NASA Image Application

GOBRIDGET Assignment

NASA IMAGE MICRESERVICE

Server Dependencies

Dependencies	Reason	Peer-Dependencies
Apollo-server-express	If we are exposing only as graphql server. We are restricted within graphql operations. So to overcome that issue, i selected Graphql with Express server.	• Apollo-server-core • Express • Graphql
mocha	Well known testing framework for javascript	
chai	assertion library used during testing	
nock	Used for mock the response during testing	
config	To separate a configure for each environment like staging, production, development. I am using this plugin	
winston	To debug any application, logging is most important thing. To do that we are using this plugin	• winston-daily-rotate-file
uuid	During logging, we need some unique id to track. To generate it we are using this plugin	
node-fetch	Use are using this library to call third party API	
express-http-context	For each request, you need a context to share allow the application. To achieve that we are using this library	

Server Dev Dependencies

Dependencies	Reason	Peer-Dependencies
@babel/core	Helps us to transpile the application to the lower version of javascript. To do that we can run this server any version of Node.js we want .	• @babel/cli • @babel/node • @babel/preset-env • @babel/register
babel-plugin-module-resolver	To support absolute path import we are using this plugin	
nodemon	nodemon will help us to watch the change in the file and restart the server automatically. It will helpful during development.	

Folder structure

- **Config** → All environment configs files are maintained in this folder
- **Dist** → After build all the transpile code will be available here and ready to ship.
- **Logs** → All the log files rotated per day will be available here (Inbound, transport, info, trace, debug)
- **Src**
  - **Controller** → controller is the place where i want to maintain all the business logic, you can create new controller for each modules or operations
    - **__test__** → Folder used to maintain test cases for our application
    - **Imagecontroller.js** → This controller consist of logic for fetch and filter data from NASA Image API
  - **Corelib** → If you want to use any libraries across application we can use this folder
    - **__test__** → Folder used to maintain test cases for our application
    - **Fetch.js** → FIle is a single point entry/exit for all third party API. To do that it is easy to log, track, error handling
    - **Logger.js** → File used to register and init the logging operation.
  - **Graphql** → All Graphql related code should be maintained with in this folder & Each module should have separate folder with in this.
    - **Image** → Image module folder
      - **Resolver.js** → Used to connect schema and controller file.
      - **Schema.js** → Used to create type definition for image module
    - **Resolver.js** → Used to combine all the resolvers
    - **Schema.js** → Used to combine all the schema
    - **Index.js** → This file have all configuration related to graphql and used to register and start graphql server
  - **Utils** → Use this folder to manage common utils like (DateUtils, Currency utils)
  - **Index.js** → Entry point for this service. Use to create and start server
  - **Routes.js** → Used to attach graphql with the routes.
- **.babelrc** → Used to manage transpile configuration
- **Jsconfig.json** → Used to manage JS config for IDE
- **Package.json** → Used for manage Dependencies

Script to Run Server

To start server for development

CD ~/gobridget _assignment/Server
npm start

To run in production mode

CD ~/gobridget _assignment/Server
npm run build
npm run prod

To run Test cases

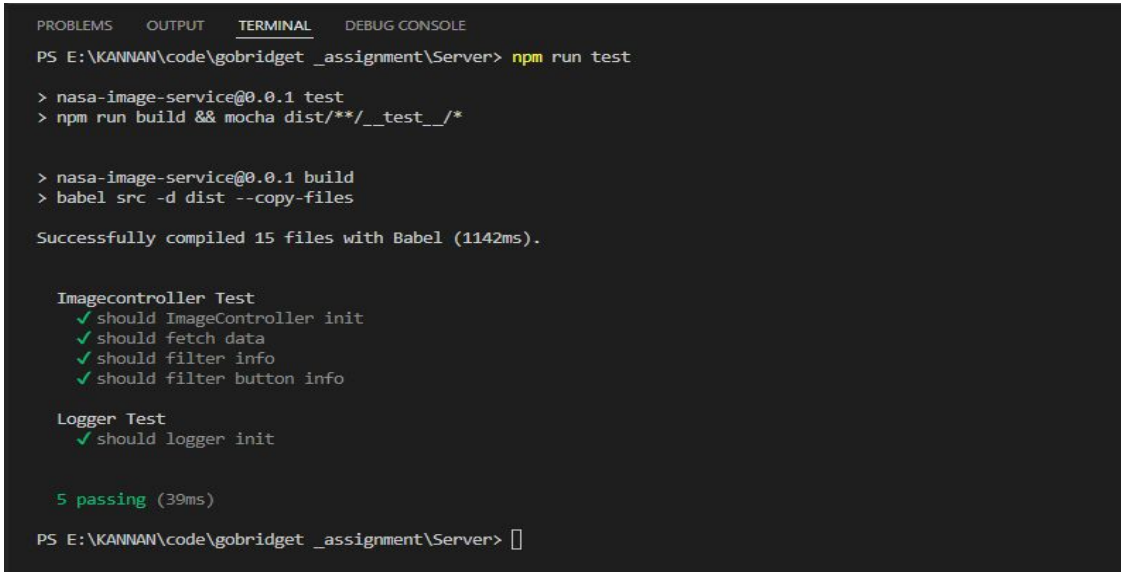
CD ~/gobridget _assignment/Server
npm run test

Note: We should always run test cases in build code.

Special mention

- Logger file will be rotated every day
- All type of network error handled in fetch.js
- UUID help us to track the entire flow
- Graphql plugin configuration help us to track the data flow in graphql
- Implemented absolute path import

Test case result



NASA IMAGE APPLICATION UI

This application UI created from CREATE-REACT-APP template

Created this folder structure to adopt modularization and To support micro-frontend / Remote Component in the future.

UI Dependencies

Dependencies	Reason
Bootstrap	To create responsive layout. I am adopted twitter bootstrap
redux	We can create local & global state using context & useReducer hooks or we can use Apollo client cache to manage global store. But if we are going to create enterprise application. We need need take consider few important things. Code Quality, maintainability, debugging, easy to adopt design pattern. For those reason i selected redux as global store
react-redux	Used to connect react application with redux
Immutable	Util library to easy access object and array and create immutable object. Important for React
redux-thunk	Attach util middleware to redux. Like common using util function are attached using this plugin
graphql-request	Apollo-client provide lots of features but i think graphql-request is light weight and more than enough
prop-types	Used for assign type definition for the component props
react-router-dom	Used for client side SPA routing
• @testing-library/jest-dom • @testing-library/reac • @testing-library/react-hooks	Use these libraries for writing unit test case for the component & hooks

Folder structure

Base Structure

- **Public**
- **Src**
  - **Assets** → This folder contains static asserts Images, CSS, Js
  - **Common** → This folder consist of reusable Component, Hooks, Utils, Translations, Layout
  - **Modules** → This folder will help us to manage modules use in our application
  - **Store** → This folder consist of global store configuration mechanism
  - **App.css** → CSS used all over the app
  - **App.js** → Bootstrap app configuration file like Router, Store
  - **Index.js** → Use for Entry point of this UI
- .**gitignore** → Ignore configuration
- **Jsconfig.json** → Config used for JS version and import
- **Package.json** → Config used for Manage dependency

Common Folder Expansion

- **Common**
  - **Components** → Common Components like Dropdown,Navbar,Side Navbar, etc
  - **Context** → ommon Context like User Context, Theme Context, etc
  - **Hooks** → Common hooks used in our application
  - **Layouts** → Common styled component, Layouts
  - **Translations** → Files used for application translation
  - **Utils** → Common util function like date utils, currency utils, etc.

Folder structure

Module Folder Expansion

- **modules**
  - **App**
    - **Component → All the components used in app module**
    - **Containers → All the container used in app module**
    - **Hooks → Hooks specifically used for APP module**
    - **Action.js → Action only used for App module**
    - **Reducer.js → Reduce specific to APP module**
    - **Schema.js → Schema only used in APP module**
  - **NASAImage**
    - **Component → All the components used in NASAImage module**
    - **Containers → All the container used in NASAImage module**
    - **Hooks → Hooks specifically used for NASAImage module**
    - **Action.js → Action only used for NASAImage module**
    - **Reducer.js → Reduce specific to NASAImage module**
    - **Schema.js → Schema only used in NASAImage module**
  - **……**
  - **…..**

Script to UI

To start server for development

CD ~/gobridget _assignment/Ui
npm run ui

To run in production mode

CD ~/gobridget _assignment/Ui
npm run build

Note: You can host the build folder using any static file servers

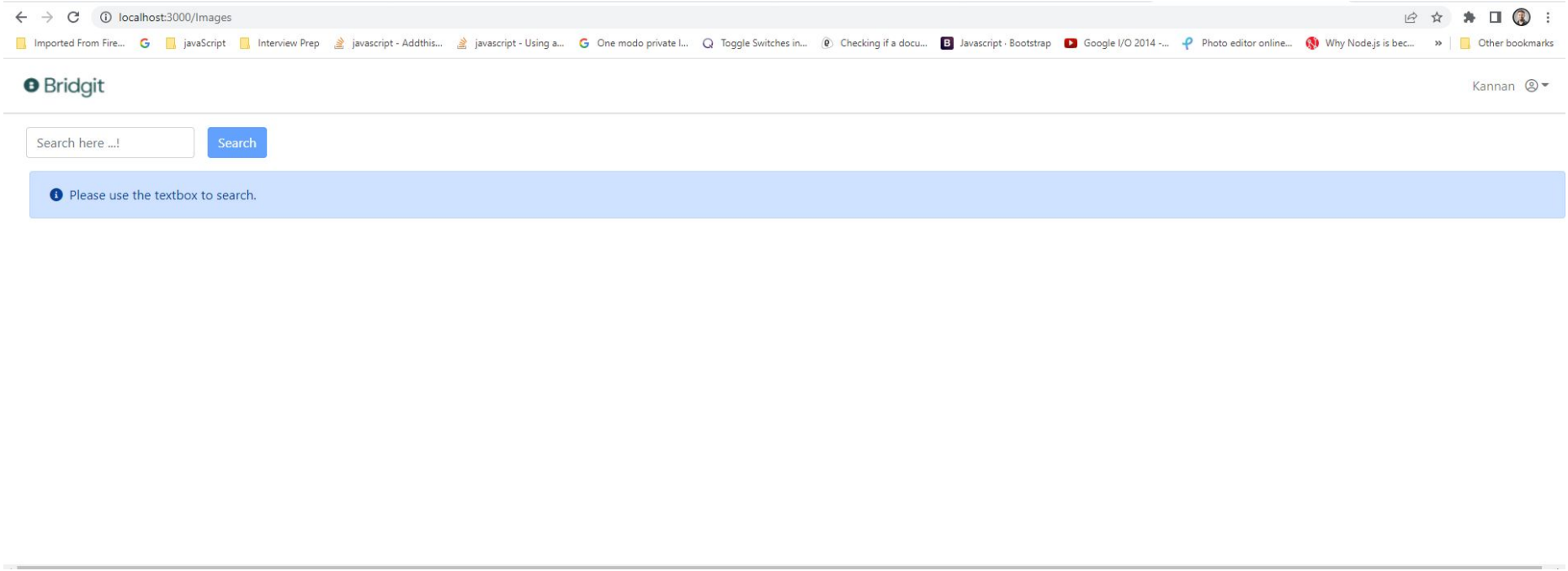
To run Test cases

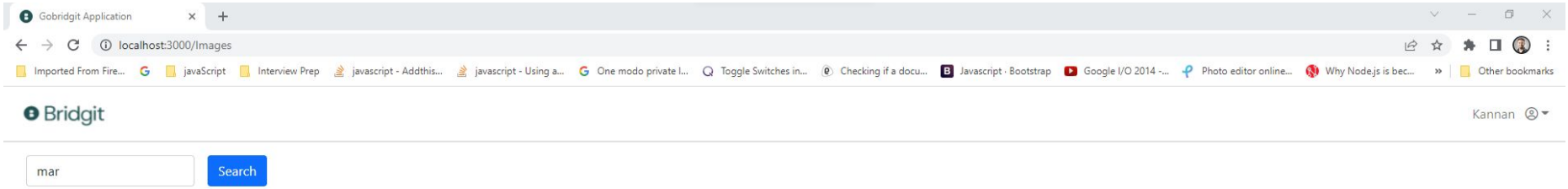
CD ~/gobridget _assignment/Ui
npm run test

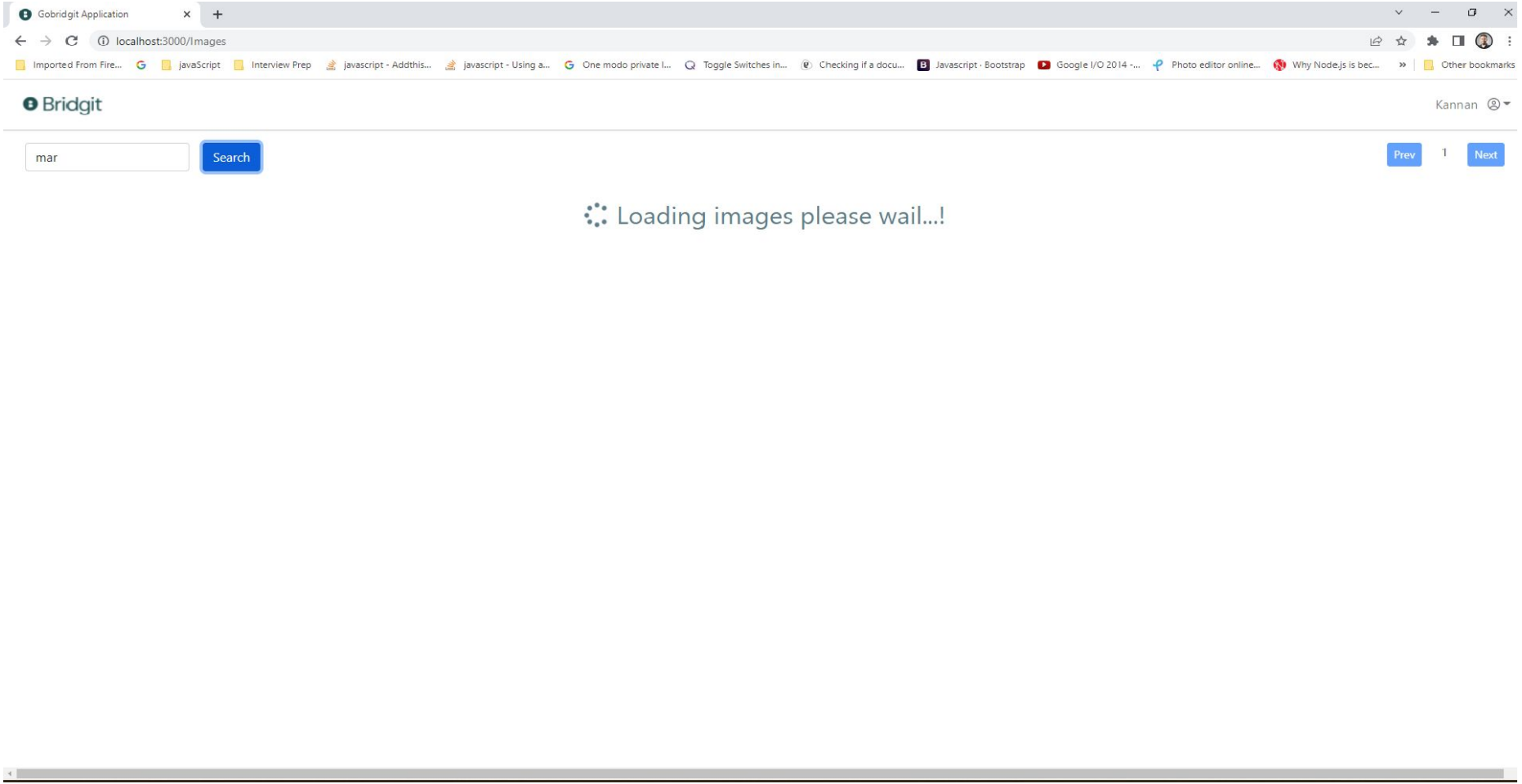
Special mention

- Redux store is created with store manage to support dynamic reduce injection
- Create Error boundary to capture child component error
- Created common logger file to collect all the errors. We can connect this file to API in the future
- Global store to easy maintains of data
- Created User context to share data across application
- Created single point route to easy management
- Designed the layout to support responsive

Screenshots of Application


localhost:3000/Images
Imported From Fire...
javaScript
Interview Prep
javascript - Addthis...
javascript - Using a...
One modo private l...
Toggle Switches in...
Checking if a docu...
Javascript · Bootstrap
Google I/O 2014 -...
Photo editor online...
Why Node.js is bec...
Other bookmarks
Bridgit
Kannan
Search here ...!
Search
Please use the textbox to search.


Gobridgit Application
localhost:3000/Images
Imported From Fire...
javaScript
Interview Prep
javascript - Addthis...
javascript - Using a...
One modo private l...
Toggle Switches in...
Checking if a docu...
Javascript · Bootstrap
Google I/O 2014 -...
Photo editor online...
Why Node.js is bec...
Other bookmarks
Bridgit
Kannan
mar
Search


Gobridgit Application
localhost:3000/Images
Imported From Fire...
javaScript
Interview Prep
javascript - Addthis...
javascript - Using a...
One modo private l...
Toggle Switches in...
Checking if a docu...
Javascript · Bootstrap
Google I/O 2014 -...
Photo editor online...
Why Node.js is bec...
Other bookmarks
Bridgit
Kannan
mar
Search
Prev
1
Next
Loading images please wail...!

Gobridgit Application

localhost:3000/Images

Imported From Fire... | javaScript | Interview Prep | javascript - Addthis... | javascript - Using a... | One modo private l... | Toggle Switches in... | Checking if a docu... | Javascript · Bootstrap | Google I/O 2014 -... | Photo editor online... | Why Node.js is bec... | Other bookmarks

Bridgit

Kannan

mar

Search

Prev 1 Next

Mars Celebration

The Mars celebration Friday, May 31, 2019, in Mars, Pennsylvania. NASA is in the small town to celebrate Mars exploration and share the agency's excitement about landing astronauts on the Moon in five years. The celebration includes a weekend of Science,...

Mars Celebration

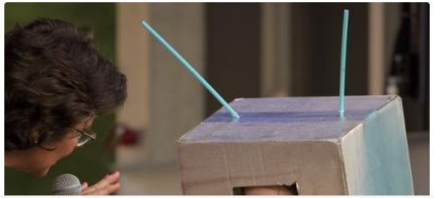
The Mars celebration Friday, May 31, 2019, in Mars, Pennsylvania. NASA is in the small town to celebrate Mars exploration and share the agency's excitement about landing astronauts on the Moon in five years. The celebration includes a weekend of Science,...

Mars Celebration

The Mars celebration Friday, May 31, 2019, in Mars, Pennsylvania. NASA is in the small town to celebrate Mars exploration and share the agency's excitement about landing astronauts on the Moon in five years. The celebration includes a weekend of Science,...

Mars Celebration

The Mars celebration Saturday, June 1, 2019, in Mars, Pennsylvania. NASA is in the small town to celebrate Mars exploration and share the agency's excitement about landing astronauts on the Moon in five years. The celebration includes a weekend of Science,...



Gobridgit Application

localhost:3000/Images

Imported From Fire... javaScript Interview Prep javascript - Addthis... javascript - Using a... One modo private l... Toggle Switches in... Checking if a docu... Javascript · Bootstrap Google I/O 2014 -... Photo editor online... Why Node.js is bec... Other bookmarks

Bridgit

Kannan

ma#

Search

Please use only letters, numbers and space