# Introduction

The report is about the modern web development technology and tools of reliable sources and suitability for different platforms. The following are the modern tools which are very useful for today’s modern technology. Below are the following Tools going to discuss about how to use and benefit as well as their reliability in modern world:

* Git
* Package and Package management Systems
* Version control tools
* GitHub (Collaboration tools)
* Front end tool(bootstrap)
* Back end tools(node.js)
* Task runners

# Git (Version control Tool)

According to (git, n.d.) Git is a free open source platform of version control where all the codes or project are shared and saved to maintain worldwide with good speed and high quality.

It is very easy to learn and understand and has a fast performance.

According to (Atlassian Bitbucket, n.d.)Git is used by every organization these days in modern world like for developers or anyone , git has a branching capabilities which is the main benefit of using git which help many users to share their workflow. When we work on git creates a new branch where the master branch always keep the quality code so it is more reliable platform because of direct editing code facility .

Another benefit of git is every user have their own local repository where they keep all their work which they maintain their complete work history with ‘commit’ and can access from anywhere in the world. As well as we can clone or download anyone’s work and start working on that project or refer to get help. Git creates more reliable platform for everyone to work.

Git has a bitbucket tool to enhance the work by using the pull request. ‘Pull request ‘is to pull your work from your desktop to your repository.

**How to work with git:**

According to (Balos, 2007-2009) to work on windows or mac with git we need to set up git with some following steps as we need to install and configure Git first then have to connect with the remote repositories with clone, pull and push. These steps to be done we first have to make an account name Beanstalk.

Now we have to choose Git distribution. There are two types of git packages to choose for windows are a Cygwin-based git and a version called msysGit, among the two packages mostly a Version msysGit is recommended as it is easy to work with.

To install Git firstly we download msysGit then double click on it to start the installation wizard. Here let the path be in default directory. Then from command prompt ‘run git’. To run git command, we can use git Bash as well.

# Package and Package management Systems

According to (techopedia, 2019)package consists of many software which combine together to form a group of software which perform certain specific task to get the desired result like Microsoft office package have all the documents to work on are Word, Excel, Access and PowerPoint together to perform certain task.

**Npm:**

According to (wikipedia, 2019)known as node package manager. The package Node.js known as the default runtime environment for JavaScript programming language, which has npm command line and npm registry for database.

The use of npm is to manage the national dependency of a certain project or JavaScript tools which are installed worldwide or internationally.

For a national project, need to have npm in a command where dependency project should create as package.json file also npm creates package-lock.json file which access the true version of the project used in the package. json.

**Package Management system:**

According to (Bower, 2016)package manager or management system keep track of the packages and keep them updated with all the version needed. Can use a Bower which is a package manager for the web application.

Bower manages HTML, CSS, JavaScript components as well as fonts and images by installing the correct version required for the dependency.

It installs packages and keep track of bower. json packages in tools.

It is most effective for the front-end like jQuery. To install bower, have to install npm as well as need node, npm and git.

Best to consider using bower to combine with Grunt, RequireJS, and many other tools. Or with the API to have your own workflow. Another way is to directly install the packages like jQuery. Such as

<script src=” bower-components/jquery/dist/jquery.min.js”></script>.

So, the conclusion is package management system allows to configure the dependencies required for the application dependencies don’t need to be part of the binary. For that reason, we use the package manager. Also allows you to bind yourself to a specific version in case the app doesn’t need the latest version.

**Node Package Manager:**

According to (RisingStack, 2017)npm is the default package manager for Node.js. npm generates automated lock. json to have all the updates and modifications of the file.

Npm electronically install package. json which are dedicated to version control we are using today.

Npm is one of the modern tools and technology mostly used to manage dependencies due to its lock file support.

# Version control tools

According to (ATLASIAN BitBucket, n.d.)version control tools are the group of software tools which help to manage the changes of source code on worktime. Keep track of any changes in the database. Version control system track all the work to keep updates so that anyone can access and see all the modification. Another name of version control system is the source code management system.

The most popular version control tool is the Git which is used mostly in todays modern world. Git is a distributed version control which is free to use and an open source code.

**The benefit of using version control tools are:**

1. Keeps track of all the work history of any new files have been created or deleted and edited files too. Also help to go back to find the previous versions to find the main reasons for any bugs to fix the issues/error.
2. Help a team or individual to work together and share the work with the help of branching and merging.
3. Tracking all the work is another main benefit of using version control tools.

**How does version control work?**

According to (Fairely, 2019)version control has control and manages all the projects, keep track the master copy so that various people can work together globally.

Version control hold various plans to implement to manage all the source codes. Those are as follows:

* File locking is the process where people cannot modify the file at the same time.
* Merging is the process which allow multiple users to view the same file at the same time.

**Why version control important:**

According to (Perforce, 2018)version control gives end-to-end traceability to improve the collaboration. It also helps the team work and deliver project faster.

There are few main important facts to use version control are as follows:

* Version control tools creates single source for all the files and give better performance.
* It provides us with pre-commit to have good integrity for the codebase which give continuous integration.
* Version control also plays crucial role of the entire process as it coordinates and actively participate in continuous delivery with regular and reliable deployments.
* It contains a flexible working solution to review codes and browse all the essential documents. So, provide customer satisfaction and iteration process.
* Version control tool gives the link of the development tool chain.
* It ensures all the work or source code are safe and secure.

**Why use version control tools:**

According to (Tower, 2013-2019)version control tools are use for many reasons like:

* For the collaboration where we share folders to work together.
* Using to store versions as storage where we save all our work after making any changes in the project/work.
* Store the previous version of a file where you can go back if any issues arise to fix accurately.
* You can understand the exact version of what’s happening with your work as version control provides a small description of the change.
* Finally, version control has a Backup always where everyone has an access to the full version of the code in the disk as well as have complete work history.

# GitHub (Collaboration tools)

According to (Cutrell, 2013)GitHub is very effective technique to collaborate with the workflow. It’s a place where anyone can work if have an internet connection and share code globally for free.

The GitHub is very simple to start with where we carry with few steps:

* Firstly, we have to create an account
* Then clone the repository to the local machine
* Make a new folder and do all the require changes
* Push the created folder to your repository
* Create a pull request to use for different users.

**Collaborating GitHub with Team:**

According to (Basu, 2013)GitHub is very important tool for the open source projects. GitHub is mainly used for its collaboration tool. GitHub has a very essential feature of working together as a worldwide team.

So, to collaborate GitHub have few tools are mention below:

1. To setup with GitHub where we can add team members as Organization and collaborators like:

* Pull to merge and fetch with repository
* Push and pull to update the remote repository
* Pull, push and administrative to create teams or delete or cancel any organization

1. To have pull request by initiating:

* Fork and pull model in a public repo where we don’t have access to push
* Share repository model which is private and can access to push

1. To track the bug
2. Are analytics to all our work lists and updates of all the work commits.
3. Manages the project using GitHub and trello also pivotal tracker to manage the agile development.
4. Able to integrate continuously
5. Have both formal and informal documentation to read

**Collaborating on GitHub:**

According to (Jeblee & chung, 2017)Git is a version control system of command line whereas GitHub is an internet hosting service of the repository. In GitHub public repository are free to use but to have private repository we have to pay.

Basic command to shared GitHub Repos:

* :> git pull origin master
* :> git checkout -b branch
* :> git add
* :> git commit -m ‘first commit’
* :> git push origin branch

Collaboration of clone’s repo command:

* <https://github.com/>
* Git clone URL-of-origin-repo dir-address-of-local-repository

Command to very the status of your repository:

* :> git status
* :> git commit -m “message”
* :>git push origin master

Command to pull and see the new version:

* :> git pull origin master

To allow your file for another user to view you have to choose the README.md file.

# Front end tool(bootstrap)

According to (BACINGER, 2010-2019)bootstrap is a group of HTML, CSS, and JavaScript framework for web application and for the development of website/webpage.

**Why bootstrap:**

Bootstrap is very well-known tool used by all developers these days because of its flexibility and it is very simple to work on.

**Benefit:**

Main benefit of using bootstrap tool is to make website or web application responsive and supply vast browser compatibility. Bootstrap has an accurate design to reusable elements. bootstrap is very and and anyone can learn bootstrap fast. It has built-in-support for jquery plugins and API of JavaScript.

According to (EOOD, 2019)bootstrap studio is very strong local machine app for designing and prototyping the websites. Which is available with a huge number of built-in-components. Just drag the components and drop to construct responsive web application. Also bootstrap has a complete control on the mark-up which can be edited and imported CSS, html and JavaScript in any editor.

**Bootstrap tool setup:**

According to (B, n.d.)for the built-in system bootstrap use the npm script where package. json includes best way to work with the framework.

To get bootstrap tool we have to follow certain steps as:

* Download and then install Node.js to manage the dependency
* Navigate the bootstrap directory and then run ‘npm install’ in package. Json

**NPM Scripts:**

Command and task of package. json are

* Npm run dist
* Npm test
* Npm run docs

**Local documentation:**

To run the local documentation, we have to open ‘ <http://localhost:8080> ‘in your browser.

**Troubleshoot:**

To troubleshoot if any problem arises the have to uninstall all the version globally and locally then again run the ‘npm install’.

According to (Rechsteiner, 2017)bootstrap give style and theme to the package of the html, CSS and JavaScript. It has a user interface components and layouts to use in the wen application.

**Benefits:**

Bootstrap have ready-made styling components and themes to design which make the work easy and take less time to build and also same lot of money.

It also gives a good-looking layout for the website in less time and without spending big amount related to the UI design.

# Back end tools(node.js)

According to (tutorialsPoint, 2019)node.js is a server-side platform made on Google chrome JavaScript engine.

**Node.js features:**

* All APIs Node.js library is non-blocking and asynchronous.
* As Node.js based on google chrome therefore the process is very fast to execute the code.
* Node.js is a single threaded programme with the event loop and make the server scalable.
* Node.js don’t buffer any data.

According to (w3schools, 1999-2019)node.js is free and an open source environment to run on different platforms using JavaScript on the server side.

According to (node.js, n.d.) the framework and tools of node.js are like

* Express which is one of the simplest and robust way to build a web server.
* Next.js is a server-side framework to provide React application
* Nx make the Angular CLI to build full-stack applications using Express and Angular

**Express:**

According to (tutorialspoint , 2019)express is a node.js framework for the web application which have multiple features to build web and mobile application.

Features of Express framework are as follows:

* To reply to http requests express permits to set up the middleware
* It explains a table routing to perform various actions on http methods and the URL.
* Enable dynamic html pages to contribute on passing the arguments.

Following are the few commands to install express:

* Initially have to install the npm, then have the command as

$npm install express

* To run the command is

$node index.js

* For the request and response command is

App.get(‘/’, function (req,res){ })

According to (npm, n.d.)the features of express are as follows:

* Have powerful routing.
* Give the fast performance
* Provide http for caching and redirecting the files
* Contain with 14+ template engines to view the support system.
* Have better and quick execution
* Negotiate the content.

# Task runners

According to (Hunter, n.d.)to run each task in the development process with their complexity we use task runners’ tool. There are two main task runners are

1. Grunt
2. Gulp

Gruntfile.js need module to be install as

* Grunt
* Grunt-contrib-sass
* Grunt -contrib-clean

Also, we can run as

* Grunt style
* Grunt script
* Grunt build

Gulpfile.js versions installed are as

* Gulp
* Gulp-sass
* Node-sass

Also, we can run

* Gulp style
* Gulp script
* Gulp build

According to (DBS interactive, 2015)task runners just run the task which we want to work to give good performance.

Task runner will perform all the work we want to complete if we run only once and will perform everything for us so it is considered to save time and cost.

According to (grunt the javascript task runner, 2018)task runner automatically runs every task for us, so we have less work incase of same repeated work.

# Difference of Git and GitHub

According to (Stackoverflow, 2017) git is a tool which manages all the source code history. Git is also consider as the revision control. Whereas the GitHub is the service of hosting the Git repositories.

According to (Mckenzie, 2018)Git is distributed tool to manage a source code or work history.

Whereas GitHub is a cloud based platform build in the Git tool. So Git is tool which install in the local computer and GitHub is an online service which maintain all the record of work history as code pushed from computer running the Git. In short Git is the open source tool which install locally to manage the code and GitHub is the online service who use the tool Git to connect and download or upload the files.

# Node and Node Package

According to (Valaven, 2018)Node runtime which understand the JavaScript code and execute to give the result. Whereas the Node Package is a tool which allow to install software libraries using the command line as **npm install express.**

According to (Quora, n.d.)Node is an interpreter for the JavaScript files or JavaScript runtime environment which help us to write the server side code whereas Node Package stays in the internet which is used to install the software like libraries, plugins, frameworks and applications using npm. Also Node Package manager manages the node module.

# References

*ATLASIAN BitBucket*. (n.d.). Retrieved from Atlasian: https://www.atlassian.com/git/tutorials/what-is-version-control

*Atlassian Bitbucket*. (n.d.). Retrieved from https://www.atlassian.com/git/tutorials/why-git

*B*. (n.d.). Retrieved from https://getbootstrap.com/docs/4.0/getting-started/build-tools/

BACINGER, T. (2010-2019). *Developers*. Retrieved from https://www.toptal.com/front-end/what-is-bootstrap-a-short-tutorial-on-the-what-why-and-how

Balos, I. (2007-2009). *beanstalk guide*. Retrieved from http://guides.beanstalkapp.com/version-control/git-on-windows.html

Basu, S. (2013, February 28). *Team collaboration with Github*. Retrieved from envatotuts+: https://code.tutsplus.com/articles/team-collaboration-with-github--net-29876

*Bower*. (2016, November 6). Retrieved from https://bower.io/

Cutrell, J. (2013, August 22). *How to collaborate on Github*. Retrieved from envatotuts+: https://code.tutsplus.com/tutorials/how-to-collaborate-on-github--net-34267

*DBS interactive*. (2015, february 24). Retrieved from the advantages of using task runners: https://www.dbswebsite.com/blog/the-advantages-of-using-task-runners/

EOOD, Z. (2019). *bootstrap studio*. Retrieved from https://bootstrapstudio.io/

Fachat, A. (January 27, 2015). *Learn the workings of Git, not just the commands.*

Fairely, S. (2019, September 14). *Mudbath*. Retrieved from Mudbath&co: https://www.mudbath.com.au/blogs/the-importance-of-version-control

*git*. (n.d.). Retrieved from https://git-scm.com/

*grunt the javascript task runner*. (2018, august 15). Retrieved from https://gruntjs.com/

Hunter, T. (n.d.). *LogRocket*. Retrieved from M: https://blog.logrocket.com/node-js-task-runners-are-they-right-for-you-bb29ea30b7fa

Jeblee, S., & chung, K. (2017, August 2). *COLABORATING ON GITHUB*. Retrieved from UofTCoders: https://uoftcoders.github.io/studyGroup/lessons/git/collaboration/lesson/

McKenzie, C. (2018, November 6). *The server side*. Retrieved from https://www.theserverside.com/video/Git-vs-GitHub-What-is-the-difference-between-them

Mckenzie, C. (2018, November 6). *TheServerSide*. Retrieved from https://www.theserverside.com/video/Git-vs-GitHub-What-is-the-difference-between-them

mudbath. (2019). *Mudbath.* Retrieved from Mudbath&co: https://www.mudbath.com.au/blogs/the-importance-of-version-control

*node.js*. (n.d.). Retrieved from https://nodejs.dev/

*npm*. (n.d.). Retrieved from https://www.npmjs.com/package/express

*Perforce*. (2018, June 26). Retrieved from https://www.perforce.com/blog/vcs/why-version-control-important

*Quora*. (n.d.). Retrieved from https://www.quora.com/What-are-the-difference-between-node-and-nodejs-npm-vs-nvm

Rechsteiner, A. (2017, March 14). *hackerthemes*. Retrieved from https://hackerthemes.com/what-are-bootstrap-themes/

*RisingStack*. (2017). Retrieved from Risingstack: https://blog.risingstack.com/yarn-vs-npm-node-js-package-managers/

*Stackoverflow*. (2017, November 8). Retrieved from https://stackoverflow.com/questions/13321556/difference-between-git-and-github

*techopedia*. (2019). Retrieved from https://www.techopedia.com/definition/4360/software-package

*Tower*. (2013-2019). Retrieved from https://www.git-tower.com/learn/git/ebook/en/command-line/basics/why-use-version-control

*tutorialspoint* . (2019). Retrieved from https://www.tutorialspoint.com/nodejs/nodejs\_express\_framework.htm

*tutorialsPoint*. (2019). Retrieved from https://www.tutorialspoint.com/nodejs/nodejs\_introduction.htm

Valaven, A. (2018, February 24). *stackoverflow*. Retrieved from https://stackoverflow.com/questions/41675848/what-is-the-difference-between-node-js-runtime-and-npm-package-manager-options-w/41676076

*w3schools*. (1999-2019). Retrieved from https://www.w3schools.com/nodejs/nodejs\_intro.asp

*wikipedia*. (2019, May 20). Retrieved from https://en.wikipedia.org/wiki/Npm\_(software)

# References

*ATLASIAN BitBucket*. (n.d.). Retrieved from Atlasian: https://www.atlassian.com/git/tutorials/what-is-version-control

*Atlassian Bitbucket*. (n.d.). Retrieved from https://www.atlassian.com/git/tutorials/why-git

*B*. (n.d.). Retrieved from https://getbootstrap.com/docs/4.0/getting-started/build-tools/

BACINGER, T. (2010-2019). *Developers*. Retrieved from https://www.toptal.com/front-end/what-is-bootstrap-a-short-tutorial-on-the-what-why-and-how

Balos, I. (2007-2009). *beanstalk guide*. Retrieved from http://guides.beanstalkapp.com/version-control/git-on-windows.html

Basu, S. (2013, February 28). *Team collaboration with Github*. Retrieved from envatotuts+: https://code.tutsplus.com/articles/team-collaboration-with-github--net-29876

*Bower*. (2016, November 6). Retrieved from https://bower.io/

Cutrell, J. (2013, August 22). *How to collaborate on Github*. Retrieved from envatotuts+: https://code.tutsplus.com/tutorials/how-to-collaborate-on-github--net-34267

*DBS interactive*. (2015, february 24). Retrieved from the advantages of using task runners: https://www.dbswebsite.com/blog/the-advantages-of-using-task-runners/

EOOD, Z. (2019). *bootstrap studio*. Retrieved from https://bootstrapstudio.io/

Fachat, A. (January 27, 2015). *Learn the workings of Git, not just the commands.*

Fairely, S. (2019, September 14). *Mudbath*. Retrieved from Mudbath&co: https://www.mudbath.com.au/blogs/the-importance-of-version-control

*git*. (n.d.). Retrieved from https://git-scm.com/

*grunt the javascript task runner*. (2018, august 15). Retrieved from https://gruntjs.com/

Hunter, T. (n.d.). *LogRocket*. Retrieved from M: https://blog.logrocket.com/node-js-task-runners-are-they-right-for-you-bb29ea30b7fa

Jeblee, S., & chung, K. (2017, August 2). *COLABORATING ON GITHUB*. Retrieved from UofTCoders: https://uoftcoders.github.io/studyGroup/lessons/git/collaboration/lesson/

McKenzie, C. (2018, November 6). *The server side*. Retrieved from https://www.theserverside.com/video/Git-vs-GitHub-What-is-the-difference-between-them

Mckenzie, C. (2018, November 6). *TheServerSide*. Retrieved from https://www.theserverside.com/video/Git-vs-GitHub-What-is-the-difference-between-them

mudbath. (2019). *Mudbath.* Retrieved from Mudbath&co: https://www.mudbath.com.au/blogs/the-importance-of-version-control

*node.js*. (n.d.). Retrieved from https://nodejs.dev/

*npm*. (n.d.). Retrieved from https://www.npmjs.com/package/express

*Perforce*. (2018, June 26). Retrieved from https://www.perforce.com/blog/vcs/why-version-control-important

Rechsteiner, A. (2017, March 14). *hackerthemes*. Retrieved from https://hackerthemes.com/what-are-bootstrap-themes/

*RisingStack*. (2017). Retrieved from Risingstack: https://blog.risingstack.com/yarn-vs-npm-node-js-package-managers/

*Stackoverflow*. (2017, November 8). Retrieved from https://stackoverflow.com/questions/13321556/difference-between-git-and-github

*techopedia*. (2019). Retrieved from https://www.techopedia.com/definition/4360/software-package

*Tower*. (2013-2019). Retrieved from https://www.git-tower.com/learn/git/ebook/en/command-line/basics/why-use-version-control

*tutorialspoint* . (2019). Retrieved from https://www.tutorialspoint.com/nodejs/nodejs\_express\_framework.htm

*tutorialsPoint*. (2019). Retrieved from https://www.tutorialspoint.com/nodejs/nodejs\_introduction.htm

Valaven, A. (2018, February 24). *stackoverflow*. Retrieved from https://stackoverflow.com/questions/41675848/what-is-the-difference-between-node-js-runtime-and-npm-package-manager-options-w/41676076

*w3schools*. (1999-2019). Retrieved from https://www.w3schools.com/nodejs/nodejs\_intro.asp

*wikipedia*. (2019, May 20). Retrieved from https://en.wikipedia.org/wiki/Npm\_(software)

# References

*ATLASIAN BitBucket*. (n.d.). Retrieved from Atlasian: https://www.atlassian.com/git/tutorials/what-is-version-control

*Atlassian Bitbucket*. (n.d.). Retrieved from https://www.atlassian.com/git/tutorials/why-git

*B*. (n.d.). Retrieved from https://getbootstrap.com/docs/4.0/getting-started/build-tools/

BACINGER, T. (2010-2019). *Developers*. Retrieved from https://www.toptal.com/front-end/what-is-bootstrap-a-short-tutorial-on-the-what-why-and-how

Balos, I. (2007-2009). *beanstalk guide*. Retrieved from http://guides.beanstalkapp.com/version-control/git-on-windows.html

Basu, S. (2013, February 28). *Team collaboration with Github*. Retrieved from envatotuts+: https://code.tutsplus.com/articles/team-collaboration-with-github--net-29876

*Bower*. (2016, November 6). Retrieved from https://bower.io/

Cutrell, J. (2013, August 22). *How to collaborate on Github*. Retrieved from envatotuts+: https://code.tutsplus.com/tutorials/how-to-collaborate-on-github--net-34267

*DBS interactive*. (2015, february 24). Retrieved from the advantages of using task runners: https://www.dbswebsite.com/blog/the-advantages-of-using-task-runners/

EOOD, Z. (2019). *bootstrap studio*. Retrieved from https://bootstrapstudio.io/

Fachat, A. (January 27, 2015). *Learn the workings of Git, not just the commands.*

Fairely, S. (2019, September 14). *Mudbath*. Retrieved from Mudbath&co: https://www.mudbath.com.au/blogs/the-importance-of-version-control

*git*. (n.d.). Retrieved from https://git-scm.com/

*grunt the javascript task runner*. (2018, august 15). Retrieved from https://gruntjs.com/

Hunter, T. (n.d.). *LogRocket*. Retrieved from M: https://blog.logrocket.com/node-js-task-runners-are-they-right-for-you-bb29ea30b7fa

Jeblee, S., & chung, K. (2017, August 2). *COLABORATING ON GITHUB*. Retrieved from UofTCoders: https://uoftcoders.github.io/studyGroup/lessons/git/collaboration/lesson/

McKenzie, C. (2018, November 6). *The server side*. Retrieved from https://www.theserverside.com/video/Git-vs-GitHub-What-is-the-difference-between-them

Mckenzie, C. (2018, November 6). *TheServerSide*. Retrieved from https://www.theserverside.com/video/Git-vs-GitHub-What-is-the-difference-between-them

mudbath. (2019). *Mudbath.* Retrieved from Mudbath&co: https://www.mudbath.com.au/blogs/the-importance-of-version-control

*node.js*. (n.d.). Retrieved from https://nodejs.dev/

*npm*. (n.d.). Retrieved from https://www.npmjs.com/package/express

*Perforce*. (2018, June 26). Retrieved from https://www.perforce.com/blog/vcs/why-version-control-important

Rechsteiner, A. (2017, March 14). *hackerthemes*. Retrieved from https://hackerthemes.com/what-are-bootstrap-themes/

*RisingStack*. (2017). Retrieved from Risingstack: https://blog.risingstack.com/yarn-vs-npm-node-js-package-managers/

*Stackoverflow*. (2017, November 8). Retrieved from https://stackoverflow.com/questions/13321556/difference-between-git-and-github

*techopedia*. (2019). Retrieved from https://www.techopedia.com/definition/4360/software-package

*Tower*. (2013-2019). Retrieved from https://www.git-tower.com/learn/git/ebook/en/command-line/basics/why-use-version-control

*tutorialspoint* . (2019). Retrieved from https://www.tutorialspoint.com/nodejs/nodejs\_express\_framework.htm

*tutorialsPoint*. (2019). Retrieved from https://www.tutorialspoint.com/nodejs/nodejs\_introduction.htm

*w3schools*. (1999-2019). Retrieved from https://www.w3schools.com/nodejs/nodejs\_intro.asp

*wikipedia*. (2019, May 20). Retrieved from https://en.wikipedia.org/wiki/Npm\_(software)

# References

*ATLASIAN BitBucket*. (n.d.). Retrieved from Atlasian: https://www.atlassian.com/git/tutorials/what-is-version-control

*Atlassian Bitbucket*. (n.d.). Retrieved from https://www.atlassian.com/git/tutorials/why-git

*B*. (n.d.). Retrieved from https://getbootstrap.com/docs/4.0/getting-started/build-tools/

BACINGER, T. (2010-2019). *Developers*. Retrieved from https://www.toptal.com/front-end/what-is-bootstrap-a-short-tutorial-on-the-what-why-and-how

Balos, I. (2007-2009). *beanstalk guide*. Retrieved from http://guides.beanstalkapp.com/version-control/git-on-windows.html

Basu, S. (2013, February 28). *Team collaboration with Github*. Retrieved from envatotuts+: https://code.tutsplus.com/articles/team-collaboration-with-github--net-29876

*Bower*. (2016, November 6). Retrieved from https://bower.io/

Cutrell, J. (2013, August 22). *How to collaborate on Github*. Retrieved from envatotuts+: https://code.tutsplus.com/tutorials/how-to-collaborate-on-github--net-34267

*DBS interactive*. (2015, february 24). Retrieved from the advantages of using task runners: https://www.dbswebsite.com/blog/the-advantages-of-using-task-runners/

EOOD, Z. (2019). *bootstrap studio*. Retrieved from https://bootstrapstudio.io/

Fachat, A. (January 27, 2015). *Learn the workings of Git, not just the commands.*

Fairely, S. (2019, September 14). *Mudbath*. Retrieved from Mudbath&co: https://www.mudbath.com.au/blogs/the-importance-of-version-control

*git*. (n.d.). Retrieved from https://git-scm.com/

*grunt the javascript task runner*. (2018, august 15). Retrieved from https://gruntjs.com/

Hunter, T. (n.d.). *LogRocket*. Retrieved from M: https://blog.logrocket.com/node-js-task-runners-are-they-right-for-you-bb29ea30b7fa

Jeblee, S., & chung, K. (2017, August 2). *COLABORATING ON GITHUB*. Retrieved from UofTCoders: https://uoftcoders.github.io/studyGroup/lessons/git/collaboration/lesson/

mudbath. (2019). *Mudbath.* Retrieved from Mudbath&co: https://www.mudbath.com.au/blogs/the-importance-of-version-control

*node.js*. (n.d.). Retrieved from https://nodejs.dev/

*npm*. (n.d.). Retrieved from https://www.npmjs.com/package/express

*Perforce*. (2018, June 26). Retrieved from https://www.perforce.com/blog/vcs/why-version-control-important

Rechsteiner, A. (2017, March 14). *hackerthemes*. Retrieved from https://hackerthemes.com/what-are-bootstrap-themes/

*RisingStack*. (2017). Retrieved from Risingstack: https://blog.risingstack.com/yarn-vs-npm-node-js-package-managers/

*Stackoverflow*. (2017, November 8). Retrieved from https://stackoverflow.com/questions/13321556/difference-between-git-and-github

*techopedia*. (2019). Retrieved from https://www.techopedia.com/definition/4360/software-package

*Tower*. (2013-2019). Retrieved from https://www.git-tower.com/learn/git/ebook/en/command-line/basics/why-use-version-control

*tutorialspoint* . (2019). Retrieved from https://www.tutorialspoint.com/nodejs/nodejs\_express\_framework.htm

*tutorialsPoint*. (2019). Retrieved from https://www.tutorialspoint.com/nodejs/nodejs\_introduction.htm

*w3schools*. (1999-2019). Retrieved from https://www.w3schools.com/nodejs/nodejs\_intro.asp

*wikipedia*. (2019, May 20). Retrieved from https://en.wikipedia.org/wiki/Npm\_(software)

# References

*ATLASIAN BitBucket*. (n.d.). Retrieved from Atlasian: https://www.atlassian.com/git/tutorials/what-is-version-control

*Atlassian Bitbucket*. (n.d.). Retrieved from https://www.atlassian.com/git/tutorials/why-git

*B*. (n.d.). Retrieved from https://getbootstrap.com/docs/4.0/getting-started/build-tools/

BACINGER, T. (2010-2019). *Developers*. Retrieved from https://www.toptal.com/front-end/what-is-bootstrap-a-short-tutorial-on-the-what-why-and-how

Balos, I. (2007-2009). *beanstalk guide*. Retrieved from http://guides.beanstalkapp.com/version-control/git-on-windows.html

Basu, S. (2013, February 28). *Team collaboration with Github*. Retrieved from envatotuts+: https://code.tutsplus.com/articles/team-collaboration-with-github--net-29876

*Bower*. (2016, November 6). Retrieved from https://bower.io/

Cutrell, J. (2013, August 22). *How to collaborate on Github*. Retrieved from envatotuts+: https://code.tutsplus.com/tutorials/how-to-collaborate-on-github--net-34267

*DBS interactive*. (2015, february 24). Retrieved from the advantages of using task runners: https://www.dbswebsite.com/blog/the-advantages-of-using-task-runners/

EOOD, Z. (2019). *bootstrap studio*. Retrieved from https://bootstrapstudio.io/

Fachat, A. (January 27, 2015). *Learn the workings of Git, not just the commands.*

Fairely, S. (2019, September 14). *Mudbath*. Retrieved from Mudbath&co: https://www.mudbath.com.au/blogs/the-importance-of-version-control

*git*. (n.d.). Retrieved from https://git-scm.com/

*grunt the javascript task runner*. (2018, august 15). Retrieved from https://gruntjs.com/

Hunter, T. (n.d.). *LogRocket*. Retrieved from M: https://blog.logrocket.com/node-js-task-runners-are-they-right-for-you-bb29ea30b7fa

Jeblee, S., & chung, K. (2017, August 2). *COLABORATING ON GITHUB*. Retrieved from UofTCoders: https://uoftcoders.github.io/studyGroup/lessons/git/collaboration/lesson/

mudbath. (2019). *Mudbath.* Retrieved from Mudbath&co: https://www.mudbath.com.au/blogs/the-importance-of-version-control

*node.js*. (n.d.). Retrieved from https://nodejs.dev/

*npm*. (n.d.). Retrieved from https://www.npmjs.com/package/express

*Perforce*. (2018, June 26). Retrieved from https://www.perforce.com/blog/vcs/why-version-control-important

Rechsteiner, A. (2017, March 14). *hackerthemes*. Retrieved from https://hackerthemes.com/what-are-bootstrap-themes/

*RisingStack*. (2017). Retrieved from Risingstack: https://blog.risingstack.com/yarn-vs-npm-node-js-package-managers/

*techopedia*. (2019). Retrieved from https://www.techopedia.com/definition/4360/software-package

*Tower*. (2013-2019). Retrieved from https://www.git-tower.com/learn/git/ebook/en/command-line/basics/why-use-version-control

*tutorialspoint* . (2019). Retrieved from https://www.tutorialspoint.com/nodejs/nodejs\_express\_framework.htm

*tutorialsPoint*. (2019). Retrieved from https://www.tutorialspoint.com/nodejs/nodejs\_introduction.htm

*w3schools*. (1999-2019). Retrieved from https://www.w3schools.com/nodejs/nodejs\_intro.asp

*wikipedia*. (2019, May 20). Retrieved from https://en.wikipedia.org/wiki/Npm\_(software)