PATUAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

COURSE CODE CCE-122

SUBMITTED TO:

Prof. Dr. Md Samsuzzaman

Department of Computer and Communication Engineering
Faculty of Computer Science and Engineering

Sarna Majumder

Department of Computer and Communication Engineering
Faculty of Computer Science and Engineering

SUBMITTED BY:

Md. Sharafat Karim

ID: **2102024**,
Registration No: **10151**
Course code: **CCE-122**
Level–1, Semester–2
2nd Semester Final examination of B.Sc. Examination
Faculty of Computer Science and Engineering

Project Title : Safesync Contacts

Submission Date : 10 March, 2024

GitHub repository :
https://github.com/SharafatKarim/SafesyncContacts/

Description

"Safesync contacts" is an open source android isolated contact management application with integration of MySQL for local contacts and Firebase for a dynamic public database.

Motivation & Purpose

Contact sharing applications like "True Caller" has been violating user privacy for a long time and limiting its users behind a premium subscription plan. To solve this problem, we tried to create an open source project where we will give great importance to how data is stored and will be shared.

Objective

The core objective behind this project is to create a decentralized application with user privacy in mind to store and share contacts more securely. And to prevent other application to access our secure database.

Features

1. Isolated contact management application.
2. Easy search and sort with native android SDK.
3. MySQL database for local contacts.
4. Auto importing contacts from phone-book.
5. One click destructor.
6. Easy editing support.
7. Firebase database for public contacts.

Limitations

1. Manual sorting functionalities are currently not available. We will try to introduce it soon.

2. Some device may face errors due to the use of **deprecated API**. We will try to switch to **Kotlin** to fix those issues.

Target audience

1. Our target audience is from 3 years old kids to everyone. Minimal UI design pattern is used to ensure simplicity.

Languages

1. For logical unit java language is used with full OOP paradigm.

2. XML is used to handle designing part.

Development Tools

1. Android Studio

2. **Gradle** build system with **AndroidSDK**

Libraries

1. Android SDK for core development.

2. meow-bottom-navigation for drawing bottom navigation bar.

3. Circleimageviewer for displaying images in an aesthetic way.

4. Imagecropper for redirecting users to enable print support.

Development Methodology

Currently we are using agile methodology to fixing bugs and providing support through GitHub's issue section.

Testing

To ensure perfect build environment GitHub automation is integrated in our project. Besides we are trying to use bleeding edge technology as much as possible.

Timeline

Full timeline is available on our GitHub repository.

Milestones

1. **Version 0.1**
Introduced bottom activity design with four different fragment support.

2. **Version 0.2**
Home fragment: Home fragment contains two linked activity with adapter support to integrate mySQL database into the main program and to provide search functionality.

Second fragment: Second fragment contains one inflated layout with linkup to an activity to easily create a new entry or to import massively from database.
Third fragment: Third fragment contains the integratio with google firebase to cloud support and data sharing.
Info fragment: Info fragment is intended to showcase the versioning and GitHub URL.

3. **Version 0.3**
   Contact sharing ability with re-designed xml file and fragments implemention.

4. **Version 0.4**
   First stable release with lot of UX bug fix support and sharing contacts as plain text support.

Team members

1. Sharafat Karim

2. Mehedi Hasan Monir

3. Sadman Kabir

Hardware Requirements

1. Android minimum SDK 24

2. Android recommended SDK 34

3. Camera, gallery and storage permission **[OPTIONAL]**

Licensing

Our code is being distributed under **MIT license** in GitHub. Anyone can read, copy and modify to improve our application.

Risks

1. Lacking of latest API may arise many securely issues over time.

2. Deprecated API may not work on some devices.

Future plans

1. Switching to kotlin with latest API tools.

2. XML elements conversation to android jetpack compose UI.

3. Switching from java to kotlin for better library support.