ASSIGNMENT 02 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	Bùi Hương Linh	**Student ID**	GBH200662
Class	GCH1002	**Assessor name**	Đỗ Tiến Thành
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	Linh

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:

Table of Contents

Introduction

Tune Source has approved the project of creating a new website for this job. This study outlines some of the final steps in bringing the project to market. To begin, we must define the case study's stakeholders, their roles, and their interests... This report will go over and demonstrate how to implement a software development life cycle as well as the appropriateness of software behavior design techniques when working on the TuneSource Project.

Task 1 – Analysis (1)

I. Identify the stakeholders, theirs roles and interests in the case study.

1. Requirement

- Definition of requirement:

In software engineering, a requirement is a feature of new software that someone wants, needs, or commands. It describes the software's capabilities and any limitations it should have.

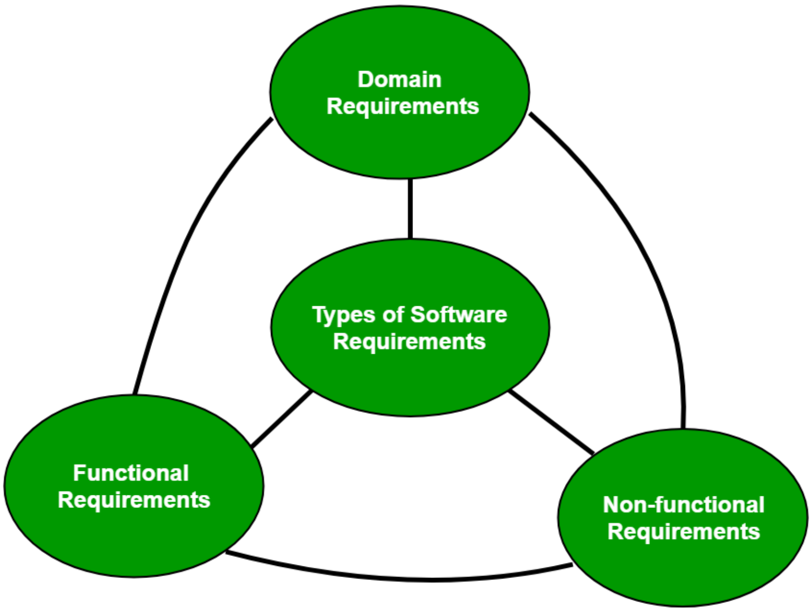


Figure 1: Requirement (Source: Internet)

- Requirement will be created that describe:

+ Business requirements

+ User requirement – What the user need to do

+ Functional requirements – What the software should do

+ Non-functional requirements – Characteristics the system should have

+ System requirements – How the system should be built

2. Stakeholder and their interest

a. Stakeholder

- A stakeholder is any person, group, or organization whose interests are affected by the success or failure of a project or business venture. Stakeholders can be inside or outside the company funding the project, and they all have a vested stake in the project's success. Stakeholders are significant because their choices can positively or negatively affect the project. There are additional crucial or important stakeholders, whose backing is necessary for the project to proceed. *(CFI Team, 2022)*

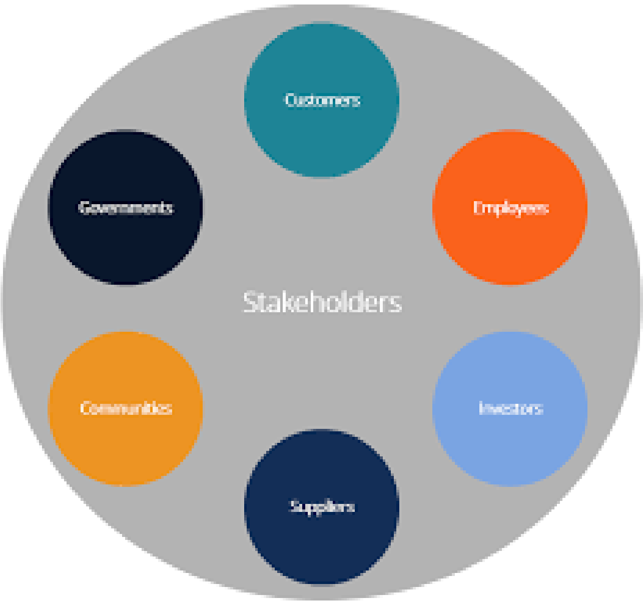


Figure 2: Stakeholder (Source: Internet)

No	Stakeholder	Role	Interest
	Bui Huong Linh	Project Manager	The project manager is involved in the planning, execution, supervision, control,...
2	User	Customer	Make purchases on the website, the downloads, users increase, revenue increases to help the company grow more. Give feedback for website development.
3	The IT department at Tune Source	Website administration,	Receive a salary from the company

		upkeep, and monitoring.	
4	John Margolis, Megan Taylor, and Phil Cooper	Co-Founders of Tune Source	They will have the right to submit ideas, develop project, control spending, …
5	Carly Edwards, Assistant Vice President, Marketing	Project Sponsor	Selection of the project manager and team, development of success criteria, and successful completion of the project.

Table 1: Stakeholder and their interest

3. Discuss the relationships between the FRs and NFRs.

   a. Functional requirement

   - Definition of functional requirement:

     In software engineering, a functional requirement identifies a system or one of its parts. It lists the duties that the program is required to carry out. Three components make up a function: inputs, behavior, and outputs. What a system is likely to do can be determined by calculation, data manipulation, business process, human interaction, or any other function. You can better understand the system's intended behavior by capturing functional software requirements. A system's required functions, services, or activities can be used to express this behavior. *(Martin, 2022)*

   - For an IT solution this would focus on:

     + Data Storage

     + Data Retrieval

     + Data Processing

   b. Non – Functional requirement

   - Definition of non – functional requirements:

     An non-functional requirement identifies a software system's quality trait. They stand for a set of criteria used to assess a system's particular performance. For instance, how quickly does the website load?

     To guarantee the overall software system's usability and efficacy, a non-functional requirement is necessary. Systems that don't meet non-functional requirements may not be able to meet user needs.

   - Constraints on the features offered by a solution such as:

+ Performance

+ Safety

+ Security

+ Dependability

+ Costs

+ Timescales

c. Relationship

The basis for a software development project's success is laid by the FRs and NFRs requirements. Specific attributes were stated in functional criteria to encourage software development. Software features and non-functional requirements are closely related. Non-functional requirements specify how a project should be carried out to create an end-user experience from the user's point of view. The software development process is greatly accelerated when the necessary product features are identified and the requirements for each of those features are created. If the functional and nonfunctional needs are not stated, the project will suffer tremendously. NFRs represent the overall experience of a product or application, such as security or performance, whereas FRs focus on a single function.Functional requirements (FRs) can be collected via user stories, use cases, and functional scores. NFRs are typically discovered in a product's user experience and user interface (UI / UX), in particular. Both NFR and FR need to coexist on the same project.

No	Requirement	Type of requirement	Explain
1	Customers must have an account to log in, because I want their details to be easily controlled.	FR	Because this function relates to the business process.So, this is a functional requirement to have in the company system as it would be great if the executive could know the information so that it can be easily controlled.
2	We need to have the look and feel of the app similar to the colour schemes of the theatre's interior to preserve the branding.	NFR	This is a requirement for user experience. If there is a good interface, it will be beneficial to maintain the brand. However, the bad interface also affects the business.
3	The mobile application can trigger notifications when new songs come	FR	As this function involves user interaction. So this is a functional

	out.		requirement to have in the system as it would be great if users could get a notification when new songs are coming out. Moreover, it also helps to increase the interaction between the user and the application.
4	There should be a feedback system where people can make suggestions for improvements to the facilities.	FR	Because this is a function related to user interaction. So this is a functional requirement to have in the system as it would be great if the customer could give his suggestions to improve the facility. Moreover, it also helps to increase the interaction between the customer and the application. However, there will be different opinions when customers make suggestions.
5	The site must be safe and secure.	NFR	This is a security requirement. It is also related to system reliability. If the website is protected, users will feel more secure when using it. On the contrary, the website is not safe and the security is low, which will lead to the website being easily hacked.
6	The app will allow people to download songs for free.	FR	So this is a functional requirement that should be in the system as it would be good for the customer that the app allows people to download selected free songs. Moreover, it also helps to increase the interaction between the user and the application.
7	The app should be error free. Nothing is more infuriating than an error message.	NFR	This claim is security. So this is a viable functional requirement in the system as it can keep the application error free. However, some cases still have errors that cannot be avoided.

Table 2: FR and NFR

II. Discuss the technique are using to obtain the requirements

1. JAD method

- Definition of JAD:

Joint Application Development (JAD) is a method of speeding up the development of information technology solutions. JAD employs customer involvement and group dynamics to accurately depict the user's perspective of the business need and to collaborate on the development of a solution. Prior to the advent of JAD, requirements were identified by conducting individual interviews with stakeholders. The JAD approach was developed in response to the ineffectiveness of this interviewing technique, which focused on individual input rather than group consensus. *(Mrsic, 2022)*

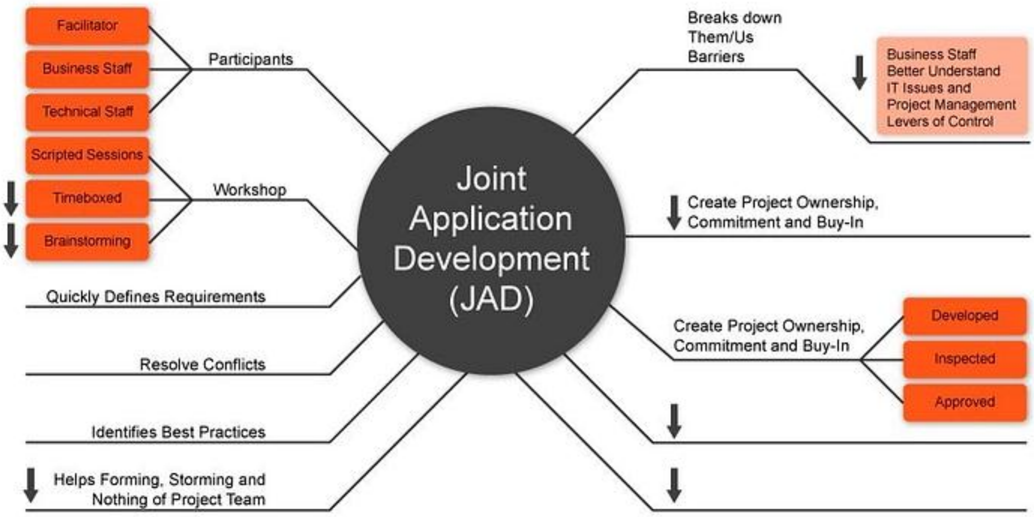

Facilitator
Business Staff
Technical Staff
Participants
Scripted Sessions
Timeboxed
Brainstorming
Workshop
Quickly Defines Requirements
Resolve Conflicts
Identifies Best Practices
Helps Forming, Storming and Nothing of Project Team
Joint Application Development (JAD)
Breaks down Them/Us Barriers
Business Staff Better Understand IT Issues and Project Management Levers of Control
Create Project Ownership, Commitment and Buy-In
Create Project Ownership, Commitment and Buy-In
Developed
Inspected
Approved

Figure 3: JAD (Source: Internet)

- Advantages:

  + JAD enables you to create better, error-free software and troubleshoot issues more easily.

  + JAD encourages the group to work more quickly and deliver on schedule.

  + The close communication makes progress more rapid

  + All risks are reduced by the business and its clients working together.

  + JAD decreases project development expenses and time requirements.

  + Clear requirements enhance system quality.

- Disadvantages:

  + Depending on the project's scale, JAD can necessitate a sizable time commitment.

  + It's challenging for the team to keep focus and align goals due to divergent viewpoints.

2. User observation method

- Definition of observation method:

  Observation methods are generally used when it is important to avoid the types of errors that can occur in interview methods or 'bias' as a result of evaluation and interpretation processes on the part of the workers, or when no workers are yet available for the planned jobs in future workplace design. *(Bhasin, 2022)*

- Advantages:
  - The observer will get a practical insight into the work.
  - Improvement areas can be easily identified.
- Disadvantages:
  - The working methods of the participants may change during the observation, and the observer may not obtain a clear image.
  - Knowledge-based activities cannot be observed.
  - Participants might get disturbed.

3. Interview method

- Definition of interview method:

  The most common technique of acquiring requirements is through an interview. The interview process consists of five basic steps: choosing interview candidates, creating interview questions, getting ready, performing the interview, and post-interview follow-up. Using this strategy, the interviewer questions stakeholders to gather information. The most common method used in interviews is face-to-face communication. The interviewer poses a series of predefined questions during a structured interview. An unstructured interview is one in which the interviewer does not use a predetermined format or ask specific questions. *(questionpro, 2022)*

- Advantages:
  - Speech correction is simple: In an interview, any misunderstanding or mistake can be easily corrected. Because both the interviewer and the interviewee are physically present in front of the interview board.
  - Timesaving: An interview can help you save time when choosing the best candidate. The interview allows for communication in a very short period of time.
  - Versatile: One of the primary benefits of interviews is their portability. It can be framed differently depending on the situation.

- Relationship development: An interview can help to develop the relationship between the interviewer and the interviewee. It improves the parties' mutual understanding and cooperation.
- Gathering primary information: Interviews can assist in gathering fresh, new, and primary information as needed.
- Enough information: Enough information can be gathered through the interview process. Because the interviewer is free to ask the interviewee any question.

  ………

- Disadvantages:
  - Time-consuming: One of the major constraints of the interview process is time. Preparation for the interview, conducting the interview, and interpreting the responses all took a long time, making the interview method time-consuming.
  - Incomplete process: A suitable candidate cannot be chosen solely through an interview. The written test takes precedence over the interview.
  - No record: In the case of the interview, some confusion may arise in the future because there is no evidence of what was discussed at the interview.
  - Unsuitable for personal matters: The interview method may not reveal personal information.
  - Interviewer inefficiency: An interview is a systematic data collection process. The effectiveness of the interviewer determines the success of the interview. An interviewer's inefficiency can lead to misleading results.
  - Expensive: In general, the interview method is costly.

  ....

Task 2 – Analysis (2)

1. Use case

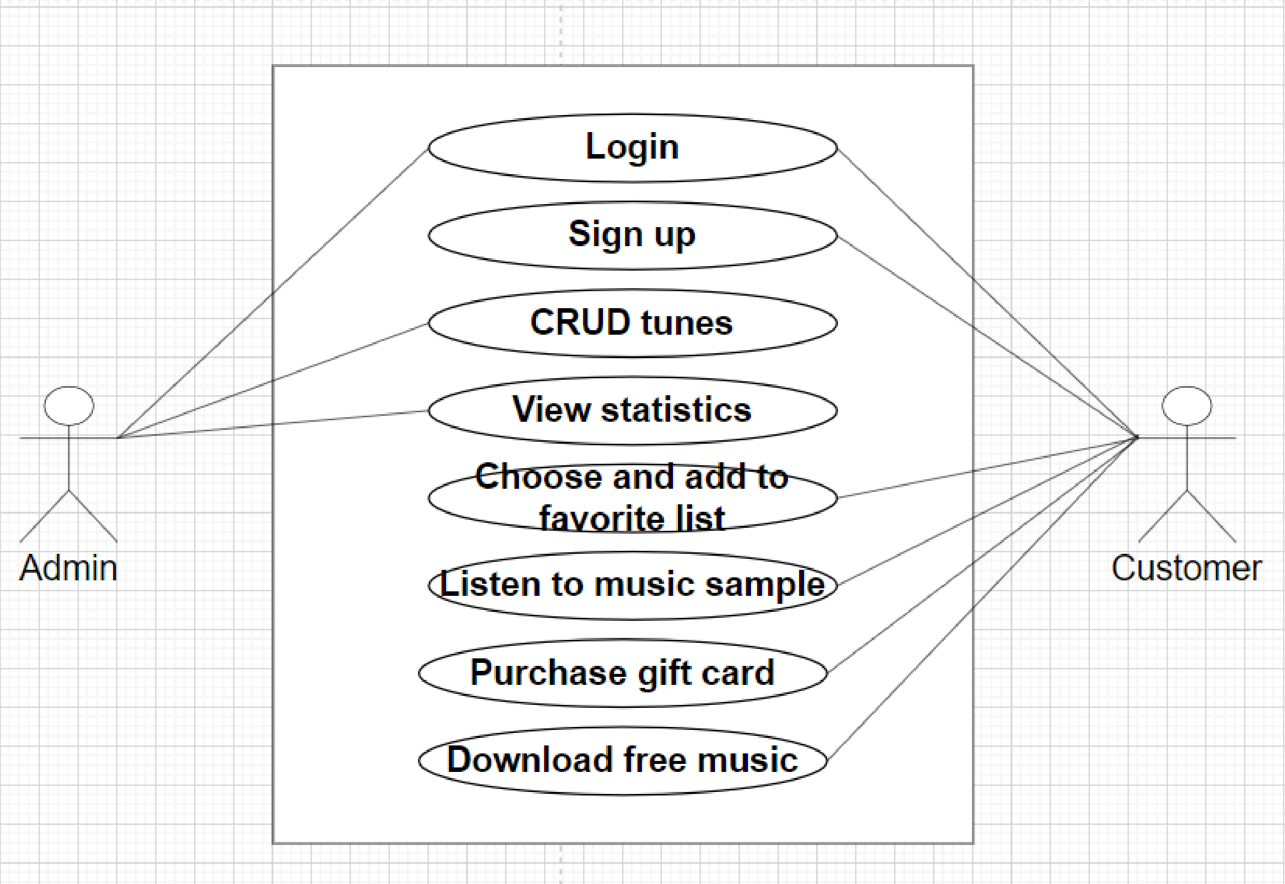


Figure 4: Use case

- Admin can login and view, add, edit, and delete tracks, as well as view statistical data.
- Customers can login, listen to music samples, choose and add to favorites list, ...

2. Use case description for two case

Use case name: Listen to music sample abd buy tunes	ID: UC-1
Actor: Tune shopper	
Description: This use case describes a customer who is listening to music.	
Trigger: Tune shopper visits website to listen to music. Type: External	
Preconditions: The tune database is available online. The Tune website is now live.	
Nomal Course: 1. System displays default home page or customer page. 2. The website will display a request to invite you to access the system (login) 3. After successful login, the user wants to search for songs. 4. After searching for the song you want to download. User need to double click on selected song then user can see product information and price. 5. There are 2 forms of payment: payment by credit card and payment after delivery. 6. After the user has successfully logged in, on every sale, ... the system will automatically send a notification to the user.	Information for Steps ← Username/Password ← Tunes matching search ← Show detailed song information. → New Interest ← New shopping cart enter ← System activation
Postconditions:	
Exceptions: Payment error(Step 5) - Information is incorrect or does not exist. - Please enter correct information. Account is not valid (Step 2) - The system notifies the customer that the password or usernamme is incorrect - Ask user to re-enter password or username.	

Table 3: Listen to music sample abd buy tunes

Use case name: Search for songs through the website	ID: UC-1
Actor: Customer	
Description: Visit the website to search for the best, most popular songs....	
Trigger: Tune shopper visits search for song to web Type: External	
Preconditions: The customer has searched for the best songs.	
Nomal Courese: 1. Customers downloading songs must browse through the links on the page or enter their account username and password.	Information for Steps ← Username/Password

2. The customer who wants to create an account: Performs Registration. 3. Customer enters search request. 4. The system displays songs according to the customer's search requirements. 5. The customer chooses a song and wants to hear a sample 6. The customer selects a favourite song to add to favourites. 7. The customer selects a song to remove from the favourites list.	← Search criteria ← Songs matching search → Songs samples → New favourites → Modifies favourites
Postconditions: Favourites list can be modified	
Exceptions: Account is not valid (step 1) - The system notifies the customer that the password or usernamme is incorrect - Ask user to re-enter password or username.	

Table 4: Search for songs through the website

3. Context diagram

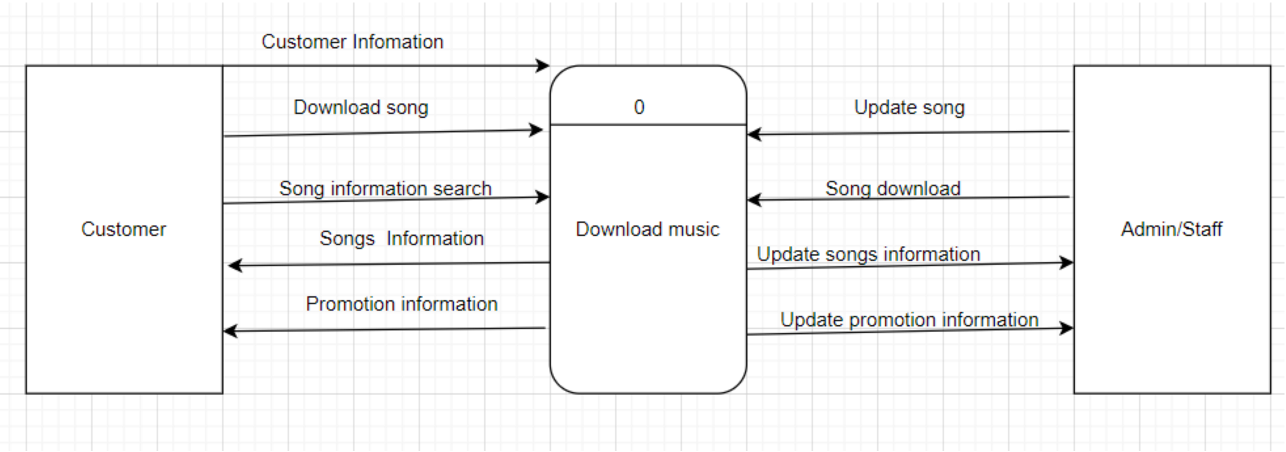


Figure 5: Context diagram

4. Data flow diagram level 1

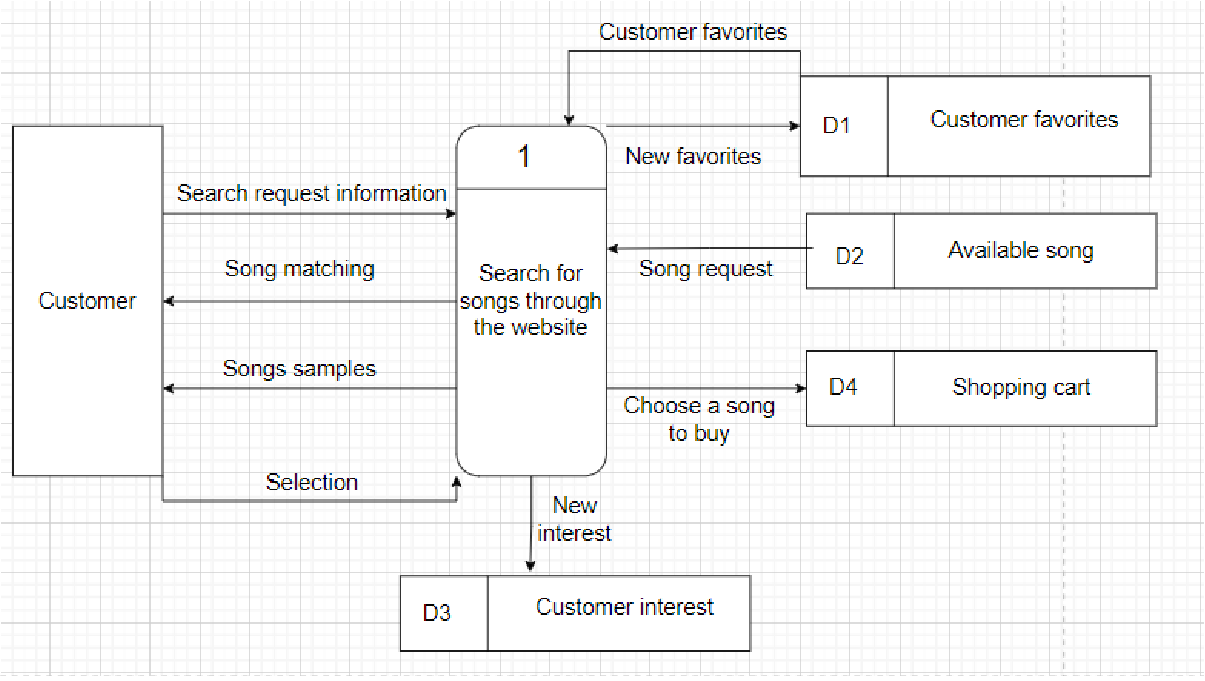


Figure 6: DFD 1

5. ERD

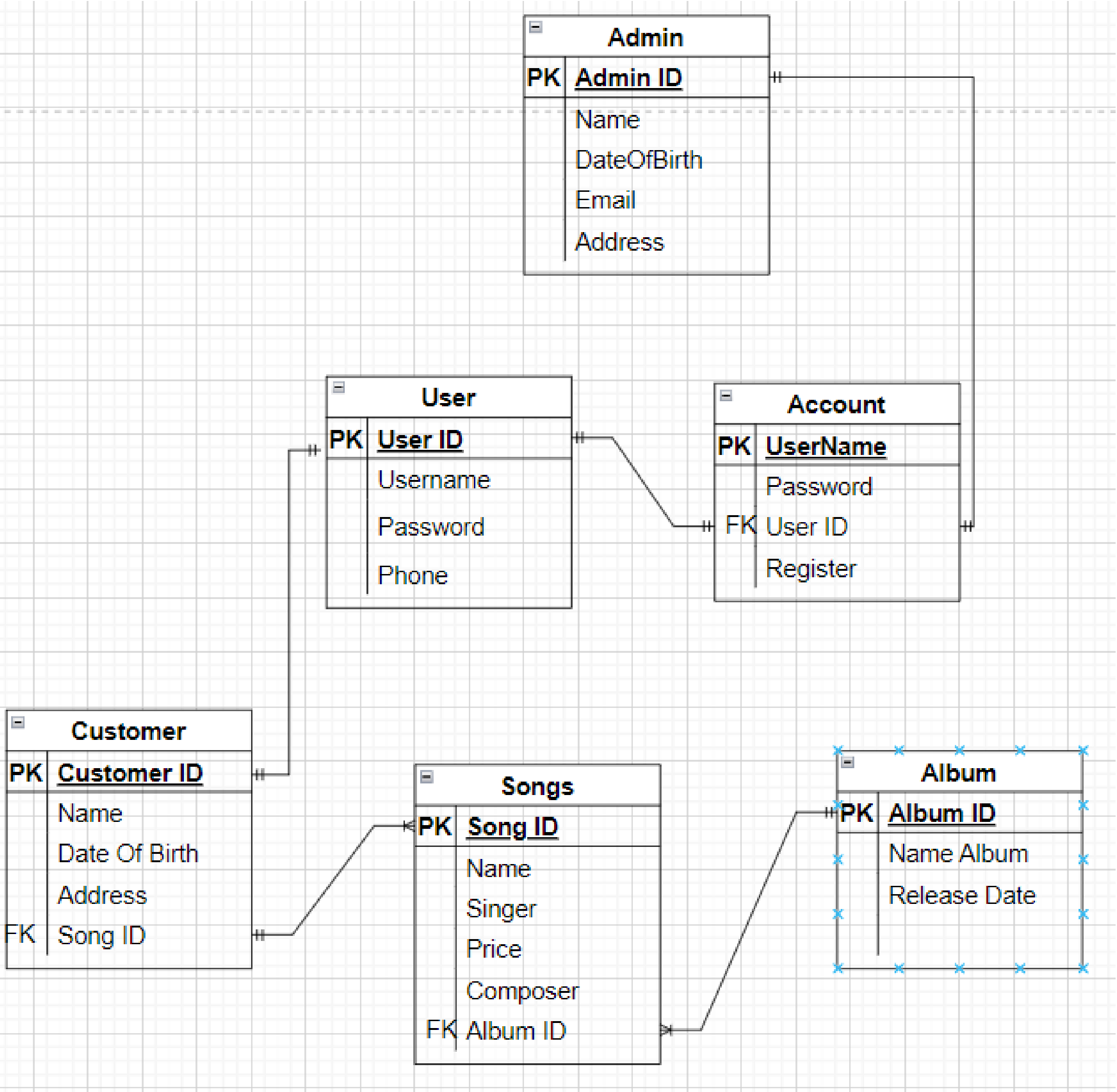


Figure 7: ERD

Task 3 – Analysis (3)

I. Discuss how the user and software requirements are addressed in the design phase.

1. What is the design phase in SDLC ?

a. Definition of design phase in SDLC:

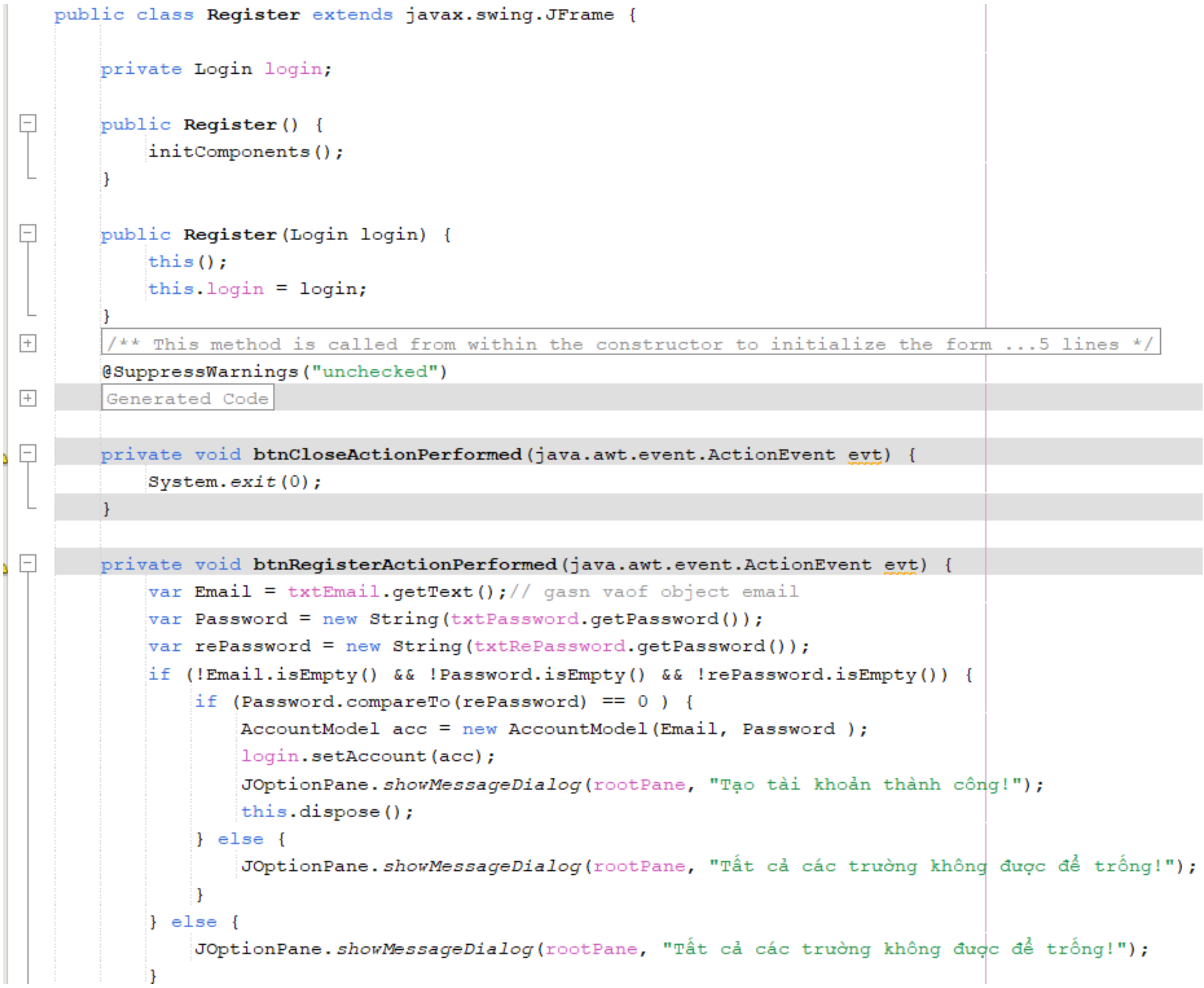
The design phase of SDLC is a stage in which software developers define the technical details of the product. These details may include screen designs, databases, sketches, system interfaces, and prototypes, depending on the project.

These details are used by clients to make final product design decisions. All details are compiled in a Software Requirement Specification (SRS) document for convenience. Before any development can begin, the SRS has requirements, standards, and expectations for the future product that the client must approve. (Halwai, 2022)

b. Roles & Responsibilities

- Quality assurance (QA) specialist: Involved in requirements analysis and product testing to ensure that there are no performance issues.
- Developer of software: Assists clients in translating their requirements into product features and writing original code for the product
- Analyst for business: Creates product specifications for UX designers and reviews their work.
- Designer of UX: Creates user - centred product interface designs to ensure that the user's interaction with the product is logical.
- Client: Provides business - related information, reviews team materials, and approves project milestones.
- Manager of a project: Leads the project and makes decisions to ensure its success (project organisation, task allocation, budget planning, progress monitoring, and others).

2. Source code

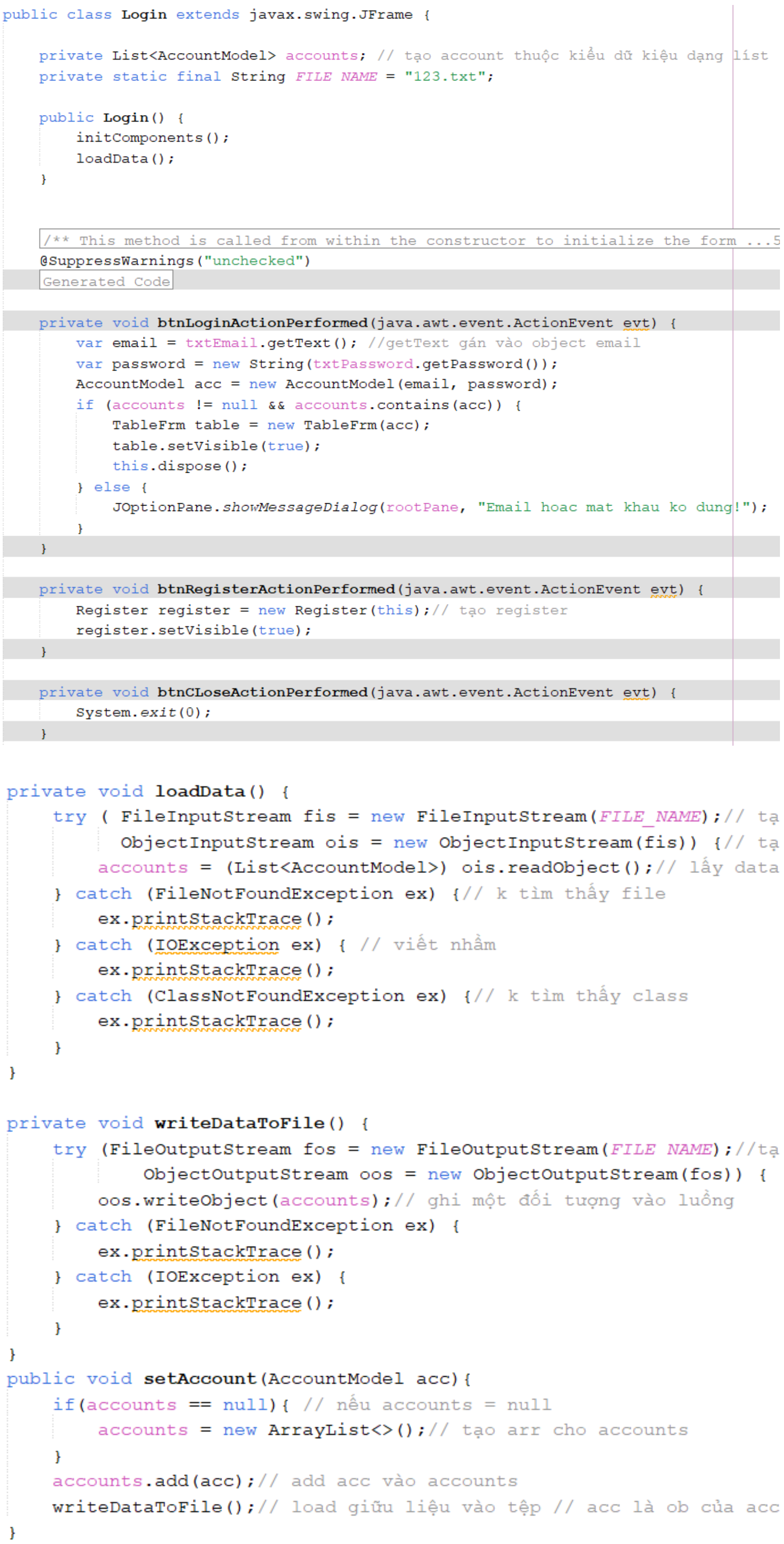


```
public class Register extends javax.swing.JFrame {

    private Login login;

    public Register() {
        initComponents();
    }

    public Register(Login login) {
        this();
        this.login = login;
    }
    /** This method is called from within the constructor to initialize the form ...5 lines */
    @SuppressWarnings("unchecked")
    Generated Code

    private void btnCloseActionPerformed(java.awt.event.ActionEvent evt) {
        System.exit(0);
    }

    private void btnRegisterActionPerformed(java.awt.event.ActionEvent evt) {
        var Email = txtEmail.getText();// gasn vaof object email
        var Password = new String(txtPassword.getPassword());
        var rePassword = new String(txtRePassword.getPassword());
        if (!Email.isEmpty() && !Password.isEmpty() && !rePassword.isEmpty()) {
            if (Password.compareTo(rePassword) == 0 ) {
                AccountModel acc = new AccountModel(Email, Password );
                login.setAccount(acc);
                JOptionPane.showMessageDialog(rootPane, "Tạo tài khoản thành công!");
                this.dispose();
            } else {
                JOptionPane.showMessageDialog(rootPane, "Tất cả các trường không được để trống!");
            }
        } else {
            JOptionPane.showMessageDialog(rootPane, "Tất cả các trường không được để trống!");
        }
```

Figure 8: Register

Get data from users; if entered correctly, it will display the message "Account creation successful," but if incorrectly entered or left blank, the error message "All cases cannot be blank" will be shown. The information is then put into a list using setAccount.

Exit system



```
public class Login extends javax.swing.JFrame {

    private List<AccountModel> accounts; // tạo account thuộc kiểu dữ kiệu dạng líst
    private static final String FILE NAME = "123.txt";

    public Login() {
        initComponents();
        loadData();
    }


    /** This method is called from within the constructor to initialize the form ...5
    @SuppressWarnings("unchecked")
    Generated Code

    private void btnLoginActionPerformed(java.awt.event.ActionEvent evt) {
        var email = txtEmail.getText(); //getText gán vào object email
        var password = new String(txtPassword.getPassword());
        AccountModel acc = new AccountModel(email, password);
        if (accounts != null && accounts.contains(acc)) {
            TableFrm table = new TableFrm(acc);
            table.setVisible(true);
            this.dispose();
        } else {
            JOptionPane.showMessageDialog(rootPane, "Email hoac mat khau ko dung!");
        }
    }

    private void btnRegisterActionPerformed(java.awt.event.ActionEvent evt) {
        Register register = new Register(this);// tạo register
        register.setVisible(true);
    }

    private void btnCLoseActionPerformed(java.awt.event.ActionEvent evt) {
        System.exit(0);
    }
```

```
private void loadData() {
    try ( FileInputStream fis = new FileInputStream(FILE_NAME);// tạ
          ObjectInputStream ois = new ObjectInputStream(fis)) {// tạ
        accounts = (List<AccountModel>) ois.readObject();// lấy data
    } catch (FileNotFoundException ex) {// k tìm thấy file
        ex.printStackTrace();
    } catch (IOException ex) { // viết nhầm
        ex.printStackTrace();
    } catch (ClassNotFoundException ex) {// k tìm thấy class
        ex.printStackTrace();
    }
}

private void writeDataToFile() {
    try (FileOutputStream fos = new FileOutputStream(FILE NAME);//tạ
            ObjectOutputStream oos = new ObjectOutputStream(fos)) {
        oos.writeObject(accounts);// ghi một đối tượng vào luồng
    } catch (FileNotFoundException ex) {
        ex.printStackTrace();
    } catch (IOException ex) {
        ex.printStackTrace();
    }
}
public void setAccount(AccountModel acc){
    if(accounts == null){ // nếu accounts = null
        accounts = new ArrayList<>();// tạo arr cho accounts
    }
    accounts.add(acc);// add acc vào accounts
    writeDataToFile();// load giữu liệu vào tệp // acc là ob của acc
}
```

Figure 9: Login

Create a list containing the ‘AccountModel’ assigned to ‘accounts’. Create a file-name = "123.txt' of type string.

Get the value from the UI variable and assign it to an intermediate variable. Create an 'AccountModel' containing the email and password assigned to an intermediate variable. If 'account ! = null' and 'account.contains(acc)' will bring up the table 'TableFrm'. Otherwise, an error 'Email or password is incorrect' will be displayed.

Get the value from the UI variable and assign it to an intermediate variable. Create an 'AccountModel' containing the email and password assigned to an intermediate variable. If 'account ! = null' and 'account.contains(acc)' will bring up the table 'TableFrm'. Otherwise, an error 'Email or password is incorrect' will be displayed.

Get the value from the UI variable and assign it to an intermediate variable. Create an 'AccountModel' containing the email and password assigned to an intermediate variable. If 'account ! = null' and 'account.contains(acc)' will bring up the table 'TableFrm'. Otherwise, an error 'Email or password is incorrect' will be displayed.

Get the value from the UI variable and assign it to an intermediate variable. Create an 'AccountModel' containing the email and password assigned to an intermediate variable. If 'account ! = null' and 'account.contains(acc)' will bring up the table 'TableFrm'. Otherwise, an error 'Email or password is incorrect' will be displayed.

If 'accounts == null' create an 'ArrayList' assigned to 'accounts'. Then add the object to 'accounts'. Finally load the data into the file

Exit system

3. Test case

Test	Test Description	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fall
1	Register	1/ Enter 2/ Login	Email: 1 Password: * Re-Password: *	Successfully registered an account.	Successfull	Pass
2	Login	1/ Register 2/ Login	Email: 1 Password: *	Successfully logged into the account.	Successfull	Pass

Table 5: Test case

4. Wireframe

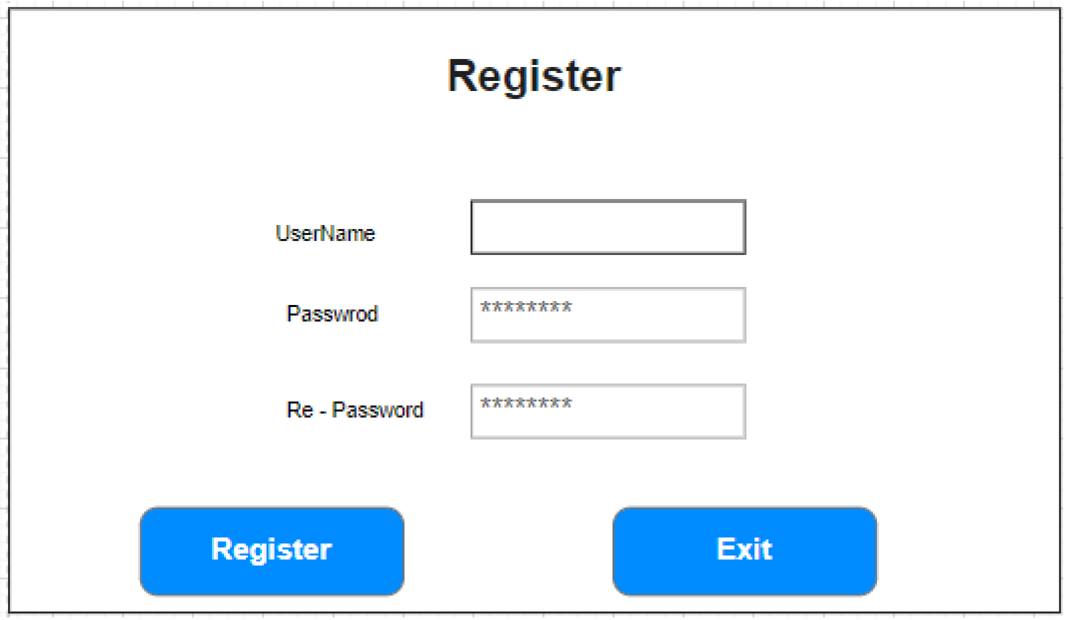


Figure 10: Wireframe Register

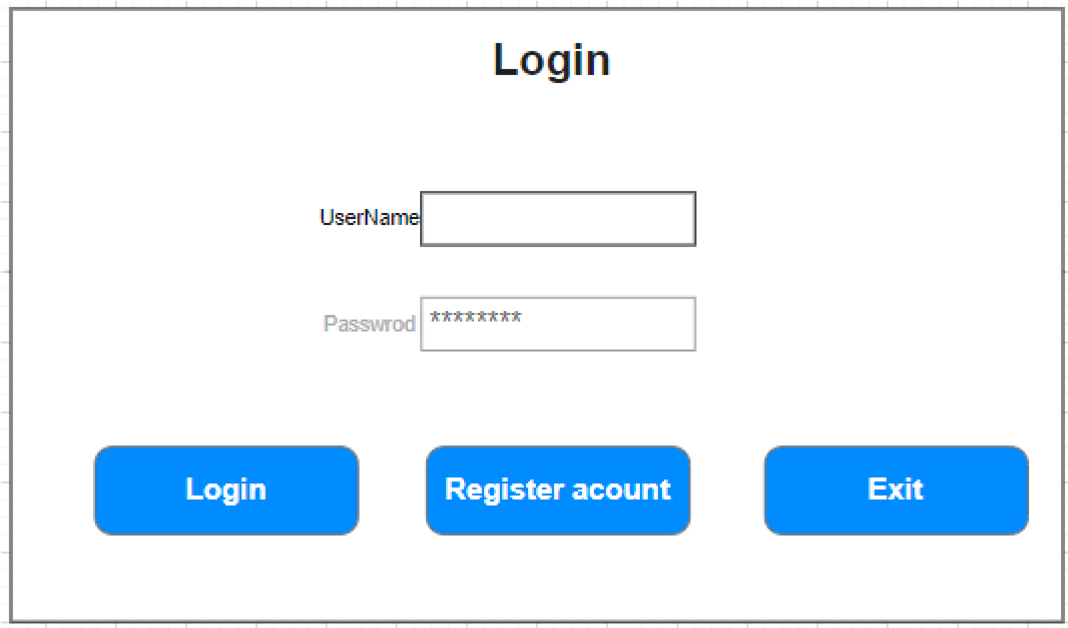


Figure 11: Wireframe Login

Conclusion

The assignment descriptive learning outcomes, with a focus on performing software investigations to meet business needs and using appropriate software analysis tools and techniques to conduct software investigations and create supporting documentation. Moreover, we understand more about DFD, requirement, appropriateness of software behaviour design techniques discussed by describing how to meet user and software requirements.

Bibliography

Bhasin, H. (2022). Retrieved from Observation Methods – Definition, Types, Examples, Advantages: https://www.marketing91.com/observation-method/

CFI Team. (2022, 05 07). Retrieved from Stakeholder: https://corporatefinanceinstitute.com/resources/knowledge/finance/stakeholder/?fbclid=IwAR29Pkf30WP1EN_ynBgJwZ8ITp6G11EHg-1cd42DhuDnPDMJrVWwy7CK2Qo

Halwai, S. (2022). Retrieved from SDLC Design Phase – Everything You Need to Know: https://www.openxcell.com/blog/design-phase-in-sdlc/

Martin, M. (2022, 06 25). Retrieved from Functional Vs. Non Functional Requirements: Differences: https://www.guru99.com/functional-vs-non-functional-requirements.html

Mrsic, M. (2022). Retrieved from Joint Application Development (JAD): https://activecollab.com/blog/project-management/joint-application-development

questionpro. (2022). Retrieved from Types of Interviews in Research and Methods: https://www.questionpro.com/blog/types-of-interviews/

Index of comments

2.1 Correct identity of stakhoder interest and requirement

Functional and non functional requirement are provided

Some discussion of techniques have been provided and choose Interview for Tune source

Some diagrams are provided. DFD1 is not correct, Pls refer to google classom to re-do DFD1.

Some wireframe, coding and testing have been addressed based on requirement.