

ASSIGNMENT 02 FRONT SHEET

Qualification	BTEC Level 5 HND Diploma in Computing		
Unit number and title	Unit 09: Software Development Life Cycle		
Submission date	9/9/2022	Date Received 1st submission	
Re-submission Date		Date Received 2nd submission	
Student Name	Phạm Thế Huy	Student ID	GCC200190
Class	GCC0904	Assessor name	Nguyễn Thái 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	Huy

Grading grid

P5	P6	P7	M3	M4	M5	M6	D3	D4

☐ **Summative Feedback:**

☐ **Resubmission Feedback:**

Grade:

Assessor Signature:

Date:

Internal Verifier's Comments:

Signature & Date:

Assignment Brief 02 (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:	Undertake a software development life cycle
Issue Date:	07/12/2020
Submission Date:	
Internal Verifier Name:	
Date:	

Submission Format:
<p><i>Format:</i></p> <p>The submission is in the form of 1 document11You 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 style11<i>Submission:</i></p> <p>Students are compulsory to submit the assignment in due date and in a way requested by the Tutor11The form of submission will be a soft copy posted on http://cms.greenwich.edu.vn/11Remember to convert the word file into PDF file before the submission on CMS11<i>Note:</i></p> <ol style="list-style-type: none"> 1. The individual Assignment must be your own work, and not copied by or from another student11 If you use ideas, quotes or data (such as diagrams) from books, journals or other sources, you must reference your sources, using the Harvard style11Make 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:
<p>LO3 Undertake a software development lifecycle11LO4 Discuss the suitability of software behavioural design techniques.</p>

Assignment Brief and Guidance:

Tasks

At this stage, you have convinced Tune Source to select your project for development. Complete the following tasks to analyse and design the software

Task 1 – Analysis (1)

Identify the stakeholders, their roles and interests in the case study
 Review the requirement definition of the project. Clearly indicate which stakeholder(s) provide what requirements
Word limit: 150 – 200
 Identify FRs and NFRs of Tune Source Project
 Discuss the relationships between the FRs and NFRs
Word limit: 300 – 400 words
 Discuss the technique(s) you would use to obtain the requirements
 If needed, you may state suitable additional assumptions about the project in order to justify the technique(s) that you choose
Techniques: JAD, Interview, Observation, etc
 Demonstrate how to collect requirements based on chosen technique
Word limit: 700 – 1000
 Discuss how you would trace these requirements throughout the project by using Requirement Traceability matrix. You will have to provide real usage of it
Word limit: 400 – 500 words

Task 2 – Analysis (2)

Analyze the requirements that you identified in Task 1 using a combination of structural and behavioral modelling techniques that you have learnt
Scope: You only need to construct following items for the system. You will have to include:

Use Case Diagram for the whole system
 Use Case specification for 2 Use cases
 Context Diagram for the whole system
 Data Flow Diagram – Level 0 for the whole system
 ERD for the whole system
 For each diagram, you will have to explain properly
Word limit: 1000 – 1200 words

Task 3 – Design

Based on the analysis result, discuss how you would conduct the design phase:

Discuss how the user and software requirements are addressed in the design phase
 You will explain how Mock-up, and Wireframe are used in the project. You should include some of the mockup or wireframe (at least 5) design of the Tune Source project to justify that it matches users' requirements
 You will explain which architecture (client – server, n-tier, microservices, etc.) is suitable for the project with clear illustrations and why
 Then you will address which technical solution stack could be suitable to implement the project with clear explanations
 Discuss how activity diagram and pseudocode are used to specify the software behaviour
 Discuss how UML state machine can be used to specify the software behaviour. Differentiate between FSM and extended FSM using the case study
 Discuss how the data-driven approach improves the reliability and effectiveness of software
Word limit: 800 – 1500

Task 4 – Software quality management

Discuss two software quality attributes that are applicable to the project
 Discuss two quality assurance techniques that can help improve the software quality in the project
 Discuss how the design techniques and approaches that you have used can help improve the software quality
Word limit: 400 – 1500.

Learning Outcomes and Assessment Criteria (Assignment 02):			
Learning Outcome	Pass	Merit	Distinction
LO3 Undertake a software development lifecycle	P5 Undertake a software investigation to meet a business need11 P6 Use appropriate software analysis tools/techniques to carry out a software investigation and create supporting documentation.	M3 Analyse how software requirements can be traced throughout the software lifecycle11 M4 Discuss two approaches to improving software quality.	D3 Critically evaluate how the use of the function design paradigm in the software development lifecycle can improve software quality.
LO4 Discuss the suitability of software behavioural design techniques	P7 Explain how user and software requirements have been addressed.	M5 Suggest two software behavioural specification methods and illustrate their use with an example11 M6 Differentiate between a finite state machine (FSM) and an extended-FSM, providing an application for both.	D4 Present justifications of how data driven software can improve the reliability and effectiveness of software.

Table of Contents

Assignment Brief 02 (RQF)	1
Table of Table	5
<i>I. Undertake a software investigation to meet a business need (P5)</i>	6
1. Identify the stakeholders, their roles and interests in the case study	6
a) Requirement	6
b) <i>Discuss the relationships between the FRs and NFRs.</i>	7
c) <i>The techniques are used to obtain the requirements</i>	8
d) <i>Requirement Traceability matrix</i>	10
References	12

Table of Table

Table 1: Stakeholder of Tune Source	6
Table 2: Interview Question	9

ASSIGNMENT 2

I. Undertake a software investigation to meet a business need (P5)

1. Identify the stakeholders, their roles and interests in the case study

a) Requirement

Table 1: Stakeholder of Tune Source

No.	The Stakeholder	Roles	Interest
1	Co-founder: John Margolis, Megan Taylor, and Phil Cooper	As the highest managers of the Tune Source project	John Margolis , Megan Taylor and Phil Cooper will be the beneficiaries of Tune Source business.
2	Carly Edwards, Assistant Vice President, Marketing	John Margolis , Megan Taylor and Phil Cooper will be investors in the project's capital.	They will be split the profits from the products sold, which can be marketed on the Tune Source website.
3	The IT department	Website operations	Get a salary from the company.
4	Customer	Carry out shopping activities, bring in the number of subscribers, the number of downloads, the number of people using the site is large, increase revenue and grow the Company. Give feedback to develop the website.	Get good music back, a great experience when visiting the site.

- Project manager (Co-founder):

+As a project manager, I want the system to fully meet the company's requirements, without any errors so that the user experience is improved.

+As a project manager, I want the system to be completed within 3 months so that it can be released to the market this fall11Sponsor:

+As a sponsor, I want to be advertised, introduced on Tune Source website so that many people know me.

+As a sponsor, I want the site to have good, quality music that increases the number of users and revenue

- IT department:

+As an IT department, I want to have an admin login account so that I can login and manage the Tune Source website.

+As an IT department, I want CRUD function so that I can view, add, update, delete tunes.

As an IT department, I want the system to give an error when it encounters any error so that the system runs smoothly.

+As an IT department, I want functions like search, filter so that I can find music more easily.

+As an IT department, I want statistical functions so that I can see metrics like revenue, number of subscribers

- Customer:

+As a customer, I want registration and login functionality so that I can buy tracks11+As a customer, I want my account to be secure so that my information is not exposed.

+As a customer, I want the system to have an easy-to-use interface so that shopping is easier11

+As a customer, I want good quality website compared to that fast access speed.

+As a customer, I want to see a list of tracks so that I can choose the music I want.

+As a customer, I want the search function so that I can select tracks faster.

+As a customer, I want the listening function so that I know the quality of that track.

+As a customer, I want download function so that I can download selected track.

+As a customer, I want the gift card buying function so that I can get the best deals from it.

+As a customer, I want the review function so that to send my experiences and develop the website.

+As a customer, I want 2 separate functions i.e. fixed and unlimited downloads so that's convenient for downloading music or saving money.

b) Discuss the relationships between the FRs and NFRs.

* FRs: A functional requirement defines a system or its component in software engineering. It specifies the tasks the program must complete. A function is made up of three parts: inputs, behavior, and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other function that determines what a system is likely to do. Functional software requirements assist you in capturing the system's intended behavior. This behavior can be represented in terms of functions, services, or tasks that a system must accomplish.

* **NFRs:** A software system's quality characteristic is defined by a non-functional need. They are a set of criteria used to judge and evaluate a system's specific operation. A non-functional requirement is required to ensure the overall software system's usability and effectiveness. Non-functional requirements that are not met can lead to systems that do not meet user needs. Non-functional Requirements allow you to place constraints or limitations on the system's architecture across several agile backlogs. When there are more than 10000 simultaneous users, the site should load in 3 seconds. Non-functional requirements must be described just as carefully as functional requirements.

* **Relationship:** FRs and NFRs requirements lay the foundation for a software development project's success. To stimulate software development, specific qualities were mentioned in functional requirements. Software features are directly linked to non-functional requirements. Non-functional requirements are product criteria that define how a project should be implemented to produce an end-user experience from the user's perspective. When the required product features are determined and the requirements for each of those features are established, the software development process is considerably accelerated. The project will suffer greatly if the functional and nonfunctional requirements are not specified. NFRs explain a product's or application's entire experience, such as security or performance, whereas FRs describe a specific function. User stories, use cases, and functional scores can all be used to collect FRs. NFRs are frequently found throughout a product, particularly in the user experience and user interface (UI/UX). On the same project, both NFR and FR must exist side by side.

c) The techniques are used to obtain the requirements

***JAD method:** Joint Application Development (JAD) is a method for gathering business and user needs while creating new information systems for a firm. The JAD process may also include methods for increasing user participation, accelerating development, and improving specification quality. A JAD session's goal is to bring together subject matter experts, business analysts, and IT specialists to develop solutions. A business analyst engages with the entire group, gathering information, analyzing it, and producing a document. In the JAD session, he is extremely crucial. JAD sessions are highly structured, led workshops that bring together customers and IT personnel to develop high-quality deliverables in a short amount of time. In other words, a JAD Session allows customers and developers to swiftly agree on the project's essential scope, objectives, and specifications, or to disagree, indicating that the project should be re-evaluated.

- **Simplify:** It consolidates months of meetings and phone calls into a structured workshop.
- **Identify:** Issues and participants
- **Quantify:** Information and processing needs

- Clarify: Crystallize and clarify all requirements agreed upon in the session.
- Unify: The output from one phase of development is input to the next.
- Satisfy: The customers define the system; therefore, it is their system. Shared participation brings a share in the outcome; they become committed to the systems success.

***Interview method:** The interview is the most typical method for gathering requirements. The interview process has five fundamental steps: selecting interviewees, designing interview questions, preparing for the interview, conducting the interview, post interview follow-up. The interviewer asks stakeholders questions to acquire information in this method. Face-to-face interviews are the most popular technique. In a structured interview, the interviewer asks a list of predetermined questions. When the interviewer does not follow a set framework or asks specified questions, it is called an unstructured interview.

***Suitable method:** The best method for this Tune source project is Interview because it will be planned in advance with a detailed, purposeful question system, in addition, the answer is not limited to yes or no but rather an open answer. Users of Tune Source website are not only internal but mainly music enjoyment customers, a workshop of JAD method will not be able to get all the requirements of users. In addition, the cost of the Interview method is also quite low compared to opening JAD workshops.

Table 2: Interview Question

Type of Question	Example
Closed-Ended questions	<p>-What statistical functions do you want the website to have?</p> <p>-What technology do you want the website to be secured with?</p>
Opened-ended questions	<p>-What do you think about the look and feel of the current website?</p> <p>-What are the problems you usually have when downloading music on a website?</p> <p>-What special functions do you want the website to have?</p> <p>-How much money do you usually spend on enjoying music?</p>

Probing questions	What is your favorite website, why do you like it? Does the site have a music download function with a promotional price?
-------------------	--

d) Requirement Traceability matrix

*What is a Requirements Traceability Matrix (RTM)?

A requirements traceability matrix is a document that demonstrates the relationship between requirements and other artifacts. It's used to prove that requirements have been fulfilled. And it typically documents requirements, tests, test results, and issues.

*Why is Requirement Traceability Important?

Requirements traceability is important to effectively manage your requirements.

- **Meeting Goals:** When done well, traceability follows the life of a requirement. It starts at the time a requirement originates. And it continues on through fulfillment of the requirement. So, it makes sure that your requirements fulfill your original goals. For example, it gives you proof that you met compliance requirements.
- **Running the Right Tests:** Requirement traceability also helps your quality assurance (QA) team understand what needs to be tested. This improves test coverage by mapping test cases back to each requirement. So, QA will be able to test all of the right things. And, as a result, you'll be able to show that your requirements have been properly implemented.
- **Making Decisions:** Traceability can also be used for decision-making throughout product development. You'll be able to understand how product design will be impacted by requirements. And, if a requirement changes, you'll be able to analyze the impact of that change across development.
- **Managing Projects:** Traceability is also useful for managing projects. You'll know exactly how far you've progressed. And you'll be able to manage the scope of your requirements. By linking your requirements to tests, you'll understand how you can realistically meet those requirements and still ship on time.

*How to Create a RTM ?

-You can create a RTM in Microsoft Excel. Or you can use specialized tools to accelerate the process.

-There are three basic steps — no matter which tool you use.

- Define your goals.
- Establish your artifacts (and their relationships).

- Fill in the traceability matrix.

*Benefits of Using a Traceability Matrix:

There are six key benefits of using a traceability matrix.

- Get visibility across development.
- Make better decisions (e.g., on requirements change).
- Accelerate release cycles.
- Rest easy knowing your requirements are fulfilled.
- Prove compliance faster.
- Pass audits without fear (perforce, n.d.)

References

perforce, n.d. *perforce.com*. [Online]

Available at: <https://www.perforce.com/resources/alm/requirements-traceability-matrix>