js/sha256'),

Payment = require('../models/payments'),

nodemailer = require("nodemailer");

middleware = require("../middleware");

const RazorpayConfig = {

key\_id: process.env.razorpay\_key\_id,

key\_secret: process.env.razorpay\_key\_secret

};

var instance = new Razorpay(RazorpayConfig);

//===================

//Customer routes

//===================

//index for viewing all users

router.get("/users", function (req, res) {

User.find().where("isConUser").equals("true").exec(function (err, allusers) {

if (err) {

console.log(err);

} else {

res.render("admin/user", { users: allusers })

}

})

})

router.get("/users/:id", function (req, res) {

User.findById(req.params.id, function (err, user) {

if (err) {

console.log(err);

} else {

res.render("admin/usershow", { user: user });

}

})

})

//payment init

router.post('/due/payment', middleware.isLoggedIn, function (req, res) {

const price = req.body.price + '00';

var options = {

amount: price, // amount in the smallest currency unit

currency: 'INR',

receipt: 'order\_rcptid\_11',

payment\_capture: '0'

};

instance.orders.create(options, function (err, order) {

console.log(order);

res.json(order);

});

});

//verifying and storing payment

router.post('/due/payment/success', function (req, res) {

const generated\_signature = CryptoJS.HmacSHA256(req.body.razorpay\_order\_id + '|' + req.body.razorpay\_payment\_id, process.env.razorpay\_key\_secret);

if (generated\_signature == req.body.razorpay\_signature) {

instance.payments.fetch(req.body.razorpay\_payment\_id).then((data) => {

const amount = data.amount;

sendmail(amount, req);

const DBdata = {

payment\_id: data.id,

amount: data.amount,

status: data.status,

method: data.method,

orderid: data.order\_id,

author: {

id: req.user.\_id,

name: req.user.name

},

purpose: data.description

}

Payment.create(DBdata, function (err, newpayment) {

if (err) {

console.log(err);

} else {

User.findByIdAndUpdate(req.user.\_id, { "rental.due": 0 }, function (err, updatedue) {

if (err) {

console.log(err);

}

});

console.log("payment data stored");

}

})

}).catch((error) => {

console.log(error);

})

console.log('payment successfull');

} else {

console.log("payment failure");

}

});

//my profile for user

router.get("/user/:id", middleware.isLoggedIn, function (req, res) {

if (req.xhr) {

User.findById(req.params.id, function (err, founduser) {

if (err) {

console.log(err);

} else {

res.json(founduser)

}

})

} else {

res.render("myprofile");

}

})

//editing route for myprofile user

router.put("/user/:id/edit", function (req, res) {

User.findByIdAndUpdate(req.params.id, req.body, { new: true }, function (err, updated) {

if (err) {

console.log(err);

} else {

res.json(updated);

}

})

})

//admin side updating(aprroval of newcon application) isConUser

router.put("/user/:id", function (req, res) {

let pack;

let rental;

const phone = req.body.phone;

const ad1 = req.body.address;

const ad2 = req.body.locality;

const address = ad1 + "," + ad2;

const plan = req.body.plan;

console.log(plan);

const lat = req.body.lat;

const lng = req.body.lng;

const isConUser = true;

switch (plan) {

case "ABP Blaze":

pack = "450 GB";

rental = 1059;

break;

case "ABP Swift":

pack = "200 GB";

rental = 685;

break;

case "ABP Lighting":

pack = "800 GB";

rental = 1399;

break;

case "ABP Rapid Plus":

pack = "350 GB";

rental = 959;

break;

case "ABP Storm":

pack = "650 GB";

rental = 1159;

break;

}

const data = {

phone: phone,

address: address,

plan: plan,

isConUser: isConUser,

data: {

total: pack,

used: 0

},

rental: {

total: rental,

due: rental

},

lat: lat,

lng: lng

}

User.findByIdAndUpdate(req.params.id, data, { new: true }, function (err, updateduser) {

if (err) {

console.log(err);

} else {

NewCon.deleteOne().where("author.id").equals(req.params.id).exec(function (err, deleted) {

if (err) {

console.log(err);

} else {

var smtpTransport = nodemailer.createTransport({

service: 'Gmail',

auth: {

user: 'actproject25@gmail.com',

pass: process.env.GMAILPW

}

});

var mailOptions = {

to: updateduser.email,

from: 'actproject25@gmail.com',

subject: 'New Connection Approval',

text: 'Your application for new connection has been aprroved \n\n' +

'Your choosen plan is:' + updateduser.plan + '\n' +

'Your monthly rental will be:' + updateduser.rental.total + '\n'

};

smtpTransport.sendMail(mailOptions, function (err) {

console.log('mail sent');

});

console.log(updateduser);

res.redirect("/admin");

}

})

}

})

})

function isAdmin(req, res, next) {

if (req.user.isAdmin == true) {

return next();

} else {

res.redirect("/home");

}

}

function sendmail(amount, req) {

const stramount = amount.toString();

const max = stramount.length;

const sliceamount = stramount.slice(0, max - 2);

var smtpTransport = nodemailer.createTransport({

service: 'Gmail',

auth: {

user: 'actproject25@gmail.com',

pass: process.env.GMAILPW

}

});

var mailOptions = {

to: req.user.email,

from: 'actproject25@gmail.com',

subject: 'Thank you for clearing your monthly due',

text: 'Hello ,\n\n' +

'We have succesfully receieved your payment of Rs.' + sliceamount + '\n\n' +

'This is your payment id:' + req.body.razorpay\_payment\_id

};

smtpTransport.sendMail(mailOptions, function (err) {

if (err) {

console.log(err);

} else {

console.log('mail sent');

}

});

}

module.exports = router;

**Authentication:**

require("dotenv").config();

const express = require("express"),

router = express.Router(),

User = require("../models/user"),

async = require("async"),

nodemailer=require("nodemailer"),

crypto = require("crypto"),

passport =require("passport"),

FacebookStrategy = require('passport-facebook').Strategy;

passport.use(new FacebookStrategy({

clientID: process.env.FACEBOOK\_APP\_ID,

clientSecret: process.env.FACEBOOK\_APP\_SECRET,

callbackURL: 'http://localhost:3000/auth/facebook/callback',

profileFields:["email","name","picture.type(large)"]

},

function(accessToken, refreshToken, profile, done) {

const picture = `https://graph.facebook.com/${profile.id}/picture?width=200&height=200&access\_token=${accessToken}`

User.findOne({'email':profile.\_json.email},function(err,user){

if(err){

done(err);

}else{

if(user){

return done(null,user);

}else{

const data={

email:profile.\_json.email,

name:profile.\_json.first\_name+profile.\_json.last\_name,

isFB:true,

profilepic:picture

}

User.create(data,function(err,newuser){

if(err){

console.log(err);

}else{

var smtpTransport = nodemailer.createTransport({

service: 'Gmail',

auth: {

user: 'actproject25@gmail.com',

pass: process.env.GMAILPW

}

});

var mailOptions = {

to:profile.\_json.email,

from: 'actproject25@gmail.com',

subject: 'Welcome to ACT',

text: 'Welcome to ACT.Thank you for signing up\n\n'

};

smtpTransport.sendMail(mailOptions, function(err) {

console.log('mail sent');

});

console.log(newuser);

return done(null,newuser);

}

})

}

}

})

}

));

//INDEX routes

router.get('/', function(req, res) {

res.redirect("/home");

});

//===================

//Auth Routes

//===================

router.get('/auth/facebook', passport.authenticate('facebook',{scope:['email','user\_photos']}));

router.get('/auth/facebook/callback',passport.authenticate('facebook',

{

successRedirect: '/home',

failureRedirect: '/login'

}

));

router.get("/register",function(req,res){

res.render("register");

})

//Sign up logic

router.post("/register",function(req,res,next){

const newUser= new User({

username:req.body.username,

name:req.body.name,

email:req.body.email,

});

User.register(newUser,req.body.password,function(err,user){

if(err){

console.log(err);

return res.render("register");

}else{

passport.authenticate("local")(req,res,function(){

req.flash("success","Successfully registered.Welcome " + req.user.name);

res.redirect("/home");

})

var smtpTransport = nodemailer.createTransport({

service: 'Gmail',

auth: {

user: 'actproject25@gmail.com',

pass: process.env.GMAILPW

}

});

var mailOptions = {

to: user.email,

from: 'actproject25@gmail.com',

subject: 'Welcome to ACT',

text: 'Welcome to ACT.Thank you for signing up\n\n' +

'These are your credentials:\n\n' +

'username:'+user.username+ '\n\n'

};

smtpTransport.sendMail(mailOptions, function(err) {

if(err){

console.log(err);

}else{

console.log('mail sent');

}

});

}

})

})

//isEmailAvailable

router.post("/register/isEmailAvailable",(req,res)=>{

const data=req.body;

User.findOne(data,function(err,user){

if(err){

console.log(err);

}else{

if(user){

res.json("");

}else{

res.json(true);

}

}

})

})

//isUsernameAvailable

router.post("/register/isUsernameAvailable",(req,res)=>{

const data=req.body;

User.findOne(data,function(err,user){

if(err){

console.log(err);

}else{

if(user){

res.json("");

}else{

res.json(true);

}

}

})

})

//show login form

router.get("/login",function(req,res){

res.render("login");

})

//handling login logic

router.post("/login",passport.authenticate("local",{

successRedirect:"/home",

failureRedirect:"/login",

failureFlash:true

}))

//logout route

router.get("/logout",function(req,res){

req.logOut();

req.flash("success","Logged you out");

res.redirect("/home");

})

router.get("/forgot",function(req,res){

res.render("forgot");

})

router.post('/forgot', function(req, res, next) {

async.waterfall([

function(done) {

crypto.randomBytes(20, function(err, buf) {

var token = buf.toString('hex');

done(err, token);

});

},

function(token, done) {

User.findOne({ email: req.body.email }, function(err, user) {

if (!user) {

req.flash('error', 'No account with that email address exists.');

return res.redirect('/forgot');

}

user.resetPasswordToken = token;

user.resetPasswordExpires = Date.now() + 3600000; // 1 hour

user.save(function(err) {

done(err, token, user);

});

});

},

function(token, user, done) {

var smtpTransport = nodemailer.createTransport({

service: 'Gmail',

auth: {

user: 'actproject25@gmail.com',

pass: process.env.GMAILPW

}

});

var mailOptions = {

to: user.email,

from: 'actproject25@gmail.com',

subject: 'ACT Password Reset',

text: 'You are receiving this because you (or someone else) have requested the reset of the password for your account.\n\n' +

'Please click on the following link, or paste this into your browser to complete the process:\n\n' +

'http://' + req.headers.host + '/reset/' + token + '\n\n' +

'If you did not request this, please ignore this email and your password will remain unchanged.\n'

};

smtpTransport.sendMail(mailOptions, function(err) {

console.log('mail sent');

req.flash('success', 'An e-mail has been sent to ' + user.email + ' with further instructions.');

done(err, 'done');

});

}

], function(err) {

if (err) return next(err);

res.redirect('/forgot');

});

});

router.get('/reset/:token', function(req, res) {

User.findOne({ resetPasswordToken: req.params.token, resetPasswordExpires: { $gt: Date.now() } }, function(err, user) {

if (!user) {

req.flash('error', 'Password reset token is invalid or has expired.');

return res.redirect('/forgot');

}

res.render('reset', {token: req.params.token});

});

});

router.post('/reset/:token', function(req, res) {

async.waterfall([

function(done) {

User.findOne({ resetPasswordToken: req.params.token, resetPasswordExpires: { $gt: Date.now() } }, function(err, user) {

if (!user) {

req.flash('error', 'Password reset token is invalid or has expired.');

return res.redirect('back');

}

if(req.body.password === req.body.confirm) {

user.setPassword(req.body.password, function(err) {

user.resetPasswordToken = undefined;

user.resetPasswordExpires = undefined;

user.save(function(err) {

req.logIn(user, function(err) {

done(err, user);

});

});

})

} else {

req.flash("error", "Passwords do not match.");

return res.redirect('back');

}

});

},

function(user, done) {

var smtpTransport = nodemailer.createTransport({

service: 'Gmail',

auth: {

user: 'actproject25@gmail.com',

pass: process.env.GMAILPW

}

});

var mailOptions = {

to: user.email,

from: 'actproject25@gmail.com',

subject: 'Your password has been changed',

text: 'Hello,\n\n' +

'This is a confirmation that the password for your account ' + user.email + ' has just been changed.\n'

};

smtpTransport.sendMail(mailOptions, function(err) {

req.flash('success', 'Success! Your password has been changed.');

done(err);

});

}

], function(err) {

res.redirect('/home');

});

});

module.exports=router;

**SUPPORT QUERY ROUTES:**

const express = require("express");

router = express.Router();

Support = require("../models/newsupport");

middleware = require("../middleware/index");

//============================

//SUPPORT routes

//============================

//user side viewing tickets

router.get("/support",function(req,res){

Support.find().where("author.id").equals(req.user.\_id).exec(function(err,allsupport){

if(err){

console.log(err);

}else{

if(req.xhr){

res.json(allsupport);

}else{

res.send("YOU CANT ACCESS IT LIKE THIS")

}

}

})

})

//creating new ticket userside

router.get("/support/new",middleware.isLoggedIn,function(req,res){

res.render("support");

})

//show for userside ticket

router.get("/support/:id",function(req,res){

Support.findById(req.params.id,function(err,ticket){

if(err){

console.log(err);

}else{

res.json(ticket);

}

})

})

router.put("/support/:id",function(req,res){

const data=req.body;

Support.findByIdAndUpdate(req.params.id,data,{new:true},function(err,updatedticket){

if(err){

console.log(err);

}else{

res.json(updatedticket);

}

})

})

//deleting tickets user side

router.delete("/support/:id",function(req,res){

Support.findByIdAndRemove(req.params.id,function(err,ticket){

if(err){

console.log(err);

}else{

res.json(ticket);

}

})

})

module.exports=router;

## HTML

**HOMEPAGE:**

<title>Home</title>

<%- include('partials/header.ejs'); -%>

</head>

<!--sliders-->

<div id="carouselExampleIndicators" class="carousel slide" data-ride="carousel">

<ol class="carousel-indicators">

<i class="fas fa-circle active test" data-target="#carouselExampleIndicators" data-slide-to="0"></i>

<i class="fas fa-circle test" data-target="#carouselExampleIndicators" data-slide-to="1"></i>

<i class="fas fa-circle test" data-target="#carouselExampleIndicators" data-slide-to="2"></i>

</ol>

<div class="carousel-inner">

<div class="carousel-item active">

<a href="#">

<img class="d-block w-100 h-55" src="images\Capture2.jpg" alt="First slide" href="#">

</a>

</div>

<div class="carousel-item">

<a href="#">

<img class="d-block w-100 h-55" src="images\Capture3.jpg" alt="Second slide" href="#">

</a>

</div>

<div class="carousel-item">

<a href="#">

<img class="d-block w-100 h-55" src="images\Capture11.jpg" alt="Third slide" href="#">

</a>

</div>

</div>

<a class="carousel-control-prev justify-content-start test1" href="#carouselExampleIndicators" role="button"

data-slide="prev">

<div class="slider-icon">

<i class="fas fa-chevron-left"></i>

</div>

</a>

<a class="carousel-control-next justify-content-end test1" href="#carouselExampleIndicators" role="button"

data-slide="next">

<div class="slider-icon">

<i class="fas fa-chevron-right"></i>

</div>

</a>

</div>

<!--body get started-->

<div class="container-fluid p-0">

<div class="get-started">

<h4>Get Started</h4>

<div class="controls">

<input type="text" name="" id="" placeholder="Enter your location"><button>Search</button>

</div>

</div>

</div>

</div>

<!--body broadband plans-->

<div class="container">

<h3 class="Bband">Broadband Plans</h3>

</div>

<!--plans-->

<div class="container">

<div class="row">

<div class="col-xl-4 col-lg-4">

<div class="card text-center">

<div class="card body">

<h4 class="py-3 px-3 text-center">ABP Swift</h4>

<p><i class="fas fa-bolt icon"></i>50Mbps Speed</p>

<p><i class="fas fa-wifi icon"></i>200 GB Data</p>

<p><i class="fas fa-tachometer-alt icon"></i>5 Mbps Post FUP</p>

</div>

</div>

</div>

<div class="col-xl-4 col-lg-4">

<div class="card text-center">

<div class="card body">

<h4 class="py-3 px-3 text-center">ABP Rapid Plus</h4>

<p><i class="fas fa-bolt icon"></i>150 Mbps Speed</p>

<p><i class="fas fa-wifi icon"></i>300 GB Data</p>

<p><i class="fas fa-tachometer-alt icon"></i>1 Mbps Post FUP</p>

</div>

</div>

</div>

<div class="col-xl-4 col-lg-4">

<div class="card text-center">

<div class="card body">

<h4 class="py-3 px-3 text-center">ABP Storm</h4>

<p><i class="fas fa-bolt icon"></i>150 Mbps Speed</p>

<p><i class="fas fa-wifi icon"></i>500 GB Data</p>

<p><i class="fas fa-tachometer-alt icon"></i>1 Mbps Post FUP</p>

</div>

</div>

</div>

</div>

</div>

<div class="text-center plansBtn">

<a href="/plans" class="btn btn-danger">View All Plans</a>

</div>

<h2 class="text-center Bband mt-5">Routers</h2>

<div class="container">

<div class="row">

<div class="col-xl-4 col-lg-4">

<div class="card">

<img

src="https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcT7ZAmUcKo6W4xTIHYVzvRCEGOg9M2CqGcKh0ZG8ro1iVySGM4J"

class="card-img-top" alt="...">

<div class="card-header text-center">

<h4>T1 Router</h4>

</div>

</div>

</div>

<div class="col-xl-4 col-lg-4">

<div class="card">

<img

src="https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcT7ZAmUcKo6W4xTIHYVzvRCEGOg9M2CqGcKh0ZG8ro1iVySGM4J"

class="card-img-top" alt="...">

<div class="card-header text-center">

<h4>T1 Router</h4>

</div>

</div>

</div>

<div class="col-xl-4 col-lg-4">

<div class="card">

<img

src="https://encrypted-tbn0.gstatic.com/images?q=tbn%3AANd9GcT7ZAmUcKo6W4xTIHYVzvRCEGOg9M2CqGcKh0ZG8ro1iVySGM4J"

class="card-img-top" alt="...">

<div class="card-header text-center">

<h4>T1 Router</h4>

</div>

</div>

</div>

</div>

</div>

<div class="text-center mt-3"><a href="/routers" class="btn btn-danger">View all routers</a></div>

<section class="whatwedo mt-4">

<h4 class="text-center mb-2" style="letter-spacing: 1px;font-weight: 400;text-transform: uppercase">What Service Do We

Offer ?</h4>

<hr id="hr" class="mb-3">

<p class="text-center"><strong>We offer 3 kinds of services</strong></p>

<p>

<h4 class="text-center">GOOD-CHEAP-FAST</h4>

</p>

<div class="container" style="max-width: 1800px;">

<div class="row">

<div class="card col-md-4">

<div class="card-body text-center">

<img src="/svgs/goodsvg.svg" alt="" style="width: 100%;height: max-content;">

<h4 style="color: rgb(108, 101, 124);">GOOD AND CHEAP WON'T BE FAST</h4>

</div>

</div>

<div class="card col-md-4">

<div class="card-body text-center">

<img src="/svgs/fastsvg.svg" alt="" style="width: 100%;height: max-content;">

<h4 style="color: rgb(108, 101, 124);">FAST AND GOOD WON'T BE CHEAP</h4>

</div>

</div>

<div class="card col-md-4">

<div class="card-body text-center">

<img src="/svgs/cheapsvg.svg" alt="" style="width: 100%;height: max-content;">

<h4 style="color: rgb(108, 101, 124);">CHEAP AND FAST WON'T BE GOOD</h4>

</div>

</div>

</div>

</div>

</section>

<div class="container mt-5 faq">

<h4>

FAQ

</h4>

<div class="row">

<div class="col-xl-12">

<div class="card">

<div class="card-title faqtitle">

<h5>How to apply for a connection ?</h5>

<button class="faq-open"><i class="fas fa-plus faq-plus "></i></button>

</div>

<div class="faq-answer" style="display: none;">

<p>Please sign-up with your name and email id .After signing up please navigate to "New Connection" and fill

in all

the details,and our team will get in contact with you</p>

</div>

</div>

</div>

<div class="col-xl-12">

<div class="card">

<div class="card-title faqtitle">

<h5>How to pay monthly due ?</h5>

<button class="faq-open"><i class="fas fa-plus faq-plus"></i></button>

</div>

<div class="faq-answer" style="display: none;">

<p>Please navigate to your profile by clicking on your name in the right-side top corner.If your monthly due

isn't

cleared.You can view your monthly due inside your profile.Payment can be made through our payment partner

RazorPay

with thier multi-featured payment methods</p>

</div>

</div>

</div>

<div class="col-xl-12">

<div class="card">

<div class="card-title faqtitle">

<h5>How to apply for two connections ?</h5>

<button class="faq-open"><i class="fas fa-plus faq-plus "></i></button>

</div>

<div class="faq-answer" style="display: none;">

<p>At the moment we only provide one connection per user.</p>

</div>

</div>

</div>

<div class="col-xl-12">

<div class="card">

<div class="card-title faqtitle">

<h5>How to raise a support ticket ?</h5>

<button class="faq-open"><i class="fas fa-plus faq-plus "></i></button>

</div>

<div class="faq-answer" style="display: none;">

<p>Navigate to 'support' in the navbar and select the issue and please provide extra information in the

description

box.</p>

</div>

</div>

</div>

</div>

</div>

<footer class="pagefooter">

<i class="far fa-copyright mr-2"></i>Created by SAI DARSHAN.

</footer>

<%- include('partials/footer.ejs'); -%>

**LOGIN:**

<%- include("partials/header.ejs"); %>

<title>Login</title>

</head>

<div class="container" style="margin-top: 10rem;">

<div class="row justify-content-center">

<form action="/login" method="POST" class="realoginform col-md-10" >

<h4 class="plansH text-center mb-3">Login</h4>

<div class="form-group col-md-12">

<input class="form-control username" type="text" name="username" placeholder="Username" autocomplete="off">

</div>

<div class="form-group col-md-12">

<input class="form-control loginpassword" type="password" name="password" placeholder="Password">

</div>

<div class="form-group col-md-12">

<input type="submit" class="btn btn-outline-danger btn-block" value="Login"></button>

</div>

<div class="d-flex justify-content-between mt-3 forgot">

<a href="/forgot">Forgot Password ?</a>

<a href="/register">Don't have a account yet?</a>

</div>

<hr>

<label class="text-center loginor">OR</label>

<div class="fb mt-2">

<a href="/auth/facebook" class="btn btn-block btn-primary" style="font-family: 'Baloo 2', cursive;font-weight: 700"><i class="fab fa-facebook-square mr-2"></i>Log in with Facebook</a>

</div>

</form>

</div>

</div>

<%- include("partials/footer.ejs"); %>

**SIGNUP**

<%- include("partials/header.ejs"); %>

<title>Register</title>

</head>

<div class="container">

<div class="row justify-content-center" style="height: 80vh;margin-top: 15%;">

<div class="col-md-6">

<form action="/register" method="POST" class="signupform">

<div class="form-group">

<h4 class="text-center plansH">Sign Up</h4>

</div>

<div class="form-group">

<input name="name" class="form-control signupname" type="text" placeholder="name">

</div>

<div class="form-group">

<input name="email" id="email" class="form-control signupemail" type="email" placeholder="Enter your e-mail id" autocomplete="off">

</div>

<div class="form-group">

<input name="username" id="username" class="form-control signupusername" type="text" placeholder="Username" autocomplete="off">

</div>

<div class="form-group">

<input name="password" class="form-control signuppassword" type="password" placeholder="Password" autocomplete="off">

</div>

<button class="btn btn-outline-danger btn-block mr-1" href="">Sign Up</button>

</form>

</div>

</div>

</div>

<%- include("partials/footer.ejs"); %>

## CSS

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

outline: none;

}

.navbar{

padding:2rem;

}

.Bband {

color: red;

margin: 20px 20px;

text-align: center;

font-weight: 700;

}

.iconer {

margin-bottom: 2rem;

}

ul {

list-style: none;

margin: 0;

}

h4 {

font-weight: bolder;

}

h1 {

font-weight: 700;

}

.plansH {

color: #ea1c2c;

}

.profilecard .card-img-top {

height: 250px;

width: 100%;

border-radius: 70%;

border: 1px solid #6c757d33;

}

.profilecard {

padding: 20px 25px;

width: 20em;

}

.profilecard .card-header {

border: none;

}

.editbtn {

position: absolute;

}

.showplan {

height: 70vh;

display: flex;

align-items: center;

}

.showplan a {

color: black;

}

.showplan1 a:hover {

color: black;

}

.list-group-item a:hover {

color: #ea1c2c;

}

.nav-tabs .nav-item.show .nav-link,

.nav-tabs .nav-link.active {

color: white;

background-color: #ea1c2c;

}

.displaymonth {

display: none;

}

.routershow {

padding: 5px 50px;

align-items: baseline;

display: flex;

justify-content: space-around;

height: 80vh;

}

svg {

pointer-events: none;

}

.routerlist li {

margin-bottom: 20px;

}

.routerlist {

margin-left: 20px;

list-style-type: disc;

}

.routerbuynw {

position: absolute;

align-self: flex-end;

justify-self: flex-end;

}

.routername h4 {

color: #ea1c2c;

}

.newconreqcontainer {

padding-top: 50px;

}

.newconreqcontainer h3 {

font-size: 1.35rem;

}

.newconcard {

margin-bottom: 30px;

border-left: 5px solid #ea1c2c;

padding: 10px 30px;

}

.newconfooter {

display: flex;

justify-content: space-between;

padding: .75rem .25rem;

font-size: .9rem;

font-family: 'Baloo 2', cursive;

font-weight: 700;

}

.newconfooter .time {

color: #707070;

}

.newconreqcontainer {

padding-top: 50px;

}

.list-group-item.active {

z-index: 2;

color: #fff;

background-color: #ea1c2c;

border-color: #ea1c2c;

}

.list-group-item {

cursor: pointer;

}

.inputs input[type="text"] {

width: 70%;

}

#map {

height: 400px;

width: 100%;

}

.inputs {

padding-top: 50px;

}

.AceeptDeclineBTN {

margin-top: 20px;

display: flex;

padding-left: 200px;

}

.Rimage {

margin-top: 50px;

height: 40vh;

}

#v-pills-home-tab {

height: 3rem;

display: flex;

align-items: center;

}

.planselect {

background-color: #ea1c2c;

}

a:hover {

text-decoration: none;

}

.icon {

margin-right: 10px;

color: #ea1c2c;

padding-bottom: 20px;

font-size: large;

}

#hr {

margin: 0 auto;

width: 20%;

height: 1;

background: #ea1c2c;

}

#hr1 {

margin-left: 21em;

width: 20%;

height: 1;

background: #ea1c2c;

}

.plansD {

color: #ea1c2c;

margin-left: 15em;

}

.dashcard {

margin-top: 40px;

height: 20em;

}

.dashcard .card-body h4 {

font-size: 3rem;

}

.dashcontainer {

margin-left: 25em;

max-width: 100em;

}

.stretched-link::after {

position: absolute;

top: 0;

right: 0;

bottom: 0;

left: 0;

z-index: 1;

pointer-events: auto;

content: "";

background-color: rgba(0, 0, 0, 0);

}

.svg {

margin-left: 4rem;

margin-top: 0;

width: 104em;

display: flex;

justify-content: space-between;

}

.card-body {

padding: 10px 25px;

}

.card {

filter: drop-shadow(1px 1px 3px rgba(0, 0, 0, 0.521));

background-color: white;

}

.card-header {

background-color: white;

}

.card-footer {

background-color: white;

}

.statuscard {

display: none;

margin-left: 45.5em;

position: absolute;

margin-top: 50px;

}

.statuscard .card-title {

padding: 10px 10px;

text-align: center;

margin-bottom: 0;

}

## Javascript

moment.tz.setDefault('Asia/Kolkata');

var count = 0;

const tickets = $('.adminsupporticket');

$(function($) {

$('.adminsupporticket').each(function(index) {

if ($(this).find('.status').text() == 'Pending') {

$(this).addClass('pending');

$(this).find('.status').addClass('colorP');

} else {

$(this).addClass('solved');

$(this).find('.status').addClass('colorS');

}

});

});

$('#threemonth').on('click', function() {

$('#buy3').removeClass('displaymonth');

$('#buy1').addClass('displaymonth');

$('#buy6').addClass('displaymonth');

});

$('#sixmonth').on('click', function() {

$('#buy6').removeClass('displaymonth');

$('#buy3').addClass('displaymonth');

$('#buy1').addClass('displaymonth');

});

$('#onemonth').on('click', function() {

$('#buy1').removeClass('displaymonth');

$('#buy3').addClass('displaymonth');

$('#buy6').addClass('displaymonth');

});

$('.supportform').on('submit', function(e) {

e.preventDefault();

if ($(this).serialize().length > 15) {

$('#exampleModal').modal('show');

var type = $('#supporttype').val();

var comment = $('#supporttxtA').val();

var data = { type: type, comment: comment };

$.ajax({

method: 'POST',

url: '/admin/support',

data: data

});

}

});

$('.newconform').on('submit', function(e) {

if (

$('.reqfield').filter(function() {

return this.value === '';

}).length === 0

) {

e.preventDefault();

$('#exampleModal').modal('show');

const name = $('#newconname').val();

const phone = $('#newconphone').val();

const plan = $('#newconplan').val();

const address = $('#newconaddress').val();

const locality = $('#newconlocality').val();

const email = $('#newconemail').val();

const data = { name: name, phone: phone, plan: plan, address: address, locality: locality, email: email };

$.ajax({

method: 'POST',

url: '/admin/request/new-connection',

data: data

});

} else {

e.preventDefault();

}

// console.log($( this ).serialize().length);

// if($(".reqfield").each().val!=0){

// $('#exampleModal').modal('show');

// const name=$("#newconname").val();

// const phone=$("#newconphone").val();

// const plan=$("#newconplan").val();

// const address=$("#newconaddress").val();

// const locality=$("#newconlocality").val();

// const email=$("#newconemail").val();

// const data={name:name,phone:phone,plan:plan,address:address,locality:locality,email:email}

// $.ajax({

// method:"POST",

// url:"/admin/request/new-connection",

// data:data

// })

// }

});

$('#Viewticketsbtn').on('click', function() {

$('.ticketcard').show('slow');

$(this).addClass('active');

$(this).siblings('.active').removeClass('active');

$('.supportform').hide('slow');

$('.userviewticket').hide('slow');

$('#pending').removeClass('active');

$('#solved').removeClass('active');

$('#all').addClass('active');

$.get('/support', function(data) {

console.log(data);

data.forEach(function(ticket) {

const time = moment.tz(ticket.date, 'Asia/Kolkata');

const difftime = moment(time).fromNow();

if (count == 0) {

$('.ticketcontainer').append(

`

<div class="card ticketcard" style="width: 90%;">

<div class="card-header ticketheader">

<h5>${ticket.type}</h5>

<span class="optionsbtn">

<i class="fas fa-ellipsis-v"></i>

<label class="deletelbl" href="/support/${ticket.\_id}"><span style="font-size:.7rem"><i class="far fa-trash-alt mr-2" ></i>Delete</span></lablel>

</span>

</div>

<div class="card-footer ticketfooter">

<p class="mb-3 ">

<span class="status">${ticket.status}</span>

<span class="ml-3 time"><i class="fas fa-clock mr-1"></i> ${difftime}</span>

</p>

<a href="/support/${ticket.\_id}" class="btn btn-outline-danger btn-sm viewticketbtn">View</a>

</div>

</div>

`

);

}

$('.status').each(function(index) {

if ($(this).text() == 'Pending') {

$(this).addClass('colorP');

$(this).parents('.ticketcard').addClass('pending');

} else {

$(this).addClass('colorS');

$(this).parents('.ticketcard').addClass('solved');

}

});

});

count += +1;

usertickets = $('.ticketcard');

if (usertickets.length == 0) {

$('.userticketfilter').hide();

$('.nothingtoshow').show('slow');

} else {

$('.userticketfilter').show('slow');

}

});

});

$('#Raiseticketbtn').on('click', function() {

$('.nothingtoshow').hide('slow');

$(this).addClass('active');

$(this).siblings('.active').removeClass('active');

$('.ticketcard').hide('slow');

$('.supportform').show('slow');

$('.userviewticket').hide('slow');

$('.userticketfilter').hide('slow');

});

$('.ticketcontainer').on('click', '.viewticketbtn', function(e) {

count = 0;

e.preventDefault();

$('.statuscard').fadeOut();

$('.userticketfilter').hide('slow');

const url = $(this).attr('href');

if (count == 0) {

$.get(url, function(data) {

$('.ticketcard').hide('slow');

const time = moment(data.date).format('DD.MM.YYYY HH:mm');

const type = data.type;

const comment = data.comment;

const id = data.\_id;

const status = data.status;

$('.ticketcontainer').append(

`

<div class="col-md-12 userviewticket">

<div class="col-md-5 mr-5">

<form url="/support/${id}">

<label style="width: 50em;display: flex;justify-content: space-between;">

<strong>Ticket Refrence Number</strong>

<span class="status">${status}</span>

</label>

<div class="input-group mb-3" style="width:50em">

<input type="text" class="form-control ticketinput" value="${id}" name="id" disabled="true">

</div>

<label><strong>Type</strong></label>

<div class="input-group mb-3" style="width:50em">

<input type="text" class="form-control ticketinput" value="${type}" name="type" disabled="true">

</div>

<label><strong>Comment</strong></label>

<div class="input-group mb-3" style="width:50em">

<input type="text" class="form-control ticketinput" value="${comment}" name="comment" disabled="true">

<div class="input-group-append">

<button class="btn btn-outline-secondary editdonebtn" type="button" style="display: none;"><i class="fas fa-check"></i></button>

<button class="btn btn-outline-secondary editcancelbtn" type="button" style="display: none;"><i class="fas fa-times"></i></button>

<button class="btn btn-outline-secondary profileditbtn" type="button"><i class="fas fa-pencil-alt"></i></button>

</div>

</div>

<ul class="navbar-nav">

<li class="nav-item">

<a class="nav-link gobackbtn">Go back</a>

</li>

<ul>

</div>

`

);

$('.status').each(function(index) {

if ($(this).text() == 'Pending') {

$('.profileditbtn').show();

$(this).addClass('colorP');

} else {

$('.profileditbtn').hide();

$('.profileditbtn').parents().siblings('.form-control').prop('disabled', true);

$(this).addClass('colorS');

}

});

});

count += +1;

}

});

$('#pending').on('click', function() {

$('.nothingtoshow').hide('slow');

$('#all').removeClass('active');

$('#solved').removeClass('active');

$(this).addClass('active');

const userPendingTickets = usertickets.filter('.pending');

if (userPendingTickets.length == 0) {

$('.nothingtoshow').show('slow');

} else {

userPendingTickets.show('slow');

}

usertickets.not('.pending').hide('slow');

});

$('#adminpending').on('click', function() {

$('.nothingtoshow').hide('slow');

$('#adminall').removeClass('active');

$('#adminsolved').removeClass('active');

$(this).addClass('active');

const adminpending = tickets.filter('.pending');

if (adminpending.length == 0) {

$('.nothingtoshow').show('slow');

}

tickets.not('.pending').hide();

tickets.filter('.pending').show();

});

$('#adminsolved').on('click', function() {

$('.nothingtoshow').hide('slow');

$('#adminall').removeClass('active');

$('#adminpending').removeClass('active');

$(this).addClass('active');

const adminsolved = tickets.not('.pending');

if (adminsolved.length == 0) {

$('.nothingtoshow').show('slow');

}

tickets.filter('.pending').hide('slow');

tickets.not('.pending').show('slow');

});

$('#adminall').on('click', function() {

$('.nothingtoshow').hide('slow');

$('#adminsolved').removeClass('active');

$('#adminpending').removeClass('active');

$(this).addClass('active');

tickets.show('slow');

});

$('#solved').on('click', function() {

$('#pending').removeClass('active');

$('#all').removeClass('active');

$(this).addClass('active');

const adminsolvedtickets = tickets.not('.pending');

if (adminsolvedtickets.length == 0) {

$('.nothingtoshow').show('slow');

} else {

adminsolvedtickets.show('slow');

tickets.filter('.pending').hide('slow');

}

const userSolvedTickets = usertickets.not('.pending');

if (userSolvedTickets.length == 0) {

$('.nothingtoshow').show('slow');

} else {

userSolvedTickets.show('slow');

$('.nothingtoshow').hide();

}

usertickets.filter('.pending').hide('slow');

});

$('#all').on('click', function() {

$('.nothingtoshow').hide();

$('#pending').removeClass('active');

$('#solved').removeClass('active');

$('#all').addClass('active');

usertickets.show('slow');

});

$('.ticketcontainer').on('click', '.gobackbtn', function() {

$('.userviewticket').hide('slow');

$('.ticketcard').show('show');

$('.userticketfilter').show('slow');

$('#pending').removeClass('active');

$('#solved').removeClass('active');

$('#all').addClass('active');

});

$('.ticketcontainer').on('click', '.optionsbtn', function() {

const kyabre = $(this).find('.deletelbl');

kyabre.toggle('slow');

});

$('.ticketcontainer').on('click', '.deletelbl', function() {

const url = $(this).attr('href');

const parent = $(this).parents('.ticketcard');

$.ajax({

method: 'DELETE',

url: url,

success: function(data) {

parent.remove();

const parentafter = $('.ticketcard');

if (parentafter.length == 0) {

$('.nothingtoshow').show('slow');

}

}

});

});

$('.ticketcontainer').on('click', '.profileditbtn', function() {

$(this).siblings().toggle('slow');

$(this).parent().siblings('.form-control').prop('disabled', false);

$(this).parent().siblings('.form-control').focus();

});

$('.ticketcontainer').on('click', '.editcancelbtn', function() {

const input = $(this).parents().siblings('.ticketinput');

input.focusout();

input.prop('disabled', true);

$(this).hide('slow');

$(this).siblings('.editdonebtn').hide('slow');

});

$('.ticketcontainer').on('click', '.editdonebtn', function() {

$(this).hide('slow');

$(this).siblings('.editcancelbtn').hide('slow');

const input = $(this).parents().siblings('.ticketinput');

input.focusout();

input.prop('disabled', true);

const key = input.attr('name');

const data = { [key]: input.val() };

const url = $(this).parents('form').attr('url');

$originalItem = input;

$.ajax({

method: 'PUT',

url: url,

data: data,

originalItem: $originalItem,

success: function(data) {

const editdata = data[key];

this.originalItem.val(`${editdata}`);

}

});

input.css('border', '3px solid green');

setTimeout(function() {

input.css('border', '1px solid #ced4da');

input.prop('disabled', true);

}, 1500);

});

$('.profileditbtn').click(function() {

$(this).siblings().toggle('slow');

$(this).parent().siblings('.form-control').prop('disabled', false);

$(this).parent().siblings('.form-control').focus();

});

$('.editdonebtn').click(function() {

$(this).hide('slow');

$(this).siblings('.editcancelbtn').hide('slow');

const input = $(this).parents().siblings('.profileinput');

input.focusout();

const key = input.attr('name');

const data = { [key]: input.val() };

const url = $(this).parents('form').attr('url');

console.log(url);

$originalItem = input;

$.ajax({

method: 'PUT',

url: url,

data: data,

originalItem: $originalItem,

success: function(data) {

const editdata = data[key];

if (key == 'name') {

$('.profilecardname').text(`${editdata}`);

}

this.originalItem.val(`${editdata}`);

}

});

input.css('border', '3px solid green');

$(this).parents('.input-group').siblings().show('slow').delay(1000).hide('slow');

setTimeout(function() {

input.css('border', '1px solid #ced4da');

input.prop('disabled', true);

}, 1500);

});

$('.editcancelbtn').click(function() {

const input = $(this).parents().siblings('.profileinput');

input.focusout();

input.prop('disabled', true);

$(this).hide('slow');

$(this).siblings('.editdonebtn').hide('slow');

});

$('.cameraicon').click(function() {

$('.fielupload').toggle('slow');

});

$('.custom-file-input').change(function() {

var file = $('.custom-file-input')[0].files[0].name;

$('.custom-file-label').text(file);

});

$('.ticketuserinfo').click(function() {

$(this).addClass('active');

$(this).siblings().removeClass('active');

$('.adminticketform').hide('slow');

$('.personalsupportinfo').show('slow');

const url = $('.adminticketform').attr('url');

if (count === 0) {

$.get(url, function(data) {

console.log(data);

$('.realticket').append(`

<div class="personalsupportinfo">

<div class="input-group">

<label class="activeuser d-flex justify-content-between" style="width: 50em;">

<strong>Name</strong>

<span class="isconuser " style=" font-size: .9rem;font-family: 'Baloo 2',cursive;font-weight: 700;color:green">

Active Connection User

</span>

</label>

</div>

<label class="form-control mb-3">${data.name}</label>

<div class="input-group">

<label><strong>Plan</strong></label>

</div>

<label class="form-control mb-3">${data.plan}</label>

<div class="input-group">

<label><strong>E-mail ID</strong></label>

</div>

<label class="form-control mb-3">${data.email}</label>

<div class="input-group">

<label><strong>Phone</strong></label>

</div>

<label class="form-control mb-3">${data.phone}</label>

<div class="input-group">

<label><strong>Address</strong></label>

</div>

<textarea class="form-control mb-3">${data.address}</textarea>

</div>

`);

});

count += +1;

}

});

$('.ticketinfo').click(function() {

$(this).addClass('active');

$(this).siblings().removeClass('active');

$('.adminticketform').show('slow');

$('.personalsupportinfo').hide('slow');

});

$('.faq-open').click(function() {

if ($(this).parent('.faqtitle').siblings('.faq-answer').hasClass('opened')) {

$(this).parent('.faqtitle').siblings('.faq-answer').removeClass('opened');

$(this).parent('.faqtitle').siblings('.faq-answer').fadeOut('slow');

$(this).html(`<i class="fas fa-plus">`);

} else {

$(this).html(`<i class="fas fa-minus">`);

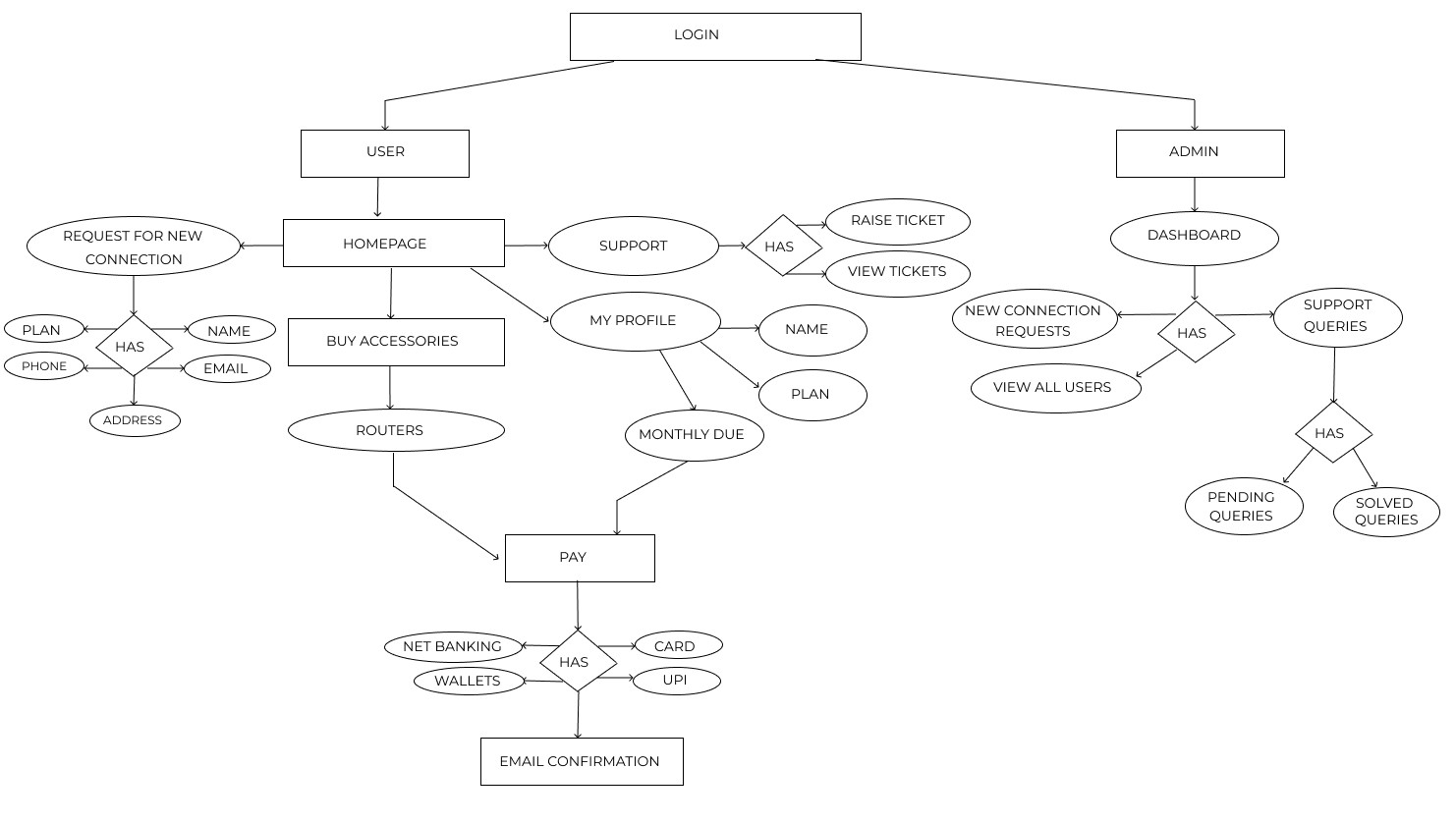
$(this).parent('.faqtitle').siblings('.faq-answer').fadeIn('slow');

$(this).parent('.faqtitle').siblings('.faq-answer').addClass('opened');

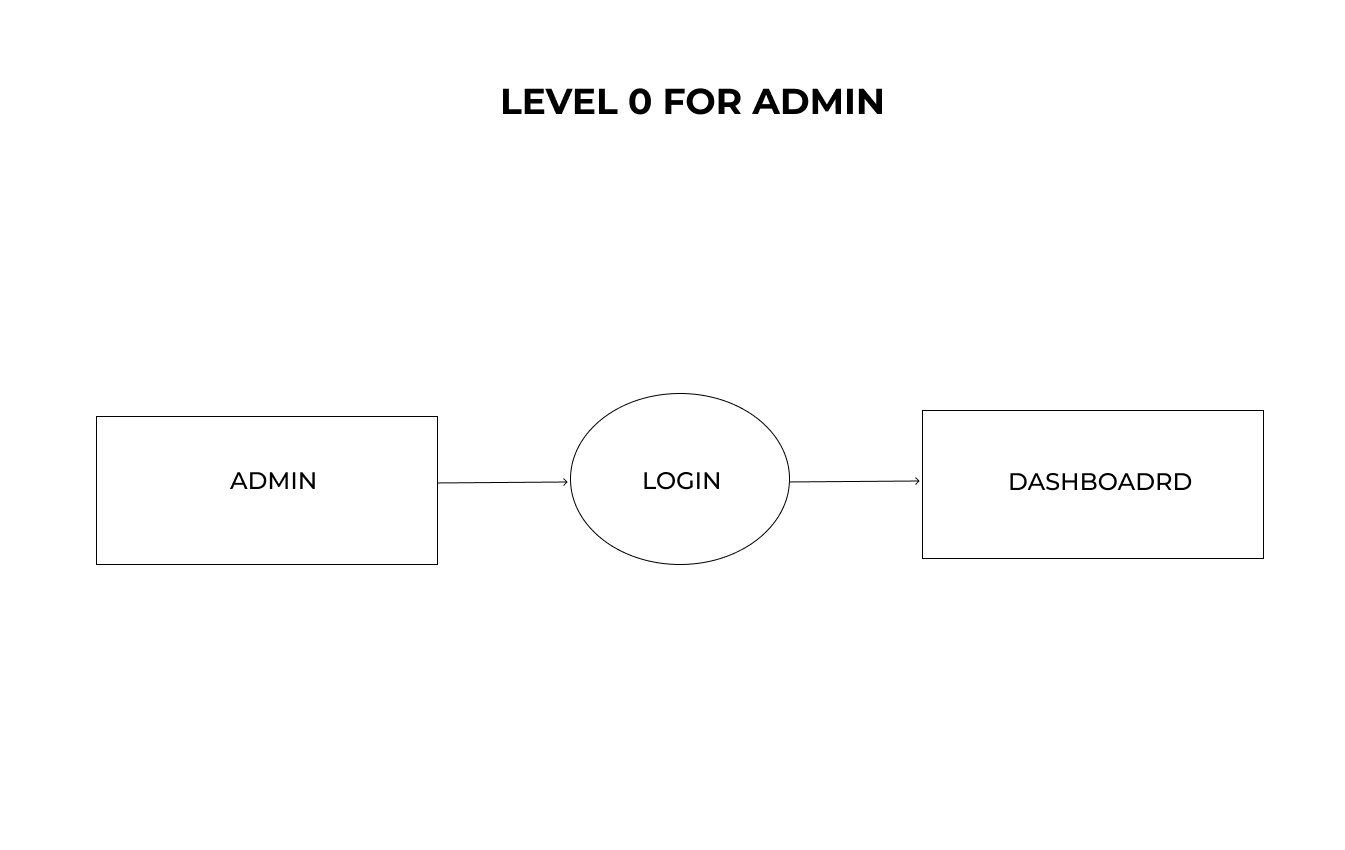
}

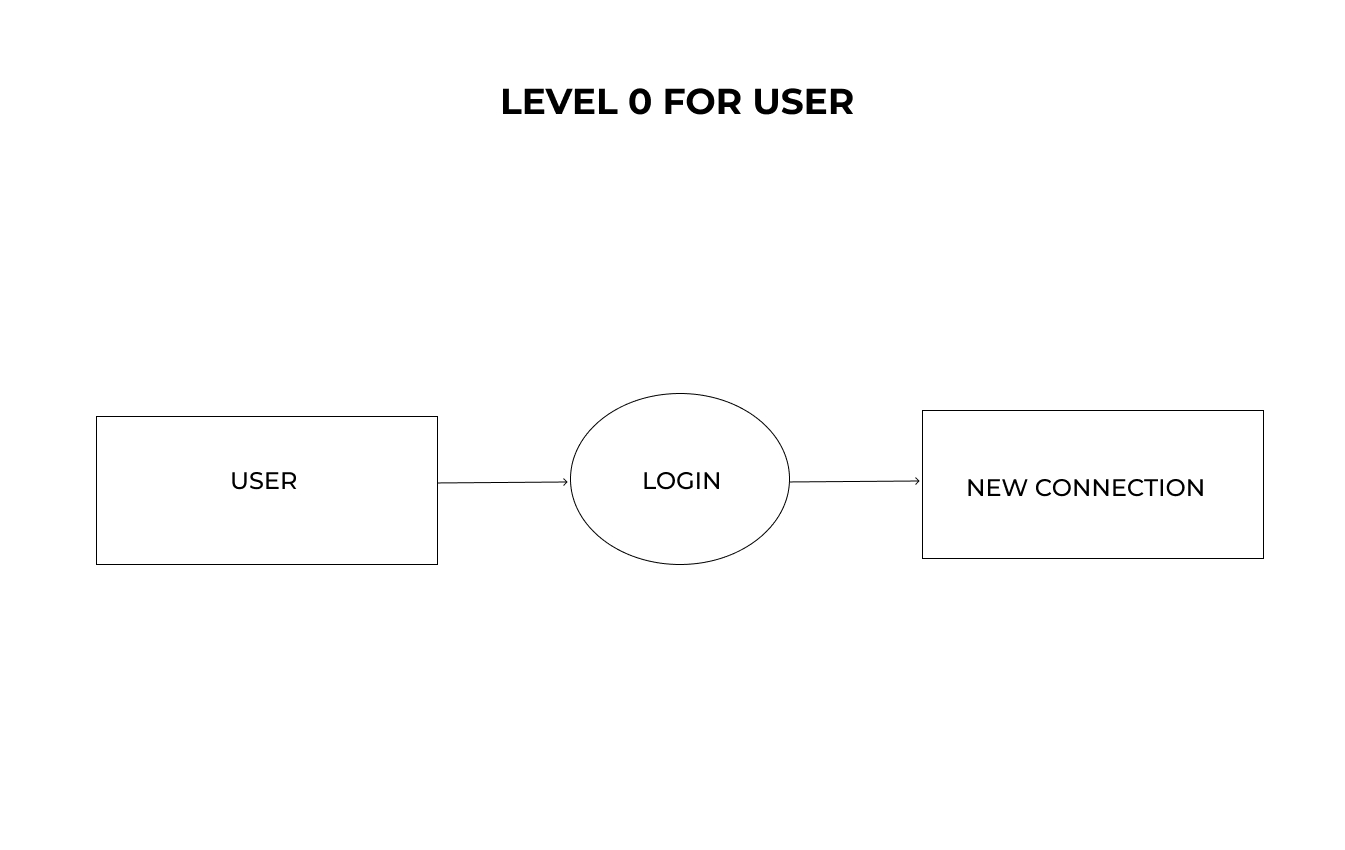
});

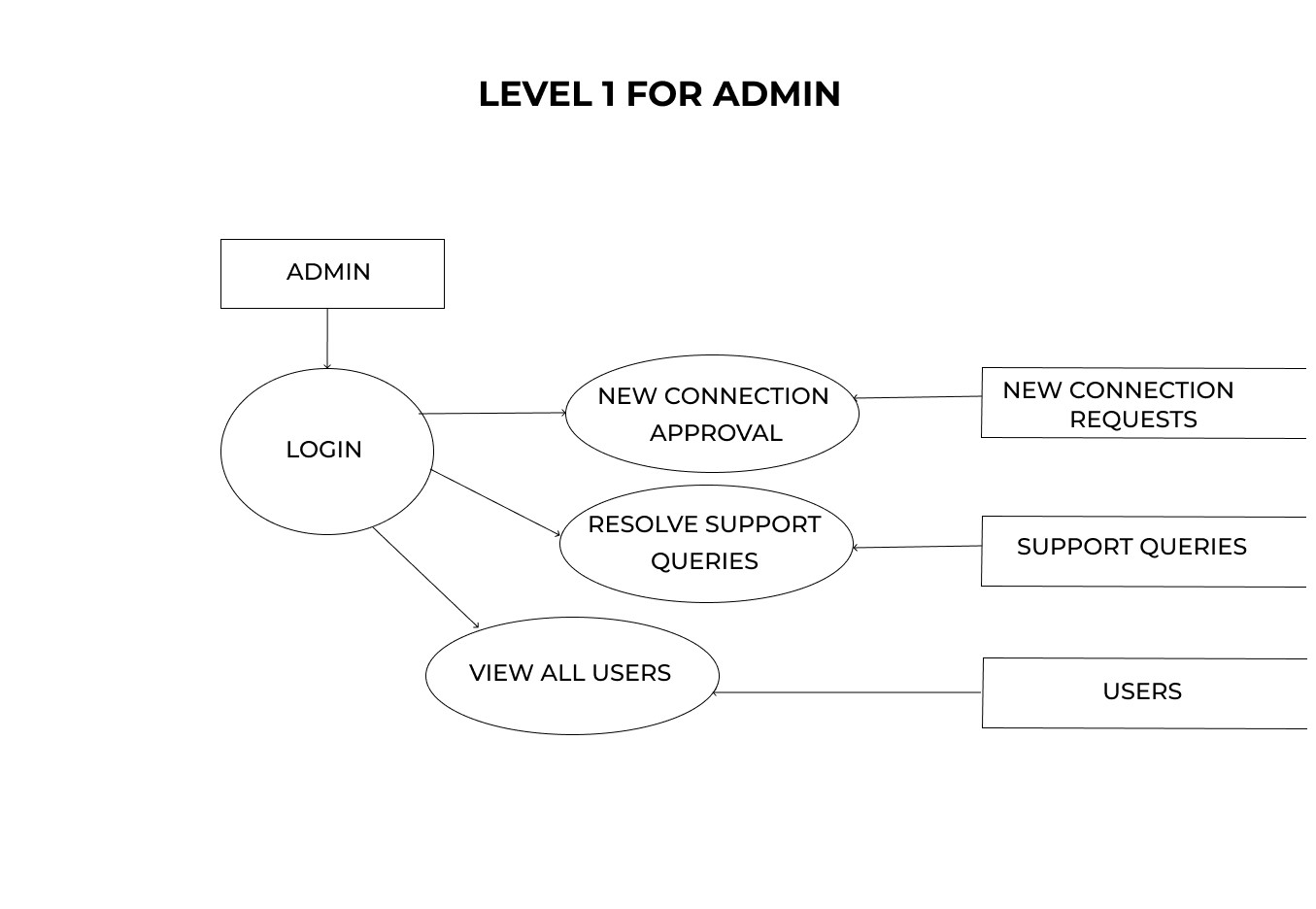
**ER Diagram**

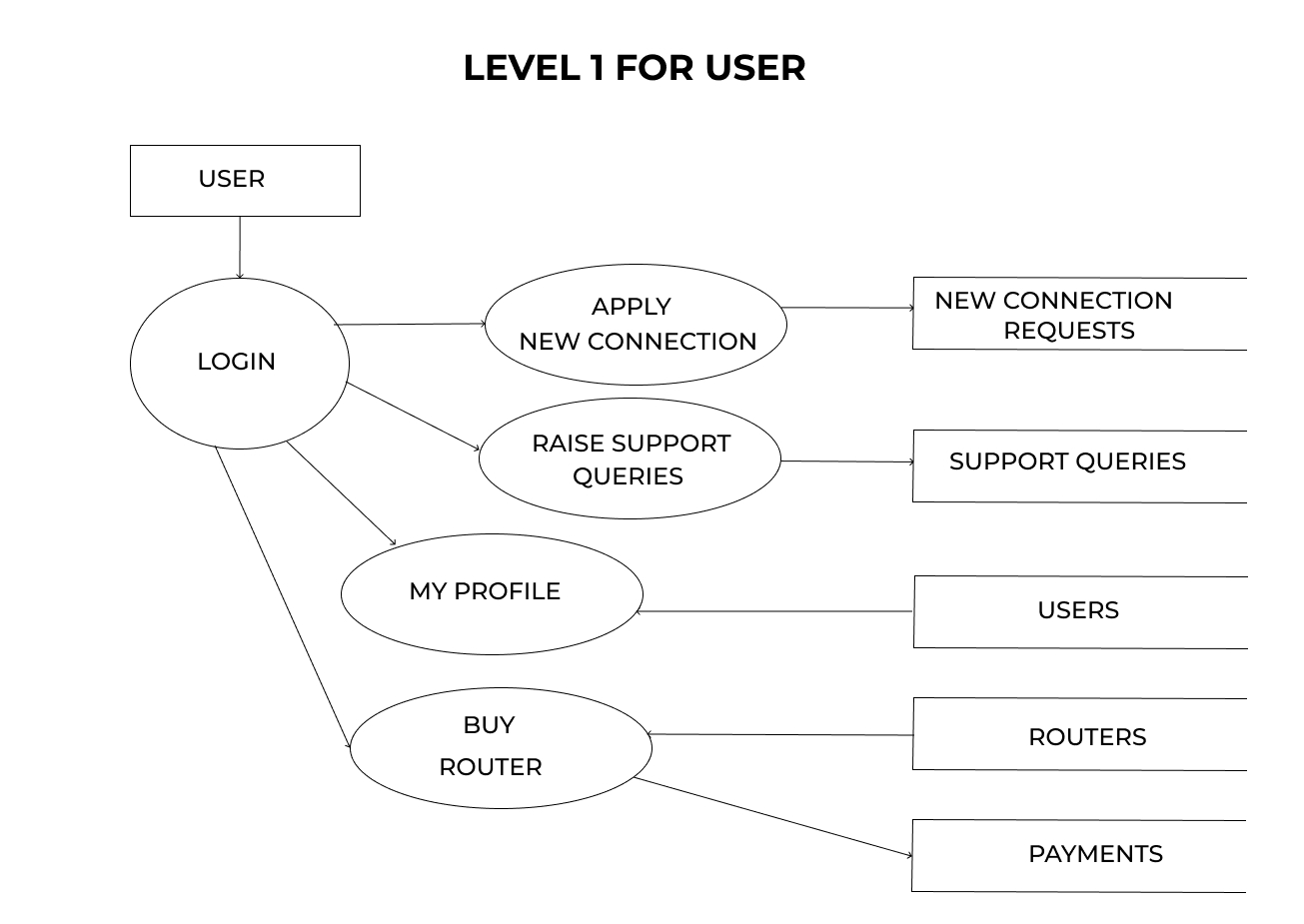
****

**Data-Flow Diagram**



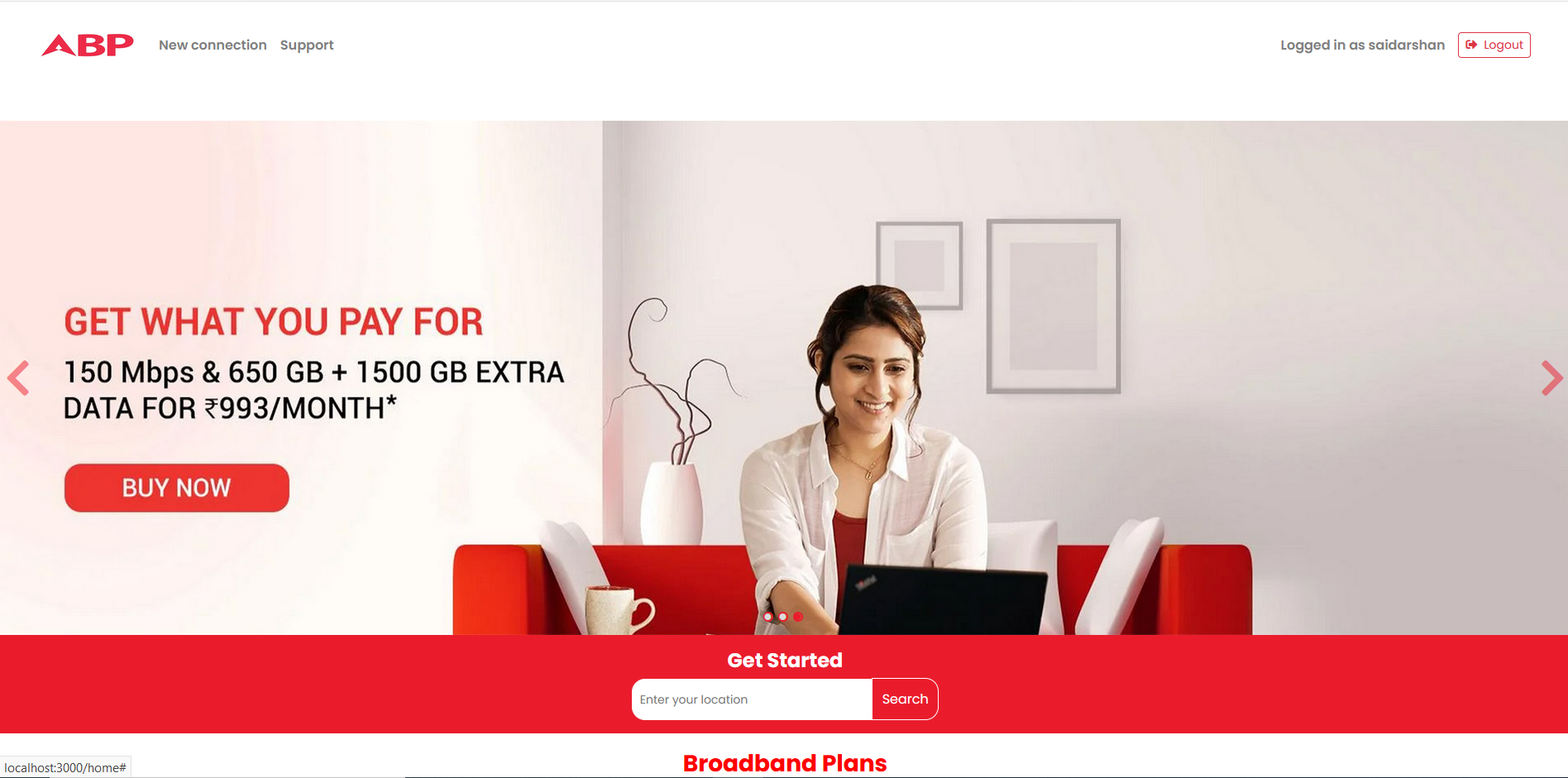




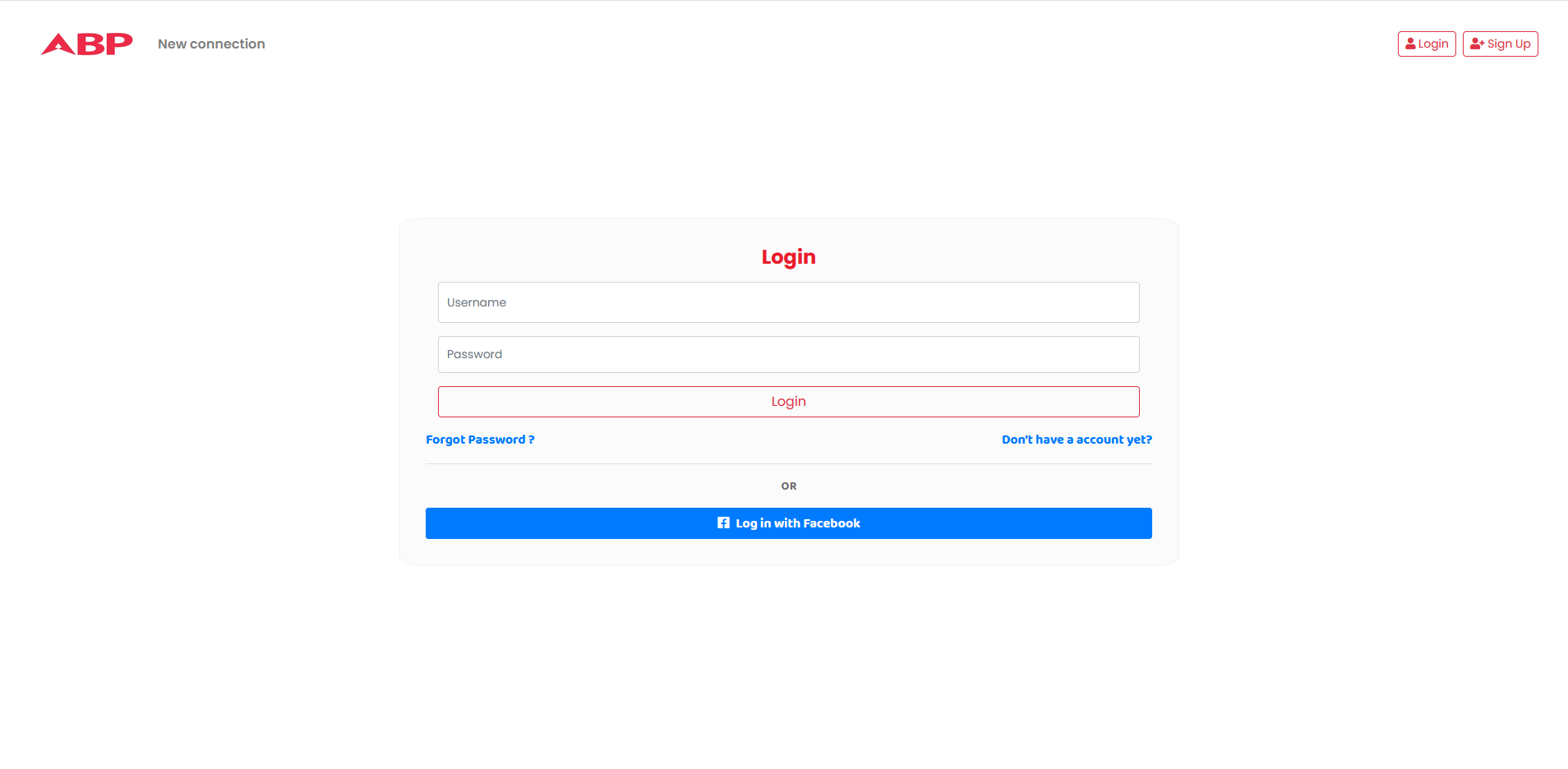


**SCREENSHOTS**

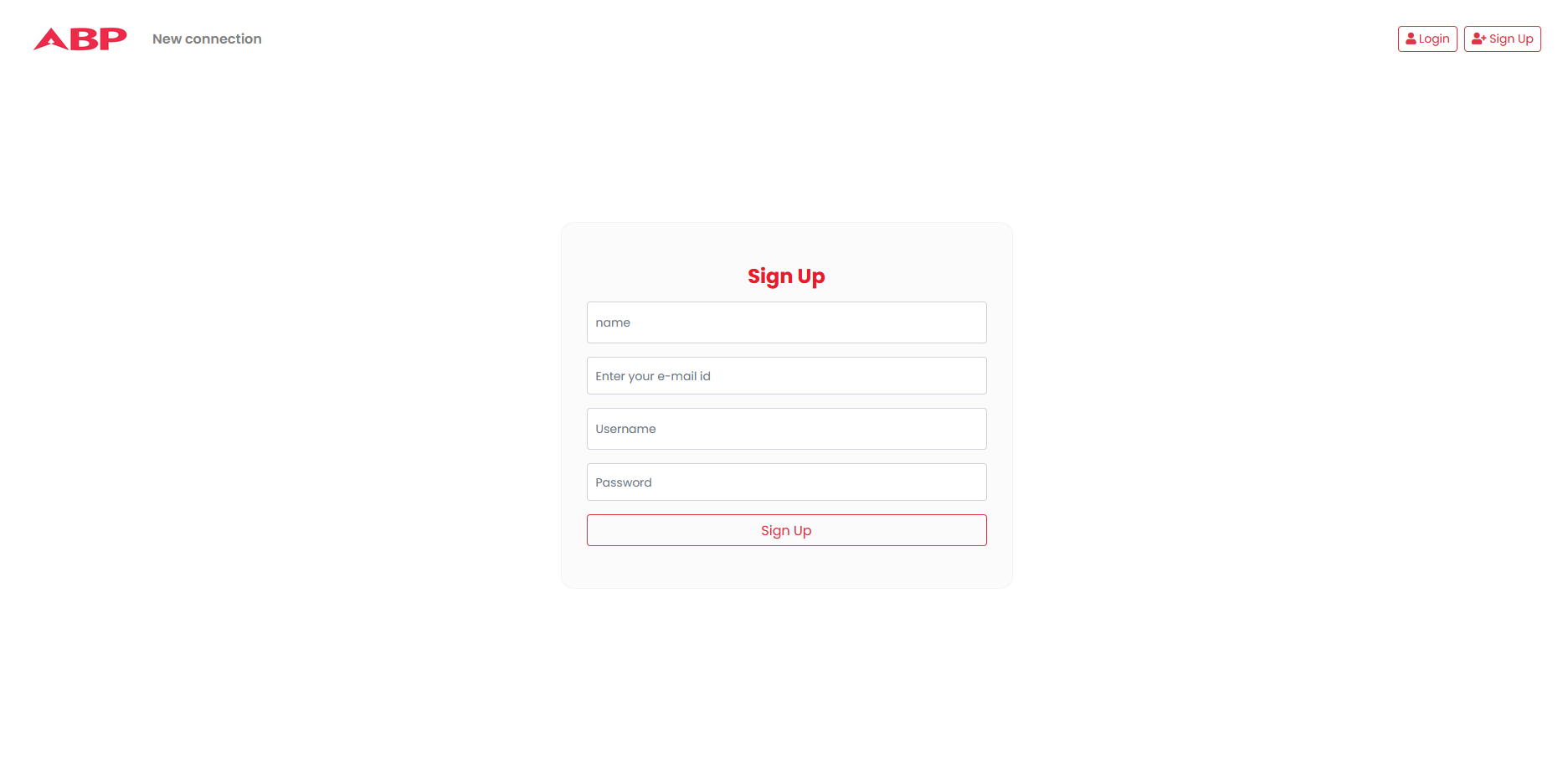
**HOME PAGE:**

****

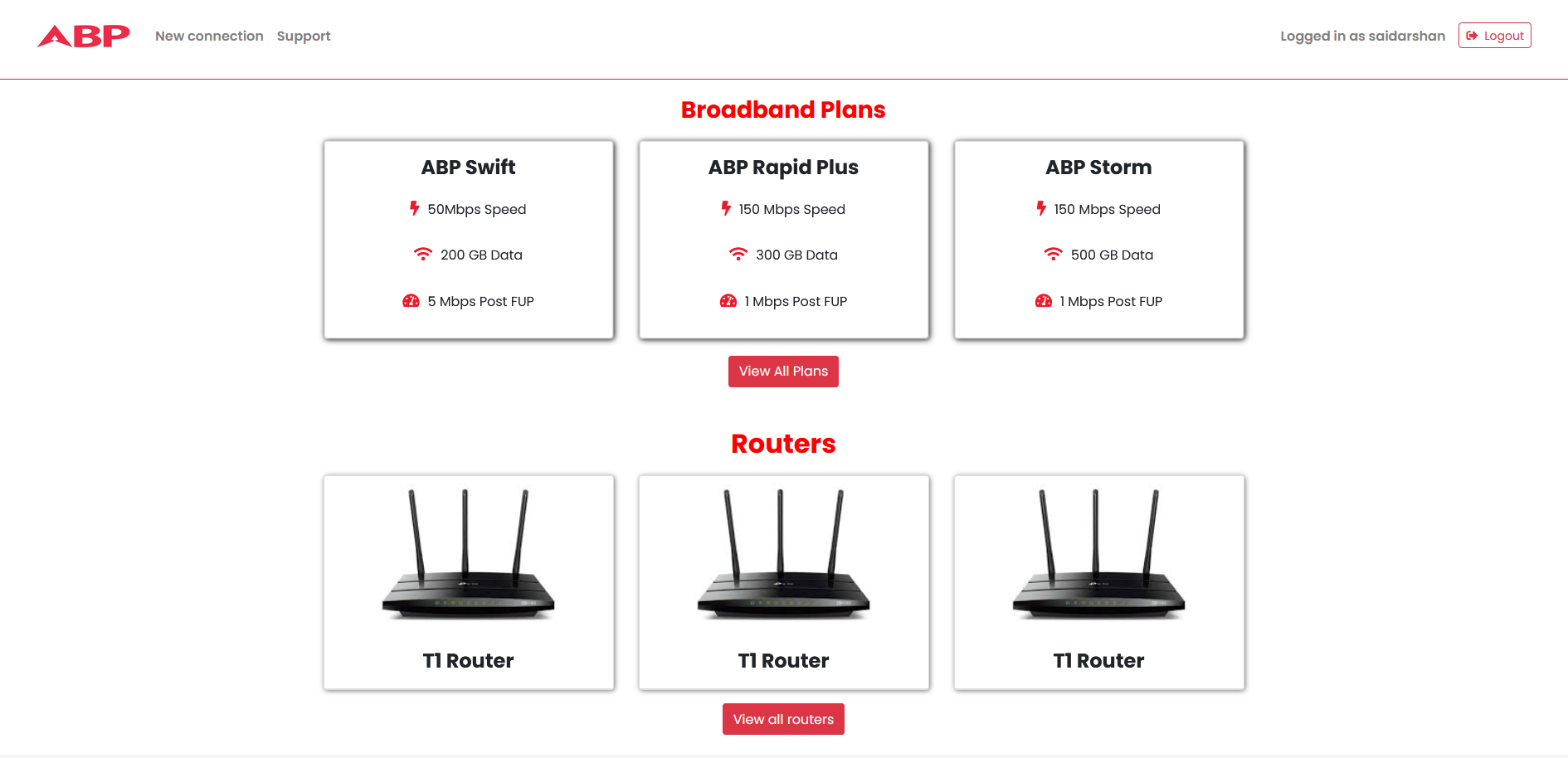
**LOGIN :**

****

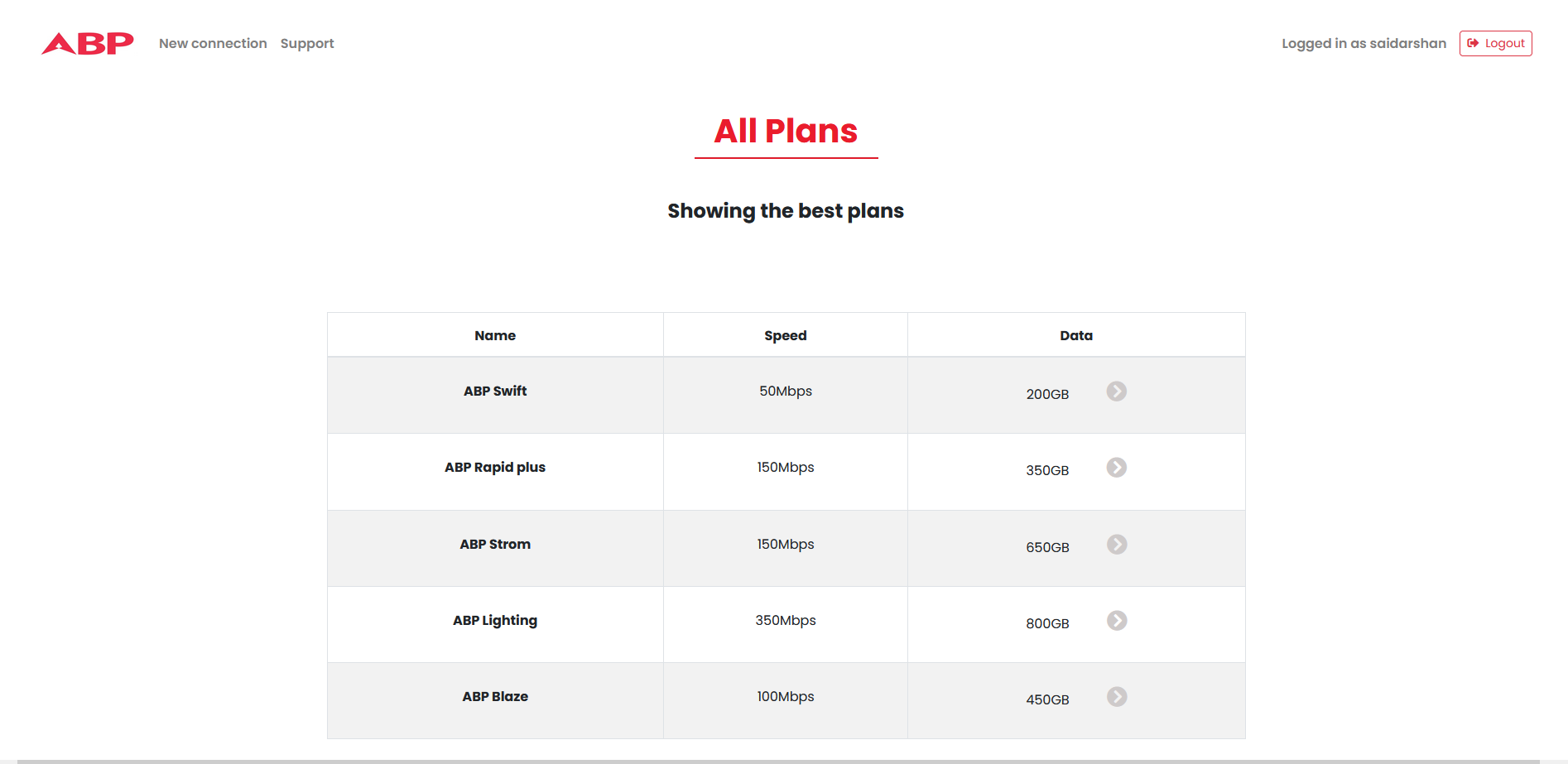
**SIGN UP:**

****

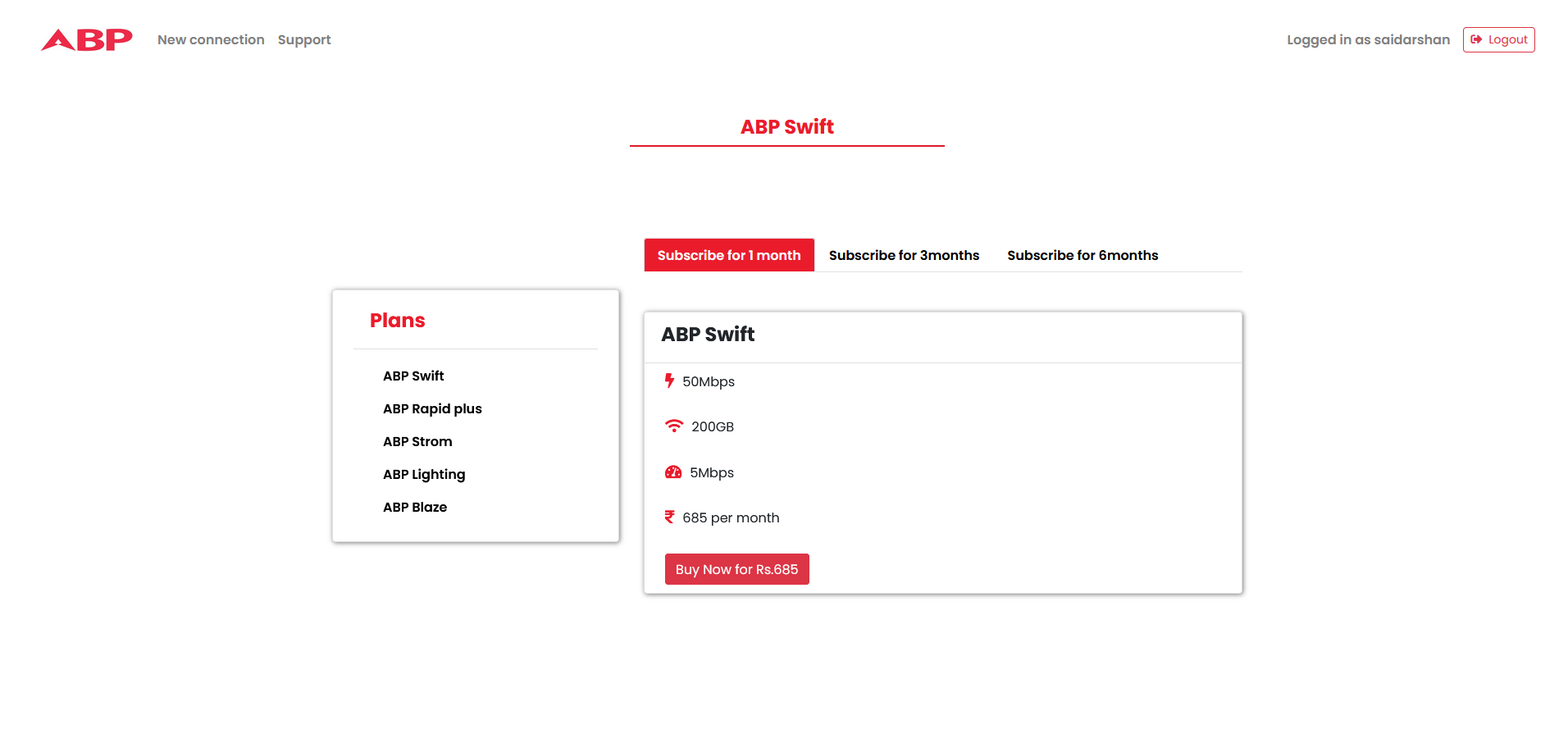
**PLANS AND ROUTERS:**

****

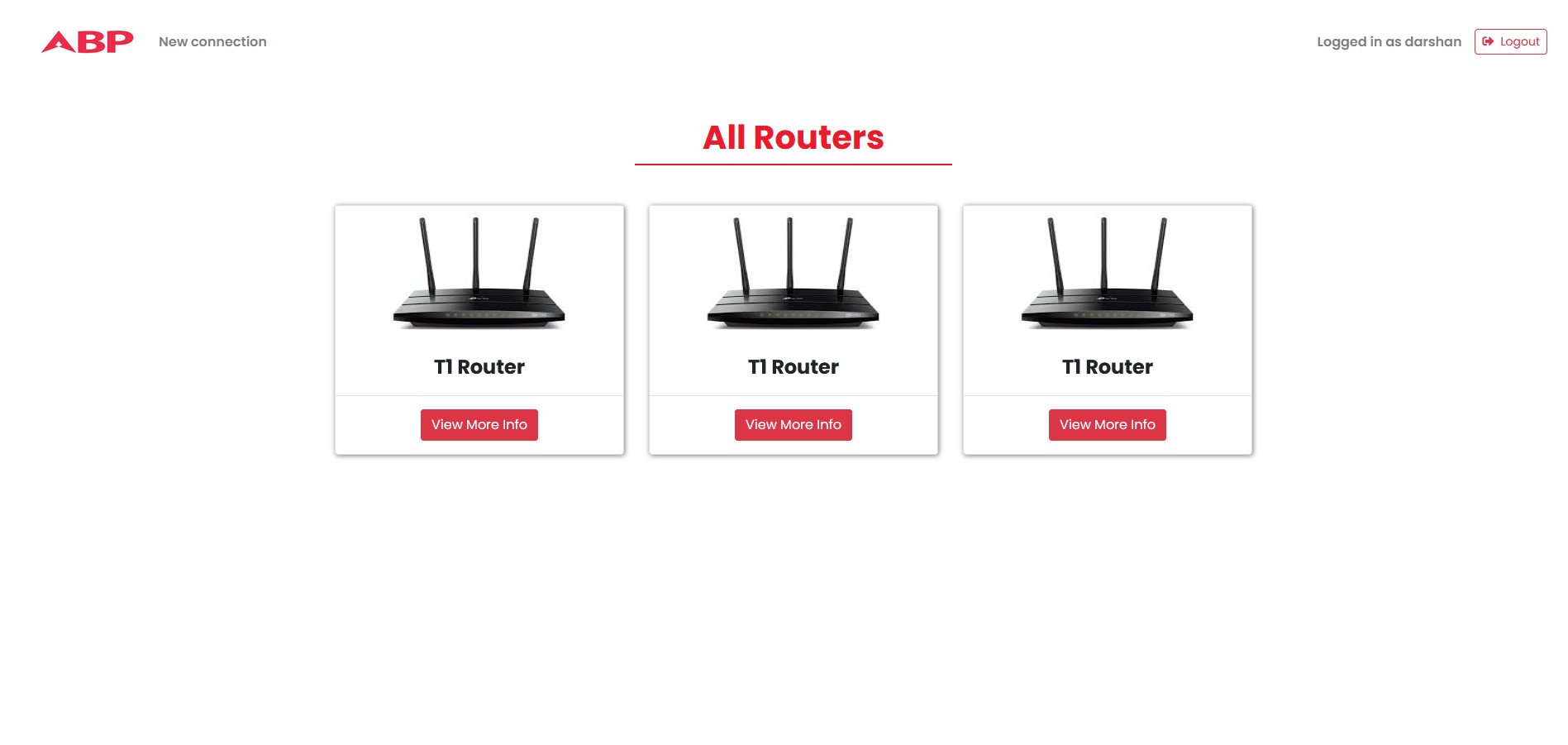
**PLANS LIST**

****

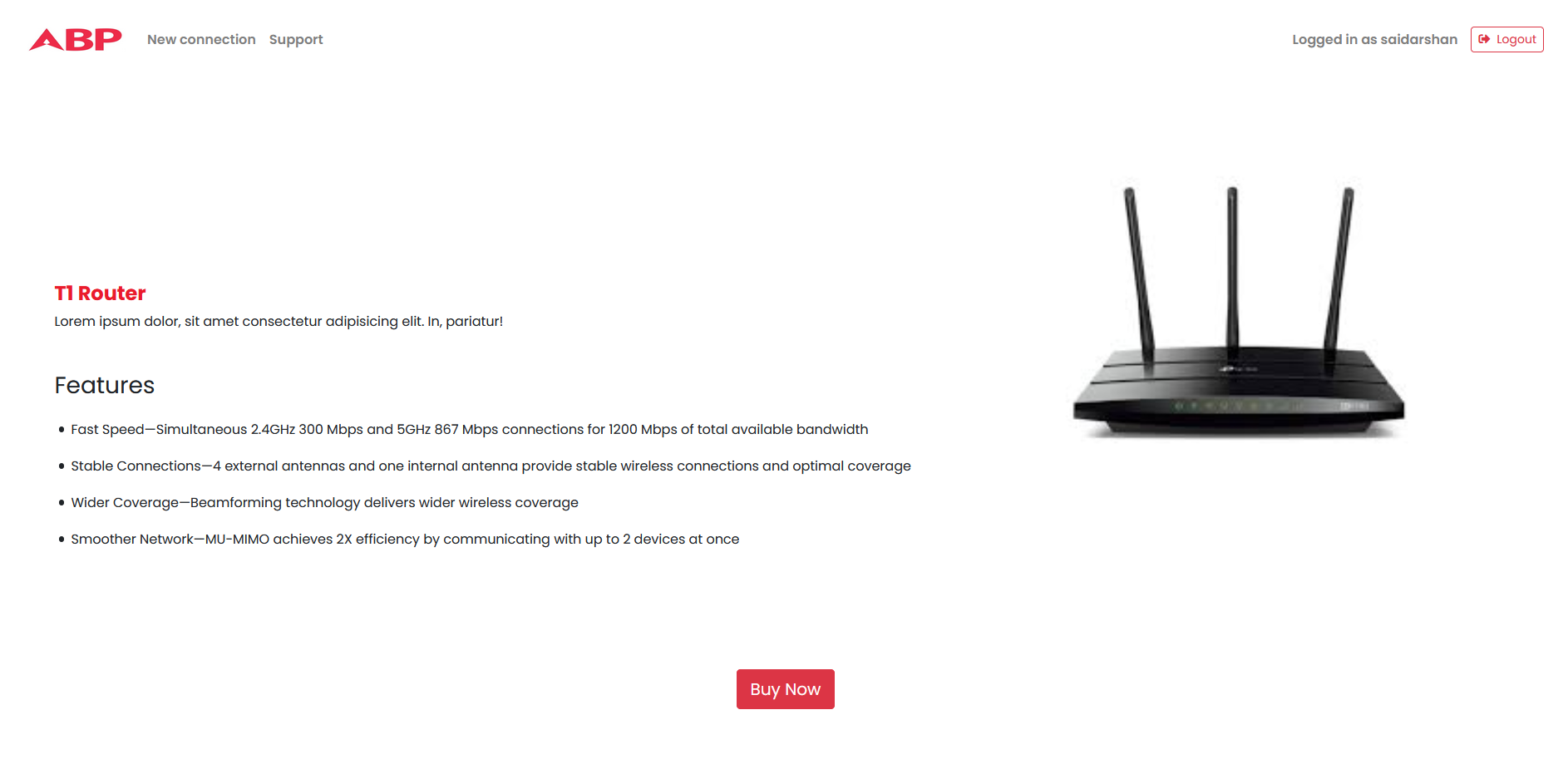
**VIEW PLAN:**

****

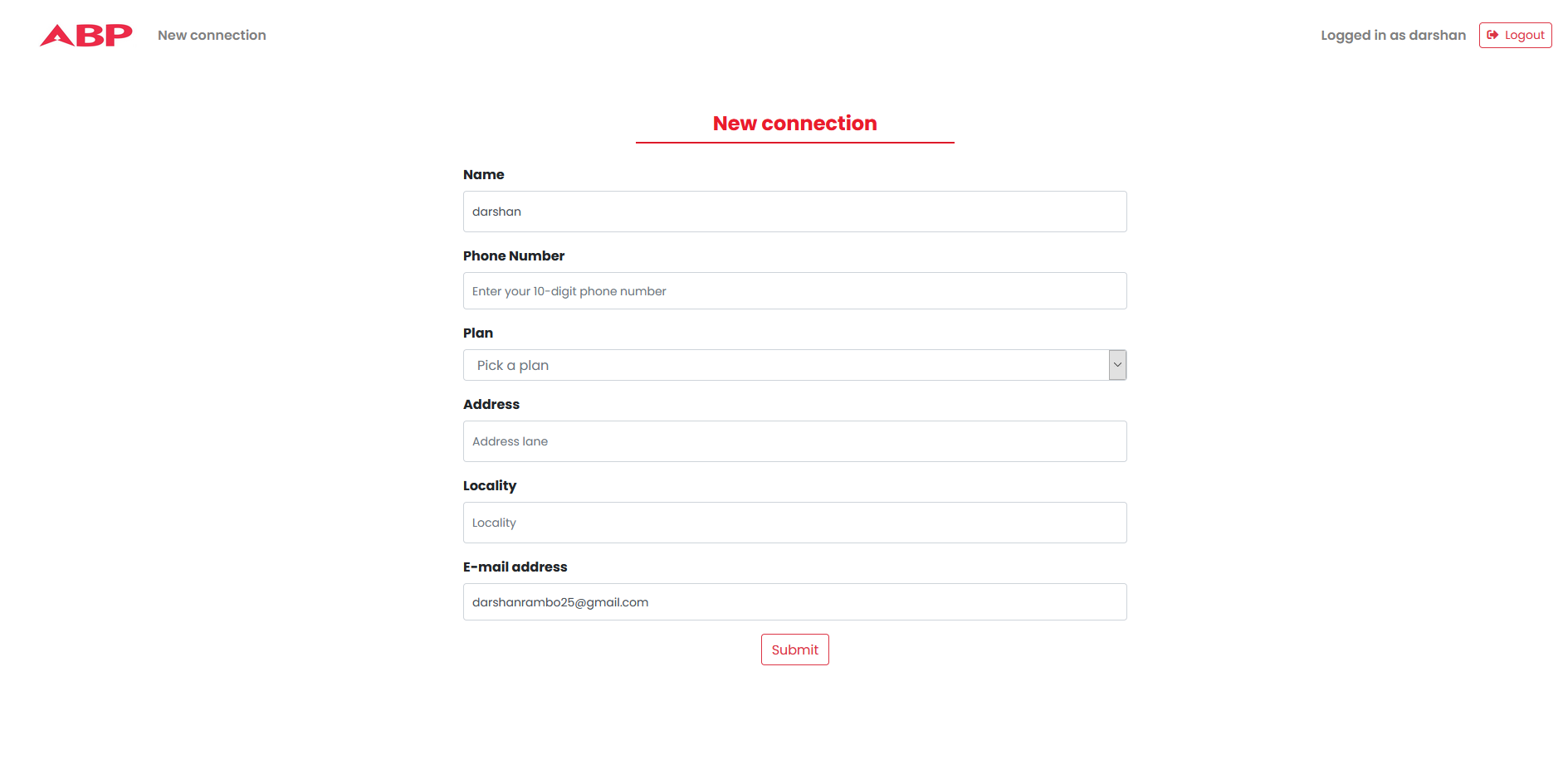
**ALL ROUTERS:**

****

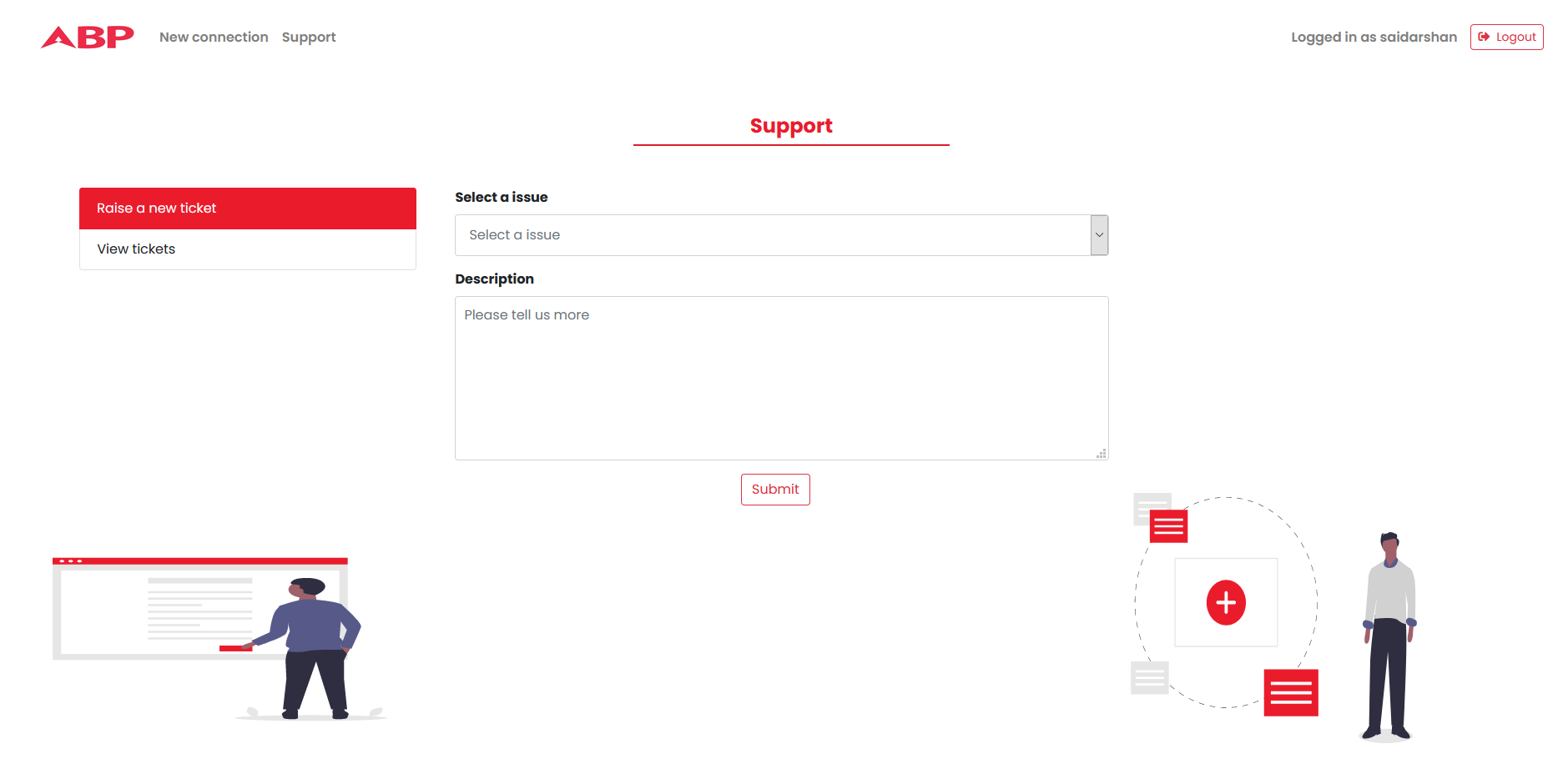
**ROUTER:**

****

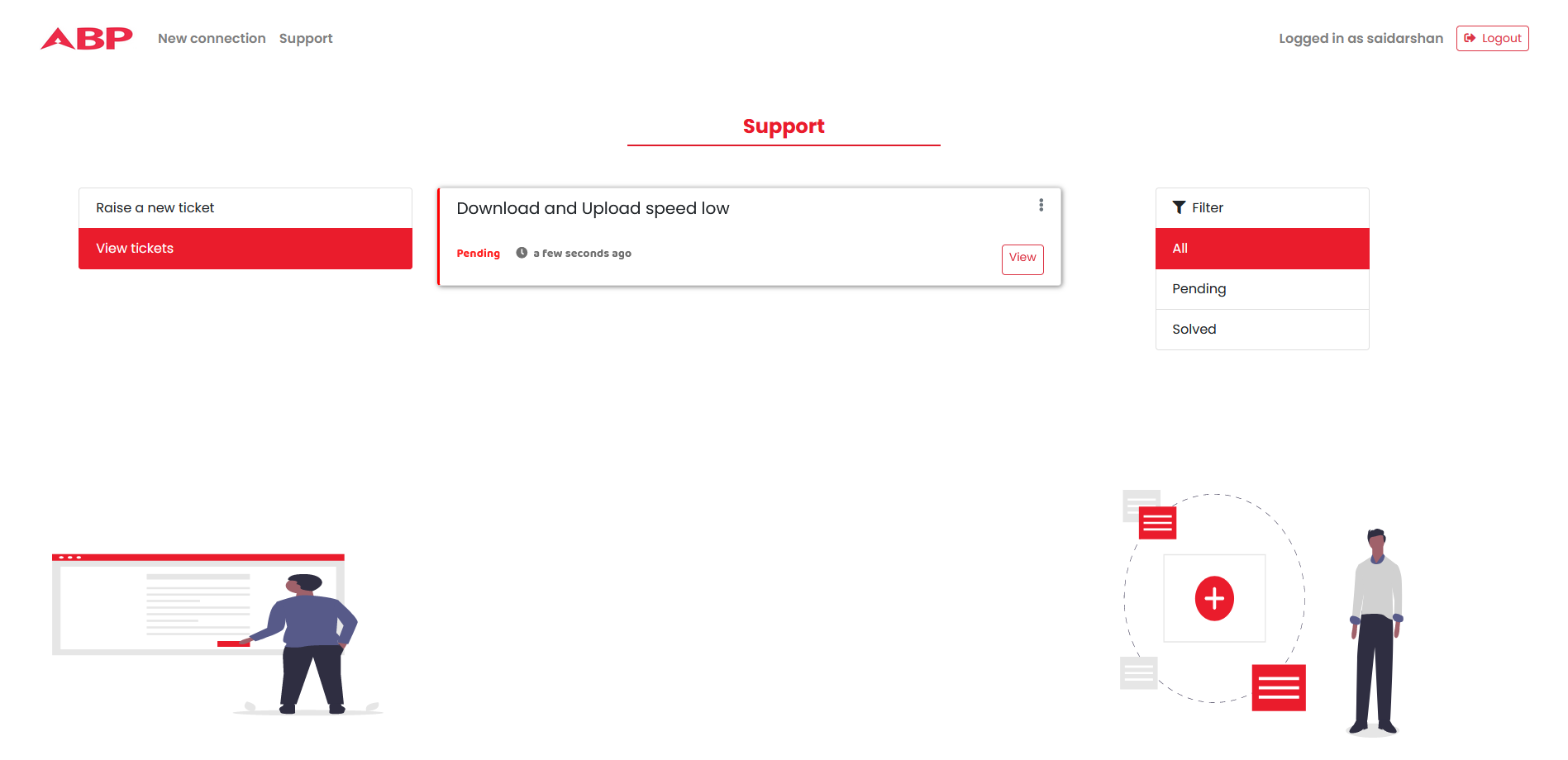
**APPLY FOR NEW CONNECTION:**

****

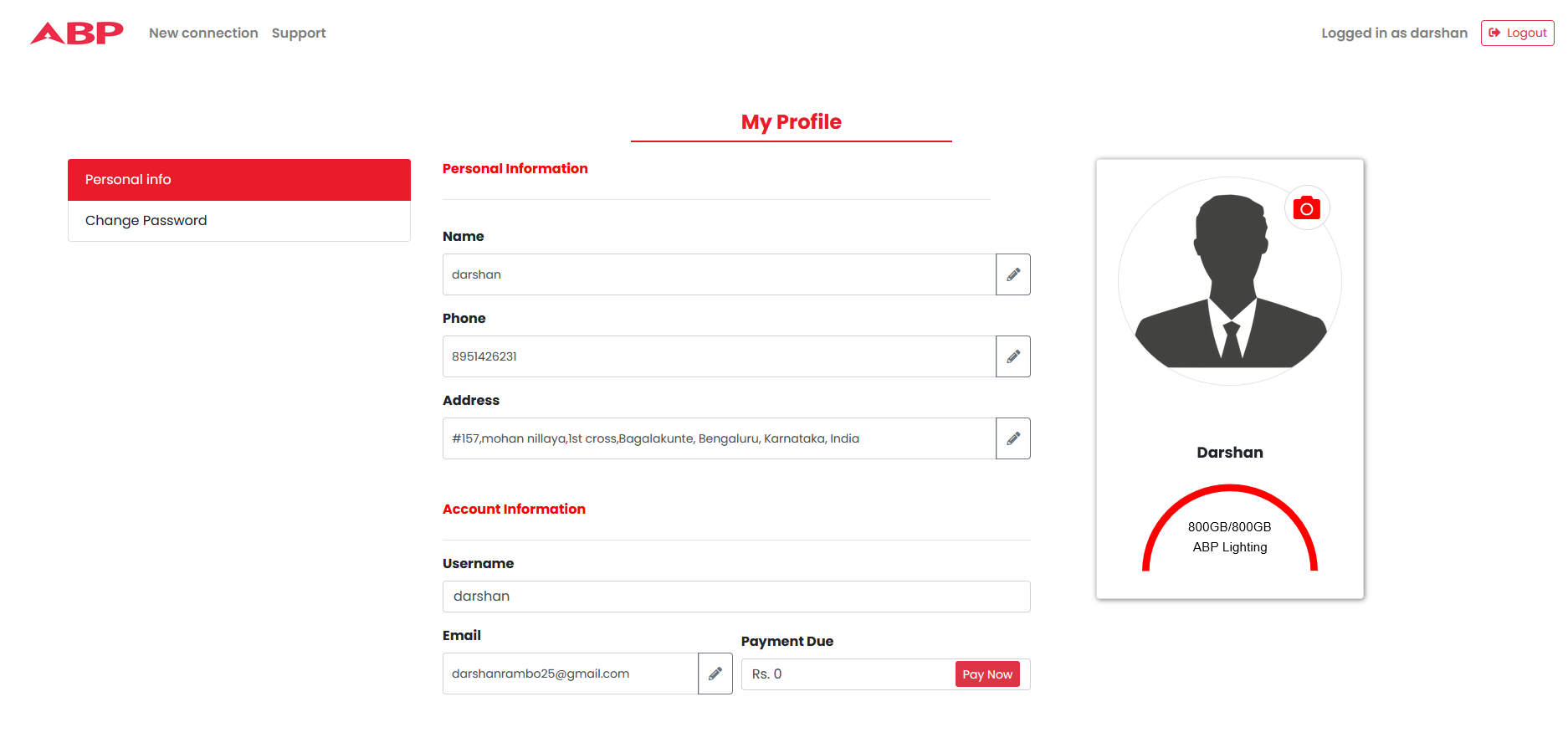
**RAISE SUPPORT QUERY:**

****

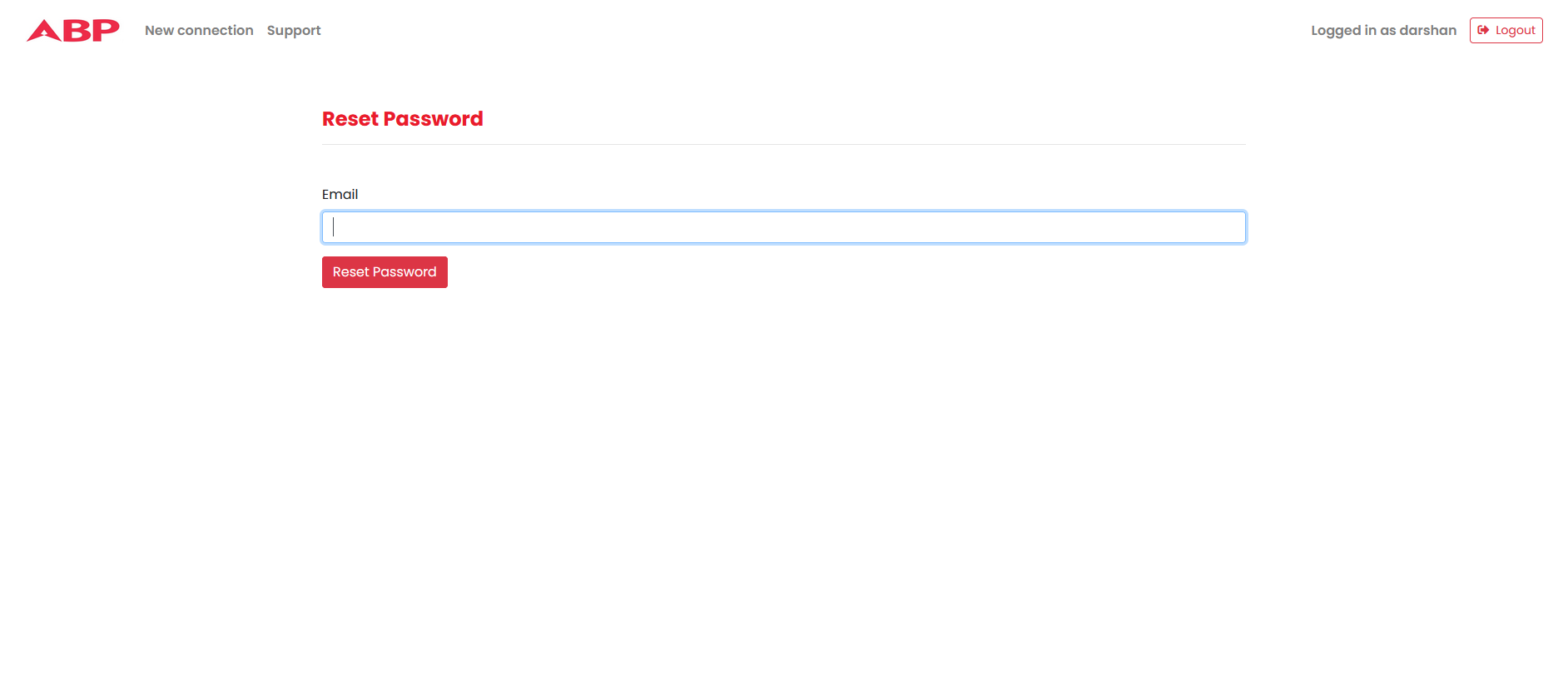
**VIEW SUPPORT QUERIES:**

****

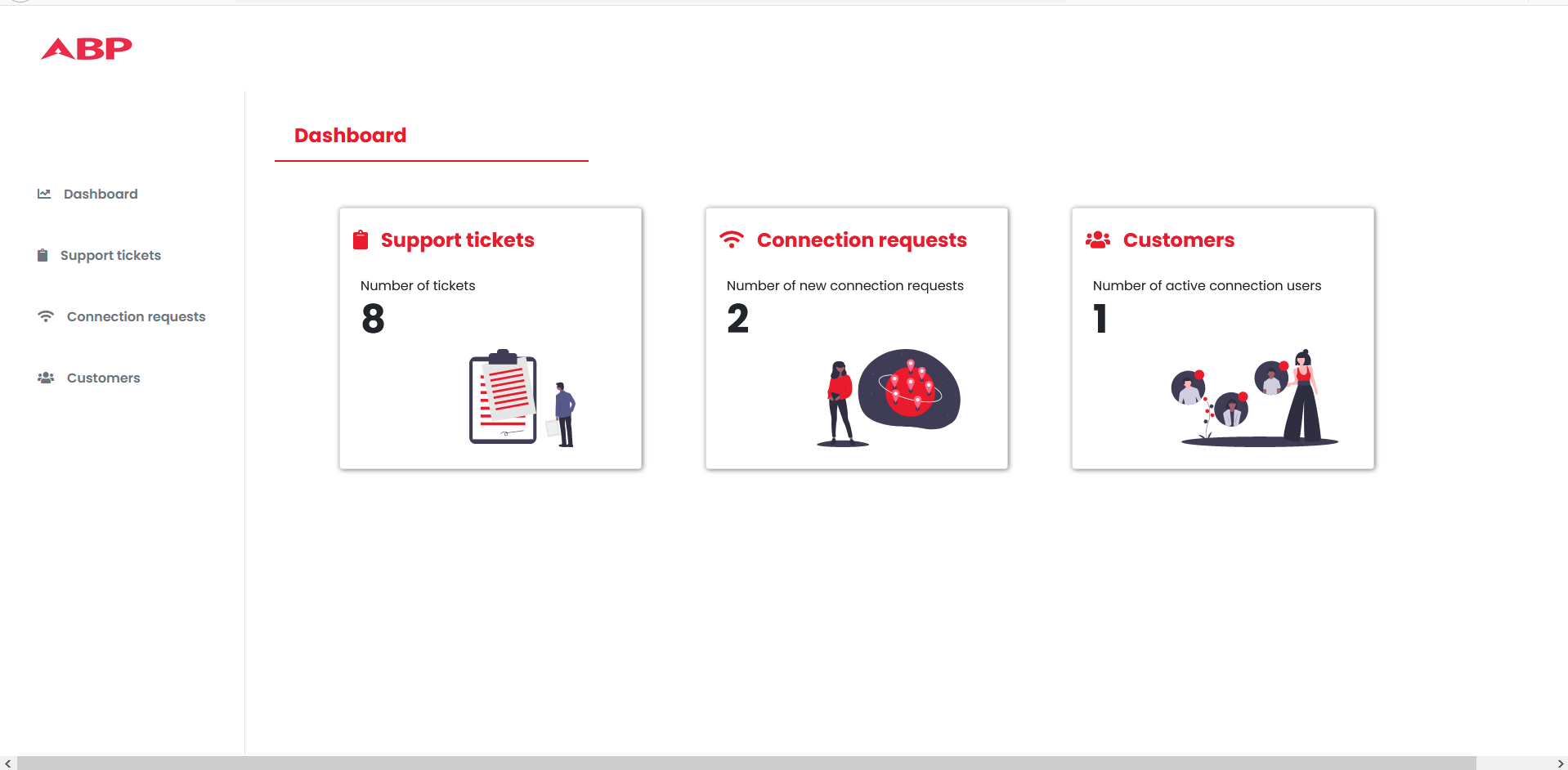
**MY PROFILE:**



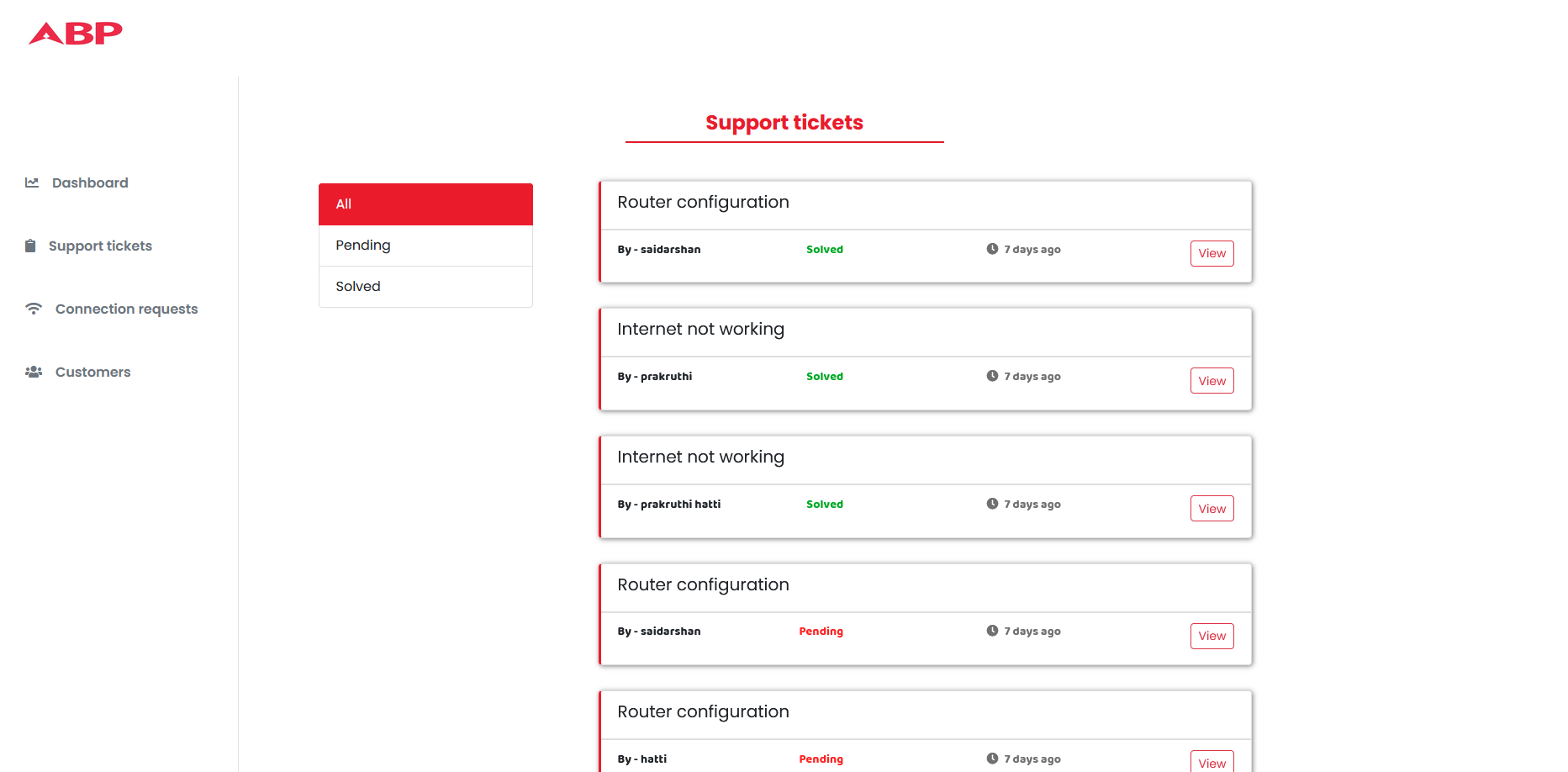
**CHANGE PASSWORD:**



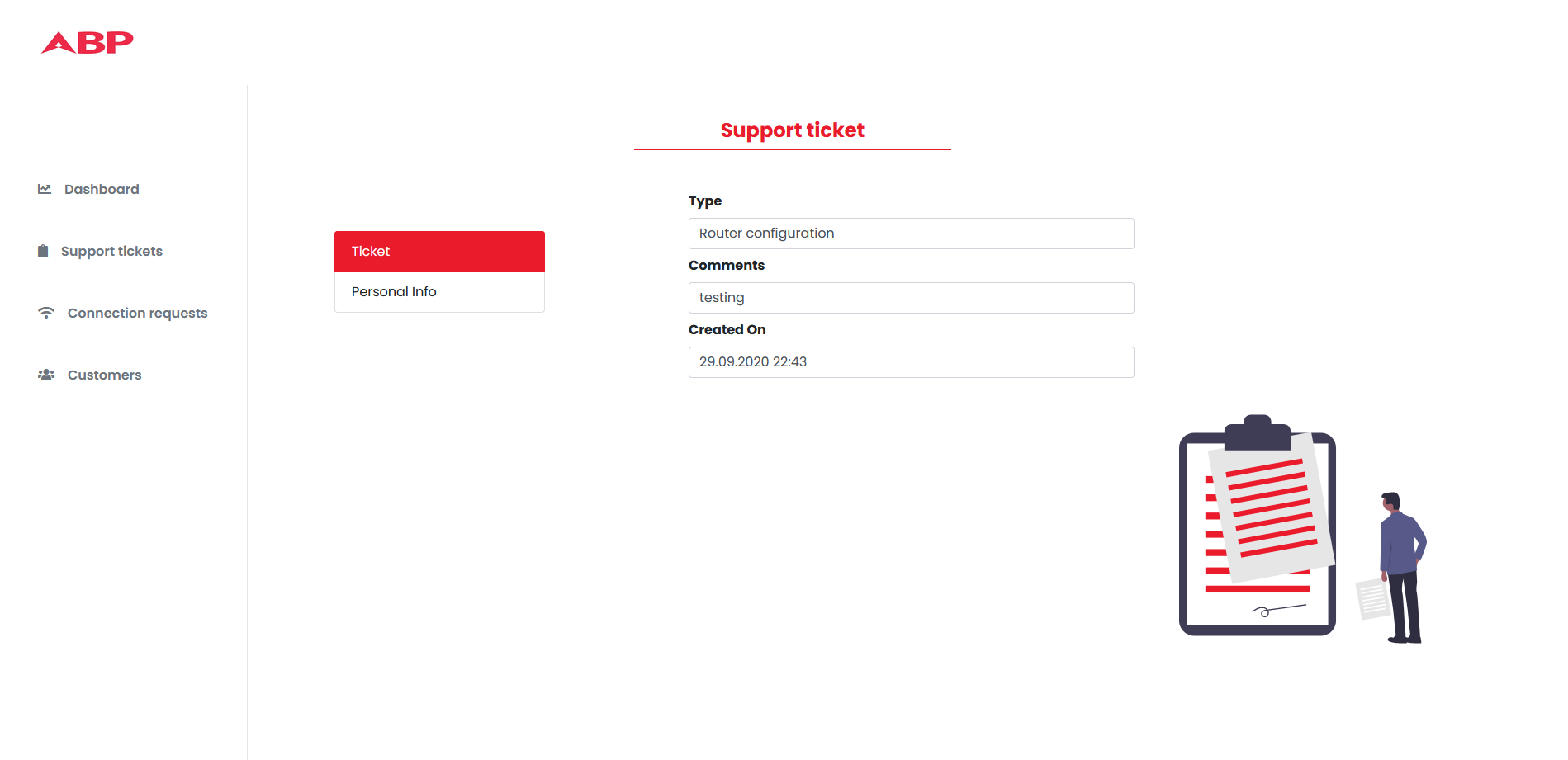
**ADMIN DASHBOARD:**

****

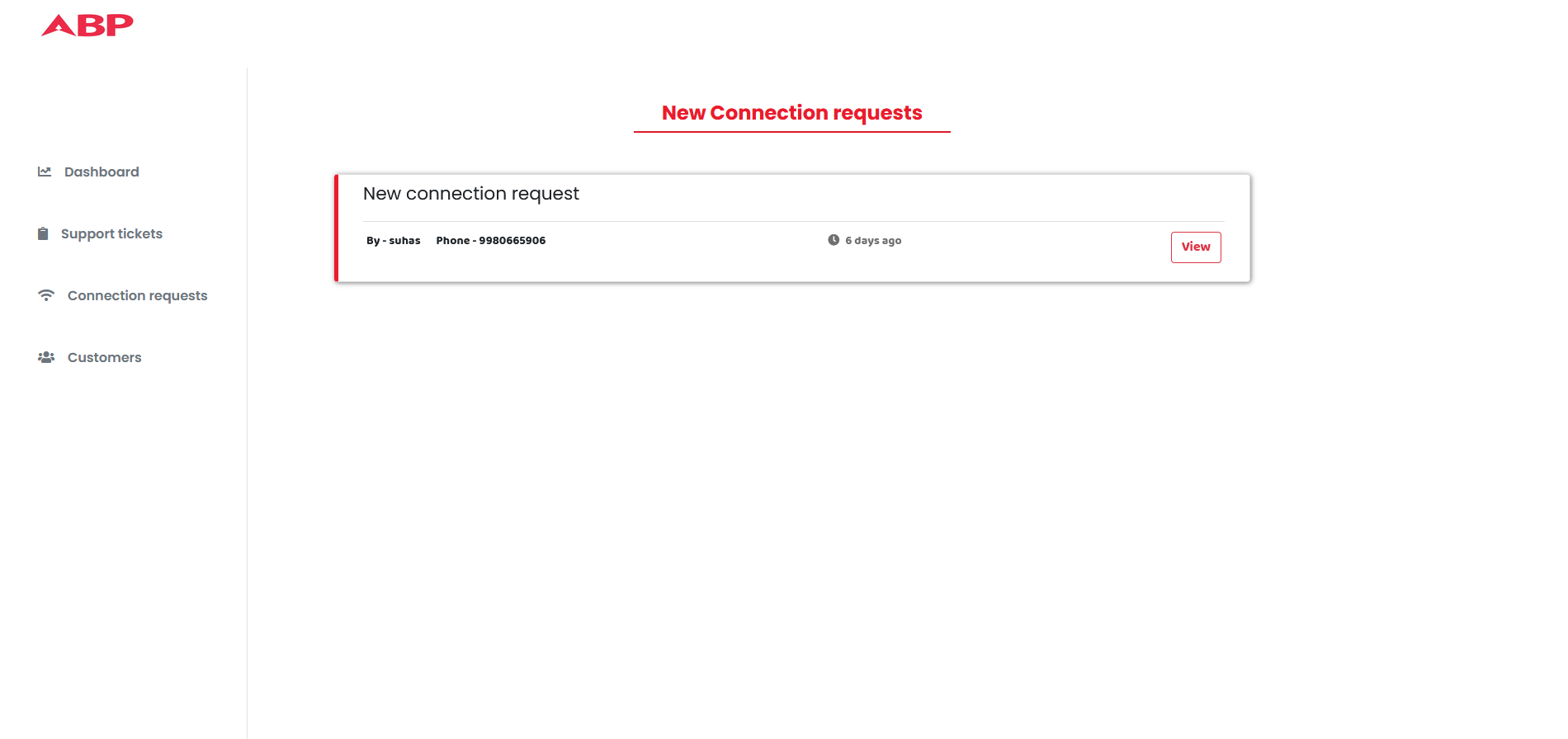
**ALL RAISED SUPPORT QUERIES:**



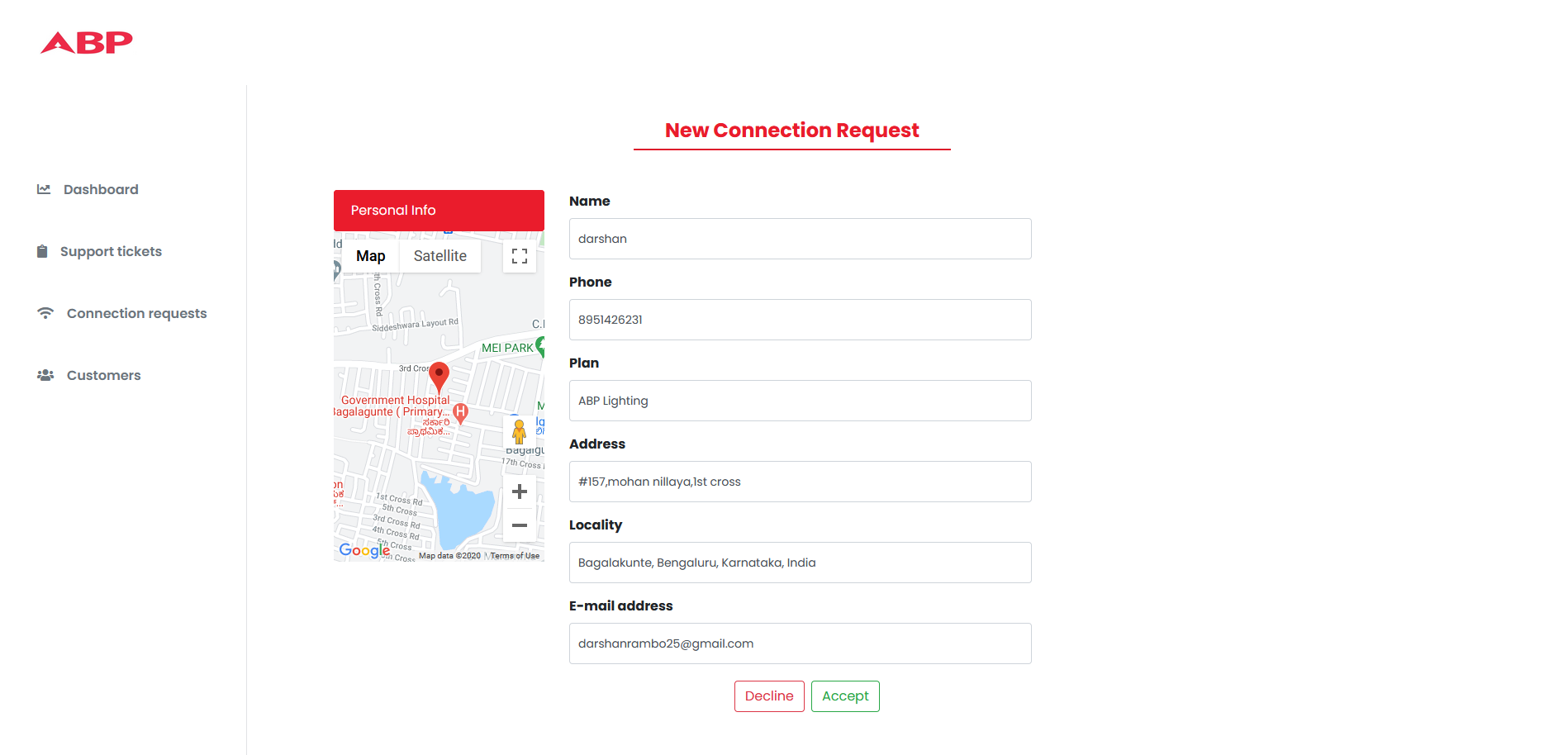
**VIEW RAISED SUPPORT QUERY:**

****

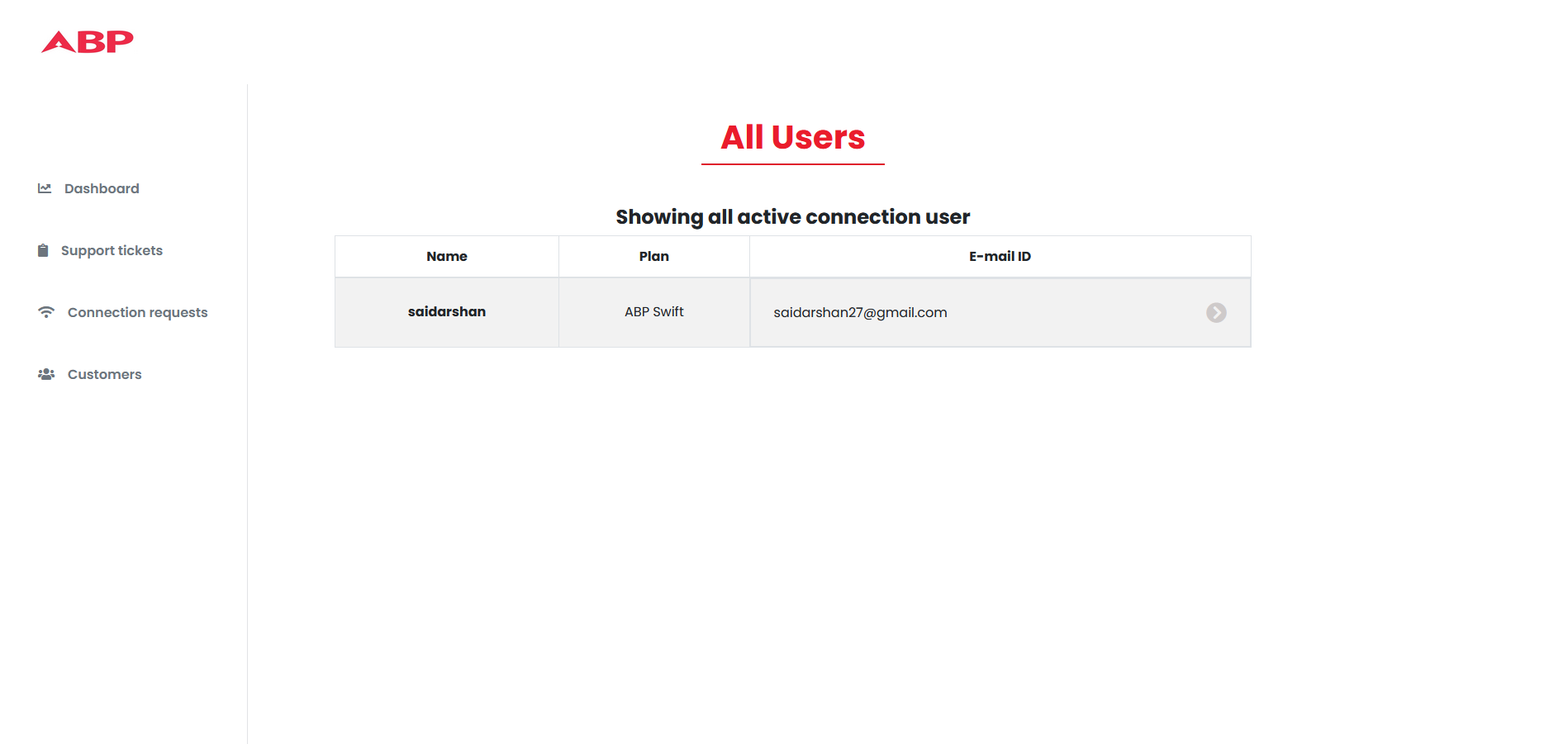
**VIEW ALL NEW CONNECTION REQUESTS**

****

**VIEW NEW CONNECTION REQUEST**

****

**VIEW ALL ACTIVE USERS**



**CONCLUSION**

The given project **“BROADBAND CONNETION PORTAL”** was successfully completed and the required reports are generated. The project has been designed to fulfill the needs of the service provider and the customer. The system is extremely user friendly.

The system is tested with real data.

The results or the live data are compacted with the actual system verified for its accuracy. The system is flexible so that there is a lot of scope to update the system. As the system is flexible the system can be changed if any changes come in future. The developed system is portable. The developed system has been completed which is customized for the satisfaction of the user

The system has been analyzed, designed and developed with meticulous care and can be executed without any faults or errors.

**BIBLIOGRAPHY**

1. **www.stackoverflow.com**

**2. The web developer bootcamp**

* **Colt Steele.**

**3. www.google.com**

**4. HTML and CSS**

* **Jon Duckett.**