**ASSIGNMENT 01 FRONT SHEET**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | |
| **Unit number and title** | Unit 09: Software Development Life Cycle | | |
| **Submission date** |  | **Date Received 1st submission** |  |
| **Re-submission Date** |  | **Date Received 2nd submission** |  |
| **Student Name** |  | **Student ID** |  |
| **Class** |  | **Assessor name** |  |
| **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:** | | |

|  |
| --- |
| **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.  1. 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.  1. 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.  1. 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.  1. 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. |

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

Contents

**No table of contents entries found.**

# I. Introduction

The Software Development Life Cycle (SDLC) is the procedure for establishing how an information system can support company requirements, developing the system, constructing it, and providing it to users. It is a structure that specifies a list of duties to be carried out at each step of the software development process. By ensuring that all functionality, user needs, goals, and end objectives are addressed jointly, it is a process that provides a comprehensive understanding of the development, design, and maintenance of a software project. The software development life cycle (SDLC) improves the integrity of the software endeavor as a whole. This might appear easy if you've attended a coding lesson or have programmed before. However, in reality, it is not so simple. So, in my assignment, I will introduce clearly about how SDLC is important.

# II. Describe two iterative and two sequential software lifecycle models

## 1. Describe the following SDLC

A software company will follow the SDLC procedure when working on a software project. It consists of a thorough strategy outlining the creation, upkeep, replacement, and modification or improvement of a particular piece of software. A technique for enhancing software quality and the entire development process is defined by the life cycle. There are six stages in SDLC:

* **Planning and Requirement Analysis:** In the SDLC, the requirement analysis step is the most crucial and essential. With feedback from the client, the sales department, market surveys, and domain specialists in the industry, it is carried out by the senior members of the team. The fundamental project strategy is then planned using this data, and a product feasibility study is then carried out in the financial, operational, and technological domains. During the planning stage, it is also done to identify project risks and prepare for the needs for quality assurance. The technical viability study's conclusion defines the various technical strategies that can be used to carry out the project effectively and with the fewest possible risks.
* **Defining Requirements:** Following completion of the requirement analysis, the next stage is to precisely describe and record the product requirements and obtain client or market researcher approval. An SRS (Software Requirement Specification) document, which includes all the product specifications to be created and developed throughout the project life cycle, is used to accomplish this.
* **Designing the Product Architecture:** The ideal architecture for a product to be created should be based on SRS, according to product architects. A DDS - Design Record Specification is typically used to suggest and record multiple design approaches for the product architecture based on the criteria outlined in the SRS. All of the suggested architecture's modules should have their internal designs, down to the smallest aspects, explicitly specified in DDS.
* **Building or Developing the Product:** During this phase of the SDLC, the product is actually developed and constructed. At this point, the programming code is produced in accordance with DDS. Different high level computer languages are used for writing, including C, C++, Pascal, Java, and PHP.
* **Testing the Product:** This stage is typically a subsection of all the stages because, in contemporary SDLC models, testing activities are primarily integrated throughout all stages. This stage, however, only pertains to the testing phase of the product, during which product flaws are discovered, recorded, corrected, and retested until the product satisfies the SRS's quality standards.
* **Deployment in the Market and Maintenance:** The product is officially launched in the relevant market once it has undergone testing and is prepared for deployment. After a product is put on the market, upkeep is carried out for the clientele already in place. The product may then be published as is or with recommended improvements in the target market group depending on the input.

(TutorialsPoint, 2022)

### a. Waterfall model

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



*Waterfall model*

#### Different phases of Waterfall model in Software Engineering

|  |  |
| --- | --- |
| **Phases** | **Activities performed in each stage** |
| Requirement Gathering and analysis | All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document. |
| System Design | In this phase, the first phase's prerequisite specifications are examined, and the system architecture is created. In addition to describing the general system architecture, this system design aids in defining the hardware and system needs. |
| Implementation | With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing. |
| Integration and Testing | Following the testing of each element created during the implementation phase, the entire system is integrated. The complete system is evaluated for errors and malfunctions after integration. |
| Deployment of system | Once the product has undergone functional and non-functional testing, it is either distributed to the market or implemented in the customer's environment. |
| Maintenance | Various problems can arise in a customer setting. Patches are issued to address those problems. Additionally, improved versions of the merchandise are published. |

#### Advantages and Disadvantages of Waterfall model

|  |  |
| --- | --- |
| **Advantages** | **Disadvantages** |
| Simple and easy to understand and use | No working software is produced until late during the life cycle. |
| Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a review process. | High amounts of risk and uncertainty. |
| Phases are processed and completed one at a time. | Not a good model for complex and object-oriented projects. |
| Works well for smaller projects where requirements are very well understood. | Poor model for long and ongoing projects. |
| Clearly defined stages. | Not suitable for projects where requirements are at a moderate to high risk of changing. So, risk and uncertainty is high with this process model. |
| Well-understood milestones. | It is difficult to measure progress within stages. Cannot accommodate changing requirements |
| Easy to arrange tasks. | Adjusting scope during the life cycle can end a project. |
| Process and results are well documented. | Integration is done as a "big bang. at the very end, which doesn't allow identifying any technological or business bottlenecks or challenges early. |

### b. V-model

The V-model is an SDLC paradigm in which operations are executed sequentially in the form of a V. Another name for it is the Verification and Validation paradigm. The V-Model, which is a refinement of the waterfall model, is predicated on the connection of a testing period with each related development step. As a result, a testing process has a direct relationship with each and every part of the development cycle. Since each step must be finished before moving on to the next, this approach requires extreme discipline. (TutorialsPoint, 2022)



#### Different phases of the Software Development Cycle

|  |  |
| --- | --- |
| **Different phases** | **Activities performed in each stage** |
| **Requirement Gathering**  **stage** | Obtain as much information as you can about the client's requirements for the preferred program. It is only the requirements collecting step at this point. |
| **Design Stage** | Prepare database, such as Oracle or MySQL, and computer languages like Java, PHP, and.net. Along with some high-level tasks & architecture, which would be appropriate for the undertaking. |
| **Build Stage** | After the planning phase, the software is actually coded during the construction phase. |
| **Test Stage** | The software is then tested to ensure that it was created in accordance with the client's requirements. |
| **Deployment stage** | Deploy the application in the respective environment |
| **Maintenance stage** | You might need to subsequently modify the code in response to a customer request once your system is available to use. |

#### Advantages and Disadvantages of V-Model

|  |  |
| --- | --- |
| **Advantages** | **Disadvantages** |
| This is a highly-disciplined model and Phases are completed one at a time | High risk and uncertainty and poor model for long and ongoing projects. |
| Works well for smaller projects where requirements are very well understood. | Not a good model for complex and object-oriented projects. |
| Simple and easy to understand and use. | Not suitable for the projects where requirements are at a moderate to high risk of changing. |
| Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a review process. | 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. |