ASSIGNMENT 1 FRONT SHEET

Qualification	**BTEC Level 5 HND Diploma in Computing**		
Unit number and title	**Unit 9: Software Development Life Cycle**		
Submission date	10/9/2020	**Date Received 1st submission**	
Re-submission Date	14/9/2020	**Date Received 2nd submission**	
Student Name	Bui Trung Duc	**Student ID**	BHAF190100
Class	BH-AF-1911-2.1	**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	Duc Bui Trung Duc

Grading grid

P5	P6	P7	P8	M4	M5	D3	D4

❒ **Summative Feedback:**	❒ **Resubmission Feedback:**

Grade:	Assessor Signature:	Date:
Signature & Date:		

Contents

Contents

Describe different software development lifecycles..... 4

Stages in the process..... 5

Preliminary analysis: The objective of phase 1 is to conduct preliminary analysis, come up with alternatives, describe costs and benefits and submit a preliminary plan with recommendations..... 5

System analysis, requirement definition: Define the project objectives into the defined functions and intended application activity. It is the process of gathering and interpreting facts, diagnosing problems, and proposing improvements to the system. Analyze the end user's information needs and also remove any inconsistencies and inadequacies in these requests..... 5

System design: detailed description of desired features and operations, including screen layout, business rules, process diagrams, pseudo code, and other documents..... 6

Development: The real code is written here..... 6

Integration and testing: Bring all the pieces into a special testing environment, then check for bugs, bugs, and interoperability. The code is tested at various levels in software testing. The following are the types of testing that may be relevant, depending on the type of system being developed:..... 6

Installation: The final stage of early development, where the software is put into production and run the actual business..... 7

Maintenance: During SDLC's maintenance, the system is evaluated to ensure it doesn't become outdated..... 7

Assess the merits of applying the Waterfall lifecycle model to a large software development..... 8

project.....8

Six phases of the waterfall model..... 8

Defining requirements:..... 8

Analysis:..... 8

Design:..... 8

Code:..... 8

Test:.....9

Operate:..... 9

Advantages of the waterfall model..... 9

Disadvantages of the waterfall model..... 9

When to apply Waterfall..... 9

Body post..... 10

Describe two iterative and two sequential software lifecycle models.(P1)..... 10

Application..... 10

Advantages..... 10

Defect..... 10

Explain how risk is managed in the Spiral lifecycle model.(P2).....10

Description..... 10

Model analysis..... 10

Application..... 11

Advantages..... 11

Defect..... 11

Describe the V-model..... 11

Description..... 11

Application..... 11

Advantages..... 11

Defect..... 12

Description..... 12

Application..... 12

Advantages..... 12

Defect..... 12

Which method is best for Tune Source and why?..... 12

Explain the purpose of a feasibility report.(P3)..... 13

Describe how technical solutions can be compared.(P4)..... 14

CONCLUSION..... 14

REFERENCES..... 14

A. Introduction

The software development life cycle (SDLC) is a process which is used to develop software. SDLC is a step by step procedure need to be followed by the organization to design and develop a high quality product. The phases of software development life cycle are which describes that how to develop, maintain particular software. The life cycle aims to develop a good quality product/software. SDLC produces intermediate products that can be reviewed to check whether they work according to customer requirement.

- SDLC is also known as Software development process.
- SDLC is an approach creates considerable documentation where this documentation helpful to make sure that requirement can be traced back to stated business requirements.
- It is a framework which has a set of tasks performed at each phase in the software development process.

SDLC:

SDLC is a step by step procedure or systematic approach to develop software and it is followed within a software organization. It consists of various phases which describe how to design, develop, enhance and maintain particular software.

It consists of various phases like requirement, feasibility study, design, coding, testing, installation and maintenance.

I. Describe different software development lifecycles

Software development process (SDLC), also known as lifecycle application development, is a term used in engineering systems, information systems, and software engineering to describe a programming process. plan, create, test, and deploy an information system. Several SDLC models or methods have been created, such as waterfall, spiral, Agile software development, rapid prototyping, increment and synchronization and stability. SDLC is used in the development of an IT project, it describes the various stages involved in the project from drawing, through the project completion.

II. Stages in the process

The system development lifecycle development framework provides a series of activities for system designers and developers to follow. It is comprised of a set of steps or stages in which each phase of the SDLC uses the results of the previous step.

SDLC adheres to the critical phases required for developers, such as planning, analysis, design, and implementation, and is explained in the following section. It includes an assessment of the current system, information gathering, feasibility study and approval request. Several SDLC models have been created: waterfall, fountain, spiral, construction and repair, rapid prototyping, augmentation, synchronization and stabilization. The oldest, and most famous, is the waterfall model: a series of stages in which the output of each stage becomes the input to the next. These stages can be described and divided in a variety of ways, including the following:

1. **Preliminary analysis: The objective of phase 1 is to conduct preliminary analysis, come up with alternatives, describe costs and benefits and submit a preliminary plan with recommendations.**

- Conduct a preliminary analysis: in this step you need to find out the objectives of the organization and the nature and scope of the problem under study. Even if a problem addresses only a small segment of the organization, you need to figure out what the purpose of the organization is. Then you need to see how the issue being studied is right for them.
- Suggest alternative solutions: In the process of delving into the organization's goals and specific problems, you may have mentioned several solutions. Alternatives may come from interviewing employees, customers, suppliers and / or consultants. You can also research what your competitors are doing. With this data, you will have three options: leave the system as is, improve it, or develop a new system.
- Describe the costs and benefits.

2. **System analysis, requirement definition: Define the project objectives into the defined functions and intended application activity. It is the process of gathering and interpreting facts, diagnosing problems, and proposing improvements to the system. Analyze the end user's information needs and also remove any inconsistencies and inadequacies in these requests.**

- Data Collection: End-user requests obtained through documentation, customer interviews, observations and questionnaires,
- Examine the current system: Identify the pros and cons of the current system in place, to transfer the pros and cons in the new system.
- Recommended system analysis: Solution for shortcomings in step two found and any user-specific recommendations used to prepare the specifications.

3. **System design: detailed description of desired features and operations, including screen layout, business rules, process diagrams, pseudo code, and other documents.**

- In system design, design functions and activities are described in detail, including screen layout, business rules, process diagrams, and other documents. The output of this stage will describe the new system as a collection of modules or subsystems.
- The design phase takes as its initial input the requirements identified in the approved request document. For each request, a set of one or more design elements will be generated as a result of interviews, work shops and/or sample attempts.
- Design elements describe desired system features in detail, and often include functional hierarchy diagram, screen layout, business rules table, business process diagram joint, pseudo code, and an entity completion diagram with a complete data dictionary. These design elements are intended to describe the system in detail, so that skilled engineers and engineers can develop and supply the system with minimal design input.

4. **Development: The real code is written here.**

- Upon receipt of the system design documentation, the work is divided into modules / units and real code begins to code. Since then, in this stage the code is generated hence it is the main focus for developers. This is the longest stage of the software development life cycle.

5. **Integration and testing: Bring all the pieces into a special testing environment, then check for bugs, bugs, and interoperability. The code is tested at various levels in software testing. The following are the types of testing that may be relevant, depending on the type of system being developed:**

- Defect testing the failed scenarios, including defect tracking
- Path testing
- Data set testing
- Unit testing
- System testing
- Integration testing
- Black-box testing
- White-box testing
- Regression testing
- Automation testing
- User acceptance testing
- Software performance testing

6. **Installation: The final stage of early development, where the software is put into production and run the actual business.**

- Deployment of the system includes changes and improvements prior to the system shutdown or sunset. System maintenance is an important aspect of SDLC. When key personnel change positions in the organization, new changes will be made. There are two ways to develop the system; take traditional (structured) and object oriented approaches. Information Engineering includes the traditional systems approach, also known as structural analysis and design techniques. The object oriented approach views the information system as a collection of objects that are integrated together to create a complete and complete information system.

7. **Maintenance: During SDLC's maintenance, the system is evaluated to ensure it doesn't become outdated.**

- Once the customer starts using the evolved system then real problems emerge and need to be resolved over time.
- This process where care is taken for product development is called maintenance.

III. Assess the merits of applying the Waterfall lifecycle model to a large software development project.

1.1. Six phases of the waterfall model

The actual application of the waterfall model in a project is a fairly straightforward process, thanks in large part to the step-by-step feature of the model itself. Depending on the developer (or at the time) there are slight variations in the numbers and details of the steps in a waterfall model. But for the most part, all concepts are the same and have a broad view of the steps to take with an idea and developing a complete application.

1.2. Defining requirements:

With the first phase, the possible requirements of an application are analyzed systematically with the aim of creating a specific document for future development. The result to be achieved in this phase is a document describing the requirements that define what the application will perform, but not specifically how it will function.

1.3. Analysis:

In the next phase, the system is analyzed so that a suitable model and system logic can be produced that will be used in the application.

1.4. Design:

This phase largely deals with technical design requirements, such as programming languages, data layers, services, etc. A typical design will be completed as specifically as possible. It will describe exactly how the system logic mentioned in the analysis will be executed.

1.5. Code:

The final coding work is done in this fourth phase, which implements all the models, logic of the system, and integration services that have been clarified in the previous phases.

1.6. Test:

At phase five, the QA, Beta tester, and all testers will search and report the bugs in the system that need to be fixed. Usually when this phase there will be some repetitive (but necessary) work of the Coding phase, so that the detected technical errors will be completely resolved.

1.7. Operate:

Eventually, the application will be deployed in the real environment. However, the operation phase is not just about getting the project out of the way, it also includes support and maintenance to keep the application up to date.

IV. Advantages of the waterfall model

Although the waterfall model has gradually disappeared over the past few years in favor of more agile models, it still offers a number of benefits, especially in large projects and organizations that need Completion stages and stages of work lie within these waterfalls.

- Adapts well to flexible groups
- Imposing a well structured organization
- Suitable for landmark-oriented projects

V. Disadvantages of the waterfall model

Although a few times when Dr. Royce first announced it, the waterfall model was considered a major breakthrough in 1970. After more than four centuries, a few major downsides have shown why the difficult model is still desirable. waited as expected and replaced by today's Agile models.

- Poor adaptive design constraints
- Ignore user feedback at the following stages
- Delayed testing time

VI. When to apply Waterfall

- Apply Waterfall when it best understands the project's requirements, requirements are clear and have high stability.
- Mastering the developed technology.
- There are no ambiguous requirements.
- Rich development resources and high technical expertise.
- Suitable for small and short term projects.

B. Body post

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

A pattern is repeated from start to completion of the 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 spec spec and then starting to execute, this model can be reviewed gradually to come to final requirement.

1.1. Application

The main requirement must be specified; however, some functions or improvements may evolve over time.

A new technology is being used and is being learned by the development team while working on the project.Suitable for large projects and mission critical.

1.2. Advantages

- Build and perfect step-by-step product steps. Document preparation time will be less than design time. Some work functions can be developed quickly and early in the life cycle.
- Less expensive when changing the required range.

- Easy to manage risk.
- Throughout the life cycle, software is produced early to facilitate customer reviews and feedback.

1.3. Defect

- Weak need many resources.
- System design or architecture problems can arise at any time.
- More complex management requirements.
- The progress of the project depends heavily on the stage of risk analysis.

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

2.1. Description

- Model is a combination of the features of prototyping model and waterfall model.
- The spiral pattern is preferred for large, expensive and complex projects.
- This model uses the same stages as the waterfall model, in terms of order, plan, risk assessment, ...

2.2. Model analysis

- The phases in the spiral development process include:
- Objective identification- Set goals: identify the goals and objects for each phase of the project.
- Alternate evaluation- Risk assessment and mitigation: assessing risks and implementing actions to minimize risks.
- Product development - Product development: Select a suitable model for system development.
- Next phase planning- Planning: project evaluation and planning for the next phase.

2.3. Application

➔ This model is often used for large applications and systems built in small phases or in segments.

2.4. Advantages

- Good for large scale software systems.
- Easy to control risks at each evolution level.
- The assessment is more realistic as a workflow, because important issues were discovered earlier.

2.5. Defect

- Manager needs good skills to manage projects and assess risks in a timely manner.
- High cost and time consuming to complete the project.
- Complicated and unsuitable for small and low-risk projects.
- Asking for frequent changes leads to infinite iterations.
- Not widely used.

3. Describe the V-model

3.1. Description

The V-model is an extension of the waterfall model and is based on a combination of a test phase for each respective development stage. This is a highly disciplined model and the next stage only begins after completing the previous stage.
With V model, the test is involved from the very beginning.

3.2. Application

- Application
- Requirements are clearly defined.
- Determine the stable product.
- Technology remains unchanged and is well understood by the project team.
- There are no unspecified or unspecified requirements.
- Short project.

3.3. Advantages

- This is a highly disciplined model and all phases are completed at the same time.
- Works well for small projects, when the requirements are well understood.
- Simple and easy to understand and easy to use, easy to manage.

3.4. Defect

- It is difficult to manage and control risks, high risks.
- Not a good model for complex and object-oriented projects.
- Poor model for long and ongoing projects.
- Not suitable for projects with medium to high risk of change of requirements.

4. Describe the Rapid model

4.1. Description

- The RAD model is a software development method that uses minimal planning in favor of rapid prototyping.
- Functional modules are developed in parallel as prototypes and are integrated to create finished products for faster product delivery.
- Ensure that the developed prototypes are reusable.

4.2. Application

The RAD model can be successfully applied to projects:

- Clearly modularized. If the project cannot be divided into modules, RAD may fail.
- RAD should be used when the need arises to create a system with changing customer requirements in a small 2-3 month period.
- Should be used when a designer is available for the model and the cost is high.

4.3. Advantages

- Reduce development time.
- Increased reuse of ingredients.
- Make an initial assessment quickly.
- Encourage customers to give feedback**.**

4.4. Defect

- The level of the group requires a certain skill.
- Only systems with modules can use this model.

4.5. Which method is best for Tune Source and why?

➔ The feasibility studies aim to objectively and reasonably uncover the strengths and weaknesses of the proposed business or venture, opportunities and threats as presented by environments, resources. performance requirements through, and ultimately the outlook for success. In its simple terms, two criteria to judge the feasibility of the required price and the value to achieve. As such, a well-designed feasibility study should provide a historical context of the business or project, description of the product or service, reporting accounting, details of operations and management. , market research, policy, financial data, legal requirements and tax liability. In general, feasibility studies ahead of technical development and implementation project.

➔ Possible technology and systems

The assessment is based on a design outlining the Input, Process, Output, Fields, Program, and Procedure System Requirements. This can be quantified in terms of volume of data, trends, frequency of updates, etc. to estimate whether the new system will fully perform. Technological feasibility is performed to determine whether the company has the software, hardware, HR and expertise capabilities of the project to complete. When writing a feasibility study the following should be taken into consideration:

- A brief description of the business to evaluate the factors that could affect the study.
- Parts of the business are examined.
- Human and economic factors.
- The solutions to the problems.

5. **Explain the purpose of a feasibility report.(P3)**

A feasibility report is a testimony that attempts to create some sort of action. Feasibility reports are created to persuade/help the decision makers to choose between available options. Remember that your option is not the only one, the decision makers will probably have many to choose from. A feasibility report also determines whether or not the investigated task can be done with the amount of resources available OR how many resources will be necessary in order to complete the task. A feasibility may be useful in a lot of different situations such as event planning, finances, or even remodeling your home.

A feasibility study is a way to evaluate the practicality and desirability of a project. Before a company invests time and money into a project, they need to know how successful the project will be before investing. Sometimes companies want to understand input costs, the amount of research that will need to be done, or even the marketability of a project. With input prices, it is essential that companies understand, (even before they put time and research into the project), how much it would cost to go through with their product. Companies also like to know if they put time into research and go through with their change or promotion of a product, how the public/people will react to the change. Will people be fighting over the new product or will it fall through. The purpose of feasibility studies is to provide companies information and analysis on whether or not you or your company should pursue this course of action.

6. **Describe how technical solutions can be compared.(P4)**

- Define the business need and goal .Write out a sentence or two on what the issue is and your goal, and how a new technology solution may help you achieve the goal.
- Determine if the goal could be achieved without technology. Posing such a simple challenge ensures an existing solution or adjustment to practices couldn't achieve the same goal, avoiding the additional cost and change management that comes with new technology.
- Conduct a simple return on investment (ROI) analysis. See how costs add up and what you'll get for the money by looking at the benefits of making such an investment.
- Compare at least three vendors
- These steps offer more of a scientific approach to evaluating technology solutions. But the reality is you will also get information (solicited or not) from vendors, friends, business associates and even articles. While all input is worth considering, validating what your analysis tells you vs what you hear vs what your business needs are can keep emotions and biases in check and allow the wisest decisions.

➔ A comparative model will either validate your decision or expose the need to look at other technology solutio Write down your minimum requirements (needs) and "nice to haves" (wants). Rank each vendor on each want and need using a score of 1-10 (10 being highly needed/wanted and 1 being a minor need/want) and compare scores. Your needs should rank highly with eligible vendors. Use wants as a tiebreaker, if necessary.

C. CONCLUSION

- ➔ Some argue that SDLC no longer applies to models like Agile, but it's still a widely used term in the tech world. rather than having to apply best practices from the SDLC model and apply it to whatever is best suited for the software to be designed.
- ➔ Software Development Life Cycle (SDLC) is a follow-up process for a software project, within a software organization. It includes a blueprint that describes how to develop, maintain, change, or upgrade specific software.
- ➔ Process is one of the extremely important factors that bring success to software manufacturers, it helps all project members from old to new, inside or outside the company can handle Management synchronizes the work corresponding to its position through the general method of the company, or at least at the project level.

D. REFERENCES

1) **https://www.tutorialspoint.com/sdlc/sdlc_overview.htm**
2) **http://istqbexamcertification.com/what-are-the-software-development-life-cycle-sdlc-phases/**
3) **https://www.tutorialspoint.com/sdlc/sdlc_overview.htm**
4) **http://istqbexamcertification.com/what-are-the-software-development-life-cycle-sdlc-phases/**
5) **How to Evaluate Technology Solutions | Gordon Food Servicewww.gfs.com › en-us › ideas › how-evaluate-technolog...**
6) **Feasibility Study: Types and Importance in Project Managementwww.simplilearn.com › Resources › Project Management**
7) **p2.docx - Explain how risk is managed in the spiral lifecycle ...www.coursehero.com › file**
8) **https://melsatar.blog/2012/03/15/software-development-life-cycle-models-and-methodologies/**