ASSIGNMENT 1 FRONT SHEET

Qualification	**BTEC Level 5 HND Diploma in Computing**		
Unit number and title	Unit 9: Software Development Life Cycle		
Submission date	19/2/2023	**Date Received 1st submission**	
Re-submission Date		**Date Received 2nd submission**	
Student Name	Vũ tiến duy	**Student ID**	BH00046
Class	It0502	**Assessor name**	Ngo Thi Mai Loan

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	Duy

Grading grid

P1	P2	P3	P4	M1	M2	D1	D2

❒ Summative Feedback:	❒ Resubmission Feedback:

Grade:	Assessor Signature:	Date:

Internal Verifier's Comments:

Signature & Date:

Nội dung

Grading grid 1
There are the various phases of Verification Phase of V-model: 7
There are the various phases of Validation Phase of V-model: 8
Advantage (Pros) of V-Model: 8
Disadvantage (Cons) of V-Model: 9
When to use V-Model? 9
3. RAD model 9
When to use RAD model? 11
Figure 4 13
When to use iterative model? 14
III. Explain how to manage risk in the screw model 14
IV. Explain the purpose of a feasibility reportWhat is project feasibility? 15
What is a feasibility study for? ..... **Error! Bookmark not defined.**
Benefits of feasibility studies ..... **Error! Bookmark not defined.**
Components of the feasibility report: ..... **Error! Bookmark not defined.**
V. Describe how technical solutions can be compared 19
VI. CONCLUSION 22

I. Introduction

SDLC (Software Development Lifecycle) is a process of managing, designing, developing, and deploying software. SDLC plays an important role in software development as it ensures that the software development is done according to qualitystandards and on time, as well as meeting customer requirements. In this ASM, I willintroduce several software development lifecycles, each with its own set of stages, methods, and goals to you.

The Spiral model is a software development lifecycle model that emphasizes the iterativeand

incremental nature of the development process. Risk management plays an importantrole in the Spiral model because it involves identifying and addressing potential risks throughout the development process. I am going to examine how this model identifies and mitigates risks throughout the development process.
Taking into account factors such as technical requirements, resource availability, andfinancial viability are important for determining feasibility of preparing a software development.
In addition, I will describe how technical solutions can be compared in the end of thisASM.

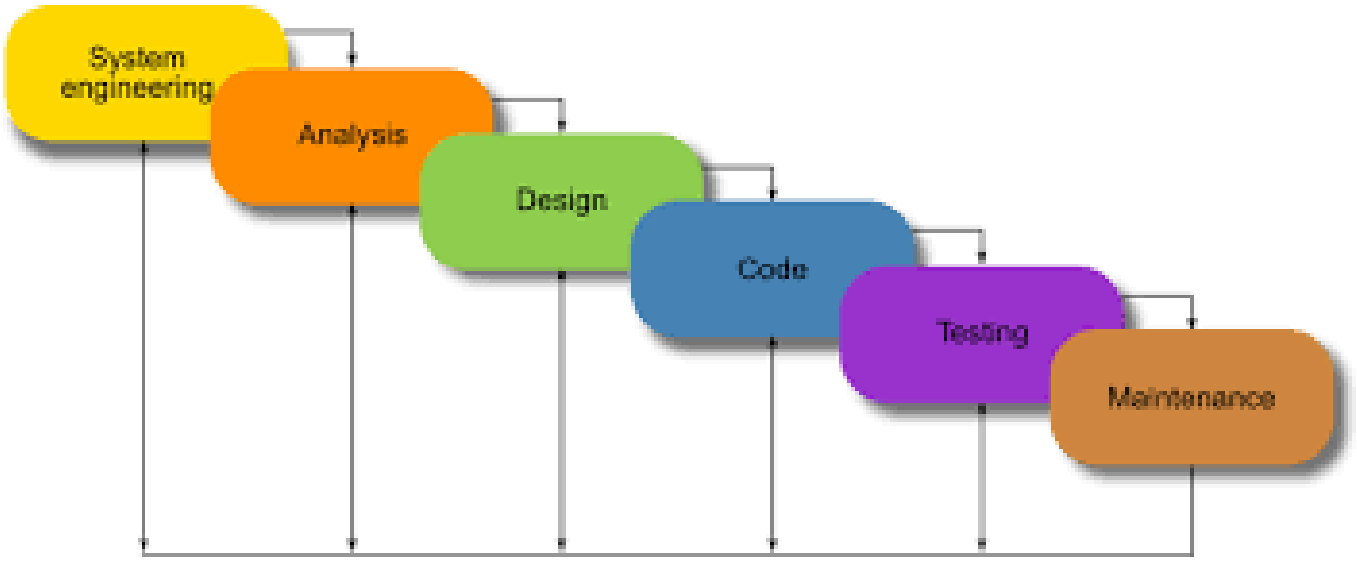
II. Describe two iterative and two sequential software lifecycle models

1. Waterfall model

* Waterfall model definition

Waterfall model is also known as waterfall model. Known as one of the easiest project management models available 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.



* Stages of the Waterfall model

- Requirements Analysis

The team does a search for requirements related to the project.

For example:

+ Determine which business need the project will address

+ User requirements for the product developed by the project

+ Constraints and associated risks.

- Design phase

  The team creates the design for the product to address any design requirements, constraints, and goals. A typical design will be completed as specifically as possible. It will describe exactly how the logic of the system mentioned in the analysis will be implemented.

- Implementation or development phase

  Products are built to support design. Sometimes, the product is built in units for testing and subsequent integration.

- Test phase

  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

  The tested product actually goes into action. 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 phase

  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.

* Advantages of Waterfall model

- Adapts well to flexible groups

- Imposing a structured organization **:**

- Allows for early design changes

- Suitable for landmark-oriented projects

* 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 over. Therefore, it has a high risk of bugs being found after the development phase, and it is very expensive to fix the bugs.

* When to apply the Waterfall model

The application of the Waterfall model is recommended when the implementer knows the requirements of the project best, requires high clarity and stability such as:

- Grasping technology development of technology.

- Eliminate vague and unclear requirements.

- There is a rich amount of development resources and a high level of expertise and technology.

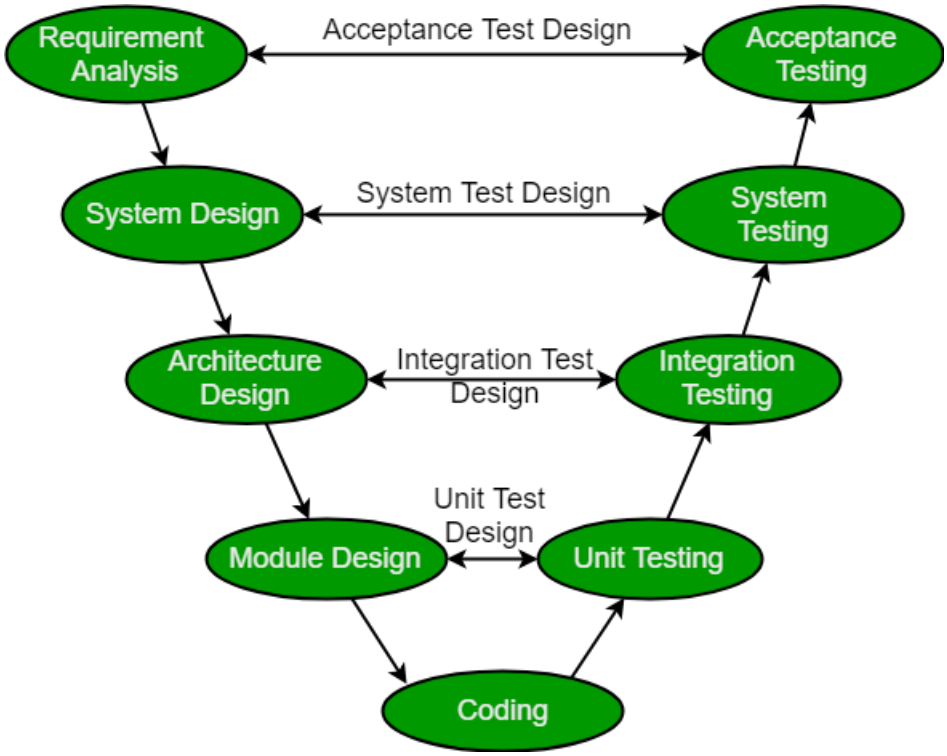
- May be suitable for small, short term project.Through the article, PMA has provided the basic concepts of Waterfall model. The key to this model is to follow the steps outlined. At the same time, the Waterfall model is still very well applied in small-scale and short-term projects.

2. V-Model

Verification: It involves a static analysis method (review) done without executing code. It is the process of evaluation of the product development process to find whether specified requirements meet.

Validation: It involves dynamic analysis method (functional, non-functional), testing is done by executing code. Validation is the process to classify the software after the completion of the development process to determine whether the software meets the customer expectations and requirements.

So V-Model contains Verification phases on one side of the Validation phases on the other side. Verification and Validation process is joined by coding phase in V-shape. Thus it is known as V-Model.



There are the various phases of Verification Phase of V-model:

1. **Business requirement analysis:** This is the first step where product requirements understood from the customer's side. This phase contains detailed communication to understand customer's expectations and exact requirements.
2. **System Design:** In this stage system engineers analyze and interpret the business of the proposed system by studying the user requirements document.
3. **Architecture Design:** The baseline in selecting the architecture is that it should understand all which typically consists of the list of modules, brief functionality of each module, their interface relationships, dependencies, database tables, architecture diagrams, technology detail, etc. The integration testing model is carried out in a particular phase.
4. **Module Design:** In the module design phase, the system breaks down into small modules. The detailed design of the modules is specified, which is known as Low-Level Design
5. **Coding Phase:** After designing, the coding phase is started. Based on the requirements, a suitable programming language is decided. There are some guidelines and standards for coding. Before checking in the repository, the final build is optimized for better performance, and the code goes through many code reviews to check the performance.

There are the various phases of Validation Phase of V-model:

1. **Unit Testing:** In the V-Model, Unit Test Plans (UTPs) are developed during the module design phase. These UTPs are executed to eliminate errors at code level or unit level. A unit is the smallest entity which can independently exist, e.g., a program module. Unit testing verifies that the smallest entity can function correctly when isolated from the rest of the codes/ units.
2. **Integration Testing:** Integration Test Plans are developed during the Architectural Design Phase. These tests verify that groups created and tested independently can coexist and communicate among themselves.
3. **System Testing:** System Tests Plans are developed during System Design Phase. Unlike Unit and Integration Test Plans, System Tests Plans are composed by the client?s business team. System Test ensures that expectations from an application developer are met.
4. **Acceptance Testing:** Acceptance testing is related to the business requirement analysis part. It includes testing the software product in user atmosphere. Acceptance tests reveal the compatibility problems with the different systems, which is available within the user atmosphere. It conjointly discovers the non-functional problems like load and performance defects within the real user atmosphere.

Advantage (Pros) of V-Model:

1. Easy to Understand.
2. Testing Methods like planning, test designing happens well before coding.
3. This saves a lot of time. Hence a higher chance of success over the waterfall model.

4. Avoids the downward flow of the defects.
5. Works well for small plans where requirements are easily understood.

Disadvantage (Cons) of V-Model:

1. Very rigid and least flexible.
2. Not a good for a complex project.
3. Software is developed during the implementation stage, so no early prototypes of the software are produced.
4. If any changes happen in the midway, then the test documents along with the required documents, has to be updated.

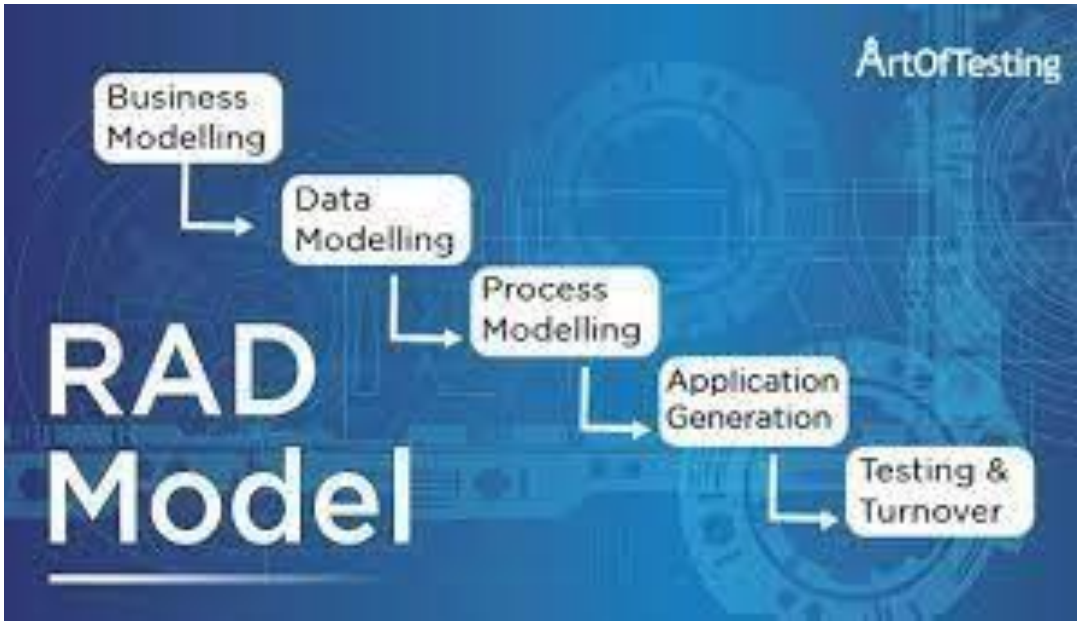
When to use V-Model?

- When the requirement is well defined and not ambiguous.

-The V-shaped model should be used for small to medium-sized projects where requirements are clearly defined and fixed.

-The V-shaped model should be chosen when sample technical resources are available with essential technical expertise.

3. RAD model



Describe

Is a form of incremental model.

In the RAD model, components or functions are developed in parallel as if they were small projects. This development is timed, delivered and assembled into a working prototype.

This can quickly come up with something for customers to see and use and provide feedback regarding their offers and requests.

Let's take a closer look at the steps shown in the diagram that illustrates the RAD model. Step 1. Business modeling.

Identify information flows between different functions of the job.

Step 2. Data modeling.

The information collected from stage 1 will be used to define the data objects it needs for the

job. Step 3. Process modeling.

After the data objects have been defined, they are transformed into business information flows, to some separate work objects.

Step 4. Application generation.

This is the coding stage like in other processes, once the data and execution has been outlined, this stage is to use programming tools, compilers, and programming languages to Code to software functions.

Step 5. Testing and Turnover.

Is the stage of testing the product created, after the coding has been completed, and finally putting it into practice, delivering the product, and calculating the revenue.

Advantages

• Reduced development time.

• Increased reusability of components.

• Make a quick initial assessment.

• Encourage customers to give feedback.

Defect

• The level of the group requires a certain skill.

• Only systems with modules can use this model.

When to use RAD model?

Apply RAD when the project needs to be implemented in a short time of only 2 or 3

months. Only systems with modules can use this model.

When the project requirements are clear, it is easy to divide into small projects, and there are enough resources to divide into many teams, parallel execution at the same time.

Dev / Design requirements must have a lot of experience.

When resources are abundant in both tools, supporting software, documents and people.

4. Iterative model

A model is iterated from start to full spec. The process is then repeated, creating a new version of the software at the end of each iteration of the model

Instead of developing software from the specification and then starting to execute, this model can be reviewed gradually to arrive at the final requirements.

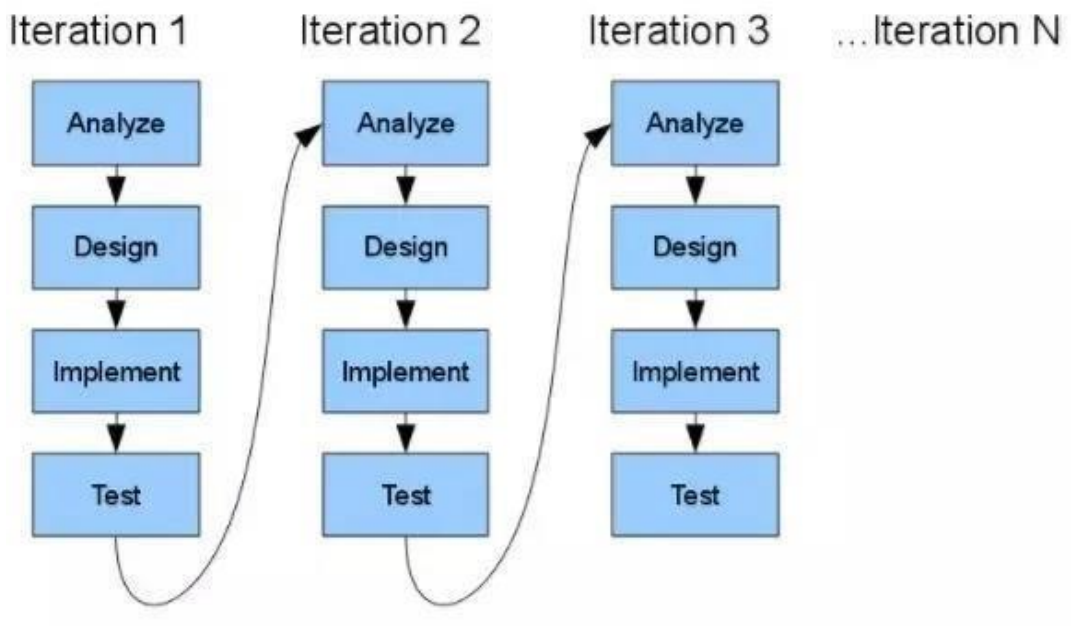


Figure 4

A model is iterated from start to full spec. The process is then repeated, creating a new version of the software at the end of each iteration of the model.

Instead of developing software from the specification and then starting to execute, this model can be reviewed gradually to arrive at the final requirements.

Advantages

Build and perfect the product step by step.

Documentation time will be less than design time.

Some working functionality can be developed quickly and early in the life cycle.

Less expensive to change scopes, requirements.

Ease of risk management. Throughout the lifecycle, software is produced early to facilitate customer evaluation and feedback.

Disadvantages

Demand for many resources.

System architecture or design problems can arise at any time.

Management requirements are more complex.

The progress of the project is highly dependent on the risk analysis phase.

Use cases:

Large, mission-critical enterprise applications that preferably consist of loosely coupled parts, such as microservices or web services.

When to use iterative model?

Here are some situations where an iterative model might be appropriate:

When the requirements are not clear:

When the project is complex:

When time-to-market is a priority:

When risk management is important:

Overall, an iterative model can be useful in situations where change is likely and requirements are not well defined. By working in short, cyclical iterations, teams can adapt and adjust to changes, deliver working software early and frequently, and collaborate more effectively with stakeholders.

III. Explain how to manage risk in the screw model

The Spiral model is a software development process model that combines elements of both the iterative and sequential models. It is a highly flexible and adaptable process model that incorporates iterative feedback loops andrisk management principles.
In the Spiral model, the development process is divided into a series of phases, with each phase being followed by a review and risk analysis. Thephases in the Spiral model are:

1.Planning: In this phase, project goals and objectives areestablished, and the scope of the project is defined.

2.Risk Analysis: In this phase, potential risks to the project areidentified, and a plan is developed to address those risks.
Engineering: In this phase, the actual development of the softwaretakes place, with a focus on designing, coding, and testing the software.
3.Evaluation: In this phase, the software is evaluated to determinewhether it meets the project goals and objectives.

After each phase, there is a review and risk analysis, and the project teamdecides whether to continue with the development process or to stop and reassess the project. If the team decides to continue, the process begins again, starting with the Planning phase.

The Spiral model is best suited for complex and large-scale projects wherethere is a need to manage risks and ensure that the software meets the project goals and objectives. It is particularly useful in situations where therequirements are unclear or changing and where thedevelopment process requires frequent feedback and adaptation.

- Risk management in software development involves the identification, assessment, and mitigation of potential risks and uncertainties that couldimpact the success of a software development project
- Risk management is an essential component of the Spiral model, which isdesigned to address the uncertainty and complexity of software development projects. The Spiral model incorporates risk management principles throughout the software development process.
  In the Spiral model, risk management involves the following steps:

1.Risk identification: The first step is to identify potential risks thatcould impact the success of the project. This may include risks related to technology, personnel, budget, timelines, or other factors.
2.Risk analysis: The second step is to analyze the identified risks todetermine their likelihood of occurring and their potential impacton the project.
3.Risk mitigation: The third step is to develop a plan to mitigate theidentified risks. This may involve implementing measures to reduce the likelihood of a risk occurring or to reduce the potentialimpact of a risk if it does occur.
4.Risk monitoring: The fourth step is to monitor the identified risksthroughout the project life cycle. This helps to ensure that the mitigation strategies are effective and that new risks are identifiedas they emerge.
5.Risk control: The final step is to take action to control the risks thathave been identified. This may involve modifying the project plan or taking other corrective actions to address the risks that have been identified.
The Spiral model is designed to facilitate risk management by incorporating iterative feedback loops and regular risk analyses. The model recognizes that software development projects are complex and uncertain, and that risks are inherent in the process. By incorporating riskmanagementprinciples into the software development process, the Spiralmodel helps to ensure that potential risks are identified, assessed, and mitigated throughout the project. This can help to minimize the impact ofrisks on the project timeline, budget, and quality, and increase the likelihood of project success.

IV. Explain the purpose of a feasibility report

What is project feasibility?

- Project feasibility refers to the assessment of whether a proposed project is technically, economically, and operationally viable. It involves evaluating the project from different perspectives to determine if it is worth pursuing.
- There are three key areas that are typically evaluated during a feasibility study:
  1. Technical feasibility: This involves assessing whether the project can be completed with the available technology, resources, and expertise. The goal is to determine whether the proposed project is technically achievable and whether any technical challenges can be addressed.
  2. Economic feasibility: This involves evaluating the financial viability of the

project. The goal is to determine whether the project is financially feasible, and whether it is likely to generate a positive return on investment. Economic feasibility may involve evaluating the project's cost-benefit ratio, payback period, and other financial metrics.

3. Operational feasibility: This involves assessing whether the project can be integrated into the existing operations of the organization. The goal is to determine whether the proposed project can be implemented without disrupting existing operations or requiring significant changes to existing systems and processes.

– Advantages of conducting feasibility:

1. Helps to identify potential problems and risks: A feasibility study allows organizations to assess the potential risks and challenges associated with a proposed project. This helps to identify potential issues before they become major problems, and enables organizations to develop strategies to mitigate or manage risks.
2. Provides a basis for decision-making: The results of a feasibility study provide valuable information that can be used to make informed decisions about whether to pursue a project or initiative. This helps to avoid wasted time and resources on projects that are not feasible or viable.
3. Enables organizations to plan and prepare: A feasibility study helps organizations to plan and prepare for a project. This includes identifying the resources, skills, and expertise required to successfully complete the project, as well as developing a timeline and budget.
4. Facilitates communication and collaboration: Conducting a feasibility study requires input from different stakeholders, including managers, employees, and external consultants. This helps to promote communication and collaboration within the organization, and ensures that everyone is working towards the same goals.
5. Improves project success rates: By identifying potential problems and risks, and by providing a basis for decision-making and planning, a feasibility study can improve the success rates of projects and initiatives. This helps organizations to achieve their objectives and meet their strategic goals.

– A feasibility report typically includes the following components:
    1. Executive Summary: This section provides a brief overview of the project, including the purpose, objectives, and key findings of the feasibility study.
    2. Introduction: This section provides background information about the project, including the context in which the project is being proposed and the reasons for conducting the feasibility study.
    3. Scope: This section outlines the boundaries of the project and identifies the key stakeholders and resources required to complete the project.
    4. Technical Feasibility: This section evaluates the technical feasibility of the project by assessing the technology and resources required to complete the project, including any potential challenges or limitations.
    5. Financial Feasibility: This section evaluates the financial viability of the project by estimating the costs and benefits associated with the project, including the return on investment.
    6. Operational Feasibility: This section evaluates the operational feasibility of the project by assessing the impact of the project on the organization, including the availability of resources and the potential risks and challenges.
    7. Schedule: This section outlines the timeline for completing the project and identifies any critical milestones or deadlines.

8. Conclusion: This section provides a summary of the key findings of the feasibility study and recommends a course of action based on the evaluation of technical, financial, and operational feasibility.
9. Appendices: This section includes any additional information that supports the findings of the feasibility study, such as technical drawings, financial projections, or market research data.

Overall, the composition of a feasibility report is designed to provide decision-makers with a comprehensive assessment of the proposed project, including an evaluation of the technical, financial, and operational feasibility, as well as a recommended course of action based on the findings of the study.

– Feasibility analysis

Feasibility is an important process in determining whether a project is viable or not. The analysis is usually conducted to assess the economic, organizational, and technical feasibility of a project. Here's an example of how feasibility analysis can be applied to a scenario:

Scenario: A music streaming website is considering implementing a new feature that allows users to create and share custom playlists.

- Economic feasibility

  The cost is estimated: $1,195,000

  The potential revenue from increased user engagement and subscriptions is estimated to be Total potential revenue from sales on the kiosk and website: $1,915,500 ($757,500 from individual music sales and $1,158,000 from customer subscriptions)

  Potential revenue from increased CD sales on the store and website: $205,000

  Potential revenue from music download gift card sales: $153,000

- Organizational feasibility

  The project requires a team with a diverse set of skills including software development, design, digital marketing, and finance The team must be capable of managing the technical aspects of the project such as website and kiosk development, music licensing, and payment processing

  The project will also require significant financial resources for initial development and ongoing maintenance, as well as for marketing and advertising. Therefore, it is essential to have a solid financial plan and to secure funding from investors or other sources.

- Technical feasibility

  The technical feasibility of the Tune Source project is generally high. The development of a music application with the features outlined (searching and streaming music, purchase and subscription options, gift cards) is a common and well-established practice

  There are many existing frameworks and platforms that can be used to build such an application.

However, there are some technical considerations that need to be taken into account, such as the size of the music archive and the number of concurrent users that will be accessing the service. The application will need to be designed and optimized to handle a large volume of requests, and appropriate infrastructure (such as servers and bandwidth) will need tobe in place to support it.

- Conclusions about feasibility to implement

Based on the information provided, the Tune Source project appears to be feasiblefor implementation. The market research indicates that there is a demand for digital music and the various features the project offers. The financial projections also suggest that the project has the potential to generate significant revenue, making it economically feasible.

In terms of organizational feasibility, the project seems to have a well-defined scope and clear goals. However, it would require a team with expertise in areassuch as music licensing, web development, and digital marketing. Therefore, finding the right team and resources may be a challenge.

The technical feasibility of the project depends on the availability of suitable technology to support the various features, such as music streaming, downloads,and subscriptions. Given the current state of technology, it is likely that the project can be technically feasible with the right expertise and resources.

Overall, while there are challenges to be addressed, the Tune Source projectappears to be feasible for implementation

V. Describe how technical solutions can be compared

There are solutions: self-developed web, outsourced outright, buy a few modules

Solutions	Advantages	Disadvantages
self-developed web	Modify and update easily, cheaply.	low security, limited bandwidth, irrelevant ads appear.
outsourced outright	high security, optimize page load speed and maximum web capacity, beautiful interface.	Updating features is difficult, costs extra, not in the style of the buyer.
buy a few modules	Easy and fast operation, High aesthetics, few security holes, Built-in with social networks.	Limited functionality, Not completely free, Depends heavily on terms of use, Limited to updating content.

For TuneSource, which already has a proficient IT team with ISP, we won't need to worry about network bandwidth. The interface does not need to be too beautiful to the eyes and neatly arranged is enough for users. The website needs very few complicated features, so even an inexperienced team can still do it. So you should choose to develop the system yourself or buy some modules. We have a weight matrix that replaces technical solutions when self-developed as follows:

Plot alternative weights matrix to compare engineering solutions. From the replacement weight matrix, a suitable technical solution is given.

Feasibility criteria	Weight	ASP.NET	Self Code php , HTML, CSS, JS	Code it yourself in Java
Technical feasibilty	30%	Score: 10 (No experience yet)	Score: 80 (Experienced members)	Score: 30 (Experienced members)
Economical Feasibility: - Cost to develop - Payback period - Net present Value: 9000$	40%	-15000$ to develop - Payback time :6 years Score: 50	-11000$ to develop - Payback time :3 years Score:90	-13000$ to develop -Payback time: 5 years Score:65
Organization feasibility	30%	Basically, when an organization comes up with a music website, people	Score:70	Score:70

		will always come and see if anything has changed compared to other sites. Since there are so many music websites out there these days, and music is all the same everywhere, it's likely that those organizations will have similar products. Users will quickly get bored. But TuneSource has people who are passionate about big music who own a large amount of music from a long time ago and music of the times. So users will be more diverse and quickly create credibility for the organization. Score:70		
Ranking	100%	44$%	82%	56%

Choose the solution Self Code php ,HTML, CSS, JS

VI. CONCLUSION

In conclusion, the software development lifecycle plays a critical role in the success of any software project. The choice of lifecycle model can significantly impact the project's outcome, and it is crucial to select the right model based on project requirements, timelines, and team size.

https://www.tutorialspoint.com/sdlc/sdlc_quick_guide.htm

https://www.techtarget.com/searchsecurity/definition/What-is-risk-management-and-why-is-it-important https://www.indeed.com/career-advice/career-development/feasibility-studies

https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm?fbclid=IwAR1a40fpWBaX9FLh-QCW5qtRnAouEvUlK4yPQCXpL7tXdL3l0bawBn0NelM

https://www.tutorialspoint.com/sdlc/sdlc_iterative_model.htm

https://www.tutorialspoint.com/sdlc/sdlc_spiral_model.htm

https://www.tutorialspoint.com/sdlc/sdlc_v_model.htm

https://www.tutorialspoint.com/sdlc/sdlc_agile_model.htm

https://www.techtarget.com/searchsoftwarequality/definition/spiral-model

https://www.indeed.com/career-advice/career-development/feasibility-report#:~:text=The%20purpose%20of%20a%20feasibility%20report%20is%20to%20determine%20the,the%20feasibility%20of%20each%20approach.

https://www.ukessays.com/assignments/feasibility-study-importance-structure-and- comparing-solutions-6122-2021.php