



# NORTH SOUTH UNIVERSITY

*Center of Excellence in Higher Education*

## CSE299: Junior Design

### Week: 01 Progress Report

**Submitted to:** Tanzilah Noor Shabnam  
**Submitted on:** 07/10/2024  
**Team Name:** Machine Minds  
**Project Title:** Flood Guard



**Task completed:** Homepage UI

#### Team members

Name	ID
Sabbir Hossain	2131272042
Md. Misbah Khan	2132089642
Aritra Islam Saswato	2132629642

# Languages and Tools

**Programming language:** JavaScript, typescript, HTML, CSS

**Framework:**

Name	Language
React	JavaScript and typescript
Tailwind CSS	CSS

**React packages:**

Name	Purpose
Context API	Passing data between components
Axios	Fetching external APIs
Dotenv	Securing API keys and sensitive information
React-icon, Lucid icon	Using icons
Sonner	Producing toast
Rramer-motion	Animating component

**Component Library:**

Name	Links
Shadcn/ui	<a href="https://ui.shadcn.com/">https://ui.shadcn.com/</a>
Aceternity UI	<a href="https://ui.aceternity.com/">https://ui.aceternity.com/</a>

**Collaboration and storing code:** GitHub

**Contribution:**

Name	Contribution
Md. Misbah Khan	Environment setup, Weather Dashboard, Navbar, Footer, Developer Page, Not found Page
Sabbir Hossain	Precaution, Latest News, Impact
Aritra Islam Saswato	Donation, Emergency Contact, FAQ

**External API:**

Name	Use	Links
Open Cage	converting latitude and longitude to location name	<a href="https://opencagedata.com/api">https://opencagedata.com/api</a>
Weather data	getting current weather data and 5 days forecast by location name	<a href="https://www.weatherapi.com/">https://www.weatherapi.com/</a>

## Code Snippet

```
● ● ●

//App.jsx

// @ts-nocheck

import { Toaster } from "sonner";
import { router } from "./Routes/Routes";
import { RouterProvider } from "react-router-dom";

function App() {
  return (
    <section className="bg-secondary font-nunito">
      <RouterProvider router={router} />
      <Toaster />
    </section>
  );
}

export default App;

//main.jsx

import React from "react";
import ReactDOM from "react-dom/client";
import App from "./App.jsx";
import "./styles/globals.css";
import { WeatherProvider } from "./context/WeatherContext.jsx";

ReactDOM.createRoot(document.getElementById("root")).render(
  <React.StrictMode>
    <WeatherProvider>
      <App />
    </WeatherProvider>
  </React.StrictMode>
);
```

Figure 1: App.tsx & main.tsx

```

1 // src/components/WeatherDashboard.jsx
2 // @ts-ignore
3 import React from "react";
4 import { FaThermometerHalf, FaCloudSunRain } from "react-icons/fa";
5 import { CirclesWithBar } from "react-loader-spinner";
6 import { useWeather } from "../../context/WeatherContext";
7 import { weatherData } from "./weatherData";
8
9 const WeatherDashboard = () => {
10   const { currentWeatherData, loading, error } = useWeather();
11
12   if (loading) {
13     return (
14       <div className="flex items-center justify-center h-screen w-full py-4">
15         <CirclesWithBar
16           height="100"
17           width="100"
18           colors="#4fa94d"
19           outerCircleColor="#c4c4d5"
20           innerCircleColor="#c4c4d5"
21           barColor="#c4c4d5"
22           ariaLabel="circles-with-bar-loading"
23           wrapperStyle={}
24           wrapperClass=""
25           visible={true}
26         />
27       </div>
28     );
29   }
30
31   if (error) {
32     return <div className="text-center py-4 text-secondary">(error)</div>;
33   }
34
35   if (!currentWeatherData) {
36     return (
37       <div className="text-center text-primary py-4">
38         No weather data available.
39       </div>
40     );
41   }
42
43   const { current, forecast } = currentWeatherData;
44   const { forecastday } = forecast;
45
46   const currentWeatherInfo = weatherData(current);
47
48   return (
49     <section className="my-6">
50       <h1 className="text-5xl text-primary font-museo text-center py-4">
51         Weather Dashboard
52       </h1>
53       <div className="relative rounded-lg mt-12 flex h-[90vh] overflow-hidden">
54         <video
55           className="absolute top-0 left-0 w-full h-full object-cover"
56           autoPlay
57           loop
58           muted
59           playsInline
60         >
61           <source src="/BgVideo/BG-Video.mp4" type="video/mp4" />
62           Your browser does not support the video tag.
63         </video>
64         <div className="absolute top-0 left-0 w-full h-full bg-secondary bg-opacity-20"></div>
65         <div className="grid grid-cols-1 py-2 md:grid-cols-3 w-full">
66           /* Current Weather Section */
67           <div className="md:grid-col-1 flex justify-center items-center">
68             <div className="bg-secondary bg-opacity-30 backdrop-blur-md shadow-lg rounded-lg p-6 max-w-md mx-auto">
69               <div className="flex flex-col items-center">
70                 <div className="text-xxl font-thin text-primary">
71                   {currentWeatherData.location.name}
72                 </div>
73                 <div className="text-xxl font-light text-primary">Today</div>
74                 <img
75                   src={current.condition.icon}
76                   alt={current.condition.text}
77                   className="w-16 h-16"
78                 />
79                 <p className="text-lg text-primary mb-4">
80                   {current.condition.text}
81                 </p>
82                 <div className="grid grid-cols-1 gap-2 w-full">
83                   {currentWeatherInfo.map((info) => (
84                     <div
85                       key={info.id}
86                       className="flex items-center px-4 py-2 bg-primary text-secondary shadow-md rounded-lg transition-transform transform hover:scale-105">
87                       <img icon className="text-2xl mr-2" />
88                       <span className="el-2 text-ig">
89                         {info.label}: {info.value}
90                       </span>
91                     </div>
92                   )));
93                 </div>
94               </div>
95             </div>
96           </div>
97         </div>
98
99         /* Weather Forecast Section */
100        <div className="md:grid-span-2 flex justify-center items-center container mx-auto py-4 px-4">
101          <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-4">
102            {forecastday.map((day) => {
103              const date = new Date(day.date);
104              const options = { weekday: "long" };
105              const dayName = date.toLocaleDateString("en-US", options);
106
107              return (
108                <div
109                  key={day.date_epoch}
110                  className="bg-secondary bg-opacity-30 backdrop-blur-md p-6 rounded-lg shadow-md flex flex-col items-center transition duration-300 ease-in-out transform hover:scale-105 hover:bg-opacity-20">
111                  <h3 className="text-xxl text-primary font-thin mb-2">
112                    {dayName}
113                  </h3>
114                  <img
115                    src={day.condition.icon}
116                    alt={day.condition.text}
117                    className="w-16 h-16"
118                  />
119                  <p className="text-lg text-primary font-light mb-2">
120                    {day.condition.text}
121                  </p>
122                  <div className="text-primary">
123                    <div className="flex justify-center font-thin items-center">
124                      <FaThermometerHalf className="text-primary me-2" /> Max
125                      Temp: {day.maxtemp_c}°C
126                    </div>
127                    <div className="flex justify-center font-thin items-center">
128                      <FaThermometerHalf className="text-primary me-2" /> Min
129                      Temp: {day.mintemp_c}°C
130                    </div>
131                    <div className="flex justify-center font-thin items-center">
132                      <FaCloudSunRain className="text-primary me-2" />
133                      Chance of Rain: {day.daily_chance_of_rain}%
134                    </div>
135                  </div>
136                </div>
137              </div>
138            );
139          );
140        </div>
141      </div>
142    </div>
143  </div>
144);
145
146 export default WeatherDashboard;

```

Figure 2: WeatherDashboard.tsx

```
// src/data/weatherData.js
import { FaThermometerHalf, FaCity, FaWind, FaCloudRain } from
"react-icons/fa";
import { FaMaskVentilator } from "react-icons/fa6";
import { GiRaining } from "react-icons/gi";

export const weatherData = (current) => [
  {
    id: 1,
    label: "Temperature",
    value: `${current.temp_c}°C`,
    icon: FaThermometerHalf, // Importing icon
  },
  {
    id: 2,
    label: "Feels Like",
    value: `${current.feelslike_c}°C`,
    icon: FaCity, // Importing icon
  },
  {
    id: 3,
    label: "Wind",
    value: `${current.wind_kph} km/h`,
    icon: FaWind, // Importing icon
  },
  {
    id: 4,
    label: "Humidity",
    value: `${current.humidity}%`,
    icon: FaCloudRain, // Importing icon
  },
  {
    id: 5,
    label: "AQI",
    value: `${current.air_quality["us-epa-index"]}`,
    icon: FaMaskVentilator, // Importing icon
  },
  {
    id: 6,
    label: "Precipitation",
    value: `${current.precip_mm}mm`,
    icon: GiRaining, // Importing icon
  },
];
```

Figure 3:WeatherData.tsx

```

1 // src/context/WeatherContext.jsx
2 import React, { createContext, useContext, useState, useEffect } from "react";
3
4 const WeatherContext = createContext();
5
6 export const useWeather = () => useContext(WeatherContext);
7
8 export const WeatherProvider = ({ children }) => {
9   const [location, setLocation] = useState(null);
10  const [currentWeatherData, setCurrentWeatherData] = useState(null);
11  const [loading, setLoading] = useState(true);
12  const [error, setError] = useState("");
13
14  const weatherApiKey = import.meta.env.VITE_WEATHER_API_KEY;
15  const openCageApiKey = import.meta.env.VITE_OPENCAGE_API_KEY;
16
17  const getUserLocation = () => {
18    return new Promise((resolve, reject) => {
19      if (navigator.geolocation) {
20        navigator.geolocation.getCurrentPosition(
21          (position) => {
22            const { latitude, longitude } = position.coords;
23
24            resolve({ latitude, longitude });
25          },
26          (error) => {
27            reject(error);
28          }
29        );
30      } else {
31        const error = new Error(
32          "Geolocation is not supported by this browser."
33        );
34        reject(error);
35      }
36    });
37  };
38
39  const getLocationFromCoordinates = async (latitude, longitude) => {
40    try {
41      const response = await fetch(
42        `https://api.opencagedata.com/geocode/v1/json?q=${latitude}&${longitude}&key=${openCageApiKey}`
43      );
44      const data = await response.json();
45
46      if (data.results.length > 0) {
47        const city =
48          data.results[0].components.city || data.results[0].components.country;
49
50        return city || "Location not found";
51      } else {
52        throw new Error("No results found");
53      }
54    } catch (error) {
55      console.error("Geocoding error:", error);
56      throw new Error("Geocoding error: " + error.message);
57    }
58  };
59
60  useEffect(() => {
61    const fetchWeatherData = async () => {
62      setLoading(true);
63      setError("");
64      try {
65        const { latitude, longitude } = await getUserLocation();
66        const locationName = await getLocationFromCoordinates(
67          latitude,
68          longitude
69        );
70        setLocation(locationName);
71
72        const weatherResponse = await fetch(
73          `https://api.weatherapi.com/v1/forecast.json?key=${weatherApiKey}&q=${locationName}&days=6&aqi=yes`
74        );
75        const weatherData = await weatherResponse.json();
76
77        if (!weatherData || weatherData.error) {
78          throw new Error(weatherData.error.message);
79        }
80
81        setCurrentWeatherData(weatherData);
82      } catch (error) {
83        setError("Error fetching weather data: " + error.message);
84      } finally {
85        setLoading(false);
86      }
87    };
88
89    fetchWeatherData();
90  }, [weatherApiKey, openCageApiKey]);
91
92  return (
93    <WeatherContext.Provider
94      value={{ location, currentWeatherData, loading, error }}
95    >
96      {children}
97      </WeatherContext.Provider>
98    );
99  };
100

```

Figure 4: WeatherContext.tsx

```
// @ts-ignore
import React from "react";
import { Link } from "react-router-dom";

const NotFound = () => {
  return (
    <div className=
"flex items-center justify-center max-w-2xl mx-auto h-scr
een bg-secondary text-primary"
>
    <div className=
"flex flex-col md:flex-row items-center space-y-4 md:space
y-0"
>
      {/* Left part - Big 404 */}
      <div className="text-8xl font-thin font-museo">
        404</div>

      {/* Vertical line */}
      <div className="h-44 w-px bg-primary mx-4"></div>

      {/* Right part - Description */}
      <div className="text-center md:text-left">
        <h1 className=
"text-4xl font-semibold font-museo mb-2">
          Page Not Found
        </h1>
        <p className="text-lg my-4">
          The page you are looking for might have been
          removed, had its name
          changed, or is temporarily unavailable.
        </p>
        <Link to="/" className=
"bg-secondary text-primary underline">
          Go to Home
        </Link>
      </div>
    </div>
  );
};

export default NotFound;
```

Figure 5: NotFound.tsx

```
// @ts-ignore
import React from "react";
import { Link } from "react-router-dom";
import { NavLink } from "./NavLink";

const Navbar = () => {
  return (
    <div className=
"bg-primary opacity-95 flex justify-between text-secondar
y"
>
  <Link to="/">
    
  </Link>
  <div className=
"flex w-1/3 items-center text-2xl font-museo justify-betw
een me-20"
>
  {NavLink.map(({ name, path }, idx) => (
    <Link
      to={path}
      key={idx}
      className=
"relative text-secondary text-decoration-none fs-4 group"
>
      {name}
      <span className=
"absolute bottom-0 left-0 w-full h-0.5 bg-secondary trans
form scale-x-0 group-hover:scale-x-100 origin-left transi
tion-transform duration-300"
></span>
    </Link>
  )));
  </div>
</div>
);
};

export default Navbar;
```

Figure 6: Navbar.tsx

```
// @ts-ignore
import React from "react";
import { Link } from "react-router-dom";
import { FooterLinks } from "./FooterLinks";

const Footer = () => {
  return (
    <footer className=
"bg-primary text-secondary py-6 mt-10 pb-0 mb-0">
      <div className=
"container mx-auto px-3 flex flex-col items-center">
        <div className=
"flex flex-col md:flex-row justify-between w-full max-w-6
xl">
        >
          <div className="mb-4 md:mb-0">
            <h2 className="text-4xl font-museo">
              flood guard</h2>
            <p className="mt-4 text-sm">
              Your trusted source for rescue management a
nd flood prediction.
            </p>
          </div>
          <div className=
"flex flex-col md:flex-row space-y-2 md:space-y-0 md:space
x-4">
          >
            {FooterLinks.map((link) => (
              <a
                key={link.name}
                href={link.path}
                target="_blank"
                rel="noopener noreferrer"
                className="hover:underline"
              >
                {link.name}
              </a>
            ))}
          </div>
        </div>
        <div className=
"mt-6 border-t border-secondary w-full max-w-6xl">
          <p className="text-center text-sm py-4">
            © {new Date().getFullYear()}
          </p>
        </div>
      </div>
    );
};

export default Footer;
```

Figure 7: Footer.tsx

```
//FooterLinks.js

export const FooterLinks = [
  {
    name: "About",
    path:
    "https://github.com/MisbahKhan0009/Flood-Guard/wiki",
  },
  { name: "Github", path:
    "https://github.com/MisbahKhan0009/Flood-Guard" },
    { name: "Developers", path: "/developers" },
    { name: "News", path: "/news" },
];
}

//navLinks.js
export const NavLinks = [
  { name: "Community Portal", path: "/community-portal"
  },
  { name: "Flood Prediction", path: "/flood-prediction"
  },
];
```

Figure 8: FooterLinks.js & navBarLink.js



```
import { createBrowserRouter } from "react-router-dom";
import Main from "../Layout/Main";
import NotFound from "../Shared/NotFound/NotFound.tsx";
import HomePage from "../Modules/HomePage/HomePage";
import DeveloperPage from
"../Shared/Developers/Developer.tsx";

export const router = createBrowserRouter([
{
  path: "/",
  element: <Main />,
  children: [
    {
      path: "/",
      element: <HomePage />,
    },
    {
      path: "/developers",
      element: <DeveloperPage />,
    },
    {
      path: "*",
      element: <NotFound />,
    },
  ],
},
]);
```

Figure 9: Routes.jsx



```
import React from "react";
import Donation from "./Donation/Donation";
import Stats from "./Stats/Stats";
import LatestNewsSection from "./News/LatestNewsSection";
import PrecautionSection from
"./Precaution/PrecautionSection";
import WeatherDashboard from "./Weather/WeatherDashboard"
;
import FAQSection from "./FAQ/FAQSection";
import EmergencyContactSection from
"./EmergencyContact/EmergencyContactSection";
// import { BackgroundBeamsWithCollision } from "@/components/ui/background-beams-with-collision";

const HomePage = () => {
  return (
    <section className="w-10/12 mx-auto" >
      {
        /* <BackgroundBeamsWithCollision className="p-0 m-0" > */
        <WeatherDashboard className="backdrop-blur-md" />

        <Stats className="backdrop-blur-md" />
        <LatestNewsSection className="backdrop-blur-md" />
        <PrecautionSection className="backdrop-blur-md" />
        <Donation className="backdrop-blur-md" />
        <EmergencyContactSection className=
"backdrop-blur-md" />
        <FAQSection className="backdrop-blur-md" />
      }
    </section>
  );
};

export default HomePage;
```

Figure 10: HomePage.tsx

```

import { FaPeopleCarry, FaMoneyBillWave, FaUsers } from
"react-icons/fa";

export const statsData = [
{
  icon: FaPeopleCarry,
  value: "3.3k+",
  color: "text-blue-600",
  description: "People Rescued Till Now",
  gradient: "from-blue-50 to-blue-100",
},
{
  icon: FaMoneyBillWave,
  value: "10Cr.",
  color: "text-green-600",
  description: "Taka Donation Received",
  gradient: "from-green-50 to-green-100",
},
{
  icon: FaUsers,
  value: "10k+",
  color: "text-purple-600",
  description: "Daily Users",
  gradient: "from-purple-50 to-purple-100",
},
];

```

Figure 11:StatsData.js

```

// @ts-ignore
import { statsData } from "./statsData";

const StatsCards = () => {
  return (
    <section className=
"flex flex-col justify-center items-center my-12">
      <h1 className=
"text-5xl text-primary font-museo text-center pt-12">
        Our Impact
      </h1>
      <div className=
"grid grid-cols-1 md:grid-cols-3 gap-8 w-10/12 mx-auto te
xt-center py-12"
    >
      {statsData.map((stat, index) => (
        <div
          key={index}
          className={`p-6 bg-gradient-to-r ${stat.
gradient}
rounded-lg shadow-lg hover:shadow-xl transform hover:sc
le-105 transition duration-300`}
        >
          <stat.icon className={`${stat.color}
text-5xl mx-auto mb-4`} />{" "}
          <p className={`text-5xl font-extrabold ${stat
.color}`}>
            {stat.value}
          </p>
          <p className="text-md text-gray-500 mt-2">{stat
.description}</p>
        </div>
      ))}
    </div>
  </section>
);
};

export default StatsCards;

```

Figure 12: Stats.tsx

```
// src/components/PrecautionSection.jsx
// @ts-ignore

import React from "react";
import { precautionData } from "./precautionData";

const PrecautionSection = () => {
  return (
    <section className=
"bg-primary bg-opacity-15 rounded-lg py-12 my-12">
      <h1 className=
"text-5xl font-museo text-center mb-8 text-primary ">
        Precautions for Flood
      </h1>
      <div className=
"grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-8 w-1
0/12 mx-auto text-center">
        {precautionData.map((precaution) => (
          <div
            key={precaution.id}
            className={`p-6 bg-gradient-to-r ${precaution
.gradient}
rounded-lg shadow-lg hover:shadow-xl transform hover:sc
ale-105 transition duration-300`}>
            <precaution.icon
              className={`${precaution.text
|| "text-secondary"} mx-auto`}>
            </precaution.icon>
            <p
              className={`text-2xl font-muse ${precaution
.text || "text-secondary"}`}>
              {precaution.title}
            </p>
            <p
              className={`text-md ${precaution.text ||
"text-secondary"} mt-2`}>
              {precaution.description}
            </p>
          </div>
        )));
      </div>
    </section>
  );
};

export default PrecautionSection;
```

Figure 13:Precaution.tsx



```
1 // src/data/precautionData.js
2 import {
3   FiArrowUpCircle,
4   FiAlertTriangle,
5   FiRadio,
6   FiBox,
7   FiPower,
8   FiUsers,
9 } from "react-icons/fi";
10
11 export const precautionData = [
12   {
13     id: 1,
14     title: "Move to Higher ground",
15     description:
16       "Immediately move to higher ground or stay on high ground during a flood.",
17     icon: FiArrowUpCircle,
18     gradient: "from-red-50 to-red-50",
19     text: "text-red-700",
20   },
21   {
22     id: 2,
23     title: "Avoid Flooded Areas",
24     description:
25       "Do not walk or drive through floodwaters to avoid being swept away.",
26     icon: FiAlertTriangle,
27     gradient: "from-teal-50 to-teal-50",
28     text: "text-teal-700",
29   },
30   {
31     id: 3,
32     title: "Get Weather Updates",
33     description:
34       "Stay updated on weather forecasts and flood warnings via radio or apps.",
35     icon: FiRadio,
36     gradient: "from-blue-50 to-blue-50",
37     text: "text-blue-700",
38   },
39   {
40     id: 4,
41     title: "Prepare Emergency Kit",
42     description:
43       "Ensure you have an emergency kit with essential supplies like food and water.",
44     icon: FiBox,
45     gradient: "from-green-50 to-green-50",
46     text: "text-green-700",
47   },
48   {
49     id: 5,
50     title: "Turn Off Utilities",
51     description:
52       "Turn off utilities such as electricity and gas to prevent electrical hazards.",
53     icon: FiPower,
54     gradient: "from-purple-50 to-purple-50",
55     text: "text-purple-700",
56   },
57   {
58     id: 6,
59     title: "Help Others",
60     description:
61       "Check on neighbors, especially the elderly or disabled, during a flood.",
62     icon: FiUsers,
63     gradient: "from-pink-50 to-pink-50",
64     text: "text-pink-700",
65   },
66 ];
67
```

Figure 14:PrecautionData.js

```
// src/components/LatestNewsSection.jsx
// @ts-ignore
import React from "react";
import { newsData } from "./newsData";

const LatestNewsSection = () => {
  return (
    <section className=
"bg-primary bg-opacity-15 py-12 my-12 rounded-lg">
      <h1 className=
"text-5xl font-museo text-center mb-8 text-primary">
        Latest News
      </h1>
      <div className="w-9/12 mx-auto space-y-8">
        {newsData.map((news) => (
          <div
            key={news.id}
            className=
"flex flex-col md:flex-row items-center bg-primary shadow-
lg rounded-lg overflow-hidden transition hover:shadow-x
l"
          >
            <div className="md:w-1/3">
              <img
                src={news.imageUrl}
                alt={news.title}
                className=
"w-full h-full object-fit md:h-full"
              />
            </div>

            <div className="md:w-2/3 p-6">
              <h2 className=
"text-2xl font-bold mb-4 text-secondary">
                {news.title}
              </h2>
              <p className="text-gray-600 ">{news.
description}</p>
              <a href={news.link} className=
"text-secondary underline text-sm">
                Read More
              </a>
            </div>
          </div>
        )));
      </div>
    </section>
  );
};

export default LatestNewsSection;
```

Figure 15:News.tsx

```
// src/data/newsData.js
export const newsData = [
  {
    id: 1,
    imageUrl:
      "https://www.aljazeera.com/wp-content/uploads/2024/08/AFP_20240822_36EN4HU_v3_HighRes_TopshotBangladeshWeatherFlood-1724392163.jpg?resize=1170,780&quality=100",
    title: "Deadly floods leave millions stranded in Bangladesh",
    description:
      "More than four million people have been affected by the floods triggered by torrential rains, with hundreds of homes under water...",
    link:
      "https://www.aljazeera.com/gallery/2024/8/23/deadly-floods-leave-millions-stranded-in-bangladesh"
  },
  {
    id: 2,
    imageUrl:
      "https://www.aljazeera.com/wp-content/uploads/2024/08/AFP_20240823_36EN3LM_v1_HighRes_BangladeshWeatherFlood-1724392266.jpg?resize=1170,780&quality=100",
    title: "When floods came, students rose to challenge",
    description:
      "Hundreds of students from Dhaka, Chattogram and districts in between were seen marching towards Feni on August 2. They travelled in trucks, with speedboats..."
  },
  {
    id: 3,
    imageUrl:
      "https://www.aljazeera.com/wp-content/uploads/2024/08/AFP_20240822_36EK2HP_v4_HighRes_BangladeshWeatherFlood-1724392143.jpg?fit=1170,780&quality=80",
    title: "Floods kill more than 30, impact millions in Bangladesh",
    description:
      "Rescuers are scrambling to evacuate flooded communities after heavy rains inundated parts of Bangladesh and north east India, killing at least 36..."
  },
];
```

Figure 16:NewsData.jsx



```
// @ts-ignore
import React from "react";
import {
  Accordion,
  AccordionContent,
  AccordionItem,
  AccordionTrigger,
} from "@/components/ui/accordion";
import { faqData } from "./FAQs";

const FAQSection = () => {
  return (
    <section className=
"w-full bg-primary bg-opacity-15 my-16 rounded-lg">
    <section className="w-full mx-auto py-12">
      <h2 className=
"text-5xl text-primary font-museo text-center w-full mb-
6">
        Frequently Asked Questions
      </h2>
      <Accordion
        type="single"
        collapsible
        className="text-primary mx-auto max-w-xl"
      >
        {faqData.map((item, index) => (
          <AccordionItem key={index} value={`item-${
            index
          }`}>
            <AccordionTrigger>{item.question}</
AccordionTrigger>
            <AccordionContent>{item.answer}</
AccordionContent>
          </AccordionItem>
        )));
      </Accordion>
    </section>
  </section>
);
};

export default FAQSection;
```

Figure 17:FAQSection.tsx

```
  export const faqData = [
    {
      question: "Are you affiliated with any donors?",
      answer:
        "No, we operate independently to provide unbiased information and services."
    },
    {
      question: "How accurate is the flood prediction?",
      answer:
        "Our predictions are based on advanced algorithms and historical data, but actual conditions may vary."
    },
    {
      question:
        "What should I do in case of a flood warning?",
      answer:
        "Follow local emergency services' guidelines, evacuate if necessary, and seek higher ground."
    },
    {
      question: "Can I reask for help through the Website?"
      answer:
        "Yes, users can ask for help directly through the website."
    },
    {
      question: "Is there a mobile app available?",
      answer: "No, currently we don't offer a mobile app."
    },
    {
      question:
        "How can I contribute to the flood relief efforts?",
      answer:
        "You can contribute by volunteering, donating, or sharing information about our platform."
    },
    {
      question:
        "What resources are available for flood preparedness?",
      answer:
        "We provide guides, tips, and resources to help you prepare for potential flooding in your area."
    },
    {
      question:
        "How does Flood Guard ensure user data privacy?",
      answer:
        "We take user privacy seriously and implement strict measures to protect your data."
    },
    {
      question:
        "Where can I find updates on flood conditions?",
      answer:
        "Updates on flood conditions can be found in the app and on our website's dashboard."
    },
  ];
]
```

Figure 18:FaqData.js

```
// @ts-ignore
import React from "react";
import { emergencyContacts } from "./EmergencyInfo";
import { Phone } from "lucide-react";

const EmergencyContactSection = () => {
    return (
        <section className=
"bg-primary bg-opacity-15 p-12 rounded-lg mx-auto">
            <h2 className=
"text-5xl text-primary text-center font-museo my-6 mb-1
2">
            >
                Emergency Contact Information
            </h2>
            <div className=
"grid grid-cols-1 text-center w-11/12 mx-auto sm:grid-col
s-2 lg:grid-cols-3 gap-6">
            >
                {emergencyContacts.map((contact, index) => (
                    <div
                        key={index}
                        className=
"text-secondary p-4 rounded-lg shadow-md bg-primary hove
r:shadow-xl transform hover:scale-105 transition duration
-300">
                        >
                            <contact.icon className=
"text-6xl mx-auto my-6 text-secondary" />
                            <h3 className="font-thin mb-6 text-4xl">{contact.name}</h3>
                            <p className=
"my-2 font-light flex items-center justify-center">
                                <Phone className="me-2 font-light" /> {
                                contact.number}
                            </p>
                            </div>
                        ))}>
                    </div>
                </section>
            );
};

export default EmergencyContactSection;
```

Figure 19:EmergencyContact.jsx



```
import { FaPersonMilitaryRifle } from "react-icons/fa6";
import { LuHelpingHand } from "react-icons/lu";
import { MdOutlineFireTruck } from "react-icons/md";
import { TbSailboat } from "react-icons/tb";
import { IoHelpBuoySharp } from "react-icons/io5";
import { PiAmbulance } from "react-icons/pi";
export const emergencyContacts = [
  {
    name: "National Helpline",
    number: "999",
    icon: LuHelpingHand,
  },
  {
    name: "Fire Service",
    number: "+02-2233555555",
    icon: MdOutlineFireTruck,
  },
  {
    name: "Army Helpline",
    number: "+88015501-50061",
    icon: FaPersonMilitaryRifle,
  },
  {
    name: "Coast Guard Helpline",
    number: "+8804662-75341",
    icon: IoHelpBuoySharp,
  },
  {
    name: "Emergency Ambulance",
    number: "+8801234-56789",
    icon: PiAmbulance,
  },
  {
    name: "Emergency Boat Service" ,
    number: "+8809876-54321",
    icon: TbSailboat,
  },
];
```

Figure 20:FaqData.js



```
// CardContainer.jsx
// @ts-ignore
import React from "react";
import { DonationSites } from "./DonationSites";
import ImageCard from "../../../../../components/ui/ImageCard";

const Donation = () => {
  return (
    <section className=
"bg-primary rounded-lg bg-opacity-15 flex flex-col justify-center items-center pb-6 my-12"
>
    <h1 className=
"text-5xl text-primary font-museo text-center pt-12 pb-6"
>Donate Now</h1>

    <div className=
"grid place-content-center grid-cols-1 md:grid-cols-3 lg:grid-cols-3 mx-auto min-h-screen items-center justify-center pb-6 w-10/12"
>
      {DonationSites.map((item, index) => (
        <div key={index} className="p-2">
          <ImageCard
            imgUrl={item.imgUrl}
            title={item.title}
            donationLink={item.donationLink}
          />
        </div>
      ))}
    </div>
  </section>
);
};

export default Donation;
```

Figure 21:Donations.jsx



```
//donationSites.js is a file that contains the data of the donation sites that are shown in the donation section of the homepage.

export const DonationSites = [
  {
    title: "Chief Advisor's Relief and Welfare Fund",
    imgUrl:
    "/DonationSites/Chief Advisor's Relief and Welfare Fund.png"
  ,
    donationLink:
    "https://donation.bkash.com/en/carwf/donate",
  },
  {
    title: "Armed Forces Division Relief Fund",
    imgUrl:
    "/DonationSites/Armed Forces Division Relief Fund.png",
    donationLink:
    "https://donation.bkash.com/en/afdrf/donate",
  },
  {
    title: "Bangladesh Army Relief Fund",
    imgUrl:
    "/DonationSites/Bangladesh Army Relief Fund.png",
    donationLink:
    "https://donation.bkash.com/en/barf/donate",
  },
  {
    title: "Bangladesh Navy Relief Fund",
    imgUrl:
    "/DonationSites/Bangladesh Navy Relief Fund.svg",
    donationLink:
    "https://donation.bkash.com/en/bnfwa/donate",
  },
  {
    title: "As-Sunnah Foundation Relief Fund",
    imgUrl: "/DonationSites/As-Sunnah Foundation.png",
    donationLink:
    "https://donation.bkash.com/en/assf/donate",
  },
  {
    title: "Bangladesh Red Crescent Society",
    imgUrl:
    "/DonationSites/Bangladesh Red Crescent Society.png",
    donationLink:
    "https://donation.bkash.com/en/brcs/donate",
  },
];
```

Figure 22:DonationSites.js