

qASSIGNMENT 01 FRONT SHEET

Qualification	BTEC Level 5 HND Diploma in Computing		
Unit number and title	Unit 09: Software Development Life Cycle		
Submission date	08/02/2023	Date Received 1st submission	
Re-submission Date	15/02/2023	Date Received 2nd submission	
Student Name	Nguyễn hải Việt	Student ID	GCD210136
Class	GCD1101	Assessor name	Phạm Thanh Sơn
Student declaration I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice.			
		Student's signature	

Grading grid

P1	P2	P3	P4	M1	M2	D1	D2

☐ **Summative Feedback:**

☐ **Resubmission Feedback:**

Grade:

Assessor Signature:

Date:

Internal Verifier's Comments:

Signature & Date:

Contents

I. Describe two iterative and two sequential software lifecycle models(P1).....	4
1. What is SDLC	4
2. How SDLC work.....	6
3. Type mode	6
A. Waterfall.....	6
B. V-model	9
C. Prototyping.....	10
D. Scrum.....	12
E. Spiral.....	13
4. What is the suitable model for the project, and explain why?	15
5. Discuss the suitability of each of the SDLC models for the Project.....	15
6. Discuss the merits of applying the waterfall model to a large software development Project.....	16
II. Explain how risk is managed in the Spiral lifecycle model. (P2).....	16
1. What is risk management	16
2. Why is risk management is important.....	16
3. Five step of the Risk Managenent	16
4. Risk management matrix.....	17
III. Explain the purpose of a feasibility report. (P3).....	18
1. What is feasibility report	18
2. What is the purpose of a feasibility report?	18
3. Discuss how the three feasibility criteria	19

4. Discuss whether the project is feasible or not	19
IV. Describe how technical solutions can be compared. (P4).....	20
a. Magento	20
b. Shopify	20
*References:	22
Figure 1: SDLC.....	5
Figure 2: SDLC work.....	6
Figure 3: waterfall mode	7
Figure 4: V-model	9
Figure 5: Prototyping.....	11
Figure 6: Scrum.....	13
Figure 7: Spiral.....	14

I. Describe two iterative and two sequential software lifecycle models(P1)

1. What is SDLC

SDLC stands for Software Development Life Cycle, which refers to the process of developing software, from the initial conceptualization of an idea to the delivery of a final product. The SDLC provides a framework for organizing and managing the various stages

involved in software development, ensuring that the end product meets the needs of the stakeholders and is delivered on time, within budget, and to the required quality standards.

The SDLC typically includes the following stages:

- Requirements gathering and analysis: In this stage, the project requirements are gathered, analyzed, and documented.
- Design: In this stage, the software design is developed and documented, including the architecture and interfaces.
- Implementation: In this stage, the code is written and tested.
- Testing: In this stage, the software is tested and any bugs or issues are identified and fixed.
- Deployment: In this stage, the software is deployed to production and made available to end-users.
- Maintenance: In this stage, the software is maintained and updated as necessary to address any ongoing issues or to add new features.

There are various SDLC models that organizations can choose from, including the Waterfall model, the Agile model, and the Iterative model. Each model has its own strengths and weaknesses, and the choice of model will depend on the specific requirements and constraints of the project. Regardless of the specific SDLC model used, the goal is to ensure a consistent and structured approach to software development, enabling organizations to deliver high-quality software products.

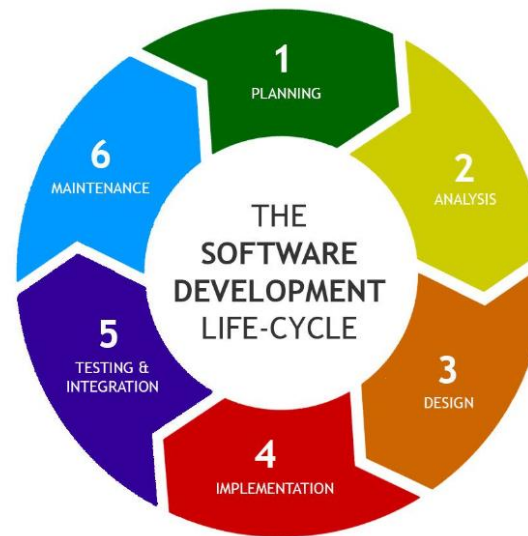


Figure 1: SDLC

2. How SDLC work

SDLC works by reducing software development costs while both raising quality and speeding up production. By adhering to a strategy that eliminates the common hazards of software development projects, SDLC accomplishes these seemingly incompatible aims. This approach begins by looking for flaws in the current systems.

The requirements for the new system are then defined. After that, it goes through the processes of analysis, planning, design, development, testing, and deployment to actually construct the program. By foreseeing expensive errors like forgetting to get input from the client or end-user, SDLC may reduce the need for further effort and post-hoc corrections.

It's crucial to understand that the testing stage is given a lot of attention. You must guarantee code quality at every cycle since the SDLC is a recurring approach. Many businesses prefer to put little effort into testing, despite the fact that doing so might save them a lot of time, money, and rework. Be wise and create the proper kinds of exams.

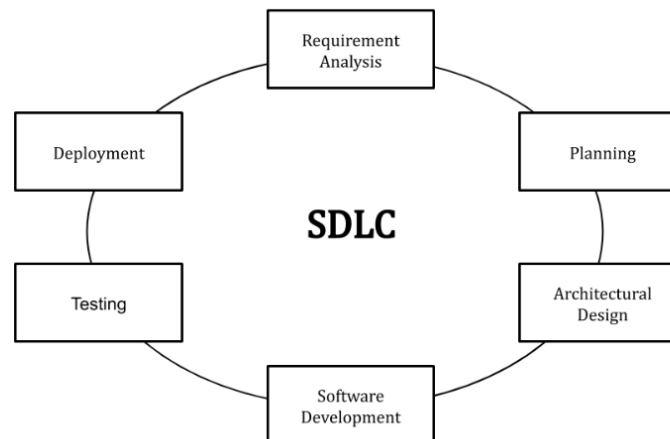


Figure 2: SDLC work

3. Type mode

A. Waterfall

a. Definition

Known as one of the easiest project management models to understand today, the Waterfall model is a project management methodology based on a sequential and sequential design process. In the Waterfall model, the phases of the project are executed one after the other. A new phase is only started when the previous one has been completed.

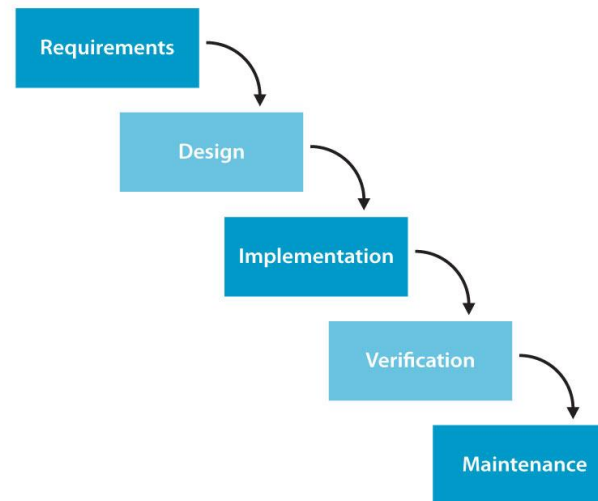


Figure 3: waterfall mode

*Different Phases of Waterfall Model in Software Engineering:

Different phases	Activities performed in each stage
Design Stage	<ul style="list-style-type: none"> Plan the programming language, for example Java, PHP, .net or databases like Oracle, MySQL, etc. Or other high-level technical details of the project
Built Stage	After the design stage, it is the built stage, which is nothing but coding

	the software
Test Stage	In this phase, you test the software to verify that it is built as per the specifications given by the client.
Deployment stage	Deploy the application in the respective environment
Maintenance stage	Once your system is ready to use, you may later require to change the code as per customer request

b. Stages

- Request: Finding requirements to the project such as determining the business needs of the project, user requirements for the product or associated risks
- Design: a design for the product must address all requirements, constraints, and design goals. It should describe exactly how the logic of the system mentioned in the analysis will be implemented.
- Execute (build): Products are built to support design. Sometimes, the product is built in units for testing and subsequent integration.
- Test: Parts of the product are inspected. If necessary, they will be integrated together for testing. The entire system is tested to find defects and ensure design goals.
- Deployment: For IT projects, the product is deployed into the environment so that users can start using it. For a construction project, the implementation phase is when the building is completely ready for occupancy.
- Maintenance: Is a short monitoring period. In which the project team solves customer problems. For software projects, this usually means releasing patches and updates to fix problems. In other projects, environmental adjustments are made to address the issue. For example, optimizing air conditioning in a new building.

c. Advantages of the Waterfall model

- Adapts well to flexible groups
- Imposing a structured organization
- Allows for early design changes

- Suitable for landmark-oriented projects

d. Disadvantages of Waterfall model

- It is not an ideal model for a large size project.
- If the requirement is not clear from the start, it is a less efficient method.
- It is very difficult to move back to the previous stage to change.
- Testing begins when development is finished. Therefore, it has a high risk of bugs being found after the development phase, and it is very expensive to fix the bugs.

B. V-model

a. Definition

Vmodel is an extension of the waterfall model. Not like the waterfall model. In the V model, corresponding to a testing phase is a software development phase, testing in the V-model is performed in parallel with the software development cycle.

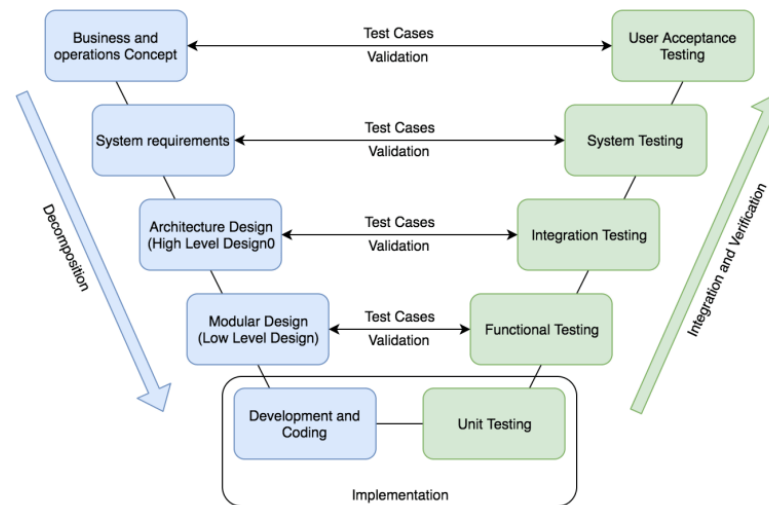


Figure 4: V-model

b. Stages

- Request: At this stage only collect requirements

- Design: Plan to choose programming languages like JAVA, PHP... databases like Oracle, My SQL....
- Execute (build): At this stage, code implementation will be implemented
- Test: Perform testing to verify that it is built to the specifications provided by the customer
- Deployment: Implement application deployment in a real environment
- Maintenance: When your system is ready to use. You can perform maintenance upgrades and code changes according to customer requirements

c. Advantages of the V – model

- Simple, friendly and time saving
- Have specific plans and activities for the testing process
- The probability of success is higher than the waterfall model
- Quickly detect errors and find the cause from the beginning

d. Disadvantages of the V – model

- There is still rigidity, little flexibility (after each step must check and confirm)
- Failure to meet service requirements is to develop and sell products at the same time
- If the project requirements are simple, the verification process will take a lot of time
- Have to go back to the first steps, update the documentation if there is a technical change midway
- The product of the project is only released when the steps are completed, there is no prototype at the beginning

C. Prototyping

a. Definition

Prototypes are early samples, models, or releases of products built to test a concept or process. There are many contexts in which semantics can be used, for example, in design, electronics, and software programming. Generally, prototypes are used by system analysts and users to improve the precision of a new design.

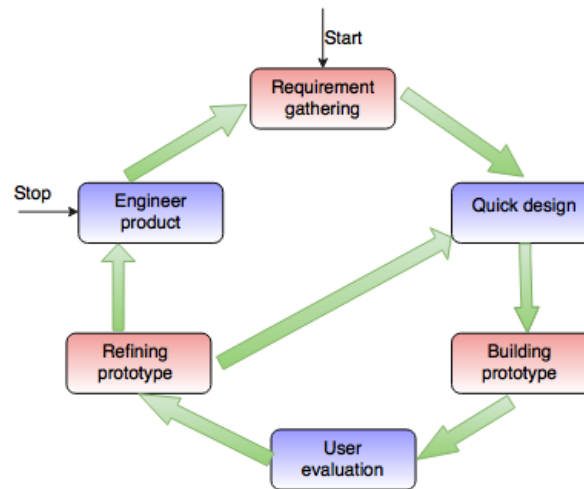


Fig.- Prototype Model

Figure 5: Prototyping

b. Stages

- A preliminary project plan is developed
- An partial high-level paper model is created
- The model is source for a partial requirements specification
- A prototype is built with basic and critical attributes
- The designer builds
 - the database
 - user interface
 - algorithmic functions
- The designer demonstrates the prototype, the user evaluates for problems and suggests improvements.
- This loop continues until the user is satisfied

c. Advantages of prototyping

- Customers can “see” the system requirements as they are being gathered

- Developers learn from customers
- A more accurate end product
- Unexpected requirements accommodated
- Allows for flexible design and development
- Steady, visible signs of progress produced
- Interaction with the prototype stimulates awareness of additional needed functionality

d. Disadvantages of prototyping

- Tendency to abandon structured program development for “code-and-fix” development
- Bad reputation for “quick-and-dirty” methods
- Overall maintainability may be overlooked
- The customer may want the prototype delivered.
- Process may continue forever (scope creep)

D. Scrum

a. Definition

Scrum is a management framework used by teams to organize themselves and work towards a common goal. This framework describes a range of meetings, tools, and roles for effective project delivery. Just as a sports team rehearses for the big game, practicing Scrum enables the team to self-manage, learn from experience, and adapt to change. Software development teams use Scrum to solve complex problems economically and sustainably.

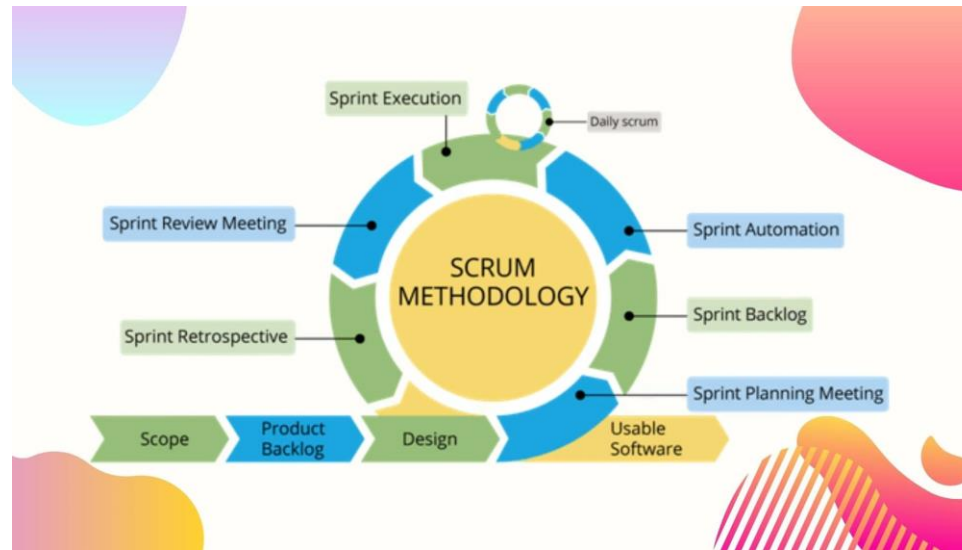


Figure 6: Scrum

b. Stages

- Sprint Cycle Planning
- Sprint
- Daily Scrum meeting or standing meeting
- Sprint Cycle Assessment
- A look back at the Sprint cycle

c. Advantage of scrum

- The ability to maintain quality in difficult situations
- Increase return on invested capital
- Teams feel more comfortable, work more productive
- Matching metrics improve estimation results

E. Spiral

a. Definition

A risk-based strategy for software development is the spiral model. It combines waterfall and iterative methodologies. The Spiral Model helps software projects based on distinct risk patterns incorporate features of software development from several process models, leading to an effective development process.

According to the SDLC Spiral model, the development process starts out with a small number of requirements and advances according to those needs through each development phase. The software engineering team spirals in features for the additional demand up until the program is prepared for production.

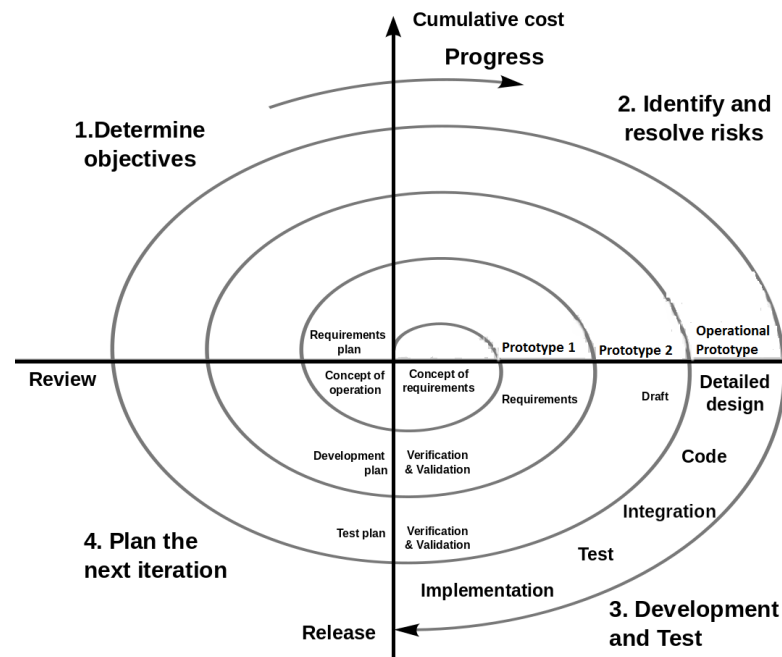


Figure 7: Spiral

b. Advantage of spiral

- You may expand the functionality or make changes in the future.
- Cost estimation is easy since the prototype is developed in manageable pieces.

- Continuous or recurring improvement is advantageous for risk management.
- Spiral development adds things gradually and quickly.

c. Disadvantages of spiral

- There is a danger that you won't complete it on schedule or on budget.
- Large projects benefit most from spiral development, albeit it does need proficiency in risk assessment.
- For proper operation, the spiral model protocol must be adhered to.
- Due to the inclusion of intermediary steps, documentation is more thorough.
- Because spiral software development may be costly, it is not advised for minor applications.

4. What is the suitable model for the project, and explain why?

Our company received a big project named TUNE SOURCE. The name of this project is also the name of the company that asked our company to create an application that could meet their requirements. Since this is a big project, our company has received a great investment from the very beginning, besides there are many experienced people who will be involved in this project and this project must absolutely be avoided. Make big mistakes that lead to loss of time, prestige and money. Summarizing all the above reasons, we have decided to use the spiral model for this project, the benefits of this model will help a lot when we implement this project. The process of applying this method can take place within 6 months with 4 main activities: planning, risk analysis, prototyping, component evaluation. This software development model requires a lot of interaction between the technical team and the engineer, so the intervention of the customer will also be significantly more. They have the right to participate in the ideation phase, assessing the quality of work through each prototype supervision.

5. Discuss the suitability of each of the SDLC models for the Project

Factors	Unclear User Requirement	Unfamiliar Technology	Complex System	Reliable system	Short Time Schedule	Strong Project Management	Cost limitation	Visibility of Stakeholders	Skills limitation	Documentation	Component reusability
Waterfall	Poor	Poor	Good	Good	Poor	Excellent	Poor	Good	Good	Excellent	Excellent
Prototyping	Good	Excellent	Excellent	Poor	Good	Excellent	Poor	Excellent	Poor	Good	Poor
Spiral	Excellent	Excellent	Excellent	Excellent	Poor	Excellent	Poor	Excellent	Poor	Good	Poor
Scrum	Excellent	Poor	Poor	Good	Excellent	Excellent	Excellent	Excellent	Poor	Poor	Poor
Iterative	Good	Good	Good	Good	Excellent	Excellent	Excellent	Good	Good	Excellent	Excellent

and Incremental											
--------------------	--	--	--	--	--	--	--	--	--	--	--

6. Discuss the merits of applying the waterfall model to a large software development Project

First, this model is only suitable for small or short projects so that it can be deployed easily. Secondly, with this model, the project must be intuitive and easy to understand because with the nature of this project once done, it cannot be changed. Third, as well as the second reason, it is difficult for the project to adapt to changes from the requester. And finally about the seamlessness, when implementing projects using this model, the visualization is low and the delivery progress is slow because at the end of the cycle the user can see and use the product. The product cannot be viewed while it is in progress.

II. Explain how risk is managed in the Spiral lifecycle model. (P2)

1. What is risk management

Identification, assessment, and control of financial, legal, strategic, and security threats to the assets and profits of an organization constitute the process of risk management. These dangers or hazards might be caused by a broad range of things, such as monetary instability, legal obligations, poor strategic management, mishaps, and natural calamities.

Your company may only see a little effect from an unexpected incident, such as a slight increase in overhead expenditures. In the worst-case situation, though, it may be disastrous and have severe repercussions, including a heavy financial load or possibly the liquidation of your company.

2. Why is risk management is important

Any business or organization when operating must manage the risks that organizations face, which can include:

- Saving the organization's limited resources: Time, assets, people, etc.
- Protect the image of the business in the society.
- Avoid harm to the organization when operating.
- Avoid violating legal regulations when doing business.
- Protect resources and the environment.

3. Five step of the Risk Management

The steps that must be done are outlined in the risk management process. The risk management process, which consists of these five fundamental components, is used to manage risk. Starting with risk identification, it then moves on to risk analysis, risk prioritization, solution implementation, and risk monitoring. Each stage in manual systems requires a significant amount of administration and paperwork.

These are the five categories of risk management:

- Identify the Risk
- Analyze the Risk
- Evaluate or Rank the Risk
- Treat the Risk
- Monitor and Review the Risk

4. Risk management matrix

No	Risk	Level	Cause	Solution
1	Scope and Object	Extreme	This risk will almost double the work for programmers and inevitably increase costs.	To avoid this risk from happening, we need to have a clear plan first. In addition, the programmer must also receive the request clearly to avoid misunderstanding between the customer and the receiver.
2	High cost	High	As a result, the project can be sketchy and error-prone or worse, it won't be able to continue and stop.	understand the requirements, understand the technology to be used, then apply to the project to be able to optimize the most reasonable cost.

3	Technology Employed	Medium	As a result, the completed project will not last long, making maintenance difficult, costly, and difficult for those who will modify it later if the project needs an upgrade and amendments	Identify the technology that the project needs to suit the team's personnel and the company's perception of the technology.
4	project time	High	As a result, when there is not enough time, an incomplete application will be created, many errors will occur, leading to many difficulties in future maintenance, upgrades and bug fixes.	Have a clear and intuitive love, have a specific plan for each part so that you can understand which parts need speeding up and which parts need to spend a lot of time on it

III. Explain the purpose of a feasibility report. (P3)

1. What is feasibility report

A feasibility study is a document that assesses the viability of a group of suggested project pathways or solutions. When writing a feasibility report, the author assesses the viability of several alternatives before making a suggestion for the best one. After that, they give their advice to the firm and provide the feasibility report.

2. What is the purpose of a feasibility report?

Finding out if solutions or project pathways are feasible can help you choose the best course of action. The purpose of the feasibility report is to explain several ways to a project or issue and assist readers in determining the viability of each strategy. Readers may choose to follow the study's

advice for the best course of action based on the assessment presented in the report. Making the best options for projects and issues may be made by businesses with the aid of this detailed review of various techniques.

3. Discuss how the three feasibility criteria

a. Technical Feasibility

The technological resources that the organization has access to are the main focus of this examination. It helps companies in determining if the technical resources are enough and whether the technical team has the skills necessary to turn concepts into functional systems. Evaluation of the proposed system's hardware, software, and other technical needs is another aspect of technological viability.

b. Economic Feasibility

Before allocating financial resources, this evaluation often includes a cost/benefit analysis of the project to assist businesses examine the feasibility, costs, and advantages related to a project. Additionally, it improves project credibility and acts as an impartial project evaluation, assisting decision-makers in identifying the favorable economic advantages that the proposed project would bring to the business.

c. Legal Feasibility

This evaluation looks at any potential legal infractions of the planned project, including zoning rules, data protection laws, and social media laws. Let's imagine that a company wishes to develop a new office building in a certain area. The organization's preferred site may not be permitted for that kind of operation, according to a feasibility assessment. By realizing that their project was unworkable from the start, that company has just saved a great deal of time and effort.

4. Discuss whether the project is feasible or not

a. Technical feasibility

Nowadays, web technologies have developed a lot, the team behind this project have also used these technologies many times and have achieved a high level of proficiency. Besides, the employees have also worked together many times, so they can easily solve the problems in the project.

b. Economic feasibility

Last year's annual sales were \$40 million with an annual growth of about 3%–5% per year. So this is a company that has a lot of Stable and growing customers. Staff working on this project can be paid \$10 an hour with a team of no more than 10 members. Besides, the additional costs plus the right space and equipment for employees will help them carry out the project more successfully. Therefore, the economic feasibility of this project is feasible

c. Organizational feasibility

For those who came up with the idea for this project, they wanted to be able to raise capital from digital songs or old songs that are now difficult for newcomers to access and buy. For the sponsors, they realize the potential of the project and investment. This program started with the goal of increasing sales by allowing us to sell digital music downloads to customers through in-store kiosks and on our website. For users, there are many customers who need these ideas to be able to buy music by discs that cannot be bought and it is very helpful for them.

IV. Describe how technical solutions can be compared. (P4)

a. Magento

Magento is an e-commerce platform that helps businesses build an effective eCommerce Website system. Magento is created from Zend Framework and applies MVC patterns to manage and store all data.

The Magento platform includes:

- Magento Open Source (formerly known as Magento Community Edition). This is the free version that Magento has set up for users.
- Magento Commerce or the second version of Magento. Magento Commerce provides businesses with On-Premises on-premises solutions (formerly known as Magento Enterprise Edition) or provides businesses with Platform-as-a-Service hosting platforms (formerly known as Platform-as-a-Service). also known as Magento Enterprise Cloud Edition). Magento Commerce is a paid version, often used by large businesses with a minimum price of 24,000 USD per year.

b. Shopify

Shopify is a SaaS (Software as Services) solution for e-commerce that allows you to create an online sales website with full features of shopping cart and checkout, order processing, omnichannel selling, marketing automation, all both integrated in Shopify.

Unlike platforms like Magento or OpenCart, you need to have knowledge of website, hosting or HTML to use it. Shopify is designed for everyone, you don't need any coding skills to get started with Shopify.

Shopify is used by many small and medium-sized businesses (SMEs), as well as more than 10,000 Enterprise customers with the Shopify Plus service plan.

The finding comparison of the three Technologies are presented in the table below, along with the following conclusion:

Factors	Magento	Shopify	The better
Functionality	offering a wide range of features and capabilities, including a flexible product catalog, custom checkout process, and a range of customization options	offering basic e-commerce functionality and a range of templates and plugins	Same
Ease of use	can be more complex to use and may require technical expertise to set up and customize.	ease of use, with a user-friendly interface and drag-and-drop website builder	Shopify is better because it's ease of use and don't need technical expertise
Cost	has a lot of cost-based functionality, making Magento a more cost-effective solution in the long run.	offers a range of subscription plans	Magento is better in the long run
Performance	is highly scalable and can handle large amounts of traffic and product data but	can automatically increase processing speed when a large number of users visit the page.	Magento has better results due to its flexible structure
Security	offers strong security features, including secure payments and data encryption, and is SSL certified	highly appreciate the security as well as the level of information and data safety	Magento is better because it has higher security features and is easy to manage sensitive information and data.

Factor	Weighting factor	Magento		Shopify	
		Score	Weighting score	Score	Weighting score
Development time	5	3	15	3	15
Cost	4	1	4	3	12

Ease of Use	4	3	12	4	16
Scalability	3	3	9	3	9
Performance	3	5	15	3	9
Speed	2	4	8	2	4
Functionality	2	4	8	3	6
Score		23	71	21	71
Total score		94		92	

Conclusion: Based on this comparison, the company may choose to use Magento because it provides a more feature-rich, efficient and secure solution while remaining cost-effective in the long run. However, the final decision will depend on the specific needs and goals of the company as well as the trade-offs between cost, performance, security, and ease of use.

*References:

- Altwater, A. (2021) What is SDLC? understand the software development life cycle, *Stackify*, [online] Available at: <https://stackify.com/what-is-sdlc/#:~:text=How%20the%20SDLC%20Works,pitfalls%20of%20software%20development%20projects>. (Accessed February 8, 2023).
- Simplilearn (2023) What is feasibility study in project management and its types?: Simplilearn, *Simplilearn.com*, Simplilearn, [online] Available at: <https://www.simplilearn.com/feasibility-study-article#:~:text=The%205%20types%20of%20feasibility,Economic%20Feasibility%2C%20and%20Technical%20Feasibility>. (Accessed February 8, 2023).
- Anon (n.d.) What is a feasibility report? (definition and template), [online] Available at: <https://www.indeed.com/career-advice/career-development/feasibility-report> (Accessed February 8, 2023).
- Stokkelien, V. (1976) VI, *Amazon*, Aschehoug, [online] Available at: <https://aws.amazon.com/vi/what-is/scrum/> (Accessed February 8, 2023).
- Thu, H. (2023) V model Trong Kiểm Thử phần mềm LÀ GÌ? Tìm Hiểu Với Ví DỤ SDLC& STLC., *Viblo*, Viblo, [online] Available at: <https://viblo.asia/p/v-model-trong-kiem-thu-phan-mem-la-gi-tim-hieu-voi-vi-du-sdlc-stlc-Qbq5QMEL5D8> (Accessed February 8, 2023).
- Simplilearn (2023) Prototyping in design thinking: Definition, types & benefits: Simplilearn, *Simplilearn.com*, Simplilearn, [online] Available at: <https://www.simplilearn.com/prototyping-in-design-thinking-article> (Accessed February 8, 2023).
-