

Разработка трекера по криптовалюте.

ВЫПОЛНИЛ
МЕРКУЛОВ Д.В
11-ИТ

Актуальность проекта.

- Актуальность моего проекта заключается в том, что пользователю не надо проходить регистрации и иные требования для просмотра цен криптовалют. Ему достаточно лишь просто открыть приложение на телефоне и ввести ему нужный курс. Интерфейс я выбрал самый обычный для простоты использования.
-

- **Целью данного проекта** является получение навыков, изучение новой информации, освоение языка Java.

Задачи:

- 1)Изучить возможности среды разработки Android Studio при проектировании и разработке программы;
 - 2)Рассмотреть виды и классификацию приложений на андроид и особенности их интерфейсов;
 - 3)Спроектировать и реализовать приложение "CryptoTracker";
 - 4)Провести тестирование разработанного приложения.
-

Обзор предметной области.

- В современном мире существует большое количество приложений для криптовалют. начиная от обычного трекера заканчивая самыми требовательными софтами на любой вкус. Я подхожу к реализации моего проекта с самым простым способом; написать код, сделать простой и понятный интерфейс, сделать приложение удобным.
-

Функциональные требования:

1. Поиск курса криптовалюты.
2. Использование API ключа для отслеживания курса.

Нефункциональные требования:

1. Простой интерфейс.
2. Простота использования.
3. Быстрое реагирование приложения.
4. Скромная производительность.

Сравнение

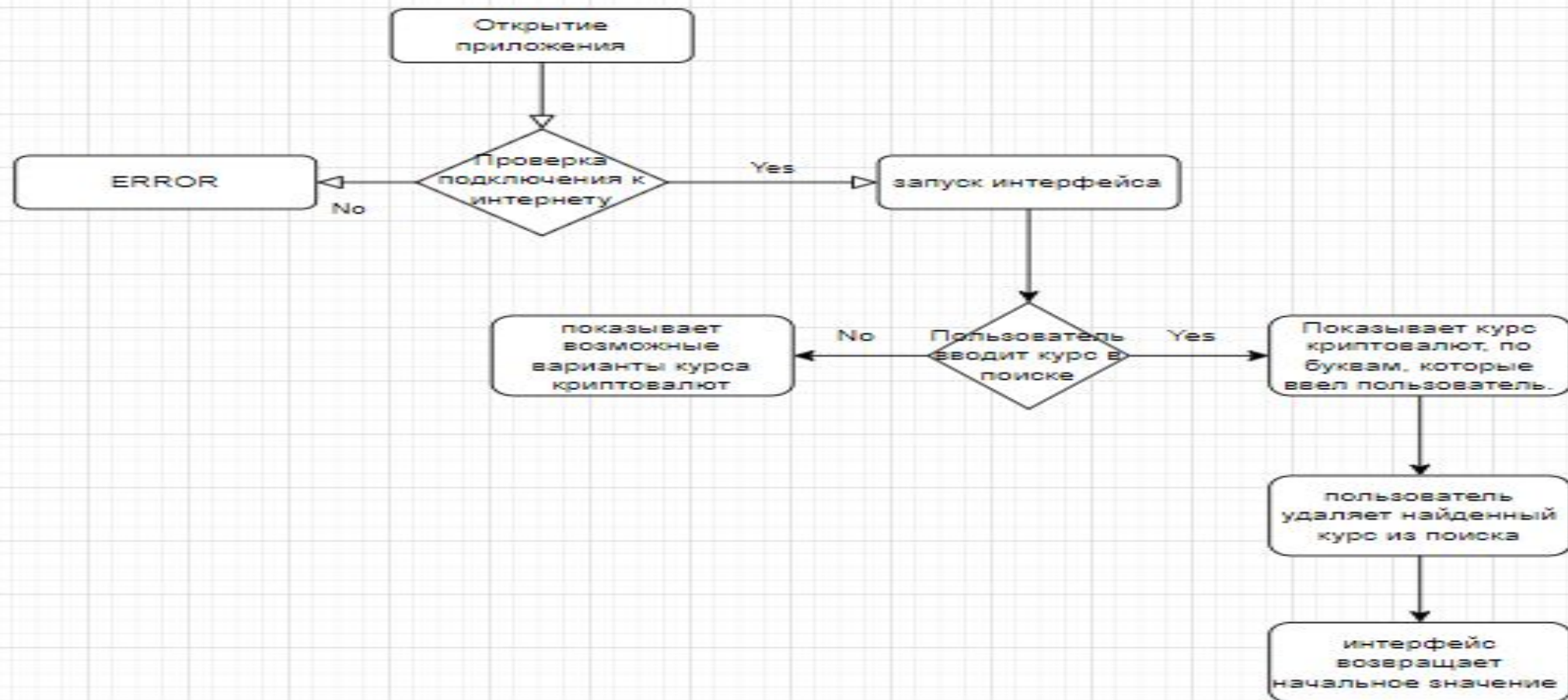
ПО/Критерий	Мой проект	<u>аналогов</u> Coinbase	Binance	Huobi
Большое кол-во монет	+	+	-	-
Скорость обработки запроса пользователя	+	-	-	+
Точный курс по американской бирже	+	-	+	+
Простота использования	+	+	-	-
Простой интерфейс	+	+	-	-
Без регистрации	+	+	-	-

Функциональная модель:



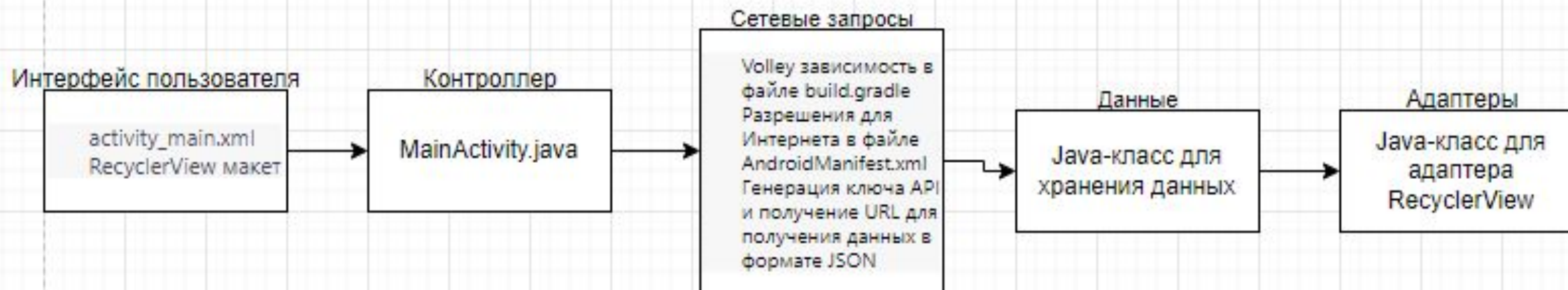
- Функциональная модель простая. Пользователю надо сделать пару кликов на экране, для того, чтобы узнать результат своего запроса.
-

Поведенческая модель:



Выбор Языка Программирования:

	Java	C++	C	Python
Возможность компиляции	+	+	+	+
Статическая типизация	+	+	+	-
Явная типизация	+	+	+	+
Неявная типизация	-	+/-	-	+
Наличие библиотек для работы с графикой и мультимедиа	+	+	+	+
Объектно-ориентированная парадигма	+	+	+/-	+
Функциональная парадигма	+/-	+/-	+/-	+
Императивная парадигма	+	+	+	+



Концептуальная модель:

Сначала пользовательский интерфейс [UI] взаимодействует с контроллером [MainActivity], который использует `Volley` для отправки сетевых запросов на сервер. Полученные данные обрабатываются в `Java`-классе, который затем используется для заполнения адаптера `RecyclerView`. Адаптер связывается с `RecyclerView` макетом, который в конечном итоге отображает данные для пользователя.

Скриншоты приложения:

