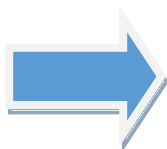


Министерство образования Пензенской области
Муниципальное бюджетное общеобразовательное учреждение
средняя общеобразовательная школа им. М.Ю.Лермонтова
с. Засечное Пензенского района

Научно-практическая конференция «Старт в науку»

секция «Информатика»

Клиент-серверное приложение «Мобильный репортер».



Выполнил:
Колягин Владислав
ученик 11 класса
МБОУ СОШ им.М.Ю.Лермонтова
с.Засечное
Пензенского района
Научный руководитель:
Емельянов Игорь Борисович
учитель информатики

2020 год

СОДЕРЖАНИЕ

Введение.....	3
Создание прототипа программного комплекса.....	4
Интегрирование комплекта на реальный сайт.....	7
Заключение.....	9
Список использованной литературы.....	10
Приложения.....	11

1. Введение

Актуальность работы: Практически каждая организация, а в последнее время и персоналии, имеют свой сайт в сети интернет. Некоторых, например, образовательные или любые другие муниципальные и государственные учреждения, в этом обязывает закон. Другие используют свой сайт для рекламы и продвижения товаров или услуг. Интернет сайты можно создать на базе «движка», можно заказать стороннему разработчику, наиболее «продвинутые» пользователи создают сайты самостоятельно. Эффективность сайта зависит, прежде всего, от своевременного обновления контента. Администрирование сайта, в большинстве случаев, происходит на стационарном компьютере или ноутбуке. Другими словами, для добавления новости о каком-либо мероприятии на сайт необходимо сделать фотографии, «скинуть» их на компьютер, добавить текст и лишь потом опубликовать новость в интернете.

Альтернативный вариант изменения контента имеют приложения социальных сетей. Добавление фото и текста происходит непосредственно с мобильного устройства.

Попытка использовать админ-панель сайта при помощи мобильного устройства принесла следующий результат:

- текст, добавленный в HTML – форму успешно добавлялся на сайт
- попытка загрузить фотографию завершалась на стадии выбора файла.

Появилась идея разработать систему добавления новостного контента на любой сайт с использованием мобильного устройства под управлением операционной системы «Андроид».

Цель исследования:

1. Изучить вопрос загрузки файлов с мобильного устройства, работающего под управлением операционной системы «Андроид», на веб-сервер.
2. Исследовать возможность применения полученного программного продукта для администрирования новостного контента реальных сайтов.

Задачи:

1. Разработать прототип мобильного приложения, предназначенного для добавления текстового и графического контента на веб-сервер.
2. Разработать серверную часть, установить ее на реальный веб-сервер, протестировать полученный комплект.
3. Интегрировать код в действующий сайт, протестировать приложение в условиях практической работы.

Предмет исследования: Мобильное приложение, веб-сервер.

Методы исследования: сравнение, эксперимент, тестирование программных продуктов.

Практическая значимость: полученное в результате работы мобильное приложение для ОС «Андроид» должно облегчить работу администраторов сайтов по своевременному обновлению новостного контента.

2. Создание прототипа программного комплекса

2.1. Постановка задачи

В ходе работы предполагалось создать комплект:

- прототип мобильного приложения;
- серверную часть;

Задача прототипа приложения заключалась в захвате изображения с камеры мобильного устройства, добавления в запрос файла изображения и текстовой информации, отправка запроса на веб-сервер

2.2. Сервер

Задача кода сервера, который был написан на PHP, заключалась в приеме POST запроса, сохранения файла изображения на сервере, сохранение принятого текста в отдельном файле. Использование локального сервера для связки «Андроид»-веб-сервер довольно проблематично, поэтому код PHP писался в «блокноте» и отдавался для размещения администратору сайта <http://leninoschool.edu-penza.ru>.

2.3 Разработка прототипа приложения

2.3.1. Среда разработки

Разработка велась в интегрированной среде разработки Android Studio. В процессе создания программного кода использовались стандартные библиотеки **java** и **android**. Тестирование происходило на всех эмуляторах, доступных в этой среде разработки. После выпуска релиза приложение проверено на реальных устройствах Андроид 4.1, Андроид 7.1, Андроид 8.0, Андроид 10

2.3.2. Установка разрешений

Для работы приложения нужны следующие разрешения:

- Доступ к работе с камерой;
- Доступ к записи файлов на устройство;
- Доступ в интернет;
- Разрешение на работу с локальной сетью.

2.3.3. Интерфейс приложения

Интерфейс приложения содержит 2 активности. Первая активность представляет собой интерфейс подготовки фотографии. Здесь находится кнопка, позволяющая сделать фотографию. Также на этом экране можно увидеть превью получившегося фото.

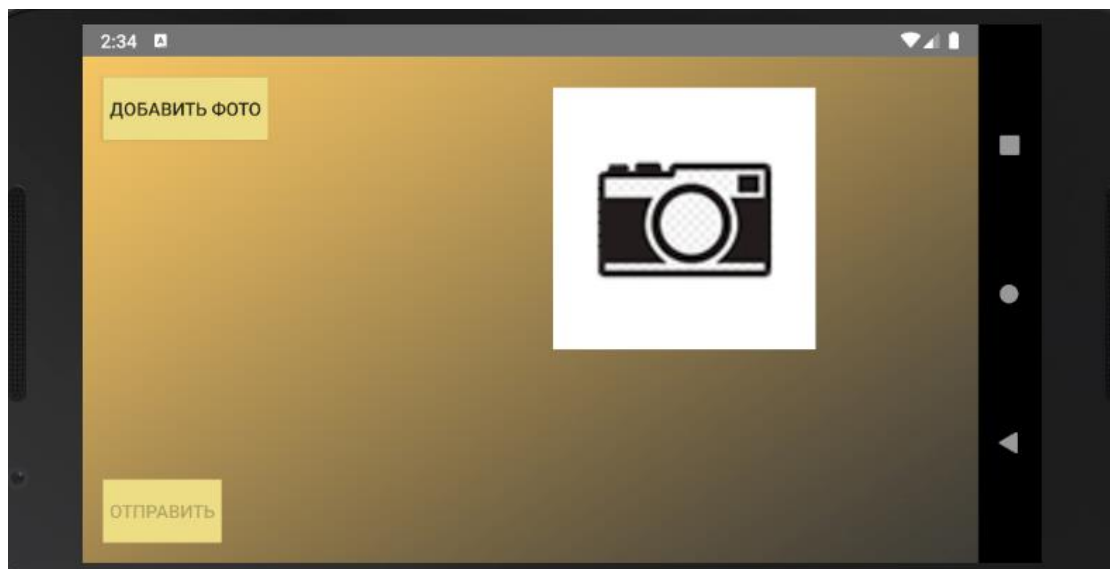


Рисунок 1. Активность добавления фотографии

Вторая активность позволяет вам добавить текст к изображению и отправить результат на сайт.

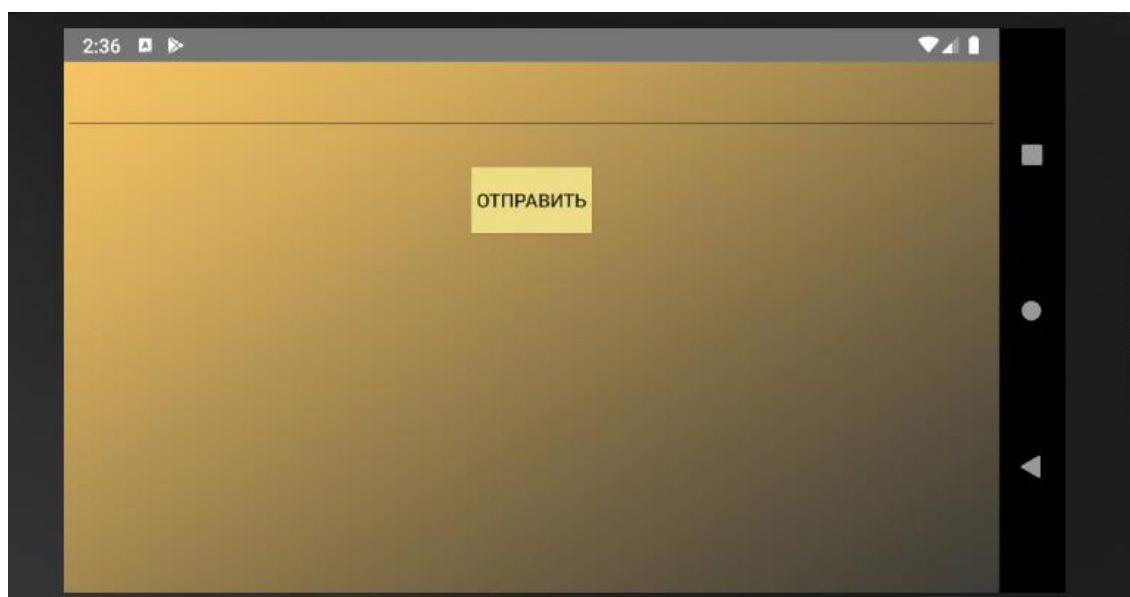


Рисунок 2. Активность добавления текста и отправки

2.4 Обработка возможных ограничений

В процессе изучения дополнительного материала по данной теме выяснилось, что процесс загрузки файлов на веб-сервер может требовать дополнительных ограничений.

2.4.1 Размер загружаемых файлов

Стандартные настройки PHP предполагают ограничение по размерам загружаемого файла. Попытка загрузки файла более 1 МБ возвращает 413 ошибку. Камеры современных мобильных устройств сохраняют фото в файлы размером намного больше, чем 1 Мб. Но в то же самое время размер картинок в новостной ленте сайта редко превышает 1000x800 px, что соответствует размеру **jpg** файла не более 100 Кб.

Было принято решение «сжимать» полученные с камеры фотографии перед отправкой на сервер.

Для этого использовались классы **Bitmap** и **BitmapFactory**.

2.4.2 Попытка повторной загрузки

Время, в течении которого файл будет загружаться на веб-сервер, зависит от скорости мобильного интернета и может быть значительным. С целью устранения каких-либо действий со стороны пользователя во время процесса отправки картинки, было принято решение переводить все активные виджеты приложения в «пассивный» режим до полной загрузки файла на сервер.

2.5 Тестирование прототипа

Для тестирования прототипа серверный код был загружен в папку:

<ftp://leninoschool.edu-penza.ru/prototip>

Серверная часть комплекта предполагала получение **multipart POST** – запроса, который содержит файл и текст. Скрипт создает папку с уникальным именем, копирует в нее файл изображения и создает файл, содержащий текст новостного контента.

Для удобства тестирования был дополнительно написан скрипт, который сортирует добавленные новости и выводит их в окно браузера.

Внешний вид новостной ленты прототипа изображен на рисунке 3. Страница отображает все отсортированные «превью» картинок, по «клику» на которые отображается картинка в полном размере.

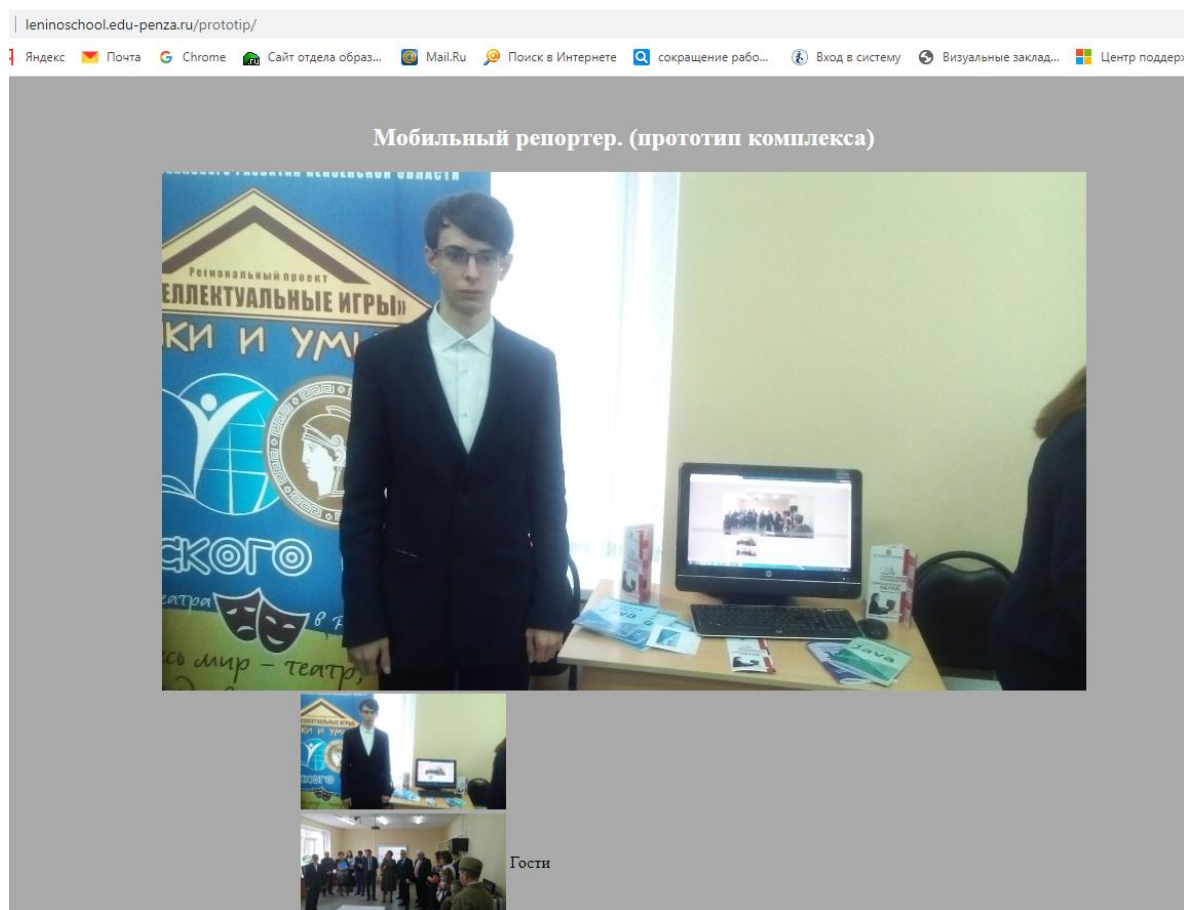


Рисунок 3. Внешний вид страницы прототипа.

3. Интегрирование комплекта на реальный сайт

Для удачного интегрирования серверной части в коды реальных сайтов необходимо знать, каким образом там хранится новостной контент.

У моего научного руководителя имеется FTP доступ к реальным сайтам некоторых образовательных организаций. По моей просьбе, с целью тестирования комплекта в реальных условиях работы, на соответствующие серверы был загружен код. Код мобильного приложения был адаптирован для формирования соответствующих запросов.

Так как серверный код формировал контент новостей реальных сайтов, то для отслеживания факта загрузки новости с мобильного приложения к тексту новости добавляется специальный логотип.

В настоящее время приложение используется для добавления новостного контента на сайте МБОУСОШ им.М.Ю. Лермонтова (<http://schoolsputnik.ru>) Внешний вид новостной страницы изображен на рисунке 4.

Новостной контент, добавленный с помощью мобильного приложения, отличается от остальных наличием специального значка.

Муниципальное бюджетное общеобразовательное учреждение средняя общеобразовательная школа им М.Ю.Лермонтова с. Засечное Пензенского района Пензенской области

Официальный сайт 17.02.2020 18:42:26

Версия для слабовидящих Фотогалерея Видеоролики

На главную
Объявления
Карта
Сведения об ОО
Информационная безопасность

Наши новости

17.02.20
«Горжусь тобой, Отечество»
С 12 по 19 февраля 2020 года в школе продолжается ГТО у мальчиков.
Сейчас активно начался прием у учащихся ...[Читать...](#)
Просмотров: 23 Комментариев:

17.02.20
Школьный этап международного конкурса "Живая классика" ...[Читать...](#)

Рисунок 4. Новостная лента сайта МБОУ СОШ им. М.Ю. Лермонтова.

И сайте отдела образования администрации Пензенского района.
(<http://roopenza.ru>)

Отдел образования администрации Пензенского района

ФЕДЕРАЛЬНЫЙ ЦЕНТР ИНФОРМАЦИОННО-ОБРАЗОВАТЕЛЬНЫХ РЕСУРСОВ fci.or.edu.ru
8-800-2000-122 ТЕЛЕФОН ДОВЕРИЯ ДЛЯ ПОДРОСТКОВ
ГИА

| Новости | Фотогалерея | Видео | Школы | МБДОУ | МБОУДОУ |

Анонс новостей :

17.02.20
ВТОРОЕ МЕСТО В ЧЕМПИОНАТЕ КЭС-БАСКЕТ
14 февраля 2020 ...

17.02.20
50 ЛЕТ ДЕТСКОМУ САДУ п. Мичуринский
15 февраля ...

14.02.20
Об итогах районного конкурса художественного творчества ...

14.02.20
Празднование Дня зимних видов спорта, которое приурочено ...

Информация :

Instagram
на главную

Наши новости

10.12.19
10 декабря на базе детского сада "Лукоморье" проходил областной семинар дошкольных работников "Модели деятельности консультативного центра на базе ДООУ". В работе семинара ...
Просмотров: Комментариев:

Рисунок 5. Новостная лента сайта отдела образования

Еще несколько образовательных организаций заинтересовались приложением. В принципе, интеграция возможна в любой сайт, необходим FTP доступ к серверу для загрузки скрипта. Для написания серверной части комплекта разработчик должен знать структуру хранения новостного контента на сайте.

4. Заключение

В результате выполнения работы я изучил вопрос загрузки файлов с мобильного устройства, работающего под управлением операционной системы Андроид, на веб-сервер. В процессе написания кода программных продуктов я совершенствовал свои знания в области языков программирования Java и PHP. Создан реальный программный продукт, который позволяет загружать изображения и текст на веб-сервер. Данный продукт интегрирован в реально существующие сайты, и, по отзывам администраторов этих сайтов, показал себя с положительной стороны.

Список использованной литературы

1. Емельянов И.Б. Компьютерные науки: изучение языка программирования Java. Часть 3: учебно-методическое пособие / Емельянов И.Б., Емельянов Р.И. – Пенза: Типография ИП Поповой М.Г., 2019. – 80 с.
2. «Сайт Александра Климова» [Электронный ресурс]. URL: <http://developer.alexanderklimov.ru/android> (дата обращения: 14.12.2019).

Приложения 1. Класс MainActivity

```
package ru.pwp.pro100fantom.photomaster;

import android.Manifest;
import android.app.Activity;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.os.StrictMode;
import android.provider.MediaStore;
import android.support.v4.app.ActivityCompat;
import android.support.v4.content.ContextCompat;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import java.io.File;

public class MainActivity extends Activity {

    private static int TAKE_PICTURE_REQUEST = 1;
    private ImageView imageView;
    private Uri outputFileUri;
    Button out;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);
        out = (Button) findViewById(R.id.button2);
        imageView = (ImageView) findViewById(R.id.imageView);
        out.setEnabled(false);
    }

    public void onClick(View view) {
        //getThumbnailPicture();
        saveFullImage();
    }
}
```

```

}

public void up(View view){
    Intent in=new Intent(this,UploadTask.class);
    startActivity(in);
}
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    if (requestCode == TAKE_PICTURE_REQUEST && resultCode == RESULT_OK) {
        if (data != null) {
            if (data.hasExtra("data")) {
                Bitmap thumbnailBitmap = data.getParcelableExtra("data");
                imageView.setImageBitmap(thumbnailBitmap);
            }
            } else {
                imageView.setImageURI(outputFileUri);
                out.setEnabled(true);
            }
        }
    }

}

private void getThumbnailPicture() {
    Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
    startActivityForResult(intent, TAKE_PICTURE_REQUEST);
}

private void saveFullImage() {
    int permissionStatus = ContextCompat.checkSelfPermission(this, Manifest.permission.CAMERA);
    int permissionStatus2 = ContextCompat.checkSelfPermission(this,
Manifest.permission.WRITE_EXTERNAL_STORAGE);
    int permissionStatus3 = ContextCompat.checkSelfPermission(this, Manifest.permission.INTERNET);
    int permissionStatus4 = ContextCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_NETWORK_STATE);
    if (permissionStatus == PackageManager.PERMISSION_GRANTED&& permissionStatus2==
PackageManager.PERMISSION_GRANTED&&permissionStatus3 == PackageManager.PERMISSION_GRANTED&&
permissionStatus4== PackageManager.PERMISSION_GRANTED) {
        StrictMode.VmPolicy.Builder builder = new StrictMode.VmPolicy.Builder();
        StrictMode.setVmPolicy(builder.build());
        Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
        File file = new File(Environment.getExternalStorageDirectory(), "test.jpg");
        outputFileUri = Uri.fromFile(file);
        intent.putExtra(MediaStore.EXTRA_OUTPUT, outputFileUri);
        startActivityForResult(intent, TAKE_PICTURE_REQUEST);
    } else {
        ActivityCompat.requestPermissions(this, new String[]
{Manifest.permission.CAMERA,Manifest.permission.WRITE_EXTERNAL_STORAGE,Manifest.permission.INTERNET,M
anifest.permission.ACCESS_NETWORK_STATE},2);
    }
}
}
}

```

Приложение 2. Класс UploadTask

```

package ru.pwp.pro100fantom.photomaster;

import java.io.DataOutputStream;
import java.io.File;
import java.io.FileInputStream;
import java.net.HttpURLConnection;
import java.net.MalformedURLException;
import java.net.URL;
import java.net.URLEncoder;
import android.app.Activity;
import android.app.ProgressDialog;
import android.content.Intent;
import android.database.Cursor;
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.provider.MediaStore;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;

public class UploadTask extends Activity{
    private ProgressDialog dialog;
    private int serverResponseCode = 0;
    private EditText Text;
    private String upLoadServerUri = null;
    private String imagepath=null;
    String text="";
    Button out;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_upload_task);
        Text=(EditText)findViewById(R.id.editText1);
        out = (Button) findViewById(R.id.button11);
        out.setEnabled(true);
    }
    public void up( View view){

        String file=Environment.getExternalStorageDirectory().getAbsolutePath()+"/test.jpg";
        imagepath=file;
        text=Text.getText().toString();
        out.setEnabled(false);
        new Thread(new Runnable() {
            public void run() {

                uploadFile(imagepath);

            }
        }).start();
    }
}

```

}

```

protected void onActivityResult(int requestCode, int resultCode, Intent data) {

    if (requestCode == 1 && resultCode == RESULT_OK) {
        Uri selectedImageUri = data.getData();
        imagePath = getPath(selectedImageUri);
    }
}

public String getPath(Uri uri) {
    String[] projection = { MediaStore.Images.Media.DATA };
    Cursor cursor = managedQuery(uri, projection, null, null, null);
    int column_index = cursor.getColumnIndexOrThrow(MediaStore.Images.Media.DATA);
    cursor.moveToFirst();
    return cursor.getString(column_index);
}

public int uploadFile(String sourceFileUri) {
    String fileName = sourceFileUri;
    HttpURLConnection conn = null;
    DataOutputStream dos = null;
    String lineEnd = "\r\n";
    String twoHyphens = "--";
    String boundary = "*****";
    int bytesRead, bytesAvailable, bufferSize;
    byte[] buffer;
    int maxBufferSize = 1 * 1024 * 1024;
    File sourceFile = new File(sourceFileUri);

    if (!sourceFile.exists()) {
        return 0;
    }
    else
    {
        try {
            FileInputStream fileInputStream = new FileInputStream(sourceFile);
            URL url = new URL("http://leninoschool.edu-
penza.ru/prototip/upload.php?text="+URLEncoder.encode(text, "UTF-8"));
            conn = (HttpURLConnection) url.openConnection();
            conn.setDoInput(true);
            conn.setDoOutput(true);
            conn.setUseCaches(false);
            conn.setRequestMethod("POST");
            conn.setRequestProperty("Connection", "Keep-Alive");
            conn.setRequestProperty("ENCTYPE", "multipart/form-data");
            conn.setRequestProperty("Content-Type", "multipart/form-data;boundary=" + boundary);
            conn.setRequestProperty("file", fileName);

            dos = new DataOutputStream(conn.getOutputStream());

```

```

dos.writeBytes(twoHyphens + boundary + lineEnd);
dos.writeBytes("Content-Disposition: form-data; name=\"file\";filename=\""
    + fileName + "\" + lineEnd);

dos.writeBytes(lineEnd);
bytesAvailable = fileInputStream.available();
bufferSize = Math.min(bytesAvailable, maxBufferSize);
buffer = new byte[bufferSize];
bytesRead = fileInputStream.read(buffer, 0, bufferSize);

while (bytesRead > 0) {

    dos.write(buffer, 0, bufferSize);
    bytesAvailable = fileInputStream.available();
    bufferSize = Math.min(bytesAvailable, maxBufferSize);
    bytesRead = fileInputStream.read(buffer, 0, bufferSize);

}

dos.writeBytes(lineEnd);
dos.writeBytes(twoHyphens + boundary + twoHyphens + lineEnd);
serverResponseCode = conn.getResponseCode();
String serverResponseMessage = conn.getResponseMessage();

Log.i("uploadFile", "HTTP Response is : "
    + serverResponseMessage + ": " + serverResponseCode);

if(serverResponseCode == 200){
    Intent r=new Intent(this,MainActivity.class);
    startActivity(r);
}
fileInputStream.close();
dos.flush();
dos.close();

} catch (MalformedURLException ex) {
    dialog.dismiss();
    ex.printStackTrace();
    Log.e("Upload file to server", "error: " + ex.getMessage(), ex);
} catch (final Exception e) {
    e.printStackTrace();
}
}
return serverResponseCode;

}
}
}

```