**APSSDC– ANDROID DEVOLEPMENT**

**Project**

**on**

**TRAFFIC E-CHALLAN APP**

**by**

**2200100040 B.Dhinesh Reddy**



# KONERU LAKSHMAIAH EDUCATION FOUNDATION

**DEPARTMENT OF INTERNET OF THINGS**

Green Fields, Vaddeswaram, Guntur District-522 502

April- 2024

1

# ABSTRACT

The Traffic Challan App is an innovative Android application designed to streamline the process of issuing and managing traffic violation tickets (challans). By leveraging modern technology, the application aims to enhance the efficiency of traffic law enforcement agencies through the digitization of traffic violation issuance and recording. This shift from manual, paper-based processes to a digital platform reduces paperwork and minimizes human errors, thereby improving the overall accuracy and reliability of traffic enforcement. The app integrates with a centralized database, allowing for real-time processing and storage of violation records, which can be easily accessed and analyzed by authorities to make informed decisions and strategize traffic management more effectively.

Serving both traffic officers and violators, the Traffic Challan App provides a seamless and user-friendly interface that simplifies the recording of violations, capturing of offender details, and generation of challans instantly. For traffic officers, the app offers an efficient tool to issue challans on the spot, ensuring that all necessary information is accurately recorded and validated before submission. For violators, the app ensures transparency and convenience by providing instant notifications and clear details of the issued challans. This digital approach not only streamlines the enforcement process but also enhances the overall experience for all users involved, contributing to a more organized and effective traffic management system.

2

# INTRODUCTION

In the rapidly advancing digital age, traditional methods of managing traffic violations are becoming increasingly inefficient and outdated. The manual process of issuing and recording traffic challans, often characterized by extensive paperwork and prone to human errors, poses significant challenges to law enforcement agencies. These inefficiencies not only slow down the enforcement process but also lead to inaccuracies and difficulties in maintaining comprehensive violation records. To address these challenges, there is a pressing need for a more streamlined, accurate, and efficient system for managing traffic violations.

The Traffic Challan App emerges as a cutting-edge solution designed to modernize the issuance and management of traffic violation tickets. This Android-based application leverages the capabilities of mobile technology to offer a digital platform that simplifies the process for both traffic officers and violators. By enabling on-the-spot issuance of challans and real-time recording of violation details, the app eliminates the cumbersome paperwork and reduces the likelihood of errors. This innovative approach not only enhances the operational efficiency of traffic law enforcement agencies but also ensures a transparent and user-friendly experience for violators. The introduction of the Traffic Challan App marks a significant step towards embracing technology for better traffic management and law enforcement.

3

### **OBJECTIVES**

1. **Automation**: To automate the process of issuing traffic challans to improve efficiency and accuracy.
2. **Data Management**: To provide a centralized system for storing and managing traffic violation data.
3. **User-Friendly Interface**: To offer a simple and intuitive interface for both traffic officers and violators.
4. **Real-Time Processing**: To enable real-time issuance and processing of challans.
5. **Reduction of Paperwork**: To minimize the reliance on paper-based records, promoting a more environmentally friendly approach.

### **Features**

1. **Challan Issuance**: Traffic officers can issue challans by entering vehicle number, driver name, violation type, and fine amount.
2. **Data Validation**: The app validates the entered data to ensure completeness and correctness before submission.
3. **Instant Notifications**: Offenders receive instant notifications of issued challans via SMS or email.
4. **Database Integration**: All challan records are stored in a centralized database for easy retrieval and analysis.
5. **User Authentication**: Secure login for traffic officers to prevent unauthorized access.
6. **Reporting and Analytics**: Generation of reports and analytics to help authorities understand traffic violation patterns and trends.

4

### **Software Used and App Architecture**

**Software Used:**

* **Java:** The primary programming language used for developing the Traffic Challan App. Java is utilized for implementing the app's core functionality, including user interface interactions, data handling, and processing logic.
* **XML:** Employed for designing the app's user interface layouts and defining UI components. XML is used to create and organize the app's visual elements, such as activity layouts and resource files.

**App Architecture:**

The Traffic Challan App follows a basic Model-View-Controller (MVC) architecture:

1. **Model:** This layer manages the data and business logic of the app. It includes classes and methods responsible for handling traffic violation records, challan details, and user information.
2. **View:** The XML files in the res/layout directory define the user interface components of the app. These layout files describe how the UI elements are presented to the user, including EditTexts, Buttons, and TextViews.
3. **Controller:** Java classes, such as MainActivity, act as the controller. They handle user interactions, process input data, and manage the communication between the model and view. The controller is responsible for implementing the app's functionality and ensuring that the user interface responds to user actions appropriately.

This architecture ensures a clean separation of concerns, making the app easier to maintain and extend.

5

**Code Executions**

**mainactivity.java:**

package com.example.challanapp;

import android.content.SharedPreferences;

import android.content.Context;

import android.os.Bundle;

import android.view.View;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.EditText;

import android.widget.ListView;

import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import java.util.ArrayList;

import java.util.List;

public class MainActivity extends AppCompatActivity {

private EditText vehicleNumberEditText;

private EditText ownerNameEditText;

private EditText fineAmountEditText;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

vehicleNumberEditText = findViewById(R.id.vehicleNumberEditText);

ownerNameEditText = findViewById(R.id.ownerNameEditText);

fineAmountEditText = findViewById(R.id.fineAmountEditText);

Button saveChallanButton = findViewById(R.id.saveChallanButton);

saveChallanButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

saveChallan();

}

});

loadChallans();

}

private void saveChallan() {

String vehicleNumber = vehicleNumberEditText.getText().toString();

String ownerName = ownerNameEditText.getText().toString();

String fineAmount = fineAmountEditText.getText().toString();

if (vehicleNumber.isEmpty() || ownerName.isEmpty() || fineAmount.isEmpty()) {

Toast.makeText(this, "All fields are required", Toast.LENGTH\_SHORT).show();

return;

}

SharedPreferences sharedPreferences = getSharedPreferences("ChallanPrefs", Context.MODE\_PRIVATE);

SharedPreferences.Editor editor = sharedPreferences.edit();

editor.putString("vehicleNumber", vehicleNumber);

editor.putString("ownerName", ownerName);

editor.putString("fineAmount", fineAmount);

editor.apply();

vehicleNumberEditText.setText("");

ownerNameEditText.setText("");

fineAmountEditText.setText("");

Toast.makeText(this, "Challan saved", Toast.LENGTH\_SHORT).show();

loadChallans();

}

private void loadChallans() {

SharedPreferences sharedPreferences = getSharedPreferences("ChallanPrefs", Context.MODE\_PRIVATE);

String vehicleNumber = sharedPreferences.getString("vehicleNumber", null);

String ownerName = sharedPreferences.getString("ownerName", null);

String fineAmount = sharedPreferences.getString("fineAmount", null);

List<String> challans = new ArrayList<>();

if (vehicleNumber != null && ownerName != null && fineAmount != null) {

challans.add("Vehicle: " + vehicleNumber + ", Owner: " + ownerName + ", Fine: " + fineAmount);

}

ArrayAdapter<String> adapter = new ArrayAdapter<>(this, android.R.layout.simple\_list\_item\_1, challans);

ListView listView = findViewById(R.id.challanListView);

listView.setAdapter(adapter);

    }

}

**Challan.java:**

package com.example.challanapp;

public class Challan {

private String vehicleNumber;

private String ownerName;

private String fineAmount;

public Challan(String vehicleNumber, String ownerName, String fineAmount) {

this.vehicleNumber = vehicleNumber;

this.ownerName = ownerName;

this.fineAmount = fineAmount;

} public String getVehicleNumber() {

return vehicleNumber;

}public String getOwnerName() {

return ownerName;

}public String getFineAmount() {

return fineAmount;

    }

}

**challanadapter.java:**

package com.example.challanapp;

import android.content.Context;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.BaseAdapter;

import android.widget.TextView;

import com.example.challanapp.Challan;

import java.util.ArrayList;

public class ChallanAdapter extends BaseAdapter {

private Context context;

private ArrayList<com.example.challanapp.Challan> challanList;

public ChallanAdapter(Context context, ArrayList<com.example.challanapp.Challan> challanList) {

this.context = context;

this.challanList = challanList;

}@Override

public int getCount() {

return challanList.size();

}@Override

public Object getItem(int position) {

return challanList;

}@Override

public long getItemId(int position) {

return position;

}@Override

public View getView(int position, View convertView, ViewGroup parent) {

if (convertView == null) {

convertView = LayoutInflater.from(context).inflate(R.layout.challan\_item, parent, false);

}

com.example.challanapp.Challan challan;

challan = (com.example.challanapp.Challan) getItem(position);

TextView vehicleNumberTextView = convertView.findViewById(R.id.vehicleNumberTextView);

TextView ownerNameTextView = convertView.findViewById(R.id.ownerNameTextView);

TextView fineAmountTextView = convertView.findViewById(R.id.fineAmountTextView);

vehicleNumberTextView.setText(challan.getVehicleNumber());

ownerNameTextView.setText(challan.getOwnerName());

fineAmountTextView.setText(challan.getFineAmount());

return convertView;

}

}

**Activitymain.xml:**

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:background="#FFFFFF"

android:padding="16dp"

tools:context=".MainActivity">

<ImageView

android:id="@+id/headerIcon"

android:layout\_width="110dp"

android:layout\_height="100dp"

android:layout\_gravity="center"

android:src="@drawable/header" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:text="GOVERNMENT OF AP"

android:textSize="25sp"

android:textColor="#0CA812" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:text="POLICE DEPARTMENT"

android:textSize="16sp"

android:textColor="#C81717" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="CHALLAN"

android:layout\_gravity="center"

android:textSize="25sp"

android:textColor="#073933"

android:textStyle="bold" />

<EditText

android:id="@+id/vehicleNumberEditText"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Vehicle Number" />

<EditText

android:id="@+id/ownerNameEditText"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Owner Name" />

<EditText

android:id="@+id/fineAmountEditText"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Fine Amount"

android:inputType="number" />

<Button

android:id="@+id/saveChallanButton"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_margin="25dp"

android:text="Save"

android:textColor="#123"

android:layout\_gravity="center\_horizontal" />

<ListView

android:id="@+id/challanListView"

android:layout\_width="match\_parent"

android:layout\_height="373dp"

android:background="#8CAED1" />

</LinearLayout>

**Challan\_item.xml:**

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

android:padding="16dp">

<TextView

android:id="@+id/vehicleNumberTextView"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Vehicle Number"

android:textSize="18sp"

android:textColor="#000000" />

<TextView

android:id="@+id/ownerNameTextView"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Owner Name"

android:textSize="18sp"

android:textColor="#000000" />

<TextView

android:id="@+id/fineAmountTextView"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

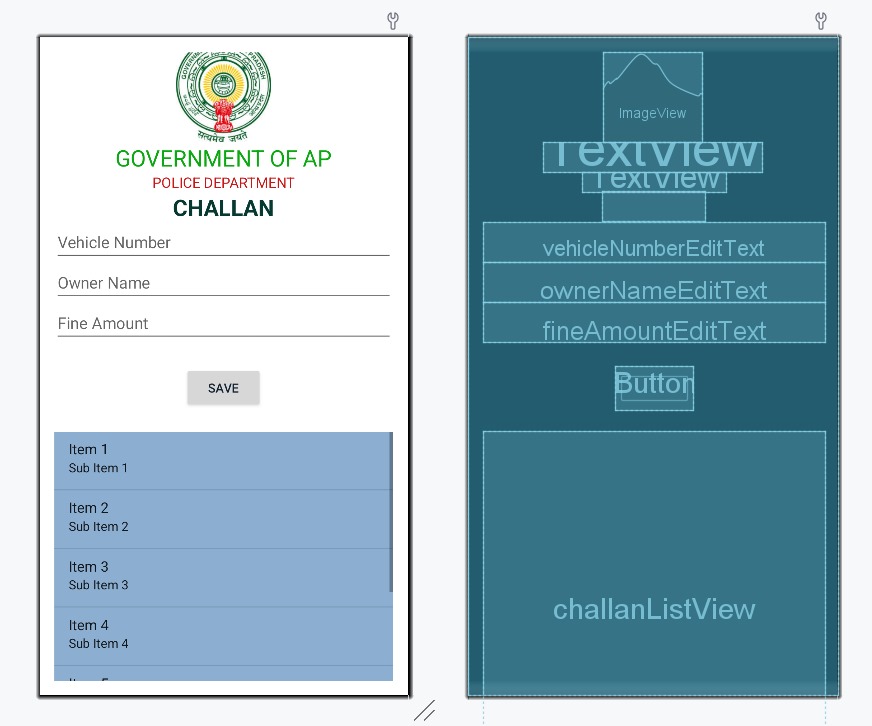
android:text="Fine Amount"

android:textSize="18sp"

android:textColor="#000000" />

</LinearLayout>

**Output**

****

**Conclusion**

The Traffic Challan App represents a significant step towards modernizing traffic law enforcement. By automating the issuance and management of traffic tickets, the app not only improves operational efficiency but also ensures greater accuracy and transparency. This digital approach aligns with the broader objectives of smart city initiatives, contributing to safer and more organized urban traffic management. Future enhancements may include integration with payment gateways for online fine payments.