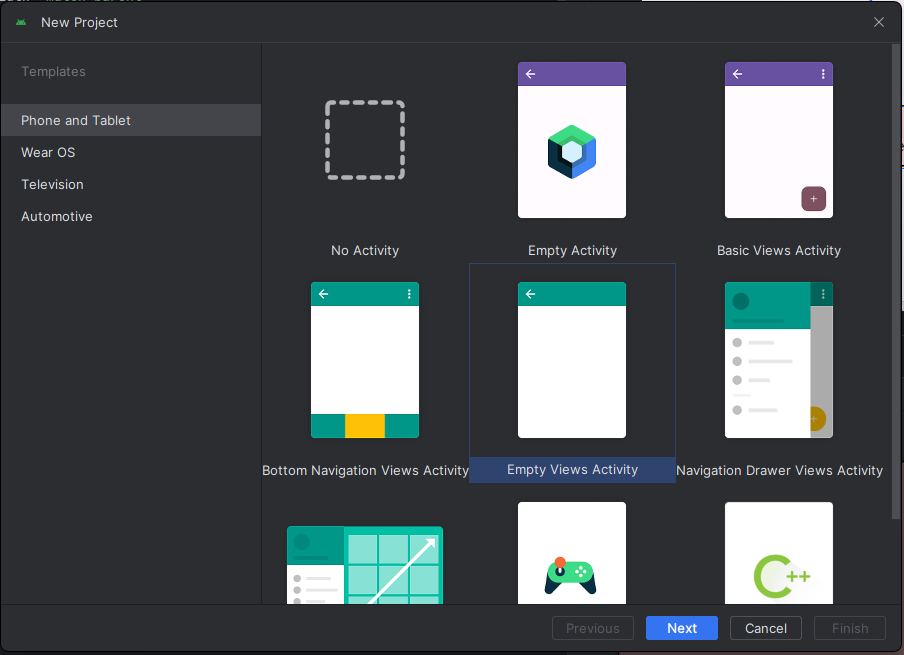
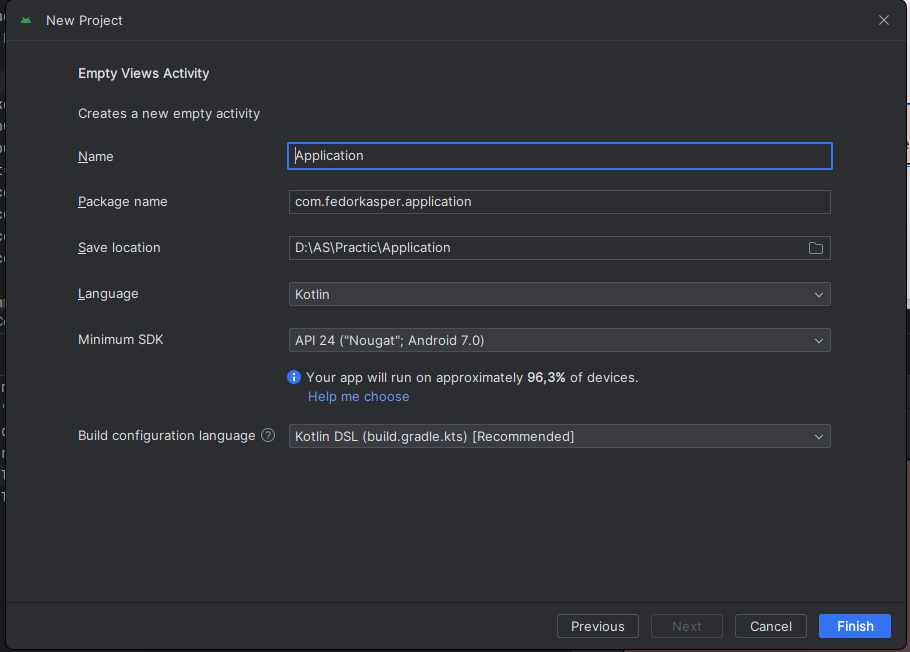
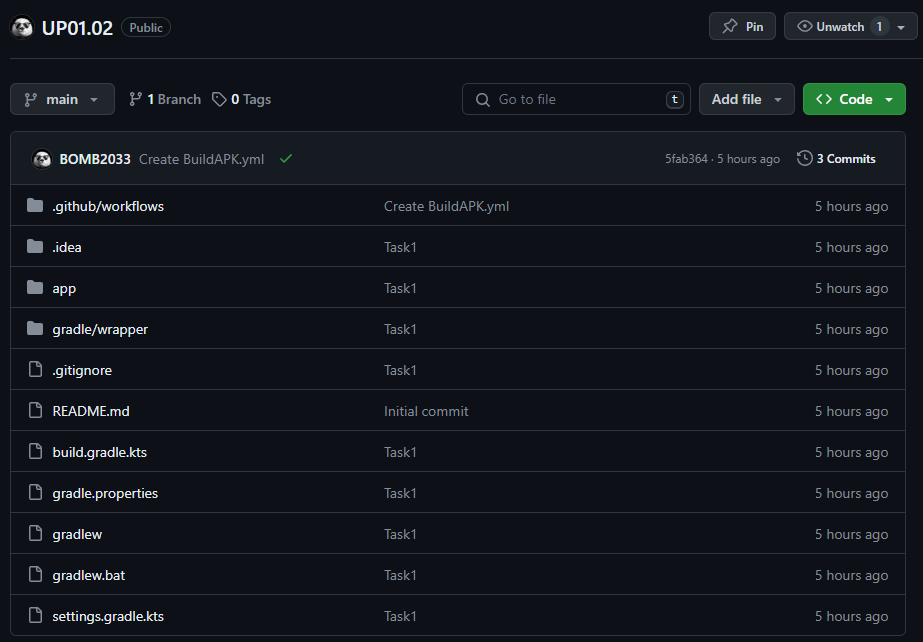
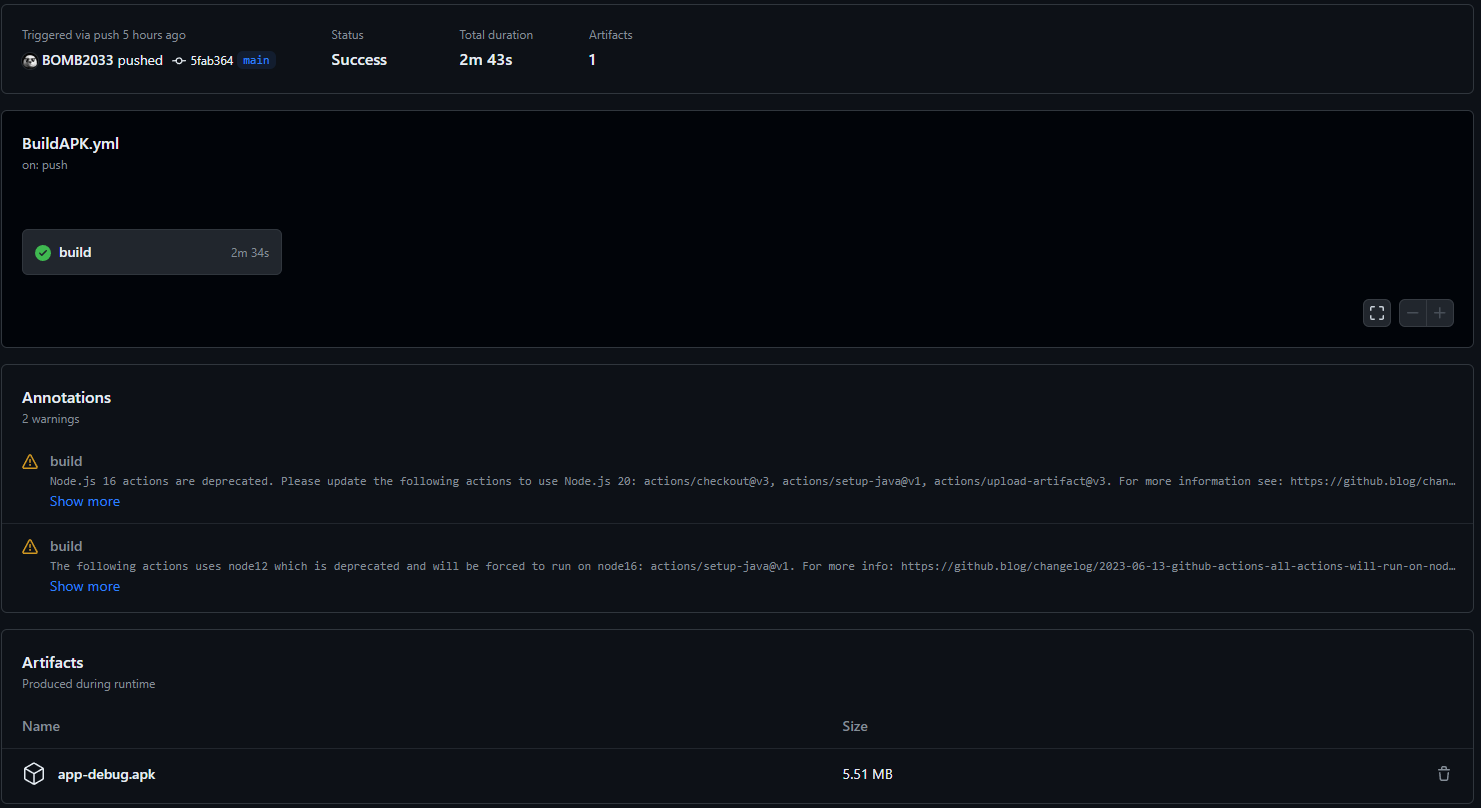
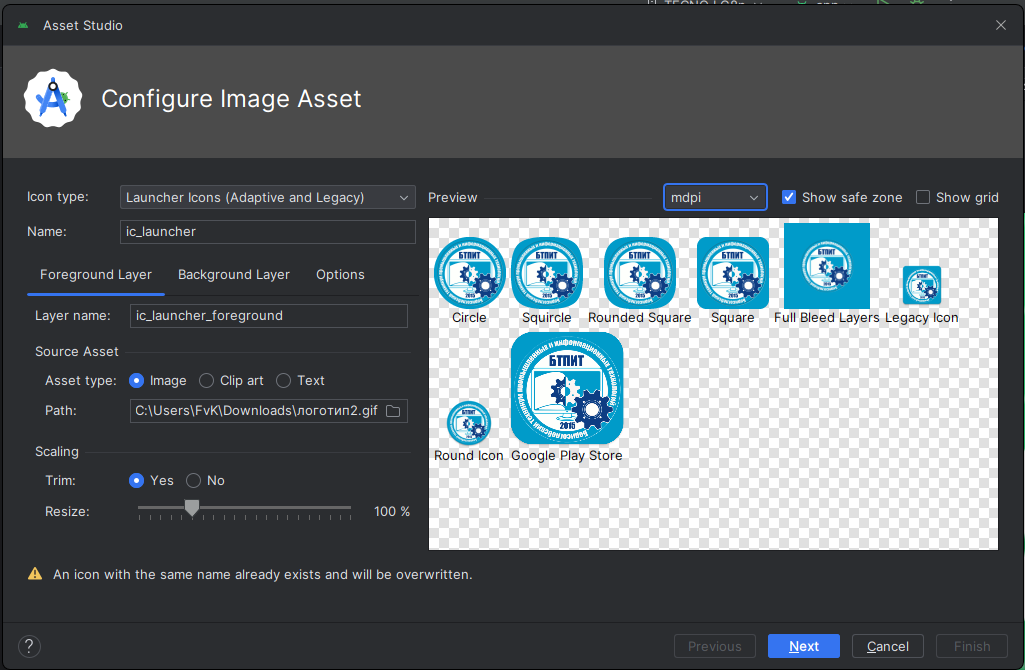
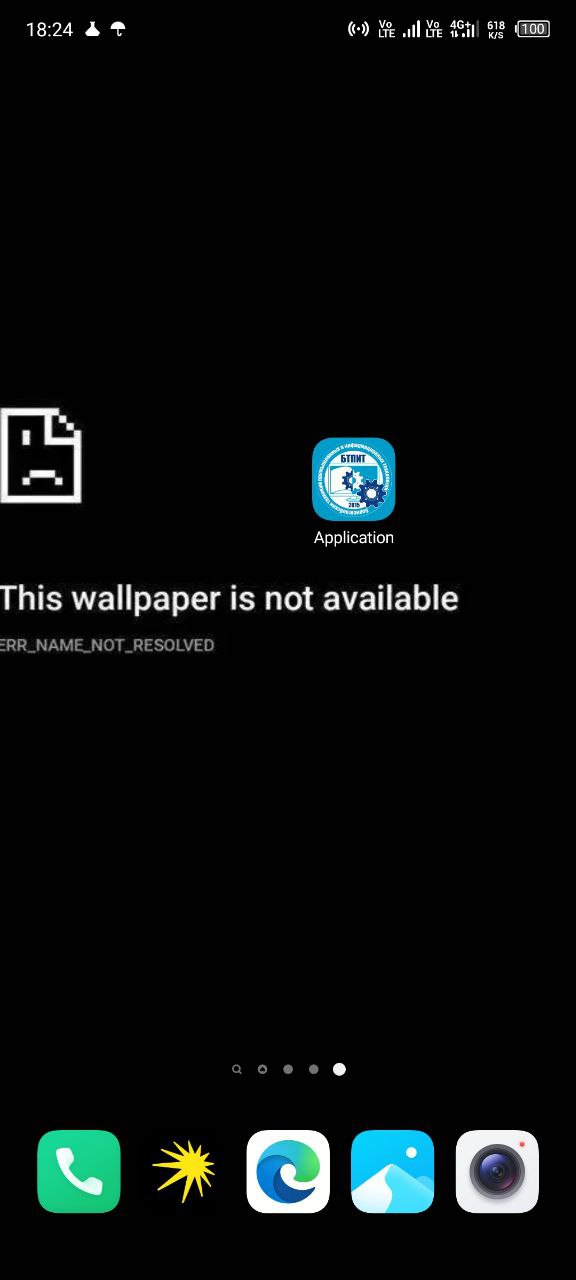
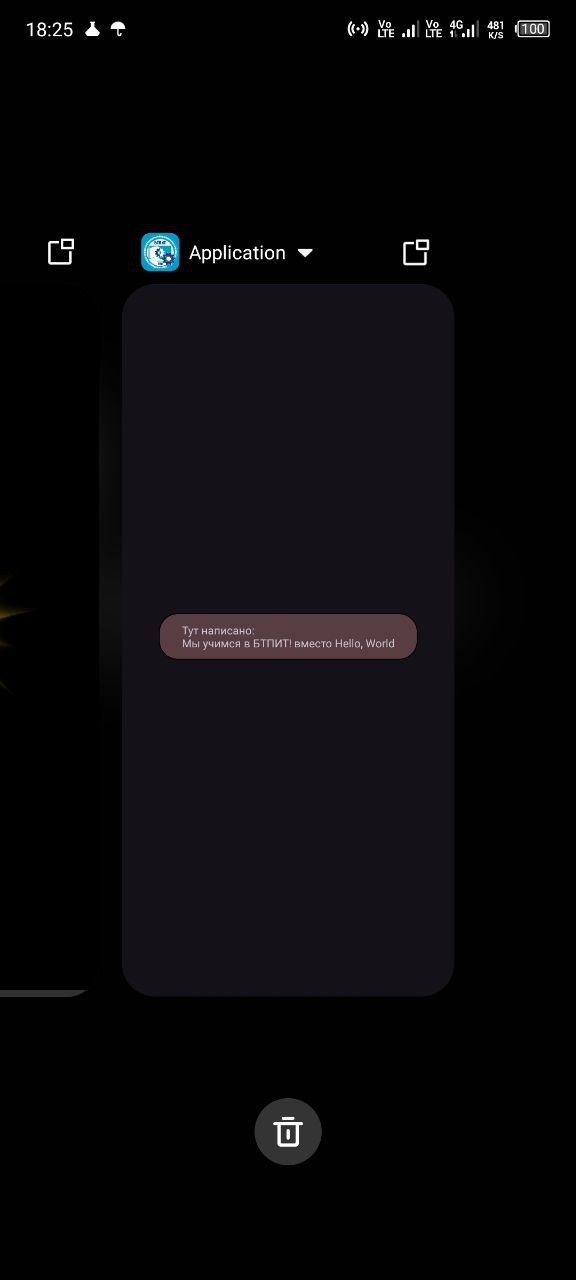
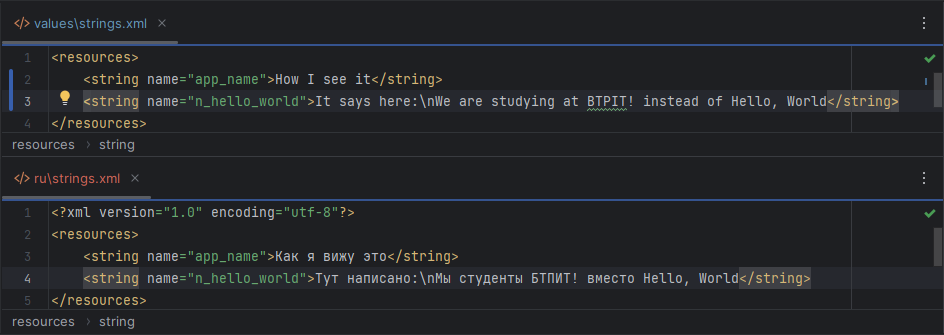
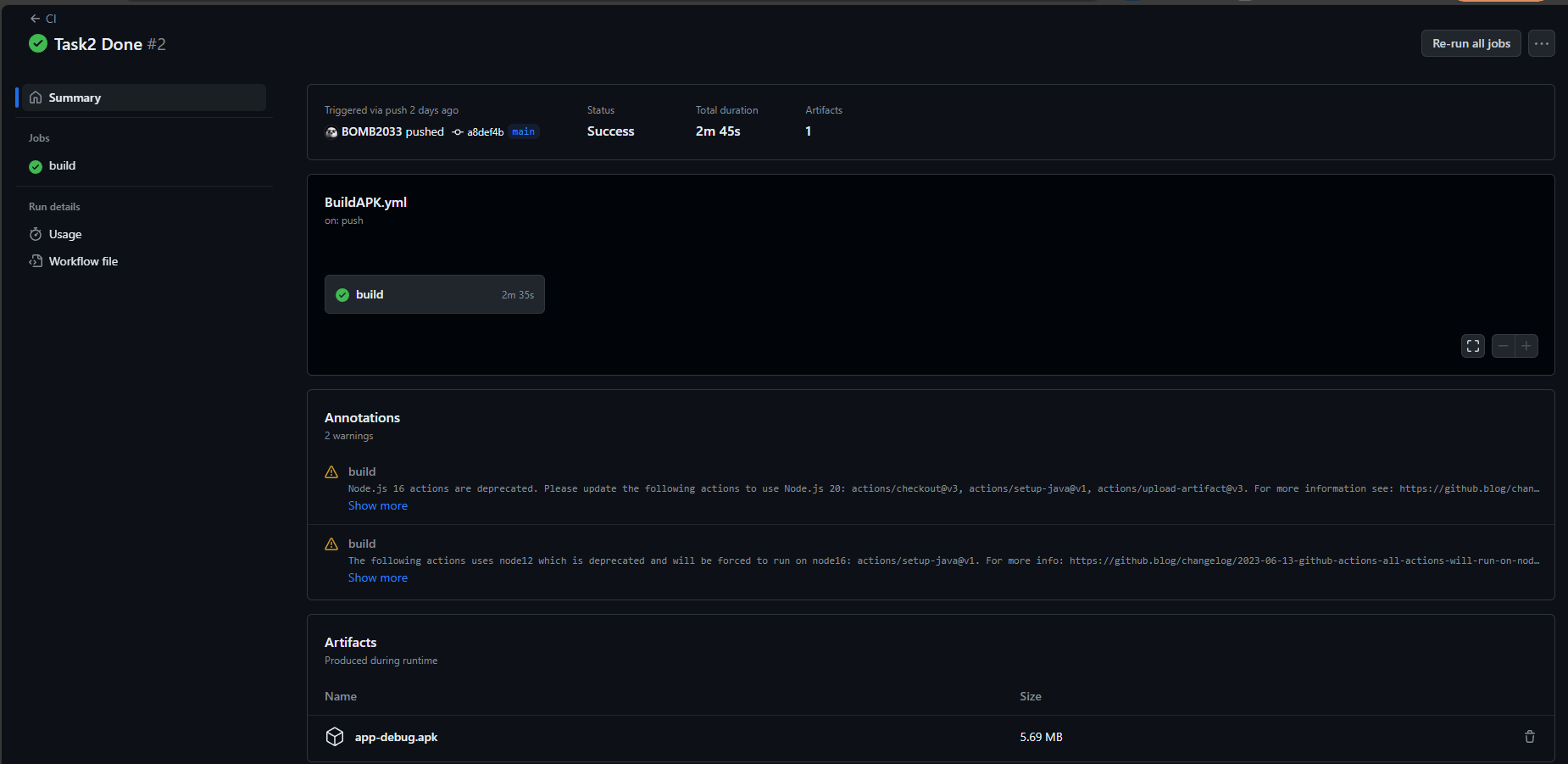
Оглавление позже

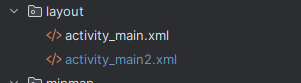
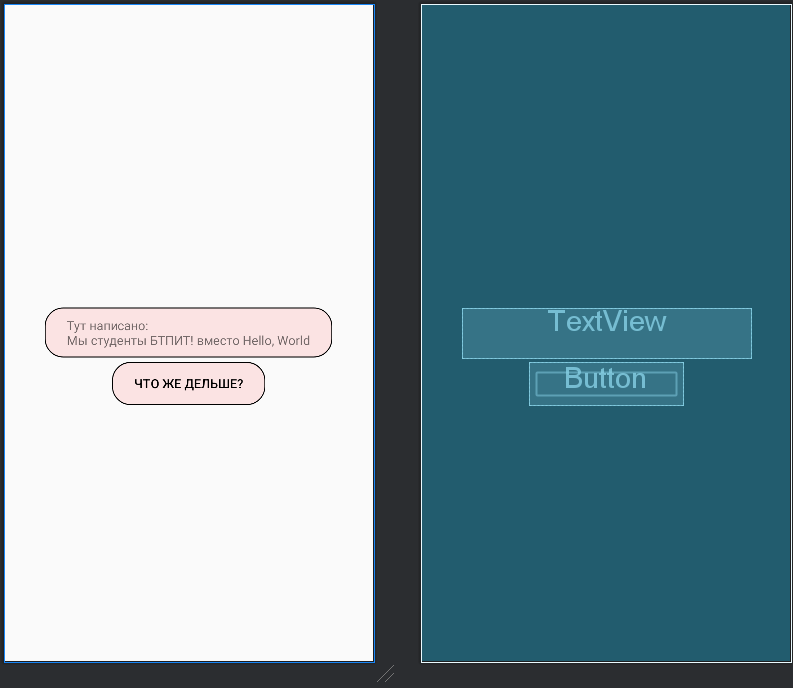
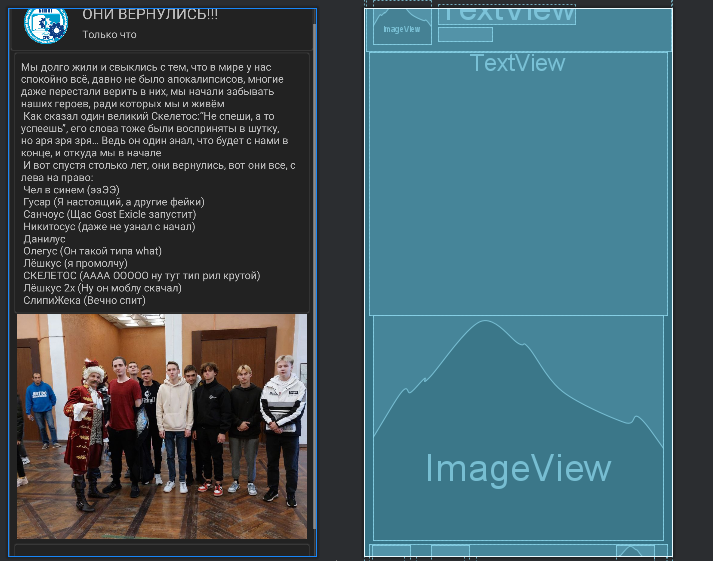
## Занятие 1

1. Создал проект в Android Studio, с пустым активити  
2. Изменил главную активити 
3. Отправил на GitHub 
4. Создал в GitHub действие сборки APK файла 

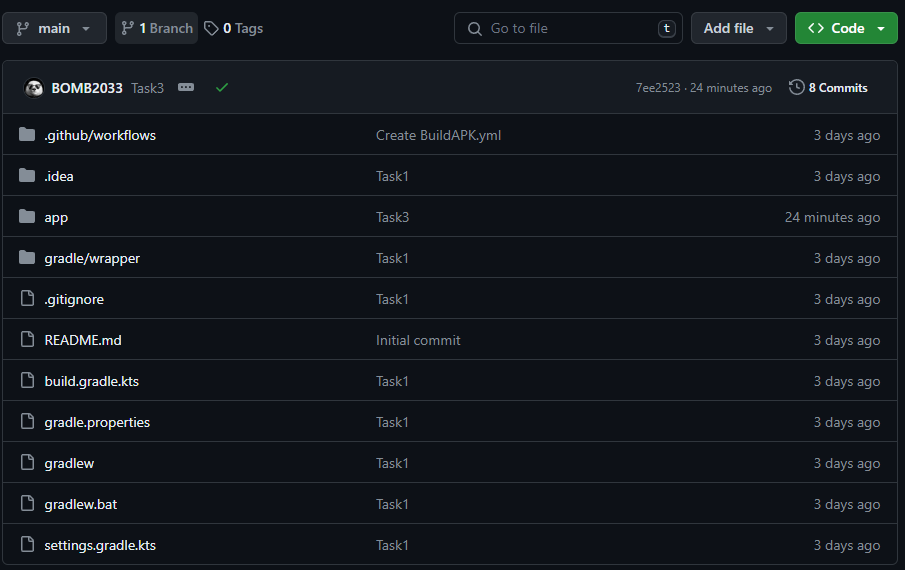
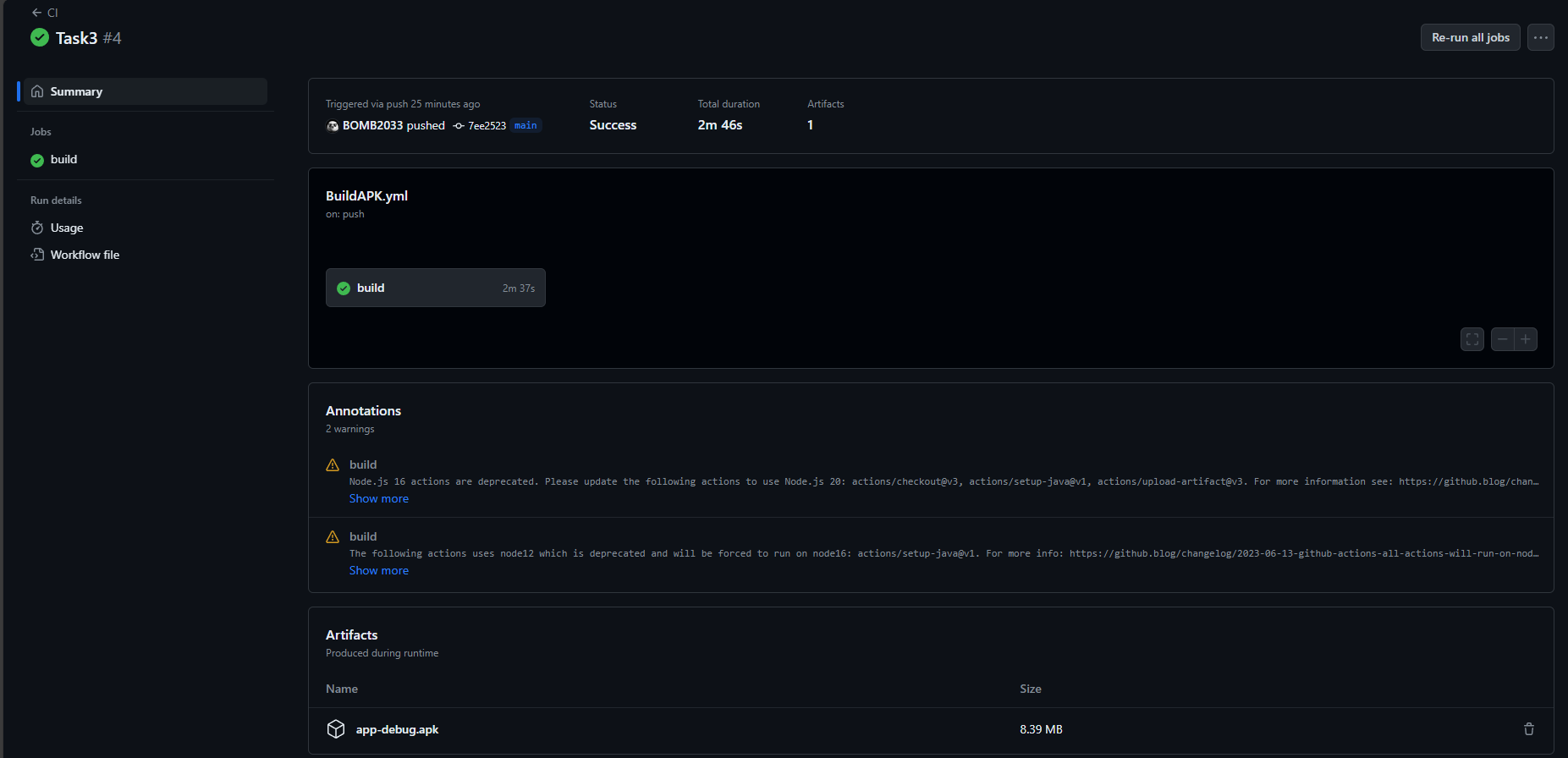
## Занятие 2

1. Открыл предыдущий проект и добавил изображения в mipmap 
2. Результат шикарный😍  
3. Создал файл файл переводов 
4. Отправил на гит
5. И выполнил сборку

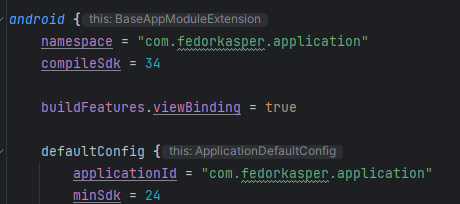
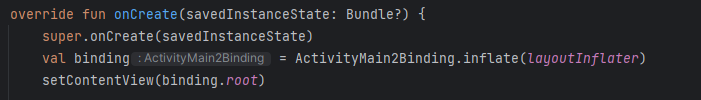
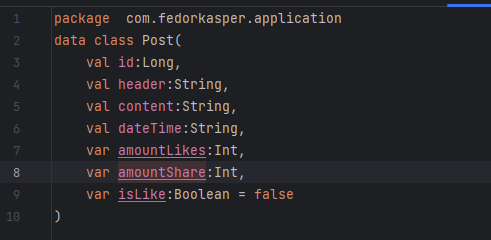
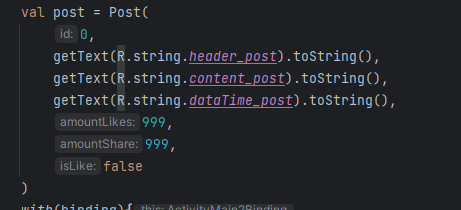
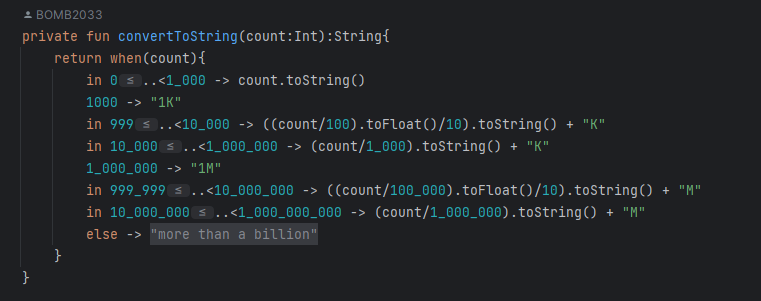
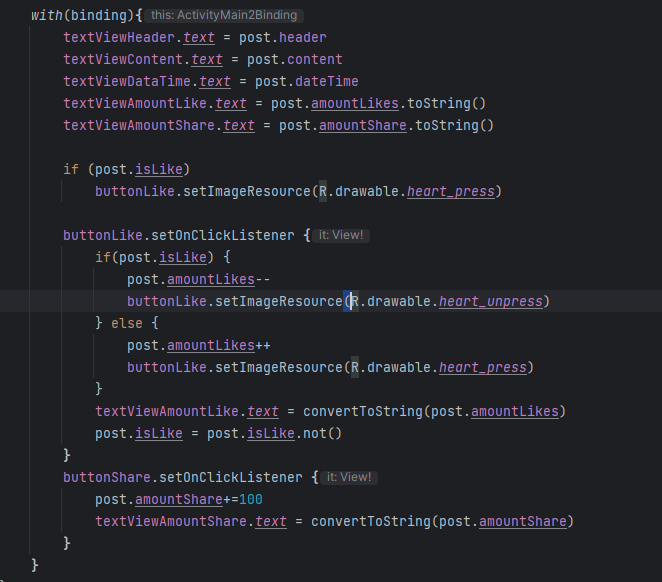
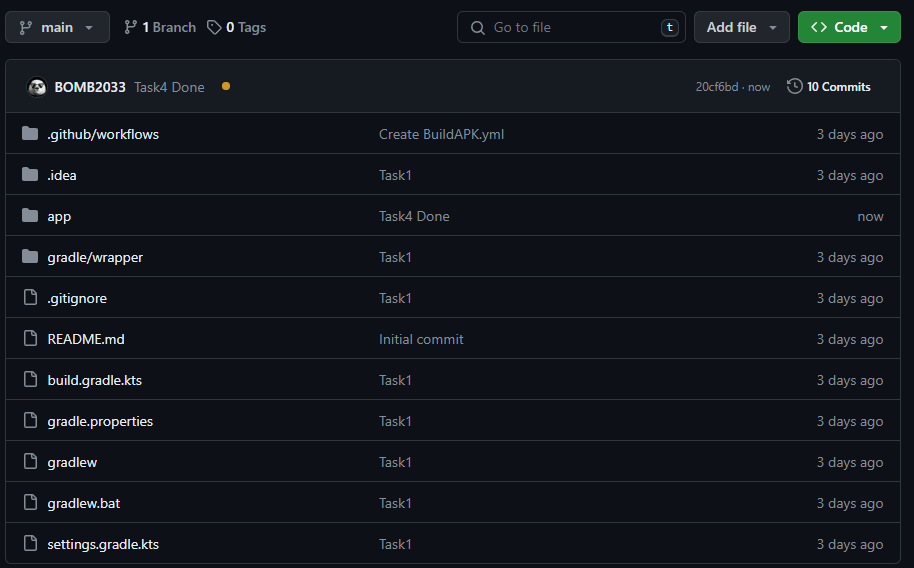
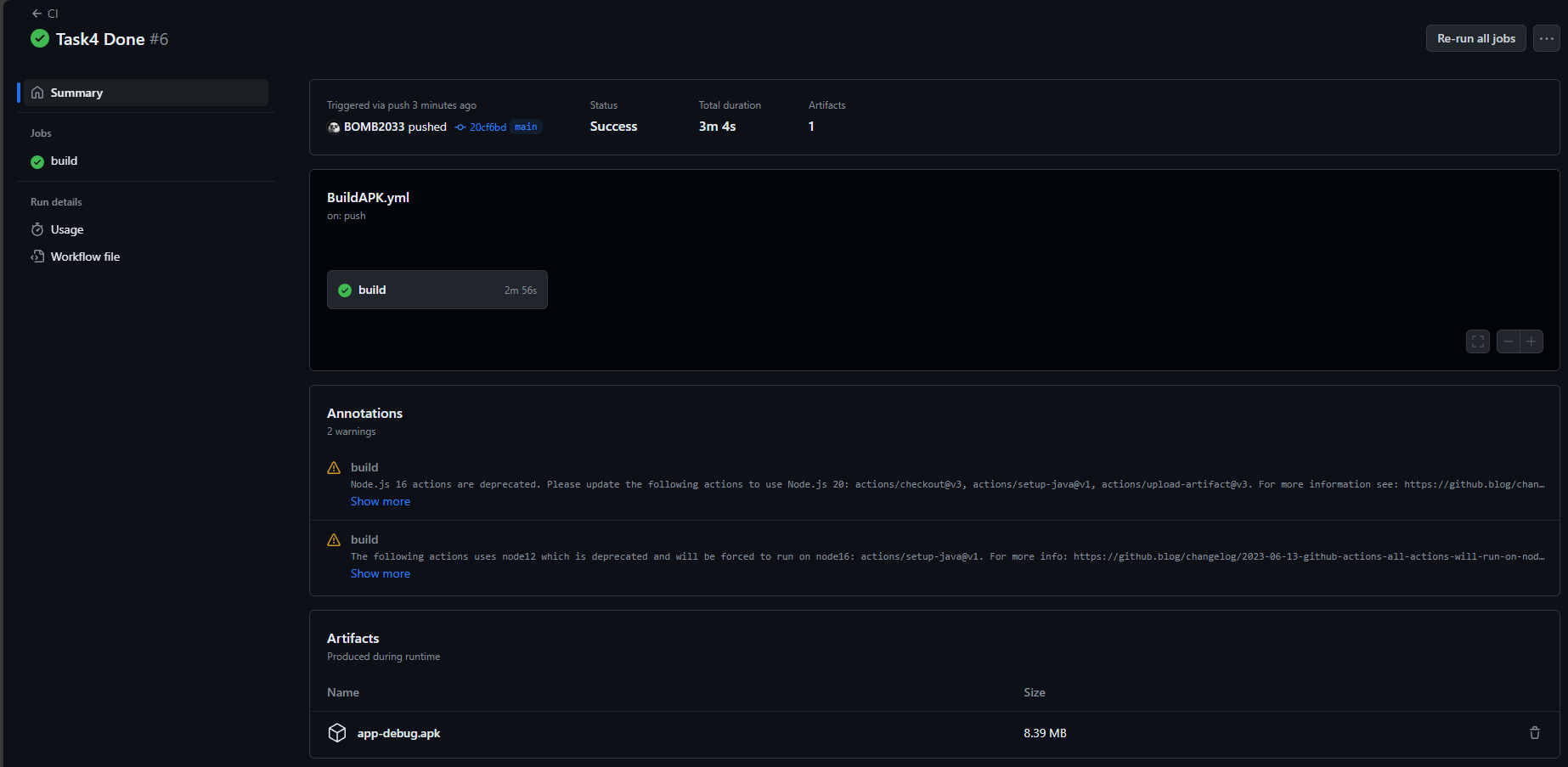
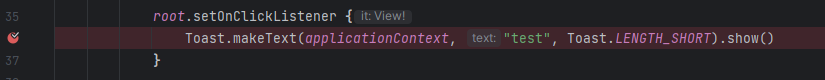
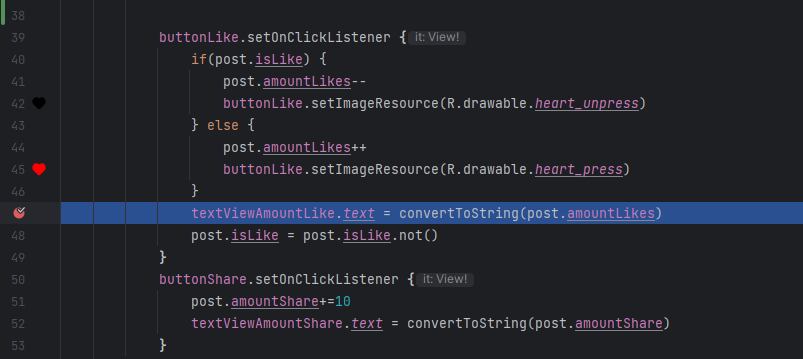
## Занятие 3

1. Создал вторую активити 
2. Добавил кнопку на первую активити и добавил ей событие с переходом на вторую активити  
3. На новом активити разместил и заполнил элементы 
4. Код:

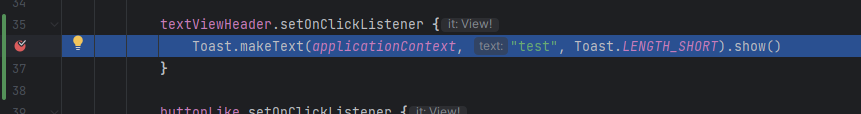
<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:background="@drawable/background\_shadow"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:scrollbarThumbVertical="@id/textViewPoem"  
 tools:context=".MainActivity2">  
 <androidx.constraintlayout.widget.ConstraintLayout android:id="@+id/constraintLayout1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:padding="10dp"  
 android:background="@drawable/background\_shadow"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintHeight\_percent="0.1">  
 <ImageView android:id="@+id/imageView1"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:contentDescription="@string/can\_t\_open\_image"  
 android:src="@drawable/logo\_btpit"  
 android:scaleType="centerInside"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintWidth\_percent="0.2"/>  
 <TextView android:id="@+id/textView1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textSize="20sp"  
 android:layout\_marginStart="10dp"  
 android:text="@string/Sunday\_June\_22\_1941"  
 app:layout\_constraintTop\_toTopOf="@+id/imageView1"  
 app:layout\_constraintBottom\_toTopOf="@id/textView2"  
 app:layout\_constraintStart\_toEndOf="@id/imageView1"/>  
 <TextView android:id="@+id/textView2"  
 android:layout\_width="wrap\_content"  
 android:layout\_marginStart="10dp"  
 android:layout\_height="wrap\_content"  
 android:text="@string/date"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView1"  
 app:layout\_constraintBottom\_toBottomOf="@id/imageView1"  
 app:layout\_constraintStart\_toEndOf="@id/imageView1"/>  
 </androidx.constraintlayout.widget.ConstraintLayout>  
 <TextView android:id="@+id/textViewPoem"  
 android:background="@drawable/background\_shadow"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="@string/poem"  
 android:layout\_marginHorizontal="5dp"  
 android:padding="10dp"  
 app:layout\_constraintTop\_toBottomOf="@+id/constraintLayout1"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"/>  
  
 <ImageView  
 android:id="@+id/imageViewBTPIT"  
 android:layout\_width="match\_parent"  
 android:layout\_height="300dp"  
 android:scaleType="centerCrop"  
 android:src="@drawable/img\_btpit\_people"  
 android:layout\_marginHorizontal="10dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@id/textViewPoem" />  
 <androidx.constraintlayout.widget.ConstraintLayout android:id="@+id/constraintLayout2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="5dp"  
 android:background="@drawable/background\_shadow"  
 app:layout\_constraintTop\_toBottomOf="@id/imageViewBTPIT"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintHeight\_percent="0.5">  
  
 <ImageButton android:id="@+id/buttonLike"  
 android:layout\_width="50dp"  
 android:layout\_height="50dp"  
 android:src="@drawable/heart\_unpress"  
 android:background="#00FFFFFF"  
 android:padding="0dp"  
  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 />  
 <TextView android:id="@+id/textViewAmountLike"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="52"  
 android:layout\_marginStart="-7dp"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintStart\_toEndOf="@id/buttonLike"/>  
  
 <ImageButton android:id="@+id/buttonShare"  
 android:layout\_width="50dp"  
 android:layout\_height="50dp"  
 android:src="@drawable/share"  
 android:background="#00FFFFFF"  
 android:padding="0dp"  
 android:layout\_marginStart="20dp"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintStart\_toEndOf="@id/textViewAmountLike"  
 />  
 <TextView android:id="@+id/textViewAmountShare"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="52"  
 android:layout\_marginStart="-7dp"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintStart\_toEndOf="@id/buttonShare"/>  
  
  
 <ImageView android:id="@+id/imageViewSee"  
 android:layout\_width="50dp"  
 android:layout\_height="50dp"  
 android:scaleType="centerInside"  
 android:src="@drawable/eye"  
 android:background="#00FFFFFF"  
 android:padding="0dp"  
  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toStartOf="@+id/textViewAmountSee"  
 />  
 <TextView android:id="@+id/textViewAmountSee"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="52"  
 android:layout\_marginStart="-7dp"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"/>  
 </androidx.constraintlayout.widget.ConstraintLayout>  
  
</androidx.constraintlayout.widget.ConstraintLayout>

1. Отправил на гит:
2. Выполнил сборку АПК

## Занятие 4

1. Включил биндинг 
2. Создал биндинг
3. Создал класс для поста
4. Заполнил класс для поста
5. Создал метод для конвертации количества в тысячи и миллионы 
6. Задал взаимосвязь класса с элементами
7. Отправил на Гит
8. Выполнил действие сборки APK
9. Установил обработчик события так же на root элементИ при отладке куда бы не нажимал, приложение не останавливается здесь (потому что элемент root скрыт другими элементами лол)
10. Точка остановки на кнопке лайка при нажатии на копку

Останавливается.

1. Так же остановка по клику на header с точкой

Изи

1. Какой из обработчиков сработал при клике на кнопку Like?

Обработчик события кнопки Like

1. Сработал ли обработчик на binding.root при клике на кнопку с тремя точками?

Нет

1. Сработал ли обработчик на binding.root при клике на текст?

Нет

1. Сработал ли обработчик на binding.root при клике на аватар до установки на avatar собственного обработчика?

Нет

1. Сработал ли обработчик на binding.root при клике на аватар после установки на avatar собственного обработчика?

Нет, он сработает, если кликнуть по свободному месту

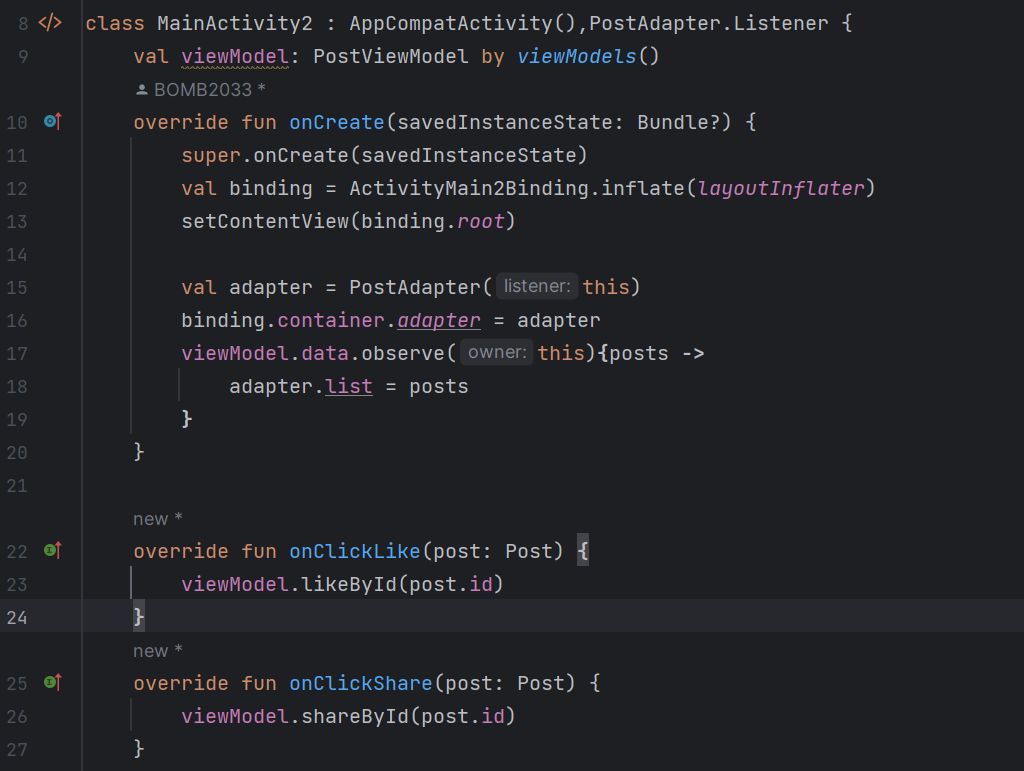
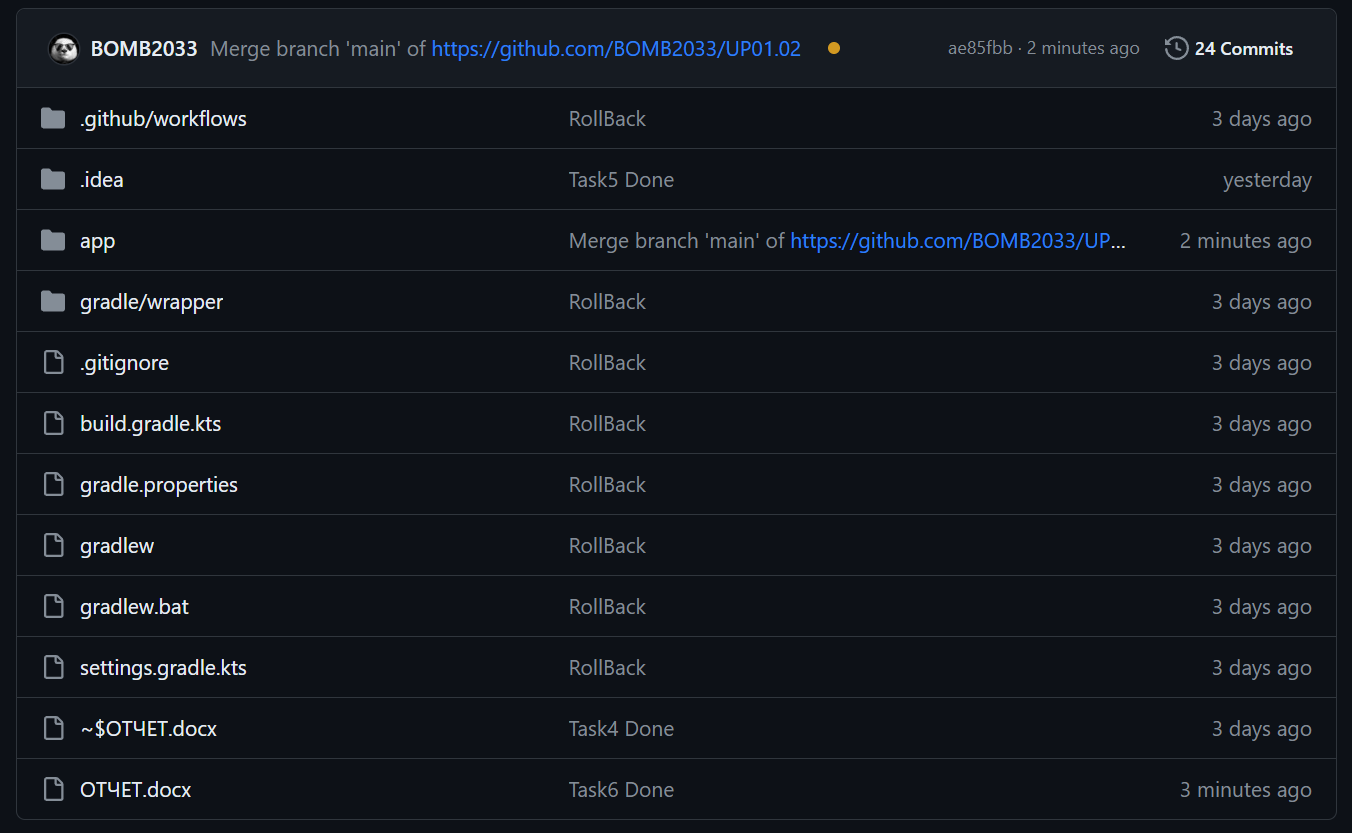
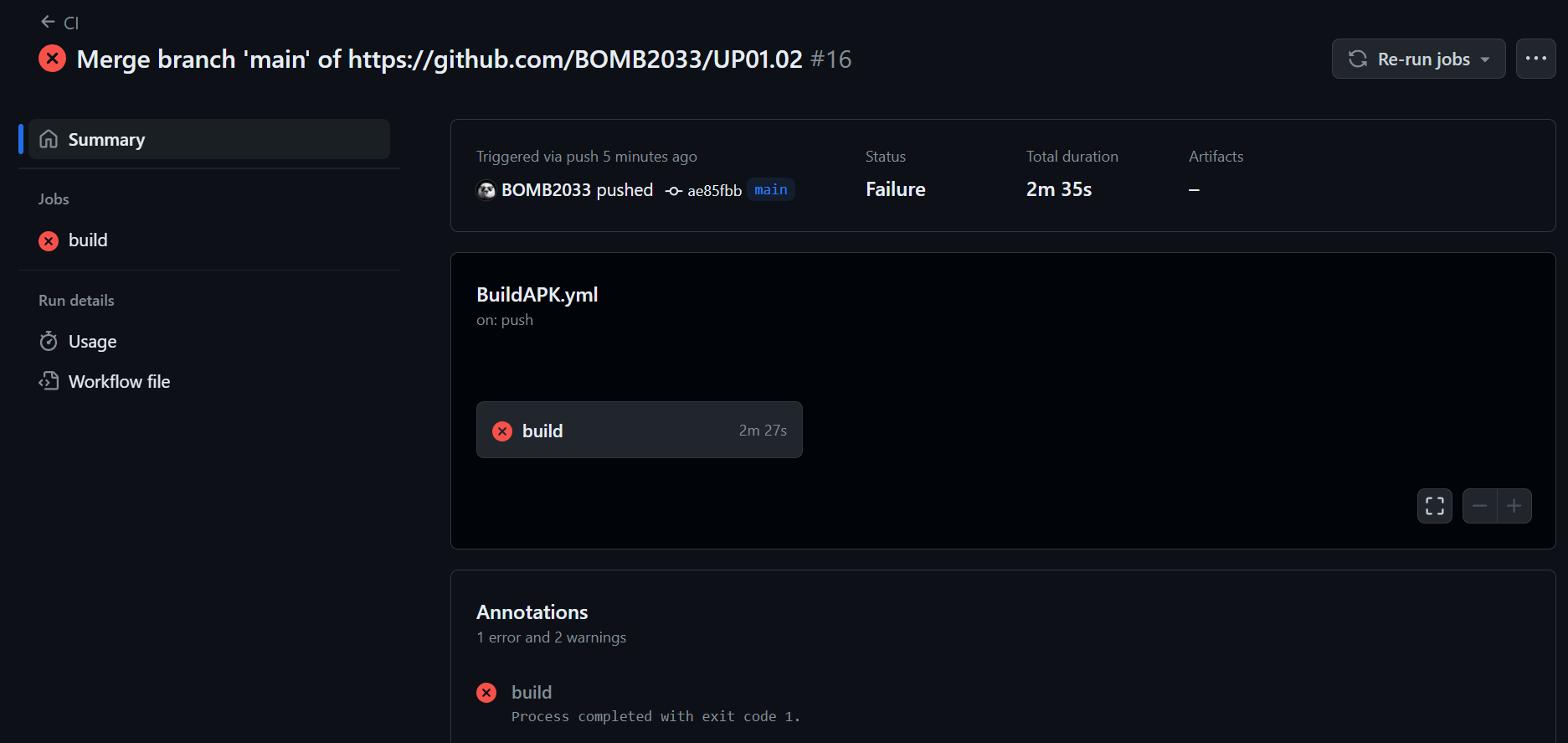
1. Отправил на гит (добавил отчет)

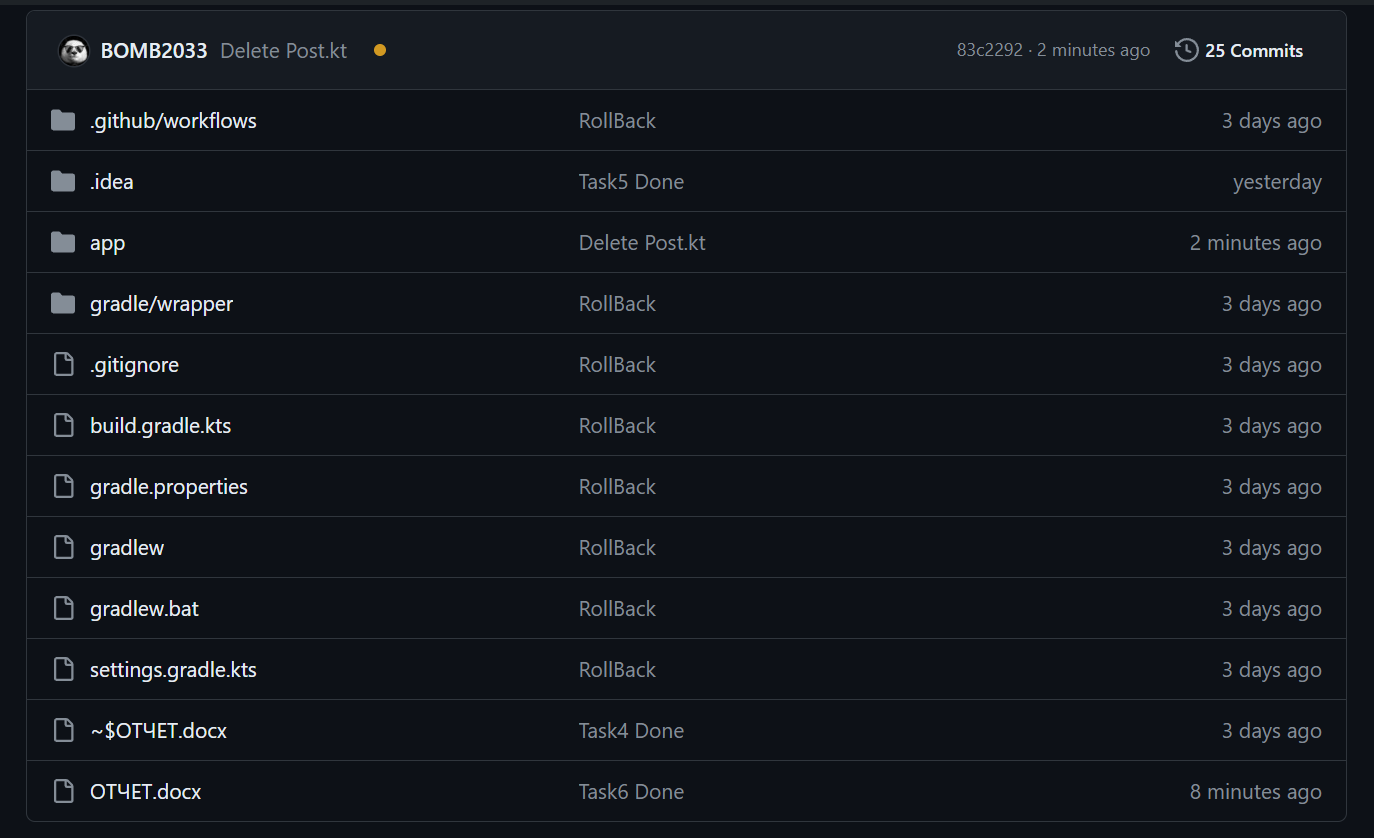
## Занятие 6

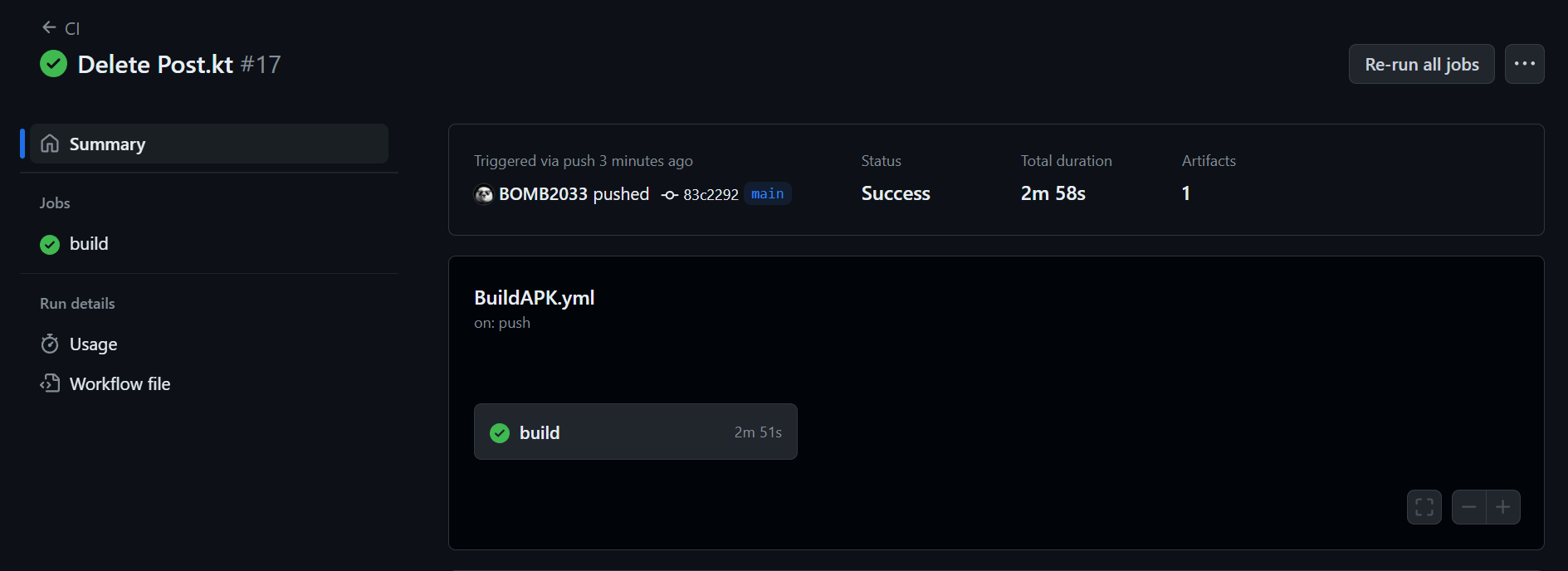
1. Реализовал отображение списков на базе RecyclerView

Класс PostAdapter:

package com.fedorkasper.application  
  
import android.view.LayoutInflater  
import android.view.ViewGroup  
import androidx.recyclerview.widget.RecyclerView  
import com.fedorkasper.application.databinding.CardPostBinding  
  
class PostAdapter(val listener: Listener):RecyclerView.Adapter<PostAdapter.PostViewHolder>() {  
 var list = *emptyList*<Post>()  
 set(value) {  
 **field** = value  
 notifyDataSetChanged()  
 }  
 class PostViewHolder(  
 private val binding: CardPostBinding  
 ):RecyclerView.ViewHolder(binding.*root*) {  
 fun bind(post: Post,listener: Listener) {  
 binding.*apply* **{** textViewHeader.*text* = post.header  
 textViewDataTime.*text* = post.dateTime  
 textViewContent.*text* = post.content  
  
 textViewAmountLike.*text* = *convertToString*(post.amountLikes)  
 textViewAmountShare.*text* = *convertToString*(post.amountShares)  
   
 buttonLike.setImageResource(if (post.isLike) R.drawable.*heart\_press* else R.drawable.*heart\_unpress*)  
 buttonLike.setOnClickListener **{** listener.onClickLike(post)  
 **}** buttonShare.setOnClickListener **{** listener.onClickShare(post)  
 **}  
 }** }  
 }  
 override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): PostViewHolder {  
 val binding = CardPostBinding.inflate(LayoutInflater.from(parent.*context*), parent, false)  
 return PostViewHolder(binding)  
 }  
 override fun onBindViewHolder(holder: PostViewHolder, position:Int){  
 val post = list[position]  
 holder.bind(post, listener)  
 }  
 override fun getItemCount() :Int = list. size  
  
 interface Listener{  
 fun onClickLike(post: Post)  
 fun onClickShare(post: Post)  
 }  
}  
private fun convertToString(count:Int):String{  
 return when(count){  
 in 0..<1\_000 -> count.toString()  
 in 1000..<1\_100-> "1K"  
 in 1\_100..<10\_000 -> ((count/100).toFloat()/10).toString() + "K"  
 in 10\_000..<1\_000\_000 -> (count/1\_000).toString() + "K"  
 in 1\_000\_000..<1\_100\_000 -> "1M"  
 in 1\_100\_000..<10\_000\_000 -> ((count/100\_000).toFloat()/10).toString() + "M"  
 in 10\_000\_000..<1\_000\_000\_000 -> (count/1\_000\_000).toString() + "M"  
 else -> "ꚙ"  
 }  
}

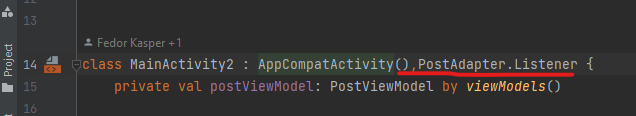
1. Добавил интерфейсы на Share тоже 
2. Опубликовал изменения на Гит 
3. И конечно же файл собрался успешно 
4. А нет, была ошибка при отправке и приёме, был принят старый дата класс Post, что и привело к ошибкам, пофиксив его всё стало успешно

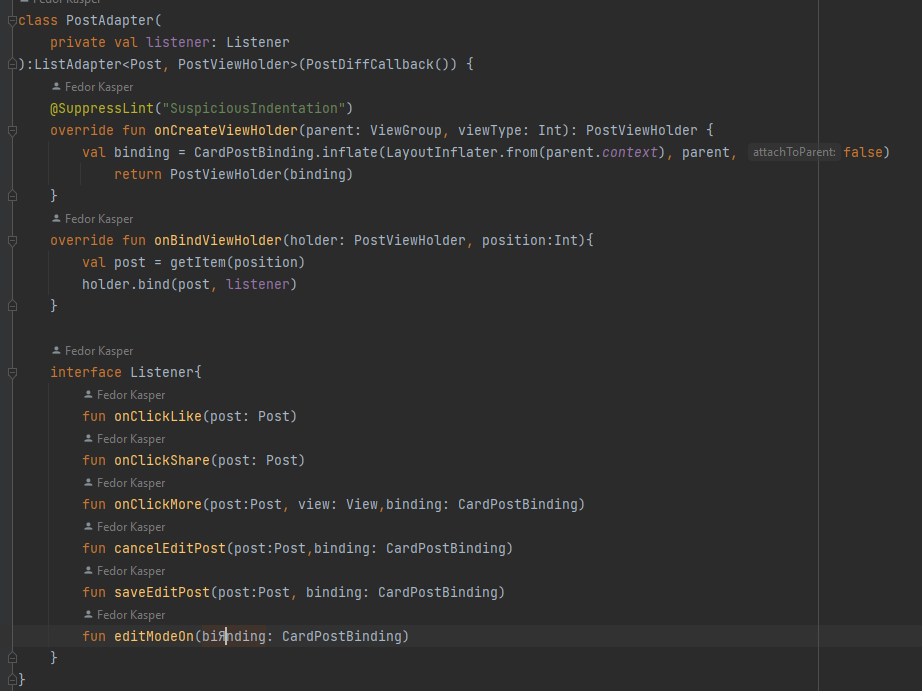
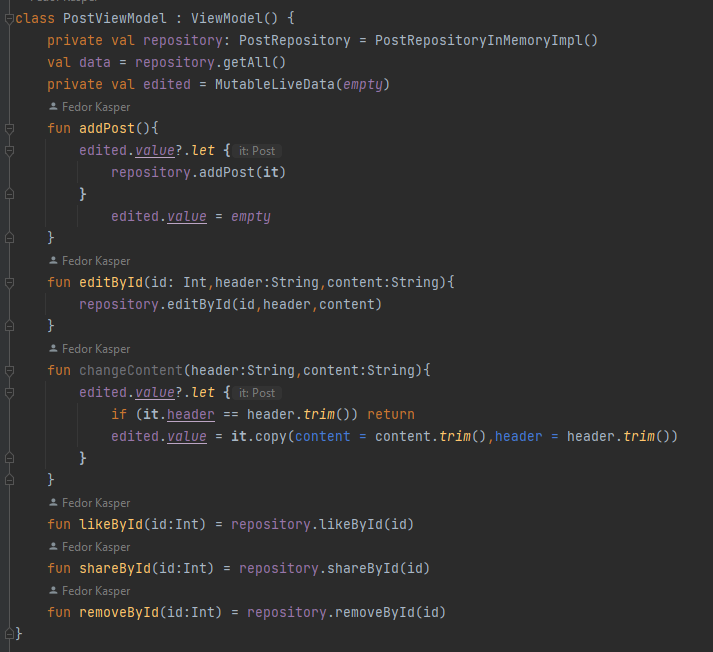


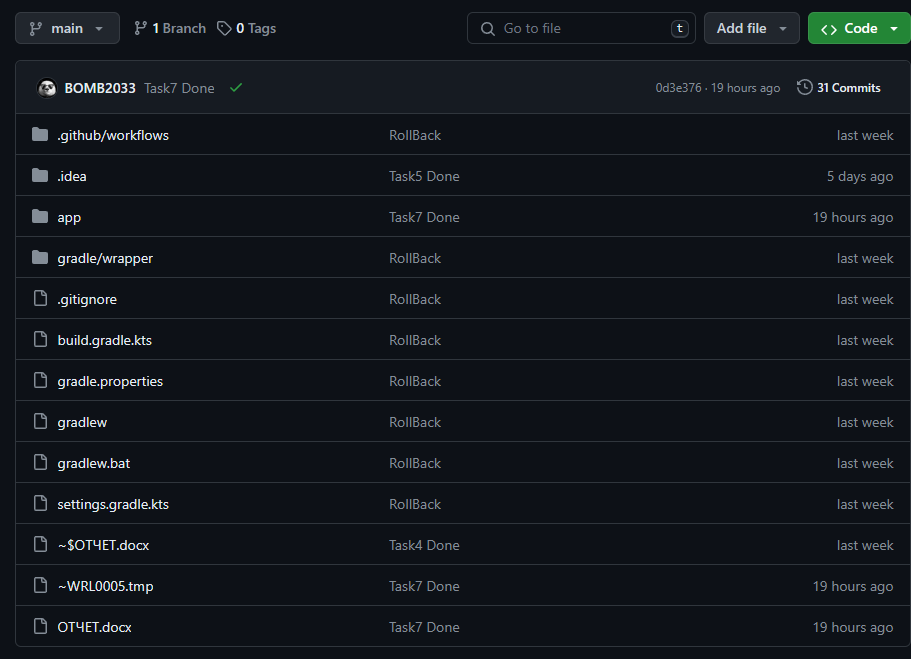
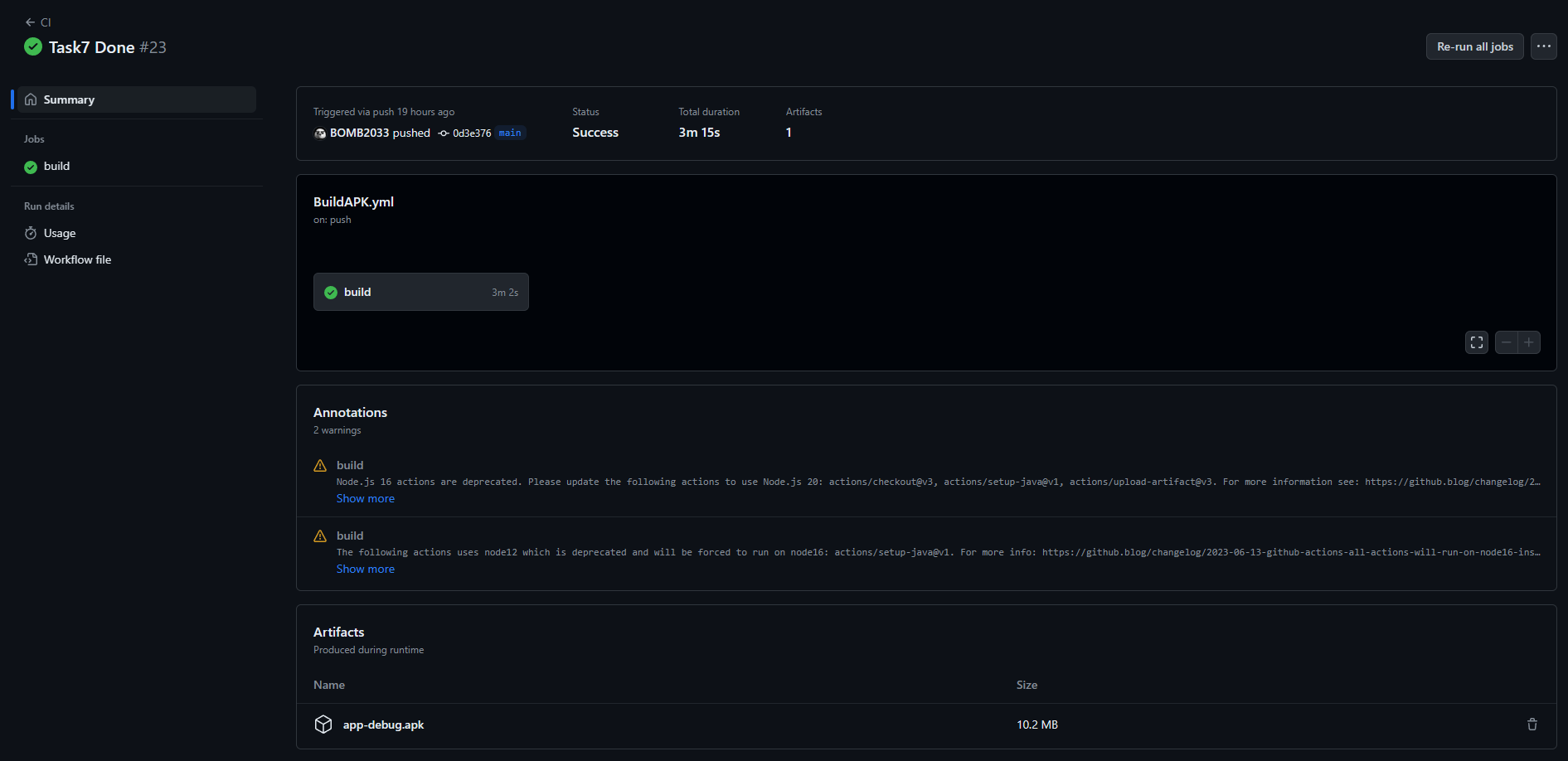


## Задание 7

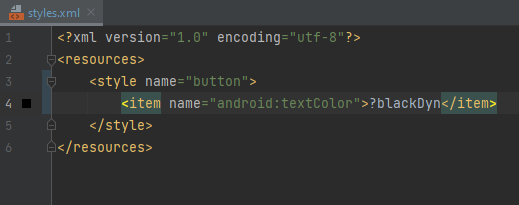
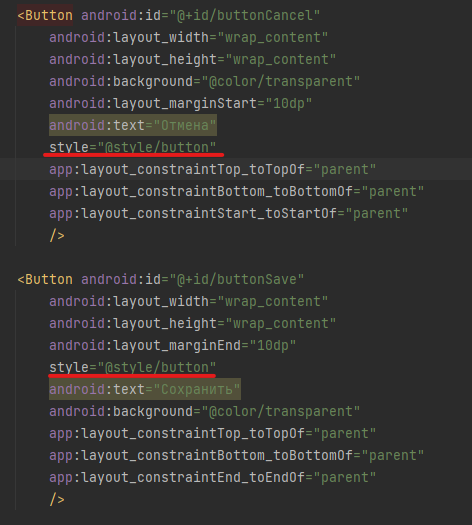
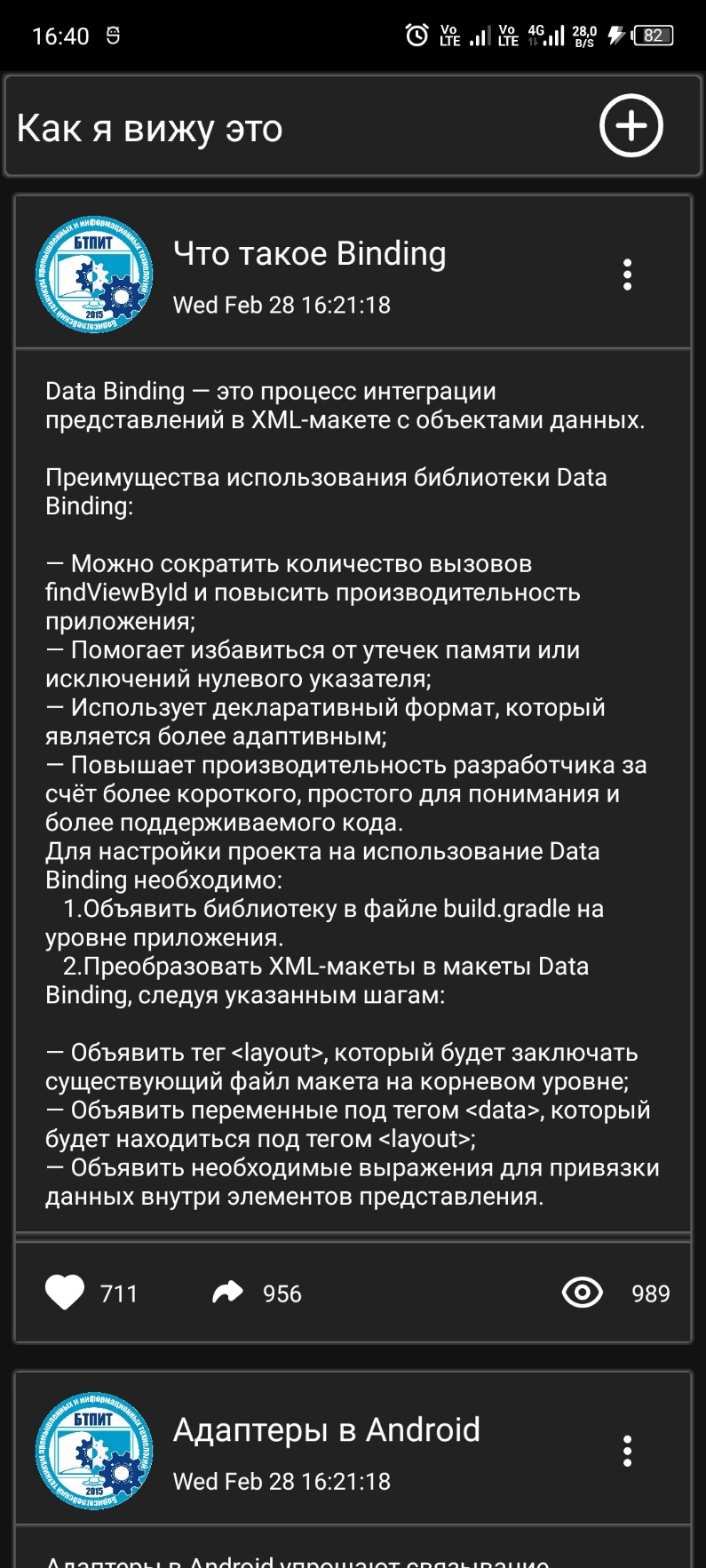
1. Функции CRUD:
   1. К майну подключил интерфейс:



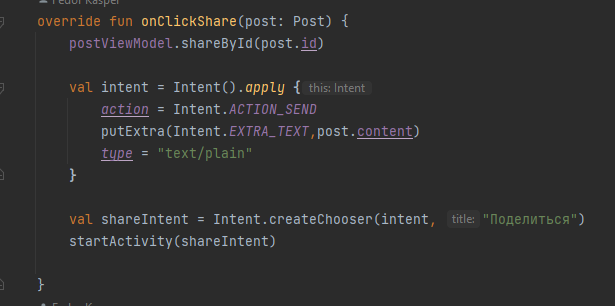
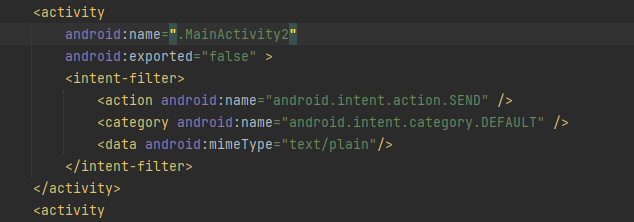
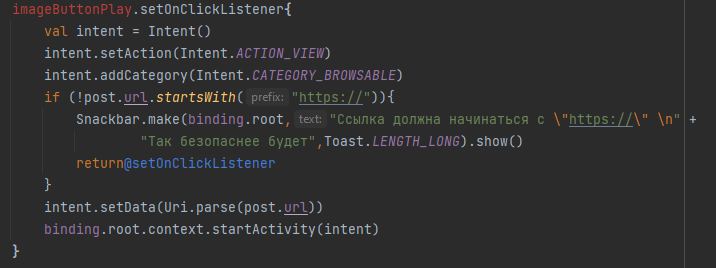
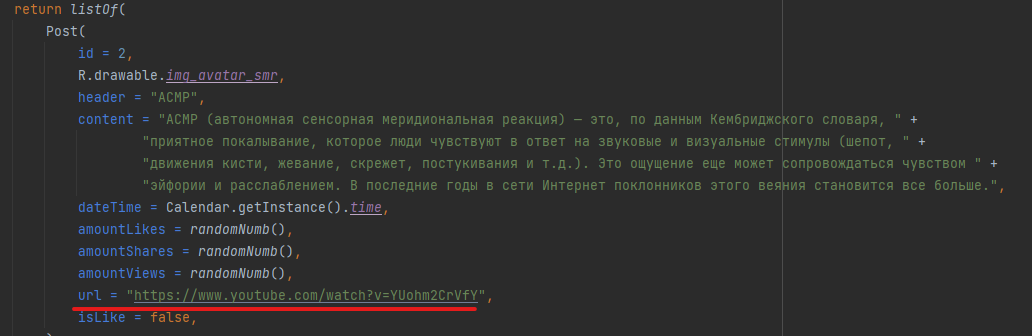
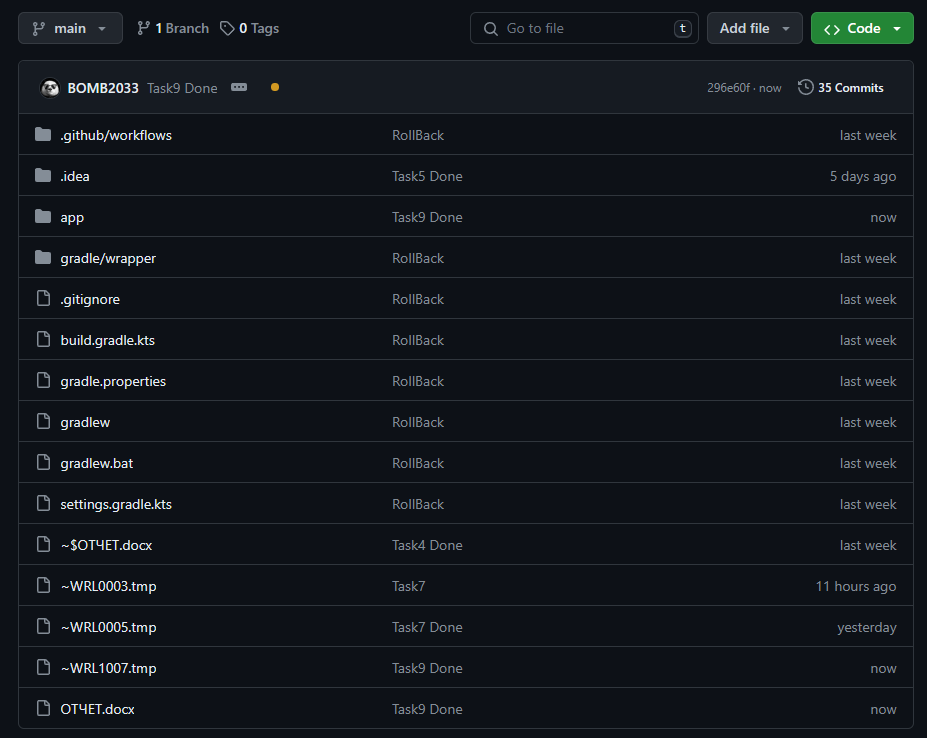
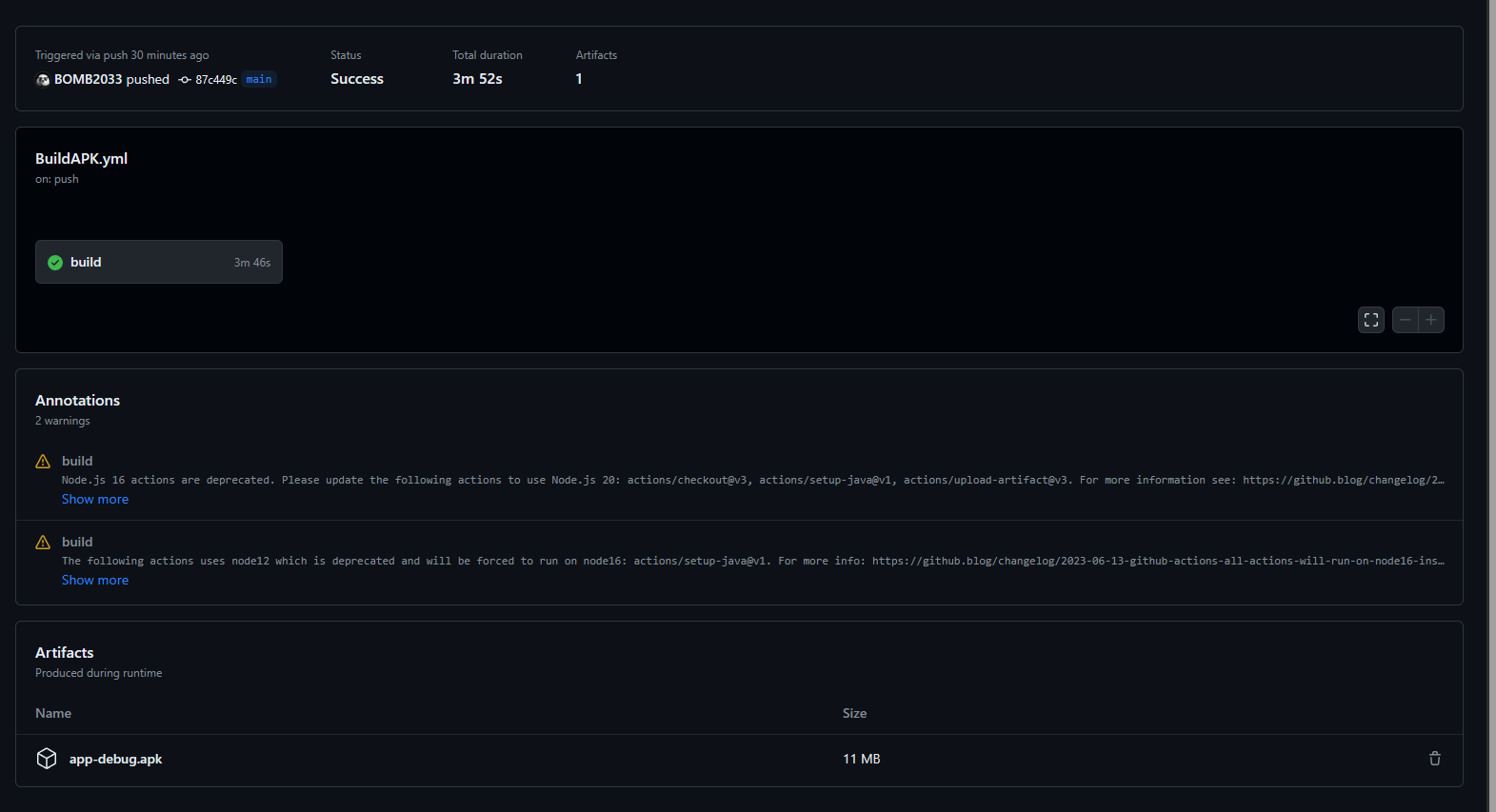
* 1. Интерфейс
  2. Функции в Майне: 
  3. postViewModel

1. Отправил на гит
2. И Action 

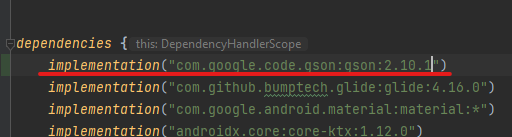
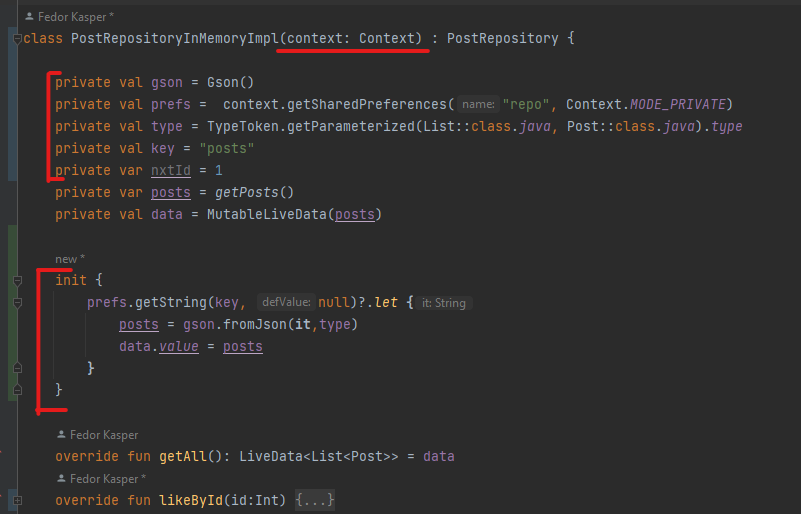
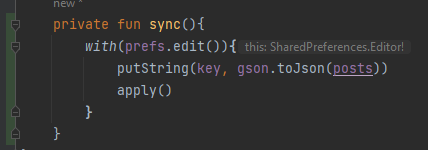
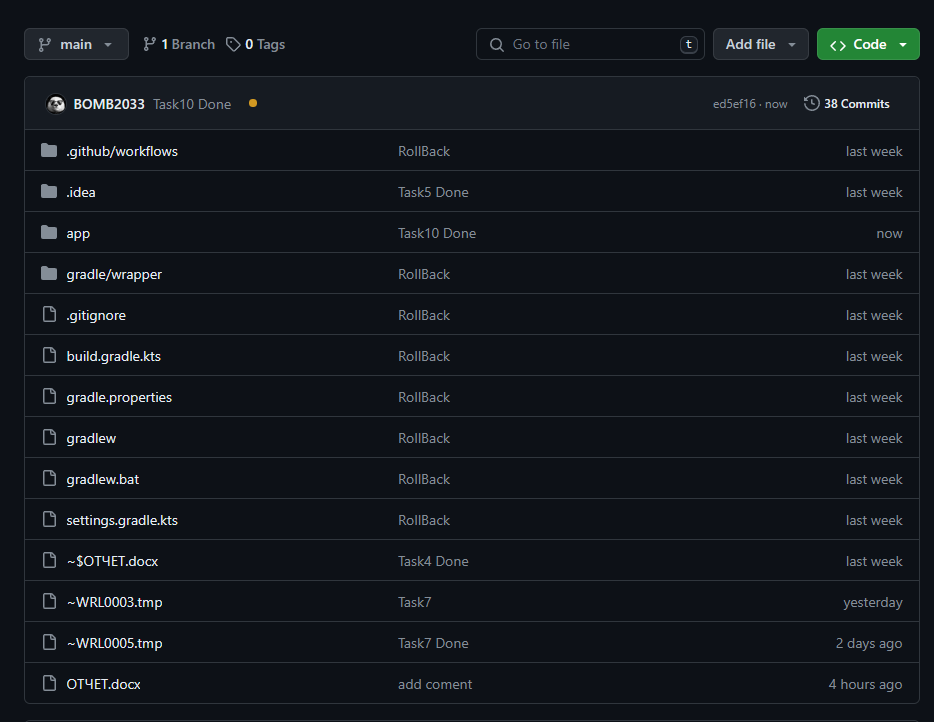
## Занятие 8

1. Создал стиль
2. Добавил его на кнопки
3. Так же поменял тему приложения 
4. И сделал приложение красивым

## Занятие 9

1. Сделал интеграцию с другими приложениями
   1. Отправка
   2. Приём
2. Ссылка на видео: 
3. Отправил на GitHub:
4. И GitHub Action: 

## Занятие 10

1. Добавил библиотеку Gson
2. Модифицировал файл Repository
3. И во все функции изменения поста добавил sync()
4. Отправил на GitHub
5. GitHub Action собрал всё