

# SYSTEM DESIGN Телеграм бот (Хакатон)

- 1. User Story
- 2. Job Story
- 3. USE CASE
- 4. Описание задачи
- 5. Прототип
- 6. Архитектура
- 7. BPMN схема процессов
- 8. Спецификация
- 9. Нефункциональные требования
- 10. Функциональные требования
- 11. Definition of Done
  - 11.1. Аналитика
  - 11.2. Тестирование
  - 11.3. JAVA Разработка
  - 11.4. DATA инженер

## 1. User Story

1	Как пользователь, я хочу иметь возможность смотреть курс криптовалют в телеграмм-боте, чтобы владеть актуальной информацией о криптовалютах.
2	Как пользователь, я хочу иметь возможность менять периодичность уведомлений об изменении курса криптовалют.
3	Как пользователь, я хочу иметь возможность отписаться от уведомлений определенного курса криптовалют, чтобы не получать лишнюю информацию.
4	Как пользователь, я хочу иметь возможность получать уведомление о нескольких курсах криптовалют, чтобы сравнивать динамику изменения цен.
5	Система должна получать информацию о курсах криптовалют с внешнего сервиса с помощью API.
6	Как пользователь, я хочу иметь возможность получать уведомление о понижении стоимости криптовалюты на заданный мной процент, чтобы купить/продать данную криптовалюту.
7	Как разработчик, я хочу видеть количество людей пользующиеся данным ботом.
8	Как разработчик, я хочу чтобы люди, когда отписывались от бота, указывали причину отписки: "что им не нравится в данном боте", для его улучшения.
9	Как клиент хочу чтобы при выведении криптовалют бот показывал капитализацию выбранной(ых) криптовалют(ы)

## 2. Job Story

1	Когда периодичность уведомлений об изменениях стоимости криптовалют мне не нравится, я хочу иметь возможность изменить время уведомлений, чтобы я поставил время удобное для меня.
2	Когда подписка на криптовалюту перестает быть актуальной, я хочу иметь возможность отписаться от уведомлений на нее, чтобы я не боялся пропустить информацию о нужной мне криптовалюте.
3.	Когда моя криптовалюта начинает падать в стоимости, я хочу быть уверенным, что стоимость моей криптовалюты не опуститься ниже заданного мной порога, чтобы я не потерял выгоду в продаже.

## 3. USE CASE

Действующие лица:

Пользователь, Telegram-bot

## Цели:

Пользователь: подписывается на изменение курса КРИПТОВАЛЮТ, выбирает периодичность уведомлений подписки, выбирает падение курса криптовалюты на определенный процент.

Telegram-bot: загружает курс криптовалют, обновляет курс криптовалют с определенной периодичностью, уведомляет пользователя о снижении стоимости криптовалюты на определенный пользователем процент.

**Сценарий 1:** Пользователь подписывается на Telegram-bot. Получает приветственное сообщение от бота: "Здравствуйте, я Telegram-bot, для отслеживания стоимости криптовалют. Чем могу помочь?"

Появляется три кнопки "Курс", "Уведомление", "Настройка".

Результат: Пользователь подписался на Telegram-bot

**Сценарий 2:** Пользователь нажимает кнопку "Курс". Telegram-bot предлагает выбрать криптовалюту из 10 наименований. Пользователь выбирает одну из криптовалют. Telegram-bot присылает стоимость выбранной криптовалюты и повторяет данное действие каждую следующую минуту. Тайминг в одну минуту задан по умолчанию. При повторе выше перечисленных действий, можно подписаться на другую криптовалюту.

Результат: Пользователь подписался на стрим стоимости криптовалюты

**Сценарий 3:** Пользователь нажимает на кнопку "Настройка". Telegram-bot предлагает выбор из 4 кнопок: "1 минута", "15 минут", "1 час", "24 часа". При нажатии на одну из кнопок, выбирается время отправки данных пользователю, о стоимости криптовалюты.

Результат: Пользователь выбрал периодичность уведомлений подписки

**Сценарий 4:** Пользователь нажимает кнопку "Уведомление". Telegram-bot предлагает выбрать криптовалюту из 10 наименований. Пользователь выбирает одну из криптовалют. Приходит сообщение "Вы можете получить уведомление если цена будет ниже X%. Напишите: 10". Пользователь указывает процент. Приходит сообщение "Когда цена упадет на X% (заданный пользователем процент) вам будет отправлено сообщение".

Результат: Пользователь подписался на уведомление о снижении стоимости криптовалюты на определенный процент.

**Сценарий 5:** Пользователь нажимает кнопку "Курс". Telegram-bot предлагает выбрать криптовалюту из 10 наименований. Пользователь выбирает одну из криптовалют выбранной ранее. Telegram-bot отменяет выбранную подписку.

Результат: Отмена подписки на выбранную криптовалюту.

## 4. Описание задачи

Пользователи хотят получать актуальную информацию о стоимости криптовалют.

Для этого нужно придумать и реализовать Telegram-bot с помощью которого подгружается стоимость криптовалют.

## 5. Прототип

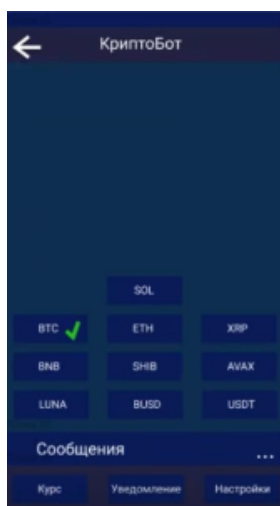
**Стартовая** страница, после приветствия и запуска бота.



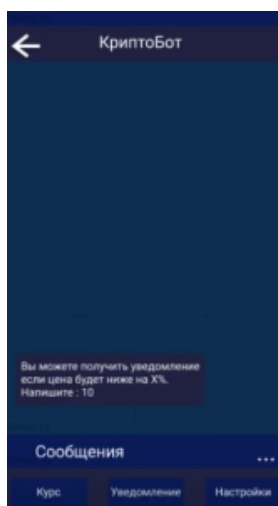
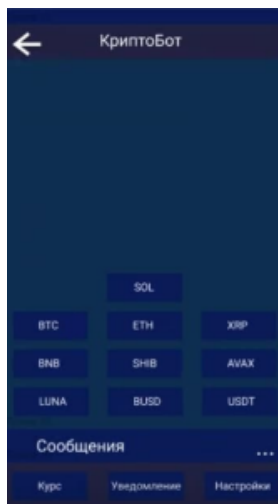
При нажатии на кнопку **настройка** появляется выбор частоты обновления подписки на курс крипты, которую выберем.



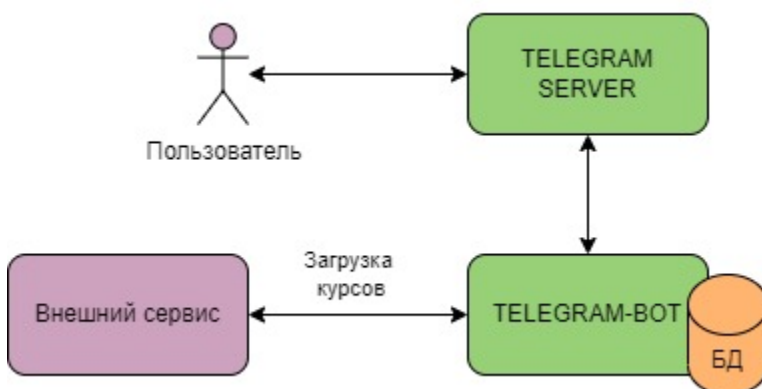
При нажатии на кнопку **курс**, появляется окно с выбором валюты, это и будет нашей подпиской на стрим.



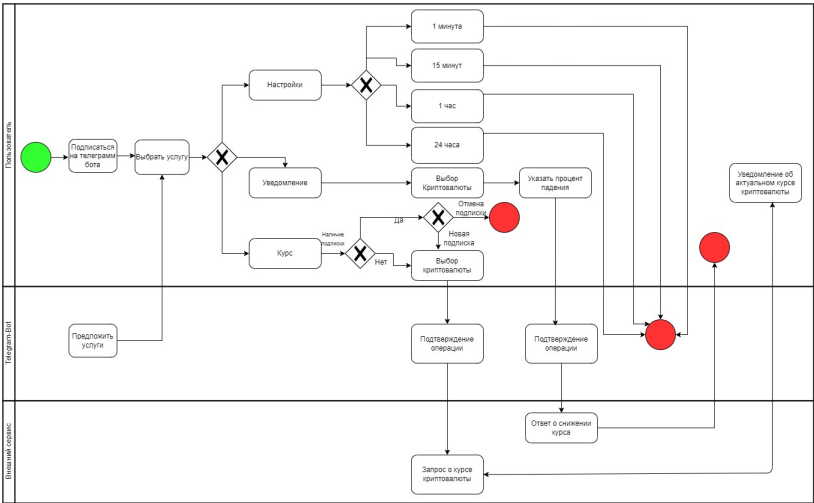
При нажатии на кнопку **уведомление**, появляется кнопка с выбором валют, как и с кнопкой курс, но после выбора валюты нам приходит сообщение с указанием процента, при понижении на который от начальной стоимости, бот нас оповестит.



## 6. Архитектура



## 7. BPMN схема процессов



8. Спецификация

Название операции	Запрос на внешний сервис					
Формат сообщений	REST JSON					
Метод	GET					
Потребители	TELEGRAM SERVER					
Поставщики	Внешний сервис					
URL	<a href="https://api.coingecko.com/api/v3/coins/markets?vs_currency=usd&amp;ids=tether">https://api.coingecko.com/api/v3/coins/markets?vs_currency=usd&amp;ids=tether</a>					
Параметры входящего сообщения	<b>№</b>	<b>Название</b>	<b>Описание</b>	<b>Тип данных</b>	<b>Обязательность</b>	<b>Пример</b>
	1	id	Внутренний уникальный код валюты	string	да	Tether
Пример входящего сообщения	{  "id": "Tether"  }					
Параметры исходящего сообщения	<b>№</b>	<b>Название</b>	<b>Описание</b>	<b>Тип данных</b>	<b>Обязательность</b>	<b>Пример</b>
	1	name	Название криптовалюты	string	да	Tether
	2	symbol	Значение	string	да	usdt
	3	market_cap_rank	Место в рейтинге	int	да	112.56
	4	current_price	Цена	float	да	1.001
	5	market_cap	Капитализация	float	да	79132967610
Пример исходящего сообщения	{ "id": { "name": "Tether", "symbol": "usdt", "market_cap_rank": 3, "current_price": 1.003, "market_cap": 79161642885 } }					

9. Нефункциональные требования

- 1. Система должна обеспечивать синхронное взаимодействие с внешним сервисом через REST API;
- 2. Запрос должен передаваться при помощи протокола TLS (HTTPS);

3. Должен быть создан экземпляр IP, для которого включено использование протокола TCP для работы с TLS;
4. HTTPS требует полный доступ к известному TCP-порту 443 для обработки всех HTTPS-запросов клиента;
5. Система должна взаимодействовать со всеми возможными ОС;
6. Время реакции от нажатия на кнопку до выполнения функции, назначенной кнопкой не должно превышать 1 сек;

## 10. Функциональные требования

1. Пользователь должен иметь возможность выбора криптовалюты;
2. Система должна обновлять курс криптовалют по заданным настройкам;
3. Пользователь должен иметь возможность выбора тайминга обновления курса криптовалюты;
4. Система должна отслеживать падение стоимости курса валюты "figi" с текущего на X%;
5. Пользователь должен иметь возможность отписаться от уведомлений определенного курса криптовалют;
6. Пользователь должен иметь возможность получать уведомление о нескольких курсах криптовалют;

## 11. Definition of Done

### 11.1. Аналитика

№	Название задачи	Выполнено
1	User Story	да
2	Job Story	да
3	Спецификация	да
4	BPMN схема	да

### 11.2. Тестирование

№	Название задачи	Выполнено
1	Тест план	да
2	План смоук тестирования	да
3	END TWO END	

### 11.3. JAVA Разработка

№	Название задачи	Выполнено
1	Подписка на стрим	да
2	Отписка от стрима	да
3	Подписка на события вида "падение курса валюты C от текущего на X%"	нет

### 11.4. DATA инженер

№	Название задачи	Выполнено
1	Стриминг данных	да
2	Агрегирование данных	да

3	Сохранение DATA фреймов в БД	да
---	------------------------------	----