**APP CODE .env**

import AppStartUp from "./startup\_components/AppStartUp";

import AppCoreComponent from "./core\_components/AppCoreComponent";

function App(){

return (

<div>

<AppStartUp />

{/\* <AppCoreComponent /> \*/}

</div>

);

}

export default App;

**App\_StartUp.jsx**

import React, { useState } from "react";

import { Grid, Stack } from "@mui/material";

import SignIn from "./SignIn";

import SignUp from "./SignUp";

import Header from "./Header";

import Banner from "./Banner";

import MiniProfile from "./MiniProfile";

import RoomCodePage from "./RoomCodePage";

function AppStartUP() {

const [isLoggedIn, setIsLoggedIn] = useState(true);

const [showSignIn, setShowSignIn] = useState(true);

// a function to handle signin-signup hook in corresponding component files...

function handleSign\_IN\_UP(needSignIn) {

if (needSignIn) {

setShowSignIn(true);

} else {

// Do something if not needSignIn

}

}

return (

<Stack direction="column">

<Header />

<Banner />

<br />

<Grid container fullWidth height={"100px"}>

<Grid item sx={{ backgroundColor: "#eeeeee" }} md={7} padding={5}>

{/\* main room section... \*/}

<RoomCodePage />

</Grid>

<Grid item md={5}>

{/\* Mini Profile Section... \*/}

{isLoggedIn ? (

<MiniProfile />

) : (

showSignIn ? (

<SignIn handleSign\_IN\_UP={handleSign\_IN\_UP} />

) : (

<SignUp handleSign\_IN\_UP={handleSign\_IN\_UP} />

)

)}

</Grid>

</Grid>

</Stack>

);

}

export default AppStartUP;

**RoomCode.jsx**

import React, { useState } from "react";

import {

Paper,

Typography,

FormControl,

FormControlLabel,

FormLabel,

Switch,

Stack,

Box,

Button,

Card,

CardContent,

CardActions,

TextField,

IconButton,

} from "@mui/material";

import ShareIcon from '@mui/icons-material/Share';

import ContentCopyIcon from '@mui/icons-material/ContentCopy';

function RoomCodePage() {

const [haveKey, setHaveKey] = useState(false);

const [roomCode, setRoomCode] = useState(null);

const [buttonText, setButtonText] = useState("Create Room");

const [roomCreated, setRoomCreated] = useState(false);

function handleCodeChange(event) {

setRoomCode(event.target.value);

}

function handleCreateRoomClick(event) {

if (haveKey) {

// Write a Room Joining Code with Room Code in useState(roomCode)

} else {

// Write Room Creation Code..

setButtonText("Join Room");

setRoomCreated(true);

}

}

return (

<Paper elevation={4} sx={{ padding: "20px", width: "360px" }}>

<FormControl>

<Stack direction={"column"}>

<Stack direction={"row"}>

<Box>

<FormLabel>

<Typography component={"span"} variant="body1">

I have a Room Joining Code

</Typography>

</FormLabel>

<FormControlLabel

sx={{ marginLeft: 1 }}

control={

<Switch

checked={haveKey}

onChange={(event, bool\_value) => {

bool\_value

? setButtonText("Join Room")

: setButtonText("Create Room");

setRoomCreated(false);

setHaveKey(bool\_value);

}}

/>

}

label={haveKey ? "Yes" : "No"}

/>

</Box>

</Stack>

<Stack direction={"row"}>

{haveKey ? (

<TextField

label={haveKey ? "Enter Room Code" : ""}

onChange={handleCodeChange}

size="small"

value={roomCode}

/>

) : null}

<Button

size="small"

variant="contained"

color="secondary"

onClick={handleCreateRoomClick}

>

<Typography variant="h6">{buttonText}</Typography>

</Button>

{roomCreated ? (

<Card sx={{ display: "flex", height: 40, marginLeft: 1 }}>

<CardContent sx={{ color: "#ff5722", backgroundColor: "#fafafa" }}>

<Typography sx={{ marginTop: -1 }} variant="h6">ABCDEF</Typography>

</CardContent>

<CardActions><IconButton>{<ContentCopyIcon size="small" color="secondary" />}</IconButton></CardActions>

<CardActions><IconButton>{<ShareIcon size="small" color="secondary" />}</IconButton></CardActions>

</Card>

) : null}

</Stack>

</Stack>

</FormControl>

</Paper>

);

}

export default RoomCodePage;

**SignIn.jsx**

import React from "react";

import { Typography, Stack, FormControl, TextField, Button } from "@mui/material";

function SignIn(props) {

function redirectSignUP() {

props.handleSign\_IN\_UP(false);

}

return (

<Stack direction={"column"} sx={{ border: "solid 1px" }} spacing={12} padding={1}>

<FormControl sx={{ paddingLeft: 8, paddingRight: 8 }}>

<Typography sx={{ textAlign: "center" }} variant="h4" gutterBottom>

Login

</Typography>

<TextField size="small" label="Email ID" type="email" required />

<br />

<TextField size="small" label="Password" type="password" required />

<Button size="small" onClick={redirectSignUP}>

Don't have an account, Create One

</Button>

</FormControl>

</Stack>

);

}

export default SignIn;

import React from "react";

import { Typography, Stack, FormControl, TextField, Button } from "@mui/material";

function SignUp(props) {

function redirectSignIN() {

props.handleSign\_IN\_UP(true);

}

return (

<Stack direction={"column"} sx={{ border: "solid 1px" }} spacing={12} padding={1}>

<FormControl sx={{ paddingLeft: 8, paddingRight: 8 }}>

<Typography sx={{ textAlign: "center" }} variant="h4" gutterBottom>

Sign UP

</Typography>

<TextField size="small" label="Email ID" type="email" required />

<br />

<TextField size="small" label="Password" type="password" required />

<br />

<TextField size="small" label="Confirm Password" type="password" required />

<Button size="small" onClick={redirectSignIN}>

Already have an account, Login

</Button>

</FormControl>

</Stack>

);

}

export default SignUp;

**Header.jsx**

import React from "react";

import { Stack, Typography, Button, ButtonGroup } from "@mui/material";

import HomeIcon from '@mui/icons-material/Home';

import DescriptionIcon from '@mui/icons-material/Description';

import SportsMartialArtsIcon from '@mui/icons-material/SportsMartialArts';

import PeopleIcon from '@mui/icons-material/People';

function Header() {

return (

<Stack direction={"column"} spacing={1}>

<Typography variant="h3" gutterBottom sx={{ color: "secondary.main", paddingLeft: "32px" }}>

Code Arena

</Typography>

<Stack direction={"row"} spacing={5} pl={4} pr={4}>

<ButtonGroup fullWidth orientation="horizontal" size="large" color="secondary" >

<Button startIcon={<HomeIcon />} >Home</Button>

<Button startIcon={<DescriptionIcon />} >References</Button>

<Button startIcon={<SportsMartialArtsIcon />} >Exercises</Button>

<Button startIcon={<PeopleIcon />} >About Us</Button>

</ButtonGroup>

</Stack>

<hr />

</Stack>

);

}

export default Header;

**Profile.jsx**

import React from "react";

import { Stack, Card, CardContent, CardMedia, CardActions, Typography } from "@mui/material";

function MiniProfile() {

return (

<Stack direction={"column"}>

<Card sx={{ padding: 2, paddingLeft: 3, display: "flex" }}>

<CardMedia

component={"img"}

sx={{ width: 200, height: 200, borderRadius: "50%" }}

alt="profile-photo"

src="https://avatars.githubusercontent.com/u/122853435?v=4"

/>

<CardContent>

<Typography variant="h5" aria-label="name" >

Komal Koutilya Penumarti

</Typography>

</CardContent>

<CardActions></CardActions>

</Card>

</Stack>

);

}

export default MiniProfile;

**AppCore.jsx**

import React from "react";

import { Stack, Grid } from "@mui/material";

import Header from "./Header";

import CodeArea from "./CodeArea";

import QuestionArea from "./QuestionArea";

import TestCaseSubmitArea from "./TestCaseSubmitArea";

function AppCoreComponent() {

return (

<Stack spacing={0} padding={0} margin={0}>

<Header />

<hr />

<Stack spacing={2}>

<Grid container>

<Grid item md={4.8} sx={{ borderRight: "solid 2px #aaaaaa" }}>

<QuestionArea />

</Grid>

<Grid item md={7.2}>

<CodeArea />

<TestCaseSubmitArea />

</Grid>

</Grid>

</Stack>

</Stack>

);

}

export default AppCoreComponent;

**Header.jsx**

import React from "react";

import { Stack, Typography, Grid, ButtonGroup, Button } from "@mui/material";

function Header() {

return (

<Stack direction={"row-reverse"}>

<Grid container>

<Grid item md={10}>

<Typography variant="h4">1037 Substring with Concatenation of All Words</Typography>

</Grid>

<Grid item md={2}>

<ButtonGroup>

<Button>1</Button>

<Button>2</Button>

<Button>3</Button>

<Button>4</Button>

<Button>5</Button>

</ButtonGroup>

</Grid>

</Grid>

</Stack>

);

}

export default Header;

**AddCard.jsx**

import { onValue, ref, update } from "firebase/database";

import {

Container,

Flex,

Box,

Heading,

Text,

IconButton,

Button,

VStack,

HStack,

Wrap,

WrapItem,

FormControl,

FormLabel,

Input,

InputGroup,

InputLeftElement,

Textarea,

} from "@chakra-ui/react";

import {

MdPhone,

MdEmail,

MdLocationOn,

MdFacebook,

MdOutlineEmail,

} from "react-icons/md";

import { BsGithub, BsDiscord, BsPerson } from "react-icons/bs";

import React, { useState } from "react";

import { v4 } from "uuid";

import {

getStorage,

ref as ref1,

uploadBytes,

listAll,

getDownloadURL,

} from "firebase/storage";

import { Link } from "react-router-dom";

export default function AddCard({ db, storage }) {

const [imageUpload, setImageUpload] = useState(null);

const uploadImage = (name) => {

if (imageUpload == null) return;

const imageRef = ref1(storage, Email/${name}/${v4()});

uploadBytes(imageRef, imageUpload);

};

const handleImageUpload = async (e) => {

const file = e.target.files[0];

setImageUpload(file);

};

async function add() {

let name = document.querySelector(".name").value;

let cat = document.querySelector(".cat").value;

let base = document.querySelector(".base-price").value;

let loc = document.querySelector(".location").value;

let date = document.querySelector(".date").value;

let img1 = document.querySelector(".pro-img").value;

let date1 = new Date(date);

uploadImage(name);

update(ref(db, Varshith/Products), {

[Email\_${name}]: {

Base: parseInt(base),

Location: loc,

Author: "Author",

Time: date1,

Cat: cat,

img : img1,

Bid: {

Bid1: "",

Bid2: "",

Bid3: "",

},

},

});

update(ref(db, Varshith/Login/Email), {

[name]: base,

});

}

return (

<>

<Container

bg="#9DC4FB"

maxW="full"

mt={0}

centerContent

overflow="hidden"

>

<Flex>

<Box

bg="#02054B"

color="white"

borderRadius="lg"

m={{ sm: 4, md: 16, lg: 10 }}

p={{ sm: 5, md: 5, lg: 16 }}

>

<Box p={4}>

<Wrap spacing={{ base: 20, sm: 3, md: 5, lg: 20 }}>

<WrapItem>

<Box>

<Heading>Need Help? Contact Us.</Heading>

<Text mt={{ sm: 3, md: 3, lg: 5 }} color="gray.500">

Fill up the Form to put your product to sale

</Text>

<Box py={{ base: 5, sm: 5, md: 8, lg: 10 }}>

<VStack pl={0} spacing={3} alignItems="flex-start">

<Button

size="md"

height="48px"

width="200px"

variant="ghost"

color="#DCE2FF"

\_hover={{ border: "2px solid #1C6FEB" }}

leftIcon={<MdPhone color="#1970F1" size="20px" />}

>

+91-988888888

</Button>

<Button

size="md"

height="48px"

width="200px"

variant="ghost"

color="#DCE2FF"

\_hover={{ border: "2px solid #1C6FEB" }}

leftIcon={<MdEmail color="#1970F1" size="20px" />}

>

hello@abc.com

</Button>

<Button

size="md"

height="48px"

width="200px"

variant="ghost"

color="#DCE2FF"

\_hover={{ border: "2px solid #1C6FEB" }}

leftIcon={

<MdLocationOn color="#1970F1" size="20px" />

}

>

Karnavati, India

</Button>

</VStack>

</Box>

</Box>

</WrapItem>

<WrapItem>

<Box bg="white" borderRadius="lg">

<Box m={8} color="#0B0E3F">

<VStack spacing={5}>

<FormControl id="name">

<FormLabel>Product Name</FormLabel>

<input type="text" className="name" />

</FormControl>

<FormControl id="name">

<FormLabel>Base Price</FormLabel>

<input type="text" className="base-price" />

</FormControl>

<FormControl id="name">

<FormLabel>Category</FormLabel>

<input type="text" className="cat" />

</FormControl>

<FormControl id="name">

<FormLabel>Last date</FormLabel>

<input type="date" className="date" />

</FormControl>

<FormControl id="name">

<FormLabel>Upload Product Image Link</FormLabel>

<input

type="type"

className="pro-img"

/>

</FormControl>

<FormControl id="name">

<FormLabel>Location</FormLabel>

<Textarea

borderColor="gray.300"

\_hover={{

borderRadius: "gray.300",

}}

placeholder="message"

className="location"

/>

</FormControl>

<FormControl id="name" float="right">

<Link to={/home} >

<Button

variant="solid"

bg="#0D74FF"

color="white"

\_hover={{}}

onClick={add}

>

Start Auction

</Button>

</Link>

</FormControl>

</VStack>

</Box>

</Box>

</WrapItem>

</Wrap>

</Box>

</Box>

</Flex>

</Container>

{/\* <div className="con">

<div className="name-con"></div>

<div className="base-price-con"></div>

<div className="img-con"></div>

<div className="Location-con">

<textarea rows={10} cols={70} className="location"></textarea>

</div>

<div className="cat-con"></div>

<div className="date-con"></div>

<div className="sub">

<button onClick={add}>Submit</button>

</div>

</div> \*/}

</>

);

}

**Bidding.jsx**

import React from 'react'

import { read } from '../App'

import { update, ref } from 'firebase/database';

import { Button } from '@chakra-ui/button';

export default function Bidding({ db, id }) {

async function bid(){

let bid1, bid2, bid3;

let nbid = document.querySelector('.bid').value;

let Bid = await read(Varshith/Products/${id})

// console.log(postid)

let date2 = new Date();

let Bid1 = Bid.Time.toString().split('T')[0].split('-')

if(date2.getMonth()+1 <= parseInt(Bid1[1])){

console.log('Done')

}

console.log(date2.getDate(),date2.getMonth()+1, date2.getFullYear() )

if(date2.getDate() <= parseInt(Bid1[2]) && date2.getMonth()+1 <= parseInt(Bid1[1]) && date2.getFullYear() <= parseInt(Bid1[0])){

console.log(id)

bid1 = Bid.Bid.Bid1.split('\_')[0];

bid2 = Bid.Bid.Bid2.split('\_')[0];

bid3 = Bid.Bid.Bid3.split('\_')[0];

if(nbid < bid3 || nbid < Bid.Base){

return

}

else if(nbid == bid1 || nbid == bid2 || nbid == bid3){

return

}

else if(nbid > bid1){

bid3 = bid2;

bid2 = bid1;

bid1 = nbid;

}

else if(nbid > bid2){

bid3 = bid2;

bid2 = nbid;

}

else if(nbid > bid3){

bid3 = nbid;

}

update(ref(db, Varshith/Products/${id}), {

Bid : {

Bid1 : ${bid1}\_User1,

Bid2 : ${bid2}\_User2,

Bid3 : ${bid3}\_User3,

}

})

}

}

return (

<>

<div className="biddingcontainer">

<div className="bidcon">

<input type="text" placeholder='Enter Your Bid' className='bid' />

</div>

<div className="bid-btncon">

<Button className='bid-btn' colorScheme='green' borderRadius={12} width={150} mt={2} onClick={bid}>Bid</Button>

</div>

</div>

</>

)

}

**Home.jsx**

import React, { useState, useEffect } from "react";

import { read } from "../App";

import {

Modal,

ModalOverlay,

ModalContent,

ModalHeader,

ModalFooter,

ModalBody,

ModalCloseButton,

Button,

useDisclosure,

} from "@chakra-ui/react";

import TextRating from "./TextRating";

import { Avatar, AvatarBadge, AvatarGroup } from '@chakra-ui/react'

import { Link } from "react-router-dom";

import { getStorage, ref as ref1,uploadBytes, listAll, getDownloadURL } from "firebase/storage";

export default function Home({db, s}) {

var logs;

const [cat, setCat] = useState("Price");

const [list, setList] = useState(false);

const { isOpen, onOpen, onClose } = useDisclosure();

const [imageList, setInameList] = useState([])

listAll(ref1(s, Email/Coins)).then((response) => {

response.items.forEach(async (item) => {

let url = await getDownloadURL(item)

setInameList((prev) => url);

return imageList

});

});

async function check() {

logs = await read("Varshith/Products");

logs = Object.entries(logs);

console.log(logs, "LOgs");

let slist = [];

let sea = document.querySelector(".search").value;

console.log(sea, "Sea");

if (cat == "Category") {

for (const los in logs) {

// console.log(logs[los][1].Cat)

if (logs[los][1].Cat == sea) {

slist.push(logs[los]);

}

}

} else if (cat == "Price") {

for (const los in logs) {

if (parseInt(logs[los][1].Base) <= parseInt(sea)) {

slist.push(logs[los]);

}

}

} else if (cat == "Location") {

for (const los in logs) {

if (logs[los][1].Location == sea) {

slist.push(logs[los]);

}

}

}

console.log(slist, "Slist");

setList(slist);

}

useEffect(() => {

async function fetchData() {

logs = await read("Varshith/Products");

logs = Object.entries(logs);

// console.log(logs)

setList(logs);

}

fetchData();

}, []);

return (

<>

<div className="search-con">

<div className="auction-cover">

<Link to={'/Add'}>

<Button colorScheme='green' borderRadius={10} mt={2} ml={2}>New Auction</Button>

</Link>

</div>

<div className="search-cover">

<Button

onClick={onOpen}

colorScheme="blue"

width={100}

borderRadius={10}

>

Filters

</Button>

<Modal blockScrollOnMount={false} isOpen={isOpen} onClose={onClose}>

<ModalOverlay />

<ModalContent>

<ModalCloseButton />

<div className="btns">

<div>

<button

onClick={() => {

setCat("Price");

document.querySelector('.search').value = '';

onClose()

}}

>

Price

</button>

</div>

<div>

<button

onClick={() => {

setCat("Location");

document.querySelector('.search').value = '';

onClose()

}}

>

Location

</button>

</div>

<div>

<button

onClick={() => {

setCat("Category");

document.querySelector('.search').value = '';

onClose()

}}

>

Category

</button>

</div>

</div>

</ModalContent>

</Modal>

<input type="text" className="search" placeholder={Enter ${cat}} />

<Button

onClick={check}

colorScheme="blue"

variant="outline"

className="search-btn"

>

Search

</Button>

</div>

</div>

<div className="home-con">

{list ? (

list.map((lil) => (

<div className="cont" key={lil} id={lil[0]}>

<div className="img-con">

<img

className="img"

src={lil[1].img}

// src="https://img.etimg.com/thumb/width-1200,height-1200,imgsize-37850,resizemode-75,msid-106580228/news/international/us/jujutsu-kaisen-preview-is-gojo-making-a-return-key-details-here.jpg"

alt=""

/>

</div>

<div className="content-con">

<div className="nameprice-cov">

<div className="cont1">{lil[0].split("\_")[1]}</div>

<TextRating rate = {lil[1].rate||0}/>

<div className="price-cov">₹ {lil[1].Base}</div>

</div>

<div className="namebid-cov">

<div className="name-cov">{lil[0].split("\_")[0]}</div>

<div className="bid-cov">

<Link key={lil} to={/products/${lil[0]}}>

<Button borderRadius={40} colorScheme="green">

Add/See Bid

</Button>

</Link>

</div>

<div className="refcover">

<Link to={/rev}>Reviews</Link>

</div>

</div>

</div>

</div>

))

) : (

<div className="loader-cover">

<div class="loader">

<div class="loader\_\_bar"></div>

<div class="loader\_\_bar"></div>

<div class="loader\_\_bar"></div>

<div class="loader\_\_bar"></div>

<div class="loader\_\_bar"></div>

<div class="loader\_\_ball"></div>

</div>

</div>

)}

</div>

</>

);

}

**PlaceBid.jsx**

import React, { useEffect, useState } from "react";

import { useParams } from "react-router";

import { read } from "../App";

import { Button } from "@chakra-ui/button";

import { Badge } from "@chakra-ui/layout";

import StarRating from "./TextRating";

import Bidding from "./Bidding";

import Swal from "sweetalert2";

import { update, ref } from 'firebase/database';

import ReviewPage from "./Review";

export default function PlaceBid({ db }) {

let bid1, bid2, bid3

const [sel, setBid3] = useState(0)

const [duedate, setdue] = useState("Bid")

const [lil, setLil] = useState(false);

const { postid } = useParams();

async function bid(){

let bid1, bid2, bid3;

let nbid = document.querySelector('.bid').value;

let Bid = await read(Varshith/Products/${postid})

// console.log(postid)

let date2 = new Date();

let Bid1 = Bid.Time.toString().split('T')[0].split('-')

if(date2.getMonth()+1 <= parseInt(Bid1[1])){

console.log('Done')

}

console.log(date2.getDate(),date2.getMonth()+1, date2.getFullYear() )

if(date2.getDate() <= parseInt(Bid1[2]) && date2.getMonth()+1 <= parseInt(Bid1[1]) && date2.getFullYear() <= parseInt(Bid1[0])){

console.log(postid)

bid1 = Bid.Bid.Bid1.split('\_')[0];

bid2 = Bid.Bid.Bid2.split('\_')[0];

bid3 = Bid.Bid.Bid3.split('\_')[0];

if(nbid < bid3 || nbid < Bid.Base){

Swal.fire({

icon: "error",

title: "Oops...",

text: "Your Bid was lower than Base price",

// footer: '<a href="#">Why do I have this issue?</a>'

});

return

}

else if(nbid == bid1 || nbid == bid2 || nbid == bid3){

Swal.fire({

icon: "error",

title: "Oops...",

text: "Your Bid is Equal to an other Bid",

// footer: '<a href="#">Why do I have this issue?</a>'

});

return

}

else if(nbid > bid1){

bid3 = bid2;

bid2 = bid1;

bid1 = nbid;

setBid3(prev => prev+1)

}

else if(nbid > bid2){

bid3 = bid2;

bid2 = nbid;

setBid3(prev => prev+1)

}

else if(nbid > bid3){

bid3 = nbid;

setBid3(prev => prev+1)

}

update(ref(db, Varshith/Products/${postid}), {

Bid : {

Bid1 : ${bid1}\_User1,

Bid2 : ${bid2}\_User2,

Bid3 : ${bid3}\_User3,

}

})

Swal.fire({

title: "Bid Succesfull",

text: Your Bid for ${nbid} is succeaafull,

icon: "success"

});

}

else{

setdue("Due Date Over!")

Swal.fire({

icon: "Bid Failed",

title: "Oops...",

text: "Due Date For Bid is Completed",

// footer: '<a href="#">Why do I have this issue?</a>'

});

}

}

useEffect(() => {

async function fetchData() {

let logs = await read(Varshith/Products/${postid});

console.log(logs);

setLil(logs);

}

fetchData();

}, [postid, sel]);

return (

<>

{lil ? (

<>

<div className="con-overlay" style={{zIndex: '9999', backgroundImage: url(${lil.img}), backgroundSize: 'contain', backgroundPosition: 'center', position: 'absolute', filter: 'blur(8px),', }}>

</div>

<div className="con" >

<div className="main-con">

<div className="main-head"></div>

<div className="main-hero">

<div className="cont">

<div className="img-con">

<img

className="img"

src = {lil.img}

// src="https://store.storeimages.cdn-apple.com/4668/as-images.apple.com/is/iphone-card-40-iphone15prohero-202309\_FMT\_WHH?wid=508&hei=472&fmt=p-jpg&qlt=95&.v=1693086369818"

alt=""

/>

</div>

<div className="content-con">

<div className="nameprice-cov">

<div className="cont1">{postid.split("\_")[1]}</div>

<div className="price-cov">₹ {lil.Base}</div>

</div>

<div className="namebid-cov">

<div className="name-cov">{postid.split("\_")[0]}</div>

<div className="bid-cov"></div>

</div>

</div>

</div>

<div className="bid-con">

<div className="bidname">

<div className="bid1-con">

<Badge colorScheme="green">

₹ {lil.Bid.Bid1.split("\_")[0]}

</Badge>

<span className="userbid">

{lil.Bid.Bid1.split("\_")[1]}

</span>

</div>

<div className="bid2-con">

<Badge colorScheme="green">

₹ {lil.Bid.Bid2.split("\_")[0]}

</Badge>

<span className="userbid">

{lil.Bid.Bid2.split("\_")[1]}

</span>

</div>

<div className="bid3-con">

<Badge colorScheme="green">

₹ {lil.Bid.Bid3.split("\_")[0]}

</Badge>

<span className="userbid">

{lil.Bid.Bid3.split("\_")[1]}

</span>

</div>

</div>

<div class="notifications-container">

<div class="success">

<div class="flex">

<div class="flex-shrink-0">

{/\* <svg

class="succes-svg"

xmlns="http://www.w3.org/2000/svg"

viewBox="0 0 20 20"

fill="currentColor"

aria-hidden="true"

>

<path

fill-rule="evenodd"

d="M10 18a8 8 0 100-16 8 8 0 000 16zm3.707-9.293a1 1 0 00-1.414-1.414L9 10.586 7.707 9.293a1 1 0 00-1.414 1.414l2 2a1 1 0 001.414 0l4-4z"

clip-rule="evenodd"

></path>

</svg> \*/}

</div>

<div class="success-prompt-wrap">

<p class="success-prompt-heading">

<span className="timename">Last Date : </span>

</p>

<div class="success-prompt-prompt">

{lil.Time.split("T")[0]}

</div>

</div>

<div class="success-prompt-wrap">

<p class="success-prompt-heading">

<span className="Authorname">Owner : </span>

</p>

<div class="success-prompt-prompt">{lil.Author}</div>

</div>

<div class="success-prompt-wrap">

<p class="success-prompt-heading">

<span className="Catname">Category : </span>

</p>

<div class="success-prompt-prompt">{lil.Cat}</div>

</div>

<div class="success-prompt-wrap">

<p class="success-prompt-heading">

<span className="locationname">Location : </span>

</p>

<div class="success-prompt-prompt">{lil.Location}</div>

</div>

</div>

</div>

</div>

<div className="Rate-con">

<StarRating cus={1} db={db} path={postid} />

</div>

<div className="Rate-con">

<div className="biddingcontainer">

<div className="bidcon">

<input

type="text"

placeholder="Enter Your Bid"

className="bid"

/>

</div>

<div className="bid-btncon">

<Button

className="bid-btn"

colorScheme="green"

borderRadius={12}

width={150}

mt={2}

onClick={bid}

>

{duedate}

</Button>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</>

) : (

<div className="loader-cover">

<div class="loader">

<div class="loader\_\_bar"></div>

<div class="loader\_\_bar"></div>

<div class="loader\_\_bar"></div>

<div class="loader\_\_bar"></div>

<div class="loader\_\_bar"></div>

<div class="loader\_\_ball"></div>

</div>

</div>

)}

<h1 style={{ marginLeft: '90px',marginTop:'30px', fontSize: '36px' }}>Reviews</h1>

<ReviewPage db = {db} id = {postid} />

</>

);

}

**Review.jsx**

import { Button, useStatStyles } from '@chakra-ui/react';

import { ref, update } from 'firebase/database';

import React, { useEffect, useState } from 'react';

import { read } from '../App';

const ReviewPage = ({id, db}) => {

// Array of random image URLs

const imageUrls = [];

const [rev, setRev] = useState(false)

const [k1, setK] = useState(0)

// Generate image URLs with increasing last digit for both men and women

for (let i = 1; i <= 8; i++) {

const urlMen = https://randomuser.me/api/portraits/men/${i}.jpg;

const urlWomen = https://randomuser.me/api/portraits/women/${i}.jpg;

imageUrls.push(urlMen);

imageUrls.push(urlWomen);

}

useEffect(()=>{

async function fetchData(){

let k = await read(Varshith/Products/${id})

setRev(k.review.split('\_'))

console.log(k.review.split('\_'))

}

fetchData()

}, [k1])

// Array of random usernames

const usernames = [

'Alice',

'Bob',

'Charlie',

'David',

'Emma',

'Frank',

'Grace',

'Henry',

'Ivy',

'Jack',

'Kate',

'Liam',

'Mia',

'Nora',

'Oscar',

'Penny',

'Quinn',

'Riley',

'Sam',

'Taylor',

'Uma',

'Victor',

'Wendy',

'Xavier',

'Yara',

'Zane'

];

async function addReview(){

let r = document.querySelector('.reviewaddtext').value

let prev = await read(Varshith/Products/${id}) || ""

if(prev.review != ''){

r = ${prev.review}\_${r}

}

update(ref(db, Varshith/Products/${id}),{

review : r

})

}

// Function to get a random element from an array

const getRandomElement = (array) => {

return array[Math.floor(Math.random() \* array.length)];

};

// JSX for a single review card

const renderReviewCard = (ro) => {

const imageUrl = getRandomElement(imageUrls);

const username = getRandomElement(usernames);

const rating = ${Math.floor(Math.random() \* 5) + 1}/5; // Generate random rating

return (

<div style={reviewCardStyle}>

<img src={imageUrl} alt="User" style={imageStyle} />

<div style={ratingStyle}>{rating}</div>

<div style={reviewTextStyle}>{ro || "Amazing product! I highly recommend it."}</div>

<div style={usernameStyle}>- {username}</div>

</div>

);

};

const renderReviewCard1 = (ro) => {

const imageUrl = getRandomElement(imageUrls);

const rating = ${Math.floor(Math.random() \* 5) + 1}/5; // Generate random rating

return (

<div style={reviewCardStyle}>

<img src={imageUrl} alt="User" style={imageStyle} />

<div style={ratingStyle}>{rating}</div>

<div style={reviewTextStyle}>{ro}</div>

<div style={usernameStyle}>- {'User'}</div>

</div>

);

};

return (

<>

<div className='reviewadd'>

<textarea rows={5} cols={100} placeholder='Enter Review Here...' className='reviewaddtext' ></textarea>

<Button width={200} colorScheme='green' onClick={()=>{

setK(prev => prev + 1)

addReview()

}}>Submit</Button>

</div>

<div className='myrev'>

{Array.isArray(rev) && rev.map((ro, index) => (

<div key={index}>{renderReviewCard(ro)}</div>

))}

</div>

<div style={{ display: 'flex', flexWrap: 'wrap', justifyContent: 'center' }}>

{/\* Render multiple review cards \*/}

{Array.from({ length: 8 }).map((\_, index) => (

<React.Fragment key={index}>

{renderReviewCard()}

</React.Fragment>

))}

</div>

</>

);

};

const reviewCardStyle = {

width: '300px',

backgroundColor: '#f0f0f0',

border: '1px solid #ccc',

borderRadius: '8px',

padding: '20px',

margin: '20px',

boxShadow: '0px 2px 4px rgba(0, 0, 0, 0.1)',

};

const imageStyle = {

width: '100px',

height: '100px',

borderRadius: '50%',

marginBottom: '10px',

};

const ratingStyle = {

fontSize: '24px',

fontWeight: 'bold',

marginBottom: '10px',

};

const reviewTextStyle = {

fontSize: '16px',

};

const usernameStyle = {

fontSize: '14px',

fontStyle: 'italic',

marginTop: '10px',

};

export default ReviewPage;

**TextRating.jsx**

import { ref, update } from '@firebase/database';

import React, { useState } from 'react';

import Rating from 'react-rating';

const StarRating = ({rate, cus, db, path}) => {

const [rating, setRating] = useState(rate);

const handleRatingChange = (value) => {

update(ref(db, Varshith/Products/${path}), {

rate : value

})

setRating(value);

};

return (

<div>

{/\* <h2>Star Rating</h2> \*/}

{cus ? (

<>

<span> {rating}</span>

<Rating

initialRating={rating}

fullSymbol={<svg xmlns="http://www.w3.org/2000/svg" width="16" height="16" fill="gold" class="bi bi-star-fill" viewBox="0 0 16 16">

<path d="M3.612 15.443c-.386.198-.824-.149-.746-.592l.83-4.73L.173 6.765c-.329-.314-.158-.888.283-.95l4.898-.696L7.538.792c.197-.39.73-.39.927 0l2.184 4.327 4.898.696c.441.062.612.636.282.95l-3.522 3.356.83 4.73c.078.443-.36.79-.746.592L8 13.187l-4.389 2.256z"/>

</svg>}

emptySymbol={<svg xmlns="http://www.w3.org/2000/svg" width="16" height="16" fill="currentColor" class="bi bi-star" viewBox="0 0 16 16">

<path d="M2.866 14.85c-.078.444.36.791.746.593l4.39-2.256 4.389 2.256c.386.198.824-.149.746-.592l-.83-4.73 3.522-3.356c.33-.314.16-.888-.282-.95l-4.898-.696L8.465.792a.513.513 0 0 0-.927 0L5.354 5.12l-4.898.696c-.441.062-.612.636-.283.95l3.523 3.356-.83 4.73zm4.905-2.767-3.686 1.894.694-3.957a.56.56 0 0 0-.163-.505L1.71 6.745l4.052-.576a.53.53 0 0 0 .393-.288L8 2.223l1.847 3.658a.53.53 0 0 0 .393.288l4.052.575-2.906 2.77a.56.56 0 0 0-.163.506l.694 3.957-3.686-1.894a.5.5 0 0 0-.461 0z"/>

</svg>}

onChange={handleRatingChange}

/>

</>

):(

<Rating

initialRating={rating}

readonly

fullSymbol={<svg xmlns="http://www.w3.org/2000/svg" width="16" height="16" fill="gold" class="bi bi-star-fill" viewBox="0 0 16 16">

<path d="M3.612 15.443c-.386.198-.824-.149-.746-.592l.83-4.73L.173 6.765c-.329-.314-.158-.888.283-.95l4.898-.696L7.538.792c.197-.39.73-.39.927 0l2.184 4.327 4.898.696c.441.062.612.636.282.95l-3.522 3.356.83 4.73c.078.443-.36.79-.746.592L8 13.187l-4.389 2.256z"/>

</svg>}

emptySymbol={<svg xmlns="http://www.w3.org/2000/svg" width="16" height="16" fill="currentColor" class="bi bi-star" viewBox="0 0 16 16">

<path d="M2.866 14.85c-.078.444.36.791.746.593l4.39-2.256 4.389 2.256c.386.198.824-.149.746-.592l-.83-4.73 3.522-3.356c.33-.314.16-.888-.282-.95l-4.898-.696L8.465.792a.513.513 0 0 0-.927 0L5.354 5.12l-4.898.696c-.441.062-.612.636-.283.95l3.523 3.356-.83 4.73zm4.905-2.767-3.686 1.894.694-3.957a.56.56 0 0 0-.163-.505L1.71 6.745l4.052-.576a.53.53 0 0 0 .393-.288L8 2.223l1.847 3.658a.53.53 0 0 0 .393.288l4.052.575-2.906 2.77a.56.56 0 0 0-.163.506l.694 3.957-3.686-1.894a.5.5 0 0 0-.461 0z"/>

</svg>}

onChange={handleRatingChange}

/>

)}

</div>

);

};

export default StarRating;