

## Of Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

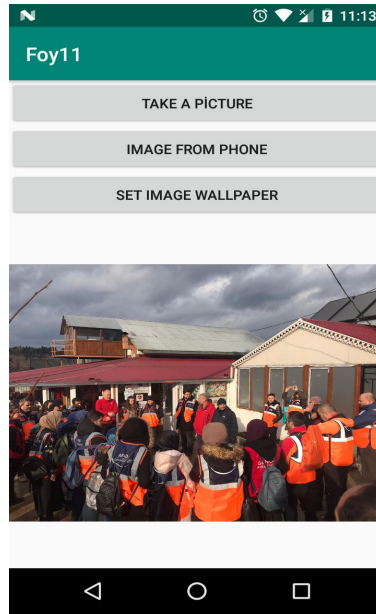
### Mobil Uygulama Geliştirme Dersi Deney Föyü

#### Galeri ve Kamerayı Kullanarak Resim Ekleme Uygulaması

Bu föy, Android Applicationda galeriden veya kamerayı kullanarak resim eklemeyi anlatan bilgilerden oluşmaktadır. Öncelikle aşağıdaki gibi bir ekran tasarımı (layout) gerçekleştirilir. Bu layoutta galeri ve kameraya erişebilecek 2 adet button, seçilen duvar kağıdı yapan bir adet buton ve buradan dönecek olan image için bir adette imageView kullanılmaktadır.

Kamerayı kullanabilmek ve external hafızaya erişebilmek için aşağıdaki permissionları yani izinleri manifest dosyasına eklemem gerekiyor;

```
<uses-permission android:name="android.permission.SET_WALLPAPER"/>
<uses-permission android:name="android.permission.SET_WALLPAPER_HINTS"/>
<uses-permission android:name="android.permission.CAMERA" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-feature android:name="android.hardware.camera" />
<uses-feature android:name="android.hardware.camera.autofocus" />
```



Layout kodu aşağıdaki gibi oluşturulmalıdır:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
    android:weightSum="1" >
```

```

<LinearLayout
    android:id="@+id/linearLayout1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical" >
</LinearLayout>

<LinearLayout
    android:id="@+id/linearLayout2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical" >

    <Button
        android:id="@+id/button_camera"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Take a Picture" >
    </Button>
</LinearLayout>

<Button
    android:id="@+id/button_from_phone"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Image from Phone" >
</Button>
<Button
    android:id="@+id/set_wallpaper"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Set Image wallpaper" >
</Button>
<ImageView
    android:id="@+id/imageView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:src="@drawable/ic_launcher_background" >
</ImageView>

</LinearLayout>

```

Activity içerisinde ilk olarak 2 adet değişken tanımlanır, bunlar onActivityResult içinde kullanılacak kamera veya galeri değişkenleri olacaktır.

```

private static final int IMAGE_PICK = 1;
private static final int IMAGE_CAPTURE = 2;

```

Şimdi ise layout içerisinde tanımlanan buttonların click eventleri yazılır. Galerı butonu için bir adet ImagePickListener adında bir OnClickListener tanımlanır. Bunun içerisinde galeriden resim seçebilmek için bir intent başlatılır. Daha sonra kod aşağıdaki gibi şekillenir;

```

class ImagePickListener implements View.OnClickListener {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(Intent.ACTION_PICK,
        android.provider.MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
        intent.setType("image/*");
        startActivityForResult(Intent.createChooser(intent, "Bir Fotoğraf
        Seçin"), IMAGE_PICK);
    }
}

```

Kamera butonu için ise kamerayı başlatan bir intent çalıştırılır. Aynı şekilde kamera activity'sini çağıran intent oluşturulup, kamera action kodu birlikte başlatılır.

```

class TakePictureListener implements View.OnClickListener {
    @Override
    public void onClick(View v) {
        Intent cameraIntent = new
        Intent(android.provider.MediaStore.ACTION_IMAGE_CAPTURE);
        startActivityForResult(cameraIntent, IMAGE_CAPTURE);
    }
}

```

Click eventler de tanımladıktan sonra sıra başlatılan activitylerin dönüşünde seçilen veya kaydedilen resmi mevcut activity içerisine çekmeye gelir. Bunun içinde onActivityResult methodu override edilmelidir. Bunun içerisinde de galeri veya kamera dan seçme durumuna göre resmi alıp imageView'a basılması gerekir. Bunlar için 2 method tanımlanır. Birisi galeriden seçilen resmi alıp imageView'a; diğeri de kameradan çekilen resmi alıp imageView'a gönderir.

```

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

    if (resultCode == Activity.RESULT_OK) {
        switch (requestCode) {
            case IMAGE_PICK:
                this.imageFromGallery(resultCode, data);
                break;
            case IMAGE_CAPTURE:
                this.imageFromCamera(resultCode, data);
                break;
            default:
                break;
        }
    }
}

```

Galeriden seçilen resmi imageView içerisine basmak için gönderilen intent datası ile resimlerin bulunduğu yol çekilir. Daha sonra cursor ile seçilen resmin yolu buradan elde edilir. En son olarak da alınan yol imageView'a parametre olarak verilir.

```
private void imageFromGallery(int resultCode, Intent data) {
    Uri selectedImage = data.getData();
    String [] filePathColumn = {MediaStore.Images.Media.DATA};

    Cursor cursor = getContentResolver().query(selectedImage,
filePathColumn, null, null, null);
    cursor.moveToFirst();

    int columnIndex = cursor.getColumnIndex(filePathColumn[0]);
    String filePath = cursor.getString(columnIndex);
    this.imageView.setImageBitmap(BitmapFactory.decodeFile(filePath));
    cursor.close();
}
```

Kameradan seçilen resmi imageView'a basmak biraz daha karmaşıktır :).

```
private void imageFromCamera(int resultCode, Intent data) {
    Bitmap photo = (Bitmap) data.getExtras().get("data");
    imageView.setImageBitmap(photo);
}
```

Setleme işlemi: btnset butonu için tanımlama yapıldıktan sonra dinleme işlemi için onclick sınıfı tanımlaması yapılır.

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    this.imageView = (ImageView) this.findViewById(R.id.imageView);
    this.buttonNewPic = (Button) this.findViewById(R.id.button_camera);
    this.buttonImage = (Button) this.findViewById(R.id.button_from_phone);

    this.buttonImage.setOnClickListener(new ImagePickListener());
    this.buttonNewPic.setOnClickListener(new TakePictureListener());
    btnSet = (Button) findViewById(R.id.set_wallpaper);
    btnSet.setOnClickListener(new OnClickListener() {
        @Override
        public void onClick(View v) {
            setWallpaper();
        }
    });
}
```

Setwallpaper fonksiyonu ile imageView den çekilen resmin boyut ayarlaması yapıldıktan sonra duvar kağıdı işlemi gerçekleştirilir.

```

private void setWallpaper() {

    wallpaperManager =
WallpaperManager.getInstance(getApplicationContext());

    bitmapDrawable = (BitmapDrawable) imageView.getDrawable();
    bitmap1 = bitmapDrawable.getBitmap();
    GetScreenWidthHeight();
    SetBitmapSize();
    wallpaperManager = WallpaperManager.getInstance(MainActivity.this);
    try {
        wallpaperManager.setBitmap(bitmap2);
        wallpaperManager.suggestDesiredDimensions(width, height);
    } catch (IOException e) {
        e.printStackTrace();
    }
}

public void GetScreenWidthHeight(){ // telefonun boyutuna göre genişlik ve
//uzunluk alınır.

    displayMetrics = new DisplayMetrics();

    getWindowManager().getDefaultDisplay().getMetrics(displayMetrics);

    width = displayMetrics.widthPixels;

    height = displayMetrics.heightPixels;

}

```

Bu işlem ise bitmap2 sınıfı için yeniden boyut ve image ataması yapar. Daha sonra bu değişken duvar kağıdı atamasında nesne olarak işlem görür.

```

public void SetBitmapSize(){

    bitmap2 = Bitmap.createScaledBitmap(bitmap1, width, height, false);

}

```

Uygulamada tanımlanan değişkenlerin tamamı aşağıdadır.

```

public class MainActivity extends AppCompatActivity {
    private static final int IMAGE_PICK = 1;
    private static final int IMAGE_CAPTURE = 2;

    private ImageView imageView;
    private Button buttonNewPic;
    private Button buttonImage;
    private Button btnSet;
}

```

```

int width, height;
DisplayMetrics displayMetrics ;
WallpaperManager wallpaperManager ;
BitmapDrawable bitmapDrawable ;
Bitmap bitmap1, bitmap2 ;

```

