DCT'S Dhempe College Of Arts And Science

Miramar - Goa

Project Conducted By

Department of Computer Science

TYBSC Project 2021-2022

Library Management Application

" LibMan "

Submitted By:-

-Akash Yerunkar

-Akhilesh Sathe

-Shreevesh Naik

-Vivek Cruz

CERTIFICATE

THIS IS TO CERTIFY THAT THE PROJECT REPORT ENTITLED LIBRARY MANAGEMENT APPLICATION, IS A RECORD OF WORK DONE BY CANDIDATES, DURING THE PERIOD OF STUDY UNDER OUR GUIDANCE TO THE BEST OF OUR KNOWLEDGE. IT HAS NOT PREVIOUSLY FORMED THE BASE FOR THE AWARD OF ANY DIPLOMA OR DEGREE BY THIS OR ANY OTHER UNIVERSITY OR ELSEWHERE

_____	_____	_____
Ms. Annie Rajan	(External Examiner)	Ms. Annie Rajan
(Project Guide)		(HOD, Computer Science Dept.)

DECLARATION

We declare that this project report entitled "Library Management Application" has been prepared by us and to the best of our knowledge. It has not previously formed the basis for the award of any diploma or degree by this or any other university.

Class: T.Y.B.Sc.

Division: Computer Science

Name	Roll no.	Signature
Akash Yerunkar	19292	
Akhilesh Sathe	19264	
Shreevesh Naik	19275	
Vivek Cruz	19283	

ACKNOWLEDGEMENT

Life is journey of excellence. Every mile that one reaches during the external journey is marked by guidance & support of the near & dear once & this endeavor of mine is no inception. We extend our kind acknowledgement to project incharge & our professor Ms. Annie Rajan for giving us this precious & useful Opportunity to improve our practical knowledge, we are also thankful to all our teaching members Mr. Snehal Harmalkar, Ms. Shilpa Naik Ms. Arati Kadam, Mr. Senon D`Souza for giving us moral support & confidence throughout the project, they provided all the data formats, full guidance, and other valuable help. Suggestion for further improvement will be gratefully received and acknowledged.

ABSTRACT

This project explores the use of Android Studio for android application development. Here, is an android application developed using the Android Studio software and Firebase Database. In a Library, the librarian always needs to keep a track of the newly added books, and books issued to users, also if any book is returned or not, and if not find the user who has borrowed the book and then alert the user for the due date. Also, the user is required to go to the library to hunt down the book the user requires, carry the id along with him/her, check if the library is open or not, and keep track of the due date to return the book. Here a Library Management Application assists both the librarian and the user a lot. The application assists the Librarian by keeping all the records in the cloud database, which can be referred any time to track the books available in the library. Also, the librarian can update the newly arrived books in the database with help of the application. Edit the information of the book or delete any books from the database if required. The application also helps librarians to keep track of borrowed books and the user who have borrowed the books. On another hand, it also helps users to browse the books present in the library without the need to physically be present in the library. Also, scan a QR code on the book and issue the book, locate the books available through the application, see if the library is open or close, get pre-notification for the due date of the book, get to see the newly arrived books, and popular books. A Library Management application help's both the librarian (Admin) and Student (users) a lot more than the traditional way of managing a library.

Table Of Content

CHAPTER-1

1. INTRODUCTION

LibMan (Library Management Application) is a mobile application developed using Android Studio. This application aims to make managing a library and its members possible just using a smartphone, running on Android operating system.

1.1. User Levels

Our App has two types of user levels:

- Admin – The Librarian
- User – The Member

The user is greeted with a login screen where they can either enter their credentials or choose to create a new account.

If the credentials are valid, then the app gives the user an appropriate user interface, according to their user level.

1.2. DEVELOPMENT

The application is developed over the academic year from 2021 to 2022 with analysis in the first term and design code for the second term. The group members were assigned different tasks for the app development. All members of our team act as lead programmers who frequently contribute to input design and code decisions.

1.3. DESIGN

From the beginning of the project, the team placed a tremendous emphasis on both the design and function aspects of our application. Our main task is to design an application that will be simple for users to understand. We choose to keep the UI user-friendly to all ages of users specifically enthusiasts so that the user can easily access the app and issue the book without any hassle or delay. In this application, we choose the Android Studio to create an application that would draw users' attention and hold their interest.

1.4. ART

The art of “LibMan” is Library Management Application. This style is

intended for an artistic, creative, and user-friendly approach to our application. All art assets were downloaded from the internet, then transferred into Android Studio for its powerful networking capabilities with the source of the application. The application is designed such that the user of the application be it the user or the admin need not require any training or walk through the application interface. The design and flow of the application are simple and easy to understand by any new user soon after successful login. Our team set out to create necessary assets and logos with quality that will be best to suit the application.

1.5. CONTEXT

The LibMan application is basically a Library Management Application. LibMan lets users explore the library and borrow books and assist librarians. The application requires the user and admin to log in with valid credentials and after logging they can use the available features of the application like the user can: issue books through QR code scan, check available books in the library in real-time, see if the library is open or not, get an alert notification to return the book 1 day before the due date. On the other, application also assists the librarian very well as the librarian being admin can: add or delete the books from the database of the application, suspend a user if required, notify use for the due date, generate the QR code for books, see the borrowed books list, available books in the library, see and set the location of the books available, edit books information.

1.6. TECHNOLOGY

Expand your knowledge and experience of working with Android Studio by creating a simple application. Using basic assets provided by Android Technologies, built-in components, and writing simple custom code, understand how to work with imported dependencies, Inbuilt classes, fragments, activities, and materials while practicing more complicated scripting principles. User can import all your assets and arrange them in the editor. This is where user build your interfaces, workings, intents, and more; it's where user add interaction via scripting, simultaneous test and edit application, and deploy it to your chosen android mobile platforms, there are different views, or windows, in the editor that contain the drag and drop tools and workflows for creating your application. Android Studio is android app development software it has built-in mono developer software used for script editing. It supports high-level programming languages such as java and kotlin. Its design philosophy emphasizes code readability, and its syntax allows programmers to express concepts in fewer lines of code than would be possible in languages such as “C”. The language provides constructs intended to enable clear programs on both a small and large scale.

1.7.1. INPUTS

The input for the application is

1) Enable the camera permission from the settings
2) User e-mail address.

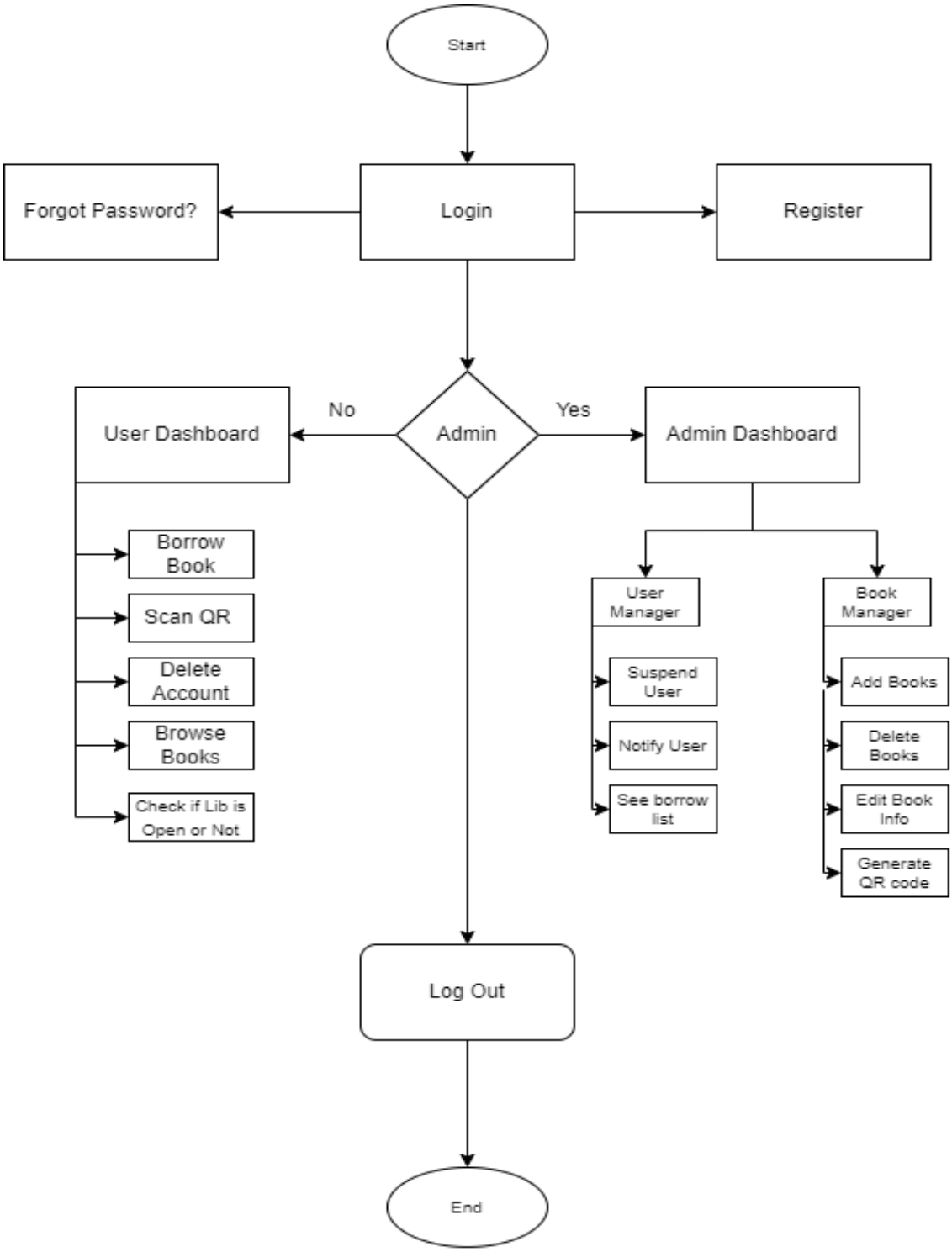
1.7.2. OUTPUTS

The output of the application is

1) The user can use the camera to scan the QR code on the book to issue it.

CHAPTER-2
SYSTEM DESIGN

2.1 FLOW CHART



2.2 DEVELOPMENT PLATFORM

2.2.1. HARDWARE USED:

Hardware	Specifications		
	System 1	System 2	System 3
Processor	Intel core i7 8th Gen	AMD Ryzen 3 1300x	AMD A6 7310
RAM	16 GB	8 GB	4GB
Graphic Card	MX 250	Nvidia 1050ti	Radeon Graphics
Device Used for Debugging and Testing	Redmi Note 10	Asus Zenfone Max pro M1	Samsung Galaxy J5 Prime
Monitor	16” Full HD Monitor	22” Full HD Monitor	14” LED Monitor

2.2.2. SOFTWARE USED:

- Android Studio version 2021.2.1 [1] was used for the development of the LibMan application. Android studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service in 2021. It is a replacement for the Eclipse Android Development Tools (E-ADT) as the primary IDE for native Android application development.
- Firebase [2] was used for database and user authentication Firebase is a platform developed by Google for creating mobile and web applications. It was originally an independent company founded in 2011. In 2014, Google acquired the platform and it is now their flagship offering for app development. Firebase is an app development platform that helps you build and grow apps and games users love. Backed by Google and trusted by millions of businesses around the world.

CHAPTER-3
APPLICATION FEATURES

3.1. Admin Features

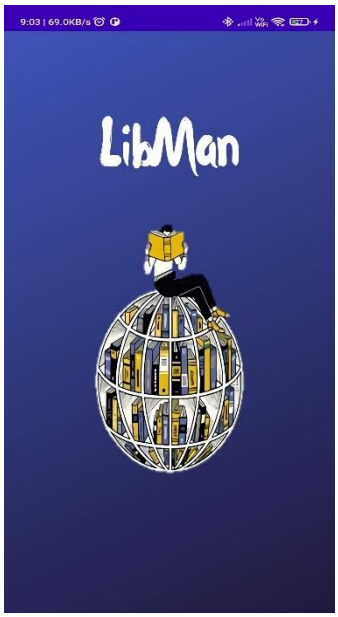
- a. Manage Books
  - i. Add Book
  - ii. Delete Book
  - iii. Edit Book
- b. Manage User
  - i. Suspend User
  - ii. Notify User
- c. Generate QR Code
- d. View Borrowed Books

3.2. User Features

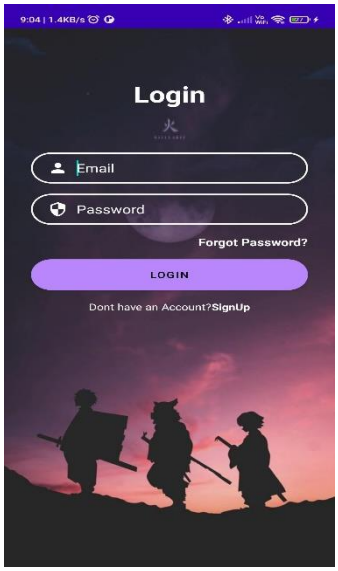
- a. Register/Login
- b. Search Library Books
- c. Borrow Books
- d. Check Availability of books
- e. Locate Books
- f. Scan QR code to get book Info
- g. Get due date notifications

CHAPTER-4
APPLICATION INTRODUCTION

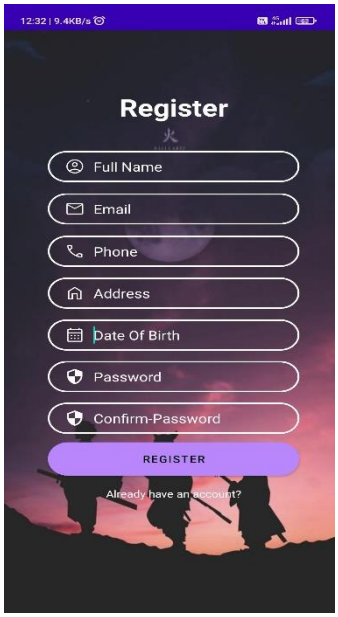
4.1. This is the screenshot is of the splash screen also know as welcome screen. This will be the first screen which the user will be seeing on application execution. It runs and stays on the screen for few seconds after which the main Login screen is flashed. It is basically the Introductory page of our application.



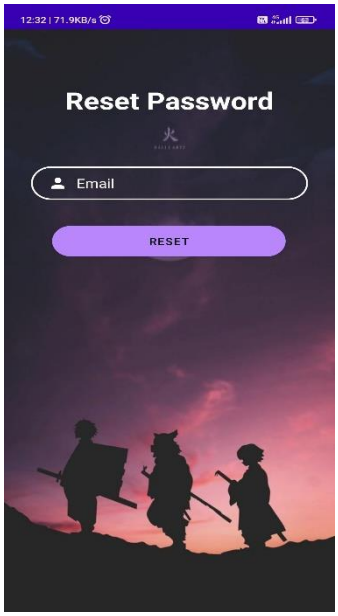
4.2. This is the screenshot is of the login screen which opens soon after the splash screen is stopped.



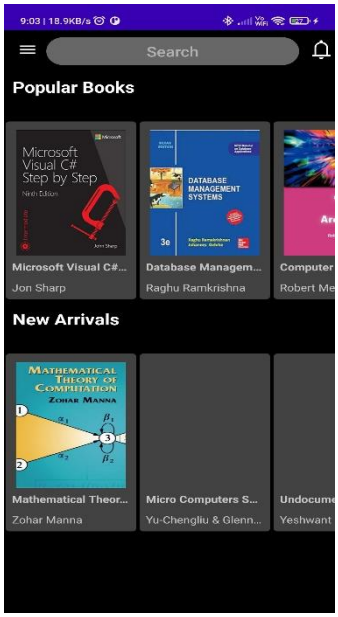
4.3. If New user wants to register himself. He/she can click on SignUp and register.



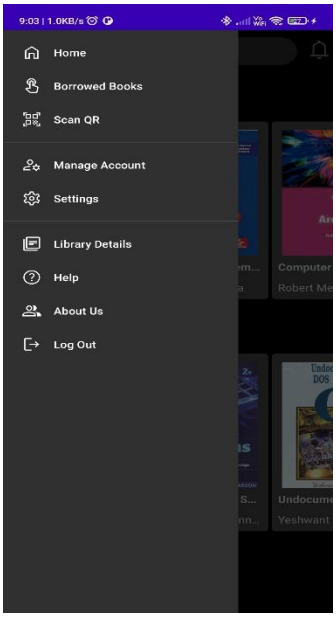
4.3 Also, if the user forgets his/her password then he/she can reset it by clicking on forgot password option.



4.4. This is the screenshot of the home screen for user which the user will be interacting after successfully login/registration.

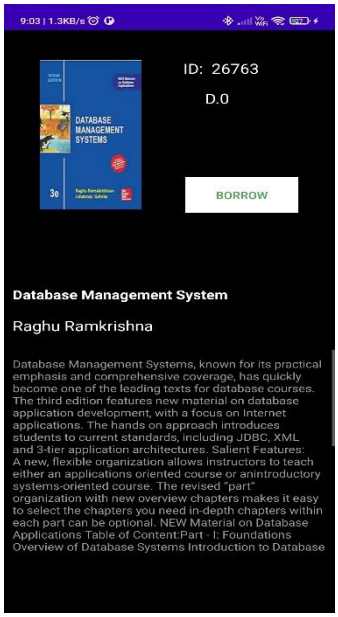


4.5. Navigation bar.

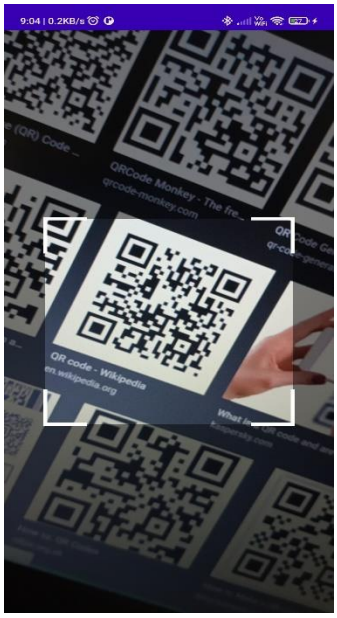


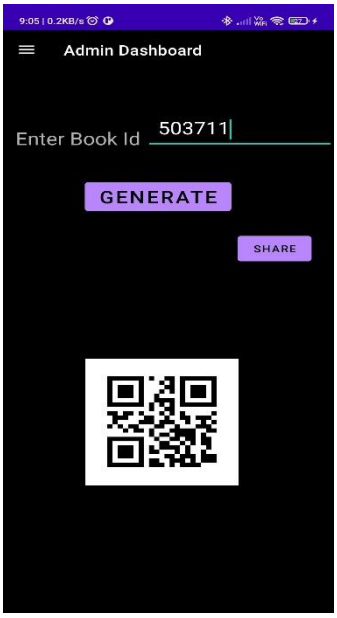
4.6. User Issued Book History.

4.7 Screenshot of Activity of Book Information.

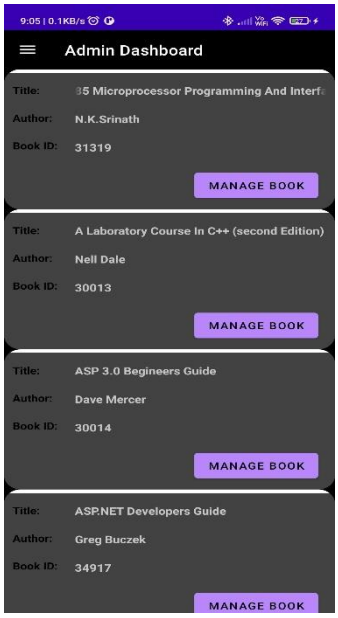


4.8 Qr Scanner and Generator

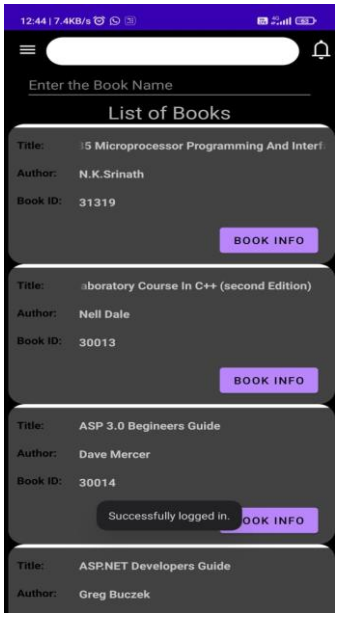




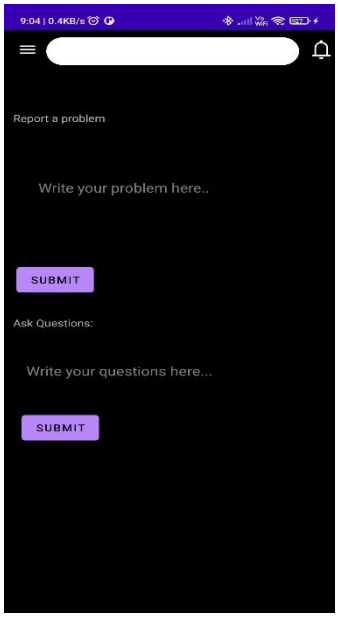
4.9 Book Manager Activity



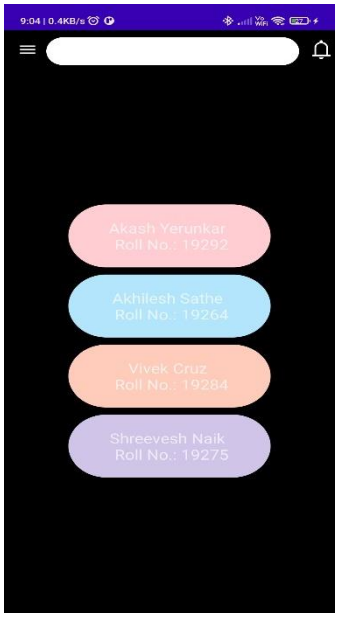
4.10 Search Activity



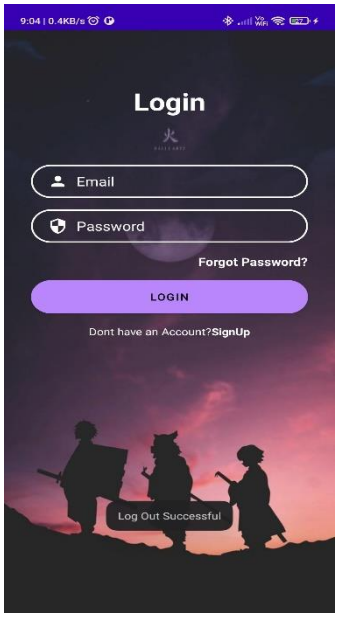
4.11 Help Activity



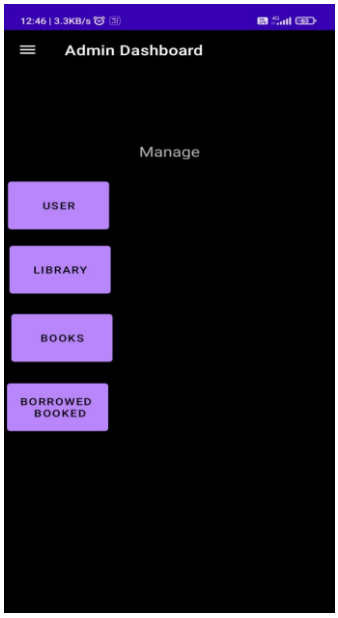
4.12 About Us Activity



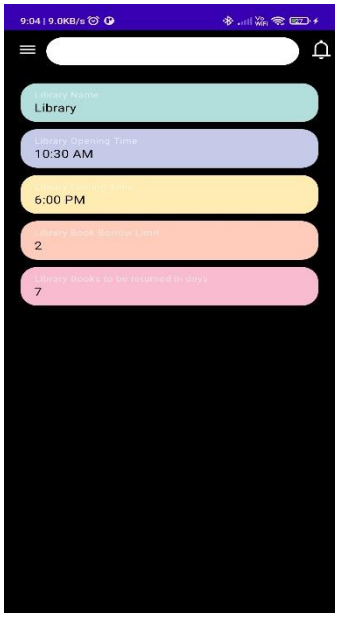
4.13 Logout: After user clicks logout. The user is logged out with a toast message.



4.14 Add Book Activity



4.15 Library Information Activity



CHAPTER-5
MAJOR FUNCTIONS USED

5.1. QR Reader Code [4]:

```
import com.budiyev.android.codescanner.CodeScanner;
import com.budiyev.android.codescanner.CodeScannerView;
import com.budiyev.android.codescanner.DecodeCallback;
import com.google.zxing.Result;

public class MainActivity extends AppCompatActivity {
    private CodeScanner mCodeScanner;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        CodeScannerView scannerView = findViewById(R.id.scanner_view);
        mCodeScanner = new CodeScanner( context: this, scannerView);
        mCodeScanner.setDecodeCallback(new DecodeCallback() {
            @Override
            public void onDecoded(@NonNull final Result result) {
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        Toast.makeText( context: com.example.qr_reader.MainActivity.this, result.getText(), Toast.LENGTH_SHORT).show();
                    }
                });
            }
        });
```

5.2. QR Generator Code [4]:

```
public class MainActivity extends AppCompatActivity {
    EditText editText;
    Button btn;
    ImageView display;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        editText = findViewById(R.id.edit_input);
        btn = findViewById(R.id.bt_generate);
        display = findViewById(R.id.iv_qr);

        btn.setOnClickListener(view -> {
            generateQR();
        });
    }
    private  void generateQR(){
        String text = editText.getText().toString().trim();
        MultiFormatWriter writer = new MultiFormatWriter();
        try {
            BitMatrix matrix = writer.encode(text, BarcodeFormat.QR_CODE,  width: 500, height: 500);
            BarcodeEncoder encoder = new BarcodeEncoder();
            Bitmap bitmap = encoder.createBitmap(matrix);
            display.setImageBitmap(bitmap);
        }catch (WriterException e)
        {
            e.printStackTrace();
        }
```

5.3. Code snippet for the login and registration of user and admin:

```
public class Login extends AppCompatActivity {

    TextView btn,rstPass;
    EditText inputEmail,inputPassword;
    Button btnLogin;

    private FirebaseAuth mAuth;
    ProgressDialog mLoadingBar;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        btn = findViewById(R.id.textviewSignup);

        rstPass = findViewById(R.id.forgotpassword);

        inputEmail = findViewById(R.id.inputUser);
        inputPassword = findViewById(R.id.inputPassword);
        btnLogin = findViewById(R.id.btnRst);
        btnLogin.setOnClickListener(view -> checkups());

        mAuth = FirebaseAuth.getInstance();
        mLoadingBar =new ProgressDialog( context: Login.this);
        btn.setOnClickListener((v)-> startActivity(new Intent( packageContext: Login.this,Register.class)));

        rstPass.setOnClickListener(view -> startActivity(new Intent( packageContext: Login.this,ResetPassword.class)));
    }
```

```
    private void checkups() {
        String email = inputEmail.getText().toString().trim();
        String password = inputPassword.getText().toString();
        if (email.isEmpty() || !email.contains("@") )
        {
            showError(inputEmail, s: "Email is not Valid!");
        }
        else if(password.isEmpty() || password.length()<6)
        {
            showError(inputPassword, s: "At least 6 character");
        }
        else
        {
            mLoadingBar.setTitle("Login");
            mLoadingBar.setMessage("Please wait while processing");
            mLoadingBar.setCanceledOnTouchOutside(false);
            mLoadingBar.show();

            mAuth.signInWithEmailAndPassword(email,password).addOnCompleteListener(task -> {
                if (task.isSuccessful())
                {
                    mLoadingBar.dismiss();
                    activityStarter();
                    Toast.makeText( context: this, text: "Successfully logged in.", Toast.LENGTH_LONG).show();



                }
                else {
                    mLoadingBar.dismiss();
                    Toast.makeText( context: this, text: "Login or Password invalid.", Toast.LENGTH_LONG).show();
                }
            });
        }
    }
```

CHAPTER-6
CONCLUSION

There are also many other applications to this like our developed application in the market like “Library Thing”, “Libib”, “Book Buddy”, and many more, among which good reads is an application which has similar working as the application developed by us. The main advantage of using our application is that the user can simply scan the QR code stick on the book to issue the book and does not need to wait for the librarian to be present also the user does not need to carry a library id card along with him/her. One more feature of our application that makes it different from other applications is our application makes it easy for the user to trace exactly on which shelf the book is kept in absence of any librarian. Hence it is more beneficial than other applications in the market also it provides well assistance to the librarian in case he/she wants to trace to whom the book is currently issued or wants to send a reminder on returning the book back.

It was indeed a great experience as a team as we feel more confident about the successful completion of our project. Thank You

BIBLOGRAPHY

[1] Android Studio Documentations:

Developer.android.com n.d. Documentation *for app developers*
Available at: <https://developer.android.com/docs>
[Accessed 1 April 2022].

[2] Firebase Documentations:

firebase.google.com. n.d. *Firebase |Learn the fundamentals*. [online]
Available at: <https://firebase.google.com/>
[Accessed 14 APRIL 2022].

[3] Firebase UI:

firebaseopensource.com. n.d. *Firebase FirebaseUI for Android — UI Bindings for Firebase*. [online]
Available at: <https://firebaseopensource.com/projects/firebase/firebaseui-android/>
[Accessed 14 APRIL 2022].

[4] Zxing android embedded

github.com. n.d. ZXing Android Embedded [online]
Available at: <https://github.com/journeyapps/zxing-android-embedded/>
[Accessed 14 APRIL 2022].

[5] Stack overflow

stackoverflow.com n.d. Stack Overflow - Where Developers Learn, Share, Build Careers
Available at: <https://stackoverflow.com/>
[Accessed 1 April 2022].