MINI PROJECT

(Group 1)

RFID Based Surveillance System for School Bus

BACHELOR OF TECHNOLOGY
IN COMPUTER SCIENCE &ENGINEERING
BY

ADHILA MP(PRP19CS005)
AMAL SOMAN(PRP19CS012)
HARIKRISHNAN KB(PRP19CS029)
GOUTHAM KRISHNA S(PRP19CS026)

Under the guidance of

Aggie Varghese

Academic batch:(2019-2023)

CONTENTS

- Introduction
- Existing model
- Problem Statement
- Objectives
- Scope
- Literature Survey
- Technology stack
- Methodology
- System Design
- System Implementation
- Results And Discussion
- Conclusion
- References

INTRODUCTION

A school bus security system should be able to:

- Monitor the vehicle and ensure students remail safe to and from school.
- Capture the bus’s interior for comprehensive monitoring.
- RFID integration to track the entry of students

EXISTING MODEL

Sl. No	Approach	Methodology	Advantages	Disadvantages
1	Mobile Surveillance	Development of Internet of Things (IoT) monitoring system with CCTV	• Mobile Surveillance • Attendance System	• No video recording system
2	RFID Based Attendance System	Radio frequency communication using IoT	• Realtime and accurate daily attendance • No roll calls required	• Data maintenance difficulty

PROBLEM STATEMENT

- To ensure that the children board the school bus correctly and that they have no other issues on the bus.
- The video of the student is delivered live through an app.
- RFID integration to track the entry of students

SCOPES

- Parent Monitoring
- Student Security
- Privacy Security
- Realtime Service
- Native Application Interface

OBJECTIVES

- Parent can access footage by login with student id .
- Parent can record video for any kind of evidence .

Literature survey

Year	**Author**	**Title**	**Methodology**	**Conclusion**
2020	Shanthan Kasarla	RFID Based attendance system	Radio frequency Communication Using RFID	RFID transceiver that communicates with a passive Tag when the tag enters the generated rf field it is able to draw enough power from the field to access its internal memory and transmit its stored information .when the transponder tag draws power in its way the resultant interaction of the rf fields causes the voltage at the tranciever antenna to drop in value.This effect is utilized by the tag to communicates its information to the reader.

Year	Author	Title	Methodology	Conclusion
2021	Dong-Ying Li, Shun-Dao Xie, Rong-Jun Chen, Hong-Zhou Tan	Design of Internet of Things System for Library Materials Management using UHF RFID	Radio frequency communication using Iot	This System for Library Materials Management using Android based UHF mobile reader (Android mobile reader) as its entry to increase the efficiency of library materials management. The functions of the Internet of Things System for Library Materials Management include user identification, inventorying, adding, refreshing, searching, and self-help borrowing & returning library materials.

Year	Author	Title	Methodology	Conclusion
2020	Thomas Lee Scott, Amna Eleyan	CoAP based IoT data transfer from a Raspberry Pi to Cloud	Development of an Internet of Things (IoT) monitoring system using ThingsBoard IoT platform.	Things board is an open source software tool, which is used to collect, monitor and visualise streams of data received in real-time by sensor devices. The platform can be hosted in the cloud and provides Message Queuing Telemetry Transport (MQTT), The Constrained Application Protocol (CoAP) and Hyper Text Transfer Protocol (HTTP) protocols support. A CoAP-based IoT architecture is proposed using a Raspberry Pi (RPi) and sensors acting as IoT endpoints and it will poll sensors and using CoAP send the latest data formatted as JavaScript Object Notation (JSON) to the ThingsBoard cloud endpoint at regular intervals.

Year	**Author**	**Title**	**Methodology**	**Conclusion**
2021	Dr. G. G Sivasankari Prerana G Joshi	Live Video Streaming using Raspberry Pi in IOT Devices	Livestreaming using RaspberryPi and camera	The system uses the algorithm to significantly decrease the storage space and to save the cost. The algorithm is implemented on the Raspberry Pi, which provide the live streaming with motion detection. The live steaming can be viewed from any web browser or even from mobile in the real time.

Year	Author	Title	Methodology	Conclusion
2020	Kuei-Chung Chang, Po-Kai Liu	Design of Adaptive Coding Approach to Support Video Streaming for Camera Surveillance Systems	Livestreaming using RaspberryPi and camera	In this paper, we design a video streaming and image processing server to support remote surveillance monitoring. The camera nodes are based on Raspberry Pi3, and the smartphone can receive video streams from the server. we demonstrate the proposed system can adjust the quality of video streaming automatically when network bandwidth drops suddenly. Once the network bandwidth is restored, the proposed system can also switch the stream.

Year	**Author**	**Title**	**Methodology**	**Conclusion**
2021	Jordan Filteau, Suk Jin Lee, Andrew Jung	Real-Time Streaming Application for IoT Using Raspberry Pi and Handheld Devices	IoT Video streaming and Android app receiver using json format file transfering	Develop a cost-effective solution to real time video streaming by using a low-cost ARM (Advanced RISC Machines)-based computer, i.e. Raspberry Pi (RPi). Develope a simple streaming server written in Java that runs efficiently on the RPi and showed that the RPi can support streaming services to multiple devices once and develop a light-weight android application that the server can stream to. The client and server communicate via JSON (JavaScript Object Notation).Divided high quality video into image files with the jpg format, into packets and then transmitted them to multiple handheld devices simultaneously.

Year	Author	Title	Methodology	Conclusion
2021	Neel Oza, N. B. Gohil	Implementation of cloud based live streaming for surveillance	Video streaming to random cloud using Raspberry Pi and FFMPEG based USB Camera	Rapid technological growth made surveillance as most promising application domain. With great extent of smart city most of the things are controlled by internet. Security is one of the applications that everyone needs to be controlled remotely. This paper presents cloud based surveillance system for live video streaming that can be surveillance from anywhere and anytime. This system provides the live streaming by using cloud; Raspberry Pi 2 module and FFMPEG based USB Camera.

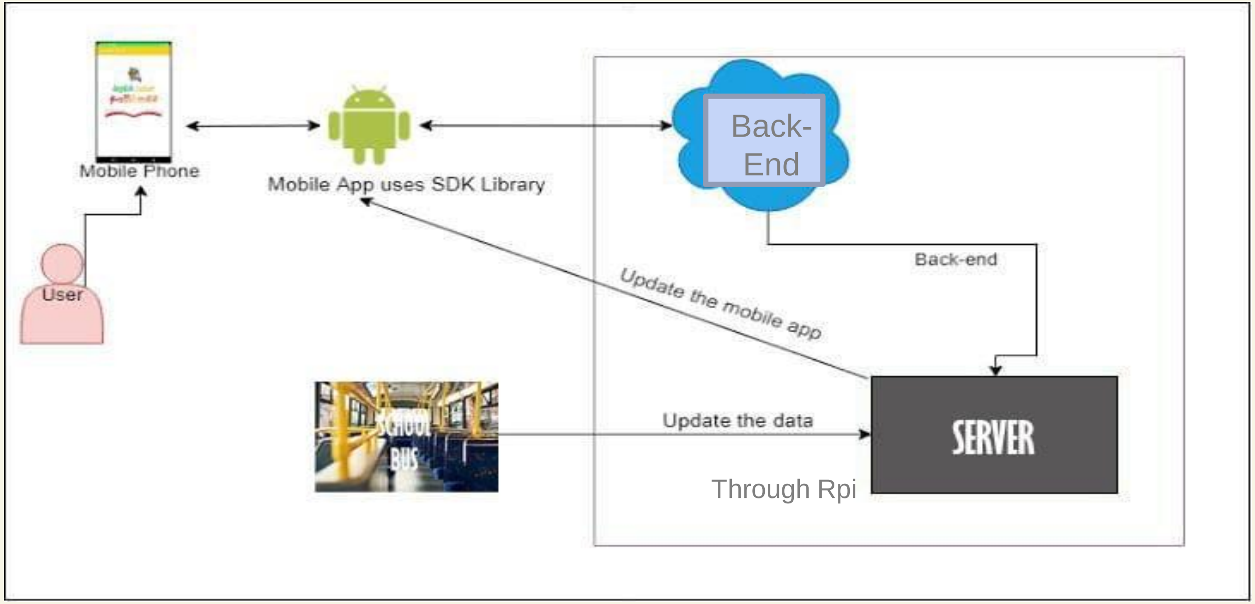
Year	Author	Title	Methodology	Conclusion
2021	Changqing Yin, Zhiguang Zhang	Research on Video Rendering on Android	Video rendering software development in Java Native Interface(Android)	Video-audio rendering plays an important role in multimedia players, which is also relates with the platform. It's known to all that Android has many versions, and one app can hardly support all versions. We introduce four methods of video rendering by using JNI(Java Native Interface), and present the process of implementation, According to the most efficient rendering way, we will develop android apps for video meeting and surveillance system.

TECHNOLOGY STACK

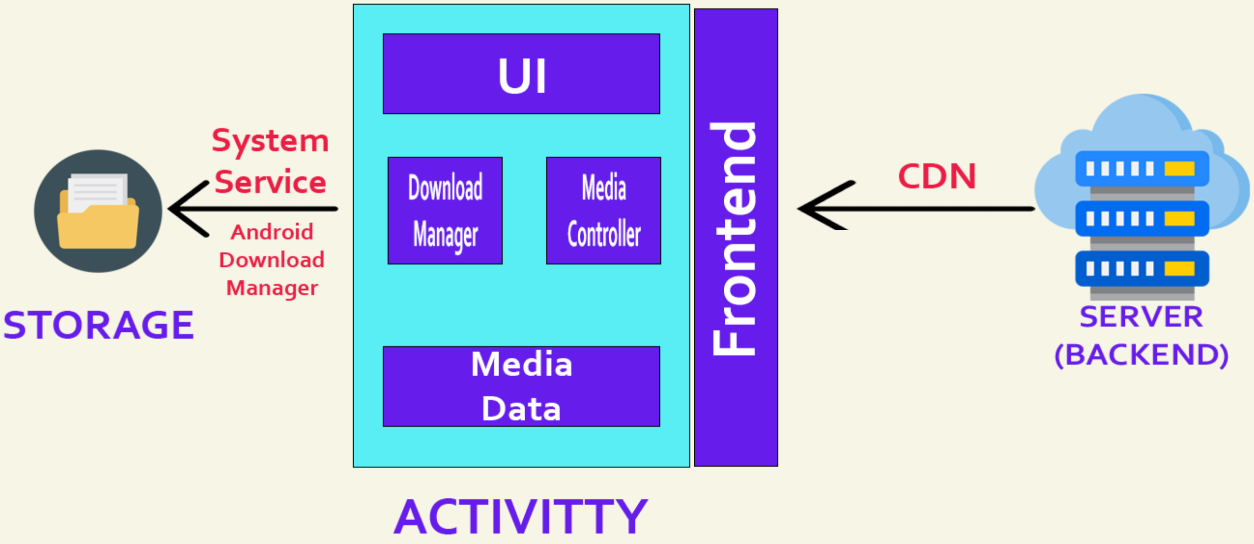
- Programming Languages Used : Kotlin , xml, python
- Frameworks : Kotlin(Java)
- Web Server : Localhost http Python Server
- Operating System : Raspbian OS, Android

METHODOLOGY

❑ System Architecture Design :



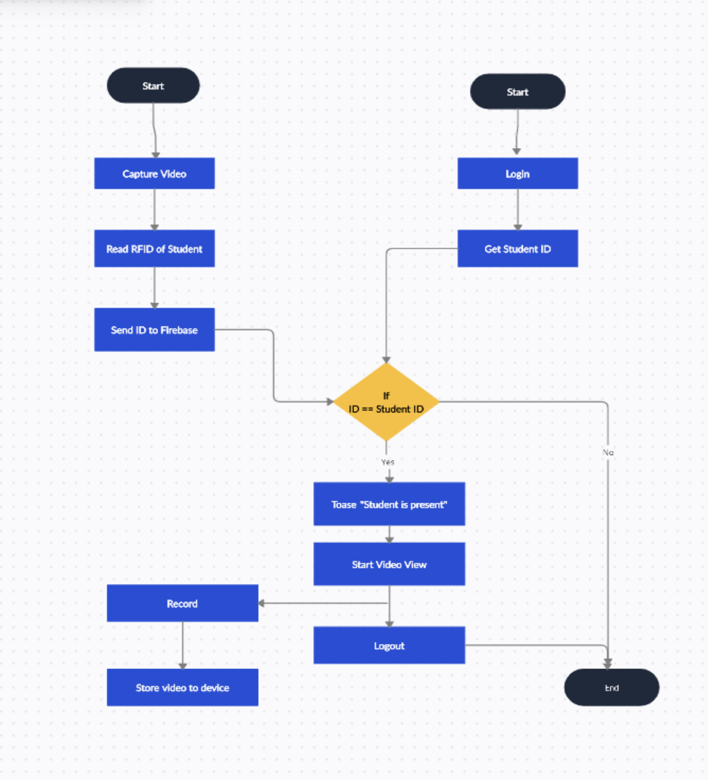
❑ Application architecture:



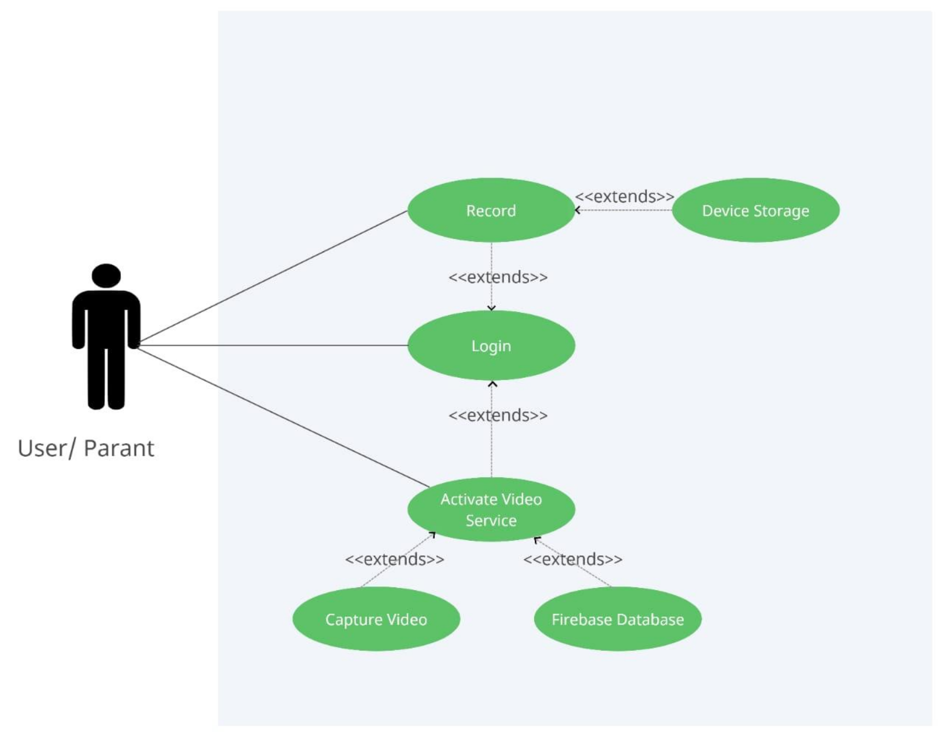
System Design

- ❑ Flowcharts
- ❑ Use Case Diagram
- ❑ Activity Diagram

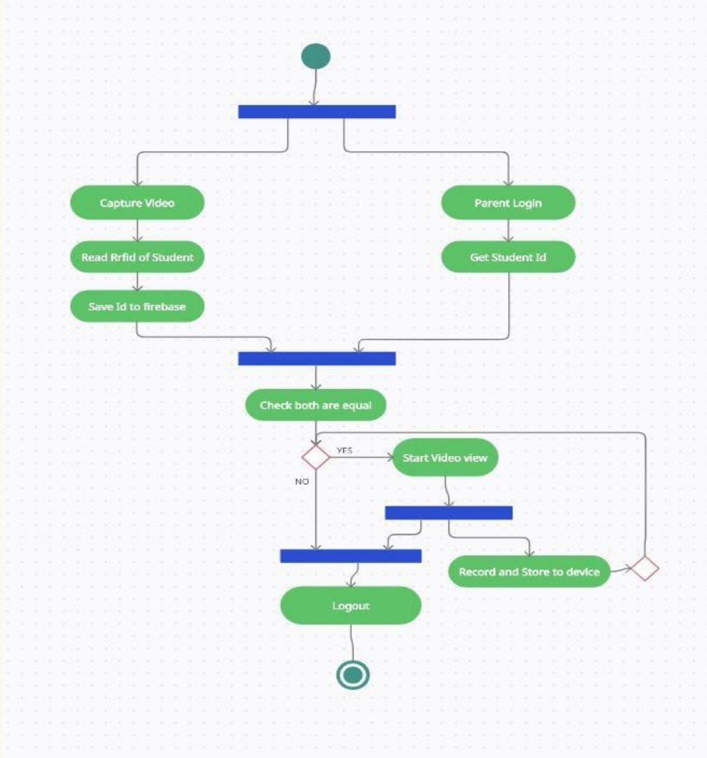
❑ Flow Chart


Start
Capture Video
Read RFID of Student
Send ID to Firebase
Start
Login
Get Student ID
If
ID == Student ID
Yes
No
Toase "Student is present"
Start Video View
Record
Logout
Store video to device
End

❑ Use Case Diagram



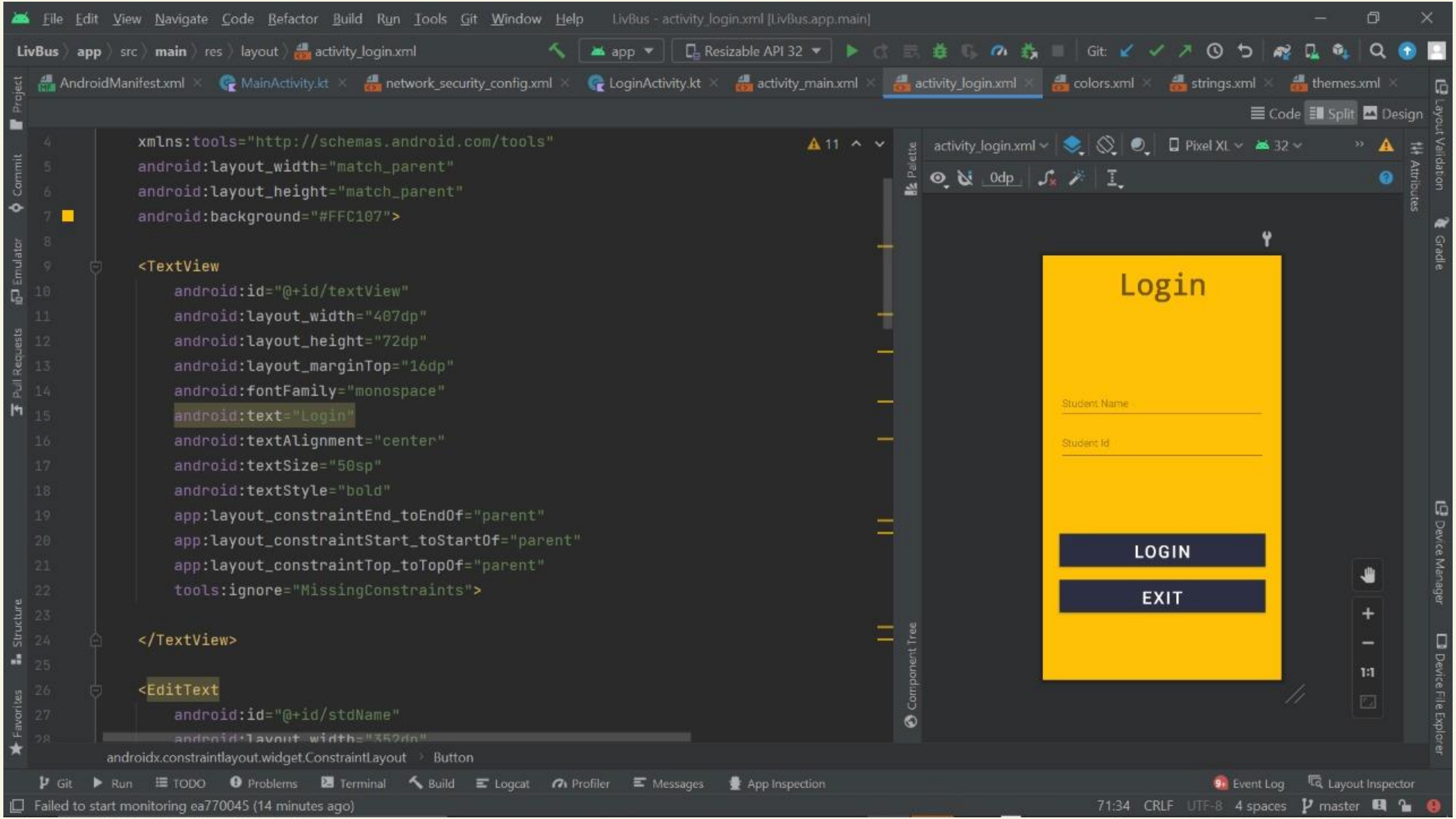
❑ Activity Diagram



SYSTEM IMPLEMENTATION

Software Implementation

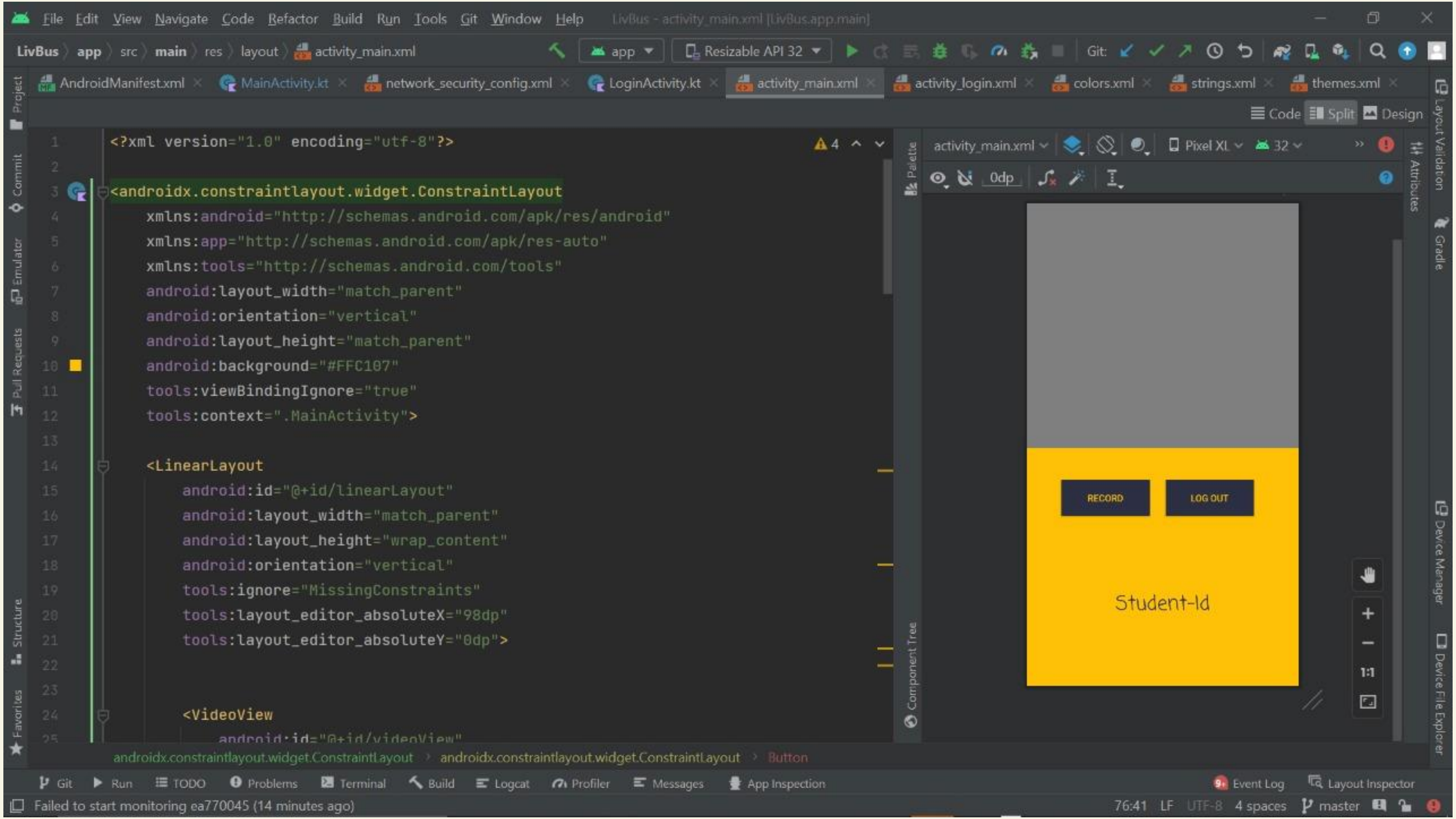
videoView.Start()	:	Start Video Player
Toast.makeText()	:	Display Toast Message
Finish()	:	Exist From Current Activity
Download()	:	Add Video To Device
DatabaseReference.get Instance().getReference()	:	Set Database Path In Firebase



```xml
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#FFC107">

    <TextView
        android:id="@+id/textView"
        android:layout_width="407dp"
        android:layout_height="72dp"
        android:layout_marginTop="16dp"
        android:fontFamily="monospace"
        android:text="Login"
        android:textAlignment="center"
        android:textSize="50sp"
        android:textStyle="bold"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        tools:ignore="MissingConstraints">

    </TextView>

    <EditText
        android:id="@+id/stdName"
        android:layout_width="352dp"
```

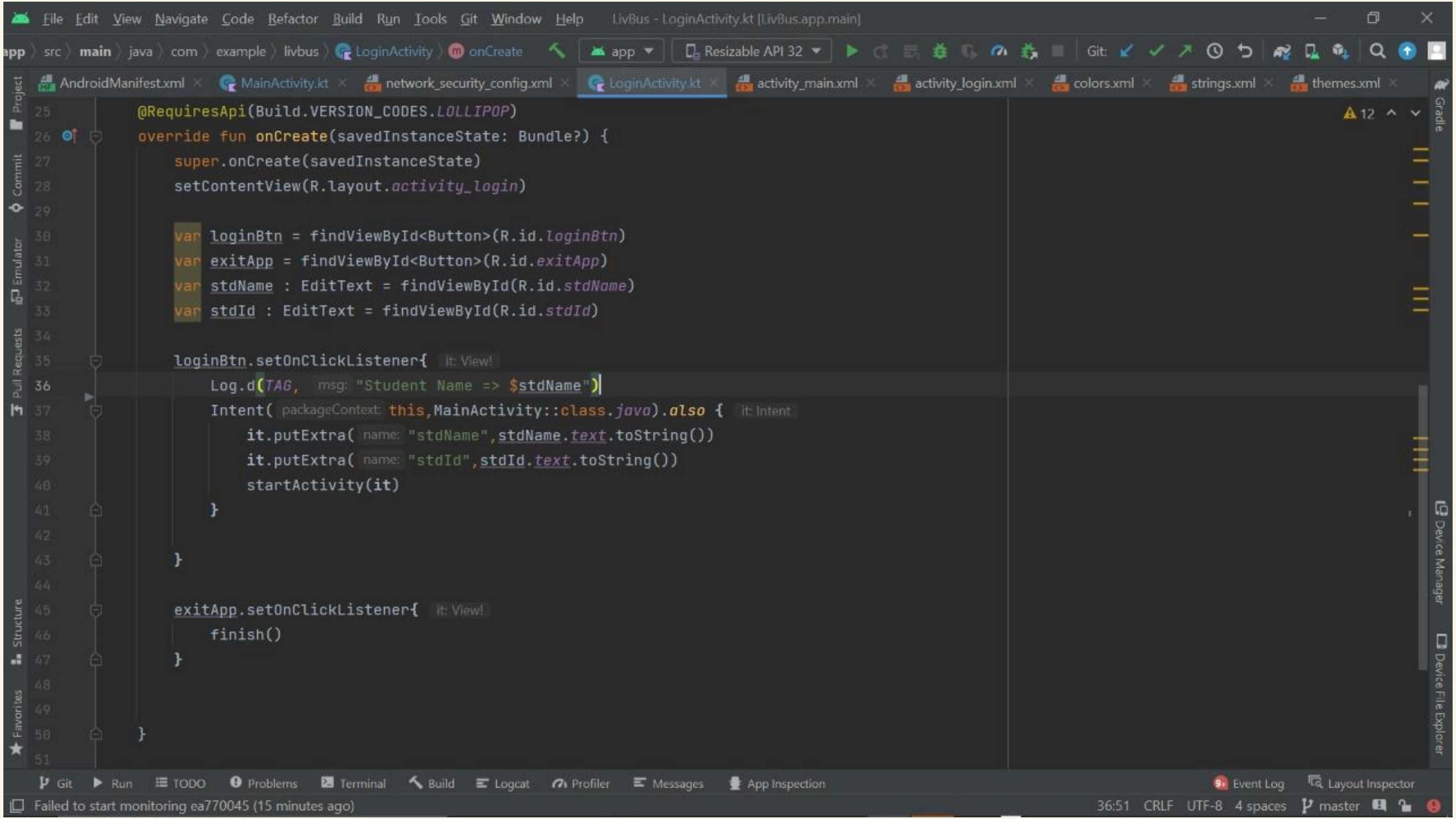
androidx.constraintlayout.widget.ConstraintLayout 〉 Button

Failed to start monitoring ea770045 (14 minutes ago)



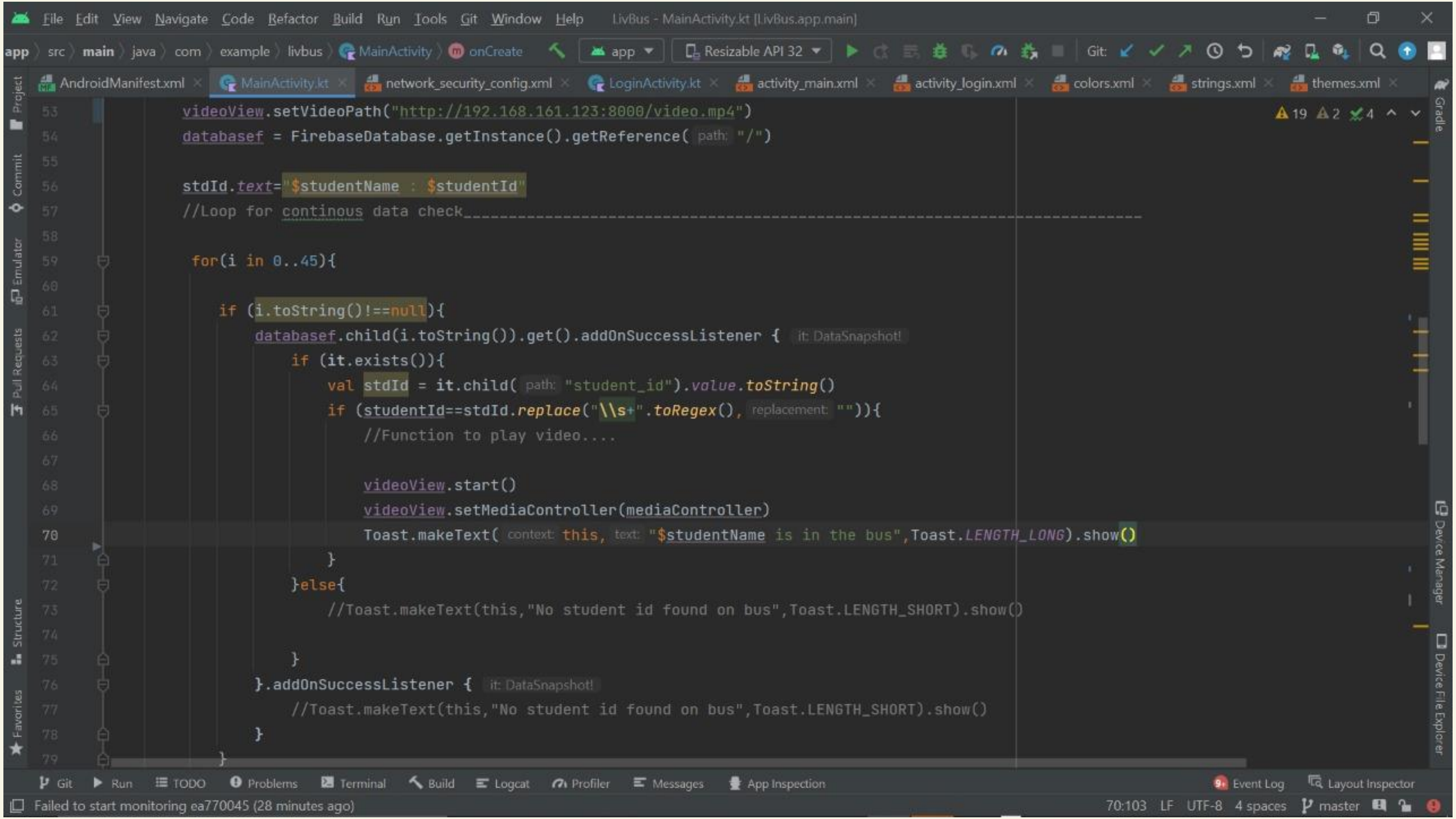
```
<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:orientation="vertical"
    android:layout_height="match_parent"
    android:background="#FFC107"
    tools:viewBindingIgnore="true"
    tools:context=".MainActivity">

    <LinearLayout
        android:id="@+id/linearLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        tools:ignore="MissingConstraints"
        tools:layout_editor_absoluteX="98dp"
        tools:layout_editor_absoluteY="0dp">


        <VideoView
            android:id="@+id/videoView"
```

Failed to start monitoring ea770045 (14 minutes ago)



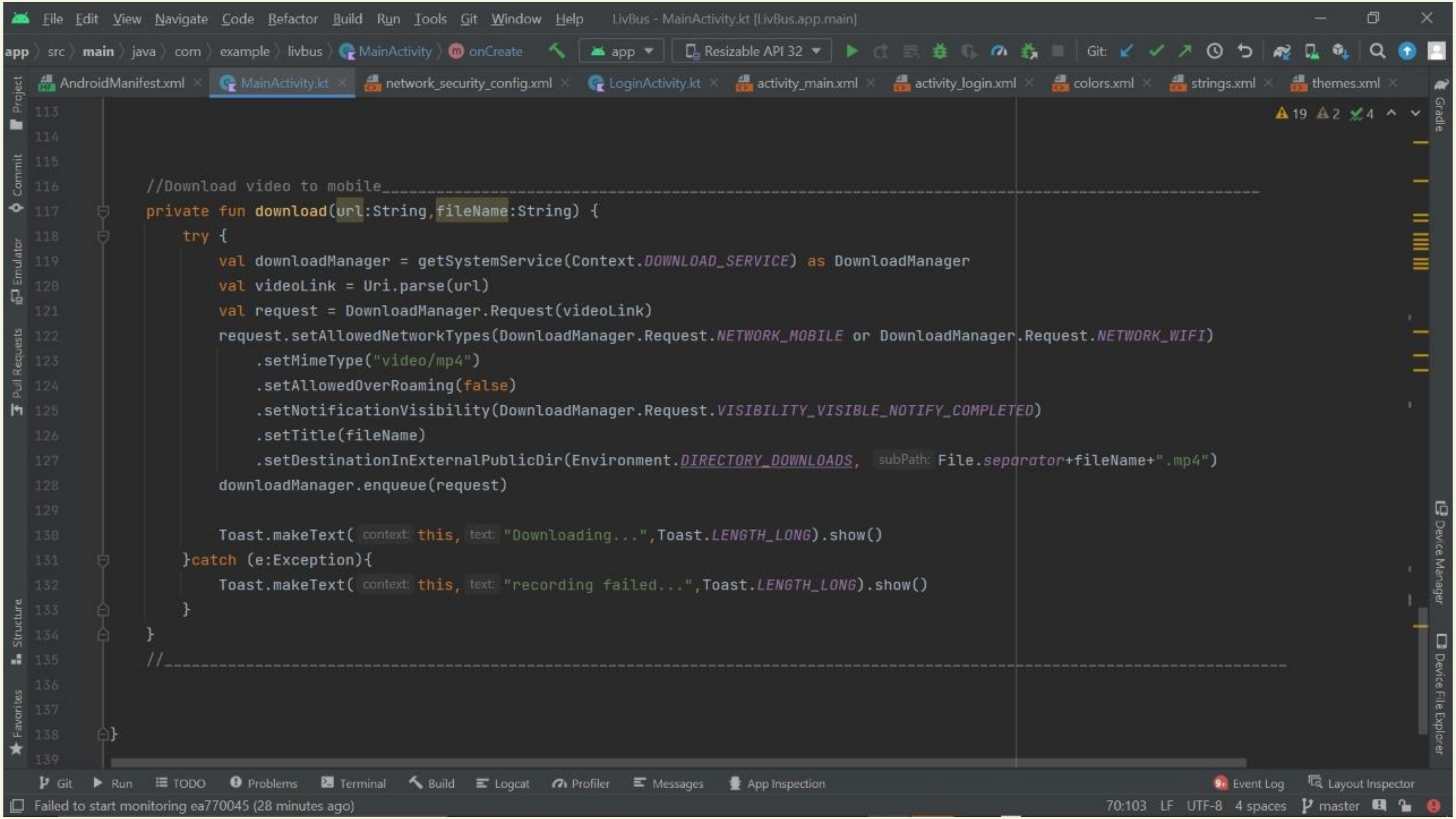
```kotlin
    @RequiresApi(Build.VERSION_CODES.LOLLIPOP)
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_login)

        var loginBtn = findViewById<Button>(R.id.loginBtn)
        var exitApp = findViewById<Button>(R.id.exitApp)
        var stdName : EditText = findViewById(R.id.stdName)
        var stdId : EditText = findViewById(R.id.stdId)

        loginBtn.setOnClickListener{ it: View!
            Log.d(TAG,  msg: "Student Name => $stdName")
            Intent( packageContext: this,MainActivity::class.java).also { it: Intent
                it.putExtra( name: "stdName",stdName.text.toString())
                it.putExtra( name: "stdId",stdId.text.toString())
                startActivity(it)
            }

        }

        exitApp.setOnClickListener{ it: View!
            finish()
        }


    }
```

Failed to start monitoring ea770045 (15 minutes ago)



```
videoView.setVideoPath("http://192.168.161.123:8000/video.mp4")
databasef = FirebaseDatabase.getInstance().getReference( path: "/")

stdId.text="$studentName : $studentId"
//Loop for continous data check__________________________________________________________________________

for(i in 0..45){

  if (i.toString()!==null){
    databasef.child(i.toString()).get().addOnSuccessListener {  it: DataSnapshot!
      if (it.exists()){
        val stdId = it.child( path: "student_id").value.toString()
        if (studentId==stdId.replace("\\s+".toRegex(),  replacement: "")){
          //Function to play video....

          videoView.start()
          videoView.setMediaController(mediaController)
          Toast.makeText( context: this,  text: "$studentName is in the bus",Toast.LENGTH_LONG).show()
        }
      }else{
        //Toast.makeText(this,"No student id found on bus",Toast.LENGTH_SHORT).show()

      }
    }.addOnSuccessListener {  it: DataSnapshot!
      //Toast.makeText(this,"No student id found on bus",Toast.LENGTH_SHORT).show()
    }
  }
```

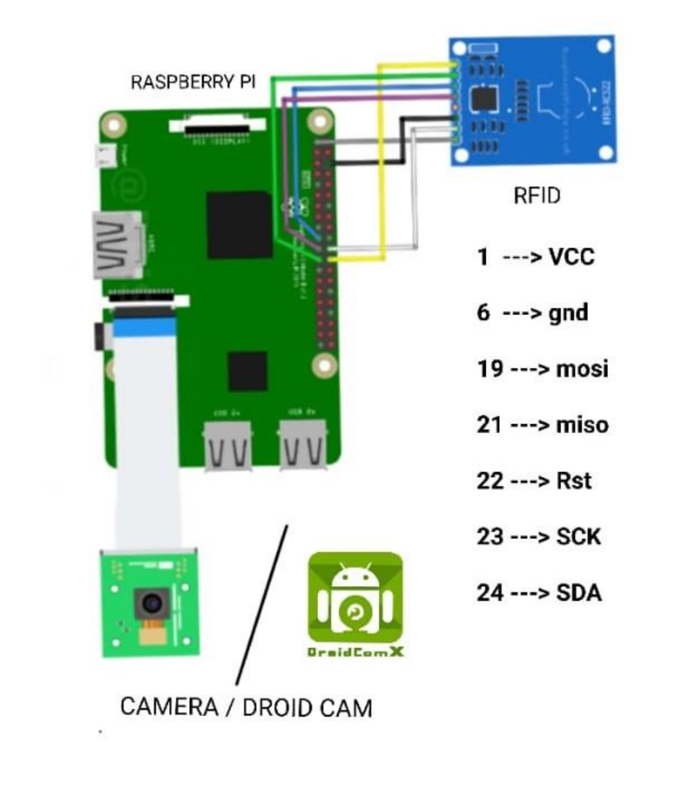
Failed to start monitoring ea770045 (28 minutes ago)



```
    //Download video to mobile___________________________________________________________________
    private fun download(url:String,fileName:String) {
        try {
            val downloadManager = getSystemService(Context.DOWNLOAD_SERVICE) as DownloadManager
            val videoLink = Uri.parse(url)
            val request = DownloadManager.Request(videoLink)
            request.setAllowedNetworkTypes(DownloadManager.Request.NETWORK_MOBILE or DownloadManager.Request.NETWORK_WIFI)
                .setMimeType("video/mp4")
                .setAllowedOverRoaming(false)
                .setNotificationVisibility(DownloadManager.Request.VISIBILITY_VISIBLE_NOTIFY_COMPLETED)
                .setTitle(fileName)
                .setDestinationInExternalPublicDir(Environment.DIRECTORY_DOWNLOADS, subPath: File.separator+fileName+".mp4")
            downloadManager.enqueue(request)

            Toast.makeText( context: this, text: "Downloading...",Toast.LENGTH_LONG).show()
        }catch (e:Exception){
            Toast.makeText( context: this, text: "recording failed...",Toast.LENGTH_LONG).show()
        }
    }
    //_____________________________________________________________________________________________


}
```

Failed to start monitoring ea770045 (28 minutes ago)

❑ Hardware Implementation



1 ---> VCC

6 ---> gnd

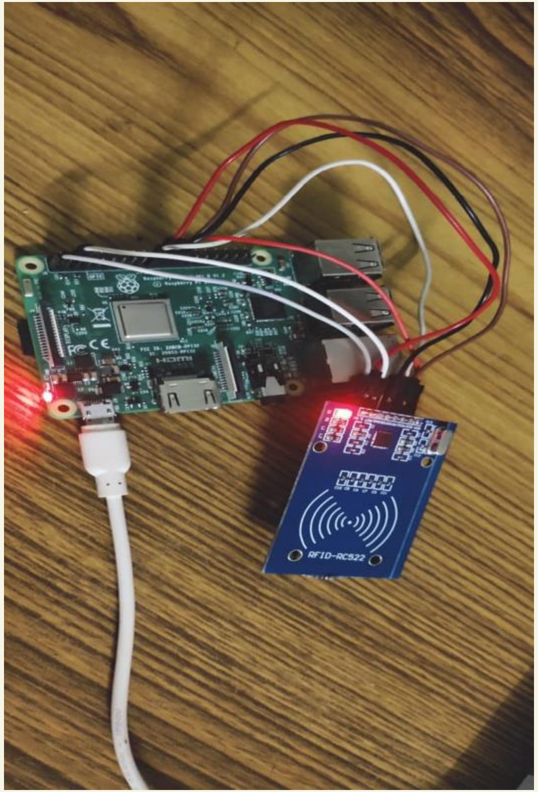
19 ---> mosi

21 ---> miso

22 ---> Rst

23 ---> SCK

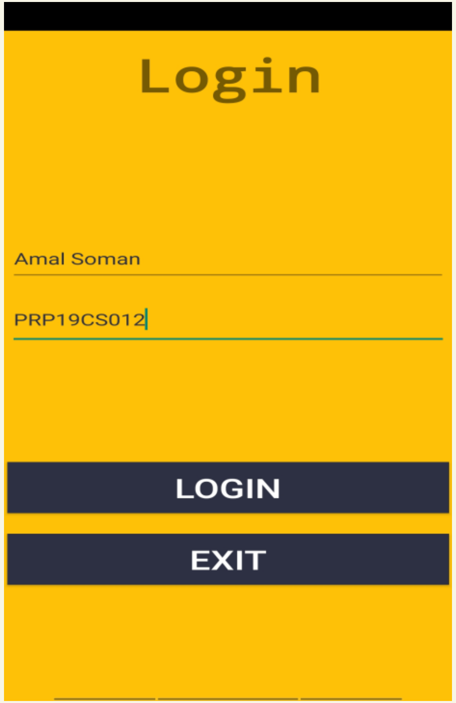
24 ---> SDA



Result And Discussion

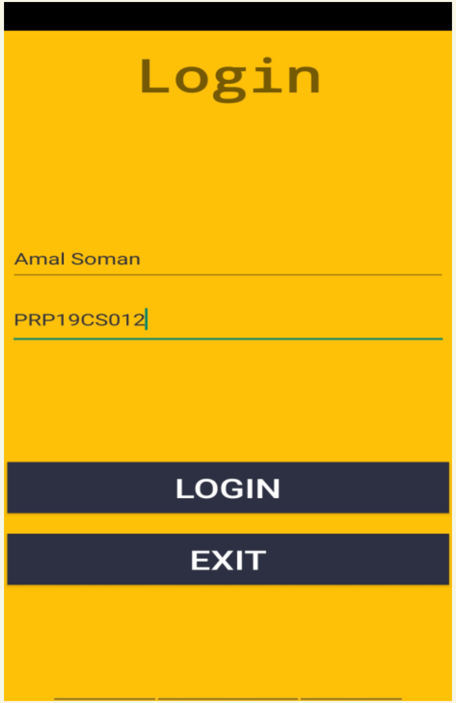
- ❑ Screenshots
- ❑ Comparison of Results
- ❑ Advantages
- ❑ Future works

❑ Screenshot:



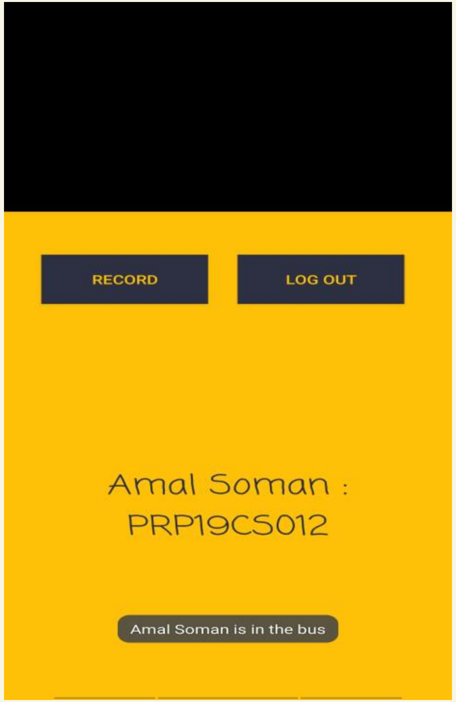
Amal Soman

PRP19CS012



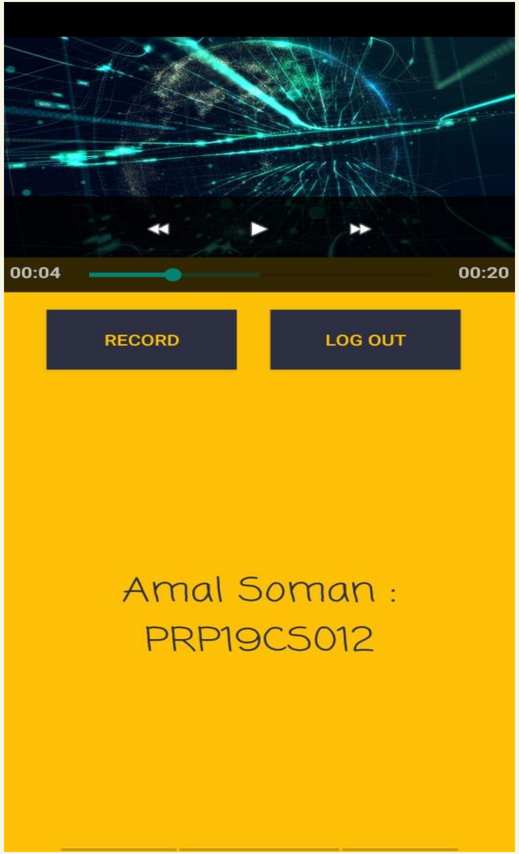
Amal Soman

PRP19CS012

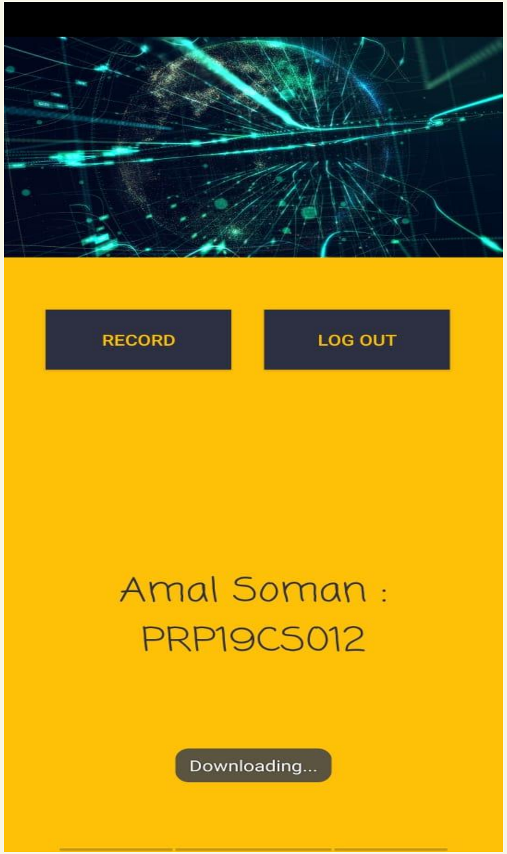


Amal Soman : PRP19CS012

Amal Soman is in the bus

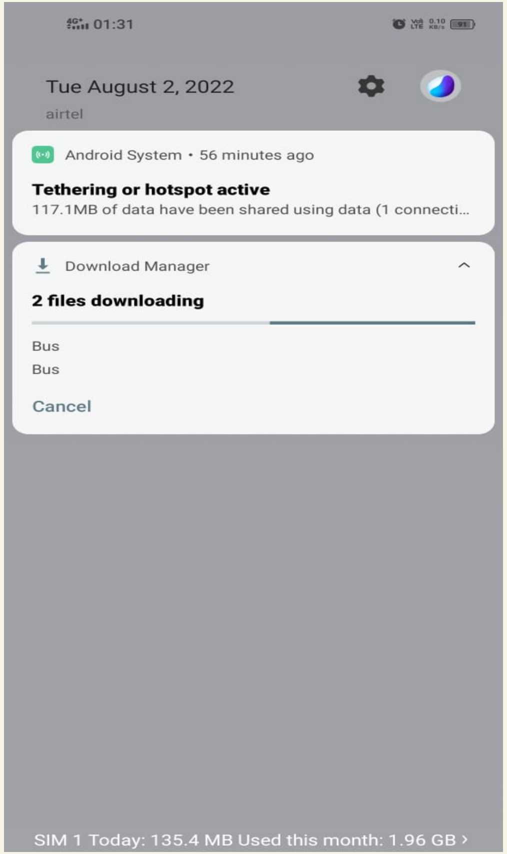


Amal Soman : PRP19CS012



Amal Soman : PRP19CS012

Downloading...



Tue August 2, 2022

airtel

Android System • 56 minutes ago

Tethering or hotspot active

117.1MB of data have been shared using data (1 connecti...

Download Manager

2 files downloading

Bus

Bus

SIM 1 Today: 135.4 MB Used this month: 1.96 GB ›

❑ Comparison Of Results

- In most of the existing model they only implements surveillance to admin, But we're providing parent monitoring and recording feature.

❑ Advantages

- Parent Monitoring
- Video can be record
- User friendly
- Attendance monitoring
- Enhanced students security
- Detect unauthorized student boarding

❑ Future Works

- Live Streaming
- Attendance report
- GPS tracking

CONCLUSION

- The RFID Based Surveillance System for School Bus provides secure travel for your child.
- Everything in School Bus is under supervision.
- Parents can view and record the video if they need.
- Hazards such as driving under the influence can easily monitored.
- This system uses camera for the safe travel of your child.
- In modern times, all we need is the security of lives and particularly, our children, A parent can perform several checks to make his or her child safe at home. But children today need to leave their homes for distinct reasons.
- For maximum reliability and supervised bus travel, you can count on RFID Based Surveillance System for School Bus .

REFERENCES

1. Dr. G. G Sivasankari Prerana G Joshi, " Live Video Streaming using Raspberry Pi in IOT Devices", *IEEE Access*, vol.10, pp.77068-77080, 2020.

2. Changqing Yin, Zhiguang Zhang, " Video rendering software development in Java Native Interface(Android)", *IEEE Access*, vol.6, pp.79038-79047, 2021.

3. Neel Oza, N. B. Gohil, “Implementation of cloud based live streaming for surveillance”, *IEEE Access*, vol.3, pp.167990 - 168003, 2021

4. Shanthan Kasarla, “RFID Based attendance system”, *IEEE Access*, vol.1,pp. 217 - 226, 2019

THANKS

Do you have any questions?

ADHILA MP(PRP19CS005)
AMAL SOMAN(PRP19CS012)
HARIKRISHNAN KB(PRP19CS029)
GOUTHAMKRISHNA S(PRP19CS26)

RFID Based Surveillance System for School Bus