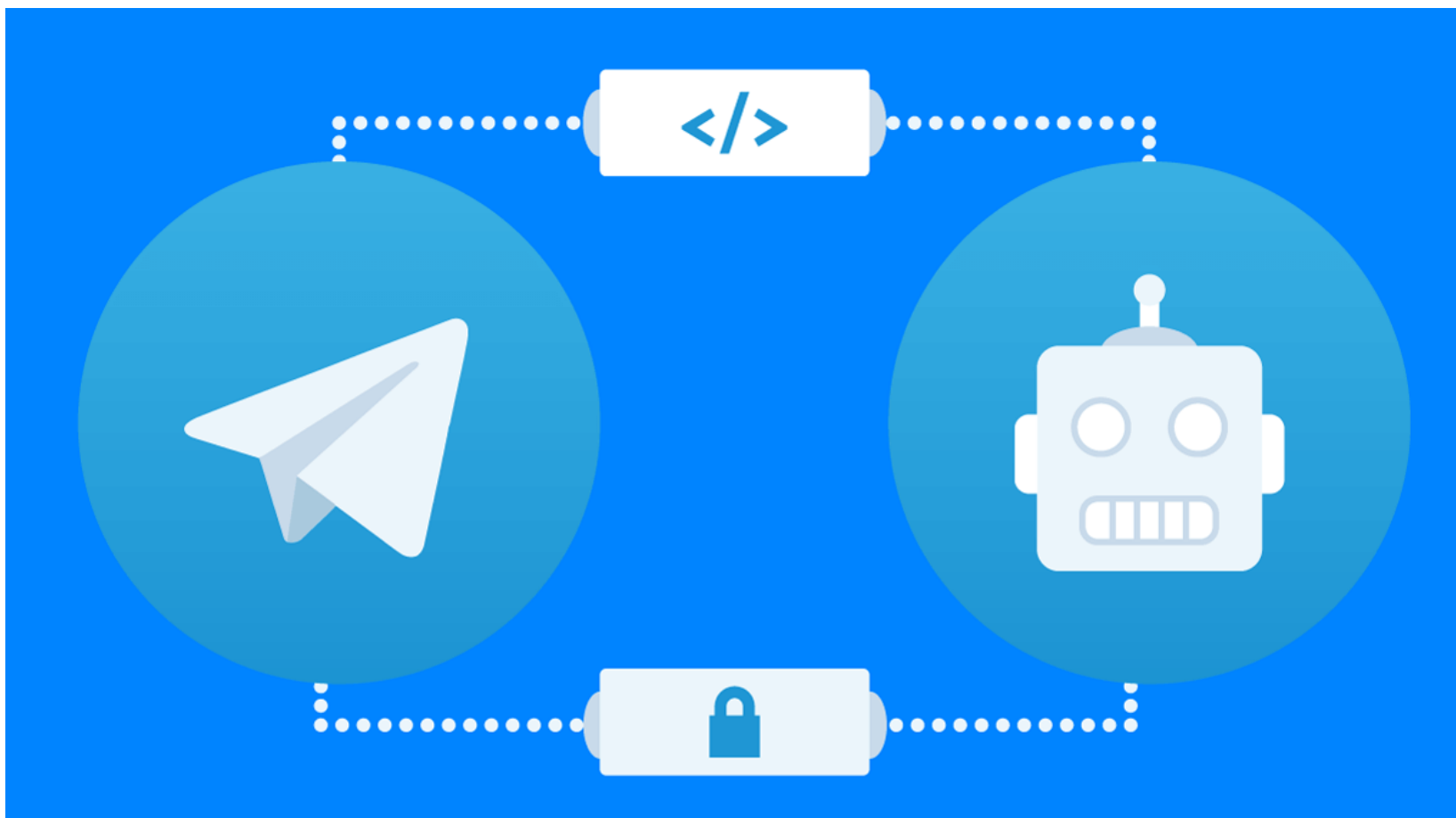




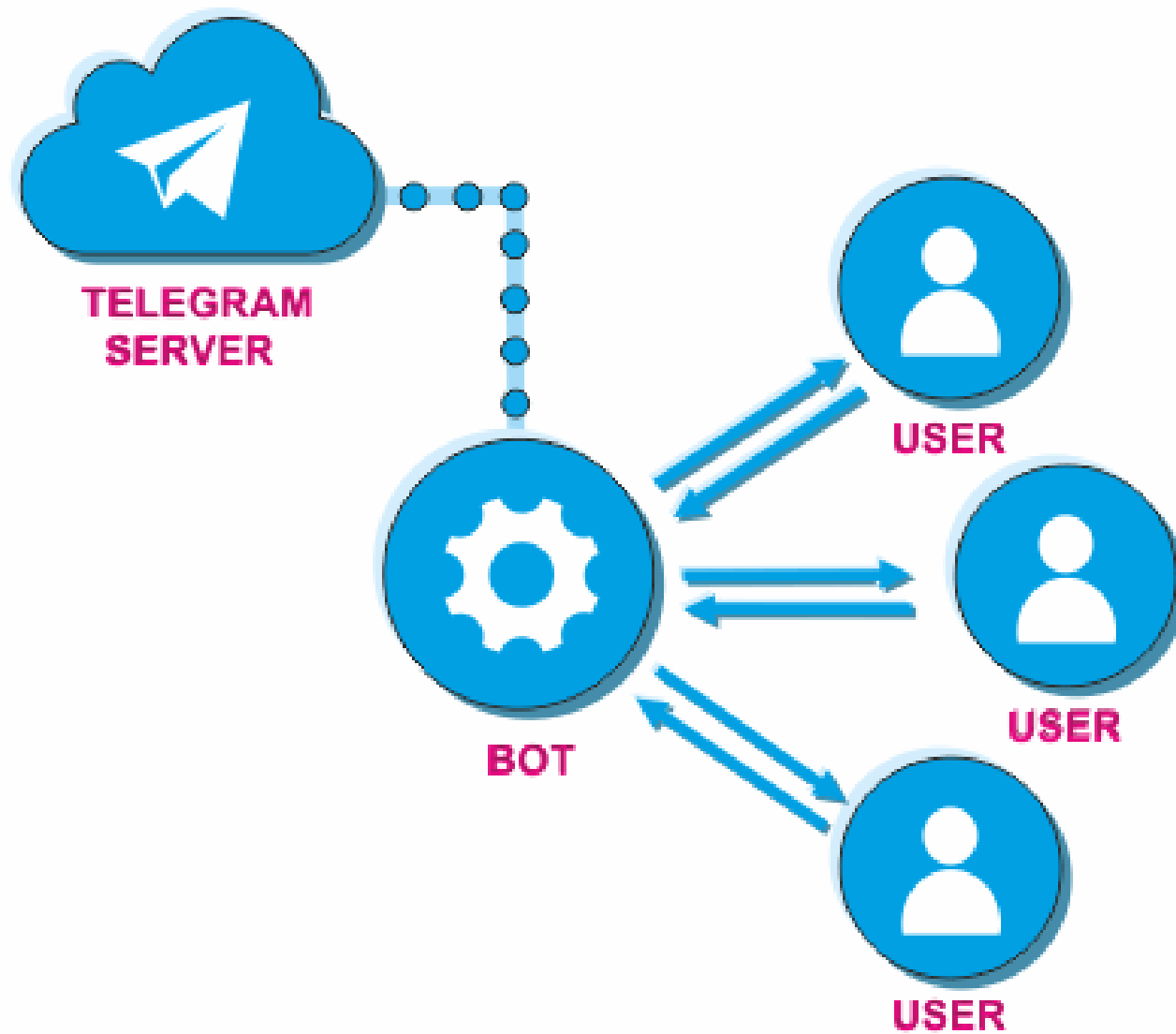
Telegram Bot API bilan ishlash

Reja:

1. Telegram bot
2. Creating new bot
3. Xabar jo'natish
4. ReplyKeyboardMarkup
5. InlineKeyboardMarkup



- Bot –bu Telegram ichida ishga tushadigan boshqalar(telegram ishlab chiqaruvchilari emas) tomonidan yozilgan dasturdir. User lar bot bilan xabar yuborish, buyruq jo'natish, [inline request](#) lar jo'natish orqali o'zaro aloqa qilishlari mumkin. Siz botingizni [Bot API](#) ga HTTPS so'rovlar yuborish orqali boshqarishingiz mumkin.



Telegram Bot bu – maxsus user bo'lib ular uchun telefon
raqam talab qilinmaydi

Yangi bot yaratish



[BotFather](#) - Bu barcha botlarni boshqaruvchi yagona botdir. U yordamida yangi bot yaratish va mavjud botlarning sozlamalarini o'zgartirish mumkin.

@BotFather

BotFather

Бот



Что может делать этот бот?

BotFather is the one bot to rule them all. Use it to create new bot accounts and manage your existing bots.

About Telegram bots:

<https://core.telegram.org/bots>

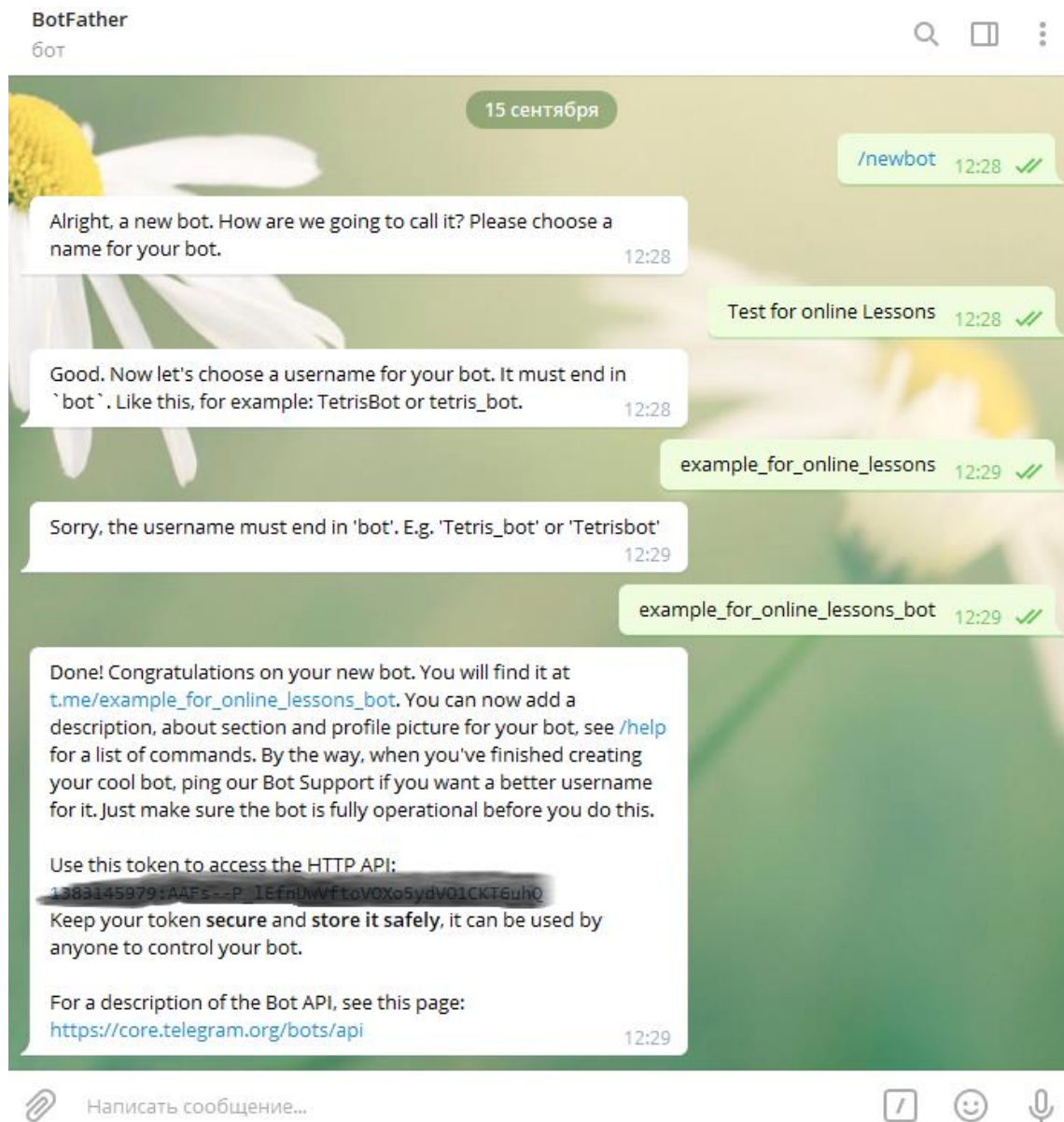
Bot API manual:

<https://core.telegram.org/bots/api>

Contact [@BotSupport](#) if you have questions about the Bot API.

/newbot

- 1) Bot ga nom berish
- 2) Bot uchun username berish
- 3) Bot uchun token olish





Telegram Bot API

Telegram Bot API



Polling

Webhook



Dependency

```
<!-- https://mvnrepository.com/artifact/org.telegram/telegrambots -->  
<dependency>  
  <groupId>org.telegram</groupId>  
  <artifactId>telegrambots</artifactId>  
  <version>4.9.1</version>  
</dependency>
```

```
public class ExampleBot extends TelegramLongPollingBot {
```

```
    public static void main(String[] args) {  
        ApiContextInitializer.init();  
        TelegramBotsApi api = new TelegramBotsApi();  
        try {  
            api.registerBot(new ExampleBot());  
        } catch (TelegramApiRequestException e) {  
            e.printStackTrace();  
        }  
    }  
}
```

```
    @Override  
    public String getBotToken() {  
        return "token here";  
    }  
}
```

```
    @Override  
    public void onUpdateReceived(Update update) {}  
}
```

```
    @Override  
    public String getBotUsername() {  
        return "username here";  
    }  
}
```


ReplyKeyboardMarkup

// Klaviatura yaratish

```
ReplyKeyboardMarkup replyKeyboardMarkup = new ReplyKeyboardMarkup();  
sendMessage.setReplyMarkup(replyKeyboardMarkup);  
replyKeyboardMarkup.setSelective(true);  
replyKeyboardMarkup.setResizeKeyboard(true);  
replyKeyboardMarkup.setOneTimeKeyboard(false);
```

// Klaviatura qatorlari listi

```
List<KeyboardRow> keyboard = new ArrayList<>();
```

// Klaviatura birinchi qatori

```
KeyboardRow keyboardFirstRow = new KeyboardRow();
```

// Birinchi qatorga tugma qo'shish

```
keyboardFirstRow.add(new KeyboardButton("Salom"));
```

// Klaviatura ikkinchi qatori

```
KeyboardRow keyboardSecondRow = new KeyboardRow();
```

// Ikkinchi qatorga tugma qo'shish

```
keyboardSecondRow.add(new KeyboardButton("Yordam"));
```

// Barcha qatorlarni qatorlar listiga qo'shish

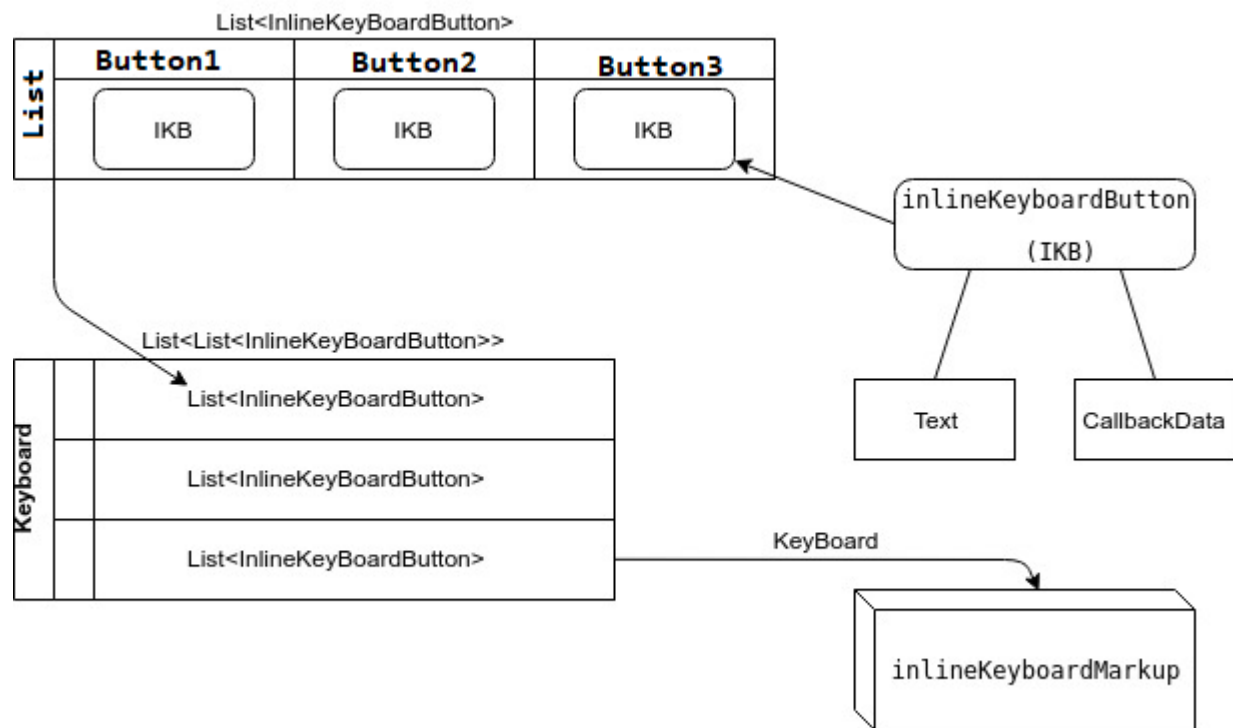
```
keyboard.add(keyboardFirstRow);
```

```
keyboard.add(keyboardSecondRow);
```

// qatorlar listini klaviatura sifatida o'rnatish

```
replyKeyboardMarkup.setKeyboard(keyboard);
```

InlineKeyboardMarkup



```
InlineKeyboardMarkup inlineKeyboardMarkup = new InlineKeyboardMarkup();
InlineKeyboardButton inlineKeyboardButton1 = new InlineKeyboardButton();
inlineKeyboardButton1.setText("Salom");
inlineKeyboardButton1.setCallbackData("Button \"Assalomalaykum\" has been pressed");

List<InlineKeyboardButton> keyboardButtonsRow1 = new ArrayList<>();

keyboardButtonsRow1.add(inlineKeyboardButton1);

List<List<InlineKeyboardButton>> rowList = new ArrayList<>();
rowList.add(keyboardButtonsRow1);

inlineKeyboardMarkup.setKeyboard(rowList);
return new SendMessage().setChatId(chatId).setText("Example").setReplyMarkup(inlineKeyboardMarkup);
```

E'TIBORINGIZ UCHUN RAXMAT