Required Installations

Frontend Installation Commands -

1.Create React App
  npx create-react-app app_name

2.Navigate to the app
  cd app_name

3.Essential packages
  npm install react-router-dom
  npm install react-icons

4.Run the app
  npm start

Backend Installation Commands -

1.Initialize backend
  npm init -y

2.Install essential packages
  npm install express mysql cors dotenv
  npm install cors

3.For password hashing (if serverless authentication is used)
  npm install bcrypt

4.Inorder to run both frontend and backend concurrently install
  npm install concurrently --save-dev

Also set your outer package.json code to -

```
{
 "name": " app_name",
 "version": "1.0.0",
 "scripts": {
   "start": "concurrently \"npm run server\" \"npm run client\"",
```

```
    "server": "cd backend && node index.js",
    "client": "cd src && npm run start-react",
    "start-react": "react-scripts start"
  },
  "devDependencies": {
    "concurrently": "^9.1.2"
  },
  "browserslist": {
    "production": [
      ">0.2%",
      "not dead",
      "not op_mini all"
    ],
    "development": [
      "last 1 chrome version",
      "last 1 firefox version",
      "last 1 safari version"
    ]
  },
  "dependencies": {
    "@fortawesome/fontawesome-free": "^6.7.2",
    "@fortawesome/free-brands-svg-icons": "^6.7.2",
    "@fortawesome/react-fontawesome": "^0.2.2",
    "react-dom": "^19.1.0",
    "react-icons": "^5.5.0",
    "react-responsive-carousel": "^3.2.23",
    "react-router-dom": "^7.6.1",
    "react-scripts": "^5.0.1"
  }
}
```
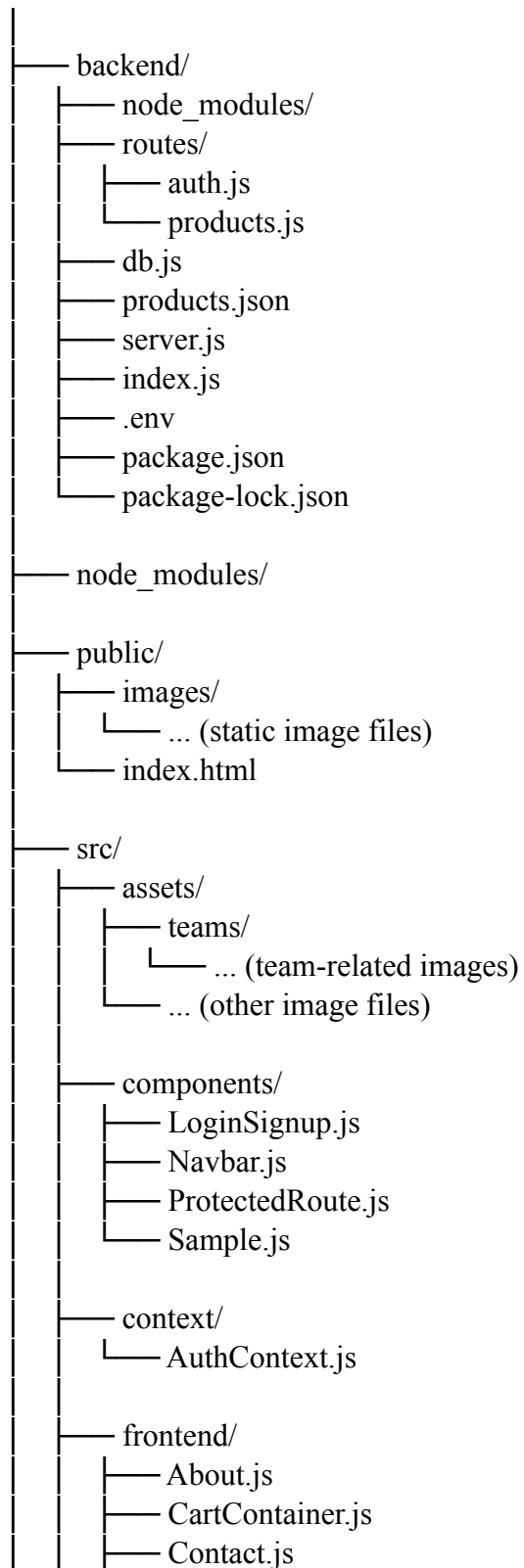
To install package.json and node modules files -

npm install

File Structure -

```
DOTGLASSESAPP

├── backend/
│   ├── node_modules/
│   ├── routes/
│   │   ├── auth.js
│   │   └── products.js
│   ├── db.js
│   ├── products.json
│   ├── server.js
│   ├── index.js
│   ├── .env
│   ├── package.json
│   └── package-lock.json
│
├── node_modules/
│
├── public/
│   ├── images/
│   │   └── ... (static image files)
│   └── index.html
│
├── src/
│   ├── assets/
│   │   ├── teams/
│   │   │   └── ... (team-related images)
│   │   └── ... (other image files)
│   │
│   ├── components/
│   │   ├── LoginSignup.js
│   │   ├── Navbar.js
│   │   ├── ProtectedRoute.js
│   │   └── Sample.js
│   │
│   ├── context/
│   │   └── AuthContext.js
│   │
│   ├── frontend/
│   │   ├── About.js
│   │   ├── CartContainer.js
│   │   ├── Contact.js
```

```
│  │        ├──── FAQ.js
│  │        ├──── Home.js
│  │        ├──── LensConcepts.js
│  │        ├──── OurImpact.js
│  │        ├──── ProductCard.js
│  │        ├──── ProductList.js
│  │        └──── WhomWeWork.js
│  │
│  ├──── styles/
│  │        ├──── About.css
│  │        ├──── Contact.css
│  │        ├──── FAQ.css
│  │        ├──── Header.css
│  │        ├──── Home.css
│  │        ├──── LensConcepts.css
│  │        ├──── LoginSignup.css
│  │        ├──── OurImpact.css
│  │        ├──── ProductCard.css
│  │        ├──── ProductList.css
│  │        └──── WhomWeWork.css
│  │
│  ├──── App.js
│  ├──── index.js
│  ├──── setupTests.js
│  └──── logo.svg
│
├──── .gitignore
├──── package.json
├──── package-lock.json
└──── README.md
```

MySQL -

1. Create Database -
CREATE DATABASE database_name;

2. Create Table -
CREATE TABLE table_name (
column1 datatype constraints,
column2 datatype constraints,
 ...
 );

3. Insert Values -
INSERT INTO table_name (column1, column2, ...)
VALUES (value1, value2, ...);

4. Update Values -
UPDATE table_name
SET column1 = value1, column2 = value2, ...
WHERE condition;

5. Drop Table -
DROP TABLE table_name;

6. Delete Values -
DELETE FROM table_name
WHERE condition;

Output -

I have used a single database with 2 tables (1 for Products and 1 for User Credentials) for backend.

The outputs of the database -

```
[mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| articles_db        |
| books              |
| information_schema |
| mysql              |
| performance_schema |
| students           |
| sys                |
+--------------------+
7 rows in set (0.022 sec)

[mysql> use articles_db;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
[mysql> show tables;
+---------------------+
| Tables_in_articles_db |
+---------------------+
| products            |
| users               |
+---------------------+
2 rows in set (0.003 sec)
```

Output of Products table -

```
mysql> select* from products;
+----+------------------------+------------------------------------------------------------------------------------+--------+------------------------+
| id | name                   | description                                                                        | price  | image                  |
+----+------------------------+------------------------------------------------------------------------------------+--------+------------------------+
|  1 | Large Vision Kit       | Complete large kit for vision testing.                                             | 800.00 | large-vision-kit.jpg   |
|  2 | Small Vision Kit       | Compact Kit suitable for basic usage.                                              | 500.00 | small-vision-kit.jpg   |
|  3 | Black Frame            | Durable frame in black color.                                                      | 500.00 | frame-black.jpg        |
|  4 | Blue Frame             | Durable frame in blue color.                                                       | 500.00 | frame-blue.jpg         |
|  5 | Brown Frame            | Durable frame in brown color.                                                      | 500.00 | frame-brown.jpg        |
|  6 | Full Rim Frame         | Durable full rim frame.                                                            | 500.00 | full-rim.jpg           |
|  7 | Oakley Frame           | Durable Oakley frame.                                                              | 500.00 | Oakley.jpg             |
|  8 | Distance Vision Tester | Tool to test distance vision clarity.                                              | 700.00 | distance-tester.jpg    |
|  9 | Microfiber Cloth       | Soft cleaning cloth for lenses.                                                    | 150.00 | accessory-cloth.jpg    |
| 10 | Pouches                | Protective pouch for glasses.                                                      | 300.00 | accessory-pouch.jpg    |
| 11 | Microfiber Cloth + Pouch | Combo of cloth and pouch.                                                        | 450.00 | accessory-combo.jpg    |
| 12 | Lens -6D               | Lens with -6 Diopters.                                                             | 700.00 | lens-minus6.jpg        |
| 13 | Lens -4D               | Lens with -4 Diopters.                                                             | 700.00 | lens-minus4.jpg        |
| 14 | Lens -2D               | Lens with -2 Diopters.                                                             | 700.00 | lens-minus2.jpg        |
| 15 | Lens 0D                | Neutral lens (0 Diopters).                                                         | 700.00 | lens-0.jpg             |
| 16 | Lens +2D               | Lens with +2 Diopters.                                                             | 700.00 | lens-plus2.jpg         |
| 17 | Lens +4D               | Lens with +4 Diopters.                                                             | 700.00 | lens-plus4.jpg         |
| 18 | Lens +6D               | Lens with +6 Diopters.                                                             | 700.00 | lens-plus6.jpg         |
| 19 | Near Vision Tester     | Chart to test near vision.                                                         | 750.00 | near-tester.jpg        |
| 20 | Eyeglass Cleaner       | A gentle spray cleaner that safely removes smudges and dirt from lenses without scratching. | 150.00 | eyeglass-cleaner.jpg |
+----+------------------------+------------------------------------------------------------------------------------+--------+------------------------+
20 rows in set (0.003 sec)
```

When the user successfully signUp into the website, the credentials of the users get stored in the database automatically.

```
mysql> select* from users;
+----+----------------+----------------------------+--------------------------------------------------------------+
| id | full_name      | email                      | password                                                     |
+----+----------------+----------------------------+--------------------------------------------------------------+
|  1 | Poojitha Reddy | poojitha@gmail.com         | $2b$10$7qhLU7BHPaD6qqxH3RmeVuXigroCNpMvmugPMdZBX.omhLhqwc2ly  |
|  2 | Akshaya Sri    | akshaya@gmail.com          | $2b$10$OO95t6SKiD/0wK5R98WfGeTT1N48TT6BCERx5ZHs/GSi1dYrFzhu.  |
|  3 | Aanam Poojitha | aanampoojithareddy@gmail.com | $2b$10$/9rwj9uj4.KphCrqwYAxyOSZXE/o53FQcUoGfDJ4Q0wxw1OeH4jxm |
|  4 | Dharani        | dharani@gmail.com          | $2b$10$z.ckj/u//2LMQEyTojvt9efQ5NeYwk9ZYFUrAXE1TRBPtw.TpDrkO  |
|  5 | Aditya         | aditya@gmail.com           | $2b$10$iTUAVjnearmhYbLsetMct.RXYPaCur8BhnXI9O9QqoxupwtVzlEHS  |
+----+----------------+----------------------------+--------------------------------------------------------------+
5 rows in set (0.002 sec)
```