Lec 2

Now, in package.json, in the devdependencies, you will see a parcel version denoted like “^2.8.2”. This ^(caret) means that your packages will get updated to the future minor/patch versions, without incrementing the major versions i.e. “^2.8.2” will use releases from 2.8.2 to < 3.0.0.

Similarly, there can also be “~2.8.2”, where `(tilde) denotes that your packages will get updated to the future patch versions, without incrementing the minor versions i.e. “^2.8.2” will use releases from 2.8.2 to < 2.9.0.

In package-lock.json, for any package, there is a key named version, which stores(or locks) the exact version that the project will be using and it is also useful for installing all the packages for a project, when you clone it from the github. That’s why, we never keep package-lock.json in the .gitignore.
There is another key named integrity, which also stores the exact version of that package, but in Hashed form.

In node-modules folder, there is also a folder named “browserlist”, which helps us to make our app compatible with older versions of any or all browsers (We will see how later on). Node-modules is the heaviest folder in your project and it can also be regenerated exactly with the help of the above 2 json files. That’s why we don’t put node-modules folder in the git repo.

Hot

Reloading :- If we change anything in our code while our server is running, those changes will be automatically reflected in out browser without -> stopping the server, then making code changes and restarting the server.

Dist Folder :- It creates a faster minimised, development version of our project and serves it to the server. When I tell

Features of Parcel / What does Parcel do to our project

- HMR – Hot Module replacement
- Implements a File Watcher using File Watcher Algo (written in C++)
- Bundling all the necessary files
- Minifying the code . For ex : All the code in the libraries i.e. react and react-dom, that we imported in App.js will be placed in a single js file, that you can find in the dist folder (there can be multiple js files for a single original js file too, don’t worry).
- Cleaning our code. For ex :- It removes all the console.logs from our project in the production build
- Image optimization. If there’s an image in our project, parcel will optimize it so that the image does not take too much time to load in the user’s website. This is very helpful, since images are the most heavy things in our website.
- Caching while development
- Compression. It compresses the files by renaming the variable names to smaller names and other techniques.
- Compatible with older versions of browsers
- It can automatically handle the port number in which our server runs.
- Uses Consistent Hashing Algorithm.
- It’s a Xero Config Module Bundler.

#NOTE :-

React is fast. But to create a React app, React needs many more packages to make it faster like Parcel etc. In turn these packages also need other packages/dependencies. This forms a Dependency Tree and this phenomenon is called Transitive Dependency.

Tree Shaking (actually taught in Day-3 class) :-

Another feature provided by Parcel. It means removing unwanted code. Suppose our code imports a library which gives us access to a lot of helper functions and even though I use only a few of those functions, all the remaining helper functions are still included in our project. So these unwanted functionalities are ignored by Parcel while making the production build of our app (It just shakes the tree).