

Пишем свой мессенджер на основе современных технологий



Русаков Алексей

vk.com/codelevel

vk.com/rusalm t.me/RusAl84

github.com/RusAl84/MyMessengerMasterClass



WhatsApp

Автор: Брайан Эктон, Ян Кумим

Разработчик: Facebook

Написан на: Erlang, XMPP

Первый релиз: январь 2009

Компания WhatsApp Inc., создавшая мессенджер, была основана Яном Кумом и Брайаном Эктоном 24 февраля 2009 года и расположена в Маунтин-Вью, США; с октября 2014 года принадлежит Facebook Inc.

Из-за высокой популярности WhatsApp снижается выручка сотовых операторов за передачу SMS-сообщений между телефонами. По отдельным оценкам 2012 года, потенциальные потери могли достигать десятков миллиардов долларов.



Signal

Разработчик: Signal Messenger

Написан на: Java, C, Objective-C, XMPP

Первый релиз: июнь 2014

Signal разработан Мокси Марлинспайком и созданной им компанией *Open Whisper Systems*.

Основной акцент в разработке делается на конфиденциальность и безопасность. До ноября 2015 года приложение называлось *TextSecure* и позволяло лишь обмен мгновенными сообщениями.



Telegram

Автор: Николай и Павел Дуровы

Разработчик: Telegram Messenger

Написан на: C++, XMPP

Интерфейс: QT

Первый релиз: сентябрь 2013

Количество ежемесячных активных пользователей сервиса, по состоянию на конец марта 2018 года, составляет более 200 млн человек. В августе 2017 года в своем Telegram-канале Павел Дуров заявил, что количество пользователей мессенджера ежедневно увеличивается более чем на 600 тысяч.

Помимо стандартного обмена сообщениями в диалогах и группах, в мессенджере можно хранить неограниченное количество файлов, вести каналы (микроблоги), создавать и использовать ботов.



Facebook Messenger

Разработчик: Facebook

Написан на: Erlang, PHP, JavaScript, C++, XMPP

Первый релиз: август 2011

В декабре 2012 приложение Facebook Messenger для Android в некоторых странах (Австралии, Индии, Индонезии, ЮАР, Венесуэле, и др.) позволило использовать его тем, кто не зарегистрирован в социальной сети Facebook, используя лишь имя и номер телефона. Это обновление позволило Facebook Messenger конкурировать с сходными системами обмена сообщениями, например WhatsApp.

В апреле 2014 года компания Facebook объявила, что отключит возможность отправки текстовых сообщений из своего основного сотового приложения «Facebook», вынудив пользователей скачать и установить Facebook Messenger.



Discord

Разработчик: Discord Inc.

Написан на: JavaScript, React, Elixir, XMPP

Первый релиз: Март 2015

Бесплатный мессенджер изначально ориентированный на пользователей компьютерных игр. Разработчик — компания Hammer & Chisel (Discord Inc.) из Сан-Франциско.

Приложение способно организовывать голосовые конференции с настройкой канала связи, создавать публичные и приватные чаты для обмена текстовыми сообщениями.



Skype

Разработчик: Microsoft (ранее Skype Limited)

Написан на: Embarcadero Delphi, Objective-C, C++, Object Pascal, JavaScript, XMPP

Интерфейс: QT

Первый релиз: август 2003

Компания Skype Technologies была основана в 2003 году шведом Никласом Зеннстремом и датчанином Янусом Фриисом. В создании программы Скайп участвовали эстонские программисты, создавшие ранее программу для файлообмена KaZaA.



iMessage

Разработчик: Apple

Написан на: Swift, XMPP

Первый релиз: октябрь 2011

Технология и служба iMessage были анонсированы Скотом Форстоллом 6 июня 2011 года на конференции разработчиков WWDC2011.

В версию iOS 5.0 (от 12.10.2011) Apple включила приложение Сообщения, которое использовало все возможности этой технологии.

16 Февраля 2012 года Apple объявила о том, что в новой OS X приложение Сообщения, заменит приложение iChat и оно будет частью OS X Mountain Lion. OS X Mountain Lion с iMessages официально вышла 25 июля 2012 года.



WeChat

Разработчик: Tencent Holdings

Написан на: C++, XMPP

Первый релиз: январь 2011

Разработана китайской компанией Tencent, первый релиз был выпущен в январе 2011 года. Приложение доступно на iPhone, а также на телефонах под управлением Android, BlackBerry, Windows Phone, Symbian и J2ME/S40, также существует веб-интерфейс и клиент для PC (Windows и OS X, которые, однако, требуют установки на смартфоне для аутентификации, при этом такая возможность доступна лишь пользователям Android, iOS и Windows Phone. По состоянию на 2013 год, в WeChat было зарегистрировано 300 миллионов пользователей; из них 70 миллионов за пределами Китая. К 2018 году количество активных пользователей "WeChat" превысило 1 миллиард. К 2019 году их насчитывалось уже 1,151 млрд.



Viber

Разработчик: Viber Media S.à r.l.

Написан на: Java, C, Python C++, Objective-C, XMPP

Первый релиз: декабрь 2010

Первая версия приложения была разработана 2 декабря 2010 года исключительно для iPhone и имела ограничение на количество пользователей в 50000. Viber для BlackBerry, Bada и Windows Phone был выпущен в мае 2012 года. Через год, в 2013 году, вышло обновление приложения для iOS до версии 3.0, с релизом которого было объявлено о доступности Viber Desktop для Windows и OS X.

14 января 2014 года японская электронная компания Rakuten заявила о своём намерении приобрести Viber за 900 млн долларов. Сделка была завершена в феврале 2014 года.



Hangouts

Разработчик: Google

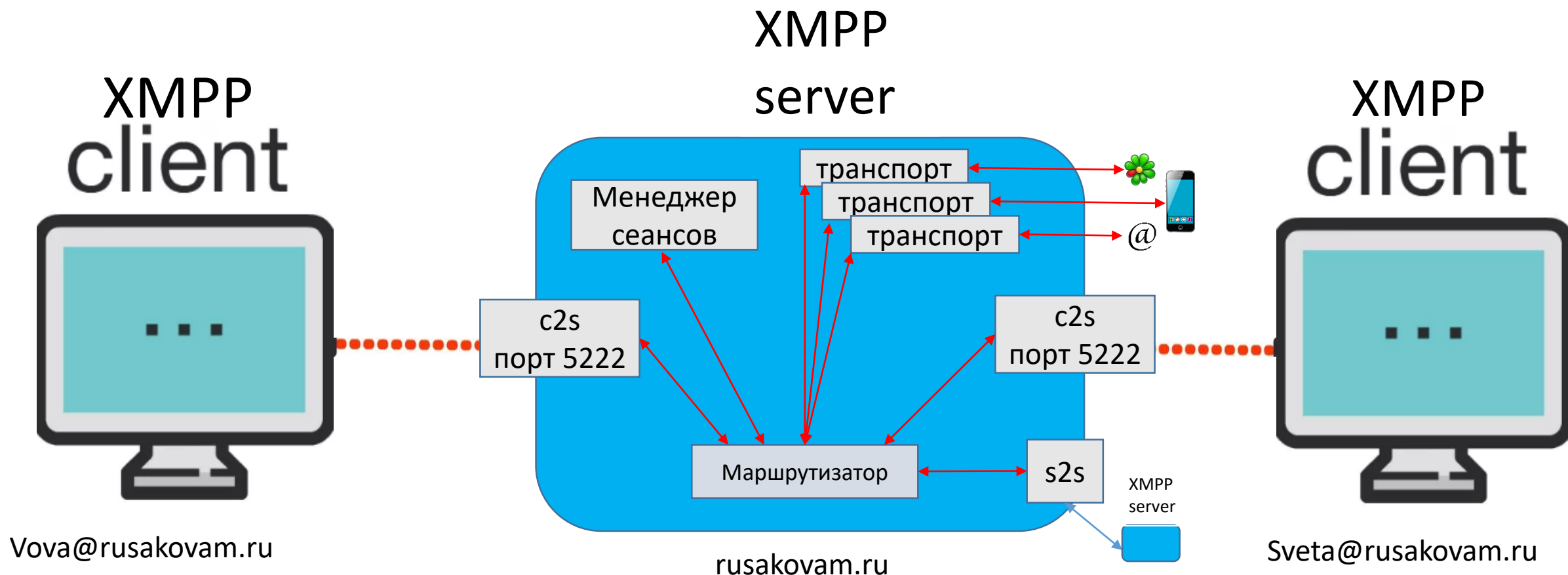
Написан на: Java, XMPP

Первый релиз: май 2013

До запуска *Hangouts* компания *Google* обслуживала несколько похожих, но технологически отличающихся систем обмена сообщениями. Это *Google Talk* (основанный на протоколе XMPP), *Google+ Messenger*, а также часть *Google+*, именованная *Hangouts*, которая позволяла создавать групповые видеоконференции на 10 человек. В 2013 году *Google* объединила три свои службы в одну. Сервис, который до релиза фигурировал в отчётах под названием *Babel*, был официально запущен как *Hangouts* во время конференции разработчиков Google I/O 15 мая 2013 года.

В августе 2015 года мессенджер *Google Hangouts* получил веб-версию.

Архитектура XMPP

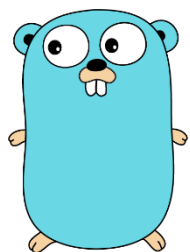
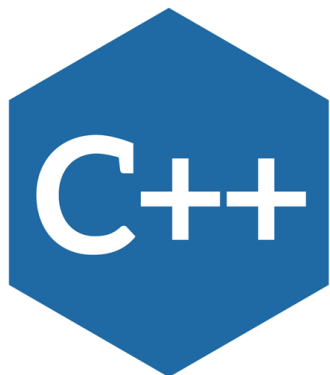


XMPР (англ. *eXtensible Messaging and Presence Protocol* «расширяемый протокол обмена сообщениями и информацией о присутствии»), «транспорт» для обмена сообщениями общего назначения

Какой инструмент разработчика выбрать?



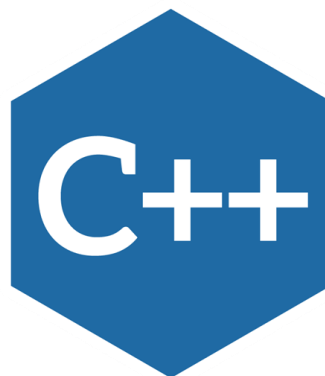
Языки программирования



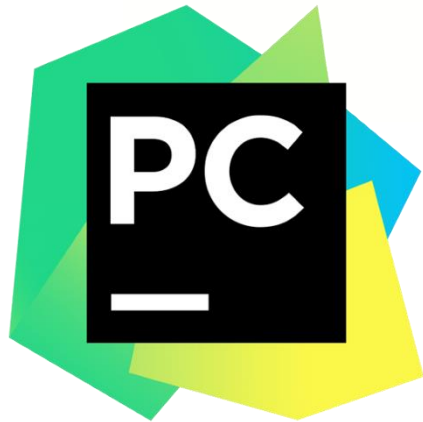
Golang



Интегрированная среда разработки



Интегрированная среда разработки



Sublime Text

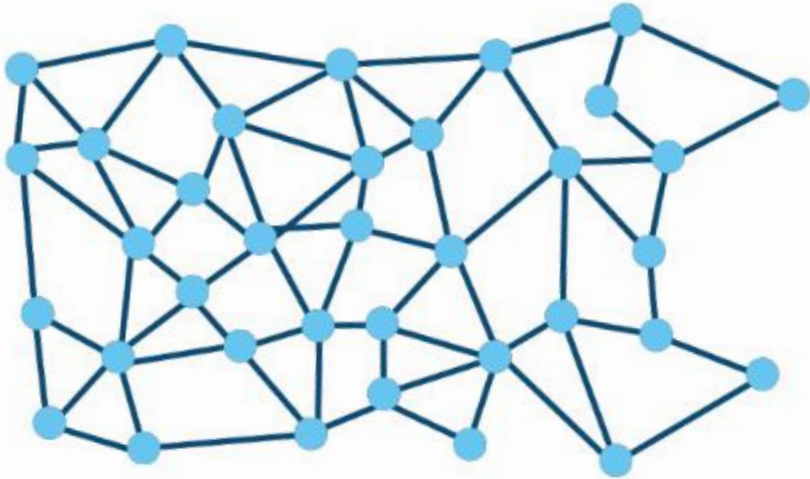


Интегрированная среда разработки

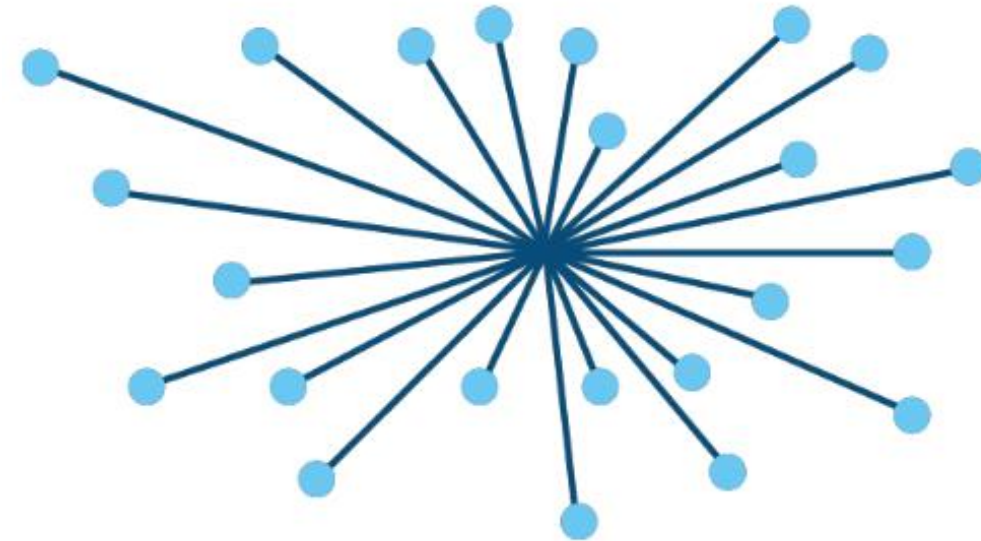


системы управления версиями

распределенные



централизованные

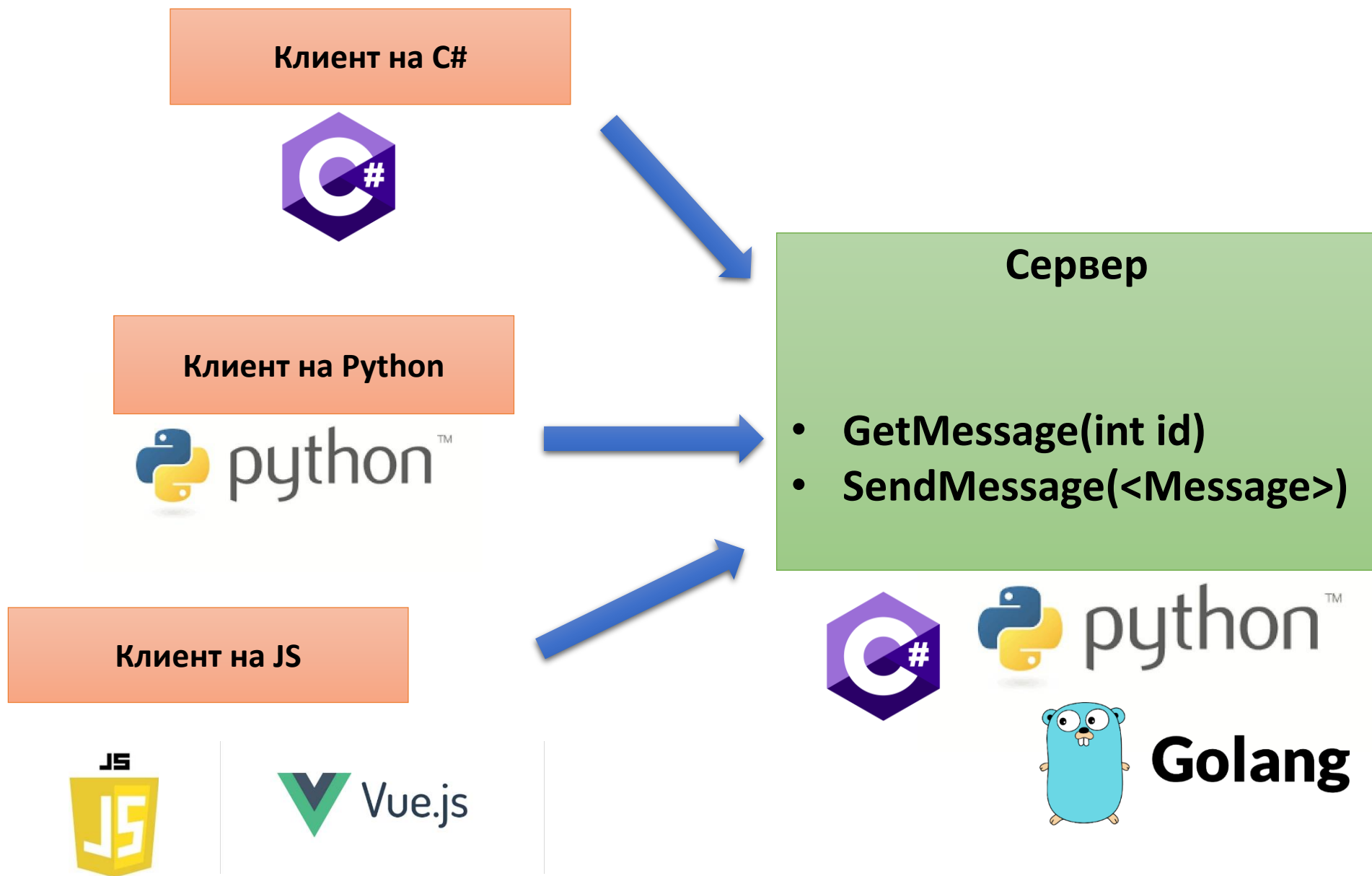


CVS

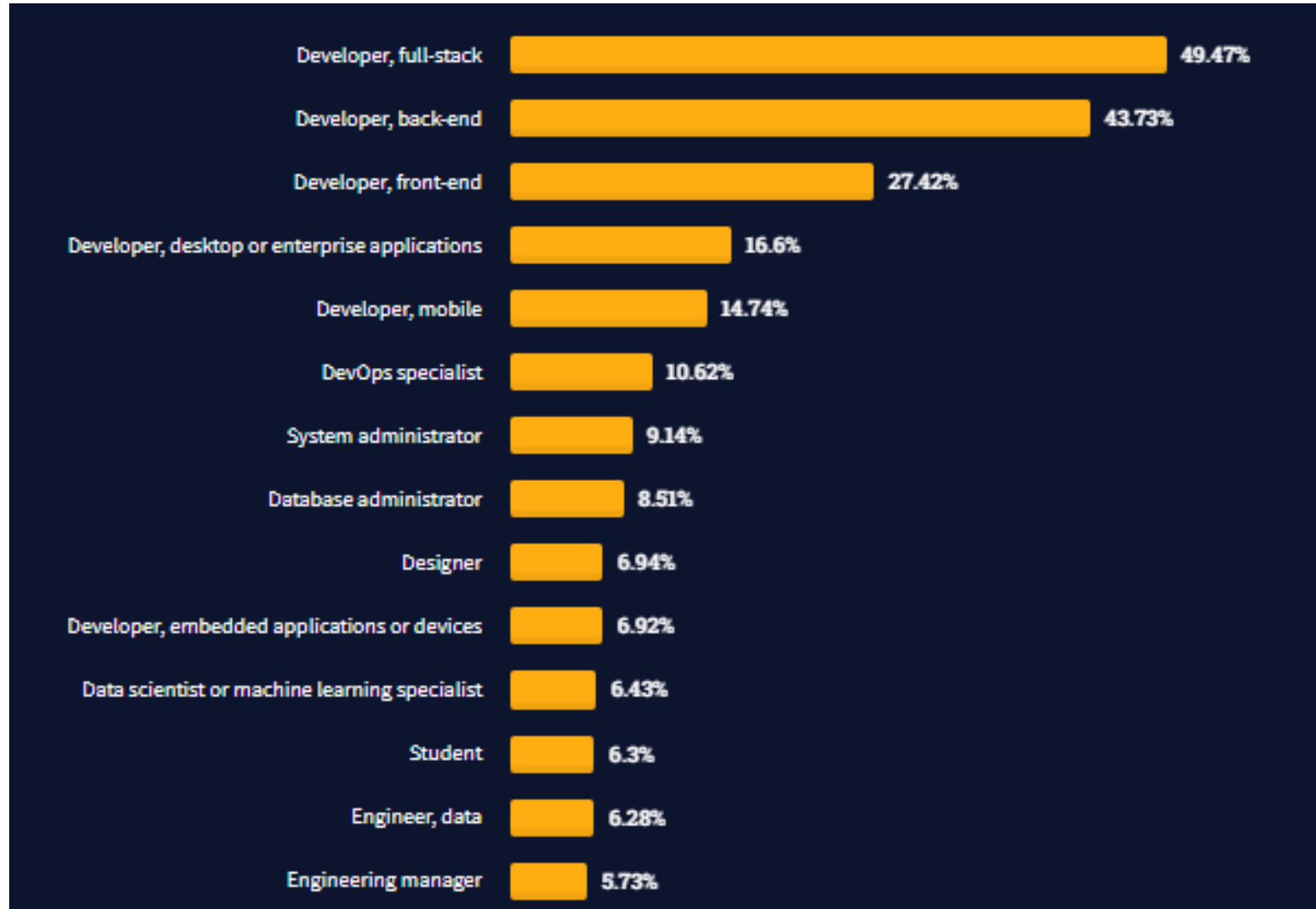
Методология разработки GitFlow



Принцип работы мессенджера

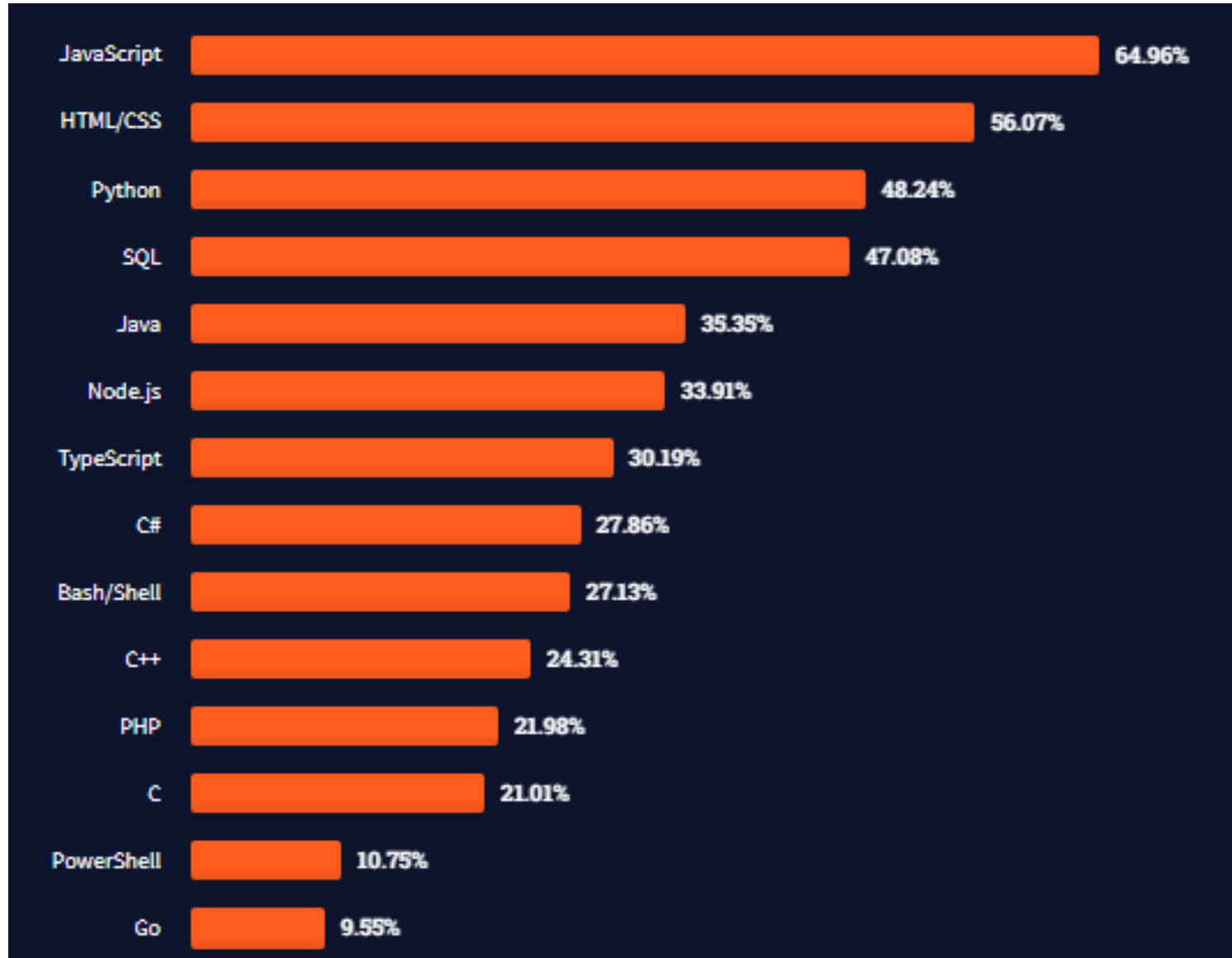


Результат опроса разработчиков на Stack Overflow 2022

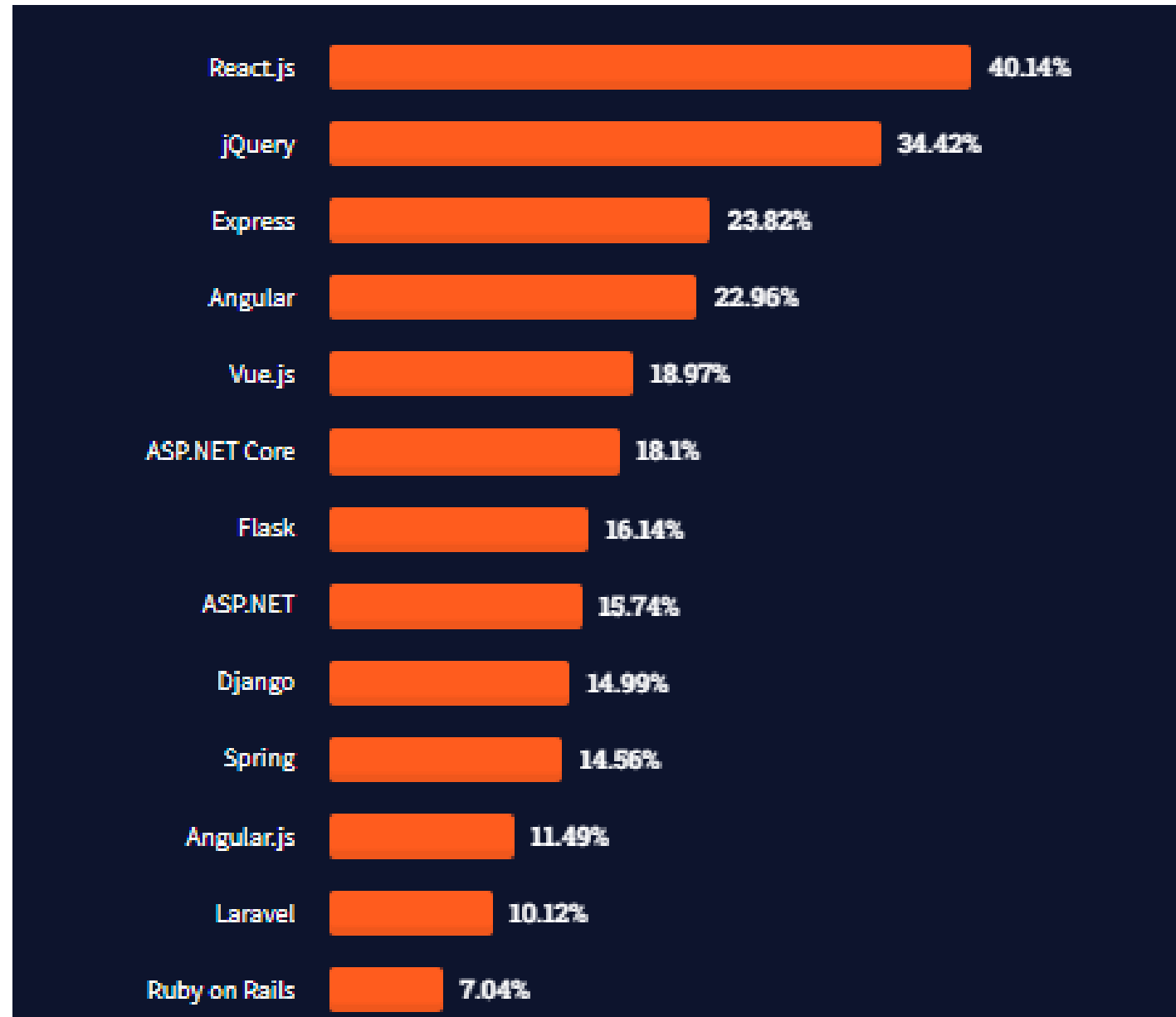


insights.stackoverflow.com/survey/2022

Самые любимые языки (Programming, Scripting, and Markup Languages)

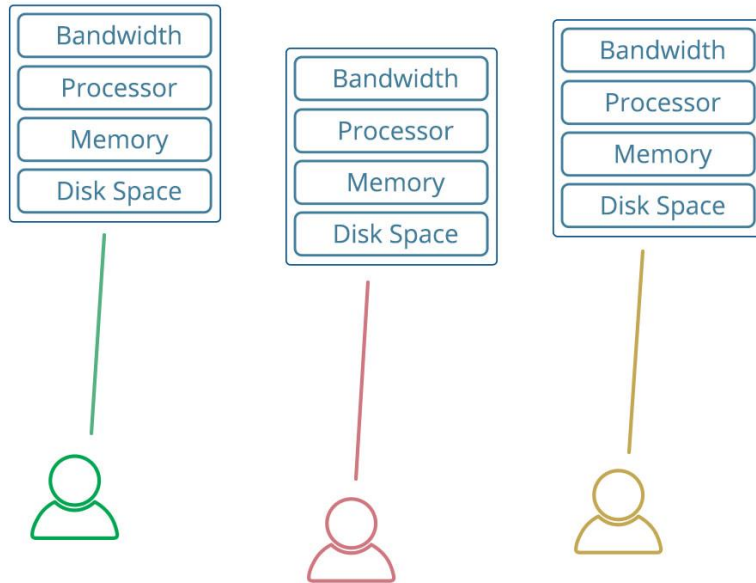


Web frameworks

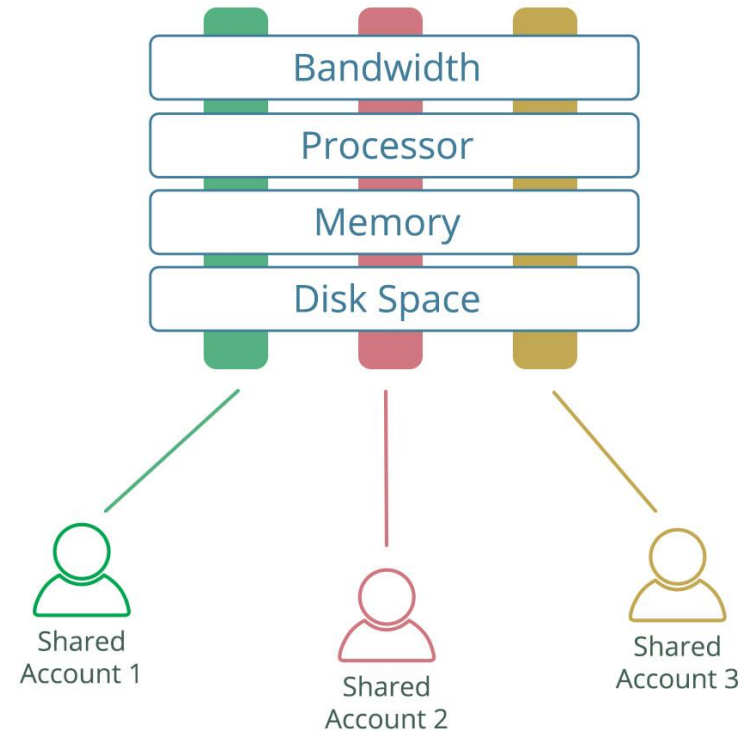


Где можно разместить сервер?

Dedicated Hosting



VPS Hosting





Купить **VPS** на **Linux** | Аренда виртуального сервера...

REG.ru > vps/linux ▾

Администрируйте **VPS** на **Linux** через панель управления. Вы наверняка уже пользовались панелью управления ISPmanager 5 на ... Администрируйте **VPS** без специальных знаний **Linux** и программирования. Управление веб-сервером, PHP...

[Читать ещё >](#)

VPS Linux virtual server за 130 руб | недорогой сервер от...

ruvds.com > ru-rub/linux ▾

Стоимость аренды практически бесплатного **VPS** хостинга всего 130 рублей в месяц, за эту цену вы получите в ваше распоряжение одно 2 GHz ядро ... Самый дешевый **VPS** сервер базируется на операционной системе **Linux**. [Читать ещё >](#)

VPS/VDS Linux — аренда виртуального сервера...

VPS.today > linux ▾

Лучшие виртуальные серверы **VPS/VDS** с ОС **Linux** ✓ Промокоды и акционные предложения ★ Сравните по более чем 20 ... На нашем сайте представлено 2 029 виртуальных серверов **VPS/VDS** с ОС **Linux** по цене от 40 рублей. [Читать ещё >](#)

VDS/VPS с Ubuntu **Linux**. Аренда виртуального сервера...

vpsville.ru > vps/templates/linux/ ▾

VPS/VDS Оптимальные по цене тарифы на скоростных NVMe дисках от 499Р/мес. Аренда физического сервера Если Вашему ... Тарифы на аренду виртуального сервера на **Linux**. Гибкие тарифные планы подходят как небольшим, так и очень крупным интернет-проектам. Предлагаем различный объем... [Читать ещё >](#)

Сериализация C#

```
[Serializable]
public class message
{
    public string username { get; set; }
    public string text { get; set; }
    public DateTime timestamp { get; set; }

    public message()
    {
        this.username = "Server";
        this.text = "Server is running...";
        this.timestamp = DateTime.UtcNow;
    }

    public message( string username, string text, DateTime timestamp)
    {
        this.username = username;
        this.text = text;
        this.timestamp = timestamp;
    }
}
```

Формат сериализации

- бинарный
- SOAP
- xml
- JSON

XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<person>
  <name>Иван</name>
  <age>37</age>
  <mother>
    <name>Ольга</name>
    <age>58</age>
  </mother>
  <children>
    <child>Маша</child>
    <child>Игорь</child>
    <child>Таня</child>
  </children>
  <married>true</married>
  <dog null="true" />
</person>
```

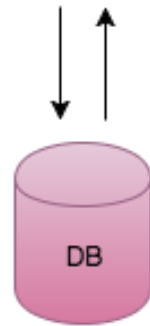
VS

JSON

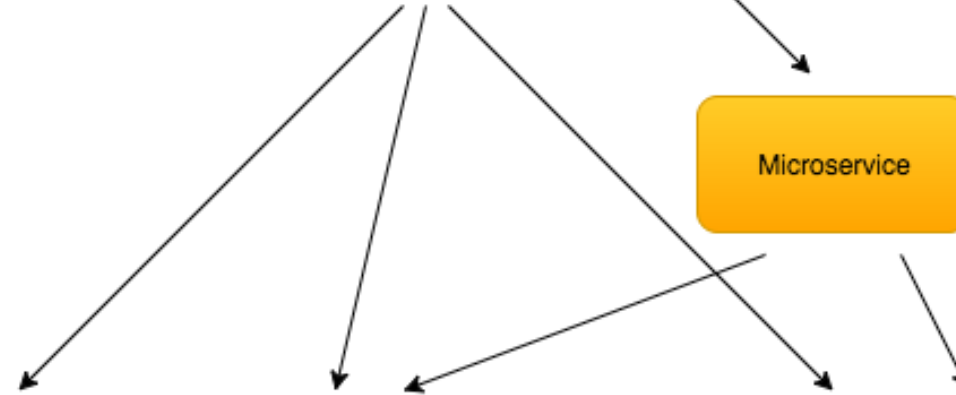
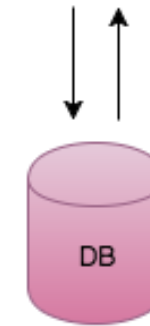
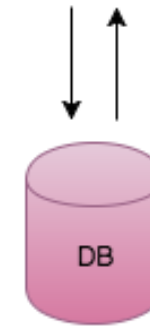
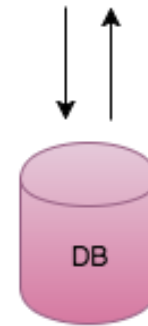
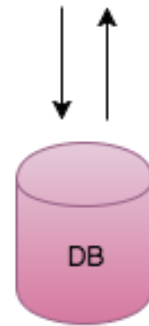
```
{
  "person":{
    "name":"Иван",
    "age":37,
    "mother":{
      "name":"Ольга",
      "age":58
    },
    "children":[
      "Маша",
      "Игорь",
      "Таня"
    ],
    "married":true,
    "dog":null
  }
}
```


Архитектура программного обеспечения

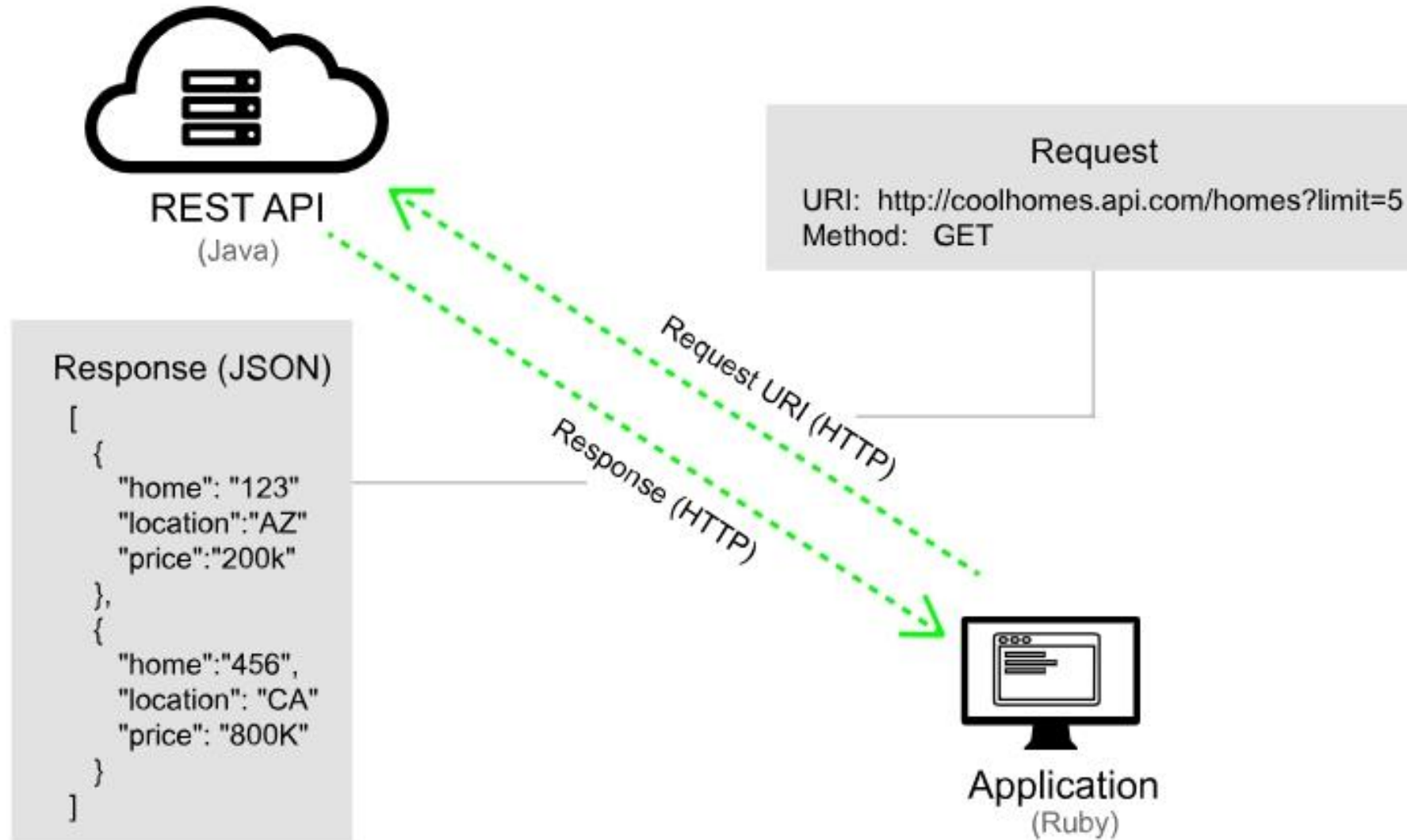
Monolithic Architecture



Microservices Architecture



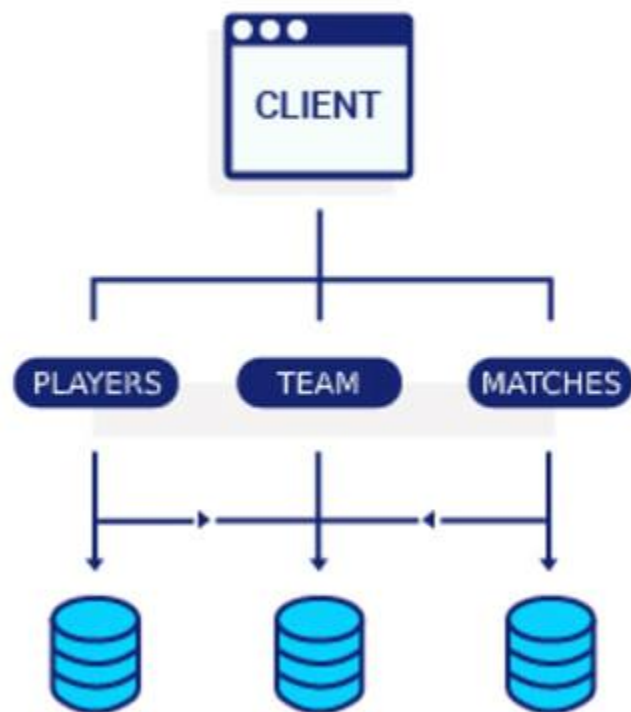
REST API model



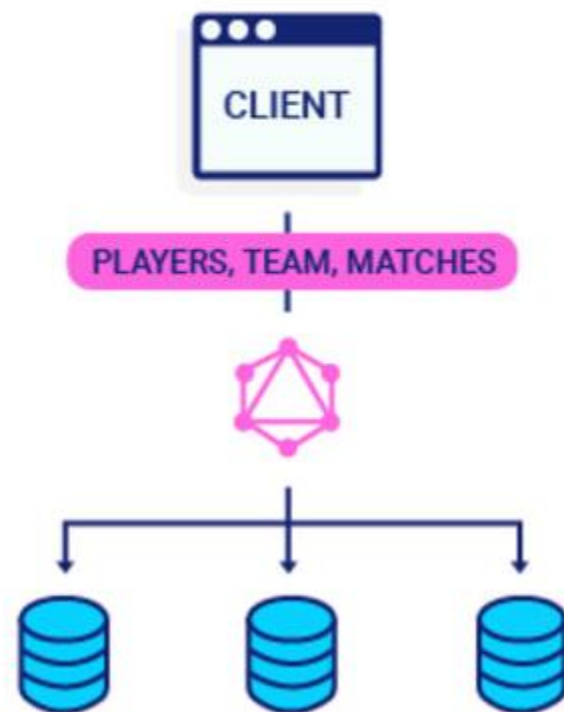
API TIMELINE



REST API



GraphQL API



Базовый стек технологий

Клиент серверное приложение. Технология связи клиент-сервер: Rest API через HTTPS

Сервер:

Среда разработки Visual Studio 2022: <https://visualstudio.microsoft.com/>

Язык программирования C#: <https://docs.microsoft.com/ru-ru/dotnet/csharp/tour-of-csharp/>

Фреймворк Asp .Net Core: <https://docs.microsoft.com/ru-ru/aspnet/core/?view=aspnetcore-6.0>

Библиотека Json.NET – Newtonsoft: <https://www.newtonsoft.com/json>

Клиент:

Среда разработки Visual Studio 2022: <https://visualstudio.microsoft.com/>

Язык программирования C#: <https://docs.microsoft.com/ru-ru/dotnet/csharp/tour-of-csharp/>

Фреймворк Windows Forms .NET: <https://docs.microsoft.com/ru-ru/dotnet/desktop/winforms/overview/?view=netdesktop-6.0>

Библиотека RestSharp: <https://restsharp.dev/>

Библиотека Json.NET – Newtonsoft: <https://www.newtonsoft.com/json>

Современный стек технологий

Клиент серверное приложение. Технология связи клиент-сервер: Rest API через HTTPS

WebSocket, MongoDB, PostgreSQL, Elasticsearch etc

Сервер:

Среда разработки pyCharm: <https://www.jetbrains.com/ru-ru/pycharm/>

Язык программирования Python: <https://ru.wikipedia.org/wiki/Python>

Фреймворк Flask: <https://flask.palletsprojects.com/en/2.0.x/>

Библиотека Flask-SocketIO

Клиент:

Среда разработки Visual Studio Code: <https://code.visualstudio.com/>

Язык программирования JavaScript: <https://ru.wikipedia.org/wiki/JavaScript>

Язык программирования TypeScript: <http://typescript-lang.ru/docs/>

Фреймворк Vue.JS 3 <https://vuejs.org/>

Фреймворк Quasar <https://quasar.dev/>

Библиотека Vue Router <https://router.vuejs.org/>

Библиотека AXIOS <https://axios-http.com/docs/intro>

Flask — фреймворк для создания веб-приложений на языке программирования Python, использующий набор инструментов Werkzeug, а также шаблонизатор Jinja2.

Относится к категории так называемых микрофреймворков — минималистичных каркасов веб-приложений, сознательно предоставляющих лишь самые базовые возможности.



Hello World!

Пример веб-приложения, которое показывает «Hello World!»:

```
from flask import Flask
app = Flask(__name__)
@app.route("/")
def hello():
    return "Hello World!"
if __name__ == "__main__":
    app.run()
```

Маршрутизация

```
@app.route('/')  
def index():  
    return 'Index Page'
```

```
@app.route('/hello')  
def hello():  
    return 'Hello World'
```

Использование переменных

```
@app.route('/user/<username>')  
def show_user_profile(username):  
    # show the user profile for that user  
    return 'User %s' % username
```

```
@app.route('/post/<int:post_id>')  
def show_post(post_id):  
    # show the post with the given id, the id is an integer  
    return 'Post %d' % post_id
```

Имеются следующие конверторы

int	Принимает целые числа
float	то же самое, что int, только с плавающей точкой
path	похоже на то, что установлено по умолчанию, но принимает слэши



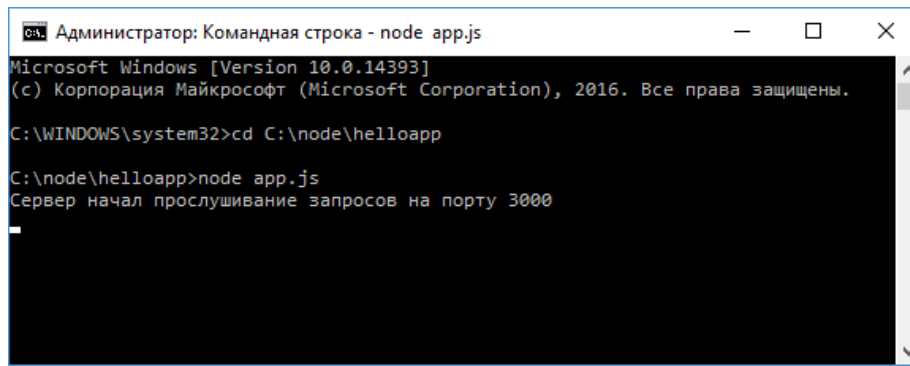
JavaScript

- Объектно-ориентированный скриптовый язык программирования.
- Является диалектом (вариация или расширение) языка ECMAScript.

Простейшее Приложение

```
const http = require("http");  
http.createServer(function(request,response){  
    response.end("Hello NodeJS!");  
}).listen(3000, "127.0.0.1",function(){  
    console.log("Сервер начал прослушивание запросов на порту  
3000"); });
```

```
cd C:\node\helloapp
```



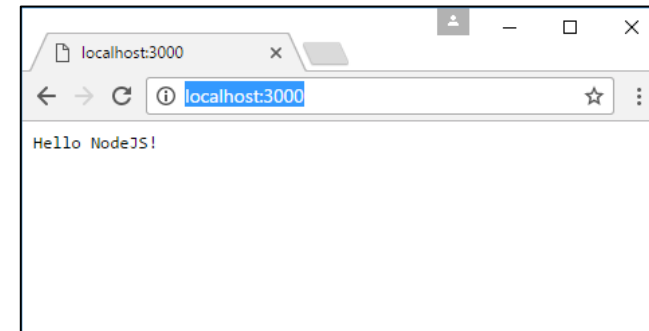
Администратор: Командная строка - node app.js

Microsoft Windows [Version 10.0.14393]
(c) Корпорация Майкрософт (Microsoft Corporation), 2016. Все права защищены.

C:\WINDOWS\system32>cd C:\node\helloapp

C:\node\helloapp>node app.js

Сервер начал прослушивание запросов на порту 3000



Основы работы с Node JS

```
1const os = require("os");  
2// получим имя текущего пользователя  
3let userName = os.userInfo().username;  
4  
5console.log(userName);
```

Основы работы с Node JS

- NPM(Node Package Manager) – менеджер для автоматизации установки и обновления пакетов
- Для нас менеджер npm важен в том плане, что с его помощью легко управлять пакетами.
- Express представляет легковесный веб-фреймворк для упрощения работы с Node.js.
- После установки express в папке проекта `modulesapp` появится подпапка **node_modules**, в которой будут храниться все установленные внешние модули.

```
npm install express
```


Package.json

```
{  
  "name": "modulesapp",  
  "version": "1.0.0"  
}
```

- Для более удобного управления конфигурацией и пакетами приложения в npm применяется файл конфигурации **package.json**.
- Удалить node_modules.
- --save указывает, что информацию о пакете нужно добавить в package.json

```
{  
  "name": "modulesapp",  
  "version": "1.0.0",  
  "dependencies": {  
    "express": "^4.14.0"  
  }  
}
```

```
npm install express --save
```

```
npm install
```



Vue is a **progressive framework** for building user interfaces.

1. Virtual DOM
2. Event Handling
3. Data Binding
4. Components
5. Computed Properties
6. Watchers

<https://ru.vuejs.org/>

- **Note:**

1. Vue **does not support IE8** and below
2. \$ npm install vue

Vue Instance

Contains:

1. Element
2. Data
3. Methods
4. Filters
5. Watchers
6. Lifecycle hooks
7. Computed properties

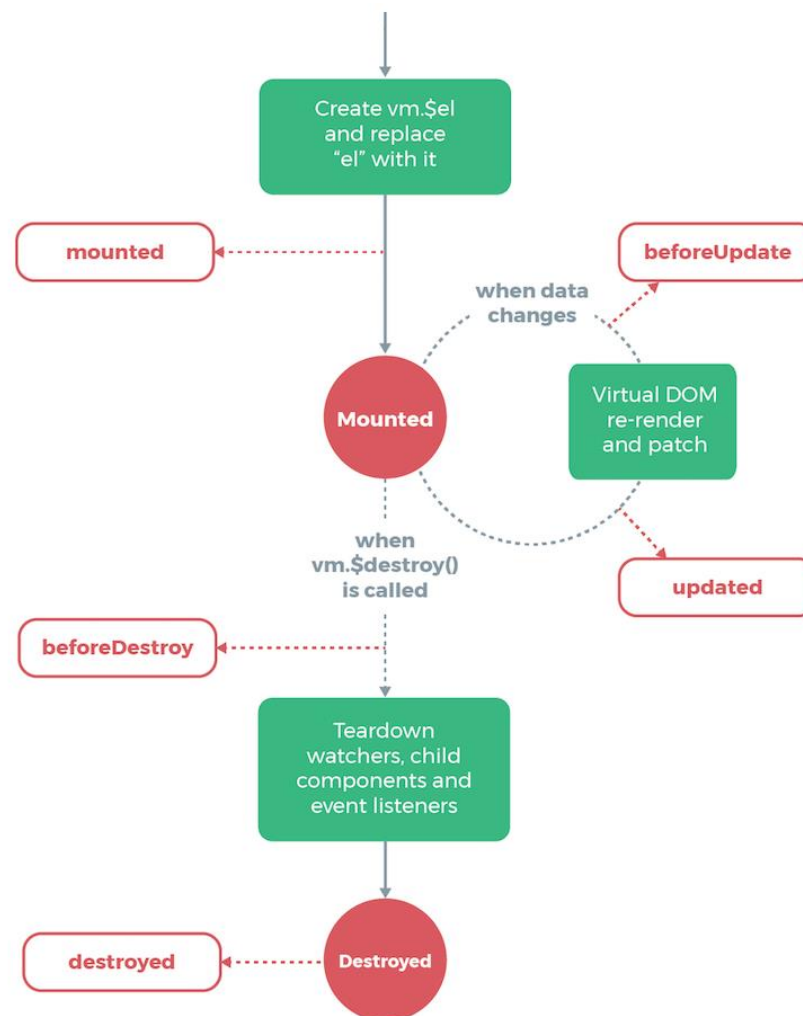
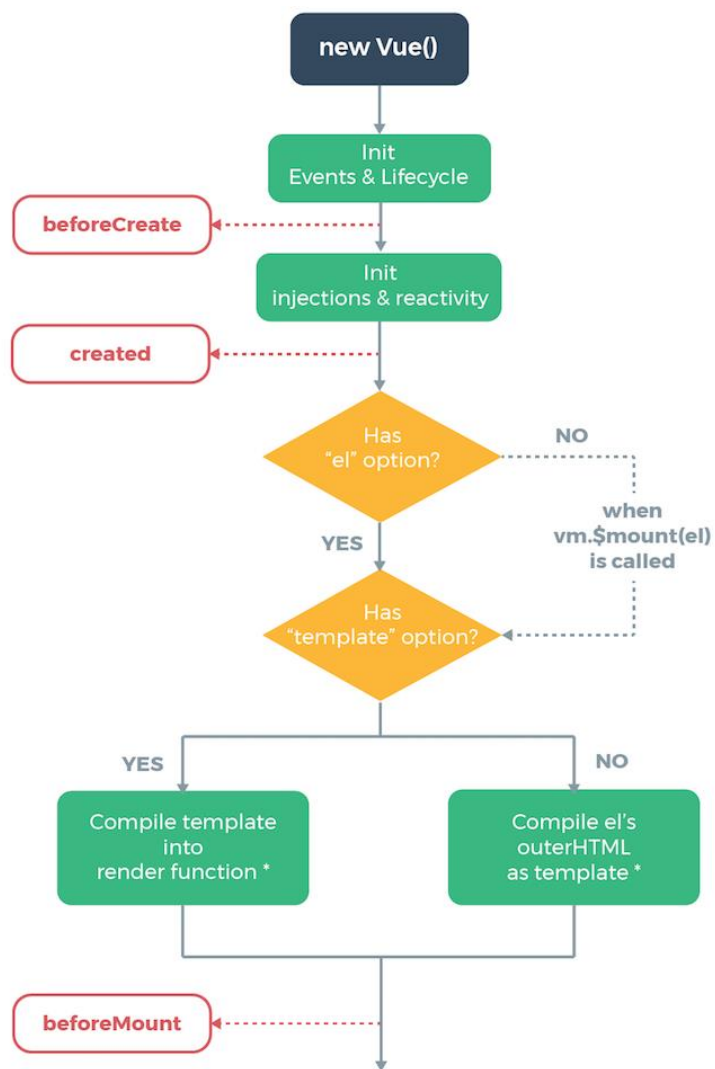
```
<div id="app-5">  
  <p>{{ message }}</p>  
  <button v-on:click="reverseMessage">Reverse Message</button>  
</div>
```

HTML

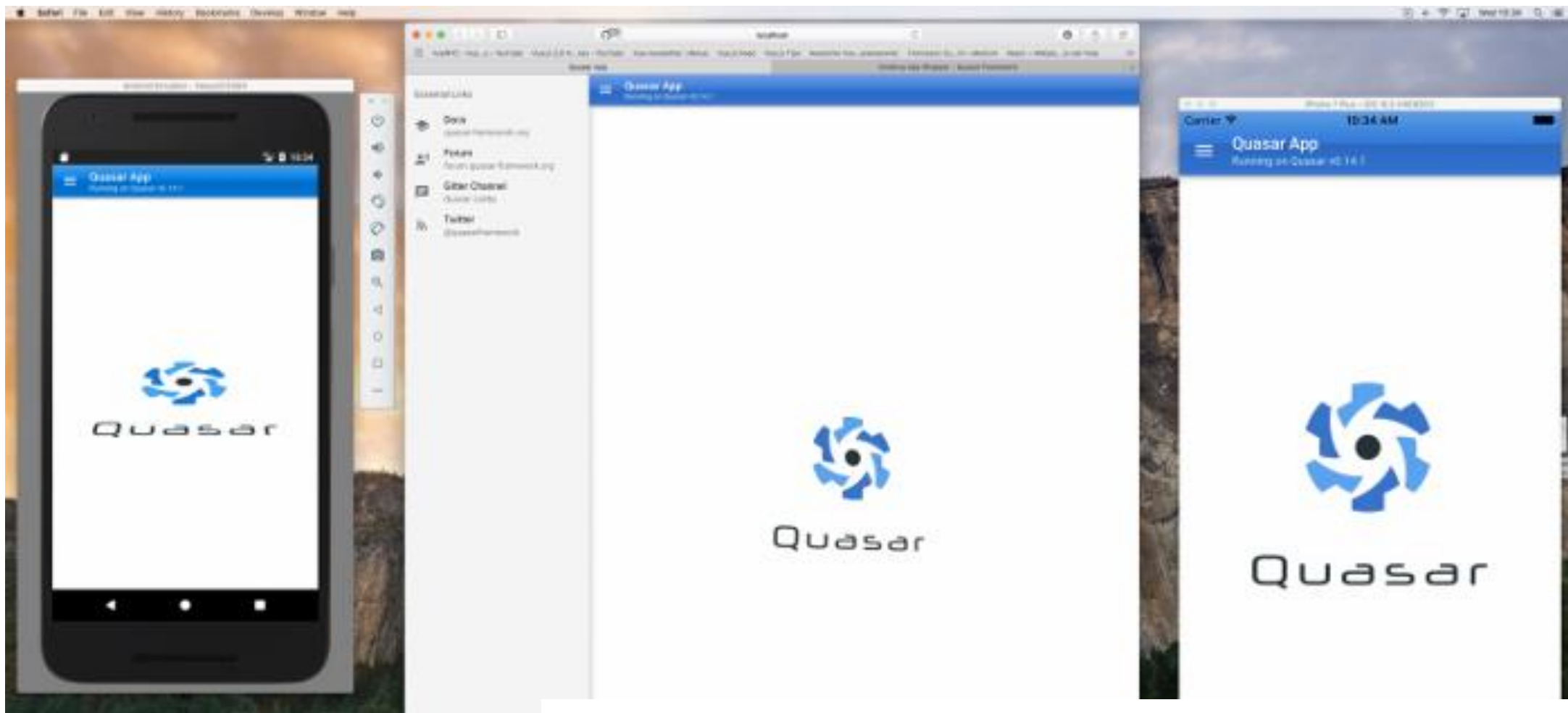
```
var app5 = new Vue({  
  el: '#app-5',  
  data: {  
    message: 'Hello Vue.js!'  
  },  
  methods: {  
    reverseMessage: function () {  
      this.message = this.message.split('').reverse().join('')  
    }  
  }  
})
```

JS

Lifecycle



Quasar.dev



All Platforms in One Go

[User Interface Components](#)



[SPA \(Single Page App\)](#)



[PWA \(Progressive Web App\)](#)



[BEX \(Browser Extension\)](#)



[SSR \(Server-side Rendered App\)](#)



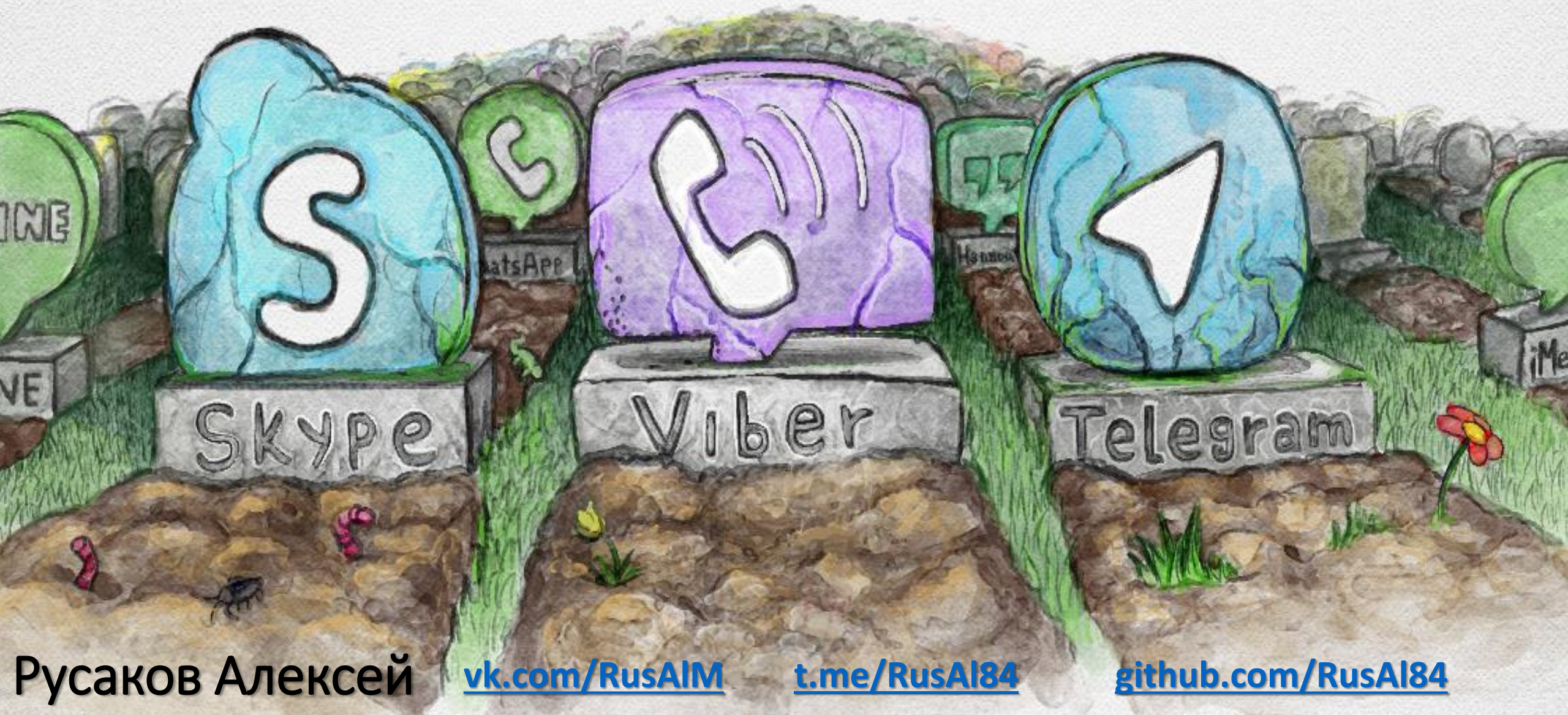
[Hybrid Mobile App](#)



[Multi-platform Desktop App](#)



Спасибо за внимание



Русаков Алексей

vk.com/RusAlM

t.me/RusAl84

github.com/RusAl84