

Дифференцированный зачет по УП. 01
по специальности
09.02.07 Информационные системы и программирование

Тема: «Разработка игрового приложения «FillWord»»

Выполнил студент группы ИС-32 Ситимов М.А.
Руководитель УП: Грахов И. В.

Цель учебной практики- разработка игрового приложения для Android.

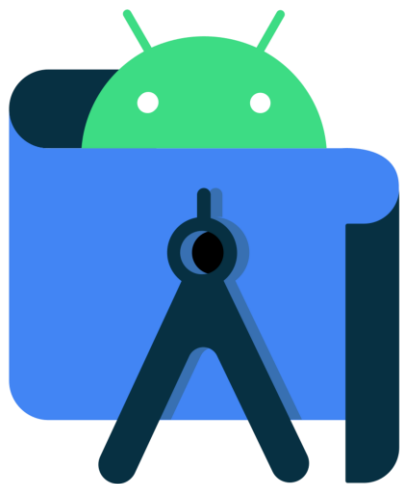
Объект исследования - мобильные игры

Предмет исследования - особенности реализации игрового приложения для Android, а также средства реализации программного продукта.

Для достижения поставленной цели в данном проекте были решены следующие задачи:

- спроектированы структура и функциональность игры;
- изучены и выбраны средства разработки мобильных игр;
- Разработан интерфейс приложения;
- реализована структура и функциональность игрового приложения;

Выбор инструментальных средств

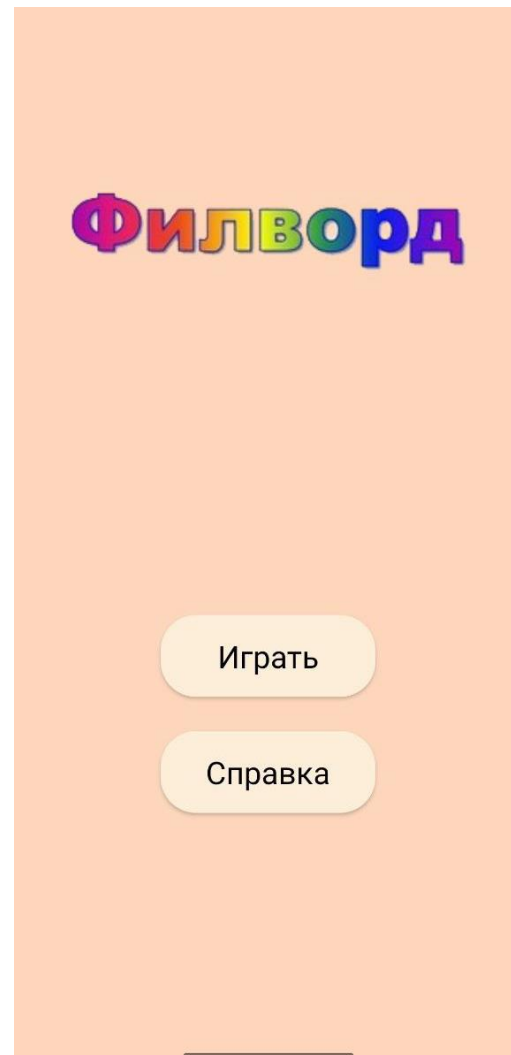


Android Studio



Adobe Photoshop

Главное меню



Основной экран игры

Найдите 6 слов

Н	К	Ж	М	Е	А	У	Н	Ё	Ж
М	З	П	Ь	Л	Д	Г	Й	Ц	Э
Ы	Д	Л	А	К	М	А	И	Т	Ш
Ч	Р	Ж	К	О	И	Д	Г	Ы	Ж
Г	А	Л	К	Т	Н	Ж	Ш	Р	Д
Ш	Й	Х	А	Л	Ч	Е	Ю	Ф	Ы
Е	В	Г	У	І	Ч	Т	Ж	Б	Ч
З	Е	Ю	Н	Н	Щ	И	А	Ч	Щ
Ч	Р	Ё	Т	Ы	Ш	Ы	Ш	Ы	В
Щ	О	Ш	И	Б	К	А	Ё	Й	Д

Подсказка

Основной экран игры

Найдите 6 слов

С	К	О	Т	Л	І	Н	Я	Ы	Н
И	Ф	Ё	Ъ	Ы	О	И	А	П	Ё
З	А	Н	О	Ш	И	Б	К	А	Г
Ь	Д	А	К	Г	Э	Ь	К	Ш	А
Х	М	С	Ы	Т	Е	Ц	А	И	Д
Ъ	И	У	Ч	А	М	Ю	У	Й	Ж
Ф	Н	Ш	З	Г	П	Р	Н	А	Е
Э	Ж	О	Ч	Г	Б	Р	Т	Я	Т
Ц	Н	Д	Р	А	Й	В	Е	Р	Б
Ц	Н	Д	Ъ	Я	Щ	Ж	Ю	Ж	Ы



Подсказка

Найдите 6 слов

Е	Ъ	А	Ь	О	Д	Л	Ф	А	Ы
О	Б	К	О	С	У	О	Т	Д	И
Т	Ч	К	Л	Ё	С	Ш	Ъ	М	А
П	Н	А	Б	Ф	Г	О	Л	И	Г
Р	Л	У	Ж	С	К	Б	У	Н	А
Ы	Ю	Н	Ф	П	У	Ц	Г	Ь	Д
С	Ц	Т	Й	Ы	Ы	Н	Й	Е	Ж
Щ	О	Ш	И	Б	К	А	Щ	Щ	Е
У	К	О	Т	Л	І	Н	Щ	Ь	Т
Щ	Ц	Д	Р	А	Й	В	Е	Р	В

Подсказка



Экран победы

Найдите 6 слов

я	ч	у	з	ё	ж	ц	ш	ш	р
ъ	б	э	г	ч	а	с	б	я	п
ъ	а							ю	г
р	к							б	а
а	к							о	д
д	а							у	ж
м	у							г	е
и	н							н	т
н	т	д	р	а	й	в	е	р	о
ю	з	б	ъ	з	з	о	й	т	ч

Поздравляе
м!!

Вы прошли
игру!

Исходный код

```
when (event.action) {
    MotionEvent.ACTION_DOWN -> {
        v.background = ContextCompat.getDrawable(context, this, R.drawable.selected_cell_background)
        xInitial = event.x
        yInitial = event.y
    }

    MotionEvent.ACTION_MOVE -> {
        if(xInitial != -1f && yInitial != -1f){

            val tag = v.tag.toString()
            val tagInt = tag.toInt()

            xDiff = xInitial - event.x
            yDiff = yInitial - event.y

            if(swipeState == SwipeState.Undefined || swipeState == SwipeState.Horizontal){
                when {
                    xDiff > cellWidth -> {
                        // движение влево
                        if(prevXDiff == -1f || prevXDiff != -1f && prevXDiff < xDiff){
                            selectSingleCell((tagInt - (xDiff / cellWidth).toInt()).toString())
                            swipeState = SwipeState.Horizontal
                        } else if ( prevXDiff != -1f && prevXDiff > xDiff){
                            unselectSingleCell((tagInt - (prevXDiff / cellWidth).toInt()).toString())
                        }
                    }
                    (-1) * xDiff > cellWidth -> {
                        // движение вправо
                        if(prevXDiff == -1f || prevXDiff != -1f && prevXDiff > xDiff){
                            selectSingleCell((tagInt + -1 * (xDiff / cellWidth).toInt()).toString())
                            swipeState = SwipeState.Horizontal
                        } else if ( prevXDiff != -1f && prevXDiff < xDiff){
                            unselectSingleCell((tagInt - (prevXDiff / cellWidth).toInt()).toString())
                        }
                    }
                }
            }

            if(swipeState == SwipeState.Undefined || swipeState == SwipeState.Vertical){
                when {
                    yDiff > cellWidth -> {
                        // движение вверх
                        if(prevYDiff == -1f || prevYDiff != -1f && prevYDiff < yDiff){
                            selectSingleCell((tagInt - 10*(yDiff/cellWidth).toInt()).toString())
                            swipeState = SwipeState.Vertical
                        } else if (prevYDiff != -1f && prevYDiff > yDiff){
                            unselectSingleCell((tagInt - 10*(yDiff/cellWidth).toInt()).toString())
                        }
                    }
                    (-1)*yDiff > cellWidth -> {
                        // движение вниз
                        if(prevYDiff == -1f || prevYDiff != -1f && prevYDiff > yDiff){
                            selectSingleCell((tagInt + -10*(yDiff/cellWidth).toInt()).toString())
                        }
                    }
                }
            }
        }
    }
}

MotionEvent.ACTION_UP -> {
    val tag = v.tag.toString()
    val tagInt = tag.toInt()
    var finalTag = tag

    if(swipeState == SwipeState.Horizontal){
        finalTag = when {
            xDiff > cellWidth -> {
                (tagInt - (xDiff/cellWidth).toInt()).toString()
            }
            -1*xDiff > cellWidth -> {
                (tagInt + -1*(xDiff/cellWidth).toInt()).toString()
            }
            else -> tag
        }
    } else if(swipeState == SwipeState.Vertical){
        finalTag = when {
            yDiff > cellWidth -> {
                (tagInt - 10*(yDiff/cellWidth).toInt()).toString()
            }
            -1*yDiff > cellWidth -> {
                (tagInt + -10*(yDiff/cellWidth).toInt()).toString()
            }
            else -> tag
        }
    }

    checkIfRangeIsValid(v.tag.toString(), finalTag)
}
```

Спасибо за внимание!