

4 x 4

1

Android research assignment

4 X 4

Android game + Firebase

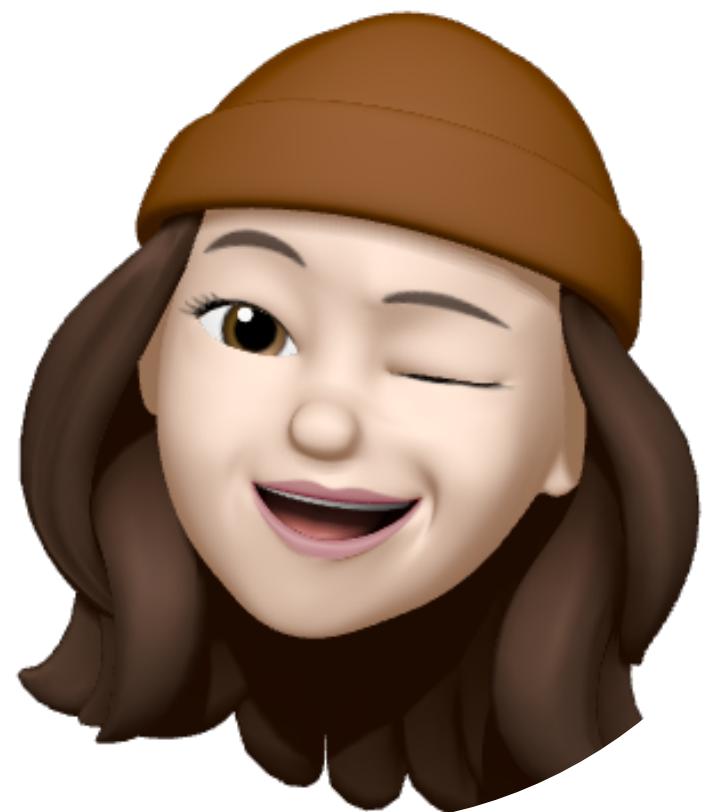
Android research assignment

4 x 4

TEAM



Kornkanok Sangwichien
N3570



Anna Paszcza
N2305

What are we
going to talk
today?

- Idea
- ReactiveX
- Firebase
- Our game

Idea

01 —————



Idea



Idea



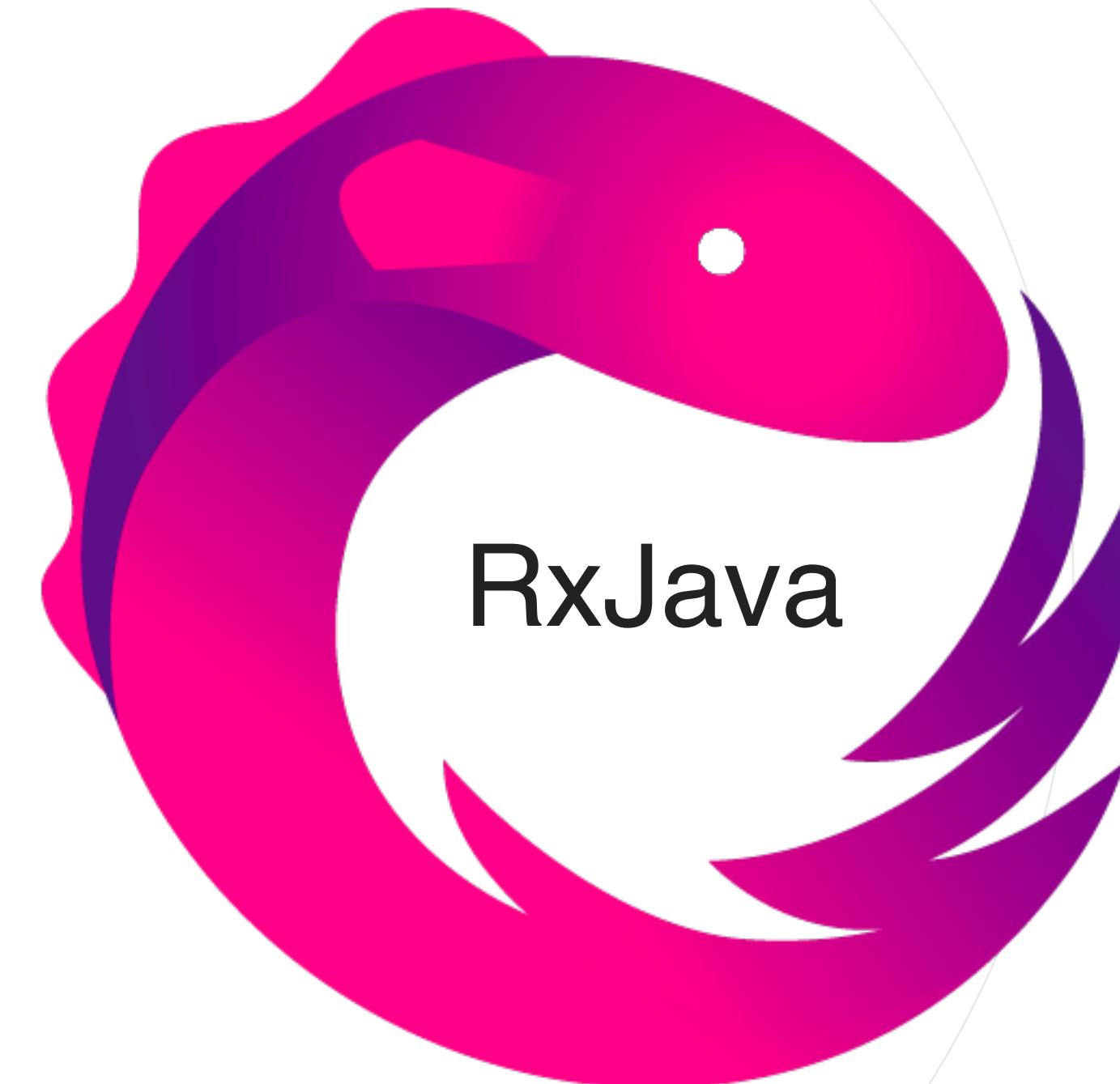
available
space

puzzle

Idea

ReactiveX

02 —



“ Java VM implementation
of Reactive Extensions ”



Reactive Extension(ReactiveX)
as a library for composing
asynchronous and event-based
programs by using observable
sequences.

Reactivex

Exam



Muta

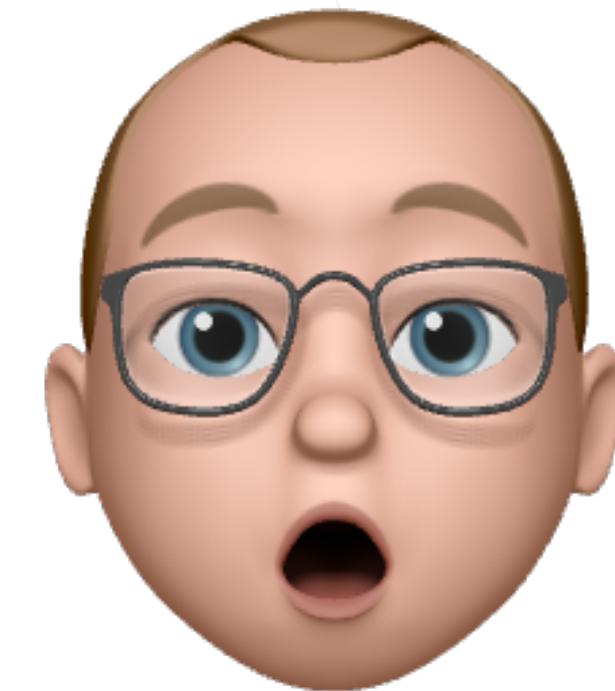
!@#\$%^&



Anna



Michał



Teacher

Exam

Cheating...



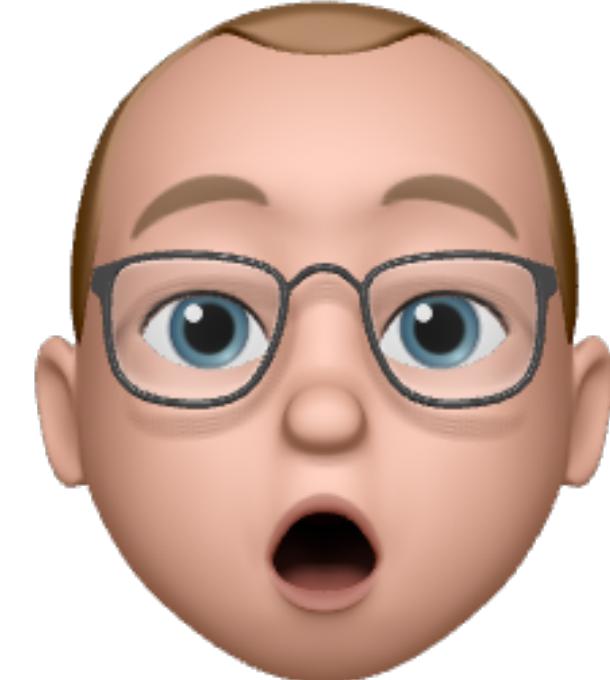
Muta



Anna



Michał



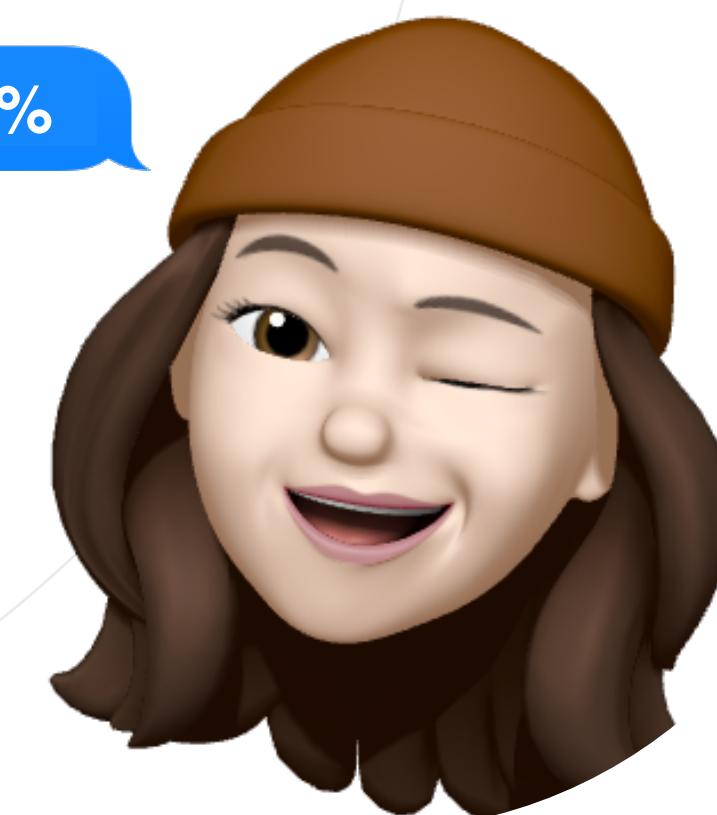
Teacher

Reactivex

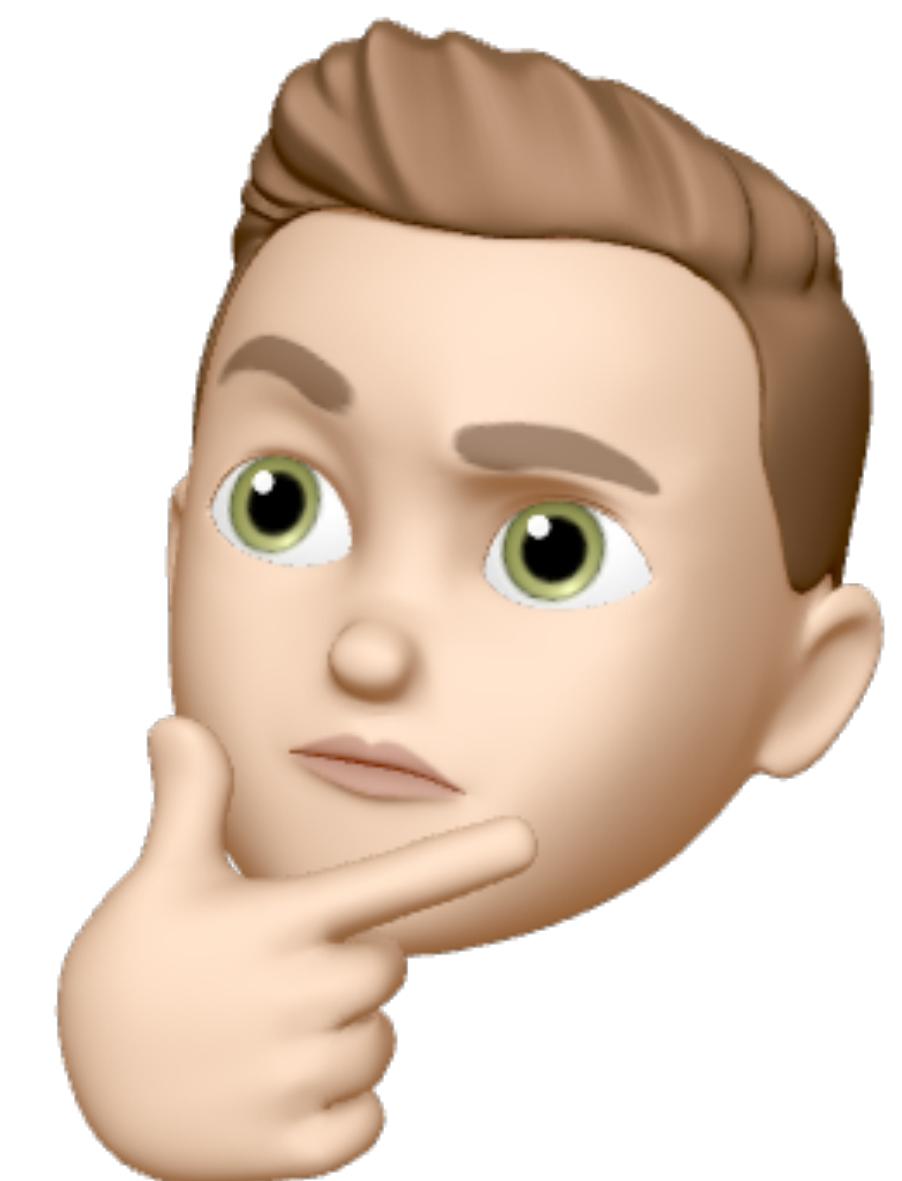


Muta

!@#\$%^&
!@#\$%



Anna



Michał



Teacher

Exam

Michał watching Anna closely

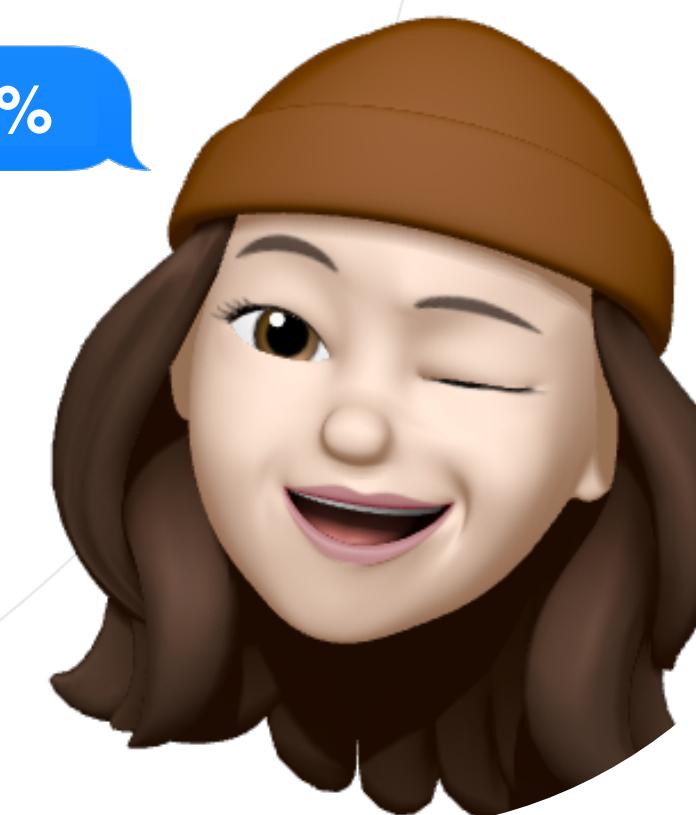
Exam

Michał reporting to teacher as the events took place

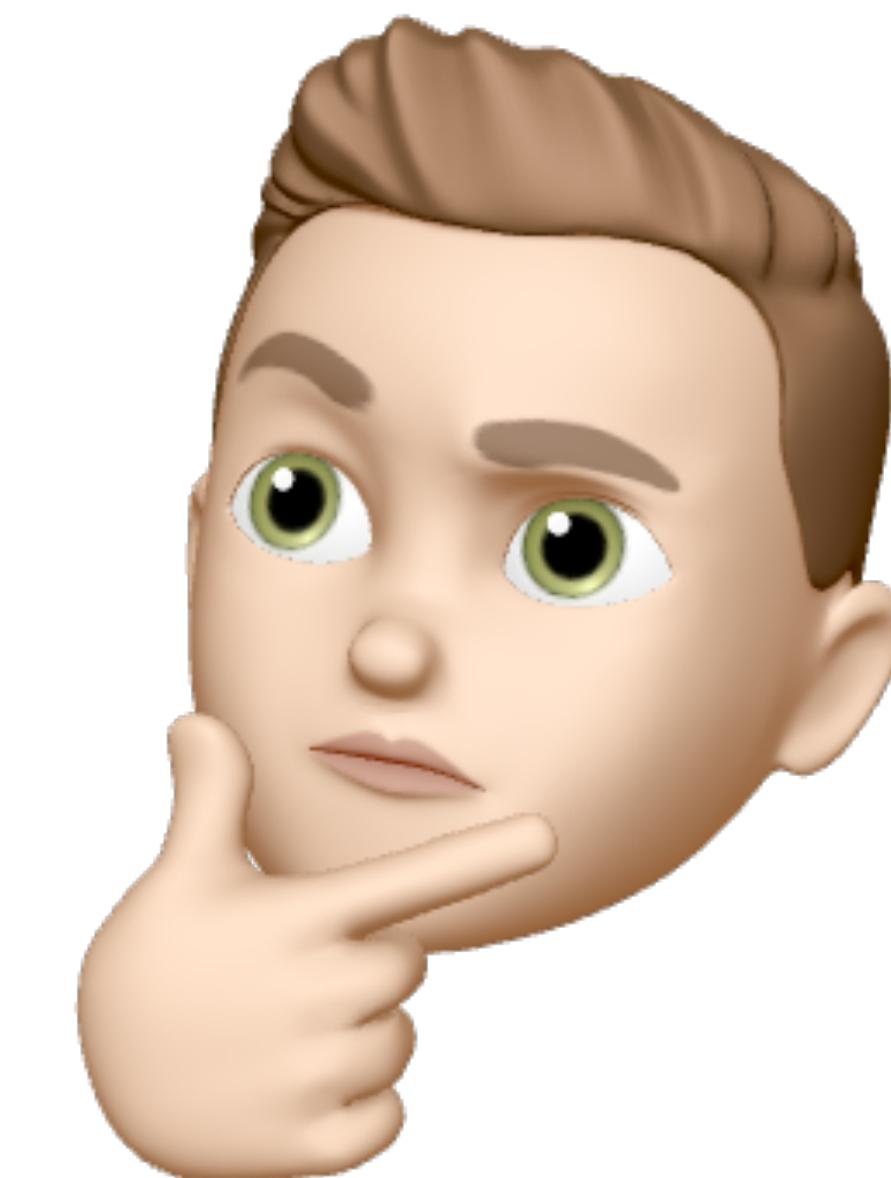


Muta

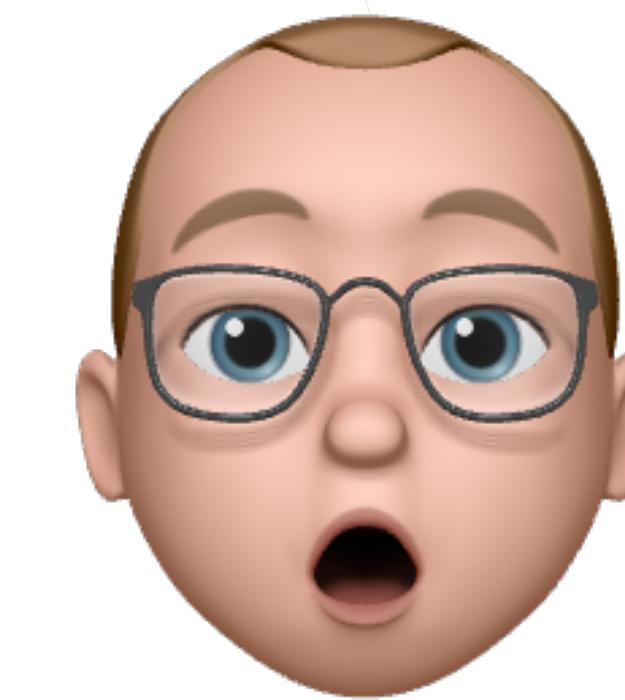
!@#\$%^&
!@#\$%



Anna



Michał



Teacher

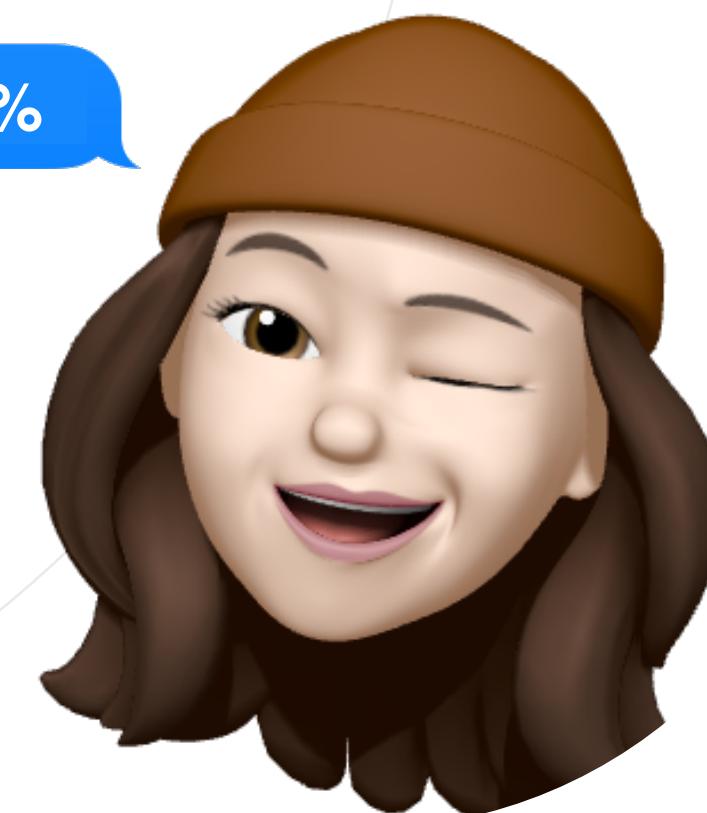
Exam

Observer

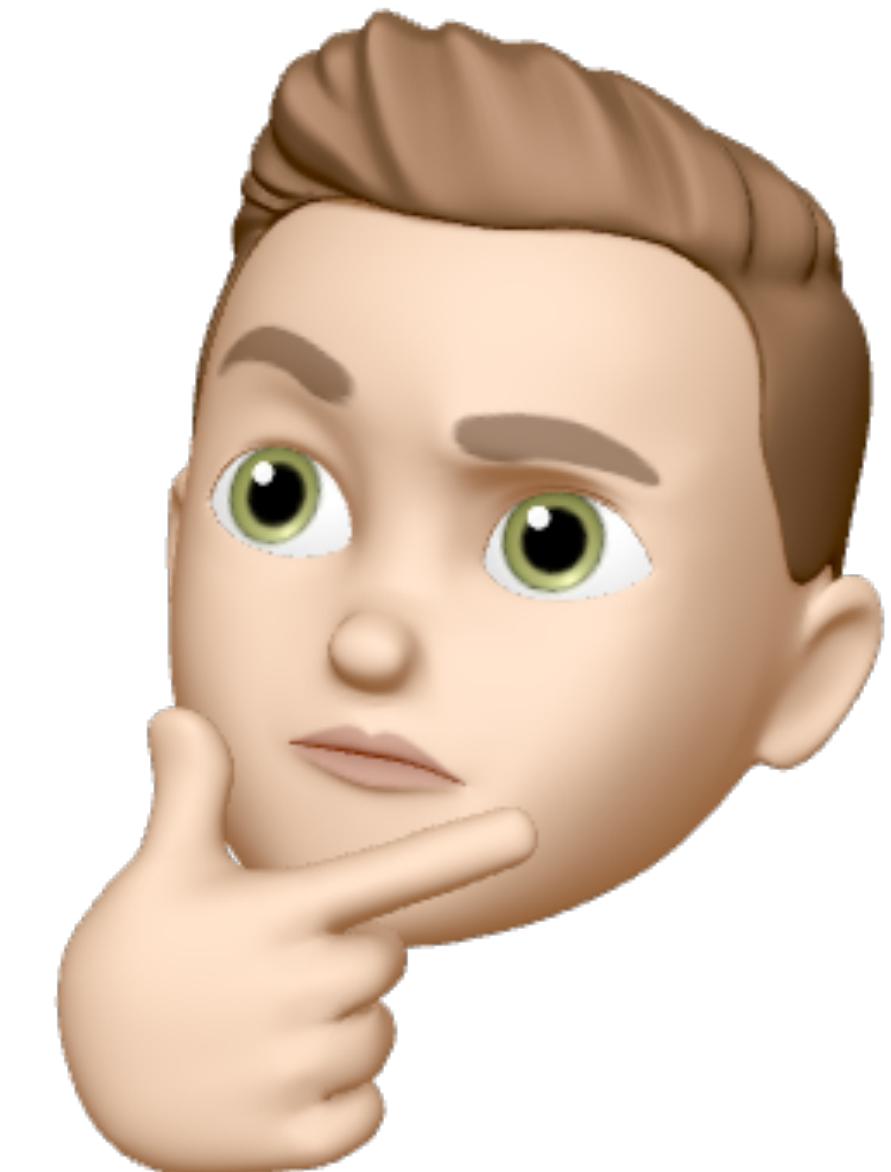


Muta

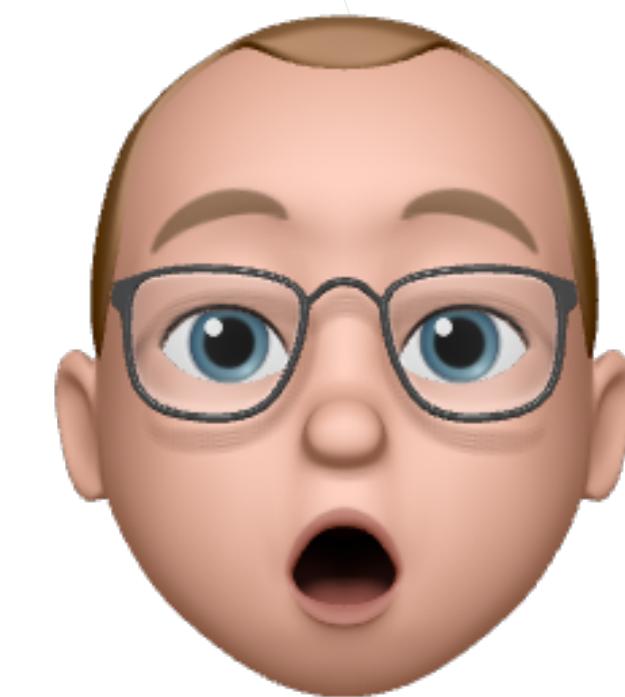
!@#\$%^&
!@#\$%



Anna



Michał



Teacher

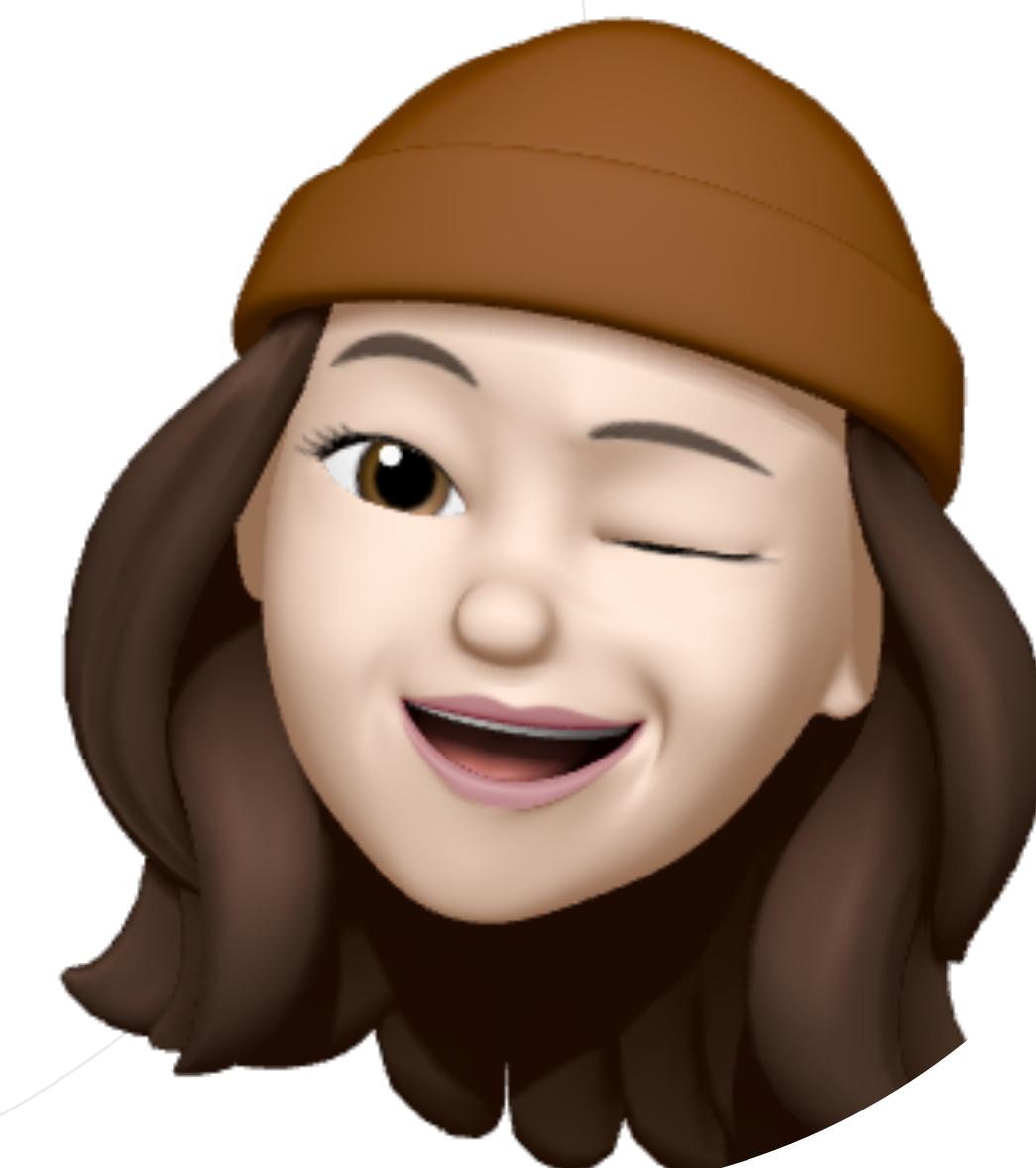
4 x 4

16

Reactivex



Muta



Anna



Michał



Teacher

Data

Exam

Observer

Exam

Michał was told to watch and report Anna as her state changes, and he's to make a callback to whoever is listening to him (the teacher)

Data



Muta

Observer



Anna



Michał



Teacher

Reactivex

```
val anna: Observable<Mistakes>
val michal: Subscriber
Anna = Observable.just(
    Mistakes("Wrong change"),
    Mistakes("Dropped change"),
    Mistakes("Wrong change")
)
michal = anna.subscribe({ whatHappened -> reportToTeacher(whatHappened) })
```



Muta



Anna Data



**Michał
Observer**



Teacher

4 x 4

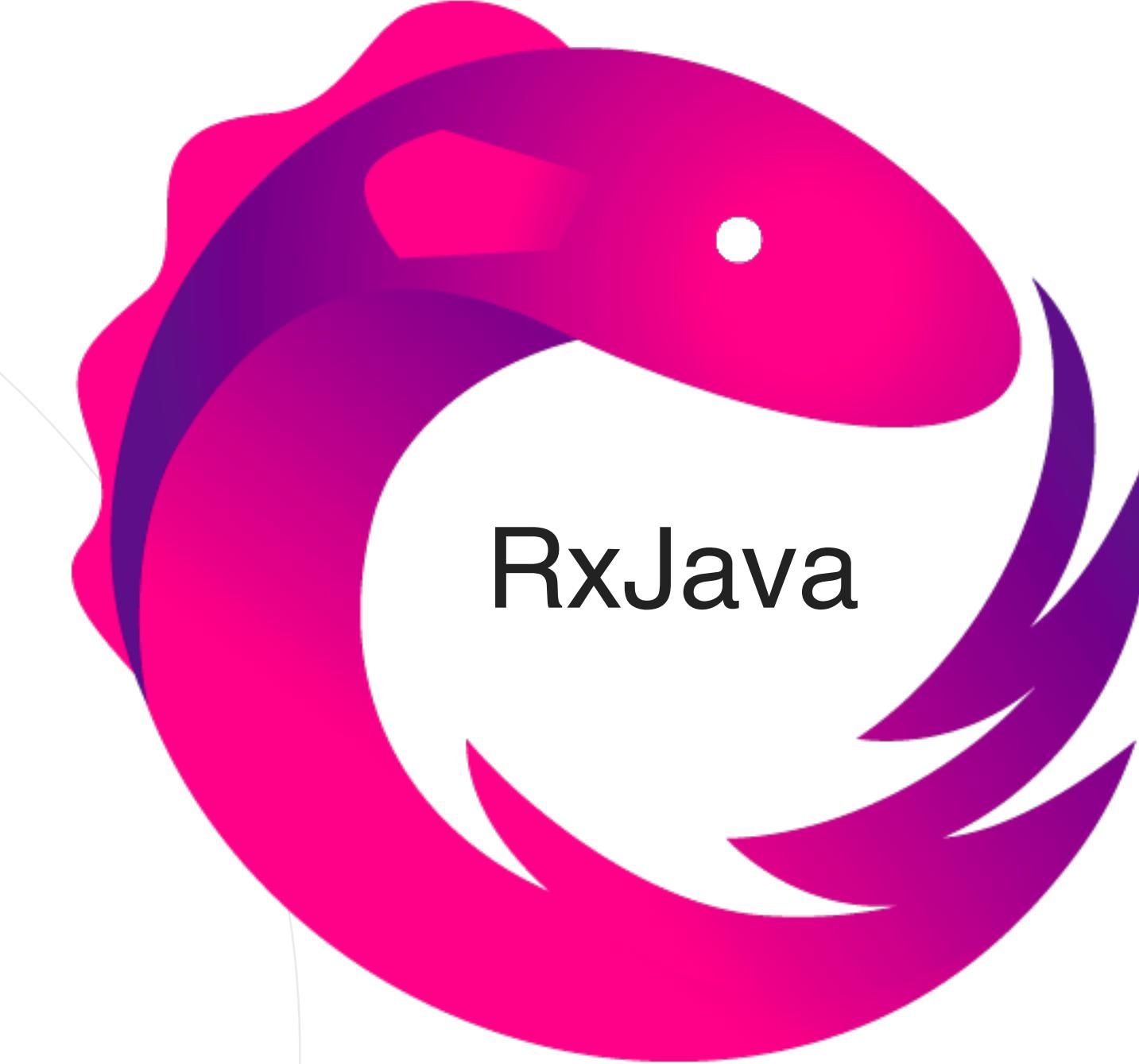
Reactivex



<http://reactivex.io/documentation>

4 x 4

ReactiveX



+



Android research assignment

20

Reactivex

```
dependencies {  
    ...  
    implementation "io.reactivex.rxjava2:rxjava:2.2.7"  
    implementation "io.reactivex.rxjava2:rxandroid:2.1.1"  
}
```

```
mergeClicks().switchMap { it: Boolean
    if (it) timerObservable() ^switchMap
    else Observable.empty() ^switchMap
} .subscribe(text_view_countdown::setText)
    .let(disposable::add)

private fun mergeClicks(): Observable<Boolean> =
listOf(
    button_start.clicks().map { true },
    button_reset.clicks().map { false })
    .merge()
    .doOnNext(::buttonStateManager)

private fun timerObservable(): Observable<String> =
Observable.interval(initialDelay: 0, period: 1, TimeUnit.SECONDS)
    .takeWhile { it <= MAXIMUM_STOP_WATCH_LIMIT }
    .map(timeFormatter)
    .observeOn(AndroidSchedulers.mainThread())
    .doOnComplete { buttonStateManager(boolean: false) }
```

Firebase

03 —

set of tools offered by
GOOGLE to build scalable
applications in the cloud



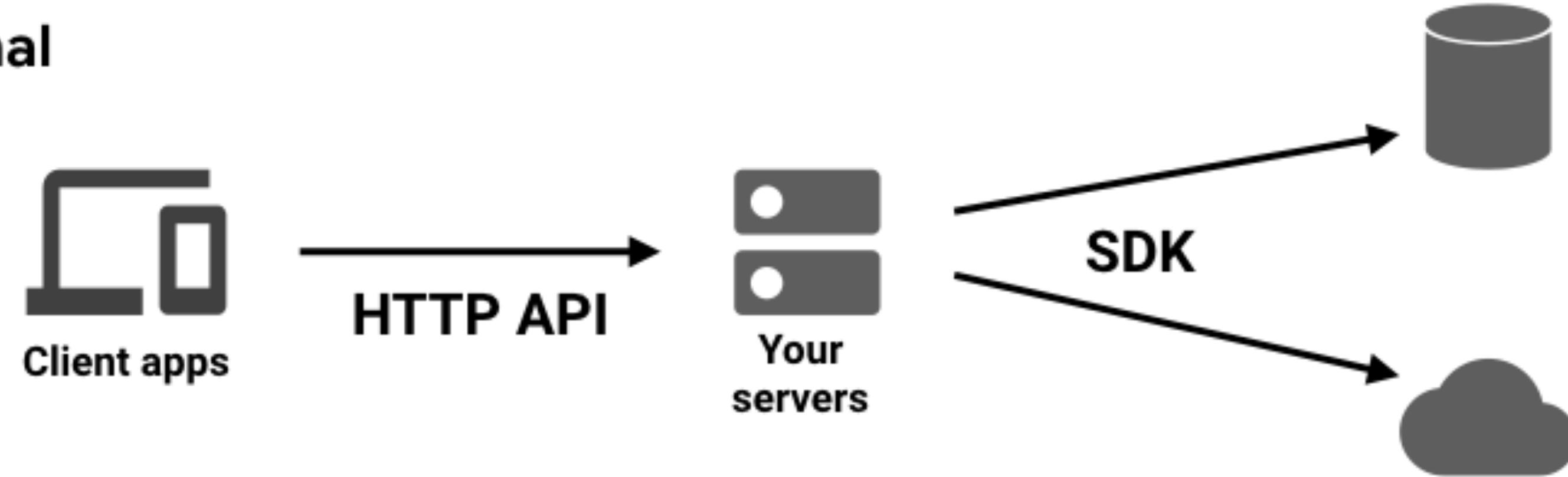
Firebase is Google's mobile application development platform that helps you build, improve, and grow your app.

REALTIME

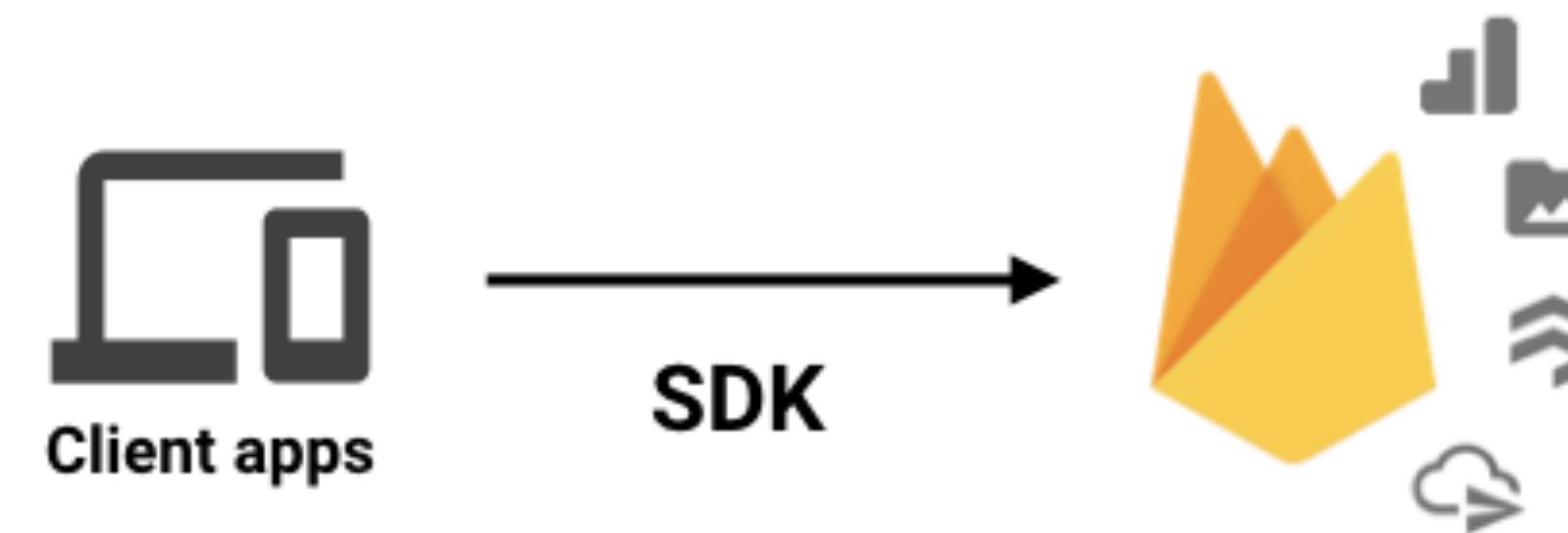


FIREBASE

Traditional



Firebase



Your Firebase projects



Add project

AndroidGame

androidgame-95656



WeatherApp

weatherapp-6f3c9



Explore a demo project

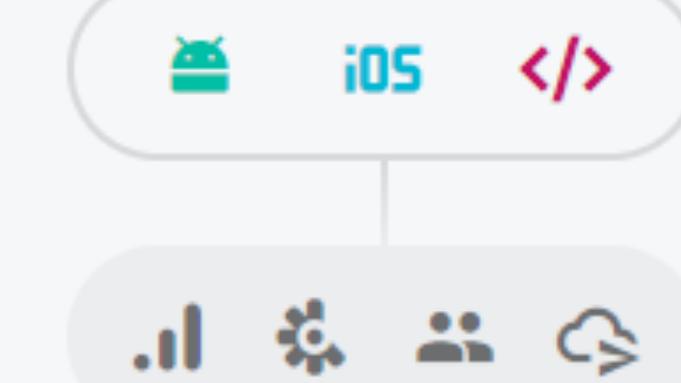


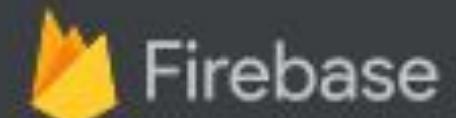
iOS

Firebase projects are
containers for your apps

Apps in a project share features like
Realtime Database and Analytics

Q [Learn more](#)





Firebase gives you the tools and infrastructure from Google to help you develop, grow and earn money from your app. [Learn more](#)

► **Analytics**

Measure user activity and engagement with free, easy, and unlimited analytics. [More info](#)

► **Cloud Messaging**

Deliver and receive messages and notifications reliably across cloud and device. [More info](#)

► **Authentication**

Sign in and manage users with ease, accepting emails, Google Sign-In, Facebook and other login providers. [More info](#)

▼ **Realtime Database**

Store and sync data in realtime across all connected clients. [More info](#)

► [Save and retrieve data](#)

► **Storage**

Store and retrieve large files like images, audio, and video without writing server-side code. [More info](#)

► **Remote Config**

Customize and experiment with app behavior using cloud-based configuration parameters. [More info](#)

► **Test Lab**

Test your apps against a wide range of physical devices hosted in Google's cloud. [More info](#)

► **App Indexing**

Get your app content into Google Search. [More info](#)

► **Dynamic Links**

Create web URLs that can be shared to drive app installs and deep-linked into relevant content of your app. [More info](#)

Save and retrieve data

Our cloud database stays synced to all connected clients in realtime and remains available when your app goes offline. Data is stored in a JSON tree structure rather than a table, eliminating the need for complex SQL queries.

[Launch in browser](#)

① Connect your app to Firebase

✓ Connected

② Add the Realtime Database to your app

✓ Dependencies set up correctly

③ Configure Firebase Database Rules

The Realtime Database provides a declarative rules language that allows you to define how your data should be structured, how it should be indexed, and when your data can be read from and written to. By default, read and write access to your database is restricted so only authenticated users can read or write data. To get started without setting up [Authentication](#), you can [configure your rules for public access](#). This does make your database open to anyone, even people not using your app, so be sure to restrict your database again when you set up authentication.

④ Write to your database

Retrieve an instance of your database using `getInstance()` and reference the location you want to write to.

```
// Write a message to the database
FirebaseDatabase database = FirebaseDatabase.getInstance();
DatabaseReference myRef = database.getReference("message");
```

FIREBASE

4 x 4

Your apps

Add app

Android apps

AndroidGame
com.example.androidgame

Download the latest config file

[google-services.json](#)

App ID ⓘ
1:1034946675171:android:b7a7a4825c245c77eda0d6

App nickname
AndroidGame 

Package name
com.example.androidgame

SHA certificate fingerprints ⓘ Type ⓘ

78:4d:06:a6:d9:a5:0e:a9:54:04:48:7d:2e:24:4b:9f:bd:57:ef:71	SHA-1
53:40:5e:0f:3b:6c:49:e2:5a:cb:78:84:58:73:97:76:38:76:65:80	SHA-1
85:35:7a:8c:21:b9:95:91:5a:f7:f5:da:26:48:d6:bd:93:d4:2a:b0:79:...	SHA-256

Add fingerprint

Remove this app

FIREBASE

4 x 4

Your apps

Android apps

 **AndroidGame**
com.example.androidgame

Add app

Download the latest config file

This file contains configuration details, such as keys and identifiers, for the services that you have just enabled.

google-services.json

App ID ②
1:1034946675171:android:b7a7a4825c245c77eda0d6

App nickname
AndroidGame 

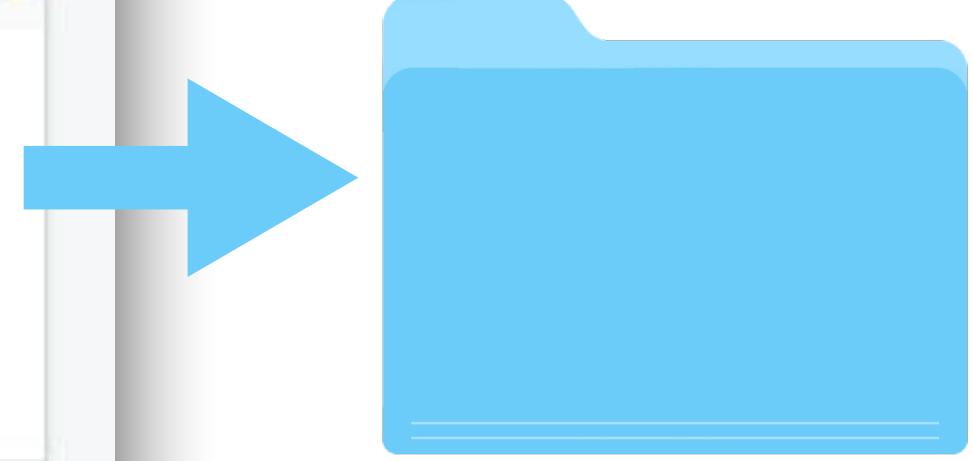
Package name
com.example.androidgame

SHA certificate fingerprints ② **Type** ②

78:4d:06:a6:d9:a5:0e:a9:54:04:48:7d:2e:24:4b:9f:bd:57:ef:71	SHA-1
53:40:5e:0f:3b:6c:49:e2:5a:cb:78:84:58:73:97:76:38:76:65:80	SHA-1
85:35 7a:8c:21:b9:95:91:5a:f7:f5:da:26:48:d6:bd:93:d4:2a:b0:79:..	SHA-256

Add fingerprint

Remove this app



ApplicationFolder

FIREBASE

```
implementation 'com.google.firebaseio:firebase-database:18.0.1'  
implementation 'com.google.firebaseio:firebase-analytics:17.2.0'
```

```
ref = FirebaseDatabase.getInstance().reference  
scoreReading(ref)
```

```
ref.child(scoreId).setValue(userScore)
```

```
fun scoreReading(reference: DatabaseReference){  
    reference.addValueEventListener(object : ValueEventListener {  
        override fun onDataChange(dataSnapshot: DataSnapshot) {  
  
            .  
            .  
            .  
  
            for (scoreSnapshot in dataSnapshot.children) {  
  
                .  
                .  
                .  
  
            }  
        }  
        override fun onCancelled(error: DatabaseError) {  
            // Failed to read value  
            println("Failed to read value.")  
        }  
    })  
}
```

FIREBASE

4 x 4

AndroidGame ▾

Database

Realtime Database ▾

Data Rules Backups Usage

The screenshot shows the Firebase Realtime Database interface. At the top, there is a URL bar with the address <https://androidgame-95656.firebaseio.com/>. Below the URL bar are three buttons: a plus sign (+), a minus sign (-), and a three-dot menu icon.

The main area displays a list of database nodes under the root node `androidgame-95656`. The nodes are listed vertically, each preceded by a small square icon. The first node, `-LtGd5aP7o_02NnBuLVY`, has a value of "0:2". The second node, `-LtGd78TBZbxJXHRkWWi`, has a value of "0:4" and is highlighted with a red rectangular selection box. The third node, `-LtGdB42EaOBne6ZnwP8`, has a value of "0:1". The fourth node, `-LtGdBUExgHEgL6GQrYr`, has a value of "0:1". The fifth node, `-LtGdB_COFDLZK6smgUW`, has a value of "0:1". The sixth node, `-LtGdBgd6D0die8MTvn8`, has a value of "0:1". The seventh node, `-LtGdBzQh0pFIW1GRN6E`, has a value of "0:1". The eighth node, `-LtGdC61Xc7JwB-QmJGN`, has a value of "0:1". The ninth node, `-LtGdPYSo6BW65NaO9Li`, has a value of "0:4". The tenth node, `-LtGdU_imHEMTdM5Jwhi`, has a value of "0:1". The eleventh node, `-LtGdUr-Wq9qmBphCk5r`, has a value of "0:1". The twelfth node, `-LtGdUyC8NlihMcNfe4V`, has a value of "0:1". The thirteenth node, `-LtGdV2dy-rFw9JmfwHY`, has a value of "0:1". The fourteenth node, `-LtGdVnH7KHKy3z-LZHe`, has a value of "0:2". The fifteenth node, `-LtGdWNypFCu7De24Glf`, has a value of "0:1". The sixteenth node, `-LtGdXUYuPijZIB2-fHV`, has a value of "0:1". The seventeenth node, `-LtGdYdKHEcnzzJkQ9fd`, has a value of "0:1". The eighteenth node, `-LtGdYfVZllqfGDls8uS`, has a value of "0:1". The nineteenth node, `-LtGdZk-BPJVIHF2Z1g`, has a value of "0:1". The twentieth node, `-LtGd_pStTN6C1pAQhlv`, has a value of "0:1". The twenty-first node, `-LtGdb6CXCSXH2LigNL`, has a value of "0:1".

Our game

04 —

OUR GAME

4 x 4



**ASK ME ANYTHING,
SUGGEST ME AN IDEA**

**ASK ME ANYTHING,
SUGGEST ME AN IDEA**