# Abstract

* Summary of the whole report

# Introduction

* Introduction to project

# Literature Review

## Review of Domain

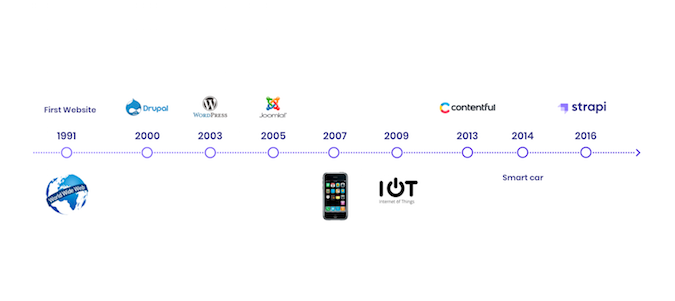
### Content Management System (CMS)

A content management system (CMS) is a software that helps users create, manage, and modify content on a website without the need for specialized technical knowledge.

In the other words, it is a program that makes it possible to create and modify of digital content for the website without having knowledge about how to make it.

A CMS is an important part of a dynamic website.

#### History



According to (Burgy, 2020), “*the history of the CMS traces back to the first website in history, by Tim Berners-Lee in 1990, which was modeled on an internet-based hypertext system HTML, which represented just text and links.”*

At the beginning, a website is only a set of static sites that served content without the need for a back-end database. They consumed very little computing resources, so they loaded quickly. When a data changes, administrators have to change the HTML content of the corresponding site.

In 1990s, as the popularity of the world wide web grows and websites increase the need for frequent updates, the need of the transition from static web to dynamic web becomes necessary. This led to the introduction of a plethora of CMS products from FileNet, StoryBuilder from Vignette, Documentum, and many others. These were all proprietary, closed source products, which was popular at that time period.

However, in the early 2000s, open-source CMS alternatives emerged, including WordPress, Drupal, and Joomla. They could be used to build websites without requiring users to have knowledge of HTML and CSS.

Today, about 33 percent of websites are built using these first-generation content management systems. These traditional CMS are monolithic systems that include the back-end user interface, plugins, front-end templates, Cascading Style Sheets (CSS), a web server, and a database. With every user request for a website page, a server first queries a database, then combines the result with data from the page's markup and plugins to generate an HTML document in the browser.

As we move deeper and deeper into the first decade of the 2000s, early mobile devices like Palm and Blackberry provide access to web content, then the introduction of smartphones and tablets around 2010 brings more and more users to the web via mobile devices. In 2016, the scales tip and web access from mobile devices and tablets exceeds desktops worldwide.

The monolithic CMS wasn't suited to serving content to these different types of access devices, which necessitated different versions of websites—usually stripped-down versions of the website for mobile users.

In the future, with the proliferation of websites as well as mobile devices, the need for multi-channel content delivery became clear.

#### Internal school information system.

An internal school information system is a CMS, where data is dynamic and can be managed with the help of an application.

The system contains many features for both students and staffs in schools. They are:

* Forum.
* Online homework.
* Teamwork management.
* Academic portal.

##### Forum

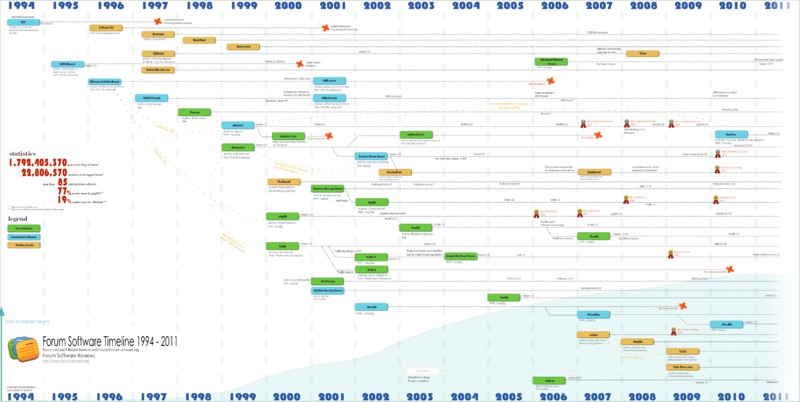
A forum is a place for Internet users to exchange data, discuss and chat with each other.

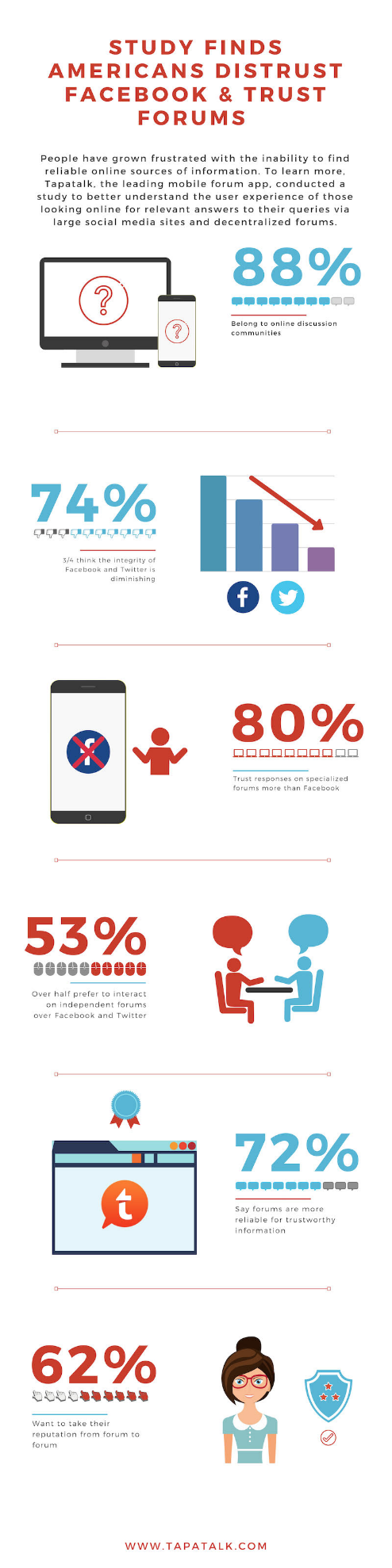
According to (Lee, 2012), in 1990s, Internet forums is introduced and began to take shape but it is based on the Bullet Board System (BBS). Functioning, appropriate to their name, like a virtual bulletin board, users could post and pin information they wanted to contribute, which would then be visible to anyone who accessed the server. The BBS possessed such features as public message boards, direct messaging capabilities among users, and interactive games.

The next development in collaborative communication came in 1988 with Internet Relay Chat (IRC) which allowed for live chatting between users, and eventually file sharing. Users could connect and join multiple chatrooms, known as channels. These early methods for organizing information through pins, posts, threads, and channels became central to the way that internet forums would function for collaborative fan activity. Internet forums function on a much larger scale than their early predecessors, acting as centralized locations for discussion, specializing in any topic imaginable.

Their official beginning came about in 1994, when the W3 Consortium developed WIT as the first software dedicated to forum protocol. The fan writing community embraced this new digital age and began to find space on the internet to post, share, comment, and collaborate on their writing. Internet forums such as fanfiction.net and Archive of Our Own (AO3) allowed users to browse by topic (or, rather, by fandom) to discover fanfiction in their chosen area. From there, they could engage with the work through likes/kudos to show their approval, or add notes and comments to allow for direct interaction with the author. The fanfiction, organized by fandom, allowed the ability to search by content (particular characters, relationships, tropes, etc). AO3 represented a space created by fans for fans, with the Organization for Transformative Works responsible for its creation itself a fan created space. The OTW embraced the internet and how it could be used to “fannish activities as accessible as possible to all those who wish to participate”.

While these sites allowed for engagement and interaction through the form of likes and comments, other forums encouraged purely collaborative writing. Websites such as proboards allowed users to create their own site built around a particular topic and create sections within the site. These sections could be different locations in the storyworld, and within the sections individual channels, or boards, could be made where discussions could be held. For example, fans of Harry Potter may set up a Proboard with the section Hogwarts, and within that section have boards for the Great Hall, the Quidditch Pitch, Classrooms, Dormitories, etc. Finally, on these channels, a user could begin a post that could turn into a thread. One user would post their portion of a story featuring their characters, another would respond with their characters, then a third, and so on. This mode of text-based roleplaying, accessible now through internet message boards, mirrors not only the writing of fanfiction about a particular storyworld, but also fosters a sense of community that has always been central to fandom. Proboards, still in use today, constantly innovates and updates its software “through the marriage of Web 2.0 technology and all the functionality of traditional message boards” to provide the best experience to their users, and “promises to define the capabilities of next-generation message boards” (proboards.com). The site also makes use of Cloud-based hosting and utilizes “distributed load balancing and memory caching” to provide speed and unlimited bandwidth to users (proboards.com).



Nowadays, with the strong development of social media (like Facebook, Twitter, etc.), forums now are less beneficial. However, in an anonymous survey of 1,000 Americans who use social media, (Spillane, 2019) said that:

We found that many people are frustrated with sites like Facebook and Twitter, and interestingly, today’s users are turning to specialized online forums for news and information, rather than mainstream social media.

In fact, 72% of respondents felt that forums, centralized or independent, were more reliable for trustworthy information, compared to the meager 18% who felt Facebook would be more reliable. The survey also found that only 9% of respondents felt Twitter was reliable for trustworthy information.

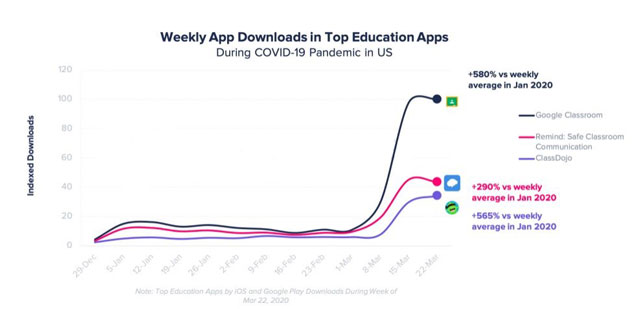
In the future, with the development and widespread use of mobile devices as well as the widespread coverage of the Internet, forums will be developed into mobile applications for users to easily use. However, currently there is no forum which has developed into a mobile application because it is not profitable (from advertising, recharge, ...).

##### Online assignment

Online homework is a feature, where teachers (or tutors) create an assignment and students submits their answers for comment(s) and mark.

In the past, teachers often ask their students to finish the remain exercises of a subject at the end of the class. Students do these exercises at home, and at the next class, their teachers will review, leave a comment, fix problems and give them a mark.

This will be a traditional method that is applied continuously for students to review what they have learned until the COVID-19 epidemic spreads and they are not able to learn directly in the classroom. Alternatives to face-to-face classes are used (Like google classroom), which is act like a directly homework assignment. In addition, teachers can view the overall status of the class (How many students handed in and assigned), and view the result of each student.

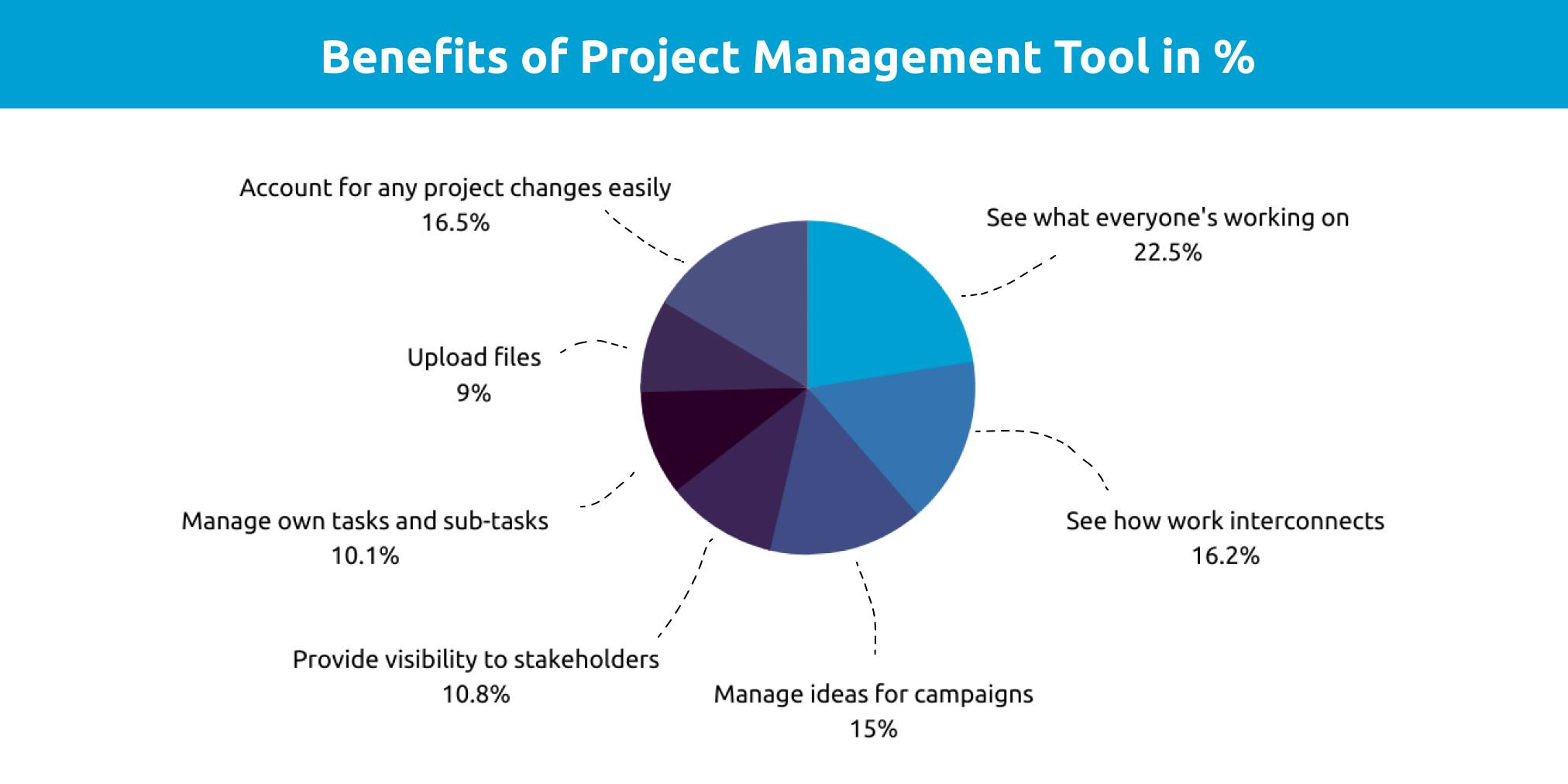


In the future, because of the usefulness of this feature, traditional classes will continue but homework will be delivered online for best management.

##### Teamwork management

Teamwork management is very important to finish a big problem, which requires to work on team. In this feature, the problem will be divided into multiple tasks (and they can also be divided into multiple subtasks) and assign to multiple people who join the team. In each task, there are some statuses for them (on-going, finish, overdue, …) to display the status of the task.

There are multiple online tools for task management, like: Trello, nulab, DragApp, etc.



This can increase the effectiveness of teamwork because the task is divided into subtasks and assigned to multiple teammates, which make the task easier to complete. In addition, team leader can view the overall progress of each subtask and evaluate the performance of the teamwork to make appropriate and timely adjustments.

##### Academic portal

In the past, student had to make a copy of their weekly schedule by their hand. In addition, when there is an upcoming event, the teacher must notify the students in advance to prepare and attend on time.

Nowadays, with the development of technology, students' weekly timetables will be uploaded via social networking groups or websites in the form of documents or class schedules. Students can view upcoming events and schedules easily.

With the advantages of this, in the futures, the timetable will be used online instead of traditional handwritten copies.

## Review of Technologies

To successfully create an internal school information system, suitable technologies are required. In this section, a comparison between front-end, back-end technologies will be shown

### Review related technologies

#### Front-end

There are multiple frameworks to develop a website. The most common frameworks are React JS, Vue js, Angular and Jquery.

##### React JS

React JS (React or React.js) is an open-source framework for JavaScript for building user interfaces (UI) and their components, part of the overall React Native framework for building mobile applications. React is maintained by Facebook and the wider community and is currently the most popular front-end framework in the past year, experiencing rapid growth due to its ease of use and flexibility. Some notable React projects include Facebook, Skype, Shopify, Discord, Instagram, Uber, Netflix, Tesla, Walmart and Airbnb.

##### Angular

Angular is an open-source web application framework based on TypeScript and maintained by the Angular Team at Google. Angular features a large ecosystem of tools and solutions contributed by a wide user base. Angular is ideally suited for highly customized enterprise-level web applications and is currently the second most popular front-end framework, according to GitHub.

Examples of the Angular framework include PayPal, Upwork, Google, and Nike.

##### Vue JS

Vue JS (Vue.js) is another JavaScript-based framework, but with greater flexibility around the use of HTML and CSS and the model-view-viewmodel (MVVM) architecture. Vue is easy to learn and has a very large community of support.

Vue is currently climbing in popularity and is used by brands including Trustpilot, Nintendo, and Behance.

##### jQuery and Bootstrap

jQuery began as a JavaScript library, and is still marketed as such, but meets the criteria of a framework. jQuery as a library is used by 18.6% of all websites and is both fast and concise. jQuery is used by major brands including WordPress, Facebook, IBM and Google.

Bootstrap is one of the most popular frameworks for creating mobile, responsive web content, currently the most popular CSS framework on the market, used by over 19.6% of all websites. Bootstrap is popular for its templates, which speed development, although some critics say this creates a more “uniform” look to the final websites.

Popular examples of Bootstrap include Lee, Fox News, Reuters, and NetGuru.

#### Back-end framework

Back-end is the most essential part of a website. Choose the suitable back-end framework can reduce the time and cost of a website development.

There are different back-end frameworks which have their own advantages and disadvantages. They are: ASP.Net (C#), Spring MVC (Java), Laravel (PHP) and Yii2 (PHP).

##### ASP.Net

ASP.NET is a web development platform, which provides a programming model, a comprehensive software infrastructure and various services required to build up robust web applications for PC, as well as mobile devices.

ASP.NET works on top of the HTTP protocol, and uses the HTTP commands and policies to set a browser-to-server bilateral communication and cooperation.

ASP.NET is a part of Microsoft .Net platform. ASP.NET applications are compiled codes, written using the extensible and reusable components or objects present in .Net framework. These codes can use the entire hierarchy of classes in .Net framework.

##### Spring MVC

A Spring MVC is a Java framework which is used to build web applications. It follows the Model-View-Controller design pattern. It implements all the basic features of a core spring framework like Inversion of Control, Dependency Injection.

A Spring MVC provides an elegant solution to use MVC in spring framework by the help of DispatcherServlet. Here, DispatcherServlet is a class that receives the incoming request and maps it to the right resource such as controllers, models, and views.

##### Laravel

Laravel is an open-source PHP framework, which is robust and easy to understand. It follows a model-view-controller design pattern. Laravel reuses the existing components of different frameworks which helps in creating a web application. The web application thus designed is more structured and pragmatic.

Laravel offers a rich set of functionalities which incorporates the basic features of PHP frameworks like CodeIgniter, Yii and other programming languages like Ruby on Rails. It has a very rich set of features which will boost the speed of web development. In addition, Laravel can save a lot time to develop a website from scratch. Moreover, a website built in Laravel is secure and prevents several web attacks.

##### Yii2

Yii2 framework is an open-source PHP framework for rapidly-developing, modern Web applications. It is built around the Model-View-Controller composite pattern. Yii2 provides secure and professional features to create robust projects rapidly.

Yii2 framework has a component-based architecture and a full solid caching support. Therefore, it is suitable for building all kinds of Web applications: forums, portals, content managements systems, RESTful services, e-commerce websites, etc.

Yii2 follows OOP (Object-Oriented Programming) framework. Hence, it requires a basic knowledge of OOP. The Yii framework also uses the latest features of PHP, like traits and namespaces. The major requirements for Yii2 are PHP 5.4+ and a web server.

### Comparison

After introduce about the technologies, the comparisons between them are required to defined the most suitable one for the project.

The comparison of them is shown in the table below:

#### Front-end

|  |  |  |
| --- | --- | --- |
|  | Advantages | Disadvantages |
| React JS | + Reusable code  + Easy to learn and use  + High performance | + Poor documentation  + Only view part support  + Complex JSX |
| Angular | + Two-way data binding  + The MVVM architecture  + Faster application prototyping  + Improved server performance | + Steep learning curve  + Limited SEO options  + Underperform if used to build complex SPAs  + CLI documentation is lacking details |
| Vue JS | + User-friendly and easy to learn  + Highly scalable and versatile  + Easily customizable  + Well-documented | + Lack of plugins  + A small community of developers  + Language barriers  + Limited support for large-scale project. |
| JQuery | + Flexible DOM  + Simple to use  + Open-source High performance | + App are comparatively slower than native  + Slower than CSS in many cases  + Page transitions and animations may not be as seamless as native apps |

#### Back-end

|  |  |  |
| --- | --- | --- |
|  | Advantages | Disadvantages |
| Laravel | + Simple coding: Laravel makes the task that much more straightforward. That’s because the software has an extensive library of pre-programmed functionalities, which minimizes the amount of coding required.  + Scalability: Project sizes differ. Highly scalable software allows you to tackle projects of any size, depending on your needs. Laravel is one such scalable framework that makes it easy for you to use for small and medium-sized web applications.  + Secure: The software has a safe, built-in access control system. Plus, it provides a robust mechanism that allows you to handle any bugs or issues with ease. In addition, the framework will enable you to store passwords in an encrypted format instead of text, so you get that extra layer of protection.  + Easy data migration: Data migration is a lot simpler if you use Laravel.  + Easy to learn: It is based on PHP as well as it has strong community all over the world. | + Lightweight framework: Laravel does have a comprehensive list of built-in features designed to make web development easier.  + High cost: Laravel isn’t the cheapest web framework you’ll come across.  + Limited support: Unfortunately, support for Laravel isn’t easily within reach.  + Frequent updates: Laravel has regular updates, which is excellent. But the downside to this is that older versions of the product quickly become buggy.  + Slow: Laravel isn’t the fastest web framework out there to work with as comparing it to other frameworks |
| Spring MVC | + Lightweight: Spring allows you to create enterprise level applications through a Java method known as Plain Old Java Object (POJO). In POJO, you are not required to use an application server.  + Configurations: Spring follows a separate way of configuration including for application logic.  + Server Requirement: Spring framework does not require you to use neither an application or web server.  + Reinvent Requirement: Some technologies like JDK timers, ORM and lagging frameworks are already included with Spring. Hence, you are not required to learn separately or define codes for those technologies.  + Classes & Packages: Another important benefit of Spring framework is the modules that it offers. Developers have lots of options in terms of classes and packages. | + Learning Curve: Spring is not user friendly when it comes to new developers. This is because Spring is incorporated with entirely new programming approach.  + Security: Threats such as Cross bite scripting and XSS are very common in applications. However, Spring does not provide sufficient guidelines to deal with these threats. Without proper security measures, hackers can easily infiltrate the apps.  + Complexity: Spring framework consists of large number of variables. Therefore, the development team must be only be included with an experienced panel who has high skills and expertise with this framework.  + Parallel Mechanisms: Wide range of applications provided by the Spring framework may create confusion among developers. The responsibility lies in the developer to the choose the right option. Otherwise, there can be some significant delays.  + XML Requirement: Developers must understand that working with Spring requires enormous amount of XML. Due to this large amount, they must be ready to allocate separate time for coding. |
| ASP.Net | + Open source: The core of ASP.NET is open source based. This makes it flawless the development process. If there is any bug or error, developers can clearly identify in this stage.  + Simplicity: Any tasks including complicated ones can be performed easily using ASP.NET.  + MVC Architecture: A framework following the Model View Controller (MVC) architecture contains interconnected ports which allows to divide concerns.  + Development Time: The development time is much lesser when it comes to ASP.NET.  + Flexibility: ASP.NET provides various flexibilities in terms of languages. It is language independent.  + Integrated Toolbox: Visual Studio Environment provides ASP.NET with a high-class toolbox. This toolbox comes with various features like automatic deployment and server control. | + Cost: ASP.NET is not an economical solution. The main reason for this is the cost involved in some features.  + Security: Compared to other frameworks, ASP.NET lacks certain security features. Some of those features are, password reset, email account verification and username reminder.  + Documentation: The documentation on ASP.NET is not up to the level of expectation.  + Application Transferring: Since the configuration settings of an ASP.NET based web application is stored inside the IIS Metabase, you cannot access them easily.  + Core Functionality: The user report suggests that the core of ASP.NET is in raw condition.  + Customization: Developers must be careful while making changes to the app. Changes made to the current version of the program may not work in the new version. |
| Yii2 | + Open-source: Yii is an open-source platform and provides the developers with the fastest option for building large-scale web applications. It is seamless in nature and gives faster performance and the most suitable to create ecommerce frames, content and data management system and forums.  + MVC architecture: The developers have the ability to develop smart web applications because of the MVC(model-view-controller) architecture of this framework.  + Support tools: It offers a set of highly intuitive and extensible tools for speedily generating code through features such as CRUD and form inputs.  + Caching support: Yii also comes with clean, elegant caching support which enables a smart web developer to create highly optimized web applications – on the fly. Yii makes a wise use of rendering optimized web apps – to provide similar user experience across a range of memory-intensive smart devices.  + Database support: Yii is supportive of ActiveRecord for NoSQL and Relational databases and has query builder feature.  + Error handling: Yii comes with efficient error handling and logging systems – to be filtered, categorized and routed to diverse destinations  + Security: Yii is also accompanied with a wide variety of security measures which is capable of preventing your websites from attacks like SQL injection, cookie tampering, cross-site request forgery (CSRF) and cross-site scripting (CSS).  + Support: It has large community as well as allows third-party interfaces by merging different frameworks with it.  + Easy to learn: If a developer yearns for systematic learning, Yii is the right framework – as it has detailed documentation as well as tutorial for every single method. | + The Ajax features are not well-drafted and require the help of JavaScript whenever needed.  + It does not provide AR queries or building up of multiple relations. The relationship between Active Record are hardly to learn.  + The dependency on static methods is what make the developers to go against the framework.  + It requires previous knowledge on PHP programming language and other add-ons. This at times make it difficult for the novice programmers to master this. |

## Review of Methodologies

Website projects can fail for a number of reasons, but most commonly, because they didn't follow a standard methodology. A web development methodology is used to control the process of developing a website, where each phase builds upon the previous phase and each phase serves as an input to the following set of activities.

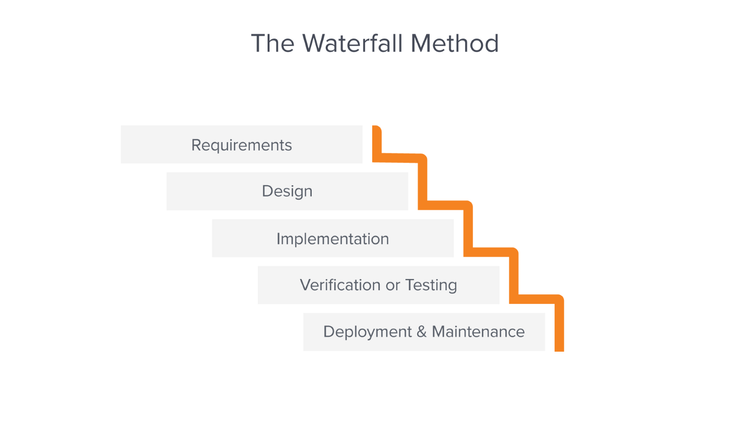
The most common methodologies for website development: Waterfall, Agile and Spiral.

### Review

#### Waterfall

The Waterfall methodology (also known as the Waterfall model) is a sequential development process that flows like a waterfall through all phases of a project (for example: analysis, design, development, and testing), with each phase completely wrapping up before the next phase begins.

The waterfall methodology is a linear project management approach, where stakeholder and customer requirements are gathered at the beginning of the project, and then a sequential project plan is created to accommodate those requirements. The waterfall model is so named because each phase of the project cascades into the next, following steadily down like a waterfall.

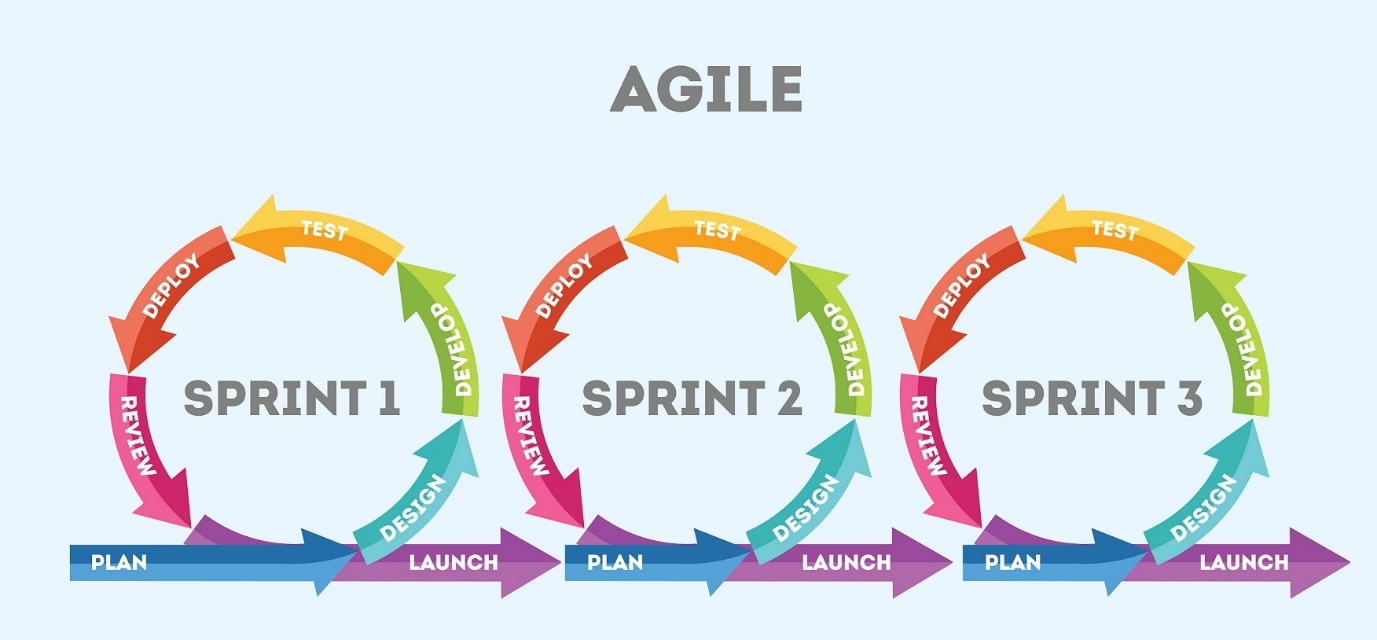


#### Agile

Agile is an approach to project management that centers around incremental and iterative steps to completing projects. The incremental parts of a project are carried out in short-term development cycles. The approach prioritizes quick delivery, adapting to change, and collaboration rather than top-down management and following a set plan.

In Agile processes, there is constant feedback, allowing team members to adjust to challenges as they arise and stakeholders an opportunity to communicate consistently. Though originally created for software development, the Agile approach is now widely used in executing many different types of projects and in running organizations.

Contrast this with more traditional forms of project management. Traditional project management generally progresses linearly through planning, designing, implementation, and closing stages. One stage must be completed before moving to the next one.

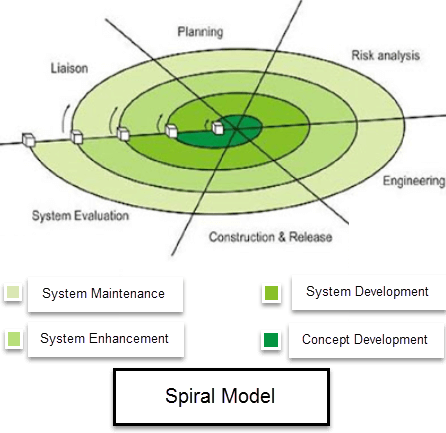


#### Spiral

Spiral Model is a risk-driven software development process model. It is a combination of waterfall model and iterative model. Spiral Model helps to adopt software development elements of multiple process models for the software project based on unique risk patterns ensuring efficient development process.

Each phase of spiral model in software engineering begins with a design goal and ends with the client reviewing the progress. The spiral model in software engineering was first mentioned by Barry Boehm in his 1986 paper.

The development process in Spiral model in SDLC, starts with a small set of requirements and goes through each development phase for those set of requirements. The software engineering team adds functionality for the additional requirement in every-increasing spirals until the application is ready for the production phase.



### Comparison

Each method has their own advantages and disadvantages. Base on the requirements, we can choose the most suitable method for project development.

The table below show the advantages and disadvantages of each method:

|  |  |  |
| --- | --- | --- |
|  | Advantages | Disadvantages |
| Waterfall | + Before the next phase of development, each phase must be completed.  + Suited for smaller projects where requirements are well defined.  + They should perform quality assurance test (Verification and Validation) before completing each stage.  + Elaborate documentation is done at every phase of the software’s development cycle.  + Project is completely dependent on project team with minimum client intervention.  + Any changes in software are made during the process of the development. | + Error can only be fixed during the phase.  + It is not desirable for complex project where requirement changes frequently.  + Testing period comes quite late in the developmental process.  + Documentation occupies a lot of time of developers and testers.  + Clients valuable feedback cannot be included with ongoing development phase.  + Small changes or errors that arise in the completed software may cause a lot of problems |
| Agile | + Customer satisfaction is rapid, continuous development and delivery of useful software.  + Customer, Developer, and Product Owner interact regularly to emphasize rather than processes and tools.  + Product is developed fast and frequently delivered (weeks rather than months.)  + A face-to-face conversation is the best form of communication.  + It continuously gave attention to technical excellence and good design.  + Daily and close cooperation between business people and developers.  + Regular adaptation to changing circumstances.  + Even late changes in requirements are welcomed. | + It is not useful for small development projects.  + There is a lack of intensity on necessary designing and documentation.  + It requires an expert project member to take crucial decisions in the meeting.  + Cost of Agile development methodology is slightly more as compared to other development methodology.  + The project can quickly go out off track if the project manager is not clear about requirements and what outcome he/she wants. |
| Spiral | + Software is produced early in the software life cycle.  + Risk handling is one of important advantages of the Spiral model, it is best development model to follow due to the risk analysis and risk handling at every phase.  + Flexibility in requirements. In this model, we can easily change requirements at later phases and can be incorporated accurately. Also, additional Functionality can be added at a later date.  + It is good for large and complex projects.  + It is good for customer satisfaction. We can involve customers in the development of products at early phase of the software development. Also, software is produced early in the software life cycle.  + Strong approval and documentation control.  + It is suitable for high-risk projects, where business needs may be unstable. A highly customized product can be developed using this. | +It is not suitable for small projects as it is expensive.  + It is much more complex than other SDLC models. Its process is complex.  + Too much dependable on Risk Analysis and requires highly specific expertise.  + Difficulty in time management. As the number of phases is unknown at the start of the project, so time estimation is very difficult.  + Spiral may go on indefinitely.  + End of the project may not be known early.  + It is not suitable for low-risk projects.  + May be hard to define objective, verifiable milestones. Large numbers of intermediate stages require excessive documentation. |

Comparison between each method:

## Choosing solutions

Base on the comparison in the previous section, I have chosen the following technologies for the project:

+ Front-end technology: I have chosen jQuery technology because it is a fast, feature-rich and lightweight JavaScript library. I can easily use JavaScript on the website as well as it can save time and cost (because it is open-source with strong community). In addition, bootstrap is required to create responsive website for multiple devices.

+ Back-end technology: I have chosen Yii2 framework for the back-end development process. It can be installed easily through composer with command and an application which is developed based on Yii2 framework is secured. Especially, it is based on PHP, which is the most popular programming languages in the worlds with strong community. In addition, it reduces the development time with the helps of Gii, which allow developers to generate Model, Controller, Form, CRUD controllers, actions and views, etc. Finally, Yii2 framework is a suitable suggestion for forum.

+ Methodology: Because the project is not large as well as require high-risk level enough and the requirements is mostly clear, the Waterfall method is suitable for the project development.

# Requirement Analysis

## Similar application 1

## Similar application 2

## Conclusion

* Confirm your application features
* What is your improvements / modification / localization / …?

# Software design

## Architecture (optional)

## GUI (optional)

## DB (optional)

## UML (optional)

# Software implementation

## Development environment

## Important technical problems & solutions

## Test (optional)

## Results

# Evaluation and conclusion

## Evaluation of results

* Pros / Cons of your application

## Conclusion

* Lessons learnt
* Problems / difficulties
* Future improvements
* Conclusion

# Appendix

* Final plan
* Screenshots (optional)