ASSIGNMENT 01 FRONT SHEET

Qualification	**BTEC Level 5 HND Diploma in Computing**		
Unit number and title	Unit 09: Software Development Life Cycle		
Submission date	3 August 2022	**Date Received 1st submission**	
Re-submission Date		**Date Received 2nd submission**	
Student Name	Truong Van Tuan Kiet	**Student ID**	GCC200203
Class	GCC0903	**Assessor name**	Nguyen Thai Nghe
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:		

Assignment Brief 01 (RQF)

Higher National Certificate/Diploma in Business

Student Name/ID Number:	
Unit Number and Title:	**Unit 09: Software Development Life Cycle**
Academic Year:	
Unit Assessor:	
Assignment Title:	**Plan a software development life cycle**
Issue Date:	**07/12/2020**
Submission Date:	
Internal Verifier Name:	
Date:	

Submission Format:

Format:

- The submission is in the form of 1 document.
- You must use the Times font with 12pt size, turn on page numbering; set line spacing to 1.3 and margins to be as follows: left = 1.25cm, right = 1cm, top = 1cm, bottom = 1cm. Citation and references must follow the Harvard referencing style.

Submission:

- Students are compulsory to submit the assignment in due date and in a way requested by the Tutor.
- The form of submission will be a soft copy posted on http://cms.greenwich.edu.vn/.
- Remember to convert the word file into **PDF** file before the submission on CMS.

Note:

- The individual Assignment must be your own work, and not copied by or from another student.
- If you use ideas, quotes or data (such as diagrams) from books, journals or other sources, you must reference your sources, using the Harvard style.
- Make sure that you understand and follow the guidelines to avoid plagiarism. Failure to comply this requirement will result in a failed assignment.

Unit Learning Outcomes:

LO1 Describe different software development lifecycles.

LO2 Explain the importance of a feasibility study.

Assignment Brief and Guidance:

Assignment scenario

Tune Source is a company headquartered in southern California. Tune Source is the brainchild of three entrepreneurs with ties to the music industry: John Margolis, Megan Taylor, and Phil Cooper. Originally, John and Phil partnered to open a number of brick-and-mortar stores in southern California specialising in hard-to-find and classic jazz, rock, country, and folk recordings. Megan soon was invited to join the partnership because of her contacts and knowledge of classical music. Tune Source quickly became known as the place to go to find rare audio recordings. Annual sales last year were $40 million with annual growth at about 3%–5% per year. Tune Source currently has a website that enables customers to search for and purchase CDs. This site was initially developed by an Internet consulting firm and is hosted by a prominent local Internet Service Provider (ISP) in Los Angeles. The IT department at Tune Source has become experienced with Internet technology as it has worked with the ISP to maintain the site.

System Request

Project Sponsor: Carly Edwards, Assistant Vice President, Marketing.

Business Need: This project has been initiated to increase sales by creating the capability of selling digital music downloads to customers through kiosks in our stores, and over the Internet using our website.

Business Requirements: Using the Web or in-store kiosks, customers will be able to search for and purchase digital music downloads. The specific functionality that the system should have includes the following:

- Search for music in our digital music archive.
- Listen to music samples.
- Purchase individual downloads at a fixed fee per download.
- Establish a customer subscription account permitting unlimited downloads for a monthly fee.
- Purchase music download gift cards.

Business Value: We expect that Tune Source will increase sales by enabling existing customers to purchase specific digital music tracks and by reaching new customers who are interested in our unique archive of rare and hard-to-find music. We expect to gain a new revenue stream from customer subscriptions to our download services. We expect some increase in cross-selling, as customers who have downloaded a track or two of a CD decide to purchase the entire CD in a store or through our website. We also expect a new revenue stream from the sale of music download gift cards.

Conservative estimates of tangible value to the company include the following:

- $757,500 in sales from individual music downloads.

- $950,000 in sales from customer subscriptions.
- $205,000 in additional in-store or website CD sales.
- $153,000 in sales from music download gift cards.

Special Issues or Constraints:

- The marketing department views this as a strategic system. The ability to offer digital music downloads is critical in order to remain competitive in our market niche. Our music archive of rare and hard-to-find music is an asset that is currently underutilized.
- Many of our current loyal customers have been requesting this capability, and we need to provide this service or face the loss of these customers' business.
- Because customers have a number of music download options available to them elsewhere, we need to bring this system to the market as soon as possible.

Tasks

Complete the following tasks:

Task 1 – SDLC model

You are a project manager of a company named ABC. Your company has been hired by Tune Source to carry out a project that helps them develop a software for the requirements specified in the system request. As the first step, you need to:

1. Describe the following SDLC models: waterfall, v-model, prototyping, scrum and spiral. Choose one that you think suitable for the project and explain why.

   - 350 - 500 words for each model.
   - Explanation: 400 – 600 words.

   Discuss the suitability of each of the SDLC models for the project. For each model, specify whether it is most, moderately or least suitable.

   - Word limit: 800 - 1000 words.

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

   - Word limit: 800 – 1200 words.

2. Identify some risks and discuss an approach to manage them.

   You will have the present what is **Risk Management process** with clear illustrations and explanations.

   Then you will create a **Risk Management Matrix** to assess and manage risks of Tune Source project.

   - Word limit: 600 – 1000 words.

Task 2 – Feasibility study

1. Discuss the purpose of conducting a feasibility study for the project.

   - Word limit: 400 – 600 words.

2. Discuss how the three feasibility criteria (technical, economic, organizational) are applied to the project. Discuss whether the project is feasible.

   Discuss alternative technical solutions using the alternative matrix.

   - Word limit: 1200 – 1500 words.

3. Explain the components of a feasibility report.

   Discussion economic feasibility study on Tune Source project (NPV, Cashflow, Break-Even Point)

   - Word limit 350 – 500 words.

   Discussion organizational feasibility study on Tune Source project

   - Word limit 350 – 500 words.

4. Assess the impact of each feasibility criterion on a software investigation.

   Discussion and represent as feasibility alternatives matrix for Tune Source project

   - Word limit: 500 – 700 words.

<table>
<tr><th colspan="4">Learning Outcomes and Assessment Criteria (Assignment 01):</th></tr>
<tr><td>Learning Outcome</td><td>Pass</td><td>Merit</td><td>Distinction</td></tr>
<tr><td><b>LO1</b> Describe different software development lifecycles</td><td><b>P1</b> Describe two iterative and two sequential software lifecycle models.
<b>P2</b> Explain how risk is managed in the Spiral lifecycle model.</td><td><b>M1</b> Describe, with an example, why a particular lifecycle model is selected for a development environment.</td><td><b>D1</b> Assess the merits of applying the Waterfall lifecycle model to a large software development project.</td></tr>
<tr><td><b>LO2</b> Explain the importance of a feasibility study</td><td><b>P3</b> Explain the purpose of a feasibility report.
<b>P4</b> Describe how technical solutions can be compared.</td><td><b>M2</b> Discuss the components of a feasibility report.</td><td><b>D2</b> Assess the impact of different feasibility criteria on a software investigation.</td></tr>
</table>

Table of Contents

Assignment 1

Task 1 – SDLC model

I. Describe the following SDLC models: waterfall, v-model, prototyping, scrum and spiral.

1. SDLC - Waterfall Model

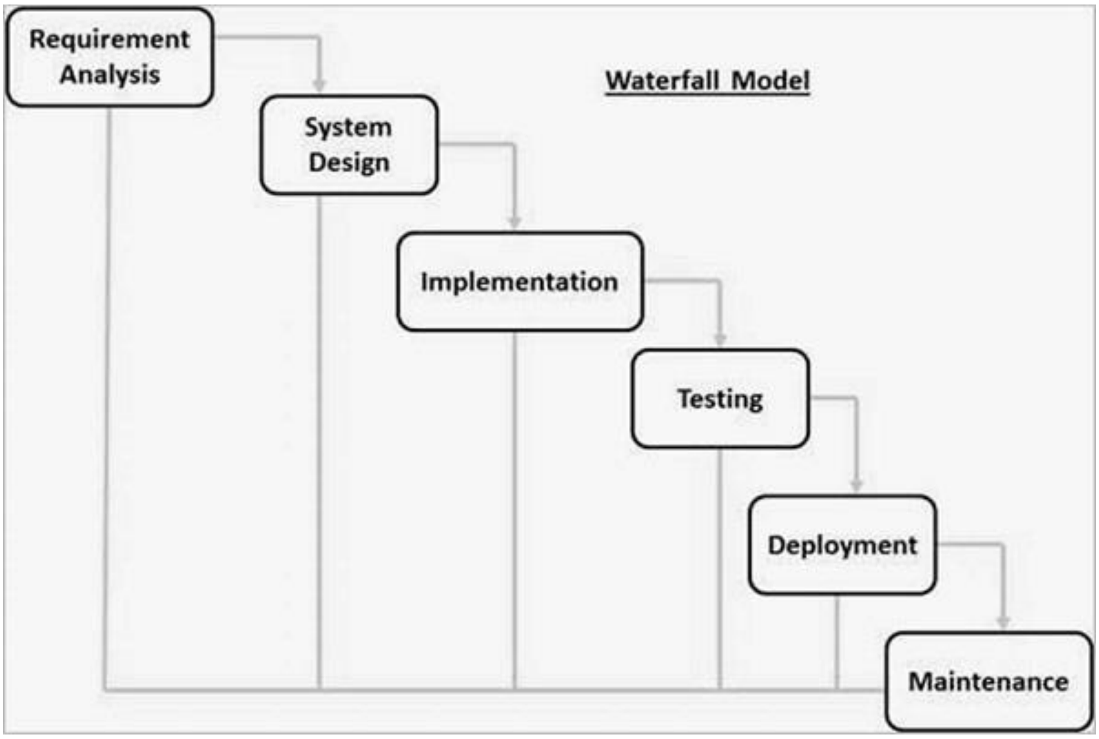
According to (Tutorials Point, 2022). The first Process Model to be introduced was the Waterfall Model. The term "linear-sequential life cycle model" is also used to describe it. It is incredibly easy to use and comprehend. There is no overlap between stages in a waterfall model; each step must be finished before the subsequent phase can start.

The first SDLC methodology for software development was the waterfall model.
The software development process is depicted using the waterfall model, which follows a linear sequential flow. This implies that a phase of development can only start if the one before it is finished. The stages in this waterfall model do not cross over.

Waterfall Model - Design

The Waterfall Approach was the first commonly used SDLC Model in software engineering to assure project success. The entire software development process is split into several phases using "The Waterfall" technique. Typically, in this Waterfall paradigm, the results of one phase serve as the input for the following step successively.

The following illustration is a representation of the different phases of the Waterfall Model.



The sequential phases in the Waterfall model are −

- **Requirement Gathering and analysis** − During this stage, all potential system needs are gathered and outlined in a requirement specification document.
- **System Design** − In this phase, the required specifications from the previous phase are examined, and the system design is created. This system design aids in determining the overall system architecture as well as the hardware and system requirements.
- **Implementation** − The system is initially built as discrete programs known as units, which are then combined in the following phase, using inputs from the system design. Unit testing is the process of developing and evaluating each unit for functionality.
- Integration and Testing − Following the testing of each unit created during the implementation phase, the entire system is merged. The entire system is tested for errors and failures after integration.
- **Deployment of system** − Once the product has undergone functional and non-functional testing, it is either published to the market or deployed in the customer's environment.
- **Maintenance** − Various problems might arise in a client environment. Patches are published to address certain problems. Additionally, improved versions of the product are issued. To bring about these changes in the surroundings of the consumer, maintenance is performed.

The progression is perceived as flowing slowly downhill (like a waterfall) through the stages as all of these phases are connected to one another. The "Waterfall Model" gets its name because the following phase doesn't begin until the prior phase's established set of goals have been met and it has been approved. Phases do not cross over in this model.

Waterfall Model - Application

Every piece of software-generated is unique, and depending on internal and external circumstances, a proper SDLC strategy must be used. The following are some scenarios when the waterfall paradigm is most useful:

- Requirements are very well documented, clear and fixed.
- Product definition is stable.
- Technology is understood and is not dynamic.
- There are no ambiguous requirements.
- Ample resources with required expertise are available to support the product.
- The project is short.

Waterfall Model - Advantages

The ability to departmentalize and exercise control is one of waterfall development's benefits. A product can move through the stages of the development process model one at a time by setting deadlines for each step and following a timetable.

Development moves from concept, through design, implementation, testing, installation, and troubleshooting, and ends up at operation and maintenance. The stages of development are carried out in a certain order.

Some of the major advantages of the Waterfall Model are as follows:

- Simple and easy to understand and use
- Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a review process.
- Phases are processed and completed one at a time.
- Works well for smaller projects where requirements are very well understood.
- Clearly defined stages.
- Well-understood milestones.
- Easy to arrange tasks.
- Process and results are well documented.

Waterfall Model - Disadvantages

Waterfall development has the drawback of not allowing for a lot of reflection or correction. It is quite challenging to go back and fix something that wasn't carefully thought out or documented during the idea stage after an application has entered the testing phase.

The major disadvantages of the Waterfall Model are as follows:

- No working software is produced until late during the life cycle.
- High amounts of risk and uncertainty.
- Not a good model for complex and object-oriented projects.
- Poor model for long and ongoing projects.

- Not suitable for the projects where requirements are at a moderate to high risk of changing. So, risk and uncertainty is high with this process model.
- It is difficult to measure progress within stages.
- Cannot accommodate changing requirements.
- Adjusting scope during the life cycle can end a project.
- Integration is done as a "big-bang. at the very end, which doesn't allow identifying any technological or business bottleneck or challenges early.

2. SDLC - V-Model

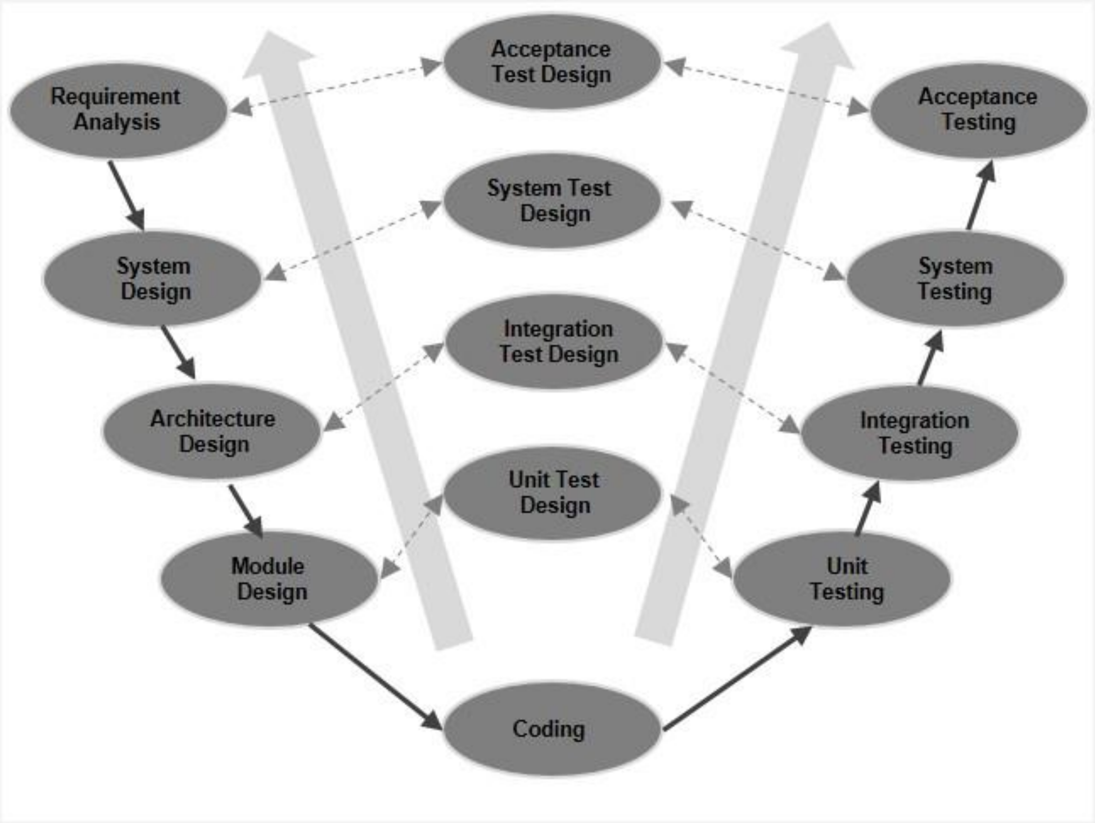
According to (Tutorialspoint.com, 2022) .The V-model is an SDLC framework where processes are executed sequentially in a V-shape. The Verification and Validation Model is another name for it.

A testing phase is linked to each relevant development step in the V-Model, which is an extension of the waterfall model. This implies that there is a testing phase that is directly related to each and every phase of the development cycle. This is a very structured approach, and the start of the subsequent phase only occurs after the conclusion of the preceding phase.

V-Model - Design

The V-Model parallel plans the equivalent testing phase of the development phase. Therefore, the Validation stages are on the other side of the "V" than the Verification phase. The two sides of the V-Model are joined during the coding phase.

The following illustration depicts the different phases in a V-Model of the SDLC.



V-Model - Verification Phases

There are several Verification phases in the V-Model, each of these are explained in detail below.

- Business Requirement Analysis

The needs for the product are first grasped from the viewpoint of the client at this phase of the development cycle. The consumer is extensively communicated with at this phase in order to ascertain his expectations and precise needs. Given that most consumers are unsure about their specific wants, this is a crucial task that has to be managed carefully. As business requirements may be utilized as an input for acceptance testing, the design planning for the acceptance test is completed at this stage.

- Systems Design

When you have a firm understanding of the requirements for your product, it's time to design the entire system. The system design will include a thorough grasp of the hardware and communication configuration for the product that is currently being developed. On the basis of the system design, the system test plan is created. By doing this sooner, more time is available for the actual test run.

- Architectural Design

In this stage, architectural specifications are comprehended and designed. The ultimate choice is often made after considering the technical and financial viability of several different technological approaches that have been offered. The system design is further divided into modules that handle various functionalities. This is additionally known as High-Level Design (HLD).

At this point, it is very obvious and stated how data will be transferred and communicated both inside the internal modules and with other systems. This knowledge may be used to develop and describe integration tests at this level.

- Module Design

This stage, known as Low-Level Design (LLD), specifies the internal details for all the system modules. The design must be compatible with the other system architecture modules as well as other external systems. Unit testing is a crucial step in any development process since it enables the early elimination of the majority of flaws and mistakes. Based on the internal module designs, these unit tests may now be created.

Coding Phase

In the coding phase, the actual coding of the system components created during the design phase is done. The system and architectural requirements determine the most appropriate programming language.

The coding is done in accordance with the coding standards and rules. Before the final build is checked into the repository, the code undergoes several code reviews and is optimized for optimum performance.

Validation Phases

The different Validation Phases in a V-Model are explained in detail below.

- Unit Testing

During this step of validation, the code is subjected to unit tests created during the module design process. Unit testing is testing at the code level and aids in the early bug elimination process, however it cannot find every flaw.

- Integration Testing

Architectural design is connected to integration testing. Integration tests are run to evaluate how well the system's internal parts get along and communicate with one another.

- System Testing

System design and testing are closely related processes. System tests examine the overall system's operation as well as the system's connectivity with external systems. The majority of hardware and software compatibility problems can be found during the execution of this system test.

- Acceptance testing

Acceptance testing involves evaluating the product in a user environment and is connected to the business requirement analysis step. Compatibility problems with other systems that are present in the user environment are discovered during acceptance tests. Additionally, it finds non-functional problems in the real user environment, such as load and performance flaws.

V- Model ─ Application

As both models are of the sequential kind, the V-model application is quite similar to that of the waterfall model. Before the project begins, the requirements must be crystal clear because it is typically expensive to go back and make adjustments. Since medical development is a highly regimented area, this technique is employed there.

The following pointers are some of the most suitable scenarios to use the V-Model application.

- Requirements are well defined, clearly documented and fixed.
- Product definition is stable.
- Technology is not dynamic and is well understood by the project team.
- There are no ambiguous or undefined requirements.
- The project is short.

V-Model - Pros and Cons

The V-Model technique has the benefit of being relatively simple to comprehend and use. This model's simplicity makes it simpler to maintain as well. The model's lack of adaptability means that in the event that requirements change—which is typical in today's dynamic world—making the adjustment becomes exceedingly expensive.

The advantages of the V-Model method are as follows −

- This is a highly-disciplined model and Phases are completed one at a time.
- Works well for smaller projects where requirements are very well understood.
- Simple and easy to understand and use.

- Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a review process.

The disadvantages of the V-Model method are as follows −

- High risk and uncertainty.
- Not a good model for complex and object-oriented projects.
- Poor model for long and ongoing projects.
- Not suitable for the projects where requirements are at a moderate to high risk of changing.
- Once an application is in the testing stage, it is difficult to go back and change a functionality.
- No working software is produced until late during the life cycle.

3. Scrum - Framework

According to (Tutorials Point, 2022). A framework called Scrum is used to create and maintain complicated products. Scrum was created by Ken Schwaber and Jeff Sutherland. They support the Scrum Rules as a unit.

Scrum Definition

Scrum is a methodology that enables people to work productively and creatively to create products with the best potential value while addressing complex adaptive issues.

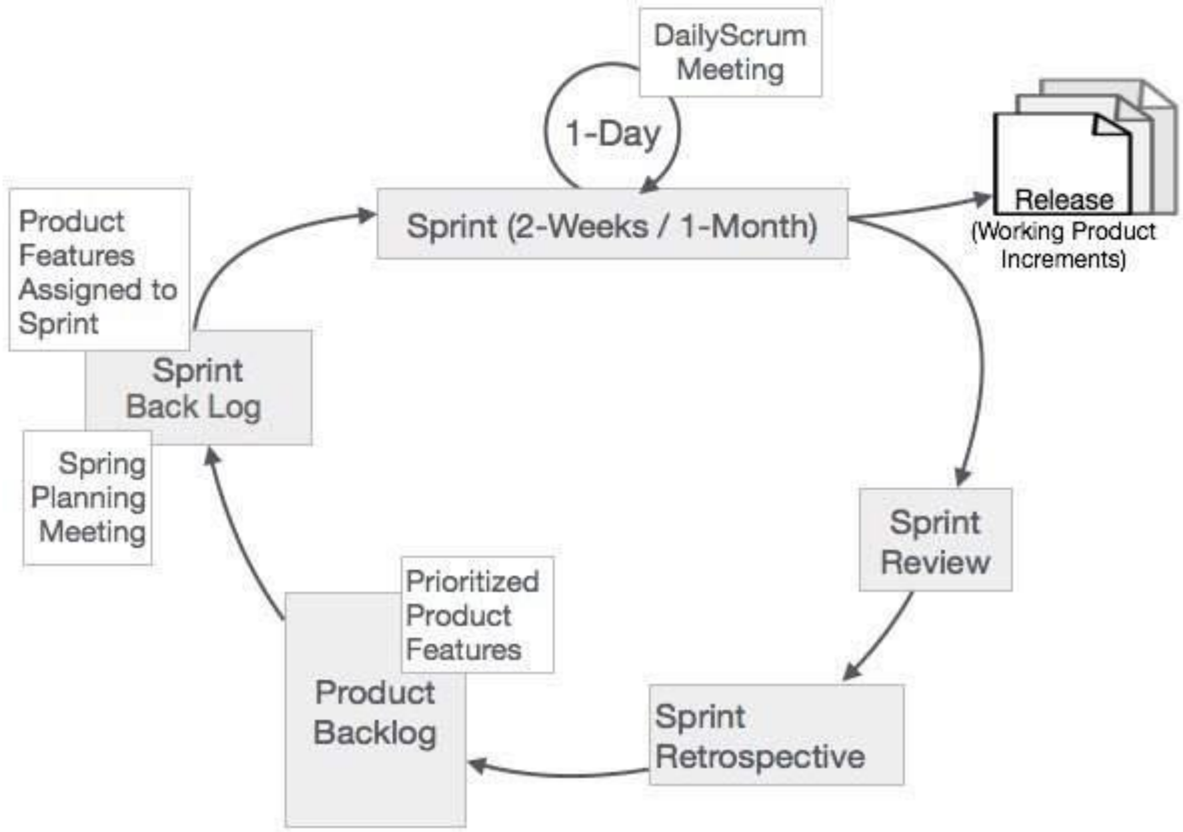
Since the early 1990s, complicated product development has been managed using the Scrum framework. Scrum is a framework within which other processes and methods can be used, not a process or methodology for creating goods. Scrum makes it evident how effective your product management and development processes are in comparison so you can make changes.

Scrum Teams, together with the roles, events, objects, and rules they are connected with, make up the Scrum framework. Each element of the framework has a distinct function and is necessary for Scrum to function and be used.

The Scrum guidelines control the interactions and linkages between the events, roles, and artifacts by connecting them all together. This course includes explanations of the Scrum rules.

Note: There are widespread misunderstandings in the industry that Scrum entails a lack of documentation, that the scrum team is made up only of engineers, etc. It is not totally accurate; in following parts, we will provide explanations.

Scrum Process Framework


DailyScrum Meeting
1-Day
Sprint (2-Weeks / 1-Month)
Release
(Working Product Increments)
Product Features Assigned to Sprint
Sprint Back Log
Spring Planning Meeting
Sprint Review
Prioritized Product Features
Product Backlog
Sprint Retrospective

Scrum uses predetermined events to establish regularity. Every event has a limited duration since they are all time-boxed events. The ensuing chapters provide a more thorough description of the events.

Sprint

The core of Scrum is a Sprint, which is a period of time, usually lasting two weeks or a month, during which a possible product increment is developed. Immediately following the completion of the preceding Sprint, a new Sprint begins. Daily scrums, development work, the Sprint planning, the Sprint review, and the Sprint retrospective are all parts of a sprint.

- The work that has to be done during the Sprint is planned jointly by the Scrum Team.
- The Daily Scrum Meeting is a timed, 15-minute meeting where the Scrum Team coordinates the day's activities and develops a strategy.
- At the conclusion of the Sprint, a Sprint Review is held to review the Increment and, if necessary, make modifications to the Product Backlog.
- After the Sprint Review and before the following Sprint Planning, the Sprint Retrospective takes place. The purpose of this meeting is for the Scrum Team to assess itself and develop a strategy for enhancements to be implemented during the upcoming Sprint.

Conclusion

Scrum is a framework for processes that specifies certain guidelines, occasions, and responsibilities to establish regularity. However, assuming the fundamental scrum rules are followed, it may be modified to fit the demands of each business.

4. SDLC - Spiral Model

According to (Tutorials Point , 2022). The spiral model combines the idea of iterative development with the systematic, controlled aspects of the waterfall model. Iterative and sequential linear development models, or the waterfall model, are combined to create the spiral model, which places a strong focus on risk analysis. Through each cycle around the spiral, it enables incremental product launches or incremental product improvement.

Spiral Model Design

Four stages make up the spiral model. In spiral-style iterations, a software project regularly moves through these stages.

Identification

The baseline spiral is where this phase begins by obtaining the business needs. This phase includes the identification of system requirements, subsystem requirements, and unit requirements, which are all completed in the spirals that follow as the product evolves.

In this phase, the client and the system analyst must maintain constant contact in order to grasp the system requirements. The product is launched in the chosen market at the conclusion of the spiral.

Design

The Design phase includes architectural design, logical module design, physical product design, and final design in the succeeding spirals. It begins with conceptual design in the baseline spiral.

Construct or Build

The real software product is produced at each spiral throughout the construction phase. A POC (Proof of Concept) is created at this phase of the baseline spiral, when the product is still only an idea and the design is being worked on, to gather consumer input.

Then, in later spirals with more precise requirements and design specifics, a workable version of the program known as the build is created. The consumer is contacted for comments on these builds.

Spiral Model Application

The Spiral Model is frequently employed in the software business because it is consistent with how any product would naturally develop—that is, by learning as it matures and posing the least amount of risk to both the client and the development company.

The following pointers explain the typical uses of a Spiral Model:

- When there is a budget constraint and risk evaluation is important.
- For medium to high-risk projects.
- Long-term project commitment because of potential changes to economic priorities as the requirements change with time.
- Customer is not sure of their requirements which is usually the case.
- Requirements are complex and need evaluation to get clarity.
- New product line which should be released in phases to get enough customer feedback.
- Significant changes are expected in the product during the development cycle.

Spiral Model - Pros and Cons

The spiral lifecycle model has the benefit that it enables the addition of product components as they become available or recognized. This ensures that the requirements and design from earlier are not in conflict.

This strategy enables for a smooth transition to a maintenance activity since it is consistent with approaches that have various software builds and releases. The spiral model's requirement for early user engagement in the system development endeavor is another advantage of this approach.
On the other hand, producing such things requires extremely rigorous control, and there is a chance that the spiral will continue indefinitely. To effectively create and distribute the product, it is crucial to have a disciplined approach to change and to the extent to which change requests are accepted.

The advantages of the Spiral SDLC Model are as follows −

- Changing requirements can be accommodated.
- Allows extensive use of prototypes.
- Requirements can be captured more accurately.
- Users see the system early.
- Development can be divided into smaller parts and the risky parts can be developed earlier which helps in better risk management.

The disadvantages of the Spiral SDLC Model are as follows:

- Management is more complex.
- End of the project may not be known early.
- Not suitable for small or low risk projects and could be expensive for small projects.
- Process is complex
- Spiral may go on indefinitely.
- Large number of intermediate stages requires excessive documentation.

5. Discuss the merits of applying the waterfall model to a large software development project.

According to (Team Clarizen, 2021), Use the waterfall model for large software projects because of its advantages:

Easy to Follow

The waterfall approach is effective because it is simple to use, much like a map or a predetermined system. Project managers may stay organized and maintain the flow of the process by always being aware of where they are in the project's life cycle.

Minimal Management

Because a waterfall model's rules are so clearly defined, these types of projects are somewhat easy to manage. Simply perform each step one at a time till completion. For projects where the criteria are simple to comprehend and the order is well specified, the waterfall model works best.

No Overlapping

In contrast to other project management techniques, this system's stages never cross over. Analysis must be finished before design can begin, and design cannot begin until coding is finished. so on, so forth. This makes it simple to track the project's phases without becoming confused between two different processes.

Final Product

In contrast to other project management techniques, this system's stages never cross over. Analysis must be finished before design can begin, and design cannot begin until coding is finished. so on, so forth. This makes it simple to track the project's phases without becoming confused between two different processes.

Choosing the right type of model to manage a large project is really the universal first step to planning a successful project. The waterfall methodology was developed years ago with a clear set of steps, advantages, and rules. Finally, I chose waterfall for my project because of the advantages it brings.

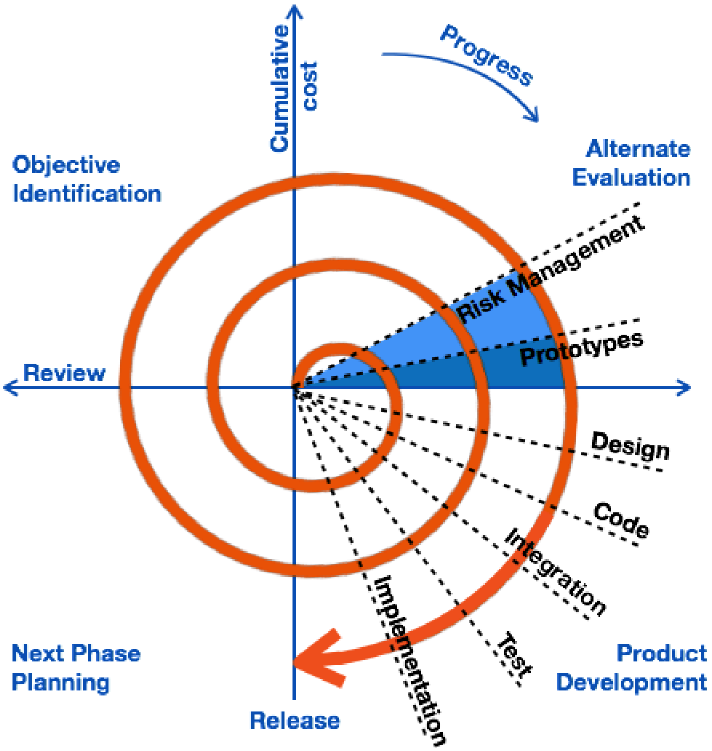
II. Identify some risks and discuss an approach to manage them.

1. Risk Management process.

Evaluation and Risk Analysis

Identification, estimation, and monitoring of management and technical risks, such as schedule slippage and cost overrun, are all part of the risk analysis process. At the conclusion of the first iteration, the client analyzes the program and offers comments after testing the build.

The following illustration is a representation of the Spiral Model, listing the activities in each phase.



The software development process enters the following iteration based on the customer evaluation and then adopts a linear strategy to execute the consumer feedback recommendations.

Throughout the software's lifespan, revisions along the spiral continue to be made (Tutorials Point , 2022).

2. Risk Management Matrix

No.	Risk	Description	Impact	Solution
1	**Financial**	Due to In the course of the project, unexpected costs may occur. A project is long for many months, so it is difficult to control all the main costs as well as rising costs.	Causing a budget deficit of the project and affecting the profit of the project.	Detailed budget planning for the project. All expenses must be recorded for easy control.

2	**Over time**	Due to unreasonable assignment of tasks, unexpected errors occurred during the programming process, prolonging the project's time.	Prolong the project's time, lose credibility with customers, risk contract penalties.	Allocate appropriate work for each employee, and regularly check errors during project development to detect errors and quickly fix errors.
3	**Poor design**	Due to poor design ideas, not grasping design trends.	Create software with a boring interface, difficult to reach customers.	Recruit a professional designer or training before making the project
4	**Can’t satisfy new technology**	Because the technology in use is outdated, employees have not updated the new technology in time.	Creating software that uses outdated technology slows down performance and makes it difficult to reach users.	Use new technology to program and train employees on new technology.

Task 2 – Feasibility study

I. Discuss the purpose of conducting a feasibility study for the project.

A feasibility study looks at the viability of an idea with an emphasis on identifying potential problems and attempts to answer one main question: Will the idea work and should you proceed with it?

Before you start creating your business strategy, you need to determine how, where, and to whom you want to offer a service or product. Additionally, you must evaluate your rivals and calculate the amount of funding required to launch and sustain your firm until it achieves success.

Studies on feasibility talk about aspects like where and how the firm will run. They serve as an important tool for creating a successful business plan and offer comprehensive information about the company to help assess whether and how it may prosper.

Feasibility studies seek to logically and objectively identify the advantages and disadvantages of a potential business endeavor or current venture, as well as the possibilities and risks it may face from its surroundings, as well as the resources it will need to succeed. The cost necessary and value to be realized are, in the simplest terms, the two factors used to assess feasibility.

The historical context of the company or project, a description of the good or service, accounting statements, information on the operations and management, marketing research and policies, financial data, legal requirements, and tax obligations should all be included in a well-designed feasibility study. In general, feasibility assessments come before project implementation and technological development.

Because a feasibility study assesses the project's likelihood of success, perceived objectivity is a crucial component of the research's credibility with potential backers and lenders. It must thus be carried out objectively and impartially in order to give data on which judgments may be made.

Reasons to Do a Feasibility Study

Conducting a feasibility study is a good business practice. If you examine successful businesses, you will find that they did not go into a new business venture without first thoroughly examining all of the issues and assessing the probability of business success.

Below are other reasons to conduct a feasibility study:

- Gives focus to the project and outline alternatives.
- Narrows business alternatives.
- Identifies new opportunities through the investigative process.
- Identifies reasons not to proceed.
- Enhances the probability of success by addressing and mitigating factors early on that could affect the project.
- Provides quality information for decision making.
- Provides documentation that the business venture was thoroughly investigated.
- Helps in securing funding from lending institutions and other monetary sources.
- Helps to attract equity investment.

The feasibility study is a critical step in the business assessment process. If properly conducted, it may be the best investment you ever made (Thinkpositive Dubai, 2020).

II. Discuss how the three feasibility criteria (technical, economic, organizational) are applied to the project. Discuss whether the project is feasible. Discuss alternative technical solutions using the alternative matrix.

1. Discuss how the three feasibility criteria are applied to the project.

Technical feasibility

Technical feasibility analysis has a big role in resource allocation. It takes into account the project's technological needs. The technical capacity of the organization is then contrasted with the technical needs. If the internal technical capacity is sufficient to satisfy the project needs, the systems project is regarded as technically viable.

The analyst must determine if the request can be fulfilled by upgrading or expanding upon the currently available technological resources. This is where system analysts' knowledge is useful since they may address the issue of technological viability by drawing on their own experience and relationships with vendors.

The essential questions that help in testing the operational feasibility of a system include the following:

- Is the project feasible within the limits of current technology?
- Does the technology exist at all?
- Is it available within given resource constraints?
- Is it a practical proposition?
- Manpower- programmers, testers & debuggers
- Software and hardware
- Are the current technical resources sufficient for the new system?
- Can they be upgraded to provide to provide the level of technology necessary for the new system?
- Do we possess the necessary technical expertise, and is the schedule reasonable?
- Can the technology be easily applied to current problems?
- Does the technology have the capacity to handle the solution?
- Do we currently possess the necessary technology?

Operational feasibility

Operational feasibility entails estimating whether the system will be used if it is designed and deployed, and it depends on the human resources available for the project.

Operational feasibility is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

Operational feasibility examines the organization's readiness to sustain the suggested system. The hardest of the possibilities to estimate is definitely this one. It's critical to comprehend the management's commitment to the suggested project in order to assess its viability. If management made the request, it is probable that management would support it, and the system will be adopted and implemented. The workforce base's acceptance of the change is equally crucial, though.

The essential questions that help in testing the operational feasibility of a system include the following:

- Does current mode of operation provide adequate throughput and response time?
- Does current mode provide end users and managers with timely, pertinent, accurate and useful formatted information?
- Does current mode of operation provide cost-effective information services to the business?

- Could there be a reduction in cost and or an increase in benefits?
- Does current mode of operation offer effective controls to protect against fraud and to guarantee accuracy and security of data and information?
- Does current mode of operation make maximum use of available resources, including people, time, and flow of forms?
- Does current mode of operation provide reliable services
- Are the services flexible and expandable?
- Are the current work practices and procedures adequate to support the new system?
- If the system is developed, will it be used?
- Manpower problems
- Labour objections
- Manager resistance
- Organizational conflicts and policies
- Social acceptability
- Government regulations
- Does management support the project?
- Are the users not happy with current business practices?
- Will it reduce the time (operation) considerably?
- Have the users been involved in the planning and development of the project?
- Will the proposed system really benefit the organization?
- Does the overall response increase?
- Will accessibility of information be lost?
- Will the system affect the customers in considerable way?
- Legal aspects
- How do the end-users feel about their role in the new system?
- What end-users or managers may resist or not use the system?
- How will the working environment of the end-user change?
- Can or will end-users and management adapt to the change?

Economic feasibility

Cost/benefit analysis is another name for economic analysis. It is the approach that is most usually used to assess a new system's efficacy. In economic analysis, the process entails calculating the expected savings and benefits from a candidate system and contrasting them with the costs. The choice to develop and deploy the system is taken if advantages outweigh expenses. Before acting, an entrepreneur must carefully compare the rewards and costs.

Possible questions raised in economic analysis are:

- Is the system cost effective?
- Do benefits outweigh costs?
- The cost of doing full system study

- The cost of business employee time
- Estimated cost of hardware
- Estimated cost of software/software development
- Is the project possible, given the resource constraints?
- What are the savings that will result from the system?
- Cost of employees' time for study
- Cost of packaged software/software development
- Selection among alternative financing arrangements (rent/lease/purchase)

(Ogbebor, 2011)

2. Discuss alternative technical solutions using the alternative matrix.

Option 1:

- Personnel:

2	System Analysts (400 h/ea $35/h)	$28 000
3	Programmer/Analysts (250 h/ea $25/h)	$18 750
1	GUI designer (200 h/ea $35/h)	$7 000
1	Telecommunications Specialist (50 h/ea $45/h)	$2 250
1	System Architect (50 h/ea $45/h)	$4 500
1	Database Specialist (15 h/ea $45/h)	$600
1	System Librarian (250 h/ea $35/h)	$2 500

- Expenses:

2	Smalltalk training registration ($3500/student)	$7 000

- New hardware and software:

1	Dev Server (Pentium Pro class)	$18,700
1	Server Software (Operating system, misc.)	$1 500
1	DBMS server software	$7 500
7	DBMS Client software ($950 per client)	$6 650

Total development Costs: **$104 950**

PROJECTED ANNUAL OPERATING COST:

- Personnel:

2	Programmer/Analysts (125 h/ea $25/h)	$6 250
1	System Librarian (20 h/ea $10/h)	$200

- Expense

1	Maintenance Agreement for Pentium Pro Server	$995

1	Maintenance Agreement for Server DBMS software	$525
	Preprinted forms (15 000/y @.22/form	$3 300

Total projected annual operating cost: **$11 270**

Detail

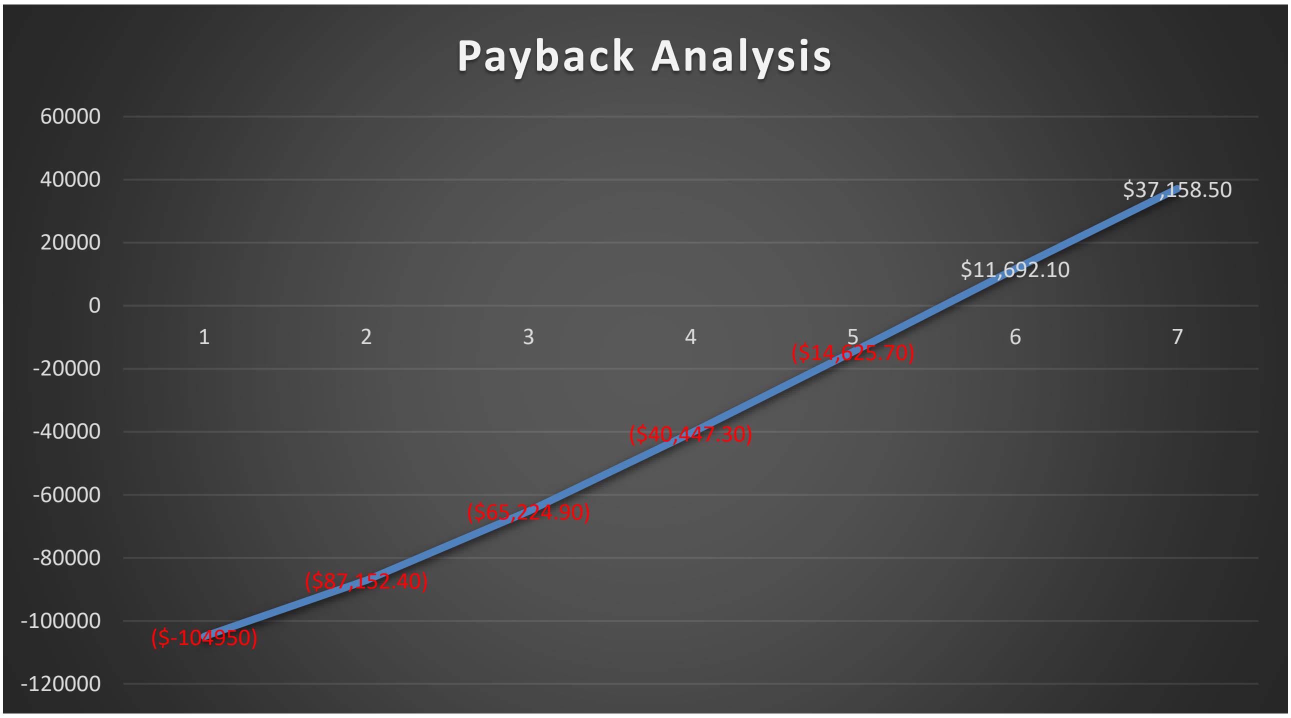
	Price	**Year 1**		**Year 2**		**Year 3**	
Customer/year							
		Quantity	**Total**	**Quantity**	**Total**	**Quantity**	**Total**
Subscription/person	$15	1000	$15 000	1200	$18000	1400	$21 000
Download/person	$18	400	$7 200	500	$9000	600	$10800
Sale CD/person	$20	300	$6000	400	$8000	500	$10000
Download gift card/person	$15	200	$3000	300	$4500	400	$6000
Total		$31 200		$39500		$47 800	

	Price	**Year 4**		**Year 5**		**Year 6**	
Customer/year							
		Quantity	**Total**	**Quantity**	**Total**	**Quantity**	**Total**
Subscription/person	$15	1500	$22 500	1600	$24 000	1 700	$25 500
Download/person	$18	700	$12 600	800	$14 400	900	$16 200
Sale CD/person	$20	600	$12 000	700	$14 000	800	$16 000
Download gift card/person	$15	500	$7 500	600	$9 000	700	$10 500
Total		$54 600		$61 400		$68 2000	

Payback Analysis for Cline-Server System Alternative

Cash flow description	Y0	Y1	Y2	Y3	Y4	Y5	Y6
Dev cost:	$104 950						
Operation & maintenance cost:		$11 270	$12 000	$13 000	$14 000	$15 000	$16 000
Discount factors for 12%	1.0	0.893	0.797	0.712	0.636	0.567	0.507
Time-adjusted benefits (current of present)	$104 950	$10 064	$ 9 564	$9 256	$8 904	$8 505	$9 120
Cumulative time adjusted costs over	$104 950	$115 014	$124 578	$133 834	$142 738	$151 234	$160 345

Benefits derived from operation of new	$0	$31 200	$39500	$47 800	$54 600	$61 400	$68 200
Discount factors for 12%	1.0	0.893	0.797	0.712	0.636	0.567	0.507
Time-adjusted benefits (current of present)	$0	$27861.6	$31481.5	34033.6	$34725.6	$34813.4	$34577.4
Cumulative time-adjusted benefits over	$0	$27861.6	$59353.1	$93386.7	$128112.3	$162926.1	$197503.5
	0	1	2	3	4	5	6
Cumulative lifetime time-adjusted costs+	-$104 950	$87152.4	$65224.9	$40447.3	$14625.7	$11692.1	$37158.5



Option 2:

Personnel:

1	System Analysts (400 h/ea $35/h)	$14 000
2	Programmer/Analysts (250 h/ea $25/h)	$12 500
1	GUI designer (200 h/ea $35/h)	$7000
1	Telecommunications Specialist (50 h/ea $45/h)	$2 250
1	System Architect (50 h/ea $45/h)	$4 500
1	Database Specialist (15 h/ea $45/h)	$600
1	System Librarian (250 h/ea $35/h)	$2500

Expenses:

2	Smalltalk training registration ($3500/student)	$7 000

Rent new hardware and software:

1	Dev Server (Pentium Pro class)	$1 500
1	Server Software (Operating system, misc.)	$500
1	DBMS server software	$500

Total development Costs: $52 850

PROJECTED ANNUAL OPERATING COST:

Personnel:

1	Programmer/Analysts (125 h/ea $25/h)	$3 125
1	System Librarian (20 h/ea $10/h)	$200

Expense

	Rent server	$1 500/ y
	Preprinted forms (15 000/y @.22/form	$3 300

Total projected annual operating cost: $8 125

Detail:

	Price	**Year 1**		**Year 2**		**Year 3**	
Customer/year							
		Quantity	**Total**	**Quantity**	**Total**	**Quantity**	**Total**
Subscription/person	$15	1000	$15 000	1200	$18000	1400	$21 000
Download/person	$18	400	$7 200	500	$9000	600	$10800
Sale CD/person	$20	300	$6000	400	$8000	500	$10000

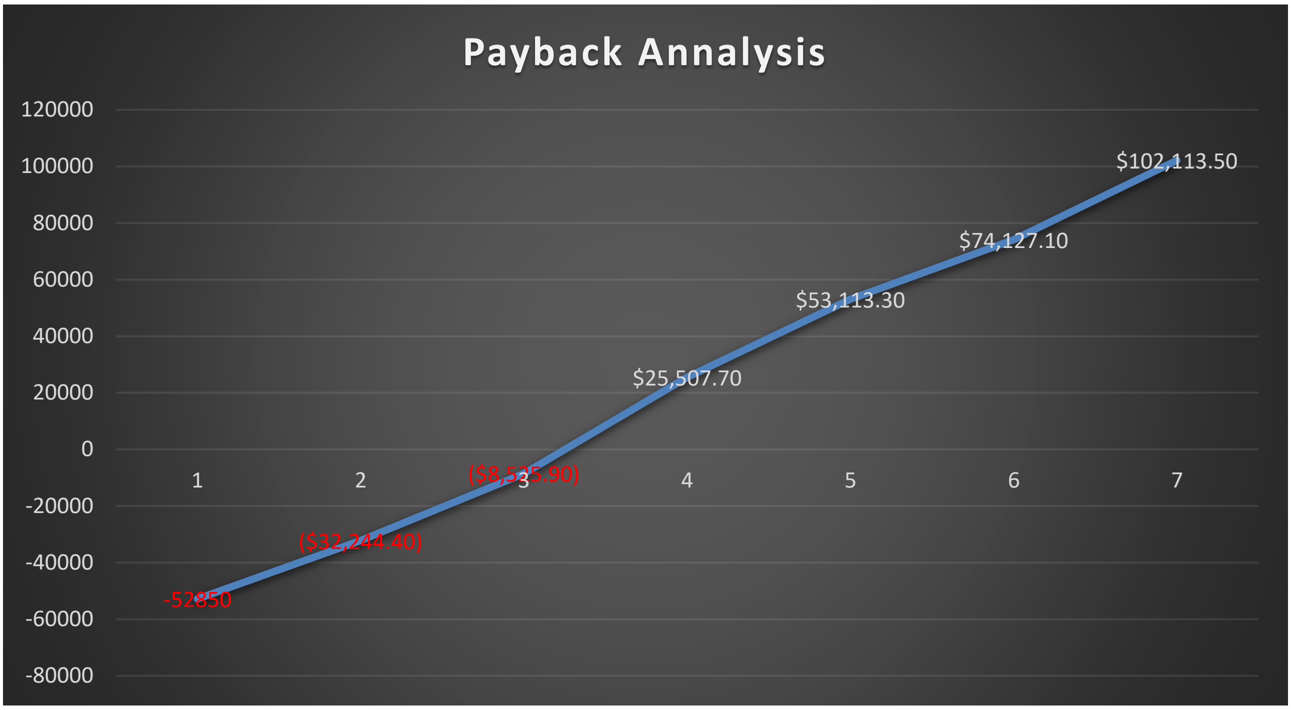
Download gift card/person	$15	200	$3000	300	$4500	400	$6000
Total		$31 200		$39500		$47 800	

	Price	**Year 4**		**Year 5**		**Year 6**	
Customer/year							
		Quantity	**Total**	**Quantity**	**Total**	**Quantity**	**Total**
Subscription/person	$15	1500	$22 500	1600	$24 000	1 700	$25 500
Download/person	$18	700	$12 600	800	$14 400	900	$16 200
Sale CD/person	$20	600	$12 000	700	$14 000	800	$16 000
Download gift card/person	$15	500	$7 500	600	$9 000	700	$10 500
Total		$54 600		$61 400		$68 2000	

Payback Analysis for Cline-Server System Alternative

Cash flow description	Y0	Y1	Y2	Y3	Y4	Y5	Y6
Dev cost:	$52 850						
Operation & maintenance cost:		$8 125	$9 000	$10 000	$11 000	$12 000	$13 000
Discount factors for 12%	1.0	0.893	0.797	0.712	0.636	0.567	0.507
Time-adjusted benefits (current of present)	$52 850	$10 064	$7 173	$7 120	$6 996	$6 804	$6 591
Cumulative time adjusted costs over	$52 850	$60 106	$67 879	$74 999	$81 995	$88 799	$95 390

Benefits derived from operation of new	$0	$31 200	$39500	$47 800	$54 600	$61 400	$68 200
Discount factors for 12%	1.0	0.893	0.797	0.712	0.636	0.567	0.507
Time-adjusted benefits (current of present)	$0	$27861.6	$31481.5	34033.6	$34725.6	$34813.4	$34577.4
Cumulative time-adjusted benefits over	$0	$27861.6	$59353.1	$93386.7	$128112.3	$162926.1	$197503.5
	0	1	2	3	4	5	6
Cumulative lifetime time-adjusted costs+	$52 850	$32244.4	$8525.9	$25507.7	$53 113.3	$74 127.1	$102 113.5



Feasibility Criteria	Wt.	Candidate 1	Candidate 2
Operational Feasibility **Functionality.** Describes to what degree the alternative would benefit the organization and how well the system would work. **Political**. A description of how well received this solution would be from both user management, user, and organization perspective.	30%	Fully supports user required functionality. Score: 60	Only basic functions are supported for users, other functions can be updated in the following updates. Score: 80
Technical Feasibility Technology. An assessment of the maturity, availability (or ability to acquire), and desirability of the computer technology needed to support this candidate. Expertise. An assessment to the technical expertise needed to develop, operate, and maintain the candidate system.	30%	Use the Symfony framework for development. Need to train staff to be able to use Symfony. Developing all the functionality requires a lot of programmers. Buy new servers and hardware. Score: 60	Using the PHP programming language, experienced staff do not need retraining. Renting servers and hardware saves on initial costs. The software will be updated continuously without causing boredom for customers. Score: 95

Feasibility Criteria	**Wt.**	**Candidate 1**	**Candidate 2**
Operational Feasibility	30%	Score: 60	Score: 80
Technical Feasibility	30%	Score: 60	Score: 90
Economic Feasibility **Cost to develop:** **Payback period (discounted):** **Net present value:** **Detailed calculations:**	30%	Approximately $104 950. Approximately 5.2 years. $102 113.5 See Attachment A.	Approximately $52 850. Approximately 2.8 years. $102 113.5 See Attachment A.
Schedule Feasibility. An assessment of how long the solution will take to design and implement.	10%	9 – 12 months. Score: 60	6 – 9 months. Score: 80
Ranking	100%	Score: 60	Score: 85

III. Explain the components of a feasibility report.

1. Net Present Value (NPV) in Project Management

The difference between the value of cash today and the worth of cash at a future date is known as net present value (NPV). In project management, net present value (NPV) is used to assess if the expected financial benefits of a project will surpass the current expenditure, indicating that the project is a worthy endeavor. An investment with a positive NPV is often lucrative and approved for consideration, whereas one with a negative NPV will likely result in a financial loss and is not permitted (Wrike, 2022).

2. What is 'Cash Flow'

Definition: Cash flow is the sum of all payments made to creditors by the company in the form of cash or cash equivalent. Often, the liquidity condition of the business is examined using cash flow analysis. It provides a quick overview of the amount of money moving into the company, where it is coming from, and how much is going out.

Description: Cash flows can be either positive or negative, as was mentioned. It is computed by deducting the cash amount at the beginning of a period, also referred to as the opening balance, from the cash balance at the conclusion of the period, also known as the closing balance (which could be a month, quarter, or year).

If the difference is positive, you will have more money at the conclusion of the specified time period. If the difference is negative, it indicates that you have less cash than you did at the beginning of the period as compared to the opening balance.

Cash flow statements are created in order to analyze the sources and destinations of the cash. Operating cash flow, which includes daily transactions, investment cash flow, which includes transactions made for business expansion, and financing cash flow, which includes transactions involving dividend payments to stockholders, make up its three primary categories.

However, the amount of cash flow is not the best statistic to use when evaluating a firm for investment. A company's income statements and balance sheet should both be carefully examined before drawing any conclusions.

A company's cash level can be rising since it may have sold certain assets, but it doesn't necessarily mean the liquidity is getting better. If the corporation has liquidated part of its assets to settle debt, this is a red flag that needs to be looked into further.

If the company is not reinvesting cash then this is also a negative sign because in that case it is not using the opportunity to diversify or build business for expansion (The Economic Times, 2022).

3. Break-Even Point (BEP)

The breakeven point (BEP), in financial terminology, is the point at which total costs and total revenues are equal. This indicates that there is now no profit or loss and that all necessary costs have been paid. To be more specific, the breakeven point (BEP) is the amount of revenues needed to cover all costs (fixed and variable costs).

The breakeven point is one of the most often employed methods in financial analysis. It is used by business owners, financiers, accountants, and even dealers. Businesses of all stripes, including those operating in the blockchain and cryptocurrency sectors, can benefit from BEPs. A corporation can be deemed to have begun to produce profits once it has managed to break above the breakeven threshold. Using a BEP analysis, firms may also determine how far they are from being profitable based on their income and fixed operating expenses.

Cryptocurrency traders in the blockchain industry can utilize breakeven point analysis to ascertain their current status of gains and losses and modify their trading methods as necessary. As a result, the BEP and the idea of breakeven multiple are closely connected.

Additionally, miners use BEP computations to assess the profitability of their mining operations. In this instance, it considers the price of the cryptocurrency that is currently being mined as well as the expenses of power and mining equipment (Taçoğlu, 2022).

$$\frac{|\ \text{beginningYear amount}\ |}{\text{endYear amount} + |\ \text{beginningYear amount}\ |}$$

IV. Assess the impact of each feasibility criterion on a software investigation.

1. Net Present Value (NPV)

Option 1:

	0	1	2	3	4	5	6
Cumulative lifetime time-adjusted costs+	$104 950	$87152.4	$65224.9	$40447.3	$14625.7	$11692.1	$37158.5

Option 2:

Year	0	1	2	3	4	5	6
Cumulative lifetime time-adjusted costs+	$52 850	$32244.4	$8525.9	$25507.7	$53 113.3	$74 127.1	$102 113.5

2. Cash flow

Option 1:

	Year 0	**Year 1**	**Year 2**	**Year 3**	**Year 4**	**Year 5**	**Year 6**
Cumulative time adjusted costs over	$104 950	$115 014	$124 578	$133 834	$142 738	$151 234	$160 345
Cumulative time-adjusted benefits over	$0	$27861.6	$59353.1	$93386.7	$128112.3	$162926.1	$197503.5

Option 2:

	Year 0	**Year 1**	**Year 2**	**Year 3**	**Year 4**	**Year 5**	**Year 6**
Cumulative time adjusted costs over	$52 850	$60 106	$67 879	$74 999	$81 995	$88 799	$95 390
Cumulative time-adjusted benefits over	$0	$27861.6	$59353.1	$93386.7	$128112.3	$162926.1	$197503.5

3. Break-Even Point (BEP)

Option 1:

Break-Even Point: 0.924233627

Therefore, the payback period is approx 4.9 years

Option 2:

Break-Even Point: 0.2505141977

Therefore, the payback period is approx 2.2 years

References

Ogbebor, O. (2011). *1. TECHNICAL FEASIBILITY 2. OPERATIONAL FEASIBILITY 3. ECONOMIC FEASIBILITY*. Retrieved from Osarome.blogspot.com: https://osarome.blogspot.com/2011/10/1-technical-feasibility-2-operational.html

Taçoğlu, C. (2022). *Break-Even Point (BEP)*. Retrieved from Binance Academy: https://academy.binance.com/vi/glossary/break-even-point

Team Clarizen. (2021). *What Are the Advantages of Waterfall Project Management?* Retrieved from Planview Blog: https://blog.planview.com/advantages-waterfall-project-management/

The Economic Times. (2022). *What is 'Cash Flow'*. Retrieved from The Economic Times: https://economictimes.indiatimes.com/definition/cash-flow

Thinkpositive Dubai. (2020). *Why to Conduct Feasibility Study?* Retrieved from Thinkpositive.ae: https://thinkpositive.ae/why-to-conduct-feasibility-study/

Thomas, C. (2022). *Five Steps of the Risk Management Process*. Retrieved from 360factors: https://www.360factors.com/blog/five-steps-of-risk-management-process/#:~:text=There%20are%20five%20basic%20steps,finally%2C%20the%20risk%20is%20monitored.

Tutorials Point . (2022). *SDLC - Spiral Model*. Retrieved from Tutorialspoint.com: https://www.tutorialspoint.com/sdlc/sdlc_spiral_model.htm

Tutorials Point. (2022). *Scrum - Framework*. Retrieved from Tutorialspoint.com: https://www.tutorialspoint.com/scrum/scrum_framework.htm

Tutorials Point. (2022). *SDLC - Waterfall Model*. Retrieved from Tutorialspoint.com: https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm

Tutorialspoint.com. (2022). *SDLC - V-Model*. Retrieved from Tutorialspoint.com: https://www.tutorialspoint.com/sdlc/sdlc_v_model.htm

Wrike. (2022). *What is Net Present Value (NPV) in Project Management?* Retrieved from wrike.com: https://www.wrike.com/project-management-guide/faq/what-is-net-present-value-npv-in-project-management/