

## СОЗДАНИЕ ПРИЛОЖЕНИЯ «ПАМЯТЬ» ДЛЯ ПЛАТФОРМЫ ANDROID

*Андрюенко Никита Сергеевич*

Студент факультета математики и технологий программирования  
УО «Гомельский государственный университет имени Ф. Скорины»

Разработка приложений всегда начинается с тщательного обдумывания деталей и интерфейса, после чего начинается программная реализация всего задуманного.

Сначала опишем используемые программные средства, необходимые для предлагаемого приложения. Для разработки приложений для ОС Android необходимо установить Android Studio версии Portable [1]. При установке следует выбрать требующиеся для разработки платформы и элементы SDK библиотек. Еще требуется среда исполнения Java Runtime Environment, набор разработчика Java Development Kit [2]. Для тестирования приложения необходим эмулятор «Android Virtual Device» (AVD). AVD позволяет проводить виртуальное тестирование приложений на эмулированном мобильном устройстве с ОС Android [3]. Эмулятор позволяет создавать несколько виртуальных устройств, а также полностью задавать их конфигурацию.

Предлагаемое приложение является игрой, направленной на тренировку визуальной памяти. История игры насчитывает несколько десятков лет – и за это время она издавалась во множестве вариаций и достаточно часто воплощалась на компьютере.

В игровом наборе содержится некое количество карточек с картинками на одной стороне и одинаковых с другой, между собой карты образуют пары. До начала игры все карточки перемешиваются, поворачиваются рубашкой вверх и раскладываются на игровом поле. Игроки по очереди переворачивают по две карты. Если картинки на них совпадают, то игрок забирает пару, а затем продолжает ход и переворачивает ещё две карты. Если игрок перевернул карточки с разными картинками, то его задача - дать возможность всем увидеть, что именно изображено на картах, положить их на место картинкой вниз и передать ход следующему игроку. По ходу игры вы будете вынуждены запоминать множество картинок и их расположение, чтобы совершать наиболее эффективные ходы. Это позволяет тренировать память.

В приложении использовались такие стандартные элементы:

TextView – элемент, который содержит в себе текст.

ViewGroup – отвечает за отображение группы элементов.

Button – элемент кнопка, при нажатии происходит действие.

**LinearLayout** – выравнивает все дочерние объекты в одном направлении – вертикально или горизонтально. Направление задается при помощи атрибута ориентации `android:orientation`.

**RelativeLayout** – позволяет дочерним компонентам определять свою позицию относительно родительского компонента или относительно соседних дочерних элементов.

**Chronometr** – счётчик времени, позволяет узнать за какое время пользователь нашел все пары.

Созданные в xml экраны хранятся в папке `res/layout`.

После создания, xml файлы подключаются, управляются и изменяются в java файлах (`Activity`, `Fragment`). Экран начального меню выполнен в ярких цветах, что полностью соответствует паттернам `Material Design`. В стартовый `Activity` вложен главный фрагмент и настроены переходы на кнопках.

Код перехода по нажатию кнопки:

```
Button                newGameBtn                =                (Button)
rootView.findViewById(R.id.newGameBtn);
Button                aboutButton                =                (Button)
rootView.findViewById(R.id.AboutBtn);
newGameBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        MainMenuFragment.this.startActivity(new Intent(getActivity(),
        GameScreenActivity.class)); });
```

Приложение полностью реализовано и проведено его тестирование.

### Литература

1 Хашими, С. Разработка приложений для Android / С. Хашими. – М.: Бином, 2011. – 2125 с.

2 Эккель, Б. Философия Java, 4-ое издание / Б. Эккель. – М.: Бином, 2009. – 1768 с.

3 Голощапов А.Л. Google Android: Программирование для мобильных устройств. - СПб. БХВ-Петербург, 2011. – 1499с.