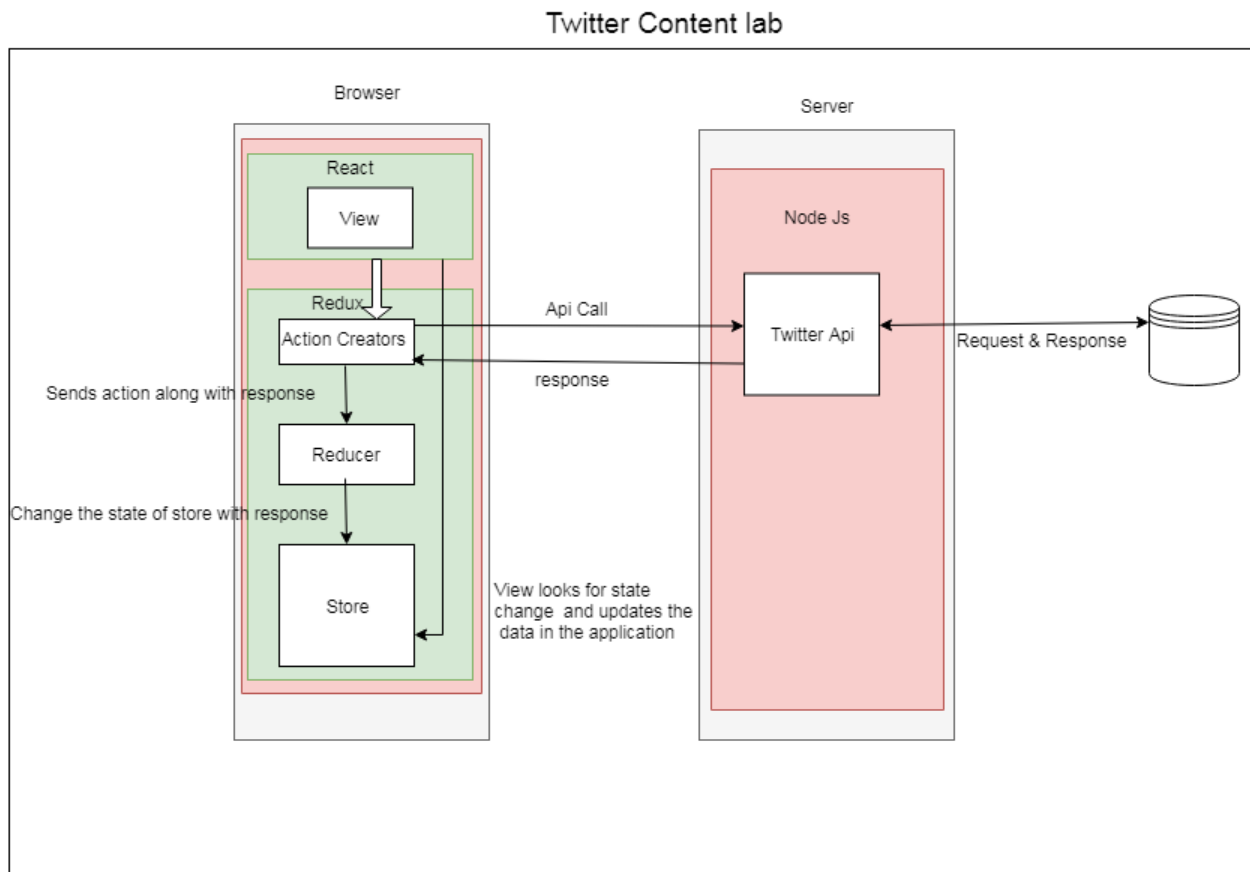


## Application Architecture



## Working of Architecture

- React renders on homepage on application startup
- Users make a request to twitter data by passing there parameters to the form and submitting it
- Request is dispatched to action creators
- Action creator make a api call to the port along with user parameters on which server is running
- Server receives the request and then passes it to the appropriate api
- API takes the request parameters and then request the access to the database along with the valid credentials for accessing the database
- Database returns the response (either requested data or error)to the api
- Api sends the response to the action creators.
- Action creators return the action along with data returned as a response from api
- Reducers looks for action type and updates the state for the appropriate action type
- React looks for change in application state and then updates the component which dependent on changed state

## Functionality

- Search the tweet by one or multiple hashtag and number
- By default tweets will be displayed sorted by favorite

- Users can sort tweet by retweet count

### **Technology Stack**

Front End: React, Redux, Bootstrap, HTML ,CSS

Backend: Node JS

Deployment: AWS EC2

### **Deployment on local host:**

- Install node js and concurrently globally
- Fork the github repo
- From cmd prompt go to the folder of repo
- In server/config/config.js apply twitter consumer, secret and bearer token
- Run npm install
- Run cd client
- Npm install
- Cd ..
- In root directory(Npm run dev)
- Page can be accessible at <http://localhost:3000>