

Finalize Admin UI & Backend for a web application with PHP as Event Managing and Scheduling System

Case company: Company X

LAB University of Applied Sciences

Bachelor of Business Administration, Business Information Technology

2021

Quan Tran

Abstract

Author(s) Quan Tran	Publication type Thesis, UAS Number of pages 59	Completion year 2021
Title of the thesis Finalize Admin UI & Backend for a web application with PHP as Event Managing and Scheduling System Case company: Company X		
Degree Bachelor (UAS)		
Name, title and organisation of the client		
Abstract <p>This study is based on a course named Real-life IT Project. The object of the course was for students to gain experience in working on a technical project in a formal business environment, in addition to academic practice. The course included front-end and back-end programming, as well as the content management system (CMS), customer service, and collaboration. Company X was the project's partner/customer, and they requested for a solution that specialized in event scheduling data management. The solution anticipates a portal that can be used to store and deliver data from various companies' partners to end-users. The platform's specifications align with the course's curriculum; however, further study as well as programming skills and implementation strategy are required before the final product can be produced. As a result, the concept of doing a systematic report on the project is born. This thesis serves as a comprehensive report of the whole working process. This suggests that the study should provide readers with relevant facts and expertise, as well as a maintenance schedule for the case business in the future.</p>		
Keywords Content management, event scheduling data management, web application		

Contents

1	Introduction.....	4
1.1	Thesis background.....	4
1.2	Thesis objective and research questions.....	4
1.3	Research and data collecting method.....	5
1.3.1	Approach.....	5
1.3.2	Research method.....	6
1.3.3	Data collection method.....	7
1.4	Thesis structure.....	8
2	Concept of CMS, WordPress and SDLC.....	9
2.1	CMS.....	9
2.1.1	Content component definition.....	9
2.1.2	Content management application (CMA).....	10
2.1.3	Meta-content management application (MMA).....	12
2.1.4	Content delivery application (CDA).....	15
2.2	WordPress.....	15
2.2.1	Definition.....	15
2.2.2	Features and capabilities.....	16
2.2.3	Application in website management.....	19
2.3	SDLC.....	19
2.3.1	System Planning.....	20
2.3.2	System Analysis.....	22
2.3.3	System Design.....	24
2.3.4	System Implementation.....	25
3	Used tools and technologies.....	26
3.1	HTML.....	26
3.1.1	Definition.....	26
3.1.2	Elements.....	26
3.2	CSS.....	27
3.2.1	Compatibility.....	29
3.2.2	Core functions.....	29
3.3	PHP.....	30
3.3.1	Definition.....	31
3.3.2	Core component.....	31
3.4	JavaScript and JSON.....	35
3.5	Laravel Framework.....	35
3.6	MySQL database.....	37

3.7	Git.....	38
3.8	Virtual machine.....	39
3.9	Communication tools: Zoom & Email	40
4	Case company: Company X.....	41
4.1	Organization review.....	41
4.2	Current situation	41
4.3	The need for a new service	44
5	Practical implementation	45
5.1	System Planning	45
5.2	System Analysis	45
5.3	System Design	46
5.3.1	Data structure.....	46
5.3.2	Authorization and control	48
5.4	System implementation	49
5.4.1	Features available for Admin Users and SuperAdmin Users.....	49
5.4.2	Features available for SuperAdmin users.....	53
5.4.3	Features available for normal and anonymous users	55
5.4.4	Finalization	56
5.4.5	Future maintenance and support	56
6	Customer experience survey.....	57
6.1	Description	57
6.2	Result	57
7	Summary	61
	References	63

Appendices

Appendix 1. Survey for “User experience on product”

List of abbreviations

CDA: Content delivery

CMA: Content management

CMS: Content management system

CSS: Cascading Style Sheets

HTML: Hypertext markup language

HTTPS: Hypertext Transfer Protocol Secure

JSON: JavaScript Object Notation

MMA: Meta-content management

PHP: Hypertext Pre-processor

PIA: Privacy impact assessment

SDLC: System development life cycle

UI: User Interface

1 Introduction

1.1 Thesis background

The case for this thesis initially started as a case from a course named Real-life IT Project. The purpose of the course was for students to familiarize themselves with working in a technical project under professional business circumstances, besides academic practice. The course involved both front-end and back-end programming, the content management system (CMS), responding to customer's requirements, and teamwork. Company X was the partner/customer of the project, their demand is to find a solution that specifies in event scheduling data management. The solution expects a platform which can be used to store data from different company's partners and distribute it to end-users. The requirements of the platform match the course curriculum, moreover, it needs further research as well as communication and implementation plan to come to the final output. Therefore, the idea of putting the project into a comprehensive study is initiated. Consequently, this thesis serves as an in-depth report of the whole working process. This means that the thesis should deliver the right information and knowledge for readers as well as a future maintenance plan for the case company.

1.2 Thesis objective and research questions

The objective of the thesis is a means to relay the projected results and outcomes that come from the study. The objective reflects the author's intended achievement through the underlying research. The objective is assembled via a combination of the data collection and analysis process along with the initial research question as well as a hypothesis to make clear the goals and objectives the author has in mind for the study going forward. The objective of the research also has to align itself with the study's limitations, theoretical and conceptual framework to be adequately inclusive and transparent (Raymond 2012, 21).

The final purpose of this thesis is to summarize the result of the research process and the output of the application. The thesis should give the customers as well as readers knowledge about the project and for later maintenance, this thesis can also serve as a reference for maintainers to understand more about the application.

Research question:

What is the most effective way to finalize an existing code base and deploy it into production so that Company X can maximize the benefit of the project?

In order to answer the research question, it is necessary to break down this question into four sub-questions, which can be outlined as follow:

- What is this system?
- What are the benefits that the system can contribute to a company's business?
- What functions does it have and what can the functions do?
- What tools and technology should be used to finalize the project in the most effective way?
- When will the application come into production?

These sub-questions will help guide the research process and orientate the study into answering the original research question, which in turn will achieve the objective of the thesis.

1.3 Research and data collecting method

This sub-chapter introduces how the initial idea of the thesis was approached, as well as the methods used to research and collect data during the thesis process.

1.3.1 Approach

Inductive approach

Inductive approach is an approach taken mainly for theses that make use of a large body of raw data. The inductive process zooms out from a data pool and by doing so, identifies patterns and trends according to the collected data. The inductive approach is ideal for research that start from the data collection process, with no clear theory or model in mind. This approach is much more malleable and enables a more open research process, as it helps opens the door to more conclusions during the theory formation process. (Locke 2007, 880-885.)

Deductive approach

The deductive approach is a method of analysis that takes a general body of information and steadily zooms in to the particular. Deductive approach needs a concrete starting point, such as a theory or assumption and involves the use of various hypotheses and tests to fully revise said theory. This approach is best used if there is a theory to test and can be applied to revise a pre-established assumption. (Nola & Sankey 2007, 108-109.)

In the scope of this thesis, the inductive approach will be taken. This is due to the study portion of the thesis has been drawn out to conclude the collected raw data. The thesis aims to formulate a theory as to what Company X can achieve from the implementation of a pre-

existing codebase into the production process. It has been decided that the research should take its first steps from observing raw data, rather than an established theory or assumption, with the latter effectively narrowing the study's result in some avenues.

1.3.2 Research method

In this thesis, both qualitative and quantitative methodologies are in use. What follows is a brief explanation and justification of the two methodologies, as well as the overview of the data collection method that will be called into use for the paper.

Qualitative methodology

The qualitative methodology is a commonly used methodology to approach research and studies. This methodology hinges upon the extensive use of naturalistic empirical data that revolves around non-numerical and statistical data pools. The scope and inclusion of empirical evidence to be considered in a qualitative approach includes a variety of case studies and practical data collected from non-statistical sources such as interviews, observation, visual, writing and texts (Denzin & Lincoln 2018, 43). In brief, the qualitative methodology does not require the use of measurements and census data to support its analysis (Murray 2003, 1-3). The extent to which this methodology will be applied in the analysis of the case company and its ongoing project, as well as during the interview/survey portion of the research.

Quantitative methodology

The quantitative approach, on the other hand, employs a mixed variety of data analysis methods and tools, which can be used to analyse a raw data pool in order to make out trends and connections to pre-existing elements of the data, or amongst elements themselves. Data that is used in the quantitative method is almost certainly numerical and statistical raw inputs, which allows for a definitive and mathematical analysis of the problem at hand and to make out the trends and connections between the data point. This method is impervious to perception-based bias as well as other naturalistic factors that commonly plagues qualitative data. Quantitative is paramount for the paper, as this paper makes use of the inductive approach, which requires a sift through large bodies of raw data in order to make out the patterns and formulate a theory. (Murray 2003, 1-6.).

1.3.3 Data collection method

Data collection is the main means by which data is collected for the research in question. For this particular paper, the two main tools for data gathering and compilation will be benchmarking and interview/survey.

Benchmarking: Benchmarking is a means to compare and contrast a software or a system against other software and systems of its type and function in order to identify its rating and value as a digital product. The benchmarking process is often used to determine the limits of a digital product's capabilities and usefulness. (Pyzdek & Keller 2013, 131.)

Survey: A survey is used to generalize the results from a select group of sample individuals from a total population of the sample group and then applying this generalized result to the total population and derive value for this group (Pyzdek & Keller 2013, 113).

A modern survey process needs to take into account a number of key areas, which exist to ensure a clear and uncompromised survey result. The so-called four cornerstones of quality surveys coined by Dillman and his colleagues were created to exemplify and reflect this need for the accuracy.

- Coverage error: Coverage error, as its namesake suggest, means that a good survey needs to ensure full coverage of the total population which it needs to generalize upon. This is important, as the survey's result will better represent the surveyed population if all characteristics within said total population have a probability of being included in the survey.
- Sampling error: Sampling error is the issue of accurately representing a total sample population with only a fraction of said population being a part of the survey. This issue cannot be eliminated unless the entire population is surveyed, which is highly impractical. However, it may be mitigated should the sample size be big enough, proportionally to the total sample population.
- Nonresponse error: This is also very self-explanatory, as the higher the percentage of the sample group responds to the survey, the better and more accurate that survey will end up being.
- Measurement error: This issue varies from case to case. Measurement errors occur when participants to a survey provide a dishonest, or inaccurate answer to the questions, which contributes to the inaccurate implication of the final results. This issue can also occur during the data compilation and categorization process and can be summed up as human error. (Dillman et al 2014, 3-8.)

This goes to show that a properly done survey needs to take into account these four aspects if it is to guarantee an accurate and generalized result that can later be applied to the whole population being surveyed. It is, thus, according to these four cornerstones that this paper's survey will proceed, and the study will be designed so as to allow for as little errors as possible from all four of these areas.

1.4 Thesis structure

The thesis will follow the following pre-determined outline:

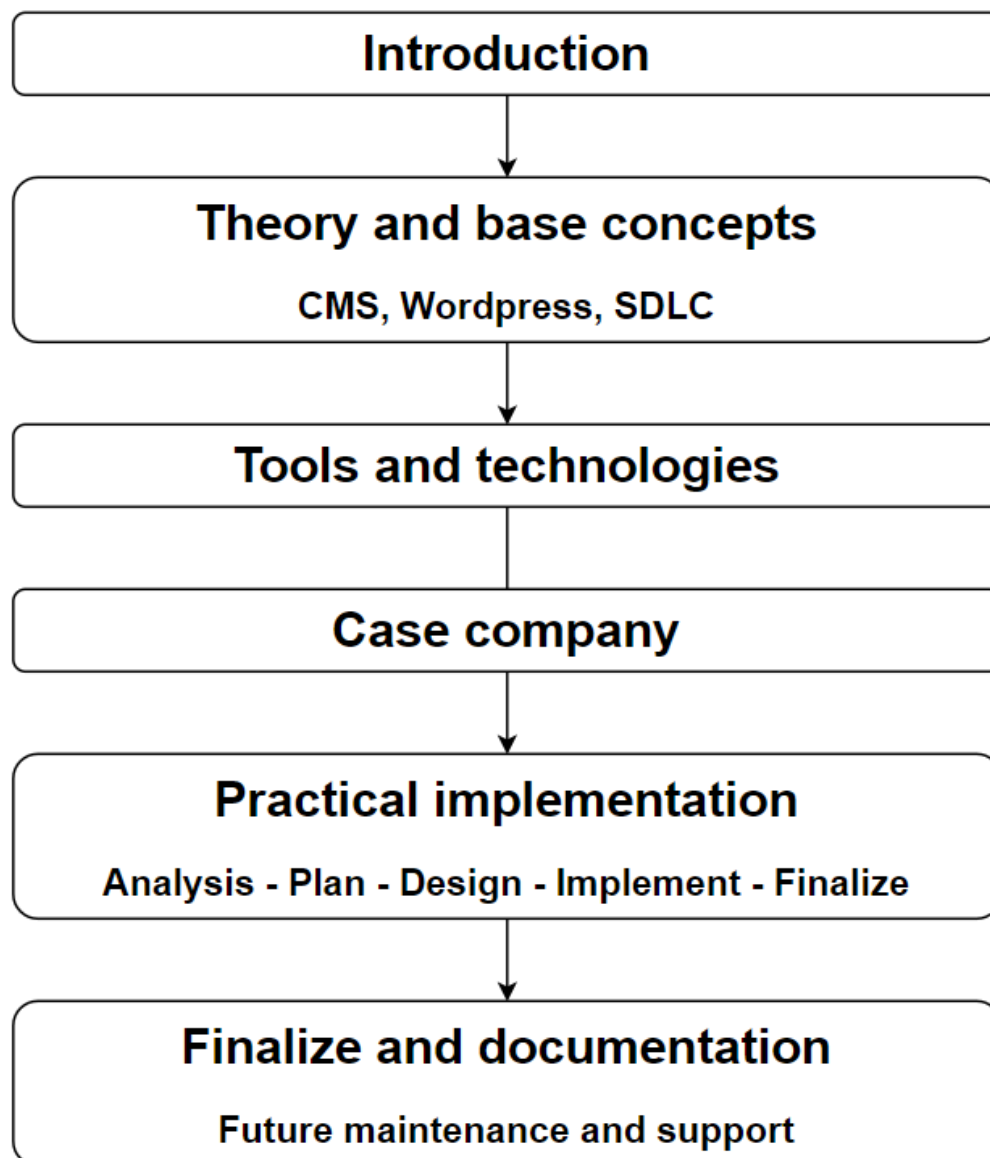


Figure 1. Thesis structure

2 Concept of CMS, WordPress and SDLC

2.1 CMS

CMS, or Content Management System, is a website administrative tool that allows fluent management of a website's content (Fraser 2002, 1). This, by itself is a quite self-explanatory concept. However, in order to fully understand to breadth of the scope of CMS, it is necessary to breakdown the core components of CMS into manageable trinkets of concepts, which will enable an easier in-depth look into the term as a whole. The most important aspect of CMS to define is the aspect of content itself. To truly understand CMS, one must ask self what content includes and what it leaves out. By looking at the three main components of CMS, namely content management (CMA), meta-content management (MMA) and content delivery (CDA) (Fraser 2002, 6-7), one can piece together a relatively complete picture of the full cycle of CMS. CMS together with its components are illustrated in Figure 2 below, and the details of each part are explained in the following paragraphs.

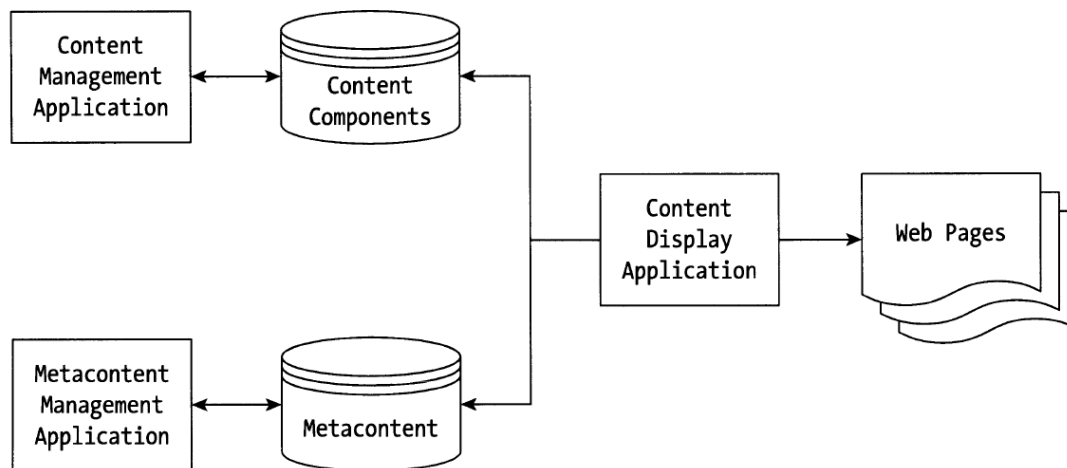


Figure 2. Basic CMS flow chart (Fraser 2002, 19).

2.1.1 Content component definition

Content can be categorized into two main forms, namely information and applications that run on the website itself (Fraser 2002, 5-6). The first form of content is quite easy to identify and recognize, as information is what websites aim to display, be it advertisements, texts, images, or other media that are usually found on a website. Meanwhile, applications can be the content that runs with the website itself and is not commonly associated with the term

content component at first glance. However, in retrospect, both information and applications are all that generally makes up the content component and thus will be taken into consideration for this study on CMS. Whether or not a CMS system contains both types of contents or specializes in a specific type of content will affect the system's overall workflow in the end.

2.1.2 Content management application (CMA)

CMAs merely manages the entire cycle of the contents found on a website. The content component will be influenced and managed by CMA from birth to the point of elimination. Through this management of content components, CMA will form a content repository, which may take the form of a database or set of files and anywhere in between. This is often described and implemented through a workflow chart. The main features of a CMA can vary considerably depending on the CMS in question. However, there are a few important main-stay components that are found in almost all CMA, regardless of type or price range. The core components in CMA are portrayed in Figure 3 below.

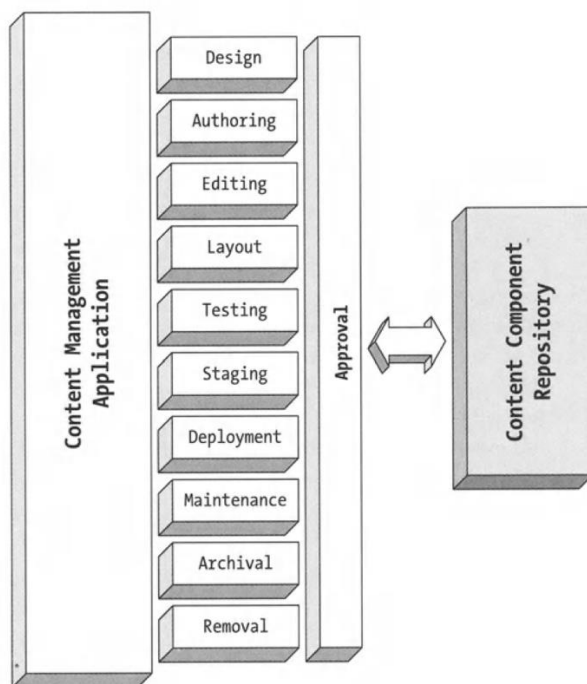


Figure 3. CMA workflow chart (Fraser 2002, 8).

Approval is the most decisive stage in the cycle of CMA. Before a stage can finish and the next can begin, it is almost required that a person of authority approve of the content components and any changes made to them during the previous stage.

Design is where every content component is determined and made out. The design process is continuous and in certain cases, can occur throughout the development process. This stage is most often reallocated to a third-party application or design tool that are acquired independently from the CMS itself.

Authoring is the act of procuring the necessary content components required for a site's operations. Authoring can run the line of either an original creation of content components, or a purchase of rights for contents from other authors or web pages for reupload. Some CMS systems can automate the authoring process, particularly with third-party contents and eliminating the need for the human factor.

Editing follows the authoring of a content and is the polishing factor of the content component. Most websites will require an editor to work hand-in-hand with the content author, which calls for the use of the CMS, as it helps smoothen the workflow and minimize the room for errors that may occur during the editing process via content tracking and content workflow.

Layout describes the distribution of the contents on the website, and it is usually embedded in the meta-content component from the MMA.

Testing is the next stage once the content component and its layout is completed. Website owners and administrators need to put the website through a few rounds of testing to weed out any unexpected errors and dead links once the website goes live. This is to ensure all website content components are in the best quality and minimizing any issues such as broken fonts, blurred images or bad links. Testing is simply done by following the links as well as hyperlinks on a separate browser to make sure all links end up where they are supposed to. Tests should also be done on various browsers as well as devices and operating systems, as not all are made alike. A website may work on Chrome but will break down on Opera.

Staging follows the testing phase and acts to minimize any effects the content production cycle may have on the site's active users. The content staging operates on a different sever from the main web site. This deters many website owners, as it requires the acquisition of a second server, thus driving up the operating costs.

Deployment of content from the staging server happens periodically and in small portions so as to ensure smooth operation on the main website's server. The deployment of contents

needs to be scheduled according to the website's traffic in order to ensure effective transition from the staging server to the actual site.

Maintenance is what occurs once a content component has successfully been deployed onto a server. This step is taken when the content on the website requires correction or updates. The content components should never be updated live, as it will pose problems for the server's operations. Content components in need of updates need to be put through the entire CM cycle again and staged on a staging server just like new content in order for a more orthodox and easier-to-trace process of content maintenance.

Archival occurs when contents are outdated and no longer needed for their intended use on the website. The archived contents will still be available for use and a robust archive of contents can be a great point to sell one's site on.

Removal is similar to archival and occurs when a content is no longer needed on the website. However, the difference is that the content will be removed from the site completely, rather than accessible in the site's archive search. This is usually applied for contents that violate regulations, or those that have infringed upon another source's copyright.

(Fraser 2002, 7-12.)

2.1.3 Meta-content management application (MMA)

Similar to CMA, MMAs are applications that manage and oversee the entire life cycle of meta-content components. Meta-content components are components that indicate how content is distributed and displayed on the website. Other than the different focus, MMAs basically share a similar workflow to that of a CMA, which can be observed in Figure 4 below.

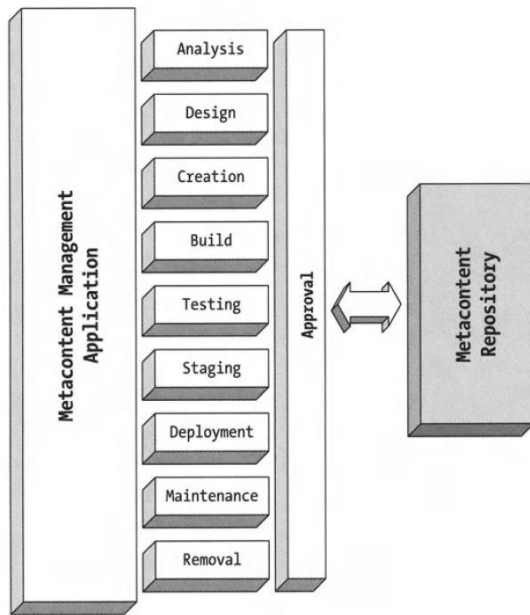


Figure 4. MMA workflow chart (Fraser 2002, 13).

Meta-content types:

The term meta-content takes under its umbrella a plethora of types and functionalities. The four main types of meta-content that will be included in the paper's discussion include:

- **Templates:** Templates are generally considered placeholder structures for content components, or other templates. Templates can be filled with content components or groups of content components on smaller templates to form a backbone for the website's appearance.
- **Scripts:** There are two main types of scripts for existing servers and websites, namely server-side scripts and client-side scripts. The four most prevalent scripting languages in use for client-side scripting are Netscape and JavaScript (Sun Microsystems and Microsoft variations), as well as VBScript. Client-side scripts, as its name suggests, operates independently from the server and can enable adjustments and alterations of the website on the user side without sending any input of these changes back to the server, as they utilize dynamic HTML. Server-side scripts, on the contrary, alters and inject logic into HTML on a server-wide scale, meaning that changes from the server-side will translate to changes on the client's side as well. The most prevalent scripting languages for server-side scripting include Active server pages, ColdFusion, Server-side JavaScript and PHP.
- **Programs:** Programs are similar to scripts. However, they are compiled prior to deployment and operation on the main server. This property makes program inherently

faster and smoother in operation, as well as providing a more varied set of functionalities.

- **Runtime dependencies:** checking content components' runtime dependencies is also a very important step to ensure a healthy server, in the sense that all hyperlinks and connections lead to where they are supposed to.

Approval in the MMA's life cycle, like CMAs is one of the most important stages to consider amidst other stages. However, unlike that of CMA, the MMA's approval process is made via a group or committee, especially when the change to the meta-content is significant. This difference is due to the fact that adjustments to the meta-content can greatly impact a website's overall appearance and feel.

Analysis is the beginning of every MMA cycle. Before any changes can be considered, a thorough and in-depth analysis of the business should be allowed to take place. The analysis should consider a few key areas and influences the changes are likely to have on the site's health. Some common aspects the analysis should encompass include predicted response to the change, predicted changes in response time, predicted effect on the site's user, predicted effect on the site's traffic. These analyses are not often part of the CMS and can be taken care of by a number of third-party software and analytic tools and service providers.

Design is another important aspect of the MMA cycle. Designing the meta-content can be done via third-party software and later imported into the CMS.

Creation of meta-content include the creation of all of the meta-content's components, such as templates, programs, scripts and runtime dependency. Meta-content creation is a very complex process and requires support from previous analytics and designs.

Building the meta-content follows the successful creation of a meta content. Some meta-content may require assembly to fully function, while others do not. This step is highly dependent on the meta-content type and functions.

Testing occurs after the meta-content has been created or built and involves a much more complex and extremely important testing process, unlike that of content components. Meta-content testing encompasses the same grounds as standard software development testing procedures, which involves inspection of unit, string, system and a release test.

Staging is the next step to take once the meta-content is ready to go live. Similar to content components, meta-content components require replication on a separate staging server. Staging can, like content components, be bypassed in favour of budget saving, as a separate server can drive the costs up, especially for server owners that do not have a lot

of way when it comes to budget. In cases like these, the meta-content can be produced directly on the main website, which may affect the performance of the website during its production, or even a temporary site shutdown.

Deployment is the process of moving the finished meta-content from the staging server to the main site. Unlike content components, meta-content components' deployment process will require the site to be temporarily shut down. This is why the meta-content deployment process is much more complicated and requires an eye to be kept on the server's overall traffic and the deployment usually takes place in low traffic hours, with the help of staging servers and quick installation platforms.

Maintenance is the upkeeping process of the meta-content life cycle. Just like in the case of content components. Meta-content components' maintenance needs to be viewed as a completely new meta-content development cycle. It is extremely problematic to apply fixes and adjustments directly onto a live, operating site.

Removal of meta-content is the process of doing away with a meta-content that is no longer useful. However, it is always a benefit to archive the obsolete code in the repository for future use, if needed.

(Fraser 2002, 13-17.)

2.1.4 Content delivery application (CDA)

The final cog to the CMS wheel, CDA, is the final piece to the puzzle, as it is the component that compile and links the content components via meta-content components from the common repository and displays them onto the website as the end product. CDAs will operate on the pre-existing databases and repositories created by CMAs and MMAs.

Data created by MMAs, or meta-contents, will be paramount to the functions of CDA. What and how information is displayed on the website is almost entirely dependent upon the meta-data inputs from the MMA cycle. (Fraser 2002, 18.)

2.2 WordPress

This sub-chapter presents the overall information about WordPress, which includes its definition, features and capabilities.

2.2.1 Definition

WordPress can currently be classified as a full-fledged CMS, whose definition and categorization have been discussed in section 2.1. Content Management Systems (CMS) and is

the driving force of almost a third of all websites and contents on the internet. WordPress started out as an open-source application meant to operate blog websites. This application can run independently from a personal computer and instead, bases itself on the web via a server. (WordPress.org 2021; Karol 2019, 9-10.). An example for WordPress is shown in Figure 5 below.

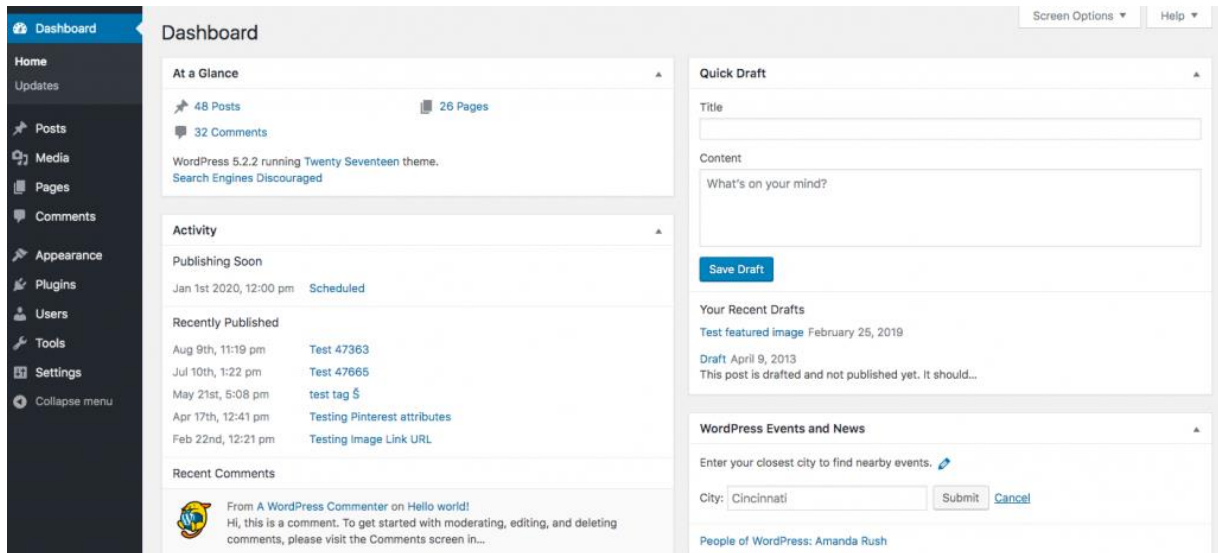


Figure 5. WordPress Dashboard (WordPress.org 2021)

2.2.2 Features and capabilities

The following findings about WordPress's features and capabilities are obtained from WordPress's official website (WordPress.org 2021) together with Karol Krol's study (2019, 25-26) on WordPress

WordPress's capabilities

WordPress is an open-source CMS, which operates entirely on another server, thus requiring no local presence in the system where the website is designed and operated and complies with World Wide Web Consortium, or W3C, standards. Its main selling points include an array of life-improvement feature for website developers, which can be summarized below.

Free-to-use: One of WordPress' greatest feature and perhaps its top selling point is that the application is a free-to-use app. WordPress' entire existence has hinged upon the concept of software freedom and thus allows for a greater pool of user to take advantage of its power as a CMS.

Simplicity: This is a rather self-explanatory feature, as WordPress is one of the easier to grasp and use CMS out there. It allows for a user-friendly interface and a mallow learning curve.

Flexibility: As mentioned above, WordPress prides itself on flexibility, as it can excel in almost any type of website, not just the three intended types that it claims to work best on. WordPress can be moulded to fit any intended use, from a blog, as its original intention was, to a full-fledged online community or government site.

Publish with ease: The publication process of a website is also much easier with the help of WordPress, as media can be inserted much like any other document creation software, which facilitates all level of web development and management skills.

Publishing tools: WordPress also eases the process of content management, as it has a wide array of publishing tools at its user's disposal.

User management: User management and user profiling is a very crucial part of any good CMS and WordPress offers this benefit of intensive and extensive user profiling, which allows for all contributors to enjoy the site's use individually, thus ensuring every interaction from every user group to be unique to that particular group of contributors.

Media management: Media management is another important area of focus for most CMS, as a properly done media management system will allow for easier uploads, update and removal. WordPress boasts some of the simplest and most user-friendly media management systems amongst most CMS of its calibre.

Full standards compliance: WordPress' full-standard compliance ensures that it runs on almost all of modern day's browsers, as well as the future generation of browsers.

Easy theme system: WordPress will have a default of three themes. However, it also holds a repository of themes, as well as the compatibility for custom themes uploaded directly from the user's end.

Extendable via plugins: WordPress is an ever-evolving CMS, with a dedicated plugin repository for extensions and alterations to its core features. Custom plugins can be uploaded, downloaded and deployed with relative ease, thus allowing for a much deeper user experience, despite its simple outwards appearance.

Built-in comments: WordPress also optimizes the interaction properties of its component websites, as well as enabling an easy-to-access comment system and facilitates discussion forums as well as discussion moderation in sites.

Search engine optimized: The application is optimized for search engines and comes with a wide host of SEO plugins as well as user generated plugins to assist in analytics and fine-tuned SEO data collection and processing.

Language availability: WordPress has been translated into over 70 languages and the platform is continuously developing in terms of language variety with the help of its vast community.

Easy installation and upgrade: WordPress is just a click away, literally and the installation as well as update process is extremely easy and simple, which is a bonus, as it is much more accessible for amateur web designers.

Importers: WordPress also provides importers for blogs and journals, which makes it easier to back up data and transfer data seamlessly across sites and servers.

Data ownership: WordPress offers secure data solutions and protects its users in terms of data ownership and privacy.

Community: WordPress' features do not stop at its boundaries of capability. WordPress is also backed by a very active and supportive community. The WordPress user support forums and their WordCamp events can be a great place to learn more about WordPress specifically as well as web design and management as a whole. Support can be provided by volunteers and other users on their forums, which helps shape WordPress into this powerful tool for web developers.

On the developer's end of the spectrum, WordPress also offers four powerful features that add great depth to the software's capability. These four features are:

Plugin system: WordPress can be extended via WordPress APIs. WordPress also offers a plugin repository where custom-made plugins can be stored and hosted.

Theme system: WordPress allows custom themes to be created and deployed, be it for personal or commercial use. WordPress also offers a theme repository where custom-made themes can be shared amongst users and developers.

Application Framework: WordPress also provides a plethora of features for an application. They include, but are not limited to, translation capability, end-user management, HTTP requests, URL routing, etc.

Custom content types: WordPress can also be customized via direct code input and can be changed in a great number of ways, from taxonomies to metadata. (WordPress.org 2021)

2.2.3 Application in website management

There are three main types of sites where WordPress has been determined to work best on. They can be classified as:

- normal sites, where content is more static
- blog sites, where content is more chronological and is updated on a regular basis
- e-commerce sites.

Apart from these three primary target website formats, a host of other non-traditional sites can take advantage of WordPress and its power as a CMS. These include membership, portfolio, education site and so on. (Karol 2017, 348- 349.)

2.3 SDLC

SDLC, or system development life cycle is the framework for building a system or program. The process involves five phases of development, with names varying from model to model. The phases are summarized briefly in Figure 6 below, which then will be explained further in later paragraphs.

Phase	Chapter	Step	Technique	Deliverable
Planning Focus: Why build this system? How to structure the project? Primary outputs: – System Request with feasibility study – Project plan	1	Identify opportunity	Project identification	System request
	1	Analyze feasibility	Technical feasibility Economic feasibility Organizational feasibility	Feasibility study
	2	Develop workplan	Time estimation Task identification Work breakdown structure PERT chart Gantt chart	Project plan – work plan
	2	Staff project	Scope management Project staffing Project charter	– Staffing plan
Analysis Focus: Who, what, where, and when for this system? Primary output – System proposal	2	Control and direct project	CASE repository Standards Documentation Timeboxing Risk management	– Standards list – Risk assessment
	3	Develop analysis strategy	Business process automation Business process improvement Business process reengineering	System proposal
	3	Determine business requirements	Interview JAD session Questionnaire Document analysis Observation	– Requirements definition
	4	Create use cases	Use case analysis	– Use cases
	5	Model processes	Data flow diagramming	– Process models
	6	Model data	Entity relationship modeling Normalization	– Data model
Design Focus: How will this system work? Primary output: – System specification	7	Design physical system	Design strategy	Alternative matrix System specification
	8	Design architecture	Architecture design	– Architecture report
	9	Design interface	Hardware & software selection Use scenario Interface structure Interface standards Interface prototype Interface evaluation	– Hardware & software specification – Interface design
	10	Design programs	Data flow diagramming Program structure chart Program specification	– Physical process model – Program design
	11	Design databases and files	Data format selection Entity relationship modeling Denormalization Performance tuning Size estimation	– Database & file specification – Physical data model
Implementation Focus: delivery and support of completed system Primary output: – Installed system	12	Construct system	Programming Software testing Performance testing	Test plan Programs Documentation Migration plan
	13	Install system	Conversion strategy selection	– Conversion plan – Business contingency plan
	13	Maintain system	Training Support selection System maintenance Project assessment	– Training plan Support plan Problem report Change request
	13	Post-implementation	Postimplementation audit	Post-implementation audit report

Figure 6. SDLC (Rosario 2017)

2.3.1 System Planning

The first phase of the system development life cycle (SDLC) is initiation, or system planning. During this stage of the development cycle, the most pivotal element to keep in mind for

any software developer is the accurate planning and outlining of the intended system development process. Key aspects to keep in mind throughout the planning phase are whether the system in question correlates with the company's plans, budgets, as well as mission. The planning process should also look into the feasibility of the system development process itself. (Rosario 2017.). Figure 7 lists down several prerequisites of planning in a SDLC.



Figure 7. System planning checklist (Rosario 2017).

The very first stage of the planning process should always introduce system security into the equation. This is a very important part when considering the applicability of said system into the business' operations. Another important clarification to make during this stage of the development life cycle is the identification and generalization of the intended information types to be included in the finalized system. A clearly defined system with pre-defined information type will require less maintenance and error fixes upon its deployment for use, thus cutting down costs even further for the end-customer. (Cleland & King 1997, 259-261.).

This can be either a great way for businesses to cut down on costs, or for software developers to use as a selling point for their system or product.

In order to achieve the aforementioned requirements from a planning phase, system development needs to categorize its information on par with a privacy impact assessment (or PIA). The phase ends with a cost analysis of the entire SDLC, with costs projections for the final system and having these planned features and expenditures approved by a committee. (Rosario 2017.)

2.3.2 System Analysis

During the analysis phase, it is paramount that a number of questions should be answered. The analysis process will look into the system's requirements. This can be collected through any number of information compilation methods. The suggested requirements for the analysis phase can be found in Figure 8.



Figure 8. System analysis checklist (Rosario 2017).

Analysis strategy: The analysis strategy is the backbone of the analysis stage. A strategy will need to be the first step of any system analysis plan and will encompass all methodologies that will be applied into the analysis process (Rosario 2017).

Requirement analysis: The requirement analysis process is a lengthy process of collecting and compiling requirements from all of the company's stakeholders. The requirements will be determined so as to guide and ensure that the system's development process flows in the correct direction and follows the company's plans and vision. The requirement analysis phase also takes place to clarify the wishes and demands of the company's stakeholders, as well as their requirements for the project in question. According to Robertson and Robertson (2012, 9-11), there are two forms of requirements to be identified during a SDLC, namely functional and non-functional requirements.

- **Functional requirements:** they are requirements that determine to the letter what the system needs to do, or what content components it has to have. These requirements include different primary operational processes, as well as the data required on the system.
- **Non-functional requirements:** They are requirements that do not directly determine a website's core function, but its behaviour and properties. These requirements can include the system's accessibility, such as the web browser needed to access the system, or the system's end-user interface, for example, how it looks and how it displays information. A brief summary of this aspect boils down to four core non-functional requirements that encompass the bulk of the cases. They are operational, performance, security and cultural and political.

The SDLC requirement analysis phase follows a two-part process, which run in conjunction with one another and feeding into each other's information pool, as well as assist in the further operation of the system in question. The two parts in requirement analysis process are:

- **Determining requirements:** The process of determining system requirements involve the developers gathering inputs from all of the company's stakeholders, as well as the project's stakeholders specifically. The information that are needed for the next step includes both types of requirements as mentioned above.
- **Structuring requirements:** The collected requirements are then compiled and prepared for the structuring process, which involves the translation of the requirements into comprehensive forms, including graphs, charts and diagrams for easier interaction during the development process. (Rosario 2017.)

2.3.3 System Design

When the analysis phase draws to a close, the design stage of the SDLC can begin. The design phase is where a prototype of the system in question will be developed. Figure 9 presents the list of steps needed in this particular phase.



Figure 9. System design checklist (Rosario 2017).

Design strategy: The first step of the design phase is the development of a design strategy for the entire design process. From a company's perspective, the design strategy includes the selection of the means by which the system will be developed. The first question to be asked during this phase is what entity will take charge of the design and development process of the new system.

Design architecture: The second step for the design process of the SDLC include the planning and acquisition of the appropriate design architecture. This calls for the identification of hardware, software and network requirements for the design process of the SDLC.

Database and file specification: This step pre-determines what types of data will be stored, as well as the storage for these data. The data can either be stored in a common repository, or in file bundles. Contemporary system designs can also be stored via cloud-based storage repositories for easy access throughout the entire development process.

Design of the system: After all of the preliminary parameters have been met, the actual design process of the system in question can begin. The design process is a long and complicated process and will depend heavily on the complexity of the system being designed.

(Avison & Shah 1997, 145-161.)

2.3.4 System Implementation

Construction: The system implementation process starts with the construction of the system. As the name suggests, this phase involves the coding and testing of the system in question. The output of this process should be the source code for the application/software which is later used as an input for testing phase and maintenance plan.

Installation: After the system has been constructed, the installation process is ready to take place. The installation of a new system involves the possible removal of the old system in the direct cutover approach of system installation, or the newly constructed system can run in conjunction with the old version, which is often coined the parallel conversion approach. The new system can also be installed in a phased conversion strategy, which involves the installation of the new system in a region of the company in order to test for issues and later replacing the rest of the company's system, hence the phrase phased conversion plan. (Avison & Shah 199, 167-174.) Implementation actions are shown in Figure 10.



Figure 10. System implementation checklist (Rosario 2017).

3 Used tools and technologies

3.1 HTML

HTML, otherwise known as a hypertext markup language, is the backbone of every consumer website in existence all over the internet. HTML has long been the starting line for any web developer and is the standard language for website structure development and management, regardless of website type and scale. Unlike many other elements of web development and management, HTML has not been through a great deal of change, and a comparison of contemporary HTML features and language against an HTML version created 2-3 decades ago does not yield a remarkable difference. In other words, a website developer of the 1990s can familiarize him or herself with modern HTML with little difficulties. (White 2020, 6.)

3.1.1 Definition

HTML is the standard language for web page creation and modification. HTML falls into the category of a markup language, meaning it specializes in document annotation. HTML is ever-changing. However, as mentioned above, the changes are merely on the feature scale only, not on the language as a whole, thus allowing for a developer to adapt quickly to a newer HTML version with experiences using an older HTML. The current HTML version is HTML5 and met the W3C standard in 2014. HTML has long been used by major and small browsers, including big players like Chrome, Safari, Firefox, and Opera (Kurniawan 2015, 1-2).

3.1.2 Elements

HTML at its core is comprised of tags with different attributes. As HTML is a relatively important tool for website development and management and will be at the heart of this paper, this section will take an in-depth look at the elements of HTML for a more complete view of the language and how it operates.

Tags in HTML are placed within a bracket. Some tags will have a corresponding closing tag, while others do not. For example, a `<html>` tag may be followed by a `</html>` closing tag, or a `<head>` tag can be followed by a `</head>` tag. The purpose of a closing tag is to signify the end of a piece of content or a structure. Some tags do not require a closing tag, such as the case of the `` tag, which consists of a forward dash (/) in of itself. (White 2020, 7-8.)

These are but a small percentage of tag types available in HTML, which will provide further intensive functions for a webpage's structure, including, but not limited to, hyperlinks, markup, table, inputs, etc.

Attributes are another element of HTML, and they are used to defined unique functions to tags (White Belt Mastery 2020, 8). An example of this could be found in the `` tag, with the `src` attribute holding the source of the image for input, while the `height` attribute holds the height value of the image in question for display. Similarly, the `width` attribute will house the width information of the image. To add an attribute to a tag, the attribute follows the tag with a space and the value or information is added after the attribute, separated by a "=" and placed within a pair of "text".

3.2 CSS

CSS (Cascading Style Sheets) is responsible for the presentational functions of a website's contents. Without CSS, a website will only consist of lines of tags from an HTML sheet, which, at the bare minimum, will be very hard and frustrating to read even for other developers, let alone for public consumption. CSS allows the creation and modification of various content element adjustments, including fonts, size, positioning of various content elements, as well as entire webpage layouts. This is in order to ensure an appealing and readable webpage. (Pouncey & York 2011, 3-10.)

In order to fully understand the gravity of CSS in a website's appearance, one can just simply look at a comparison between the same home page of Amazon with and without CSS in Figure 11 and 12.



Figure 11. Amazon's home page with CSS (Amazon.com 2021).

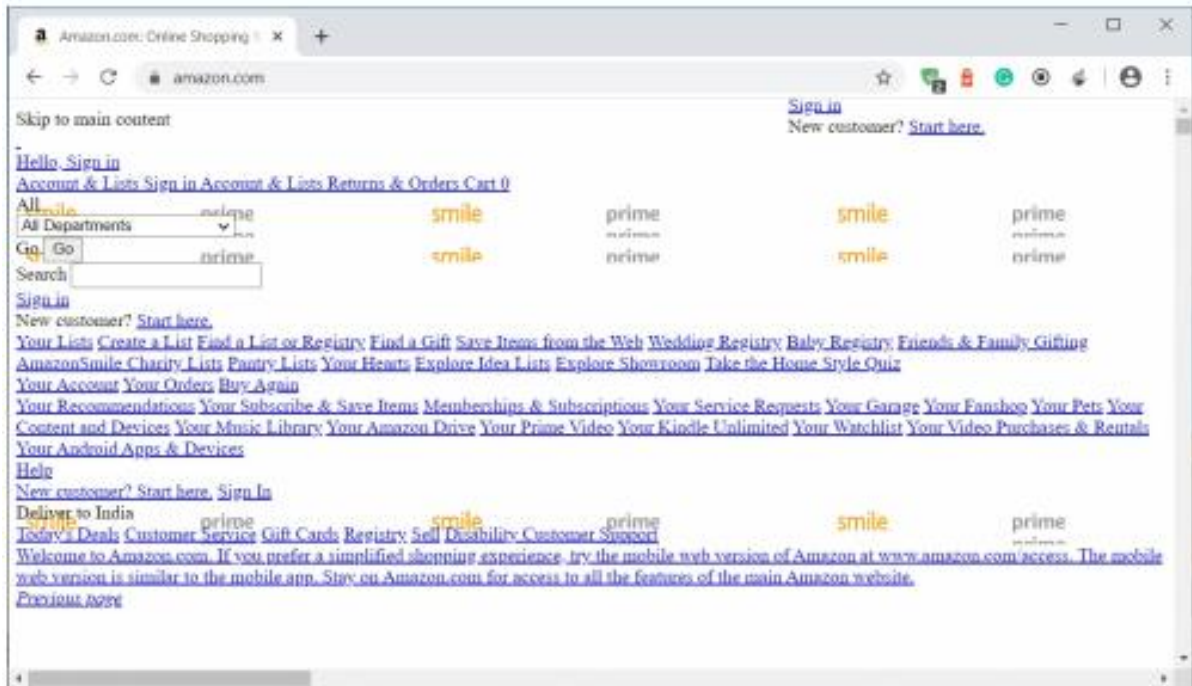


Figure 12. Amazon's home page without CSS (Amazon.com 2021).

From this comparison, it is clear that CSS plays a major role in converting a simple, plain webpage into a beautiful and interactive storefront. If HTML is the structure of a website, CSS is its façade.

This, however, does not deny HTML as a proper tool for styling a website, as websites could be created and modified for a decent appearance without the involvement of CSS. However, CSS has made the website interface customization process much more efficient and easier to grasp.

CSS exists in three forms, either external, internal, or inline.

External CSS exists in a separate file from the HTML document and must be referenced in the head section to apply these rules to the HTML document.

Internal CSS is CSS that is placed directly in the HTML document in the <head> section in the <style> tag.

Inline CSS is CSS that is placed directly in the HTML document's body using the style attribute and can be applied in small scales like a paragraph or heading. (White Belt Mastery 2020, 76-79)

3.2.1 Compatibility

CSS' displays are different depending on the end-user's browser of choice. This calls for intensive testing for every CSS code before it can go live. This needs to be done in order to ensure that the code works as intended, and the content elements show up where they are supposed to, regardless of the browser used.

Testing can also be minimized via the generalization of the browser in terms of the rendering engine, as browsers run by the same rendering engine will basically show very little difference in terms of interface and display settings, meaning they will likely have no discernible impact on the CSS. There are currently four major rendering engines in use as of today, and their power some of the biggest players in the field of browsers.

- **Webkit:** Powers Safari and Google Chrome. They are some of the most popular browsers in use as of today.
- **Trident:** Powers Microsoft's Internet Explorer series of browsers. This series of browsers is also a big player in the market and should be considered when testing CSS.
- **Gecko:** is most famous for powering the Mozilla Firefox and is quite popular amongst Mac and Linux operating systems, despite its recent decrease in popularity amongst Windows users.
- **Presto:** Powers Opera, which is seeing a decline in computer market share, but a drastic rise in the mobile sector.

(Pouncey & York 2011, 11-14).

3.2.2 Core functions

The core functions of CSS can be subdivided into bits, which work together to create a complete CSS, with all its modifications and functions. They can be summarized as points in Figure 13:

```
body {  
    width: 650px;  
    margin: 0 auto;  
    background: #000;  
    color: #FFF;  
    font: 12px sans-serif;  
}
```

Figure 13. CSS sample rule (Pouncey & York 2011, 18).

- **Style sheet:** is the main sheet of the CSS, where the CSS document will be created.
- **Rule:** is the properties that are applied to a set of HTML elements, which will change their layout and appearance on the main webpage. (Pouncey & York 2011, 17-18.)
- **Selector:** is the HTML element where the CSS rule is applied. Selectors come in many forms, two basic types of selectors include type selectors, where one element of the HTML will be selected for the style, and it will match that element, and group selectors, which will group elements and apply the same rule for these grouped elements. (Pouncey & York 2011, 18-19.)
- **Property:** is the main function of the CSS and exists to provide a property to various HTML elements. Properties are necessary for the identification of HTML element rules, such as font-size, height, colour, background, list, and text positioning. The properties are indicated by a value, which defines the properties themselves. For example, in Figure 13, width is a property, and 650px is the value defining that property. (Pouncey & York 2011, 205-219, 94-111, 73-87.)
- **Value:** Value is a single value that is added to define a property. The value type will depend on the property type and may include keywords, strings, measurements, numbers, colours, URL, etc. (Pouncey & York 2011, 19-33.)

3.3 PHP

PHP is a programming language that dates back to early 1994. Its development process has been drastic over the course of its existence, and the language went from an answer to password protection and online forms to one of the most popular languages for websites,

with over 10 million top sites currently using it, as of 2020. (Tatroe & MacIntyre 2020, 33-39.)

3.3.1 Definition

PHP or Hypertext Pre-processor is a very powerful and versatile language that excels at the creation of HTML content. It is capable of 2 main functions, namely server-side scripting for dynamic web content creation and command-line scripting for system administration. For the context of this thesis, the majority of the focus shall be placed upon the first core function of PHP, which is ideal for developing dynamic web content for deployment on a website. (Tatroe & MacIntyre 2020, 33-34.)

3.3.2 Core component

PHP has a number of core components that are needed to be fully functional. They include variables, operators, conditionals, arrays, loops, functions, and classes and can be specified as follows. (Carr & Gray 2018, 2-11.)

Variables: The fundamental building blocks of a PHP document. They are a means to temporarily store data and are defined by a \$ symbol, followed by an assignment operator and a value. This is quite similar to the CSS property and its value assignment, but with an operator involved.

Operators: A vital building block of a PHP document. They signify a relationship between components, like in the case of variables, or trigger an operation within the text. There are three primary types of operators in PHP, namely:

- **Comparison operators:** are assignment operators, which allow the user to compare and contrast between values within a condition. Below in Figure 14 is a list of comparison operators.

Usage	Name	Description
<code>\$a == \$b</code>	Equal	TRUE if \$a is equal to \$b.
<code>\$a === \$b</code>	Identical	TRUE if \$a is equal to \$b, and they are of the same type.
<code>\$a != \$b</code>	Not Equal	TRUE if \$a is not equal to \$b.
<code>\$a !== \$b</code>	Not Identical	TRUE if \$a is not equal to \$b, or they are not of the same type.
<code>\$a < \$b</code>	Less Than	TRUE if \$a is strictly less than \$b.
<code>\$a > \$b</code>	Greater Than	TRUE if \$a is strictly greater than \$b.
<code>\$a <= \$b</code>	Less Than or Equal To	TRUE if \$a is less than or equal to \$b.
<code>\$a >= \$b</code>	Greater Than or Equal To	TRUE if \$a is greater than or equal to \$b.

Figure 14. Comparison operators (Carr & Gray 2018, 6).

- Logical operators: are the next type of operators, and allow the user to check for cases, which is very important for a logic gate. Below in Figure 15 is a list of logical operators:

Usage	Name	Description
<code>! \$a</code>	NOT	TRUE if \$a is not TRUE.
<code>\$a && \$b</code>	AND	TRUE if both \$a and \$b are TRUE.
<code>\$a \$b</code>	OR	TRUE if either \$a or \$b is TRUE.

Figure 15. Logical operators (Carr & Gray 2018, 6).

- Mathematical operators: are exactly what their name implies. They allow the users to perform four main mathematical operations, as well as acquiring the modulus of a division. Below in Figure 16 is a list of mathematical operators.

Usage	Name	Description
$\$a + \b	Addition	Sum of $\$a$ and $\$b$
$\$a - \b	Subtraction	Difference of $\$a$ and $\$b$
$\$a * \b	Multiplication	Product of $\$a$ and $\$b$
$\$a / \b	Division	Quotient of $\$a$ and $\$b$
$\$a \% \b	Modulus	Remainder of $\$a$ divided by $\$b$

Figure 16. Mathematical operators (Carr 7 Gray 2018, 7).

Conditionals: Statements that allow for control of the program. They are the logic gates of the PHP document that regulates and determines the data that is allowed to be displayed and those that are not. Below is a simplified illustration of how conditionals operate.

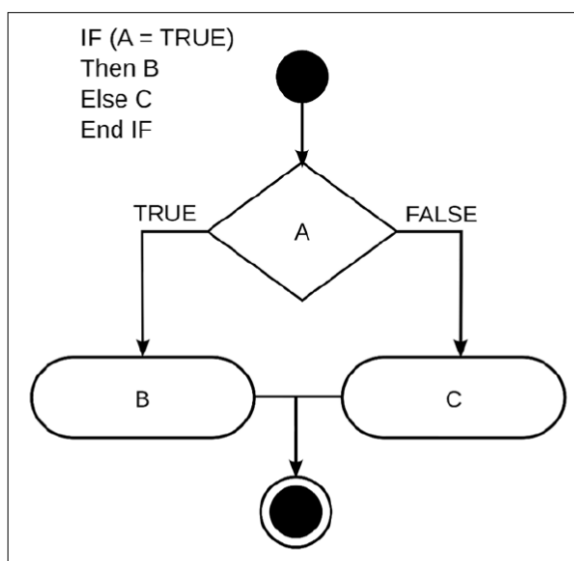


Figure 17. Conditionals (Carr & Gray 2018, 9).

Arrays: A type of variable that allows for the storage of multiple items in a single container (Carr & Gray 2018, 15). There are three main array types, which can be disclosed as follows.

- Indexed arrays: are the most frequently used type of arrays that are used to store values into variables (Carr & Gray 2018, 16).

- Associative arrays: are similar to indexed arrays, but it can be used to assign values to texts, rather than variables (Carr & Gray 2018, 18).
- Multidimensional arrays: are arrays within arrays and allow for the storage of various information for variables (Carr & Gray 2018, 22).

Loops: A means to re-apply the same code block a number of times in order to reuse a defined code block and save time for the developers (Carr & Gray 2018, 15).

Functions: A complete package with a predefined process of processing an input parameter and outputting a value. The means and measures that the function takes depend on the pre-determined code assigned to it prior to deployment (Carr & Gray 2018, 36). Below is a simplified diagram of a function.

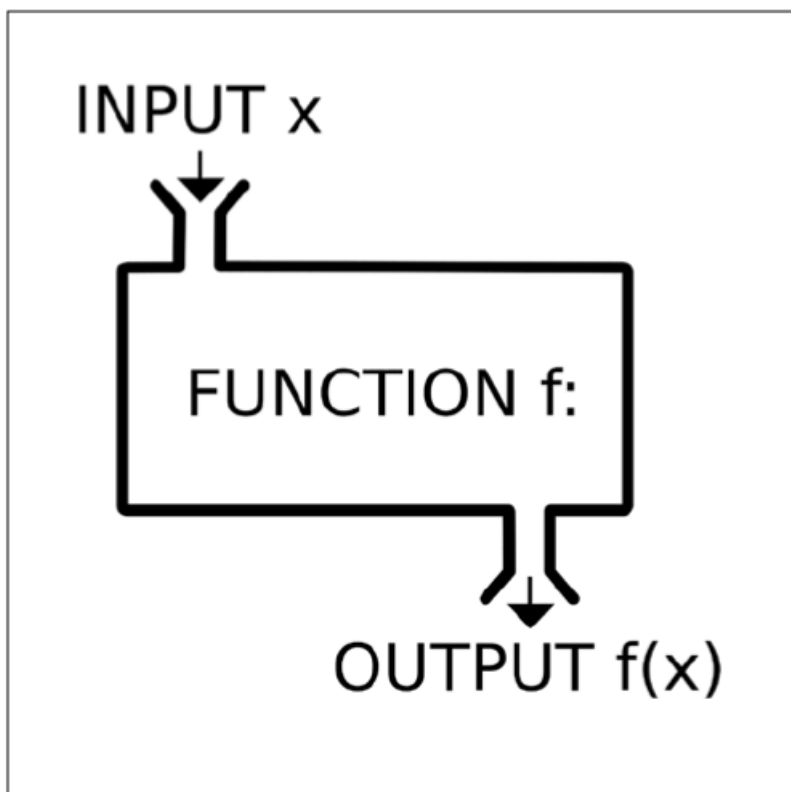


Figure 18. Function (Carr & Gray 2018, 36).

Classes: A means to assign a class to a group of variables and functions into a complete package, which will possess the overall characteristics of the class. The base class functions and variables can also be applied to the children of the class, this is known as the inheritance property of classes (Carr & Gray 2018, 40).

3.4 JavaScript and JSON

JavaScript is perhaps a household name for the majority of web developers and is the go-to language for web design and development for the majority of web developers, regardless of website scale. JavaScript is one of the most popular programming languages to date, and powers most of the contents available on the internet. (White 2020, 6.). It was first known as Mocha in 1995, then LiveScript before finally getting the name JavaScript in late 1995 (Antani et al 2016, 4).

JSON or JavaScript Object Notation is a data-interchange format and open standard file format that is used to store and transmit data objects. JSON was derived from JavaScript and is an independent programming language, but it also has conventions like the C-family programming language (e.g., C, C++, C#, Java, JavaScript, Perl, Python) and many others. JSON is built on two structures:

- A collection of name/value pairs
- An ordered list of values. (JSON 2021.)

Within the context of this thesis, the focus shall only be placed upon JavaScript and its use in conjunction with HTML and CSS to create a dynamic website. JSON will be used to transfer data from backend side to frontend side. This limitation of scope has been purposefully done in order to zone the necessary information regarding JavaScript and its relevance in the study, as JavaScript is a complex and broad-spectrum to take into account on a complete scale.

3.5 Laravel Framework

Laravel is a PHP framework, whose brief overview can be found in section 3.3. PHP. The Laravel framework is a relatively new framework, as far as PHP frameworks are concerned. It was first created and commercialized by Taylor Otwell in 2011. The Laravel framework was first conceived as a better solution to the CodeIgniter framework that has been in use for the most part of PHP's early years. The current Laravel version is Laravel 5, with a new directory structure and features. (Stauffer 2019, 3-4.). The Laravel framework is a very popular framework to be used by websites of all sizes, which can be seen from the chart in Figure 19.

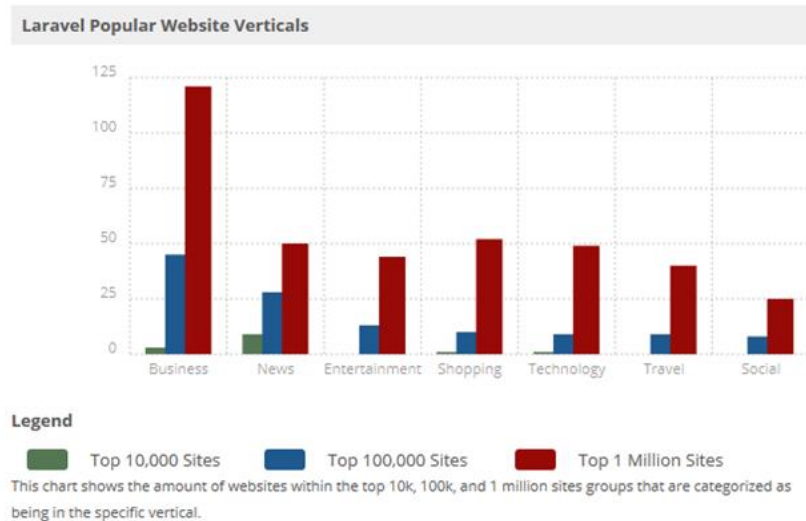


Figure 19. Laravel popular website verticals (Techcronus 2018).

Laravel includes a large number of features. However, in the interest of a comparison, this section will only look at features that are either unique to Laravel or are not very commonly found in other frameworks of its type.

- **Template engine:** Provided by Laravel and consists of a plethora of templates for use in PHP templating. The main selling point for Laravel is the template engine it offers, specifically the Blade template, which does not restrict the use of plain PHP codes in views. This is a bonus, as Blade will not add considerable overheads to the development cycle of the application.
- **MVC architecture support:** offered by Laravel, which provides a number of useful functions for use in application performance and scalability enhancement.
- **High security:** Laravel is also well-known for its high security and tight data encryption. The Laravel's password mechanism and Bcrypt hashing Algorithm ensure that passwords are well encrypted, even within the database.
- **Artisan:** One of the main boasting points for Laravel as a framework. It is a command line console that allows for the automation of programming tasks, much like a loop for PHP. Artisan comes packaged with a variety of commands that can be used to ease the programming process while facilitating more complex functions with more simple code structures.
- **Pre-installed libraries:** Laravel comes packaged with object-oriented and modular libraries for use in PHP document and application development.
- **Eloquent ORM:** Included in Laravel and is a simple Active Records implementation for databases.

- **Laravel service container:** A versatile and convenient tool that assists in the management of class dependencies as well as dependency injection into existing classes.

(Laravel 2021.)

3.6 MySQL database

MySQL database is a relational open-source database management system developed by Oracle. MySQL, as its name implies, operates on the structured query language, or SQL. This is one of the most popular database platforms in its category as of 2018, and nearly 100% of surveyed high traffic website is currently using the MySQL platform. The current version of MySQL is MySQL 8.0, which features an array of improvements and life quality improvements for developers. (MySQL 2018a, 3.). The key features that distinguish MySQL from its competitors can be summarized in section.

High security

The platform offers improved identity management and data encryption solutions. MySQL also provides more secure settings and better security for use configurations, as well as a monitoring system for cardholder data, along with the MySQL Enterprise database firewall which will protect the user's data and login information. (MySQL 2018b, 4-8,12-15.).

High Availability

MySQL is supported on all Oracle platforms and services, featuring InnoDB, Global Transaction, Performance schema, big data repositories, E-commerce apps, and high reliability on embedded programs and applications (MySQL 2018c, 5-6.).

Record speed

MySQL 8.0 promises a record speed in queries generation, and breaks previously established benchmarking record, as can be observed in Figure 20.

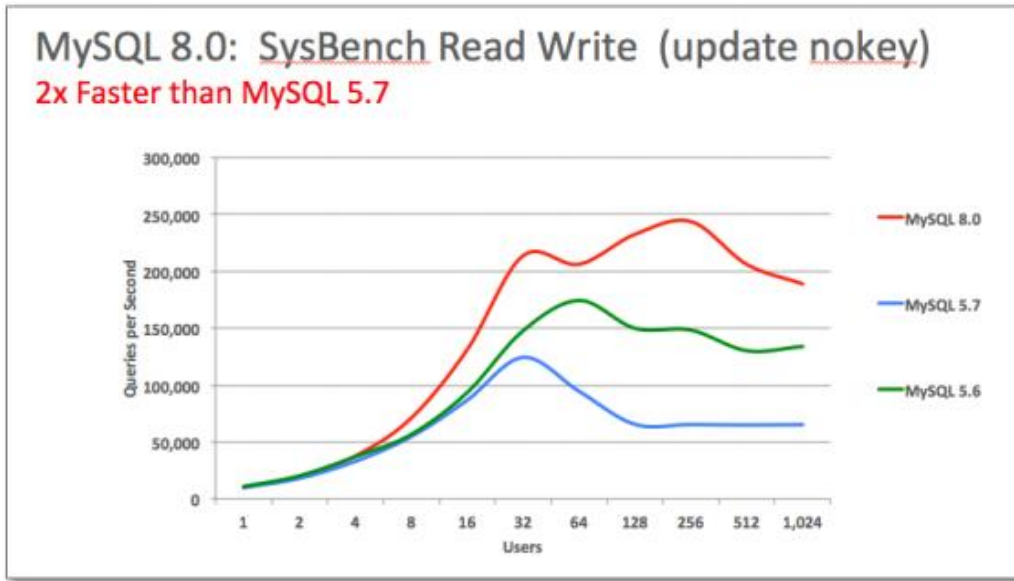


Figure 20. MySQL 8.0 vs MySQL 5.7 and MySQL 5.6 (MySQL 2018a, 4).

3.7 Git

Git is defined as an open source distributed version control system that specializes in project management and control with great efficiency and quality. It was developed by Linus Torvalds, and works in a similar fashion to other distributed version control systems, like Bazaar, Mercurial, etc. The primary function of Git is to keep track of any and all changes in the code, while also storing those changes in a common repository. These versions and data can be freely accessed by other users, hence the “distributed” part of its name. This distinguishes it from local version control systems, as they, as the name implies, store the version data locally, rather than via a server. (Chacon & Straub 2014, 10-12.) A simplified diagram of how Git works can be found in Figure 21.

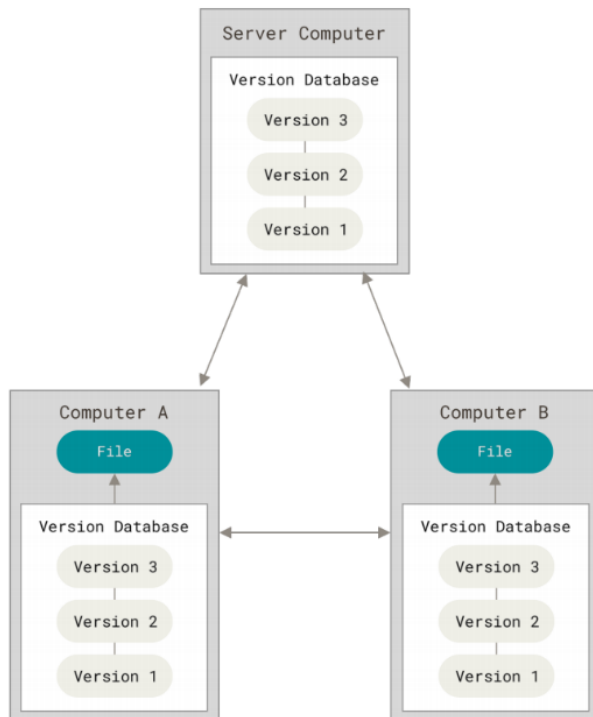


Figure 21. Distributed version control system (Chacon & Straub 2014, 13).

3.8 Virtual machine

Virtual machines are classified as virtual computer systems that can deploy and run applications, as well as perform computing tasks similar to that of a physical computer. The main difference between a virtual machine and a physical computer is that the former does not require a separate physical computer for every instance of a virtual machine in operation. This means that several virtual machines can operate on single computer hardware, thus allowing for compartmentalization or testing of new software and operating systems without the need for another piece of computer hardware. (VMware 2021.)

There are currently two types of virtual machines, namely process virtual machine and system virtual machine. Along with these two virtual machine types, there are also five types of virtualizations that can be achieved using commercial and personal means. They are hardware, software, network, storage, and desktop virtualization. All of these virtual machine types and virtualization types can be summarized as follows.

- **Process virtual machine:** A place where only one process is allowed to be deployed and run on a host machine. These types of machines are very common, as

they are used to hide the actual system information of the hardware and operating system.

- **System virtual machine:** A full-fledged virtual machine with all of the capabilities and functions of a physical computer. These types of virtual machines are more demanding and make use of hypervisor technology.
- **Hardware virtualization:** The process of virtualizing operating systems and computer applications, which are then duplicated and centralized in a physical server, which communicates with the physical computer via a hypervisor. This process allows for more than one machine to be run on single computer hardware.
- **Software virtualization:** A very common form of virtualization that sees utilization in a variety of cases regarding networking and remote access work. The software virtualization process requires a physical computer system that acts as a platform where guest operating systems can operate. Applications are also available candidates for software virtualization, which facilitates a possibility of a centrally run application that can be useable for remote devices.
- **Desktop virtualization:** The process of virtualizing a desktop interface on a centralized server, where it can be accessed from any number of devices owned by the user. This can help save costs for software as well as hardware and facilitates an easier management process.

(VMware 2021.)

3.9 Communication tools: Zoom & Email

Zoom and Email are the means for communication with survey participants, as well as conducting online interviews in the current time of the pandemic (Iqbal 2021). Most of the interactions with potential participants along with a bulk of the survey/interview phase will be done online via these two platforms for safety reasons.

4 Case company: Company X

4.1 Organization review

Company X is a social joint organization for Health and Wellbeing. The organization offers a variety of health services such as specialist medical care, services and rehabilitation for the elderly, family and social services. Members of the organization are public health centres, private operations and other associations that work in healthcare and social services. All the services are meant for the people living in several municipalities nearby.

4.2 Current situation

Since company X provides a wide range of health services, a lot of electronic services and platforms are made available for people to use and seek the most suitable help. In addition to services that are related to medical care, company X also supports family and social services. In this service section, company X has partnered and co-operated with many other public and private organizations, both on long-term and short-term events and projects. The need for hosting events is an indispensable part of the company's operations. Events and activities can be served for many purposes and meant for many people, including customers, clients, stakeholders and so on. This specific service section is where company X needs a solution for a data management system specified for their events scheduling.

A common difficulty in managing all the events is that it requires a structured data system to store and manage events and partners' information. Another obstacle is to find an efficient way to spread the news to attendants. At the moment, the majority of public activities and events held by company X are updated through many approaches. However, they are still publicized in a limited way. For instance, some events are notified traditionally through leaflets and posters on bulletin boards of hospitals or public health centers. A different method to announce activities and events to the public is through the internet, it can be social media or related websites. Figures 22 and 23 show the events section on a healthcare organization's Facebook page with both upcoming and past events. By browsing on these posts, users can have a better grasp of each event's host and partner, date and location, and a lot more details. Followed are Figure 24 and 25 showing how social events and activities are announced and updated via specific websites. These approaches act as examples for company X to utilize the internet as different information distributing channels.



Figure 22. An event on a healthcare organization's Facebook (PHHYKY 2021).

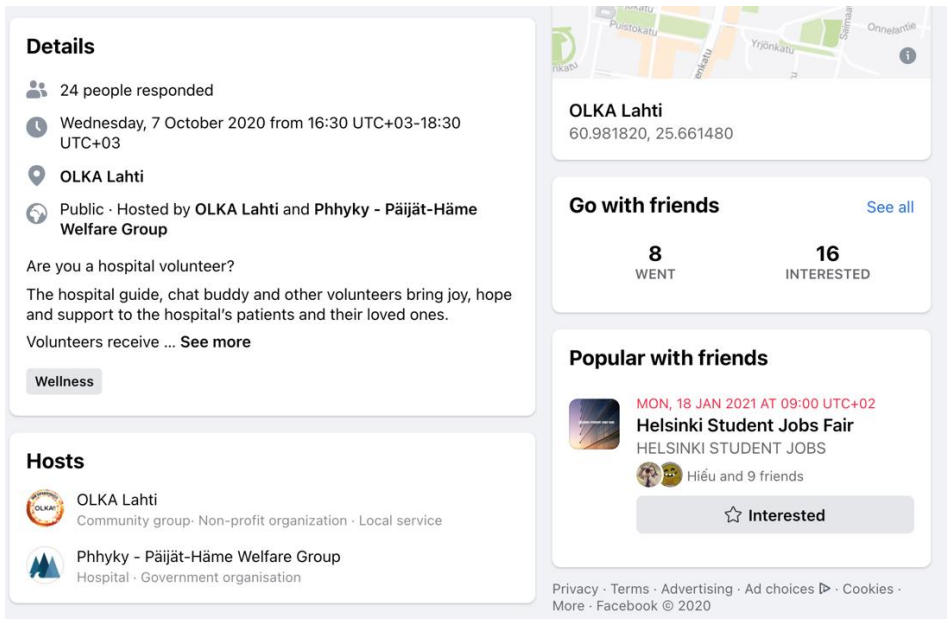


Figure 23. An event on a healthcare organization's Facebook (PHHYKY 2021).

ILMOITTAUTUMINEN

LUKUJÄRJESTYS KEVÄT 2021

[Tonttila-sali](#)
[Kivimaan ala-aste](#)
[Ahtialan nuorisotalo](#)
[Päiväryhmät](#)

Kun ilmoitat esimerkiksi lapsesi tanssitunnille niin kirjoita kommenttikenttään harrastajan nimi sekä syntymävuosi, kiitos.

Tonttila-sali, Salonaukio 5, Lahti

PÄIVÄ	OHJAAJA	KLO	LAJI	RYHMÄ	KAUSIMAKSU	
MA	Kiia H	16.00-17.00	HipHop yli 9 v	Hip'n'Hop	90	Ilmoittaudu
	Kiia H	17.00-18.00	Edustus, yli 13v	LegaciS	240	TÄYNNÄ
	Kiia H	18.00-19.00	HipHop yli 7 v	Beatz	90	Ilmoittaudu
	Kiia H	19.00-20.00	HipHop yli 11 v	Hiphoppers	90	Ilmoittaudu
	Kiia H	20.00-21.00	Commercial HipHop yli 15 v	Nonstop	90	Ilmoittaudu

Figure 24. Event section on a website (Möysän Nuorisoseura ry 2021).

Kevätkauden vauvapilates Karismassa (8301273)

1.2.2021 – 10.5.2021 (virtuaalinen)

Ma klo 10.00-10.45

13 tuntia / 70 €

Kauppakeskus Karisma, jumppaan osoitettu liiketila, Kauppiaankatu 2, Lahti

Tuija RantalaÄiti-vauvapilates -tunnilla vahvistetaan ja huolletaan kehoa huomioiden erityisesti raskauden ja synnytyksen aiheuttamat muutokset kehossa tehden paljon esim. lantionpohjan ja keskivartalon harjoituksia. Tunti etenee vauvojen ehdoilla ja harjoituksia tehdään joko itsenäisesti tai vauvan kanssa huomioiden mm. sylittelyt ja syöttämiset. Ota oma jumppa-alusta mukaan. Yhdessäoloa ja hyvinvointia äideille ja vauvoille! **Huom! Ryhmä kokoontuu Karisman PapuParkissa**, joka on suljettu muilta käyttäjiltä jumpan ajan.



Figure 25. Example event on a Welfare Association's website (Harjulan Setlementti ry 2021).

However, totally relying on social media and the internet to publicize events also has many drawbacks. The first drawback is that social media is not widely used by everyone. For

instance, if a particular event is held for the elderly, social media will be an inefficient approach that may lose most of the event's target audiences.

Another drawback in events management is the control over events' information as well as hosts and partners' data. Since company X has cooperated with many partners, it is burdensome for customers to follow all the relevant information and events if there is not a centralized data center. Customers or participants cannot possibly follow all of the social media channels and websites to keep up with the events. Moreover, this obstruction is not only challenging to organizations who want to run their events, but also to company X who also wants to administer all their partners and procedures. For these reasons, it is necessary for company X to have a concentrated system to manage and announce all the events and activities officially for their partners' and their own convenience.

4.3 The need for a new service

The difficulties in managing events mentioned above demand company X to have a new system to solve all the obstacles. The new system will be a place where company X can meet all the following requirements:

- A map for everyone to browse upcoming events, filter and search for a suitable event.
- An interface for company X's registered partners to add, modify events and event details. And data from this interface is connected and synchronized with the event map.
- Company X has control over the system, which includes its registered partners and all the information displayed to everyone.
- A database that can store all information including users and events' details.

In order to overcome the difficulties and requirements from the case company, the solution will be explained in the next chapter: Practical implementation.

5 Practical implementation

5.1 System Planning

Building a multi-functional system requires lots of time and work. Therefore, planning on system development ensures every stage of the application needed to be micro-managed very carefully. In this project, the concept of SDLC was used as a reference to keep the project follow the right path, but not strictly to the theory because different cases required different step and decision. Based on SDLC framework described in Chapter 2.3, the planning phase decided the developing process was divided into 5 stages, below are the detailed description of each stage.

- Analysis: Do a comprehensive inspection on the current stage of the project and come up with a strategy which will be described in the next sub-chapter.
- Preparation: Prepare bed-rock requirements, those include runtime environment, Laravel template installation.
- Design: Do research and make a decision on designing the project structure and interface for the application.
- Implementation: Start working on actual technical implementation. In this stage, the application will be transferred from the previous project into the new structure and new functions will be implemented.
- Finalize and documentation: In this stage, the application will be warped up and related important information will be written down for future use and maintenance.

5.2 System Analysis

Based on the theory of System Analysis that described in chapter 2.3.2, the initial stage of the project was wrapped up into three different parts:

- Backend part stored data in a database and handled requests from AdminUI and UserUI.
- AdminUI part was an interface for people in Company X as well as collaborated organization be able to add/modify/delete events data. This UI would take actions and interact with the Backend using JSON and HTTPS requests.
- UserUI part was a map that display events that were operated near a specific place. Events data was provided by Backend using JSON and HTTPS requests.

Previously, both Backend and AdminUI were built in Symfony, a PHP framework, and UserUI were built as a WordPress plugin/widget. The original design required three different

systems to run each part respectively, which could possibly cause problems for the data flow and slow down the whole application. Therefore, in this stage, Backend and AdminUI will be merged and will be built in Laravel framework as mentioned in chapter 3, UserUI will remain the same as the original implementation. The decision was made based on the following reasons:

- Both Laravel and Symfony have integrated database. In order to fulfil the requirement of the initial project, Symfony has been used due to the flexibility of the framework that allows developers to separate the database and make it into a web service. In this project, Laravel is a better choice because this framework provides more support regard database integration and fully equipped with tools to make a user interface for a website. This can improve the data flow and take down a considerable amount of risk that could badly impact the performance of the application.
- Merging Backend and AdminUI will maximize the efficiency of modern technology as Laravel has been released and constantly updated for more than 10 years. Some part of Laravel which can be greatly utilized are Security and Authorization system, Database, API interfaces, etc.

This decision will maximize the efficiency of modern technology, utilize integrated database of Laravel and make the application easier to maintain in the future.

5.3 System Design

In the current stage, AdminUI has been merged successfully into the Backend. It has the same use as AdminUI and also responsible for providing data to UserUI by JSON. All functions and APIs have been tested and work normally on development environment, except for UserUI. The UserUI, which display the map of events, still has some errors with display and visual that were left from the initial project and they have not been fixed yet.

5.3.1 Data structure

Figure 26 below illustrates the data structure used within the application.

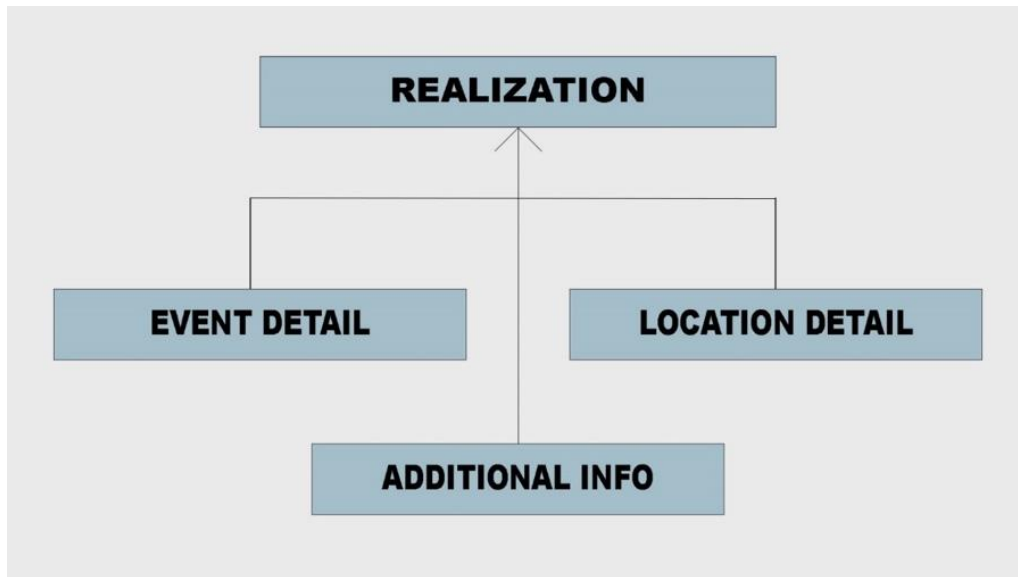


Figure 26. Data structure

All information about an event will be stored in a model called “Realization”. Each “Realization” contains three components: “Event Detail”, “Location Detail” and “Additional information”. Each component had its own table in the database for future use when needed or interact such as edit or delete. Detail about each component is described below:

Event detail:

- Event name
- Category
- Specialists
- Host Organization
- Creator name
- Editor name
- Event description

Location detail:

- Address
- Postal Code
- City
- Name of Host Organization
- Name of creator
- Name of editor
- Location description

Additional information:

- Target group
- Starting time and Ending time
- Name of creator
- Name of editor
- Notes

Each part will have its own form to add new data/edit existed data and a page to display the list of data. The default status of each instance is 'active', when a user deletes one, its status will change to 'inactive' and can only be restored or permanently deleted by Admins. There is a report page where users can transfer the list of data into an Excel sheet.

5.3.2 Authorization and control

Authorization was implemented to limit the access of users and provide the case company a complete control of the platform. By utilize authentication system of Laravel Framework, which was described in Chapter 3, there are four levels of user were implemented:

Initial User

Initial User is the very first user that has access to the platform. Credential that used to authorize will be created during installation stage. Caution: This user will ALWAYS have FULL CONTROL of the platform and can only be created by people who have access to the server and .env file.

SuperAdmin Users

Users that are managers from Company X and responsible for controlling the platform. SuperAdmin users can authorize and create new Admin users, can see/edit/update/delete all events/locations/realizations in the database (including instances that belong to other organizations). Everything will be noted down and display on the detail page so everyone will know who to contact in case something went wrong. SuperAdmin users can see the list of users, ban/unban users, see statuses and statistic of each user, or change the role of users from "Admin" to SuperAdmin". SuperAdmin users can also manage deleted instances by restore or permanently remove from the database.

Admin Users

Users have to be authorized by Company X to be able to login and access to the AdminUI platform. Users can create new instances (events, locations, realizations) and see as well as edit/update/delete their own/belong to their organization instances.

Anonymous Users

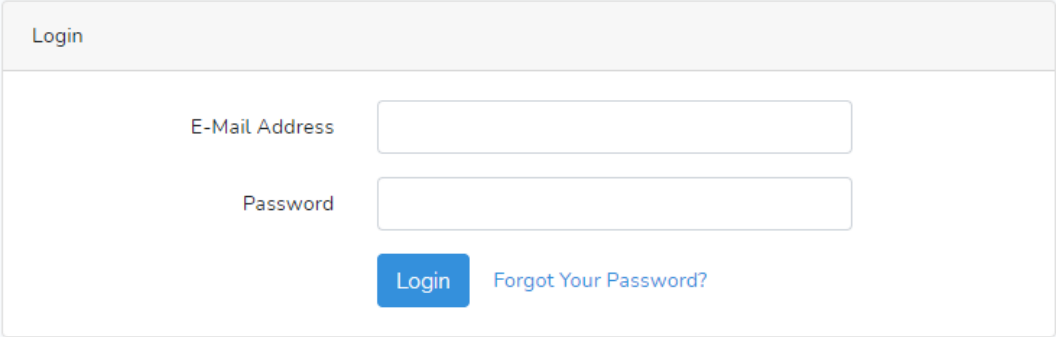
Anonymous users are End users. They only have access to the UserUI and get all information displayed on the map.

5.4 System implementation

After analysing situations and had a decision on the design, the actual implementation process of the project has been started. All features were implemented using Laravel Framework built-in utilities and several third-party libraries. In this chapter, all core features of the application are described in every detail to give the reader an overview of the product.

5.4.1 Features available for Admin Users and SuperAdmin Users

- Company X can register users and provide them credentials to login to the platform. Users can also logout when needed. The login interface is shown in Figure 27.



The image shows a login form interface. At the top left of the form area, the word "Login" is displayed. Below this, there are two input fields. The first is labeled "E-Mail Address" and the second is labeled "Password". Below the "Password" field, there is a blue button labeled "Login" and a link labeled "Forgot Your Password?".

Figure 27. Login form interface.

- Registered users can interact with each model of data in different ways, which are **create, view, edit and delete**.
- All users can create new data of events, locations, and realizations by filling a form wizard and submit to the server. On each wizard, there will be suggestion about existed data for users to avoid creating the same instance of data.
- There are two options for users to create a new realization. The first option is filling all information in a completely new form by click on "Create new Realization". The second option is creating from an existed Event/Location by click on "Create new Realization from this Event" or "Create new Realization from this Location", the data

will be automatically filled into the form, respectively. Figures 28 and 29 show the interface of these features.

EVENT DETAIL

Event name
Esimerkki: Concert for kids

Description of Event
Event Detail

Which category the event belong?
Esimerkki: Sports
Note: First letter on capital.

Host name of Event
PHHYKY
Note: First letter on capital. This should be your organization name unless you are an admin.

Next

Figure 28. Interface of form wizard for creating a new event.

Host name of Event
PHHYKY
Note: First letter on capital. This should be your organization name unless you are an admin.

Next

● ● ●

Search	Search	Search
Category 6	Your organization name: PHHYKY	Library Technician
Category 12	Runolfsson, Kuhlman and Walter	Vocational Education Teacher
Category 22	Tremblay-Dach	Brickmason
Category 21	Bins-Hodkiewicz	Soil Conservationist
Category 2	Gutmann-Kemmer	Heat Treating Equipment Operator
Category 7	Upton-Cummings	Dispatcher
Category 20	Yundt-Herman	Coremaking Machine Operator

Figure 29. Suggestion for existed data.

- A specific user will be able to get access to each of other actions based on the organization that the user working for. Every data created by users will also belong to their organization by default. The list of permissions is listed in the Table 1 below (x: allow or o: not allow).

Actions	View	Edit	Delete
Event created by user	X	X	X
Event belongs to user's organization	X	X	X
Event belongs to other organization	O	O	O
Location created by user	X	X	X
Location belongs to user's organization	X	X	X
Location belongs to other organization	X	O	O
Realization created by user	X	X	X
Realization belongs to user's organization	X	X	X
Realization belongs to other organization	O	O	O

Table 1. Actions' permissions for each user.

Interface of actions are presented in Figure 30, 31 and 32 below.

#	Events					
ID	Event name	Description of Event	Category	Specialist	Host Organization	Created by
25	Deserunt doloribus.	Perspiciatis optio fuga accusamus error aspernatur optio consequuntur. Ut eum qui esse facere odio. Doloribus cum est temporibus magni rem.	Category 12	Agricultural Inspector, Library Technician, Soil Conservationist, Dispatcher, Coremaking Machine Operator	Runofsdottir Group	User Account
24	Ea vel vitae minus.	Natus fuga dignissimos debitis eum voluptatem ut eum. Ea excepturi dolorum quae nisi dolores tempore delectus optio. Perferendis cupiditate nemo totam consequatur quia commodi ea eius.	Category 22	University	Runofsdottir Group	Waylon Carter
23	Fugiat ea.	Dolores tenetur labore aperiam minima reprehenderit veniam nostrum. Est delectus quia aliquid aliquid accusantium quasi nihil. Qui magnam quibusdam dolorem itaque voluptas dolorem. Optio sint pariatur necessitatibus officia deserunt provident quam.	Category 20	Vocational Education Teacher	Yundt-Herman	Waylon Carter
22	Esse esse rem.	Consequatur sunt unde eius doloribus et enim eaque et. Nam eum culpa repellendus deserunt necessitatibus molestiae repellendus. Eum corporis error vel voluptatem vel molestiae.	Category 21	Dispatcher, Agricultural Inspector, Vocational Education Teacher, Vocational Education Teacher	PHHYKY	admin

Figure 30. Show list of data.

Location Detail

This Location data belongs to: **Runolfsdottir Group**

ID	Address	Postal Code	City	Host Organization	Created by	Last edited by
15	Heaney Land 43	21525	Ashtonview	PHHYKY	User Account at 2020-11-09 18:18:57	Quanmuito at 2020-11-15 15:16:00

Description of Location

Nihil illo odit voluptas explicabo occaecati sint. Quos tempore sunt sed autem. Tenetur fugit nihil facere rem at magnam velit dolores. Dolores quia nihil ex ea tenetur et sint.

[Create new Location](#)
[Create new Realization from this Location](#)
[Edit Location](#)
[Delete Location](#)

WARNING!

Edit this Location data will affect the following Realizations.

Note: you cannot delete this Location when there are Realizations using this Location.

Associated Realizations

#	Events Detail					
ID	Event name	Category	Organization	Target Group	Starts at	Ends at

Figure 31. Show detail information of data.

LOCATION

Address

Postal Code

City

Host name of Location

Note: First letter on capital. Caution: Host organization of Location can be different from Host organization of Event.

Description of Location

Nihil illo odit voluptas explicabo occaecati sint. Quos tempore sunt sed autem. Tenetur fugit nihil facere rem at magnam velit dolores. Dolores quia nihil ex ea tenetur et sint.

[Next](#)

Figure 32. Show the form to edit a location data.

- All users can access to **“Report”** page where displays details of all realizations so users can transfer data into an Excel sheet. The report interface is shown in Figure 33.

Realizations Report							
Copy all data to Excel!							
On Window: Ctrl + A -> Ctrl + C -> Open Excel -> Ctrl + V.							
On Mac: Cmd + A -> Cmd + C -> Open Excel -> Ctrl + V.							
ID	Event name	Category	Specialist	Event host	Description of Event	Address	Postal Code
50	Corporis magnam.	Category 6	Soil Conservationist, Dispatcher	Upton-Cummings	Voluptatem recusandae nulla tempore qui sapiente. Vel nam voluptas nesciunt sit rem. Nisi et eaque tenetur eveniet autem pariatur cum necessitatibus. Quas pariatur quo porro accusantium hic earum maxime.	Emery Ferry 175	22902
49	Quidem quasi qui.	Category 24	Brickmason, Agricultural Inspector, Dispatcher	Runolfsson, Kuhlman and Walter	Dolorum sint modi quam unde debitis et. Saepe blanditiis repellat sed labore. Sed odit corrupti sint qui nisi fugit harum consequuntur. Minus eligendi quo voluptatem quia tempora.	Desiree Plains 163	86295
48	Fugiat ea.	Category 20	Vocational Education Teacher	Yundt-Herman	Dolores tenetur labore aperiam minima reprehenderit veniam nostrum. Est delectus quia aliquid aliquid accusantium quasi nihil. Qui magnam quibusdam	Aaliyah Summit 99	63121

Figure 33. Report page interface (image only covers half of the page).

5.4.2 Features available for SuperAdmin users

- SuperAdmin users can register a new Admin user by submitting a specific form to the server. Below in Figure 34 is the form suggestion about the organization of the new user to avoid creating a similar instance.

Register

Name

E-Mail Address

Organization

Password

Confirm Password

[Register](#)

Search available organization names.

Your organization name: PHHYKY

Runolfsson, Kuhlman and Walter

Tremblay-Dach

Bins-Hodkiewicz

Gutmann-Kemmer

Figure 34. Form for register new Admin user.

- SuperAdmin users can view the list of all users and edit information as well as view the statistic of other users for example: promote an Admin user to SuperAdmin user (Figure 35 and 36).

#	Users list						
ID	Name	E-Mail Address	Role	Organization	Registered at	Verify	Status
	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>
2	Quanmuito	tranleminhquan1102@gmail.com	admin	PHHYKY	2020-11-09	VERIFIED	ACTIVE
3	Admin Account	adminaccount@example.com	admin	PHHYKY	2020-11-09	VERIFIED	ACTIVE
4	User Account	useraccount@example.com	normal	Runofsdottir Group	2020-11-09	VERIFIED	ACTIVE
8	Dayne Monahan	slesch@example.net	normal	Runofsdottir Group	2020-11-09	VERIFIED	ACTIVE
5	Disabled Account	disabledaccount@example.com	normal	Zieme Inc	2020-11-09	VERIFIED	BANNED
6	Waylon Carter	cathryn73@example.net	normal	Yundt-Herman	2020-11-09	VERIFIED	ACTIVE
7	Mr. Joany Mitchell	harry96@example.net	normal	Tremblay-Dach	2020-11-09	VERIFIED	ACTIVE

Figure 35. List of all users.

User Information

Name

E-Mail Address

Organization

Role

This User was authorized by: admin at 2020-11-09 18:18:56
 User status: ACTIVE

User statistic

#	Events	Locations	Realizations
Created	5	2	10
Edited	5	2	10

Figure 36. Edit a user's information.

5.4.3 Features available for normal and anonymous users

- Normal and anonymous users can only get access to the EnduserUI to see available events that going to occur. Users can also select a specific range of time or location to narrow down the number of displayed events (Figure 36).

Päivämäärä: Alku

Loppu

Kunta:

Palveluluokka:

Valitse ammatit

Haku	Tyhjennä	Nimi	Palveluluokka	Ammattilainen paikalla	Kaupunki	Osoite	Päivämäärä	Kuvaus
		Corporis magnam.	Category 6	Soil Conservationist, Dispatcher	South Ryan	Emery Ferry 175, 22902 South Ryan	1970-01-01 00:00:02	Voluptatem recusandae nulla tempore qui sapiente. Vel nam voluptas nesciunt sit rem. Nisi et eaque tenetur eveniet autem pariatur cum necessitatibus. Quas pariatur quo porro accusantium hic earum maxime. Et inventore mollitia inventore ea magni aliquid aut qui. Ea qui provident ut incidunt culpa possimus quam. Voluptas consequatur nihil possimus possimus quibusdam saepe fugit ratione.
		Quidem quasi qui.	Category 24	Brickmason, Agricultural Inspector, Dispatcher	West Ambertown	Desiree Plains 163, 86295 West Ambertown	1970-01-01 00:00:01	Dolorum sint modi quam unde debitis et. Saepe blanditis repellat sed labore. Sed odit corrupti sint qui nisi fugit harum consequuntur. Minus eligendi quo voluptatem qua tempora. Quo quaerat eveniet nulla vero labore optio. Eum hic voluptas natus reiciendis et illum ut. Dolorum sequi qui sequi dolores explicabo est qui. Tenetur omnis nobis sed nesciunt.
								Dolores tenetur labore aperiam minima reprehenderit veniam nostrum.

Figure 36. Selection and list of available events.

- User can see the location of events as pinned points on the map for better visualization. Figure 37 is an example illustration for this feature.



Figure 37 Map to display events as pinned points.

5.4.4 Finalization

Finalizing stage includes clean up the project's interface, make instruction page and implement locale for languages. By using HTML, CSS and JavaScript, the interface of the application will be implemented with additional functions and responsive scale in order to allow users to access from different types of devices such as personal computers, laptops and mobile phones. Instruction page includes explanations on functions and how to use the application which displayed on Figure 38.

Functions explain

Buttons	Functions explain	Note/Caution
Create new Event Create new Location Create new Realization	Create a NEW Event/Location/Realization data.	Check existed Events/Locations/Realizations before create a new one to prevent duplicate in the database.
Create new Realization from this Event Create new Realization from this Location	Create a NEW Realization FROM an existed Event/Location.	Recommend to use for create a new Realization.
Edit Event Edit Location Edit Realization	Edit selected Event/Location/Realization data.	Caution: Edit an Event/Location data will affect all Realizations that were made from this Event/Location. Carefully consider before using this function.
Delete Event Delete Location Delete Realization	Delete selected Event/Location/Realization data.	You cannot delete the selected Event/Location data as long as there are Realizations that using data from that Event/Location. You have to delete all Realizations that associate with that Event/Location before delete that data.

Figure 38 Instruction for using the application.

Localization is also implemented so that users can choose display languages for the interface. In the final stage, the application support two languages, which are English and Finnish, in the future, more language can be added to the application so that the product can be accessible for users with different nationalities.

5.4.5 Future maintenance and support

For future maintenance, technicians are required to have experience with the following technologies to be able to handle the application:

- PHP 7.2, 7.4
- Laravel framework 7.x and below
- MySQL database
- WordPress

6 Customer experience survey

6.1 Description

The technology was surveyed in a tech company with 25 employees. Of which, 21 answered the questionnaire. The survey was conducted using Google Doc Forms for online survey and entailed four spectrum-based multiple-choice questions along with an optional supplementary recommendation box. Each of these four questions represents a category of the technology, namely clarity of instructions, website responsiveness and interactivity, ease of gathering information, ease of creating new data through forms.

The answer rate for the four multiple-choice questions was 100%, while the suggestion box received no suggestions across the 21 answers.

6.2 Result

The study indicated a relatively high approval rating for the product, which will be summarized in Table 2 below.

	The instruction was very detail and easy to understand.	The website was very responsive and easy to interact.	How easy was it to find all information on the website?	How easy was it to create new data through forms?
1	3	4	4	4
2	4	3	4	5
3	3	5	3	4
4	4	4	4	4
5	3	4	3	3
6	4	3	3	4
7	4	4	4	4
8	3	5	5	4
9	3	3	4	5
10	5	4	4	5
11	4	3	5	5
12	5	4	3	4
13	4	5	3	4
14	5	4	3	5
15	5	3	4	5
16	4	4	3	5
17	5	3	4	2
18	4	4	4	3
19	4	3	4	5

20	4	3	5	4
21	5	4	5	3

Table 2. Spreadsheet of survey result.

From the colour-coded spreadsheet displayed in Table 2, it is apparent that the technology encountered no negative feedback throughout the entire survey process. The lowest rating for the technology was average at 2 points. However, this was only one out of the 21 responses in the 4th question. The majority of the respondents gave the technology high scores, ranging from 3 (good) to 5 (very good) on all questions, with the highest percentage of responses at 4 points.

Question 1: The instruction was very detail and easy to understand. The question's result is illustrated under the form of a pie chart in Figure 39 below.

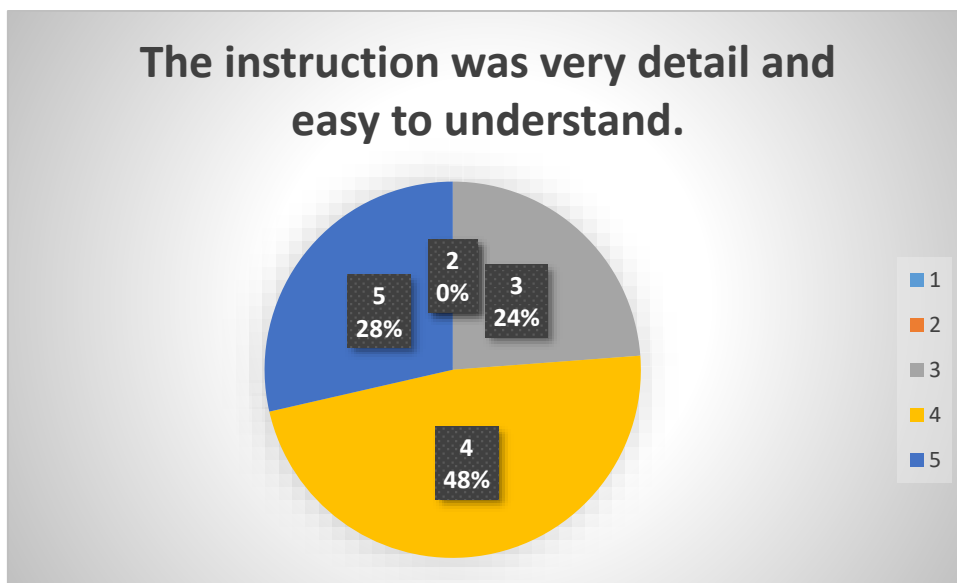


Figure 39. Question 1's result.

For the first category of feature, the instructions were rated as very easy to understand from all respondents, with the lowest rating out of 21 responses at a 3 (good). The majority of the respondents gave this category a 4 out of 5 (48%), and another 28% also gave it a 5 (very good).

The average score for this category is 4.05, which is the second highest in all 4 aspects.

Question 2: The website was very responsive and easy to interact. The question's result is illustrated under the form of a pie chart in Figure 40 below.

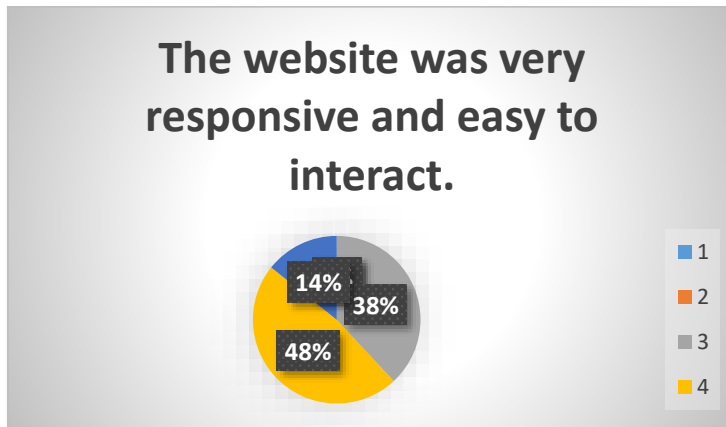


Figure 40. Question 2's result.

The majority of the respondents agreed that the website's response time and interactivity is good. This fact could be observed from the 48% rating (10 votes) at 4 points, and 38% rating (8 votes) at 3 points.

The average score for this category is 3.76, which is a very high score for this category, despite it being the lowest average for all four categories.

Question 3: How easy was it to find all information on the website. The question's result is illustrated under the form of a pie chart in Figure 41 below.

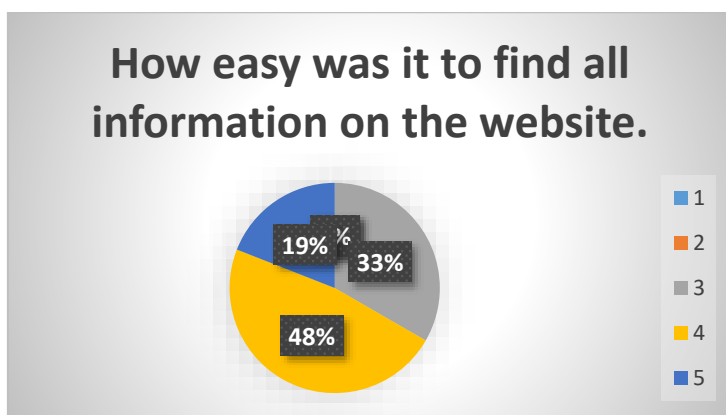


Figure 41. Question 3's result.

Question 3 determines the ease at which respondents were able to collect information on the site. The majority of the answers gave this aspect 4 points (48%), followed by 3 points (33%). No negative was given to this aspect either, much like the two previous questions.

The average score for this category is 3.86, which is the second lowest average out of the four categories. However, this is still miles above the average point.

Question 4: How easy was it to create new data through forms? The question's result is illustrated under the form of a pie chart in Figure 42 below.

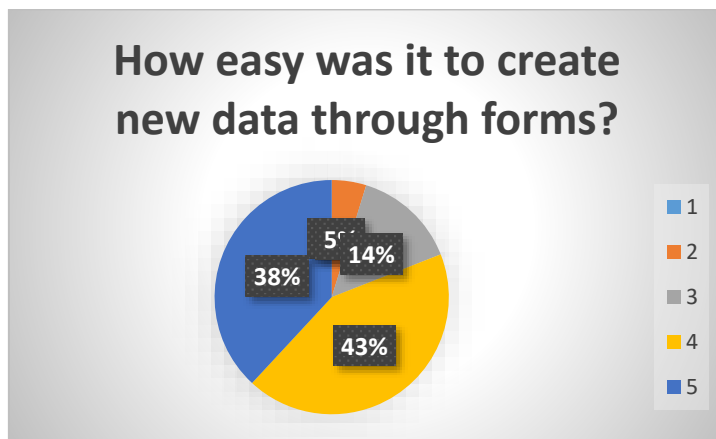


Figure 42. Question 4's result.

The final question offered a look at the ease at which the respondents could create new data through forms. This is where the technology was rated highest among the four categories. 38% of the respondents (8 votes) gave this aspect a 5, which is highest amongst all other categories in terms of 5-star votes. This is accompanied by a 43% rating at a 4. However, there was also a response that gave a 2 point (average) for this aspect of the technology. This proves to be a challenge, as there were no suggestions in the suggestion box, meaning that there is currently no way to find out what difficulties or issues this particular respondent encountered while testing the technology for future improvements.

The average score for this category is 4.14, the highest average out of the 4 categories.

7 Summary

The thesis examines event scheduling and management, as well as the architecture and functionality that could be deployed into production. To sum up this case, the findings of the studies include answers to the research question and sub-questions posed in Chapter 1. In terms of data collection, a mixture of qualitative and quantitative analysis approaches was selected. The thesis project applies the concept of System development life cycle which includes the following phases: planning, analysis, design and implementation. Chapter 2 described these phases in detail.

The research question and five sub-questions were formulated in Chapter 1. The research question is:

What is the most effective way to finalize an existing code base and deploy it into production so that Company X can maximize the benefit of the project?

The following sub-questions are:

- What is this system?
- What are the benefits that the system can contribute to a company's business?
- What functions does it have and what can the functions do?
- What tools and technology should be used to finalize the project in the most effective way?
- When will the application come into production?

Based on the final output of the thesis work, the answer for the first two questions is: the system is a solution for a data management system specified for events scheduling that brings benefits to the case company. These benefits can be stated as the solution can provide the case company:

- A map for everyone to browse upcoming events, filter and search for a suitable event.
- An interface for company X's registered partners to add, modify events and event details. And data from this interface is connected and synchronized with the event map.
- Company X has control over the system, which includes its registered partners and all the information displayed to everyone.
- A database that can store all information including users and events' details.

For the third question, the application allows users from Company X and their partner organizations to manage events data which can be named as create, edit, store, and delete

events data. It also helps end-users which are people in families to easily find suitable events for their children or their elders. The major answer for the fourth question is described detail in chapter 3. All used tools and technologies are modern and up to date so they can provide the best performance and contribute benefits for the application as well as the case company. For the last sub-question, the whole finalizing process to this point of the thesis took up to four months, after that, testing and getting feedback step will be executed and the application will be available when it is deployed on a server. Finally, to answer the main research question, based on the sub-questions, the most effective way to finalize an event scheduling and management system is focus on analyse the current situation and stage of the initial project to come up with a clear plan and a good decision on developing process. To do so, the SDLC framework which described in Chapter 2 is the most effective tool to keep the process in the right track. And with the help from modern tools and technologies, the project can produce a good solution to benefit the case company as well as users in the future.

References

Printed sources

Antani, V., Simon, T., Narayan, P. 2016. JavaScript: Moving to ES2015 Learning Path. Birmingham, England: Packt publishing, ltd.

Avison, D., Shah, H. U. 1997. The Information Systems Development Life Cycle: a First Course in Information Systems. Maidenhead: McGraw-Hill Education.

Carr, D. & Gray, M. 2018. Beginning PHP: Master the Latest Features of PHP 7 and Fully Embrace Modern PHP Development. First Edition. Packt Publishing.

Chacon, S., Straub, B. 2014. Second Edition. Pro Git. Berkley, CA: Apress.

Cleland, D. I., King, W. R. 1997. The Evolution of the Systems Development Life Cycle: An Information Systems Perspective. Project Management Handbook. Second Edition. John Wiley & Sons, Inc., pp. 251-272.

Denzin, N. K., Lincoln, Y. S. 2018. The SAGE Handbook of Qualitative Research. Fifth Edition. Thousand Oaks: California: SAGE Publications, Inc.

Dillman, D. A., Smyth, J. D., Christian, L. M. 2014. Internet, Phone, Mail and Mixed-Modes: The tailored design method. Fourth Edition. New Jersey: John Wiley & Sons, Inc.

Fraser, S. R. G. 2002. Real-World ASP.NET: Building a Content Management System. New York: Apress.

Karol, K. 2019. WordPress 5 Complete, Seventh Edition. Birmingham: Packt Publishing.

Kurniawan, B. 2015. HTML: A beginner's tutorial. Vancouver, Canada: Brainy Software.

Locke, E. A. 2007. The Case for Inductive Theory Building. Journal of Management vol. 33 (6). New York: Sage Publications, pp. 867-890.

Murray, R. T. 2003. Blending Qualitative and Quantitative Research Methods in Theses and Dissertations. Thousand Oaks, California: Corwin Press, Inc. A Sage Publications Company.

MySQL 2018a. What's New in MySQL 8.0? Retrieved on 11th April 2021.

MySQL 2018b. A Guide to MySQL and PCI Data Security Standard Compliance. Retrieved on 11th April 2021.

MySQL 2018c. MySQL Enterprise High Availability. Retrieved on 12th April 2021.

MySQL 2018d. MySQL Document Store: Top 10 Reasons. Retrieved on 12th April 2021.

Nola, R., Sankey, H. 2007. Theories of Scientific Method. Stocksfield: Acumen Publishing Limited.

Pouncey, I., York, R. 2011. Beginning CSS: Cascading Style Sheets for Web Design. Third Edition. Wrox.

Pouncey, I., York, R. 2011. Beginning CSS: Cascading Style Sheets for web design. Book. John Wiley & Sons, Inc., Indianapolis, US.

Pyzdek, T., Keller, P. 2013. The Handbook for Quality Management. Second Edition. United States: McGraw-Hill Companies, Inc.

Raymond, L.C. 2012. Getting it right: the essential elements of a dissertation. Lanham: Rowman & Littlefield Publishers.

Robertson, J., Robertson, S. 2012. Mastering the Requirements Process: Getting Requirements Right. Third Edition. Boston, United States: Addison-Wesley Professional.

Stauffer, M. 2019. Laravel: Up & Running: A Framework for Building Modern PHP Apps. O'Reilly Media, Inc.

Tatroe, K., MacIntyre, P. 2020. Programming PHP: Creating Dynamic Web Pages, Fourth Edition. United States: O'Reilly Media, Inc.

White, B. M. 2020. Web Development for Beginners.

Electronic sources

Harjulan Setlementti ry 2021. Retrieved on 15th April 2021. Available at: <https://harjulan.fi/harjula-sali/perhe-ja-lasten-liikunta/>

Iqbal, M. 2021. Business of Apps. Zoom Revenue and Usage Statistics 2020. Retrieved on 15th April 2021. Available at: <https://www.businessofapps.com/data/zoom-statistics/>

Laravel 2021. Website. Retrieved on 10th April 2021. Available at <https://laravel.com/>.

Möysän Nuorisoseura ry 2021. Retrieved on 15th April 2021. Available at <https://www.mns.fi/ilmoittautuminen/>

PHHYKY - Päijät-Hämeen hyvinvointiyhtymä 2021. Retrieved on 7th April 2021. Available at: <https://www.facebook.com/phhyky/>

Rosario, N. D. 2017. System analysis and design/business process. Retrieved on 25th March 2021. Available at: <https://sites.google.com/site/ustcomp2com/sdlc-stages>.

Techronus. 2018. Why go with PHP Laravel Framework? Retrieved on 10th April 2021. Available at: <https://www.techronus.com/blog/go-php-laravel-framework/>.

VMware, Inc 2021. Virtual machine. Retrieved on 14th April 2021. Available at: <https://www.vmware.com/topics/glossary/content/virtual-machine>.

WordPress.org 2021. Features. Retrieved on 1st April 2021. Available at: <https://wordpress.org/about/features/>

Appendix 1. Survey for “User experience on product”

Description:

The survey is an assessment of user's experience shows by score on scale of 5:

- 0: Worst
- 1: Bad
- 2: Average
- 3: Good
- 5: Very good

Questions:

1. The instruction was very detail and easy to understand?
2. The website was very responsive and easy to interact?
3. How easy was it to find all information on the website?
4. How easy was it to create new data through forms?
5. Is there anything that can be improve? (Optional)