Counselling App on Android for UTHM Students with AES

Lee Kuan Yi[1], Cik Feresa Mohd Foozy[1]*

[1]Fakulti Sains Komputer dan Teknologi Maklumat,
Universiti Tun Hussein Onn Malaysia, Batu Pahat, 86400, Johor Malaysia

Abstract: The counselling app on Android for Universiti Tun Hussein Onn Malaysia (UTHM) students with Advanced Encryption Standard (AES) is designed to provide the counselling service to the students who are facing depression and anxieties. The students can send messages to the university counsellor if they are feeling low and unmotivated. The university counsellor can then reply to the student by giving some advice and motivation. All messages will be encrypted before they are stored in the database. The encryption algorithm used is the Advanced Encryption Standard with 256-bit of key size (AES-256). There are six modules in this application which include Login, Admin, Forgot Password, Message, Bulletin Board and DASS Test. The methodology used for the development of this project is the object-oriented software development model. The testing is done by 10 students and 4 counsellors in UTHM. The app has passed the functional testing and security testing.

Keywords: Advanced Encryption Standard (AES), Counselling App

1.   Introduction

A counselling app is a type of Mental Health–related smartphone app (MHapp). It is created to provide the counselling service for the users who suffer mental health problems such as depression or anxiety. The use of counselling apps or MHapps can enhance a counsellor's relationship with the clients, and improve the lives of the clients [1].

The COVID-19 pandemic causes a lack of emotional support and broader societal effects among students. This increases the anxiety levels among them. Universities should prioritise the development of digital psychological interventions for students, such as apps and online programs other than services such as text messages, forums, and phone calls [2]. Confidentiality is significant to create a trust in the relationship between counsellors and their students [3]. The students will be willing to believe their counsellors only if the counsellors can keep their personal details and information secret. The major concern of the app is the privacy and confidentiality of the personal information after it is collected [4]. The sensitive data and information of the user that is stored in the database must be protected in a secure way.

This project involves building an app for the students who have mental issues to seek help from the university counsellors. The target users for this app are the university students and the university counsellors in Universiti Tun Hussein Onn Malaysia. The expected time taken to complete the project is about 40 weeks. The goal for this project is to provide the counselling service through the smartphone app for the students who suffer anxieties and depressions but cannot have face-to-face meetings with the university counsellors. The tool will be created using the Android Studio software and the programming language used is Java. The Firebase database is used to store the data. The encryption algorithm used is Advanced Encryption Standard with 256-bit of key size (AES-256). There are six modules in this application which are Login, Admin, Forgot Password, Message, Bulletin Board and DASS Test.

This project allows the students who suffer anxiety and depression to seek help from the university counsellor on their smartphones. By using the app, the students can consult their counsellor about the issues and problems they are facing in their studies. The app will protect the students' personal data and information. This is because all data and information will be encrypted before they are stored in the database. Other than that, the app can be used at anytime, anywhere by a student even if the student is not on the university campus. The student does not have to make an appointment to have a face-to-face meeting with the university counsellor because the counsellor can give the advice and guidance to the student directly in the app.

The remaining parts of this paper are organised as follows. Section 2 is about the related work regarding the concepts and algorithms used in the project. Section 3 explains about the software development model used in the development of the application. Section 4 discusses the results and discussion of this project.

2. Related Work

This section explains about the concepts and algorithms that are used in the project. Some similar applications are also studied and compared with the proposed application in the project.

2.1 Encryption Algorithms

This section describes the study of encryption algorithms. The first algorithm is Data Encryption Standard (DES). It is one of the oldest and most widely used encryption algorithms. It is made up of an algorithm and a key. The key is a sequence of eight bytes and each byte contains eight bits for a 64-bit key. However, the key is actually 56 bits in length because each byte contains one parity bit [5]. In DES, a 64-bit block of plaintext is enciphered into a 64-bit block of ciphertext under the control of a 56-bit key. It is implemented by using several key-controlled substitution boxes (S-boxes). Each S-box has a 6-bit input and a 4-bit output. Its cryptographic strength increases by repeating the substitution and permutation process several times. The encryption process of DES consists of 16 iterations, which are called as rounds. It uses a cipher function and a 48-bit key at each round. The cipher function is made up of the substitution and permutation. The 48-bit key is a subset of the bits in the externally supplied key and it is not the same in each round [6].

The next encryption algorithm is Advanced Encryption Standard (AES). It is a type of block cipher. It has a block length of 128 bits. Three different key lengths are allowed in AES which are 128, 192 or 256 bits. AES encryption with 128-bit keys has 10 rounds of processing, AES encryption with 192-bit keys has 12 rounds of processing and AES with 256-bit keys has 14 rounds of processing. All the rounds of processing are the same except for the last round in each case. Each round of processing consists of one single-byte based substitution step, a row-wise permutation step, a column-wise mixing step, and the addition of the round key. These four steps are performed in a different order for encryption and decryption [7].

The size of the data block in DES is 64 bits whereas the size of data block in AES is 128 bits. The key length in DES is only 64 bits. However, there are three different key lengths in AES, that is, 128 bits, 192 bits or 256 bits. There are 16 rounds of processing in DES but there are 10, 12 and 14 rounds of processing in AES-128, AES-196 and AES-256 respectively. The algorithm of AES is more secure than DES because it has a larger key length than DES. Besides that, the speed of processing in AES is faster than DES. Thus, AES is clearly the better encryption algorithm compared to DES.

2.3   Studies on Equivalent Apps

This section describes the study of equivalent apps. The first app is PlusVibes. It is a mobile application which was launched in October 2020. It combines all of an individual's mental health needs on one platform. The goal of this app is to help people who require a little bit more motivation whenever they are feeling low in their lives. The app is free to use [8].

The second app is ThoughtfullChat. It is a mobile app which was started by Joan Low [8]. The founder wanted to make psychotherapy (or talk therapy) more affordable and easily available for those who cannot clear their minds. On ThoughtfullChat, the user is able to chat with a certified mental health professional when they are in need. The app also features other personal development tools and resources, as well as checking the current stress levels, happiness levels and quality of sleep of the user.

The third app is The Help Talk. It is a popular online counselling platform that was launched by Alvin Tan, a clinical psychologist and two other co-founders in 2017. The founders found out that many people could not afford private counselling services in Malaysia. Therefore, they made this app to help the users to get therapy from licensed counsellors and clinical psychologists conveniently on their smartphones. The users are helped by the therapists to go through their problems such as depression, anxiety, stress, family conflicts and trauma [8].

The next app is TalkCampus. It is an app that can be accessed by all students in University of Technology Sydney (UTS) for free [9]. It is an online wellbeing and mental health support network. It provides peer support instantly and around the clock to the users who suffer mental health problems. The user can connect with other university students around the world. The user can share the feeling in a safe and supportive environment anonymously without any worries. Other than that, the user can also join the community in the app and support others who need help in their lives.

The last app is My Student Support Program. It offers individual brief counselling for all currently enrolled students in DePaul University [10]. MySSP can be used by the students to receive confidential and private support anytime, anywhere. All students in the university can call or chat with a Student Support Counselling in real-time or schedule a telephone or secure video session. The service is available for 24 hours and every day.

Table 1 shows the comparison between the existing apps, which are PlusVibes, ThoughtfullChat, The Help Talk, Talk Campus and My Student Support Program (MySSP) with the proposed app (Counselling App on Android for UTHM Students with AES).

Table 1: Comparison of Existing Apps with the Proposed App

Feature	Plus Vibes	Thought fullChat	The Help Talk	Talk Campus	My SSP	The Proposed app
Login or sign in module	Yes	Yes	Yes	Yes	Yes	Yes
Register or admin module	Yes	Yes	Yes	Yes	Yes	Yes

Table 1: (cont)

| Feature	Plus Vibes	Thought fullChat	The Help Talk	Talk Campus	My SSP	The Proposed app
Forgot password module	No	Yes	Yes	Yes	Yes	Yes
Messaging between counsellor and student	Yes	Yes	Yes	No	Yes	Yes
Bulletin board module	No	No	No	No	No	Yes
Encryption for message	Yes (AES-256)	Yes (AES-256)	Yes (AES-256)	Not mentioned	Not mentioned	Yes (AES-256)
Free to use	Yes	No	No	Yes	Yes	Yes

All the apps have the login or sign in module and register or sign up module. ThoughtfullChat app, The Help Talk app, Talk Campus app, MySSP app and the proposed app have the forgot password module but PlusVibes app does not have this module because it uses login code on email for user authentication. The chat function between the counsellor and student is available on all the apps except the Talk Campus app. Only the proposed app has the bulletin board module, the other five apps do not have this module. The encryption on the message is available only on the proposed app. The PlusVibes app, Talk Campus app, My SSP app and the proposed app are free to use but the ThoughtfullChat app and The Help Talk app require monthly subscription fees from the user.

3. Methodology

This section describes all the necessary information about the methodology that is applied to obtain the results of the project.

3.1 Object-oriented Software Development

The methodology used in the project is the object-oriented software development model. In an object-oriented software model, more time is spent on the design phase as compared to the time spent on the analysis phase, testing phase and implementation phase. This model is applied so that the application can be designed for maintenance, reuse and modification after the initial development. A lot of effort is required in the coding process for reuse. However, these efforts are worth it because it will reduce the time used for the implementation phase. A lot of time is used to study the problems more deeply. In the end, the implementation phase of the project becomes quicker due to a lot of the problems that have been known [14]. Figure 1 shows the phases of project development in an object-oriented software development life-cycle.
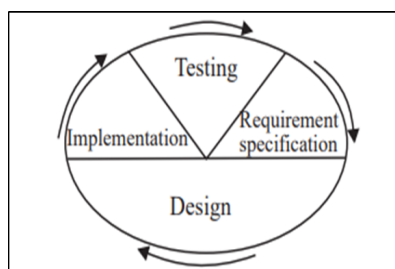
Figure 1: The object-oriented software development life cycle [11]

3.2   Requirement Specification Phase

Requirement specification is the first phase of the project. In this phase, the planning of the project will be done first. The problems and issues that need to be solved in the project are confirmed. Then, the objectives of the project are determined and the scopes of the project are finalised. The activities and the tasks that will be done throughout the duration of the project are arranged and planned in a Gantt chart. Literature review will also be done in this phase. Journals, articles and books that are related to the project will be searched. The concepts and algorithms that can be used in the project will be studied. The most suitable concepts and algorithms that can fulfil the needs of the project will be selected in this phase.

In this phase, the process of requirement gathering will also be carried out. The functional requirements and the non-functional requirements of the counselling app will be determined. The functional requirements include the functionalities and features of each module inside the app while the non-functional requirements include the performance, usability, operational and security requirements of the proposed app.

3.3   Design Phase

After the process of requirement gathering is done, the design phase will begin. The counselling app will be designed using Unified Modelling Language (UML). UML is a modelling language used in the object-oriented approach of software development. The use-case diagram, sequence diagram and activity diagram are the types of UML diagrams designed and used in this project. These diagrams are designed so that they can be used in the implementation phase of the project. After that, the database used by the counselling app will be designed using the entity relationship diagram (ERD) and class dictionary. The test plan and user interface design will also be made in the design phase of the project.

Use case diagrams are designed to show the actions and activities that can be performed by different types of users. There are three use case diagrams designed for the counselling app. One is the use case diagram for students which is shown in Figure 2. Another two are the use case diagram for counsellors shown in Figure 3 and the use case diagram for admin shown in Figure 4.
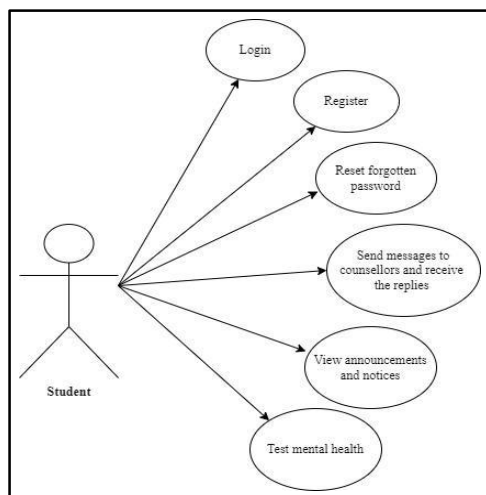
Figure 2: Use Case Diagram for Students
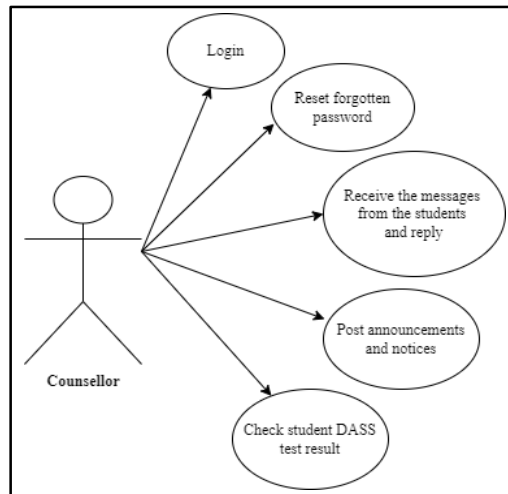
**Figure 3: Use Case Diagram for Counsellors**
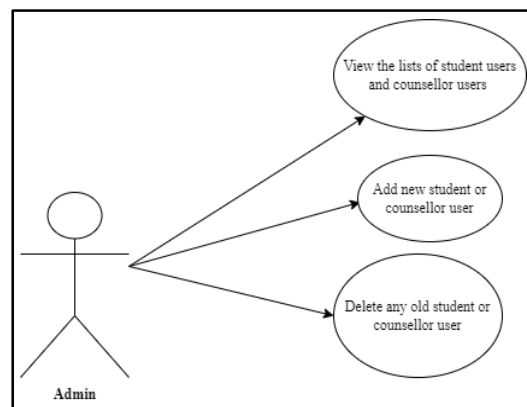
Figure 4: Use Case Diagram for Admin

The activity diagram of the counselling app is shown in Appendix A. For the proposed application, the users will first enter the "Login" module. If the users use the counselling app for the first time, they will enter the "Register" module to create a new account. After that, if the users have forgotten the password, they can reset the password in the "Forgot Password" module. Then, if the users do not forget the password, the system will verify the user using the username and password entered by the users. If the verification of the users is not successful, the users will be brought to the login menu again. However, the users will be directed to the main menu if the verification is successful. The users are authorised as either students or counsellors. The student users can send a message to counsellors and then receive the message from the counsellors. They are also able to view the announcements and notices in the app. Next, the student user can take the DASS Test and get the test result. For counsellor users, he or she can receive the message from students and send messages back to the students. Other than that, the counsellors can post any announcements and notices in the app. The counsellors can also check student DASS test results. For admin, he can view the user list in the system and add or delete the user accounts in the system.

The class diagram is used to illustrate the classes that are designed for the development of the proposed application. There are 8 classes used in the system of the application. The names of the classes are User, Login, Admin, Forgot Password, Message, Bulletin Board, DASS and Change Password. Each class has different attributes (variables) and methods (functions). Each attribute or method is set to either public or private. The class diagram of the proposed counselling app is shown in Appendix B.

The next design of the counselling app is the system design. The student or counsellor users can open the first user interface which is the "Login menu" in the app. In case the users have forgotten the password, they can reset the password in the "Forgot Password" module and the data will be saved to the database. If the password reset is successful, the users are directed to the "Login" menu. After that, the users can enter the username and password in the "Login" menu, the data will be read from the database to verify the users. If the user verification is successful, the users will be directed to the main menu. In the main menu, the users can choose any menu they want to open. The "Messaging", "Bulletin Board" and "DASS Test" modules can save new data to the database. Old data can also be read or deleted from the database in these three modules. Next, the users can click "Back" to go back to the main menu. Finally, if the users log out from the account, they will be redirected to the "Login" menu. As for the admin user, he can log in to the "Admin" menu to add new user data to the database. The admin can read or delete the user data from the database. The admin will go back to the "Login" menu if he logs out from the admin menu. The system design of the counselling app is shown in Figure 5.
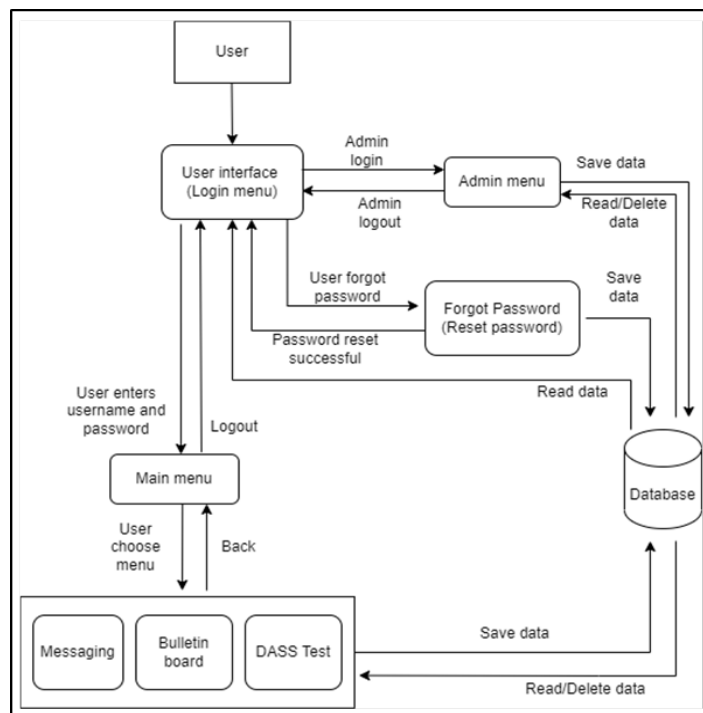
Figure 5: System design of the counselling app

There are five entities in the database of the counselling app. One student can send zero or many messages but one message can only be sent by one and only one student. One counsellor can send zero or many messages but one message can only be sent by one and only one counsellor. One counsellor can post zero or many bulletins but one bulletin can only be sent by one and only one counsellor. One student can take zero or many DASS tests but one DASS test can only be taken by one and only one student. The Entity Relationship Diagram of the counselling app is shown in Figure 6.
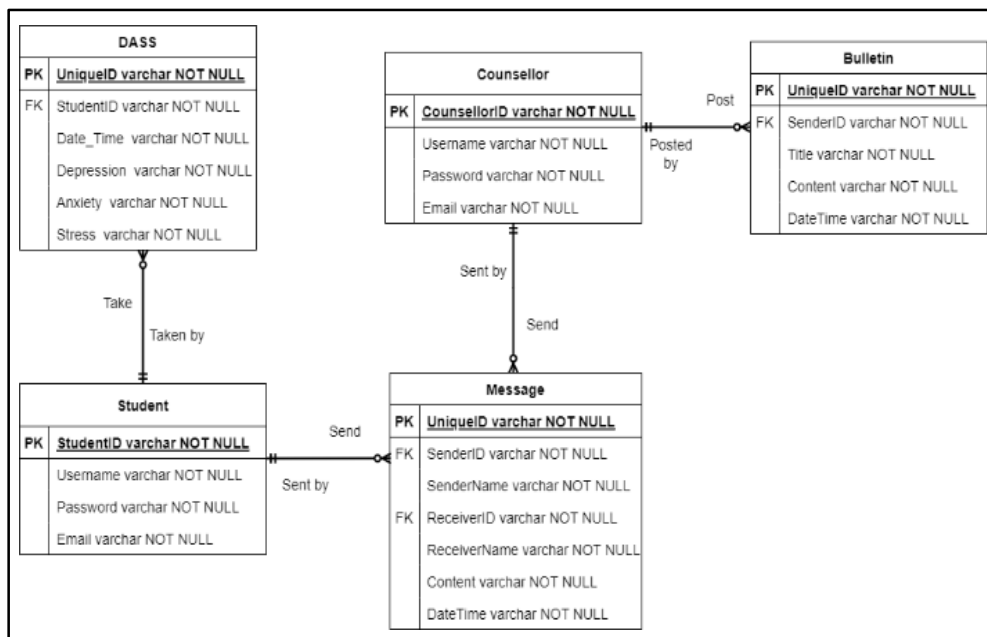
Figure 6: Entity Relationship Diagram (ERD) of the counselling app

3.4 Implementation Phase

The implementation phase will begin after the counselling app is designed. The requirement gathering and the design that have been made during the analysis and design phase ensure a smooth implementation process. The UML and database designs that are produced during the design phase will be used as a guide in the programming and development process of the counselling app. This means that the implementation phase of the app is based on the UML diagrams, the user interface design of the application and the structure of the database that are created in the previous phase.

The three security modules that are included in this application are Advanced Encryption Standard (AES), verification code and password complexity rule. Each of these security modules is intended to reduce the security risks of the application.

Advanced Encryption Standard (AES) is the encryption algorithm used to encrypt the data sent to the Firebase database. It uses the same key to encrypt and decrypt the data. The SHA-256 hash value for the date and time of the data will be used as the encryption key. The function setkey() is used to set the key for the encryption and decryption of the data. The function encrypt() is used to encrypt the string data and the function decrypt() is used to decrypt the string data. The source code to set the key for the AES algorithm is shown in Figure 7. The source code to encrypt the data using the AES algorithm is shown in Figure 8. The source code to decrypt the data using the AES algorithm is shown in Figure 9.

```java
public static void setKey(final String myKey) {
    MessageDigest sha = null;
    try {
        key = myKey.getBytes( charsetName: "UTF-8");
        sha = MessageDigest.getInstance("SHA-256");
        key = sha.digest(key);
        key = Arrays.copyOf(key,  newLength: 32);
        secretKey = new SecretKeySpec(key,  algorithm: "AES");
    } catch (NoSuchAlgorithmException | UnsupportedEncodingException e) {
        e.printStackTrace();
    }
}
```

Figure 7: Source code to set the key for the AES algorithm

```
@RequiresApi(api = Build.VERSION_CODES.O)
public static String encrypt(final String strToEncrypt, final String secret) {
    try {
        setKey(secret);
        Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");
        cipher.init(Cipher.ENCRYPT_MODE, secretKey);
        return Base64.getEncoder().encodeToString(cipher.doFinal(strToEncrypt.getBytes( charsetName: "UTF-8")));
    } catch (Exception e) {
        System.out.println("Error while encrypting: " + e.toString());
    }
    return null;
}
```

Figure 8: Source code to encrypt the data using the AES algorithm

```
@RequiresApi(api = Build.VERSION_CODES.O)
public static String decrypt(final String strToDecrypt, final String secret) {
    try {
        setKey(secret);
        Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5PADDING");
        cipher.init(Cipher.DECRYPT_MODE, secretKey);
        return new String(cipher.doFinal(Base64.getDecoder().decode(strToDecrypt)));
    } catch (Exception e) {
        System.out.println("Error while decrypting: " + e.toString());
    }
    return null;
}
```

Figure 9: Source code to decrypt the data using the AES algorithm

The verification code is used to verify the user when he or she wants to reset the account password in the Forgot Password module. After the user enters his or her email address, a verification code will be generated and sent to the user's email. The source code to generate a random verification code and send to the email is shown in Figure 10. The JavaMailAPI class is used to generate and send the email message.

```
Random r = new Random();
code = r.nextInt( bound: 899999)+100000;
String message = "Your verification code is "+code;
JavaMailAPI jma = new JavaMailAPI(stfmail, mSubject: "Verification code",message);
jma.execute();
```

Figure 10: The source code to generate a random verification code and send to the email

Password complexity rule is enforced to increase the strength of the account passwords. This will make the brute force attacks on the passwords more difficult. A strong password should be a combination of small letters, capital letters, numbers and special characters. Regular expressions (Regex) are used to test whether the password has the specified characters. Figure 11 shows the source code to check whether a password has at least one small letter. Figure 12 shows the source code to check whether a password has at least one capital letter. Figure 13 shows the source code to check whether a password has at least one number and Figure 14 shows the source code to check whether a password has at least one special character.

```
private Boolean checkSmallLetter(String str){
    if(str.matches( regex: ".*[a-z].*")){
        return true;
    }
    else{
        return false;
    }
}
```

Figure 11: The source code to check whether a password has at least one small letter

```
private Boolean checkCapitalLetter(String str){
    if(str.matches( regex: ".*[A-Z].*")){
        return true;
    }
    else{
        return false;
    }
}
```

Figure 12: The source code to check whether a password has at least one capital letter

```
private Boolean checkNumber(String str){
    if(str.matches( regex: ".*[0-9].*")){
        return true;
    }
    else{
        return false;
    }
}
```

Figure 13: The source code to check whether a password has at least one number

```
private Boolean checkSpecialChar(String str){

    if(str.matches( regex: ".*[^0-9a-zA-Z].*")){
        return true;
    }
    else{
        return false;
    }
}
```

Figure 14: The source code to check whether a password has at least one special character

The Login module is the main page that will be shown to the users when they open the application. The users must enter their user ID and password. Then, they can tap the "Login" button to login to their accounts. Figure 15 shows the interface of the Login module.
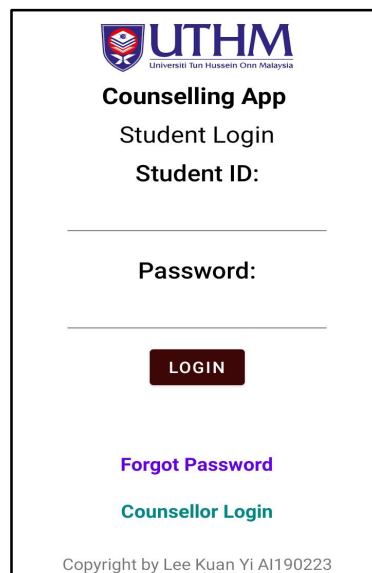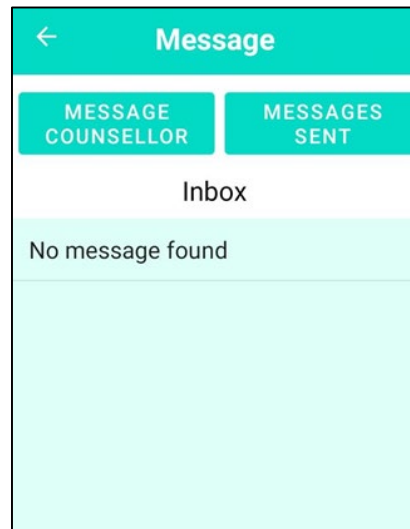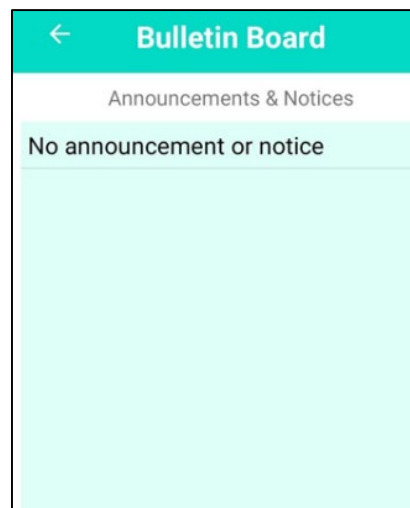

Figure 15: The interface of the Login module

The Message module is used by the users to send and receive messages. For example, a student user can send a message to a counsellor and the counsellor can receive the message in his or her inbox. Figure 16 shows the interface of the Message module.

Figure 16: The interface of the Message module

Figure 17: The interface of the Bulletin board module

The Bulletin Board module is used to post or view the announcements or notices. The counsellors can post the bulletins in this module and the students can view the bulletins. Figure 17 shows the interface of the Bulletin Board module.

The DASS Test module is used by the users to test their stress, anxiety, and depression level. The users need to answer all 21 questions. The score of each question depends on the answer given by the users. The scores will then be calculated depending on the type of mental health condition (stress, anxiety or depression) and the total score of the test will be used to determine their stress, anxiety and depression level. The scores will then be saved in the database. Figure 18 shows the interface of the DASS Test module.
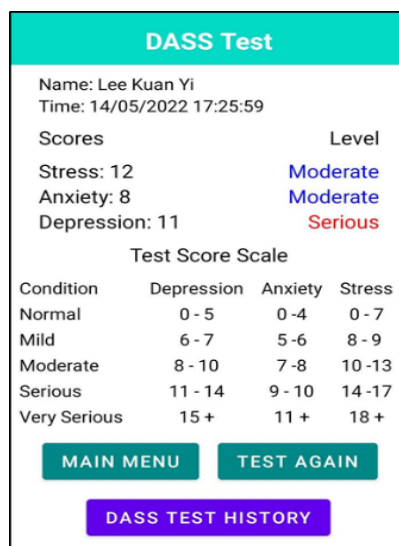
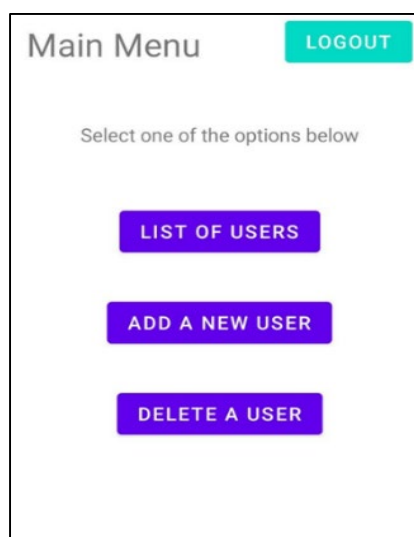**Figure 18: The interface of the DASS Test module**

Figure 19: The interface of the Admin module

The Admin module is used by the admin of the system. It can be used by the admin to view the lists of active student accounts and counsellor accounts in the system. It can also be used to add and delete the user accounts in the system. Figure 19 shows the interface of the Admin module.

3.5   Testing Phase

The last phase of the project is the testing phase. The functionality of the counselling app will be tested to make sure they are working within the expectation. The security requirements of the counselling app are also tested by the users. In this phase, bugs and errors that are present in the counselling app will be fixed immediately. The testing phase will end when all the project requirements are met. After the testing phase is completed, the project development will be concluded.

4.   Results and Discussions

This section describes the results of the testing that was conducted for Counselling App on Android for UTHM Students with AES. The User Acceptance Test is used to check whether the app has fulfilled all

the functional requirements and security requirements. Ten students from UTHM will test the student part of the counselling app whereas four counsellors from UTHM will test the counsellor part of the counselling app. The list of the UTHM students and UTHM counsellors who have tested the counselling app can be referred in Appendix C.

The results of the Functional Testing in the User Acceptance Test are summarized in Table 2. The results of the Functional Testing are based on the responses given by the users after they have tested the counselling app.

Table 2: Results of the Functional Testing in the User Acceptance Test

	Functional requirements	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
1	User can log in to his or her account in the system using the Login module.	0	0	0	1	13
2	User can reset the forgotten password in the Forgot Password module.	0	0	0	3	11
3	User can send and receive messages in the Messaging module	0	0	0	3	11
4	User can view the announcements and notices in the Bulletin Board module.	0	0	1	1	12
5	User can take the DASS test and get the results of the test in the DASS Test module.	0	0	3	2	9
6	User can change the account password in the Change Password module.	0	0	0	1	13
7	Admin can help the user to create or delete the account in the system.	0	0	1	4	9

The results of the Security Checklist in the User Acceptance Test are summarized in Table 3. The results of the Security Checklist are based on the responses given by the users after they have tested the counselling app.

Table 3: Results of the Security Checklist in the User Acceptance Test

	Security Requirements	Pass	Fail
1	An error message is shown if the username or the password field is left blank.	14	0
2	Ensure the error message does not indicate which part of the data authentication is incorrect. For example, an error message should not display "Wrong password" or "Wrong student ID".	14	0
3	Enforce the complexity of the password. For example, the password must combine numbers, small letters, capital letters and special characters.	14	0
4	Enforce the password length inside the policy. For example, the password must have minimum 8 characters and maximum 16 characters.	14	0
5	Passwords should be obscured in the textbox.	14	0
6	One time password (OTP) is sent to the email when the users want to change the forgotten password. An error message is shown if the users enter the wrong OTP.	14	0
7	Before the users change their password, they must enter the old password before they can set the new password. An error message will be shown if the old password is wrong.	14	0

Based on the results of the User Acceptance Test, it has proven that the counselling app has passed the functional testing because most of the users agree that the functional requirements of the app have been fulfilled. Other than that, the counselling app has also passed all the security testing because all the security requirements of the app are fulfilled.

5.   Conclusion

The project "Counselling App on Android for UTHM Students with AES" is designed to provide the counselling service to the students who are facing depression and anxieties. To protect the users' personal data and information, the data and messages sent by the users are encrypted before it is stored in the database. The encryption algorithm applied in this application is the Advanced Encryption Standard with 256-bit of key size (AES-256). The methodology used in the project is the object-oriented software development model. All the security modules and application modules in the app are successfully implemented. The app has passed both functional testing and security testing. The three objectives of the project are achieved.

The first advantage of the counselling app is the students can send the messages to the counsellors if they have any mental issues such as stress, anxiety and depression. The next advantage is the counsellors can post the important announcements and notices from the UTHM counselling center in the Bulletin Board module. The students can also use the app to take a DASS test to check their mental health status.

One of the disadvantages of the counselling app is it cannot be used without Internet connection. The second disadvantage of the app is it is only supported in the Android smartphone. The labels and texts in the app are only in English is also another disadvantage of the app.

In future work, an iOS version of the counselling app can be created so that the students who only have iOS smartphones can enjoy the functions and features inside the counselling app. Next, the labels and texts inside the counselling app should be expressed in English and Bahasa Melayu. Another improvement that can be made is to use the asymmetric encryption algorithm instead of the symmetric encryption algorithm.
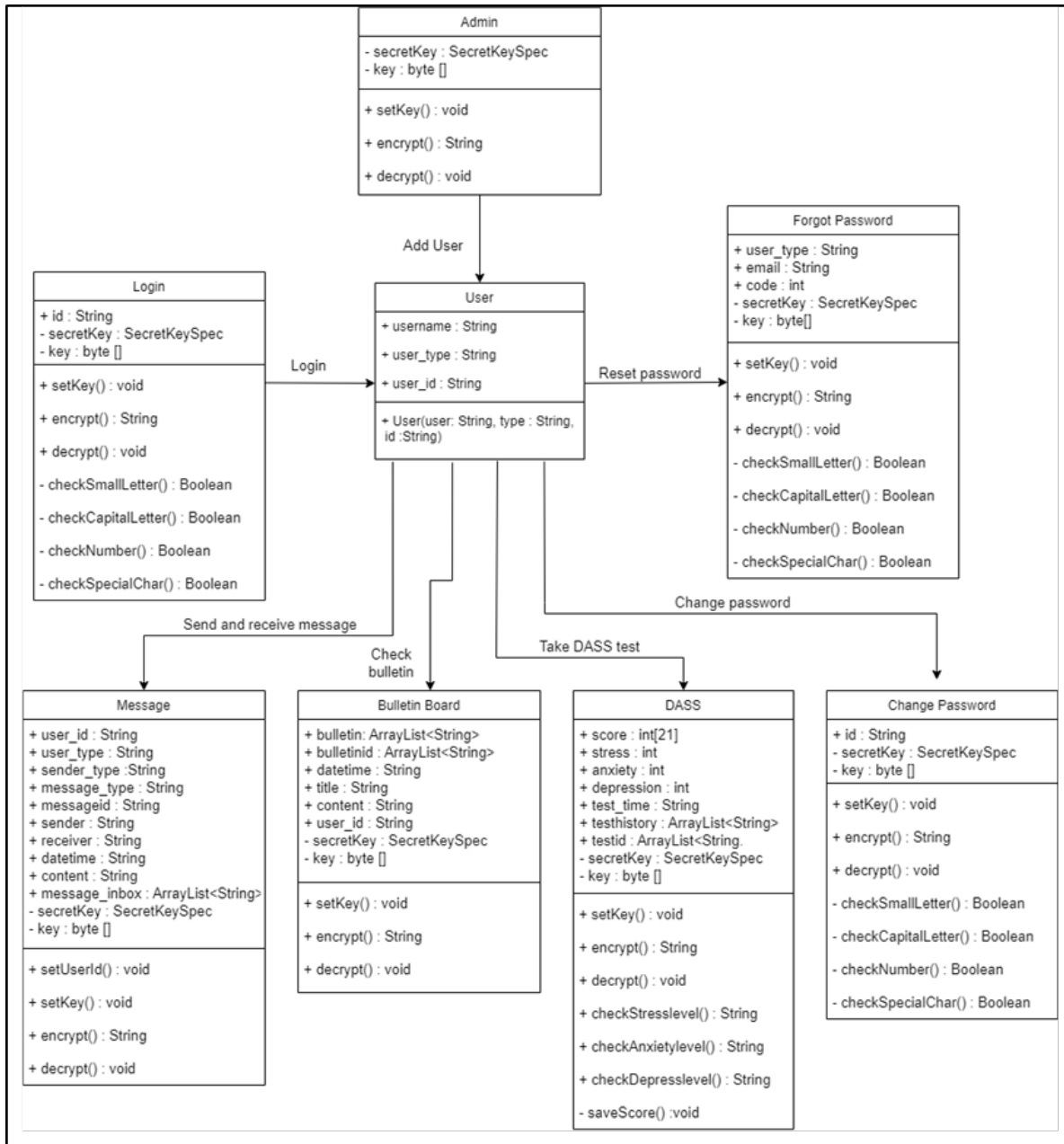
Acknowledgement
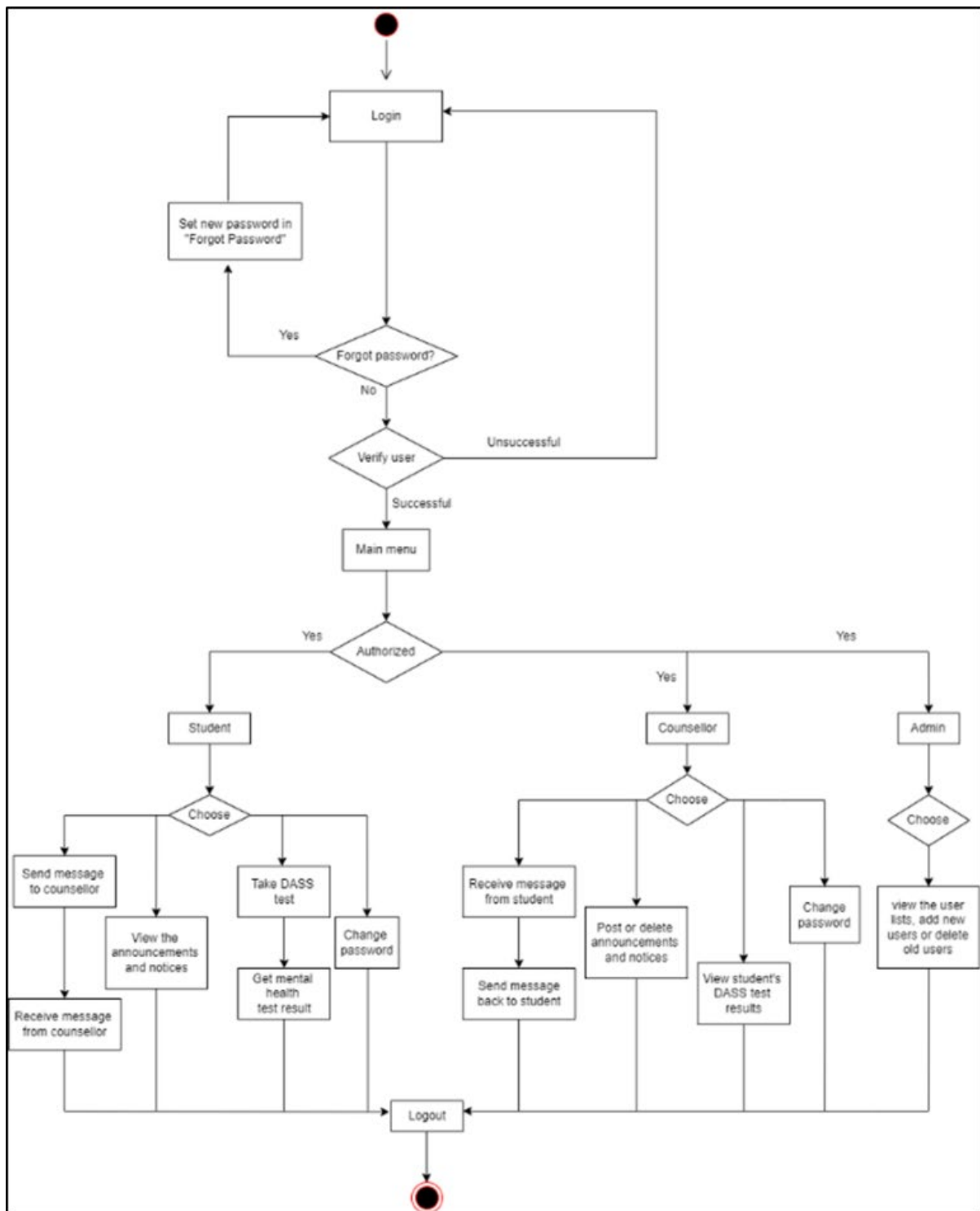
Appendix A

Activity diagram of the proposed application

Appendix B

Class diagram of the proposed application

Appendix C

The list of the users who have tested the counselling app

No	Name	Position
1	Dr. Aspalaila Binti Abdullah	UTHM counsellor
2	Dr. Noraishah Binti Daud	UTHM counsellor
3	En. Muhammad Khairullah Bin Mohd Shaari	UTHM counsellor
4	En. Syahrul Nizam Bin Lukman	UTHM counsellor
5	Aina Izzati Binti Afende	UTHM student
6	Lai Boon Shien	UTHM student
7	Leow Sheng Wei	UTHM student
8	Lew An Qi	UTHM student
9	Loo Jian Ming	UTHM student
10	Reenah A/P Manimaran	UTHM student
11	Tan Shi Pei	UTHM student
12	Teh Yu Fung	UTHM student
13	Tun Wen Kit	UTHM student
14	Yeong Mang Seng	UTHM student

References

[1]   K. M. Palmer and V. Burrows, "Ethical and Safety Concerns Regarding the Use of Mental Health-Related Apps in counselling: Considerations for Counselors," *J. Technol. Behav. Sci.*, pp. 1–14, Aug. 2020, doi: 10.1007/s41347-020-00160-9.

[2]   S. Sundarasen *et al.*, "Psychological Impact of COVID-19 and Lockdown among University Students in Malaysia: Implications and Policy Recommendations," *Int. J. Environ. Res. Public Health*, vol. 17, no. 17, p. 6206, Aug. 2020, doi: 10.3390/ijerph17176206.

[3]   M. Bokhari *et al.*, "Contribution of Awareness and Understanding in Legal and Ethics towards the Practice of Confidentiality amongst Counselors," *Asian Soc. Sci.*, vol. 10, Jul. 2014, doi: 10.5539/ass.v10n16p144.

[4]   K. M. Palmer and V. Burrows, "Ethical and Safety Concerns Regarding the Use of Mental Health–Related Apps in counselling: Considerations for Counselors," *J. Technol. Behav. Sci.*, vol. 6, no. 1, pp. 137–150, 2021, doi: 10.1007/s41347-020-00160-9.

[5]   S. R. Chauhan and S. Jangra, *Computer Security and Encryption : An Introduction*. Bloomfield, UNITED STATES: Mercury Learning & Information, 2020.

[6]   X. Wang, D. Coppersmith, S. M. Matyas, and C. H. Meyer, "Cryptography." McGraw-Hill Education, Nov. 09, 2020, doi: 10.1036/1097-8542.170600.

[7]   A. Abdullah, "Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data," 2017.

[8]   M. V. Ang, "3 Malaysian Mental Health Apps To Check Out If You Need Somebody To Talk To," *SAYS*. Dec. 18, 2021, [Online]. Available: https://says.com/my/tech/malaysian-mental-health-mobile-apps-that-offer-talk-therapy. [Accessed 18 November, 2021]

[9]   "TalkCampus app," *University of Technology Sydney*. Dec. 18, 2019, [Online]. Available: https://www.uts.edu.au/current-students/support/health-and-wellbeing/counselling-service-and-self-help/self-help-resources/talkcampus-app. [Accessed 25 November, 2021]

[10]   "My SSP: Student Support Program | Mental Well-being | Health & Wellness | Student Services

| Division of Student Affairs | DePaul University, Chicago." Nov. 18, 2021, [Online]. Available: https://offices.depaul.edu/student-affairs/support-services/health-wellness/mental-well-being/Pages/my-ssp.aspx. [Accessed 25 November, 2021]

[11]   R. Agarwal and P. Lago, "PATHOS—a Paradigmatic Approach to High-Level Object-Oriented Software Development," *SIGSOFT Softw. Eng. Notes*, vol. 20, no. 2, pp. 36–41, Apr. 1995, doi: 10.1145/224155.224156.