**TON DUC THANG UNIVERSITY**

**FALCUTY OF INFORMATION TECHNOLOGY**

**SOFTWARE ENGIRNEERING**

**REQUIRERMENT SPECIFICATION**

**Project: Friend list**

**Code: P11SRS**

**Record of amendment**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Date** | **Index** | **Reason** | **Modify**  **(Add, edit, delete)** | **Describe** | **Version** |
| **23/12-9:00 AM** | 1 |  | Add , edit | Add non-functional requirement and edit page , text | V1.1 |
| **25/12-4:00 PM** | 2 |  | Add | UML sequence diagram , UML state diagram, UML active diagram | V1.2 |
| **25/12 – 19:30 PM** | 3 |  | Add | Use case table  Non-function requirement | V1.3 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# **Sign**

Project manager: Dang Phuoc Date:

<Administrator>

Tester: ………………………… Date:

<Quality manager>

**INDEX**

[Sign 3](#_Toc529602229)

[1. Purpose 5](#_Toc529602230)

[2. Area 5](#_Toc529602231)

[3. Reference 5](#_Toc529602232)

[4. Layout 5](#_Toc529602233)

[5. Acronym 5](#_Toc529602234)

Non-functional requirement………………………..…………...…....5

[1.Security 5](#_Toc529602230)

[2.Maintainability…………………………………………………..….5](#_Toc529602230)

3.Platform……..…………………………………………………..….5

[Describe system 6](#_Toc529602235)

[1. Target of system 6](#_Toc529602236)

[2. Model of system 6](#_Toc529602237)

[Require 7](#_Toc529602238)

[1. UI require 7](#_Toc529602239)

[2. Other require 7](#_Toc529602240)

[List of function 8](#_Toc529602241)

[1. Count how many friends you have 8](#_Toc529602242)

[2. Sort the friends from a to z 8](#_Toc529602243)

[3. Sort the friends to male and female 9](#_Toc529602244)

[4. Search for recently added friends 10](#_Toc529602245)

[5. Check out friends having a birthday in the month 10](#_Toc529602246)

[6. Suggest a friend through the user added to the list 11](#_Toc529602247)

[7. Connect to social networking sites 12](#_Toc529602248)

[8. Search for relevant information 12](#_Toc529602249)

[Use case diagram 20](#_Toc529602250)

Functional [diagram 21](#_Toc529602251)

UML sequence [diagram 22](#_Toc529602251)

UML state [diagram 23](#_Toc529602251)

UML active [diagram 24](#_Toc529602251)

**Introduce**

## **Purpose**

This program to show a friend list of a person which will have information such as birthday , gender,… Also It will notice you about the birthday of your friends so you won’t forget it,…

## Area

This app will use on our class friends which will use to know each other information.

## **Reference**

<https://www.e-reading.club/bookreader.php/141481/Oppel_-_SQL_-_a_beginner%27s_guide.pdf>

<https://www.tutorialspoint.com/sql/pdf/sql-quick-guide.pdf>

<https://codewithintent.com/creating-a-new-friend-friend-listhow-to-in-java/>

## **Layout**

4.1 Introduce

4.2 Describe System

4.3 Common Requirement

4.4 Diagram

4.5 Some Function

## **Acronym**

UI:USER INTERFACE

**Non-Functional Requirement**

1. **Security**

Users will only be able to know their own personal information and other people's information if they provide information to their account. . Users will be protected identity and all related information

1. **Maintainability**

Correct defects or their cause.

Repair or replace faulty or worn-out components without having to replace still working parts.

Maximize efficiency, reliability, and safety,

meet new requirements.

1. **Platform**

Ensure environmental compatibility

# **Describe system**

1. **Target of system**

This system will help user remember about birthday and notices other. And it will annulment for user prior to that day. It will help users do not have to worry about forgetting important dates anymore. It will also help users store their information. When it should be able to review it easily.

## 2. **Model of system**

This software is suitable for all users. People who have busy jobs and can not remember all the things to do and information friends. The objects in the list are all the people that the user wants to save information. The software can connect to the alarm bell for reminders. It's like a miniature notebook.

# **Require**

1. **UI require**

+ Display information of each people: full name, birth day.

+ Can add/update new people into the list.

+ Connect database to the UI.

1. **Other require**

+ Can notification when someone’s birthday is coming before 3 day.

+ Notification when someone’s birthday is coming.

+ Have database to storage people’s info.

1. **Details**
   1. Audit-ability

* Authenticated user
* Time
* IP Address of client
  1. Correctness
* The exact value in the date and time alert to notify the users
  1. Interoperability
* Smartphone (Android, IOS, …)
* Tablet (Android, IOS,… )
* PC/ Laptop
* Smartwatch
  1. Maintainability
* Update through Play store or Appstore, or direct link in website
* If it has a trouble, contact with support to handle
  1. Performance
     1. Capacity
* Open app at most 10s
  + 1. Response Time
* Response time at most 5s
  1. Safety
* Keep your information in private

# **List of function**

1. **Count how many friends you have**
   1. Describe

the system will count all the friends that you have in the list so you can know how many friend that you have in the list.

* 1. Stages
     1. Access users database
     2. Count every friend database
     3. Show result
  2. Request
     1. User Database so that the system will begin to count

1. **Sort the friends from a to z**
   1. Describe

If you prefer to 'declare' to show your friend from the name, List will automatically search by all users name in your database, grouping all along with information. shared, content, contacted so you can easily find the one you want to see.

* 1. Stages
     1. access users database
     2. check user first name
     3. sort it form letter a to z
     4. show result
  2. Request
     1. User Database
     2. The information of name of each friend to sort.
     3. Button to show result

1. **Sort the friends to male and female** 
   1. Describe

the list will show all of your friends in the list that have gender like male or female and it will show it to you.

* 1. Stages
     1. access database
     2. check database gender tag
     3. divide it to male tag and female tag
     4. show result
  2. Request
     1. User Database
     2. The information of each friend
     3. The information of tag gender in each friends information.

Button to show result.

1. **Search for recently added friends**
   1. Describe

This function allows customer find new friends just accept or add to list

* 1. Stages
     1. Access database
     2. Query last date
     3. Show the result
  2. Request
     1. User Database
     2. The information of each friend
     3. Button to show result

1. **Check out friends having a birthday in the month**
   1. Describe

This function allows checking out friends having a birthday in the month

* 1. Stages
     1. Access database
     2. Get data come back
     3. Show it on screen
  2. Request
     1. User Database
     2. The information of date of birth of friends
     3. Button to show result

1. **Suggest a friend through the user added to the list**
   1. Describe

This function allows customer find new friends through the connection

* 1. Stages
     1. User database, social network
     2. Get information about common friends
     3. Ask customer to add friends
  2. Request
     1. User Database
     2. The information of friends
     3. Button to show result

1. **Connect to social networking sites**
   1. Describe

This function allows customer connect to social networks

* 1. Stages
     1. Link to social networking site
     2. Access database
  2. Request
     1. Internet
     2. Database of websites

1. **Search for relevant information**
   1. Describe

This function allows users to search for relevant information such as date of birth, name, age, etc… when unknown

* 1. Stages
     1. Access database
     2. Query
  2. Request
     1. Database
     2. SQL

1. . **Log in**
   1. Describe

This function allows customer manage their database

* 1. Stages
     1. Access user and password
     2. Check
  2. Request
     1. Database
     2. SQL

1. **Registration**
   1. Describe

This function allows customer sign up to use program

* 1. Stages
     1. Create new account
     2. Check
  2. Request
     1. Database
     2. SQL

## **Use case diagram**

**System**

**User**

**Friend**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **ID** | **Name** | **Brief Description** |
| **1** | UC1 | Sign in | User login to get database of themselves |
| **2** | UC2 | Count | Check the quantity |
| **3** | UC3 | Search | Check and find with keywords |
| **4** | UC4 | Sort | Use function to arrange database |
| **5** | UC5 | Connect to website | After login, at most database will update to cloud though connect website (same as firebase) |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **ID** | **Name** | **Permission** |
| **1** | UC1 | Sign in | Admin, user |
| **2** | UC2 | Count | Admin, user |
| **3** | UC3 | Search | Admin, user |
| **4** | UC4 | Sort | Admin, user |
| **5** | UC5 | Connect to website | Admin, user |

**Functional diagram**

**System recommend**

**Friend List**

**Account**

**User**

Count friend

Registration

Log in

Sort friend Male or Female

Search friend

Check out friend have birthday in month.

Suggest a friend.

Search base on information

Connect to social network site

Sort friend A-Z

**UML sequence diagram**

Radiation()

returntoMain()

putValuetoSpinner()

btnClickAndIntent()

onActivityResult()

Friend

System

User

**UML state diagram**

Save

No permission

Input Info

Return List

Add friend

Permission

Data transmission

**UML active diagram**

User

System

Add friend into list

Input info

Yes

No

Return main

Add friend