**DEVELOPMENT SUPPORT TOOL**

**ECLERX**

**A training report**

Submitted in partial fulfillment of the requirements for the award of degree of

# B.Tech

# (CSE) Submitted to

**LOVELY PROFESSIONAL UNIVERSITY PHAGWARA, PUNJAB**



**From 05/07/19 to 12/12/19 SUBMITTED BY**

## Name of student: Prerna Bhatia Submitted to:

**Registration Number: 11611526 Name of Supervisor**

**Signature of the student: Designation**

**To whom so ever it may concern**

This is to certify that **Prerna Bhatia, 11611526** from Lovely Professional University, Phagwara, Punjab, has worked as a trainee in **ECLERX** on “**Development Support Tool**” under my supervision from **August, 2019** to **December, 2019**. It is further stated that the work carried out by the student is a record of original work to the best of my knowledge for the partial fulfillment of the requirements for the award of the degree, degree name.

Name of External Supervisor Name of Internal Supervisor

Designation of the External Supervisor Designation of the Internal Supervisor

Signature of the external Supervisor Signature of the Internal Supervisor

Dated: Dated:

**To whom so ever it may concern**

I, **PRERNA BHATIA, 11611526** hereby declare that the work done by me on “**Development Support Tool**” from **August, 2019** to **December, 2019**, under the supervision of **Rahul Vanpully Associate Program Manager, Eclerx,** and **Name of Internal supervisor**, **Designation,** Lovely professional University, Phagwara, Punjab, is a record of original work for the partial fulfillment of the requirements for the award of the degree, **B.Tech.**

PRERNA BHATIA (11611526)

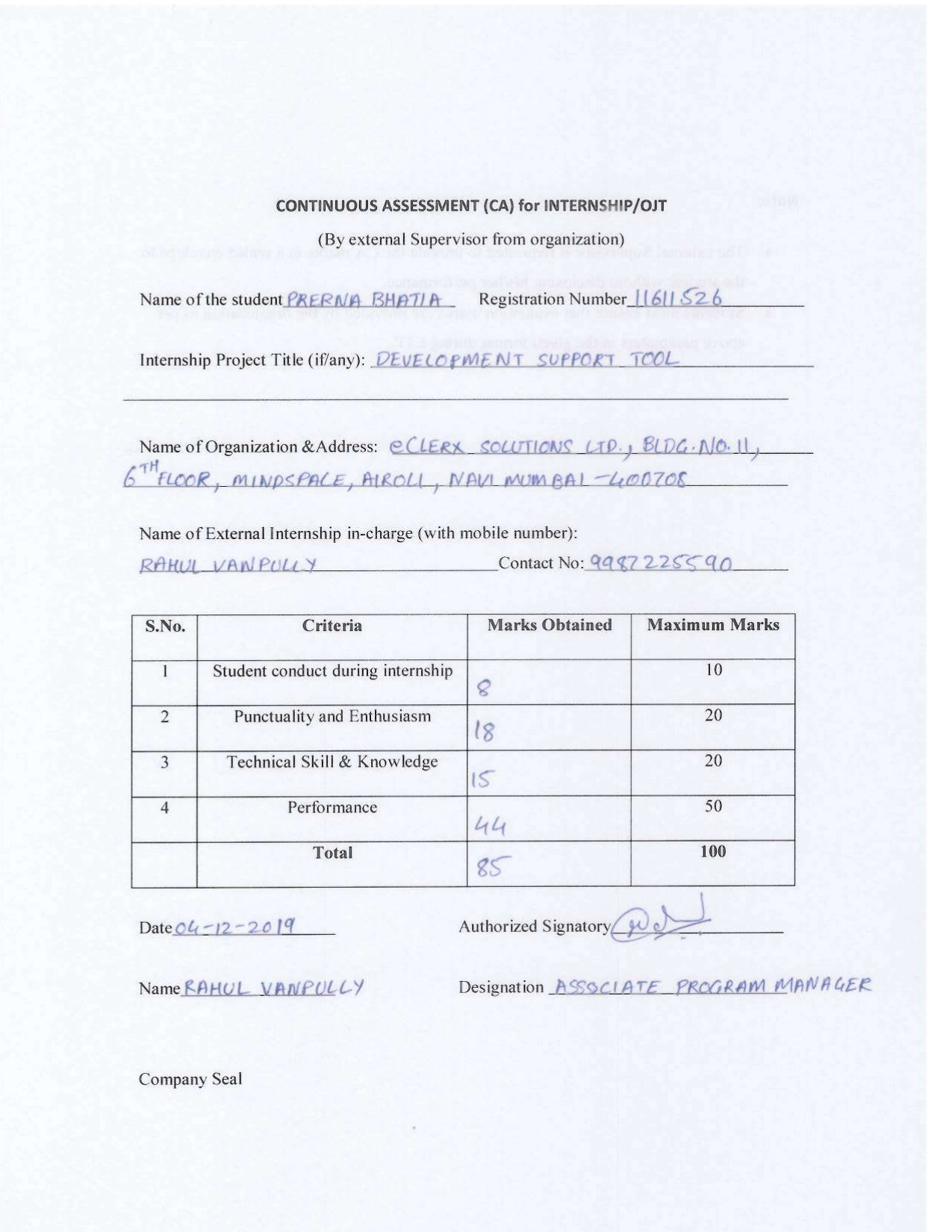
Signature of the student

Dated:

**Training Letter from EClerx**



**CONTINUOUS ASSESSMENT (CA) for INTERNSHIP**



**TABLE OF CONTENT**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Title** | **Page** |
| 1 | Declaration by Supervisors | 1 |
| 2 | Declaration by Student | 2 |
| 3 | Training Certification from organization | 3 |
| 4 | Continuous Assessment (CA) for Internship | 4 |
| 5 | Chapter-1 Introduction Of The Company | 6 |
| 6 | Chapter-2 Introduction Of The Project Undertaken | 9 |
| 7 | Chapter-2.1 AutoDesk 3D Website | 9 |
| 8 | Chapter-2.2 Gulp.Js | 13 |
| 9 | Chapter-2.3 Koa.Js and MongoDB | 22 |
| 10 | Chapter-3 Brief Description of The Work Done | 25 |
| 11 | Conclusion | 28 |
| 12 | References | 30 |

**1. INTRODUCTION OF THE COMPANY**

* 1. **Company’s Vision and Mission**
* No matter our achievements or number of clients, philanthropy is fundamental to who we are. EClerx has a strong vision of what it means to be a responsible company in today’s society. We place great importance on the role our firm and employees can play in the communities in which we live and work.
* To maximize our impact, we actively partner with NGOs and education stakeholders to find and advance effective solutions that address today’s most pressing development and educational challenges. Our partners work to secure the well-being of disadvantaged Indian youth by supporting education and child welfare initiatives throughout India.
* This includes vocational education and training for children living in city slums, rural villages and tribal areas as well as those who are unemployed or minimally employed. The eClerx Cares Committee approves and monitors projects we fund through our NGO partners.
  1. **Origin and growth of company**
* EClerx Services Limited is engaged in providing Knowledge Process Outsourcing (KPO) services to global companies. The company provides data management analytics solutions and process outsourcing services to a host of global clients through a network of multiple locations in India and abroad.
* In August 2007 the Company was converted to a public limited company and the name was changed into eClerx Services Limited. EClerx Services completed its Initial Public Offer (IPO) and the equity shares were listed on the National Stock Exchange of India Ltd. (NSE) and the Bombay Stock Exchange Ltd. (BSE) effective December 31 2007.
* During the financial year ended 31 March 2010 which marked completion of 10 years by eClerx Services the company recorded a turnover of more than Rs. 2500 million for the first time in its history.

**1.3 Various departments and their functions**

**1.3.1 ECLERX DIGITAL**

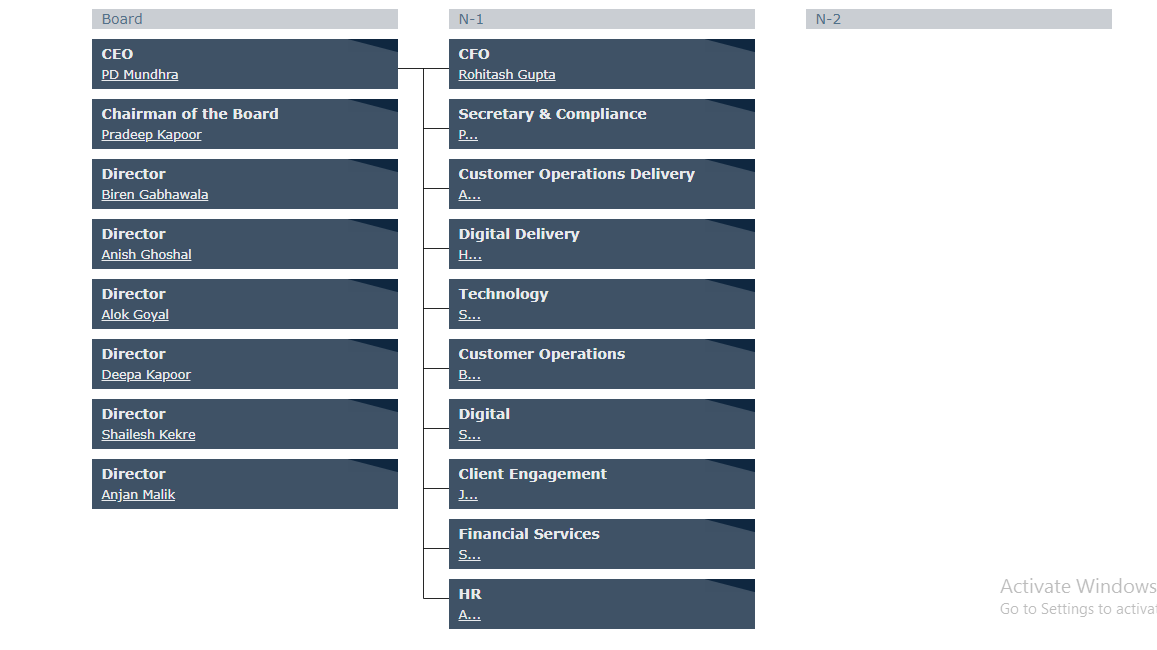
* EClerx Digital is the trusted partner of choice to the world’s largest global brands for creative production, ecommerce / web operations, and analytics & insights services. We improve profitability for their digital businesses.
* Our team of 3000+ full-time digital delivery employees at our five production hubs in Mumbai, Pune, Chandigarh, Verona and Phuket apply deep digital expertise to effectively support the most demanding global clients by employing a follow the sun delivery model.

**1.3.2 ECLERX CUSTOMER OPERATIONS**

* Our multi-disciplinary experts are committed to our clients for the duration of their engagement. Each diverse team of specialists is led by a forward-thinking eClerx Customer Operations Product Manager dedicated to improving customer experience and operational efficiencies. We operate as an extension of your company.
* We Our proven methodologies are crafted to respond quickly to your customers’ needs and goals. eClerx Customer Operations’ quality monitoring and insights, digital care services, and advanced technical operations are intuitive and designed to improve outcomes for you and your customers

**1.3.3 ECLERX FINANCIAL MARKET**

* For financial organizations across the world, eClerx Markets, offers consulting, technological innovation, and process management expertise to uniquely solve operational challenges.
* With nearly two decades of industry experience complemented by the application of smart automation and robotics, our team of experts deliver holistic solutions across the trade life cycle, change management, data analytics, compliance, cash securities operations, document digitization and generation, and outreach.
  1. **Organization chart of the company**

****

**2. INTRODUCTION OF THE PROJECTS UNDERTAKEN**

**2.1: AUTODESK WEBSITE**

* + 1. **Objectives of the work undertaken**
* We are required to design the homepage of the Autodesk 3D website. I was supposed to design a particular module from the website which I am attaching below. I learned bootstrap to make it responsive so that the website can be accessed through any internet connected device and it will show same results every time.
* The role of your website is to help validate and support communications of the sales team throughout the sales process, this could be a good goal to set. It can be tricky to measure if the website’s primary role is not to generate leads but instead to help validate and support the leads generated offline by the sales team.
* Similar to marketing, this goal measures the ability of a website to sell (convert a website visitor into a customer). It is typically measured as a percentage of website visitors who become paying customers.
  + 1. **Scope of the Work**
* This is arguably the most popular goal for business websites that have lead generation as their main purpose. This goal is best for companies that want to get more sales leads through improving their website’s marketing performance. To measure lead increase, I recommend setting a percentage rate increase as opposed to an absolute lead count.
* This accounts for monthly traffic fluctuations and allows for meaningful historical comparisons. Also, make sure to track qualified leads versus total general leads and set specifications as to what is considered a qualified lead.
* Doing so will also measure the quality of leads you are getting. This goal lends to measuring how effective your website is in educating users about your new products or services when you are not necessarily “selling” online.

**2.1.3 Importance and Applicability**

* Creating a website is not just to promote your business, it helps in creating brand awareness, gives a platform for people to interact, approach and discover as well. To market a product or service effectively, you should have a web page that is viewable in a variety of devices.
* In today’s world you have smart phones, tablets, laptops, desktops and an ongoing increase in the kind of devices and form factors. Hence to solve the issue of being able to view websites or content on all such devices effectively. The speed of development is one of its major advantages. If you want to develop an application or a website promptly, it is imperative to consider using Bootstrap.
* It helps to save your coding effort by offering less CSS functionality and pre-built blocks of code rather than structuring code from the scratch. Ready-made themes of Bootstrap will help achieve your needs through a faster route.

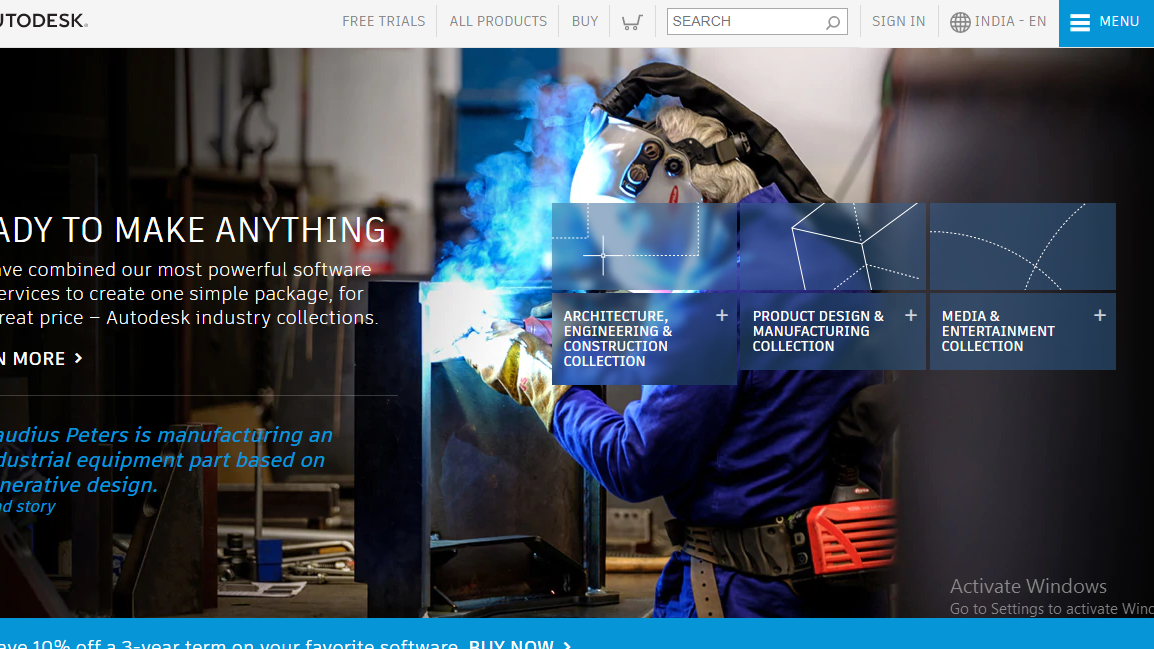


Figure 2.1: Autodesk Website

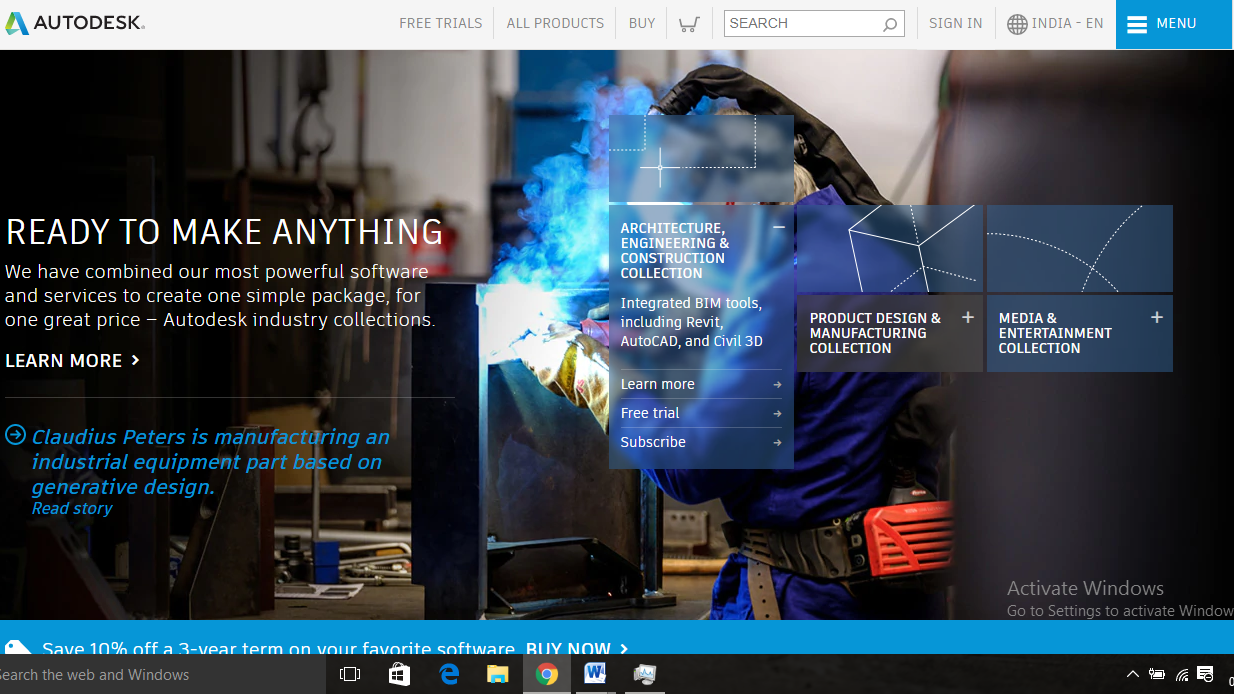


Figure 2.2: Autodesk Website – 2

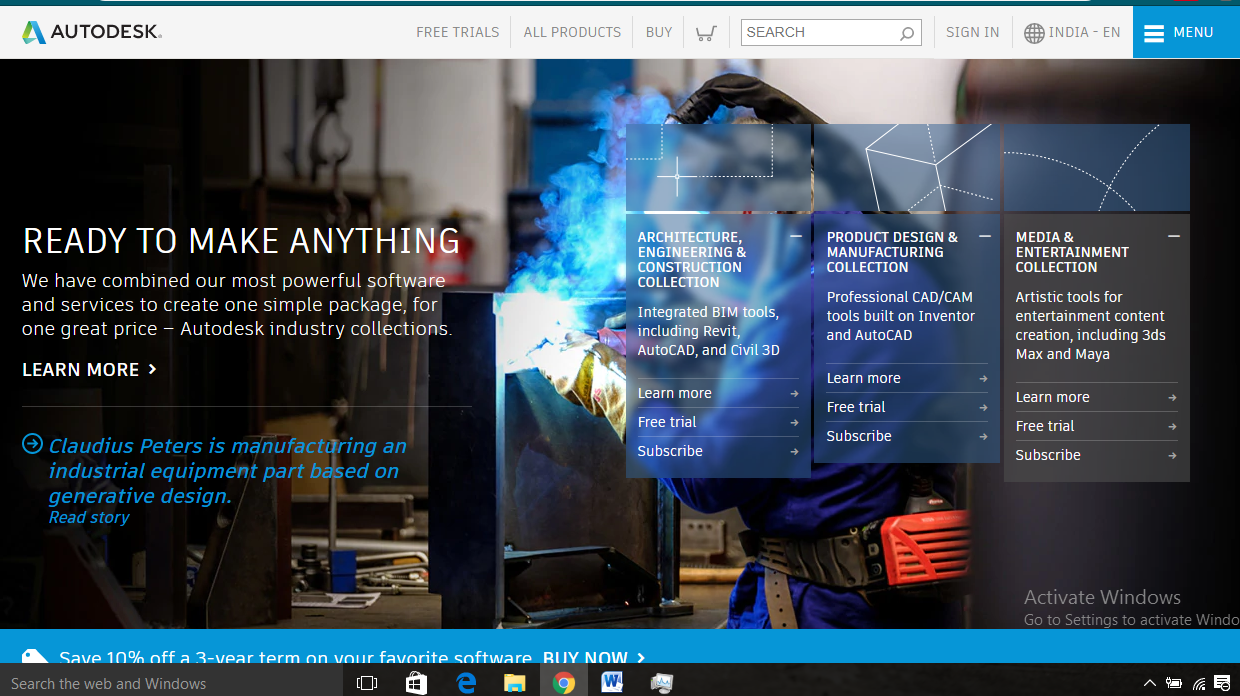


Figure 2.3: Autodesk Website – 3

* + 1. **Role and profile**
* I was responsible for the coding, design and layout of a website according to a company’s specifications. As the role takes into consideration user experience and function, a certain level of both graphic design and computer programming is necessary.
* Once a website has been created, a Web Developer will generally assist with the maintenance and upkeep of the website. A Web Developer must have experience in the planning and the delivery of web applications across multiple platforms. A typical Web Developer job description should cover.
* Creating websites/a website using standard HTML/CSS/BOOTSTRAP practices. Working closely with web designers and programmers to produce the website.

**2.2: GULP.JS**

* + 1. **Objectives of the work undertaken**
* Task-runners like gulp and [Grunt](https://en.wikipedia.org/wiki/Grunt_(software)) are built on Node.js rather than [npm](https://en.wikipedia.org/wiki/Node_Package_Manager), because the basic npm scripts are inefficient when executing multiple tasks. Even though some developers prefer [npm](https://en.wikipedia.org/wiki/Node_Package_Manager) scripts because they can be simple and easy to implement, there are numerous ways where gulp and Grunt seem to have an advantage over each other and the default provided scripts.
* Grunt runs tasks by transforming files and saves as new ones in temporary folders and the output of one task is taken as input for another and so on until the output reaches the destination folder.
* This involves a lot of [I/O](https://en.wikipedia.org/wiki/I/O) calls and creation of many temporary files. Whereas gulp streams through the file system and does not require any of these temporary locations decreasing the number of I/O calls thus, improving performance.

* + 1. **Scope of the Work**
* Gulp is a build system for automating tasks. The following is few listing which can be achieved with the help of gulpjs. For any changes and then perform some kind actions on it.
* For eg. Assume you change the styles.css or script.js files minify it manually, instead of you minifying it gulp does it for you.
* Combine files (concatenate) into another file.
* Minimize files (making it small).
* Make it do it every time a file is modified.

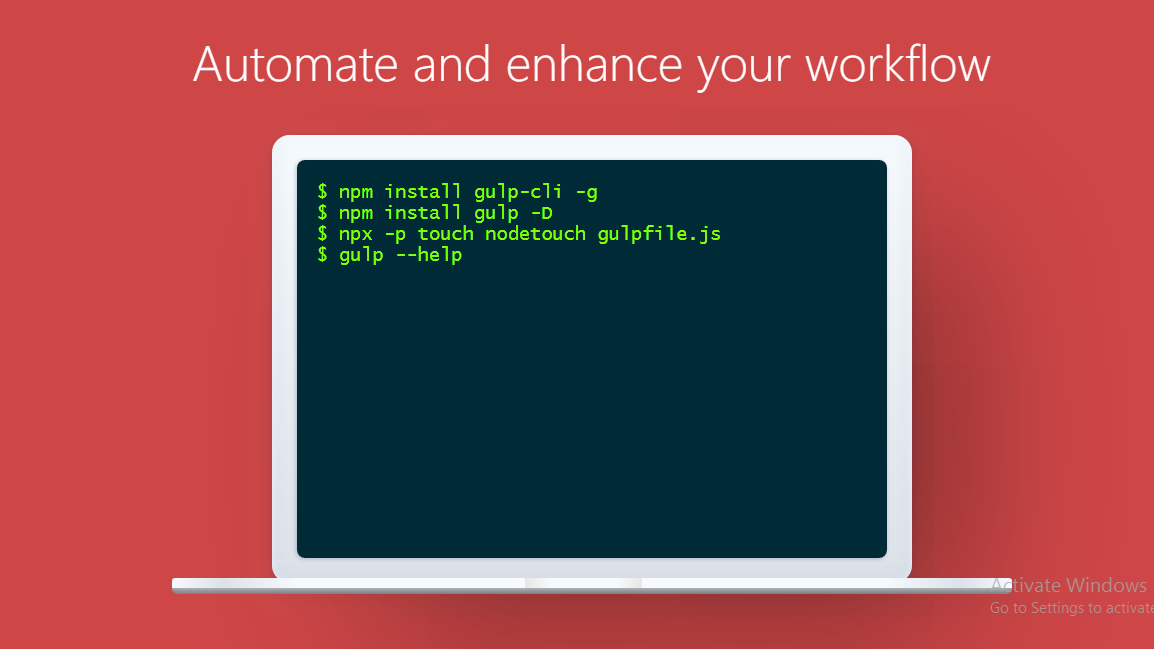


Figure 2.4: Installation of Gulp.Js

**VARIOUS PLUGINs IN GULP.JS**

**1. gulp-sass:**

* You can choose whether to use [**Dart Sass**](http://sass-lang.com/dart-sass) or [**Node Sass**](https://github.com/sass/node-sass) by setting the sass. Compiler property. Node Sass will be used by default, but it's strongly recommended that you set it explicitly for forwards-compatibility in case the default ever changes.
* Note that when using Dart Sass, **synchronous compilation is twice as fast as asynchronous compilation** by default, due to the overhead of asynchronous callbacks. To avoid this overhead, you can use the [**fibers**](https://www.npmjs.com/package/fibers) package to call asynchronous importers from the synchronous code path.

**2. gulp-clean-css:**

* This is just a simple [**gulp**](https://github.com/gulpjs/gulp) plugin, which means it's nothing more than a thin wrapper around clean-css. If it looks like you are having CSS related issues, please contact [**clean-css**](https://github.com/jakubpawlowicz/clean-css/issues). Only create a new issue if it looks like you're having a problem with the plugin itself.
* Useful for returning details from the underlying [**minify()**](https://github.com/jakubpawlowicz/clean-css#using-api) call. An example use case could include logging stats of the minified file. In addition to the default object, gulp-clean-css provides the file name and path for further analysis.

**3. gulp-concat-css:**

concatCss(targetFile, options)

* targetFile: The relative path of the generated file containing the concatenated css
* options: (since 2.1.0)
  + inlineImports: (default true) Inline any local import statement found.
  + rebaseUrls: (default true) Adjust any relative URL to the location of the target file.
  + includePaths: (default []) Include additional paths when inlining imports.
  + commonBase: (default to the base property of the first file) Common base path from which resolving files and urls.

**4. gulp-spritesmithe:**

* Convert a set of images into a sprite sheet and CSS variables via [gulp](http://gulpjs.com/). This is the official port of [grunt-spritesmith](https://github.com/Ensighten/grunt-spritesmith), the [grunt](https://gruntjs.com/) equivalent of a wrapper around [spritesmith](https://github.com/Ensighten/spritesmith). We have completed our integration with streaming outputs from engines. As a result, [Vinyl](https://github.com/gulpjs/vinyl) image files will have stream contents which were previously buffers.
* Install the module with: npm install gulp.spritesmith. In addition to the spriteData stream, we offer individual streams for images and CSS. This allows for image optimization and CSS minification. gulp.spritesmith supports retina spritesheet generation via retinaSrcFilter and retinaImgName.

**5. gulp-imagemin:**

* Images are everywhere across the internet. You would be hard pressed to find a single page or application that doesn’t contain at least one image in some form or another. Images are great way to help tell stories and emphasize critical parts of our lives.
* Gulp and an npm package called gulp-imagemin to reduce the size of your images on the fly.

**6. gulp-htmlmin:**

const gulp **=** require('gulp');

const htmlmin **=** require('gulp-htmlmin');

gulp.task('minify', () => {

**return** gulp.src('src/\*.html')

    .pipe(htmlmin({ collapseWhitespace**:** true }))

    .pipe(gulp.dest('dist'));

});

**OUTPUT OF ABOVE PLUGINS**

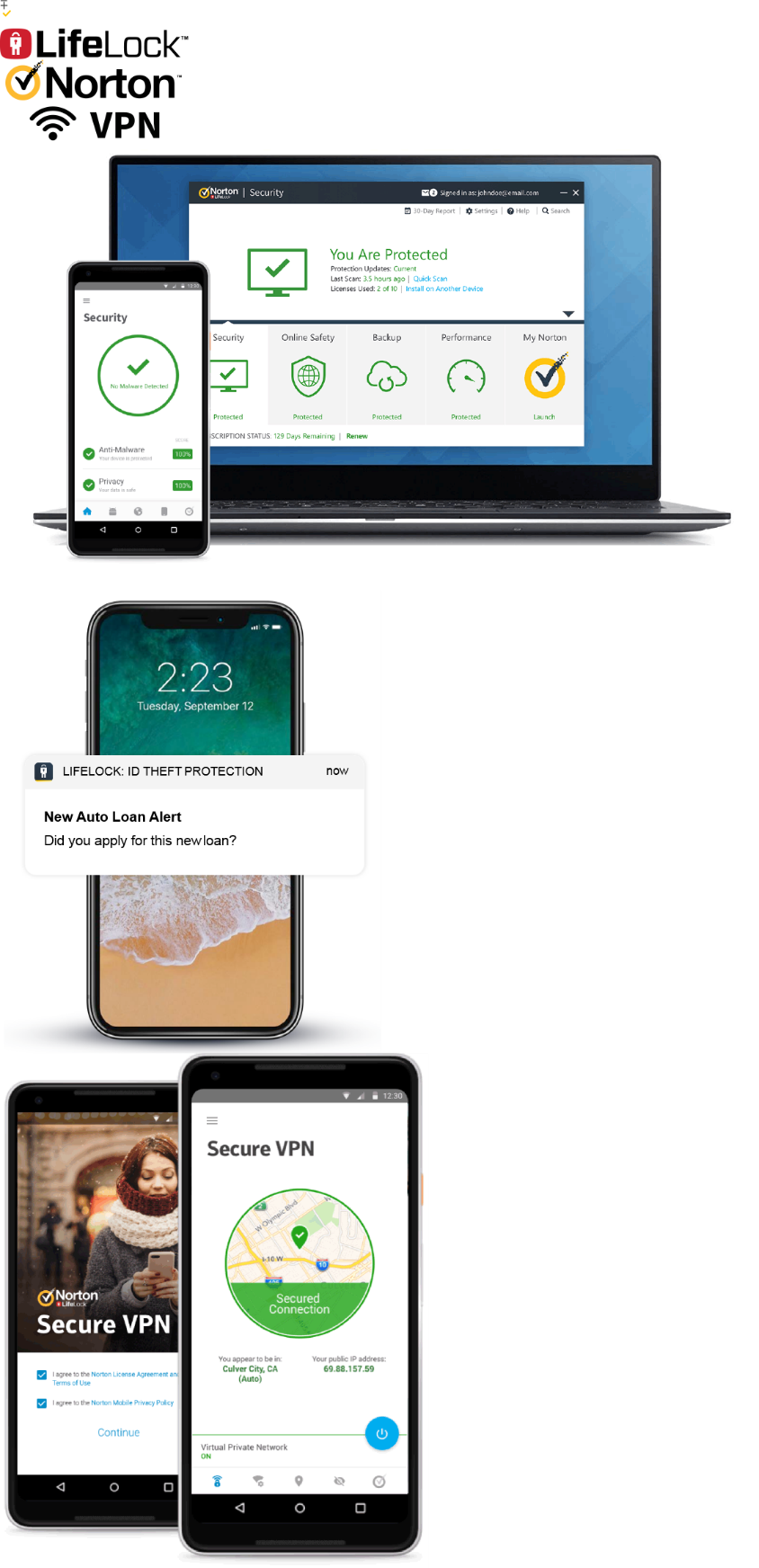


Figure 2.8: Top to Down

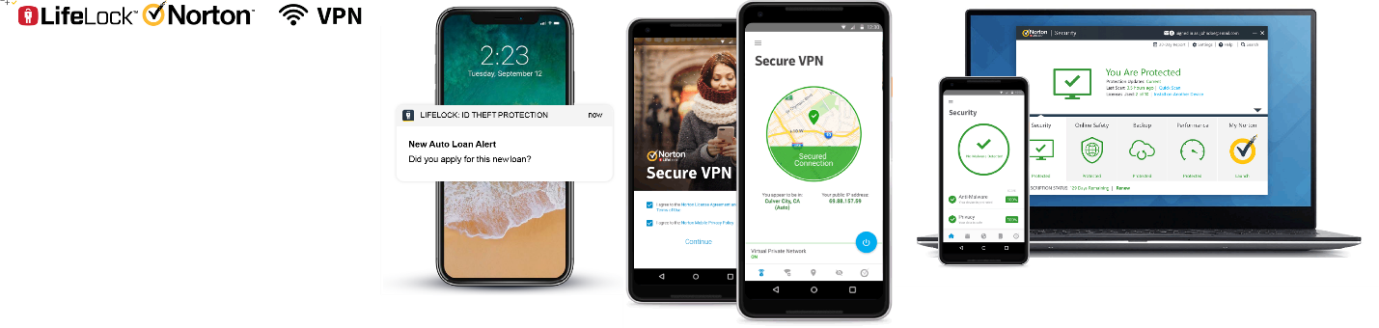


Figure 2.9: Left to Right

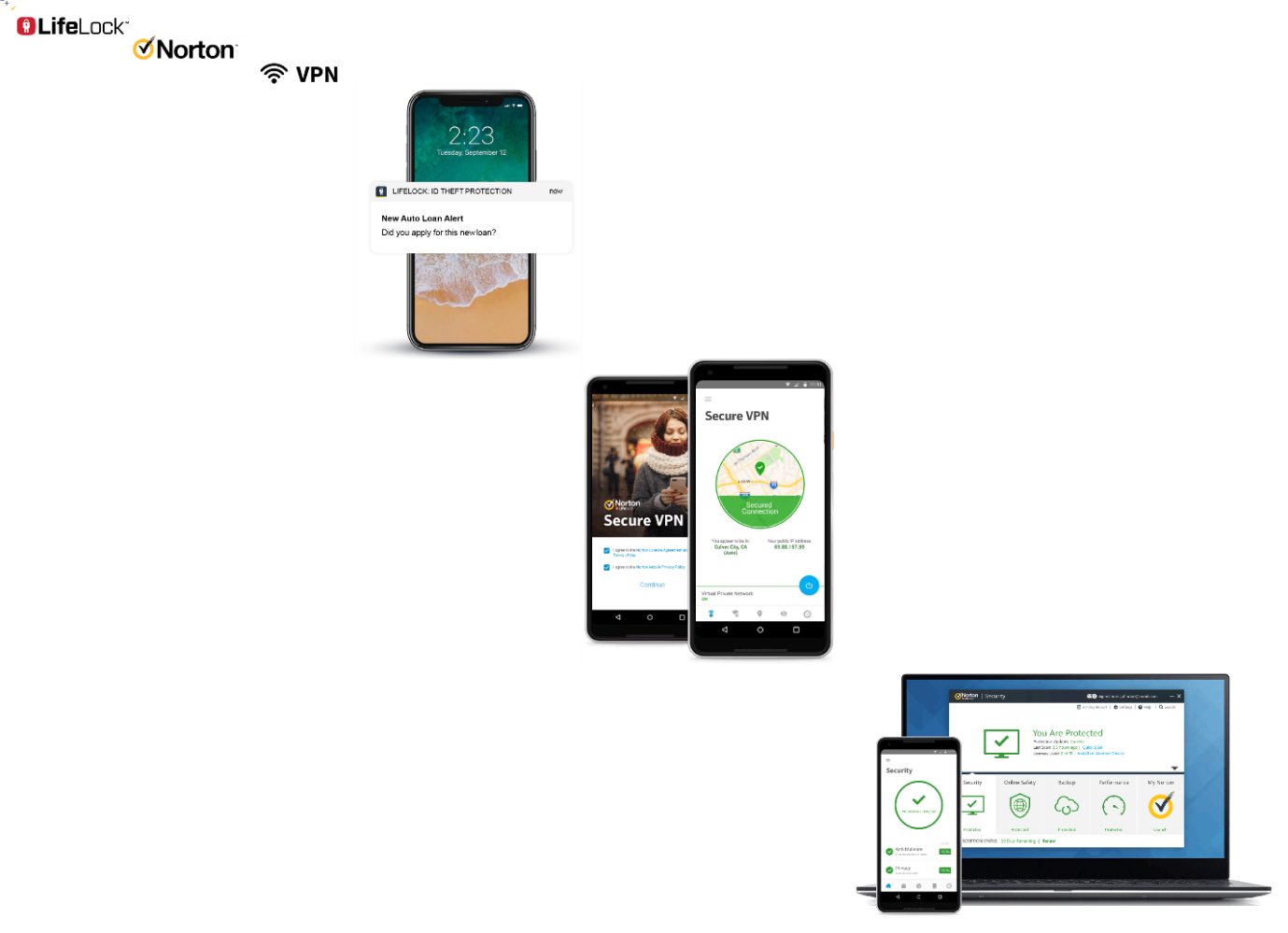


Figure 2.10: Diagonal

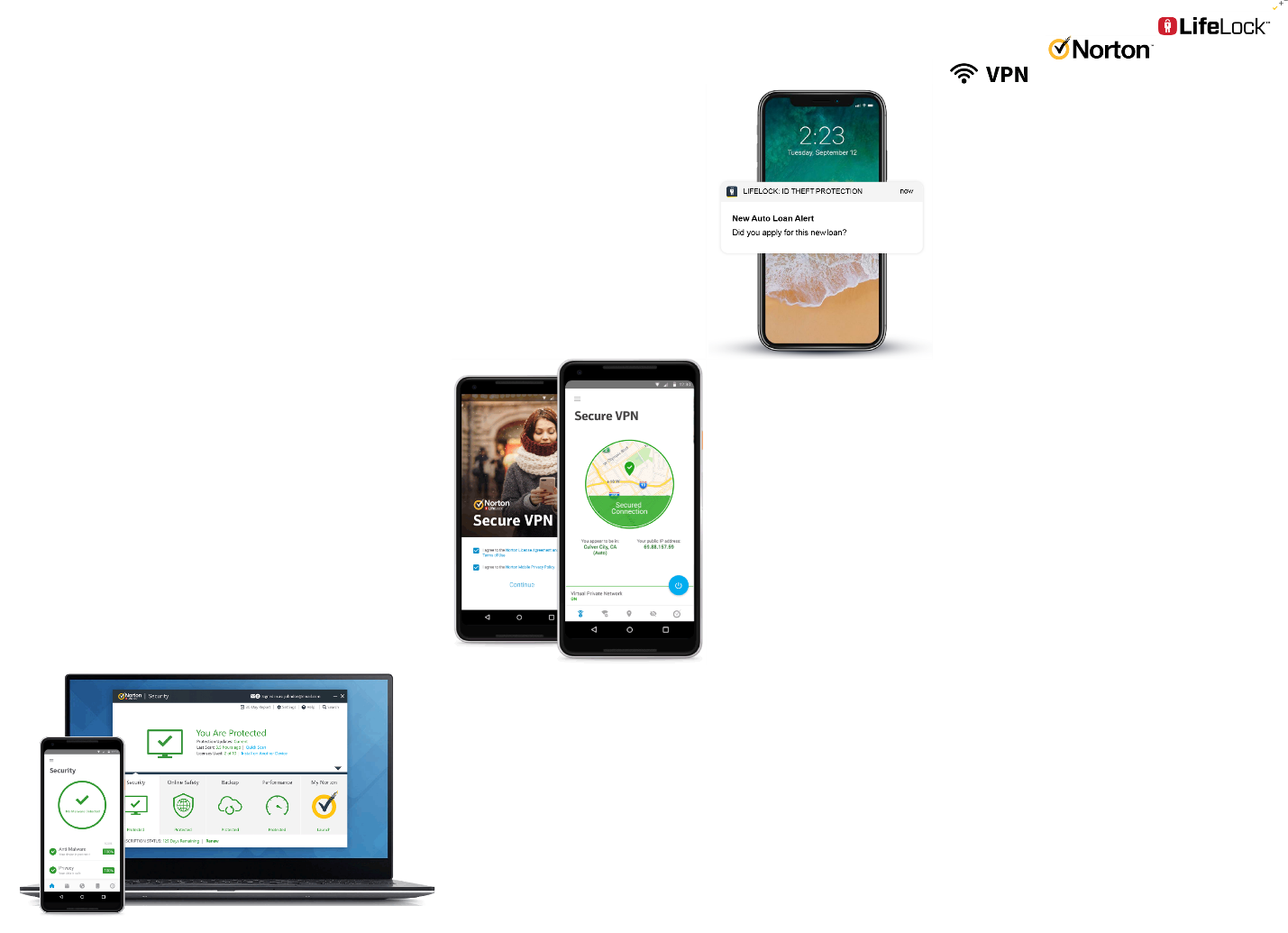


Figure 2.11: Alt Diagonal

* + 1. **Importance and Applicability**

### 1. Automate all Imports

* The number of files in a project grows daily, and so do the places these files get imported in. Under an ES5/Angular 1.x environment normally all files are included inside an index.html file, however when working with a more complex environment which includes testing, coverage reports, and automated builds.
* Gulp has quite a few great plugins for dealing with such automated imports, [gulp-inject](https://www.npmjs.com/package/gulp-inject), [wiredep](https://github.com/taptapship/wiredep), [useref](https://www.npmjs.com/package/gulp-useref), and [angular-file-sort](https://github.com/klei/gulp-angular-filesort) being a few good examples; however initial configuration could be challenging.

### 2. Understand directory structure requirements

* A well thought out directory structure can go a long way in preventing issues that can occur in the era of precompiled languages, transpilers and multiple environment modes. Due to this we understand that:
  + Our source code is not the same code that's being served to the browser.
  + Our source code is served minified and concatenated in production, but this can be confusing for development.

### 3. Provide distinct development and production builds

* With the right directory structure, injection automation and build automation in place, we can use browser-sync, the tool recommended by the gulp team, to serve either a production or a development version of our app.

### 4. Inject files with gulp-inject and wiredep

* Between [gulp-inject](https://www.npmjs.com/package/gulp-inject) and [wiredep](https://www.npmjs.com/package/wiredep) there is no excuse to ever have to manually inject a file into index.html. gulp-inject does a great job at selecting all your java script and css files.
* While wiredep assists with selecting the right files from any installed bower packages.

### 5. Create production builds with gulp-useref

* gulp-useref is a hidden gem within the gulp ecosystem. It's a wonderful plugin which reads all the files included in a html file, then concatenates and minifies them, returning and including a single file.

### 6. Separate Gulp tasks into multiple files

* As the project grows, so does the number of gulp tasks. This can leave you with a very large and messy gulpfile.js.
* A usefull trick is to split the file into multiple files inside a directory, then use require-dir to require all the task files within gulpfile.js.
  + 1. **Role and profile**
* There are several key features regarding the use of Gulp which would make you want to use it. The one I hold as the most important is the way it can **simulate** the server environment where you will ultimately be hosting your code.
* This includes moving files around your project directory, and more importantly placing them in a **development directory** where you will be running a web server. Gulp also enables you to **compile**, **minify** and **concatenate** any files you want.

**2.3: KOA.JS and MONGO DB**

* + 1. **Objectives of the work undertaken**
* Koa is a new web framework designed by the team behind Express, which aims to be a smaller, more expressive, and more robust foundation for web applications and APIs. By leveraging async functions, Koa allows you to ditch callbacks and greatly increase error-handling
* MongoDB describe themselves as a general purpose database but IMO the sweet spot for the DB is in the development of modern applications with rapidly changing requirements utilizing modern programming stacks.
* MongoDB is relatively easy to setup up, and scaling is relatively straight forward. Administration of MongoDB in production is less expensive than for many other databases.
  + 1. **Scope of the Work**
* Since, MongoDB is a NoSQL database, so we need to understand when and why we need to use this type of database in the real-life applications. Since in normal circumstances, MongoDB always preferred by the developers or project managers when our main concern is the deal with large volume of data with a high performance.
* If we want to insert thousands of records in a second, then MongoDB is the best choice for that. Also, horizontal scaling (adding new columns) is not so easy process in any RDBMS systems. But in case of MongoDB, it is very much easy since it is a schema less database.
* A Koa application is an object containing an array of middleware functions which are composed and executed in a stack-like manner upon request. Koa is similar to many other middleware systems that you may have encountered such as Ruby's Rack. This includes methods for common tasks like content-negotiation, cache freshness, proxy support, and redirection among others

**2.3.3 Importance and Applicability**

1. **Store large volumes of data that often have little to no structure**:

* Relational databases store structured data like a phonebook. But for growing, unstructured data. A NoSQL database sets no limits, and allows you to add different types of data as your needs change.
* Because MongoDB is flexible and document-based, you can store these [JSON](https://www.upwork.com/hiring/development/what-is-json/)-like binary data points (referred to as *BSON*) in one place without having to define what “types” of data those are in advance.

1. **Make the most of cloud computing and storage**:

* Cloud-based storage is an excellent cost-saving solution, but requires data to be easily spread across multiple servers to scale up.
* MongoDB can load a high volume of data and give you lots of flexibility and availability in a cloud-based environment, with built-in sharing solutions.

1. **Develop and release quickly:**

* If you’re developing within two-week Agile sprints, cranking out quick iterations, or needing to make frequent updates to the data structure without a lot of downtime between versions, modifying a relational database will slow you down.
* With MongoDB’s dynamic schemas, you can try new things, and fast. Your data doesn’t need to be prepped ahead of time, and your team can incorporate anything new, quickly, and at a lower cost.

1. **Scale database architecture efficiently and inexpensively:**

* With MongoDB, it’s easy to spread data out across commodity hardware on-site or in the cloud without needing additional software
  + 1. **Role and profile**
* The MongoDB Database DBA/developer will work closely with the development teams to implement new features and enhancements on a new application as well as maintenance and support of existing applications in production.
* The person hired for this role should have strong experience developing against a MongoDB database. This experience should include a deep understanding of proper queries for NoSQL databases.
* However, the[**JavaScript developers**](http://www.mindfiresolutions.com/Offshore-Javascript-AJAX-Development-India.htm) must install Node.js and node package manager (npm) on their systems to leverage the features and tools provided by Koa.js.
* But they must use the frameworks to build applications rapidly and keep the applications relevant in the long run.

**3.** **Brief description of the work done**

* 1. **Position of Internship and roles**
* Compile and analyze data, processes, and codes to troubleshoot problems and identify areas for improvement. Collaborating with the front-end developers and other team members to establish objectives and design more functional, cohesive codes to enhance the user experience.
* Developing ideas for new programs, products, or features by monitoring industry developments and trends. Recording data and reporting it to proper parties, such as clients or leadership.
  1. **Activities/ equipment handled**

**MongoDB:** MongoDB is a document database with the scalability and flexibility that you want with the querying and indexing that we need.

**Gulp.JS:** Gulp is a toolkit for automating painful or time-consuming tasks in your development workflow, so you can stop messing around and build something.

**Koa.JS:** Koa.js is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications. It is an open source framework developed and maintained by the creators of Express.js, the most popular node web framework.

**Node.JS:** Node.js is a platform built on [Chrome's JavaScript runtime](https://code.google.com/p/v8/) for easily building fast and scalable network applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient.

**Bootstrap:** Bootstrap is an open-source Javascript framework developed by the team at Twitter. It is a combination of HTML, CSS, and Javascript code designed to help build user interface components. Bootstrap was also programmed to support both HTML5 and CSS3.

* 1. **Challenges faced and how those were tackled**

1. **Scalability:**

* Scalability is a difficult thing for web developers to manage. Scalability is load balancing between the servers, hence when the load increases (i.e. more traffic on the page) additional servers can be added to balance it.
* All the load should not be thrown on a single server instead of which the software should be designed in such a way that it can work on multiple servers.

**2. UI/UX:**

* In the era of smartphones, web developers are expected to develop UI/UX that is responsive and user-friendly. If the web applications frustrate users, then it is difficult to maintain the customer’s loyalty on your website.
* Website navigation is another part often neglected by developers. Intuitive navigation creates a better user experience for the website visitor.

**3. Performance:**

* Slow web applications are a failure and as a result, customers abscond your website, thus damaging your revenue as well as reputation.
* Some of the performance issues developers’ faces are Poorly written code, Un-Optimized Databases, Unmanaged Growth of data.

**4. Knowledge of Framework & Platforms:**

* Frameworks are boost performance, offer libraries of coding and extend capabilities, so developers need not do hand-coding web applications from the ground up.
* Frameworks offer features like models, APIs, snippets of code and other elements to develop dynamic web applications.
  1. **Learning outcomes**
* The backend usually consists of three parts: a server, an application, and a database. If you book a flight or buy concert tickets, you usually open a website and interact with the frontend.
* Once you’ve entered that information, the application stores it in a database that was created on a server. For sake of ease, just think about a database as a giant Excel spreadsheet on your computer, but your computer (server) is stored somewhere in Arizona.
* All of that information stays on the server so when you log back into the application to print your tickets, all of the information is still there in your account.

**CONCLUSION**

* The Bootstrap framework provides an easy way of crafting website designs and allows creating greatly optimized mobile designs. It efficiently scales up the website and applications with a single code base.
* Bootstrap grid system builds mobile first, responsive websites, for small screens and then scales up the designs from there for the devices with larger display screen.
* By preferring code over configuration, node best practices, and a minimal API surface - gulp makes things simple like never before. Using the power of node streams, gulp gives you fast build that don't write intermediary files to disk.
* Node is a JavaScript environment has some great features that make it an attractive choice for building server-side application middle tiers, including web servers and web services for platform APIs.
* MongoDB stores data in flexible, JSON-like documents, meaning fields can vary from document to document and data structure can be changed over time. The document model maps to the objects in your application code, making data easy.
* Koa.js is a widely used Node.js web application framework. Koa.js is developed and maintained by the creators of another widely used Node.js framework — Express.js.

**REFERENCES**

1. www.gulpjs.com
2. [www.google.com](http://www.google.com)
3. [www.koajs.com](http://www.koajs.com)
4. [www.w3schools.com](http://www.w3schools.com)
5. [www.wikipedia.com](http://www.wikipedia.com)
6. [www.bootstrap.com](http://www.bootstrap.com)
7. [www.github.com](http://www.github.com)