

## Introduction

The report is about the modern web development technology and tools and how to use them as well as the benefit also the process and purpose to use the different tools for different platforms. The tools are as follow.

- Packages and Package Management System
- Git and GitHub
- Front end tools(Bootstrap)
- Backend tool - Node

## Packages and Package Management System

According to (techopedia, n.d.) Software package is multiple applications and code modules which work together to meet several goals and objectives, like Microsoft office package which includes applications such a Word, Excel, Access and PowerPoint.

According to (TechTarget, 2010) Package Management System (PMS) also known as Package Manager is a program use to install, uninstall and manage the software package.

Software package and package management system are always mention as the connection to UNIX and Unix-driven environments like Linux. A package manager automates where a Linux distribution may consist of thousands of software packages like Advanced Packaging Tool (APT).

According to (Ovens, 2018) Package managers have very important role in the Linux Software management. The fact is initial stage of software, products use to be tested strictly for bugs and other defects. But now in modern days software are released through internet with the intent that if incase of bugs that would be fixed using new versions of the software which is one of the benefit of using packages.

### WHAT IS PACKAGE?

According to (Ovens, 2018) Packages were invented to sort the complexity of the software updates. Packages collects multiple data files together into a single file so that the files will be transferred and stored easily or simple compressed file to reduce the storage space.

Linux have various package formats of their own like

1. .deb: This was the first package to be created and used by Debian, Linux Mint and several derivatives.
2. .rpm: This package is known as Red Hat Manager and used by Red hat and red hat and many other distributor.
3. .tar.xz. This is just a compressed tarball.

Also packages we use as a developers in modern technology are like:

1. Bootstrap: Bootstrap carry full configuration theme of TYPO3 and the CSS framework.
2. Express.js: this is the framework for web applications and express is very flexible and minimal to use.

**Most Prominent Package Manager:**

As per (Ovens, 2018) Package Managers are used to interact with the software repositories.

Following are the most Prominent Package managers

1. RPM-based package managers: Red hat used a package manager known as RPM.
2. Arch-based package managers: Arch-Linux uses a package manager known as pacman.

The advantage of using package is to work fast and package will get all the software or frameworks we need to develop a website or any software and updates automatically.

We use package to work fast and to have a bug free and quality work.

## Git and GitHub

According to (Kanaparthi, 2018) Git is known as a version control System which means git help to maintain any documents or project as well as git help in tracking all the files and work that anyone or developers are working with.

Git is very useful for developers to maintain their project and track whatever changes they make also can see their work around the world. Git is very helpful of tracking the work and can also revert back the work to old no matter how many times we have made changes.

One very important aspect of using Git is it can track ofcourse and at the same time help others from anywhere any place across world to work together without affecting each other's work or changes they make.

Advantage of working with Git is that we can have fast performance and can learn quick as it is very easy process to learn.

GitHub is the version control of the web base service. It is a social site where developers or anyone can create their profile to work on projects or save their work. Hence people can see each other's work and can look for faults or errors if any then can suggest for changes. Here this way people can learn better and improve their work. We can also consider GitHub as the hosting platform for Git Repositories.

The basic terminology of Git and GitHub are mention below:

- Repository: This help git to track the files that changes in or from GitHub.
- Branch: It is a directory in GitHub.
- Fork: This is where one can see copy of another person's work that is available in a profile.
- Clone: clone mean copies the same work in the computer instead of having in server.
- Commit: commit is to track the file in the repository.
- Master: It is the default active branch in the git repository.

Few Git commands are as follows:

- Git init – create a new file
- Git add – to add the file
- Git commit -m "message" – save changes of the file to main repository
- Git push – save file to the server

## Front end tools (Bootstrap)

According to (Bootstrap (front-end framework), 2017) bootstrap is a package of CSS web framework to use for the development of responsive website and give styles of colour, sizes, fonts and project layout.

As well as bootstrap has several JavaScript components as a jQuery plugin, which gives an extra element like dialog boxes, tooltip and carousal. All bootstrap component has the HTML and CSS with JavaScript language.

Most important factor of bootstrap is its layout because that effect complete the webpage. Container is another basic layout of bootstrap where all the elements of the page are placed on it.

Bootstrap have two types of container

- Fixed-width container
- Fluid-width container

### Advantage of Using Bootstrap

According (Gimmer, 2014) there are several benefits of using bootstrap are as listed below

1. Speed of the development is very fast because you don't have to write code from scratch instead if using bootstrap, we get code which are already in the bootstrap so the development process is very quick.
2. Website will be responsive as there are code itself in bootstrap which make your work easy and responsive.
3. Bootstrap make the work very flexible.
4. Bootstrap have a JavaScript library with lots of functions to make the website more prominent and user friendly.

## Node (Backend tool)

According (kononenko, 2018) now millions of developers used Node.js to create the backend of web applications. Also, node depends on an active open source community to build several packages with node package manager or NPM.

### NPM:

NPM have hundreds and thousands of "packages" that offer specialized functions which are written in JavaScript. All NPM packages are built by individual developers or a team and contribute to the registry. In the code all these packages are tracked in the package as json. file.

### NPM command Line:

The following are the command line when you download Node and NPM to your local computer.

- First command is "npm install".
- If installing express package then the command is "npm install express".
- Then after downloading the npm express "npm uninstall express".

According to (Dierx, 2019) (Dierx, 2019) developers use Node.js package to get the useful interface to work with other developers and the npm should get updated very frequently because the npm package Manager is different project from node.js. To update the latest npm the command to use is "Npm install npm@latest -g".

According to (CAPAN, n.d.) Node build all network applications fast and also it is scalable which handles large number of applications concurrently at the same time.

### **NPM: The Node Package Manager**

Node.js is a built-in-support for package management using NPM which is a tool and has a default node.js installation.

Few most popular and useful node.js module are as follows:

1. `express/express.js`: we can simply say Express which is a web development framework of node.js
2. `Connect`: connect is an extensible framework of node.js known as HTTP server which provide large amount of performance as a middleware.

### **Where Node.js Used:**

Node.js can be used mostly in Chats which is a multi-use application. Also Node.js can be used for the server-side web applications.

### **Benefits of Using Node.js:**

1. If in case the application doesn't have CPU computation then we can build on JavaScript.
2. It is SEO-friendly where crawler receives response from the HTML.
3. Node.js has an active open source and a JavaScript based ecosystem.

## References

(n.d.). Retrieved from techopedia: <https://www.techopedia.com/definition/4360/software-package>

(2010, april ). Retrieved from TechTarget:

<https://searchmobilecomputing.techtarget.com/definition/package-manager-or-package-management-system-PMS>

*Bootstrap (front-end framework)*. (2017, February). Retrieved from Wikipedia:

[https://en.wikipedia.org/wiki/Bootstrap\\_\(front-end\\_framework\)](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework))

CAPAN, T. (n.d.). Retrieved from Toptal: <https://www.toptal.com/nodejs/why-the-hell-would-i-use-node-js>

Dierx, M. W. (2019, April 15). Retrieved from SitePoint: <https://www.sitepoint.com/beginners-guide-node-package-manager/>

Kanaparthi, V. (2018, February 14). *Git and GitHub in a NutShell*. Retrieved from codeburst.io:

<https://codeburst.io/git-and-github-in-a-nutshell-b0a3cc06458f>

kononenko, k. (2018, october 24). *Dev*. Retrieved from <https://dev.to/kbk0125/node-package-manager-npm-explained-by-directing-a-movie-359>

Ovens, S. (2018, JULY 26). *redhat*. Retrieved from opensource.com:

<https://opensource.com/article/18/7/evolution-package-managers>

## References

(n.d.). Retrieved from techopedia: <https://www.techopedia.com/definition/4360/software-package>

(2010, april ). Retrieved from TechTarget:  
<https://searchmobilecomputing.techtarget.com/definition/package-manager-or-package-management-system-PMS>

*Bootstrap (front-end framework)*. (2017, February). Retrieved from Wikipedia:  
[https://en.wikipedia.org/wiki/Bootstrap\\_\(front-end\\_framework\)](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework))

CAPAN, T. (n.d.). Retrieved from Toptal: <https://www.toptal.com/nodejs/why-the-hell-would-i-use-node-js>

Dierx, M. W. (2019, April 15). Retrieved from SitePoint: <https://www.sitepoint.com/beginners-guide-node-package-manager/>

Gimmer, C. (2014, April 23). *Top 5 Reasons to use Bootstrap*. Retrieved from BootsTrapBay:  
<https://bootstrapbay.com/blog/reasons-to-use-bootstrap/>

Kanaparthi, V. (2018, February 14). *Git and GitHub in a NutShell*. Retrieved from codeburst.io:  
<https://codeburst.io/git-and-github-in-a-nutshell-b0a3cc06458f>

kononenko, k. (2018, october 24). *Dev*. Retrieved from <https://dev.to/kbk0125/node-package-manager-npm-explained-by-directing-a-movie-359>

Ovens, S. (2018, JULY 26). *redhat*. Retrieved from opensource.com:  
<https://opensource.com/article/18/7/evolution-package-managers>

## References

(n.d.). Retrieved from techopedia: <https://www.techopedia.com/definition/4360/software-package>

(2010, april ). Retrieved from TechTarget:  
<https://searchmobilecomputing.techtarget.com/definition/package-manager-or-package-management-system-PMS>

CAPAN, T. (n.d.). Retrieved from Toptal: <https://www.toptal.com/nodejs/why-the-hell-would-i-use-node-js>

Dierx, M. W. (2019, April 15). Retrieved from SitePoint: <https://www.sitepoint.com/beginners-guide-node-package-manager/>

Kanaparthi, V. (2018, February 14). *Git and GitHub in a NutShell*. Retrieved from codeburst.io:  
<https://codeburst.io/git-and-github-in-a-nutshell-b0a3cc06458f>

kononenko, k. (2018, october 24). *Dev*. Retrieved from <https://dev.to/kbk0125/node-package-manager-npm-explained-by-directing-a-movie-359>

Ovens, S. (2018, JULY 26). *redhat*. Retrieved from [opensource.com](https://opensource.com/article/18/7/evolution-package-managers):  
<https://opensource.com/article/18/7/evolution-package-managers>

## References

(n.d.). Retrieved from techopedia: <https://www.techopedia.com/definition/4360/software-package>

(2010, april ). Retrieved from TechTarget:  
<https://searchmobilecomputing.techtarget.com/definition/package-manager-or-package-management-system-PMS>

CAPAN, T. (n.d.). Retrieved from Toptal: <https://www.toptal.com/nodejs/why-the-hell-would-i-use-node-js>

Dierx, M. W. (2019, April 15). Retrieved from SitePoint: <https://www.sitepoint.com/beginners-guide-node-package-manager/>

kononenko, k. (2018, october 24). *Dev*. Retrieved from <https://dev.to/kbk0125/node-package-manager-npm-explained-by-directing-a-movie-359>

Ovens, S. (2018, JULY 26). *redhat*. Retrieved from [opensource.com](https://opensource.com/article/18/7/evolution-package-managers):  
<https://opensource.com/article/18/7/evolution-package-managers>

## References

(n.d.). Retrieved from techopedia: <https://www.techopedia.com/definition/4360/software-package>

(2010, april ). Retrieved from TechTarget:  
<https://searchmobilecomputing.techtarget.com/definition/package-manager-or-package-management-system-PMS>

Dierx, M. W. (2019, April 15). Retrieved from SitePoint: <https://www.sitepoint.com/beginners-guide-node-package-manager/>

kononenko, k. (2018, october 24). *Dev*. Retrieved from <https://dev.to/kbk0125/node-package-manager-npm-explained-by-directing-a-movie-359>

Ovens, S. (2018, JULY 26). *redhat*. Retrieved from [opensource.com](https://opensource.com/article/18/7/evolution-package-managers):  
<https://opensource.com/article/18/7/evolution-package-managers>

## References

(n.d.). Retrieved from techopedia: <https://www.techopedia.com/definition/4360/software-package>

(2010, april ). Retrieved from TechTarget:  
<https://searchmobilecomputing.techtarget.com/definition/package-manager-or-package-management-system-PMS>

kononenko, k. (2018, october 24). *Dev*. Retrieved from <https://dev.to/kbk0125/node-package-manager-npm-explained-by-directing-a-movie-359>

Ovens, S. (2018, JULY 26). *redhat*. Retrieved from [opensource.com](https://opensource.com/article/18/7/evolution-package-managers):  
<https://opensource.com/article/18/7/evolution-package-managers>

## References

(n.d.). Retrieved from techopedia: <https://www.techopedia.com/definition/4360/software-package>

(2010, april ). Retrieved from TechTarget:  
<https://searchmobilecomputing.techtarget.com/definition/package-manager-or-package-management-system-PMS>

kononenko, k. (2018, october 24). *Dev*. Retrieved from <https://dev.to/kbk0125/node-package-manager-npm-explained-by-directing-a-movie-359>

Ovens, S. (2018, JULY 26). *redhat*. Retrieved from [opensource.com](https://opensource.com/article/18/7/evolution-package-managers):  
<https://opensource.com/article/18/7/evolution-package-managers>

## References

(n.d.). Retrieved from techopedia: <https://www.techopedia.com/definition/4360/software-package>

(2010, april ). Retrieved from TechTarget:

<https://searchmobilecomputing.techtarget.com/definition/package-manager-or-package-management-system-PMS>

kononenko, k. (2018, october 24). *Dev*. Retrieved from <https://dev.to/kbk0125/node-package-manager-npm-explained-by-directing-a-movie-359>

Ovens, S. (2018, JULY 26). *redhat*. Retrieved from opensource.com:

<https://opensource.com/article/18/7/evolution-package-managers>

## References

(n.d.). Retrieved from techopedia: <https://www.techopedia.com/definition/4360/software-package>

(2010, april ). Retrieved from TechTarget:

<https://searchmobilecomputing.techtarget.com/definition/package-manager-or-package-management-system-PMS>

kononenko, k. (2018, october 24). *Dev*. Retrieved from <https://dev.to/kbk0125/node-package-manager-npm-explained-by-directing-a-movie-359>

Ovens, S. (2018, JULY 26). *redhat*. Retrieved from opensource.com:

<https://opensource.com/article/18/7/evolution-package-managers>