Лабораторная работа №3

СОЗДАНИЕ И ОТПРАВКА MMS

**Цель работы**: получить навыки разработки программ для редактирования и отправки MMS для мобильных устройств Android.

**ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

Для отправки MMS на Android можно использовать Намерение (Intent) и обращаться к стандартному клиенту, так как API для работы с MMS не предусмотрен, в отличие от SMS. Для начала, добавим необходимые разрешения в файл манифеста:

<uses-permission android:name="android.permission.RECEIVE\_MMS" />

<uses-permission android:name="android.permission.SEND\_SMS" />

<uses-permission android:name="android.permission.RECEIVE\_SMS" />

В интерфейсе программы необходимо предусмотреть возможность прикрепления графического файла. Для этого можно, например, добавить кнопку со следующим обработчиком:

imagePick.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent photoPickerIntent = new Intent(Intent.ACTION\_PICK);

photoPickerIntent.setType("image/\*");

startActivityForResult(photoPickerIntent, SELECT\_PHOTO);

}

});

Здесь мы фактически делаем запрос на выбор изображения из галереи телефона. Чтобы отобразить выбранный графический файл в приложении, можно использовать элемент ImageView. При этом необходимо будет переопределить метод onActivityResult(). В этом методе мы также получим URI файла, необходимый нам в дальнейшем для отправки MMS.

@Override

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

super.onActivityResult(requestCode, resultCode, data);

if (requestCode == SELECT\_PHOTO && resultCode == RESULT\_OK && data != null) {

// получим Uri изображения

Uri pickedImage = data.getData();

// получим путь к изображению

String[] filePath = { MediaStore.Images.Media.DATA };

Cursor cursor = getContentResolver().query(pickedImage,

filePath, null, null, null);

cursor.moveToFirst();

String imagePath = cursor.getString(cursor.getColumnIndex(filePath[0]));

BitmapFactory.Options options = new BitmapFactory.Options();

options.inPreferredConfig = Bitmap.Config.ARGB\_8888;

Bitmap bitmap = BitmapFactory.decodeFile(imagePath, options);

imageView.setImageBitmap(bitmap);

cursor.close();

}

}

Теперь пользователь имеет возможность выбрать файл из галереи телефона и просмотреть его в приложении. Создадим намерение, сообщающее системе Android, что наше приложение хочет отправить некоторые данные. Намерение ACTION\_SEND может быть использовано для отправки различных типов сообщений, таких как текстовые сообщения (SMS) и сообщений с мультимедийными вложениями (MMS), сообщений электронной почты и т.д.

Intent picMessageIntent = new Intent(android.content.Intent.ACTION\_SEND);

Далее необходимо установить тип отправляемого вложения в зависимости от выбранного пользователем файла. Например, следующие примеры кода предназначены для отправки jpeg-, png-изображений или видеофайла:

picMessageIntent.setType("image/jpeg");

picMessageIntent.setType("image/png");

picMessageIntent.setType("video/\*");

Прикрепим к созданному намерению телефонный номер получателя, текст сообщения и URI прикрепленного файла:

picMessageIntent.putExtra("address","38095…");

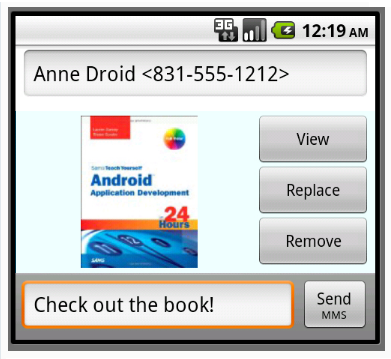
picMessageIntent.putExtra("sms\_body","hello…");

picMessageIntent.putExtra(Intent.EXTRA\_STREAM, pickedImage);

Выполним намерение picMessageIntent, вызвав метод startActivity():

startActivity(picMessageIntent);

Сообщение с прикрепленным файлом будет отправлено на указанный номер. При запуске программы пользователю, вероятно, будет предложена возможность выбрать одно из нескольких установленных приложений, способных выполнять запросы подобного рода. В процессе работы приложение может выглядеть следующим образом:



**Вопросы для самопроверки**

1. Какие разрешения могут потребоваться при работе с MMS?
2. Каким образом можно выбрать и просмотреть изображение из галереи телефона? Какие классы и методы для этого используются?
3. Какой компонент интерфейса, предназначенный для представления изображений, вам известен? Какие методы вам известны для работы с ним?
4. Какое намерение используется для отправки данных (СМС, ММС, электронных писем)?
5. Какие типы данных можно задавать в методе setType()?
6. Для чего предназначен метод putExtra()? Какие параметры он может принимать?
7. Для чего служит метод startActivity()?

**Дополнительные материалы**

1. Intent (Намерение) ACTION\_SEND – [Электронный ресурс] – <http://developer.alexanderklimov.ru/android/theory/intent.php#action_send> (дата обращения – 20.02.2017)
2. Майер Р. «Программирование приложений для планшетных компьютеров и смартфонов». – М.: Эксмо, 2011. – 672 с.
3. Магические Intents – как ими пользоваться – [Электронный ресурс] – <http://android-helper.com.ua/intents/> (дата обращения – 20.02.2017)
4. Android Pick/Select Image from Gallery with Intents – [Электронный ресурс] – <http://codetheory.in/android-pick-select-image-from-gallery-with-intents/> (дата обращения – 20.02.2017)

**Задание**

Дополнить лабораторную работу №2 возможностью создания MMS. Для создания MMS использовать список изображений, сформированный на основе данных из ресурсов, а также графические файлы, из которых пользователь может выбрать нужный. Получатель MMS указывается таким же образом, как и в лабораторной работе №2.

**Варианты заданий**

1. Реализовать возможность поиска ММС по тексту сообщения.
2. Реализовать возможность сортировки ММС по имени контактов.
3. Пользователь должен иметь возможность отправить сообщение нескольким (до 3-х) контактам.
4. Пользователь должен иметь возможность прикреплять к сообщению несколько (до 3-х) файлов.
5. Реализовать возможность поиска ММС по имени контакта.
6. Позволить пользователю устанавливать максимальный размер сообщения (например, в килобайтах).
7. Визуально выделять доставленные и недоставленные ММС. Сообщать пользователю об ошибке отправки сообщения.
8. При выходе из приложения сохранять неотправленные (но не пустые) ММС в виде черновиков.
9. Позволить пользователю ограничивать количество хранимых ММС до заданного значения. Автоматически удалять более старые сообщения.
10. Пользователь должен иметь возможность добавить тему сообщению.