

Розробка Android бібліотеки з графіками зміни курсу валют та криптовалют в світі

Виконав:

Джосан Володимир Олегович, БП ІПЗ-4

Науковий керівник:

Борозенний Сергій Олександрович

Актуальність теми

- Актуальність інформації про валюти та криптовалюти у сучасному світі
- Мобільні додатки для управління фінансами та цифровими активами актуальні, оскільки часто потрібно робити рішення “на ходу”
- Багато Android-додатків на фінансову та криптовалютну тематику

Мета дипломної роботи та постановка задачі

Мета дипломної роботи:

Розробка Android-бібліотеки, що надаватиме інструменти для візуалізації зміни курсу валют та криптовалют в світі у вигляді графіків, а також можливість роботи з цими даними через надане бібліотекою API.

Постановка задачі дипломної роботи:

1. Проаналізувати, як саме можуть бути представлені різні статистичні дані, пов'язані зі зміною курсу валют та криптовалют. Обрати та використати публічні API для валют та криптовалют.
2. Розглянути і дослідити бібліотеки Retrofit та RxJava та технологію WebSocket для створення Android-бібліотеки з використанням публічних API для валют та криптовалют.
3. Створити Android-бібліотеку для візуалізації статистичних даних про зміни курсів валют та криптовалют.

Використання публічного API для валют та криптовалют



RESTful API

- Інформація про усі символи
- Історія свічкових даних (candlesticks)

WebSocket API

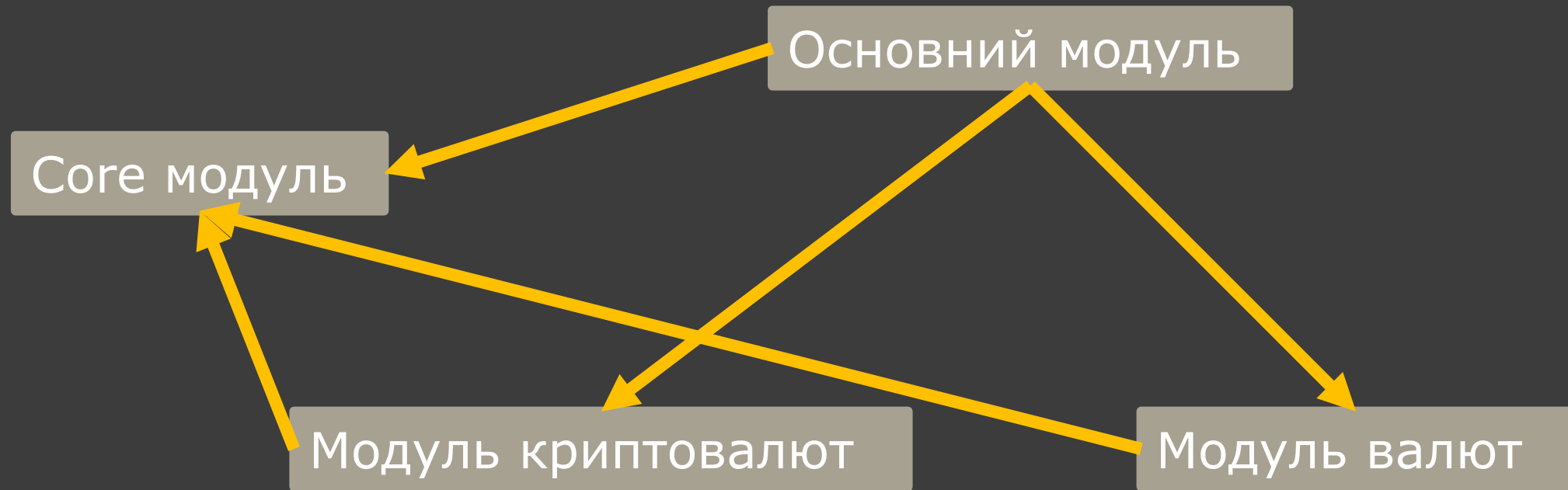
- Потoki оновлень свічкових даних (candlestick streams)

exchangerate.host

RESTful API

- Інформація про усі символи
- Історія курсів валют протягом певного періоду
- Коливання курсу валют протягом певного періоду

Архітектура бібліотеки



Використані технології та бібліотеки:

Retrofit

Retrofit

```
class BinanceRetrofit {  
    companion object {  
        private const val BASE_URL = "https://api.binance.com/"  
  
        fun makeRetrofitService() : BinanceRetrofitService {  
            val client = OkHttpClient.Builder()  
                .addInterceptor(NoSpamInterceptor())  
                .addInterceptor(WeightLimitInterceptor())  
                .build()  
            return Retrofit.Builder()  
                .baseUrl(BASE_URL)  
                .addConverterFactory(GsonConverterFactory.create(GsonBuilder().setLenient().create()))  
                .addCallAdapterFactory(RxJava3CallAdapterFactory.create())  
                .client(client)  
                .build()  
                .create(BinanceRetrofitService::class.java)  
        }  
    }  
}
```

```
interface BinanceRetrofitService {  
  
    @GET("/api/v3/exchangeInfo")  
    fun getExchangeInfo() : Single<ExchangeInfoResponse>  
  
    @GET("/api/v3/klines")  
    fun getCandlestickHistory(  
        @Query("symbol") symbol: String,  
        @Query("interval") interval: String,  
        @Query("limit") limit: Int,  
        @Query("endTime") startTime: Long? = null,  
        @Query("endTime") endTime: Long? = null  
    ) : Single<List<List<Any>>>  
}
```

Використані технології та бібліотеки: RxJava



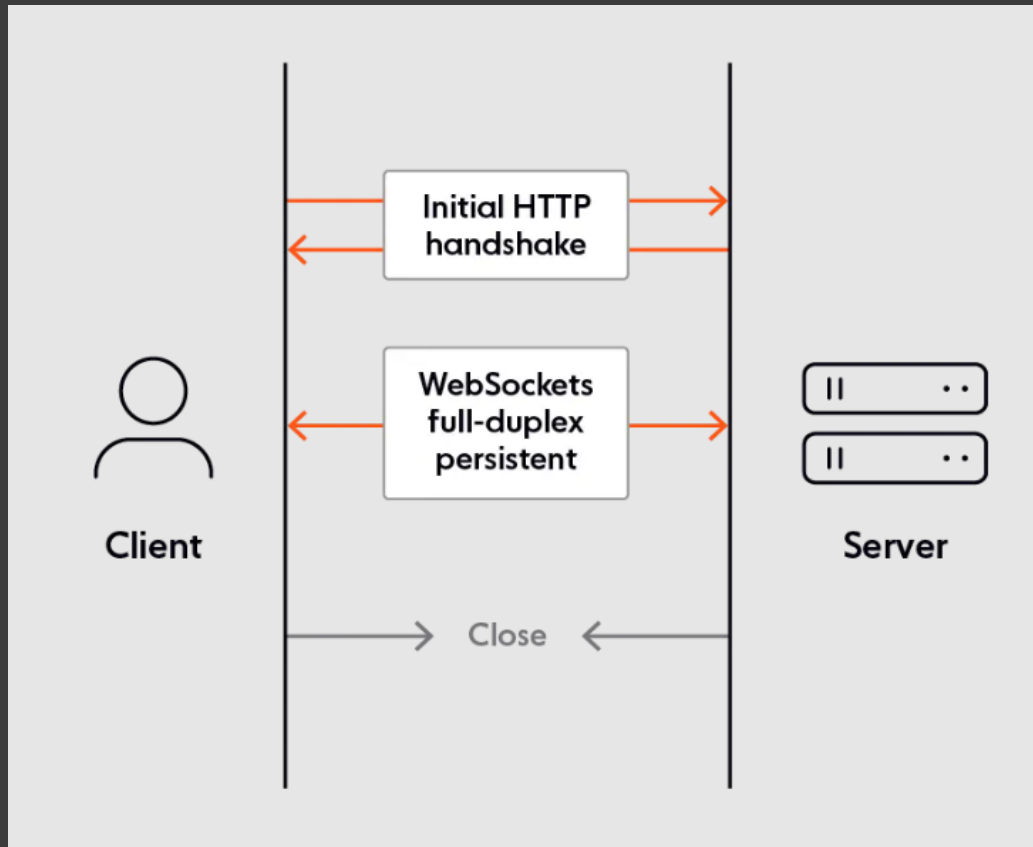
```
protected val candlestickUpdatesDataProcessor = DataProcessor.create<GraphUpdate<T>>(EmptyGraph())
```

```
candlestickUpdatesDataProcessor.onNext(HistoryUpdate(response.history, response.endOfHistory))
```

```
fun getGraphUpdates(): Flowable<GraphUpdate<T>> {  
    return candlestickUpdatesDataProcessor.toFlowable()  
}
```

```
addDisposable(  
    widgetModel.getGraphUpdates()  
        .observeOn(AndroidSchedulers.mainThread())  
        .subscribe(this::updateGraph)  
)
```

Використані технології та бібліотеки: WebSocket



+



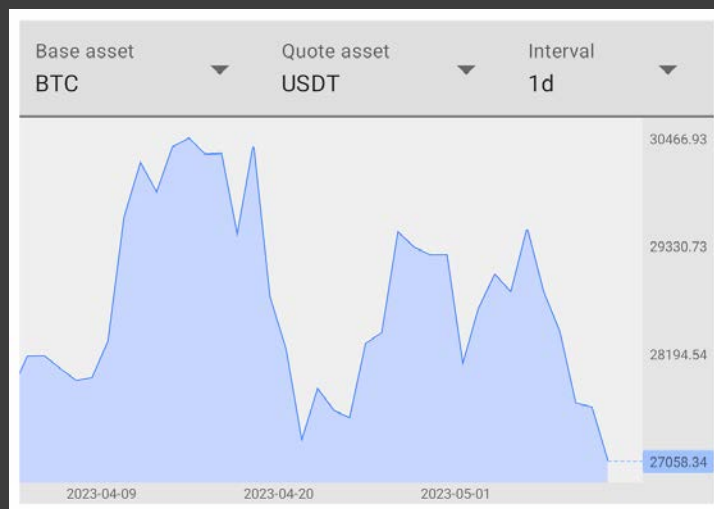
Функціонал бібліотеки: Графіки

- Класи графіків наслідують базовий клас *View*
- Для графіків перевизначені наступні методи:
 - *fun onDraw(...)* – для малювання графіку
 - *fun onTouchEvent(...)* – для налаштування інтерактивності графіку
 - *fun onSizeChanged(...)* – для отримання нових значень довжини та ширини ui-елементу

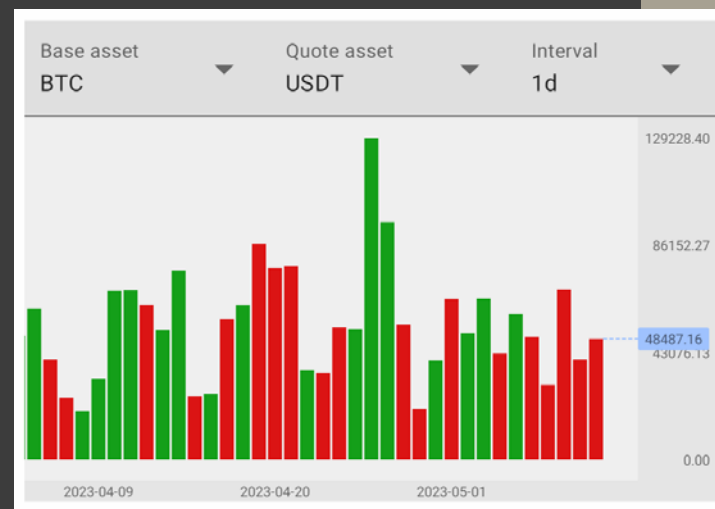
Функціонал бібліотеки: Графіки для криптовалют



Cryptocurrency
Candlestick Graph

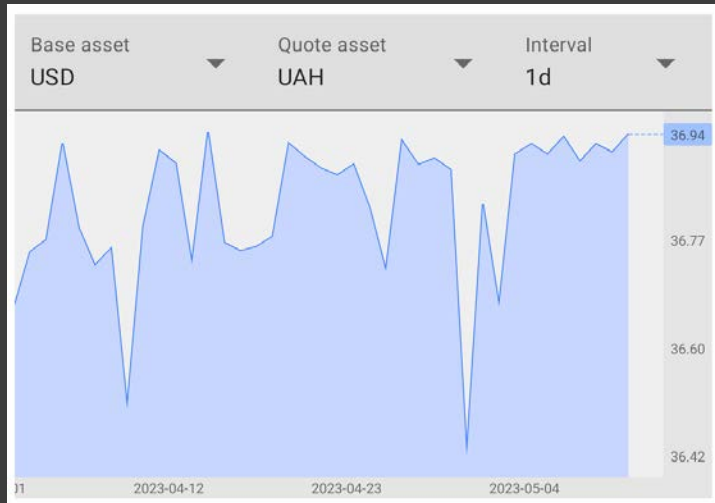


Cryptocurrency
Line Graph

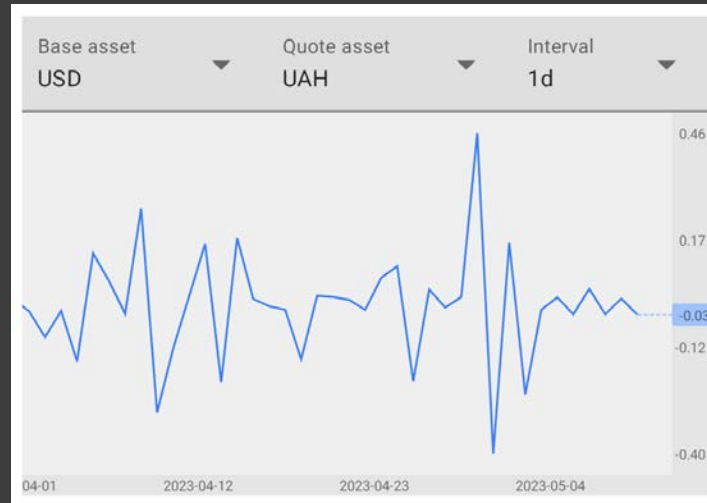


Cryptocurrency
Volume Graph

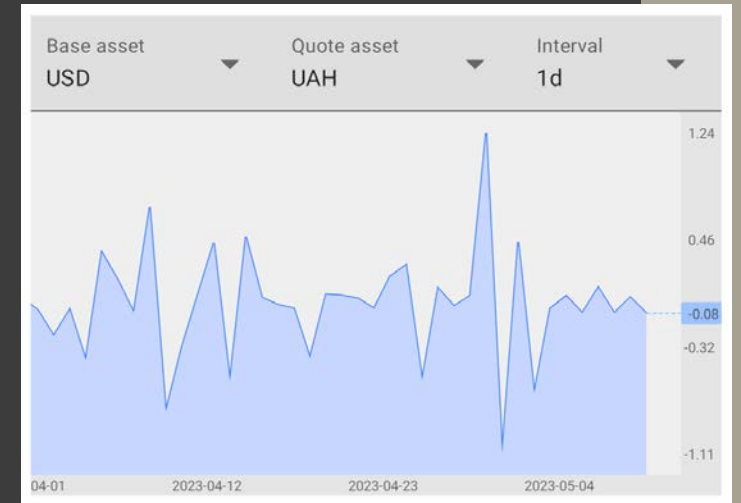
Функціонал бібліотеки: Графіки для валют



Currency Price
History Graph



Currency Fluctuation
History Graph
(Price)



Currency Fluctuation
History Graph
(Percents)

Функціонал бібліотеки: Вибір символу та інтервалу

Для кожного графіка можна вибирати:

- Базовий символ – символ, для якого буде побудовано графік
- Ціновий символ – символ, яким вимірюється вартість базового символу
- Інтервал – проміжок часу, для якого буде побудований графік



Функціонал бібліотеки: Інтерактивність графіків

З графіками, які надає розроблена бібліотека, можна взаємодіяти:

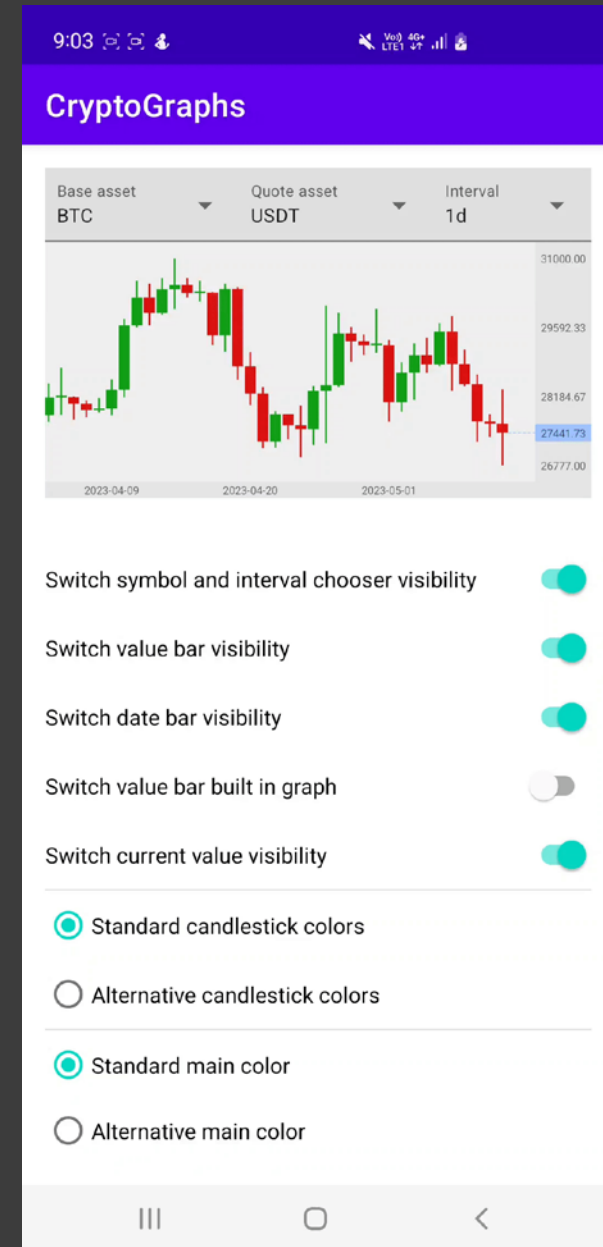
- Гортати (щоб побачити старішу статистику)
- Приближати/віддаляти графік
- Переходити до початку графіку



Функціонал бібліотеки: Кастомізація графіків

Кожен графік можна кастомізувати:

- Змінювати видимість елементів віджету:
 - панелі управління символами та інтервалом
 - вертикальної та горизонтальної осей (вісь значення та вісь часу)
 - вказівник поточного значення
- Змінювати спосіб відображення осі значень (вбудована або невбудована в графік)
- Змінювати кольори графіку
- Змінювати кількість позначок значень графіку на вертикальній осі значень



Функціонал бібліотеки: API криптовалют та валют

API для криптовалют:

- Отримати усі закешовані символи
- Отримати усі символи асинхронно
- Отримати історію змін статистики криптовалют з певним інтервалом:
 - Історія курсу у свічкових даних (candlestick)
 - Історія курсу криптовалюти (лише ціна)
 - Історія змін об'єму торгів
- Отримувати підтримувані інтервали

API для валют:

- Отримати усі закешовані символи
- Отримати усі символи асинхронно
- Отримати історію змін курсу валют з певним інтервалом
- Отримати значення коливань валют між двома датами
- Отримати історію коливань курсу валют (ціна та відсоток) з певним інтервалом
- Отримувати підтримувані інтервали

Висновки та рекомендації для подальшої розробки

Рекомендації для подальшої розробки:

1. Використання більш просунутого API для валют.
2. Підтримка оновлень у реальному часі графіків зі статистикою валют.
3. Додавання більшої кількості типів графіків та інших віджетів.
4. Розширення функціоналу API бібліотеки для криптовалют та валют.

Висновки:

Розроблена бібліотека надає користувачам інтерактивні, кастомізовані графіки для візуалізації зміни курсу та іншої статистики валют та криптовалют, а також API для роботи з цими даними.

У ході розробки даної бібліотеки було успішно використано технології та бібліотеки, зазначені у задачі дипломної роботи: Retrofit, RxJava та WebSocket.

Дякую за увагу!