



# Разработка приложения для интеграции gitlab и telegram

Группа компаний ХОСТ  
hostco.ru

Виктор Рыбаков  
Java разработчик

# План

1. Введение в предметную область и формулировка ТЗ
2. Ликбез по стеку разработки
3. Написание приложения
  - 3.1. Создание проекта
  - 3.2. Исследование api gitlab
  - 3.3. Написание части по взаимодействию с api gitlab
  - 3.4. Исследование api telegram
  - 3.5. Написание части по взаимодействию с api telegram
  - 3.6. Написание части по взаимодействию с БД
4. Ответы на общие вопросы

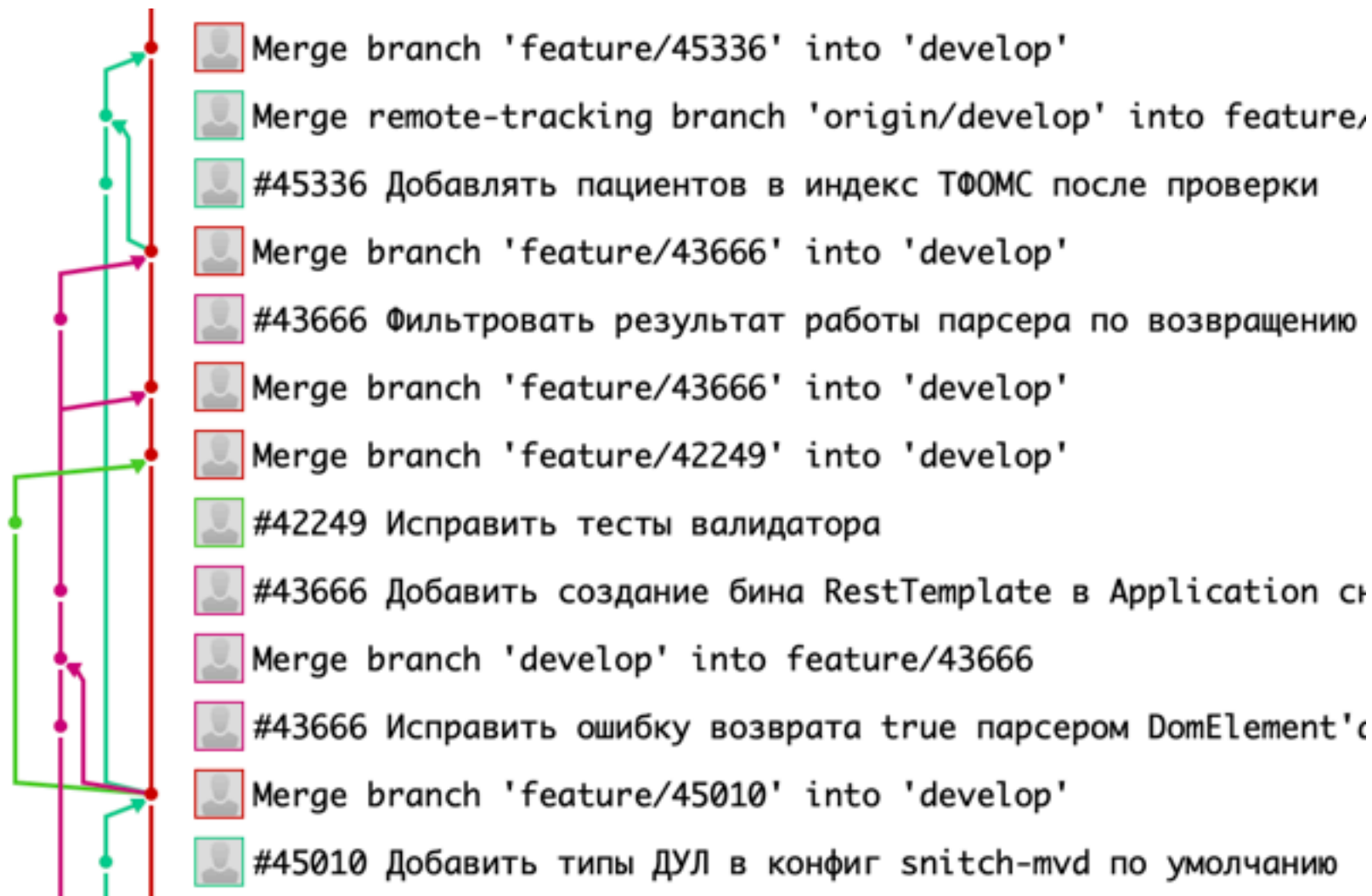


# Projects

Your projects 158
Starred projects 1
Explore projects


Filter by name...
Last updated

All	Personal
P	<div> dvs / patient_portal_telemed Maintainer </div> <p>Методы для телемеда (вынесено в отдельный сервис)</p> <div> ★ 0 Y 0 I 0 D 0 </div> Updated 21 minutes ago
P	<div> dvs / patient-portal-waitlist Maintainer </div> <p>Электронная очередь, является частью Портал пациента</p> <div> ★ 0 Y 0 I 0 D 0 </div> Updated 39 minutes ago
E	<div> dvs / emds Reporter </div> <p>Сервис для работы с хранилищем электронных медицинских документов</p> <div> ★ 0 Y 0 I 0 D 0 </div> Updated 45 minutes ago
P	<div> dvs / patient_portal_v3-0 Reporter </div> <p>Веб-портал Электронная регистратура 3.0 (фронт)</p> <div> ★ 0 Y 0 I 0 D 0 </div> Updated 5 minutes ago
	<div> dvs / ump Reporter </div> <p>Учет больных, претендующих на получение консультативной, специализированной и высокотехнологичной мед...</p> <div> ★ 1 Y 0 I 0 D 18 </div> Updated 1 hour ago
I	<div> dvs / iemk-syner Reporter </div> <p>Сервис отправки данных из МИС софтраст в центральный сервис ИЭМК</p> <div> ★ 0 Y 0 I 0 D 0 </div> Updated 59 minutes ago
P	<div> dvs / portal-microservices Maintainer </div> <p>Репозиторий микросервисов для работы с порталом пациента</p> <div> ★ 0 Y 0 I 1 D 0 </div> Updated 1 hour ago
	<div> bi / puomp_budg Developer </div> <p>ПУОМП (Б) – бюджетные реестры; Кроме прочего здесь хранится НСИ общее как для ПУОМП (Б) так и для аналь...</p> <div> ★ 2 Y 0 I 0 D 0 </div> Updated 2 hours ago
I	<div> dvs / iprp_ipra_back Reporter </div> <p>ИС Индивидуальная программа ранней помощи и ИПРА, разрабатывается на основе ИПРП (г. Самара) - BACK</p> <div> ★ 0 Y 0 I 0 D 0 </div> Updated 2 hours ago
I	<div> dvs / iemk-service-core Developer </div> <p></p> <div> ★ 1 Y 0 I 0 D 0 </div> Updated 3 hours ago



dvs > master-patient-index > Merge Requests > !167

Open

Opened 20 hours ago by  Rybakov Viktor

## Hotfix/40985

Overview 0

Commits 15

Changes 34



**Request to merge** [hotfix/40985](#)  **into** [develop](#)

The source branch is [122 commits behind](#) the target branch

Open in



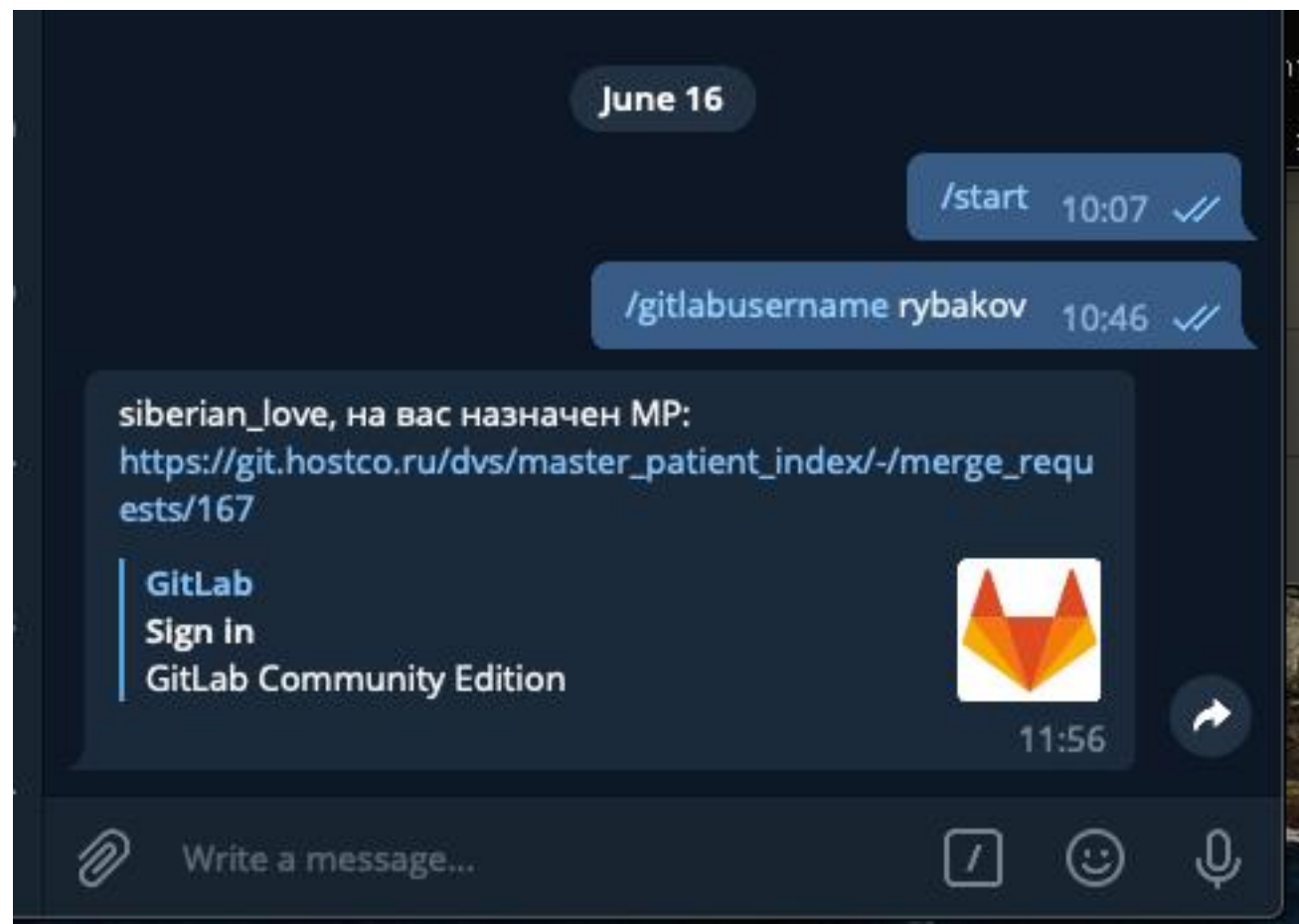
Merge

There are merge conflicts

Merge locally

You can merge this merge request manually using the [command line](#)





# JAVA

## Язык программирования

Объектно-ориентированный язык программирования, компилируемый в байт-код и выполняемый Java машиной. В отличие от, например, C++ в java отсутствует возможность явного удаления объектов из памяти. Вместо этого работает сборщик мусора.

Реализация JDK 11 от Amazon:

<https://docs.aws.amazon.com/corretto/latest/corretto-11-ug/downloads-list.html>





# IntelliJ IDEA

## Среда разработки

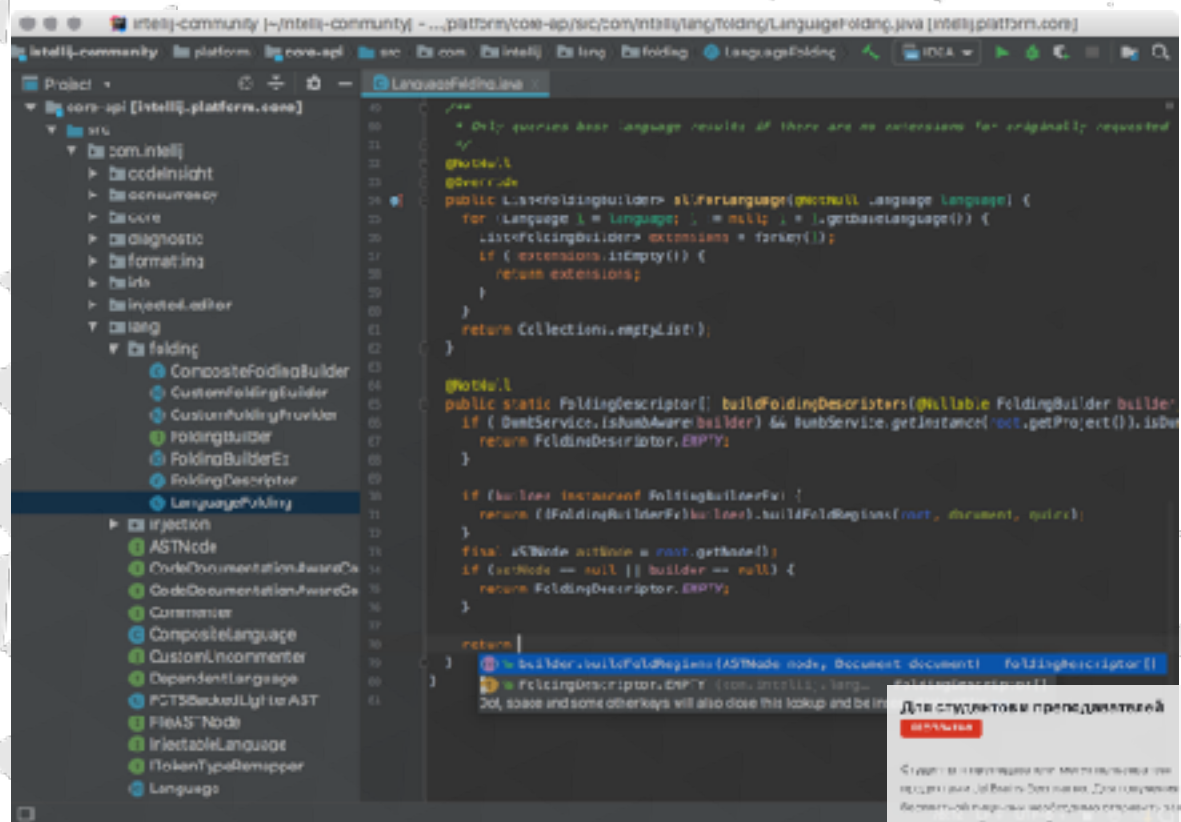
Доступны две версии: Community и Ultimate

Community бесплатная, имеет открытый исходный код. Ultimate бесплатно доступна по специальным предложениям или после покупки лицензии по подписке.

Студенческую лицензию можно получить через GitHub Student Developer Pack, получить которую достаточно отправив фото студенческого.

Ссылка для скачивания:

<https://www.jetbrains.com/ru-ru/idea/>





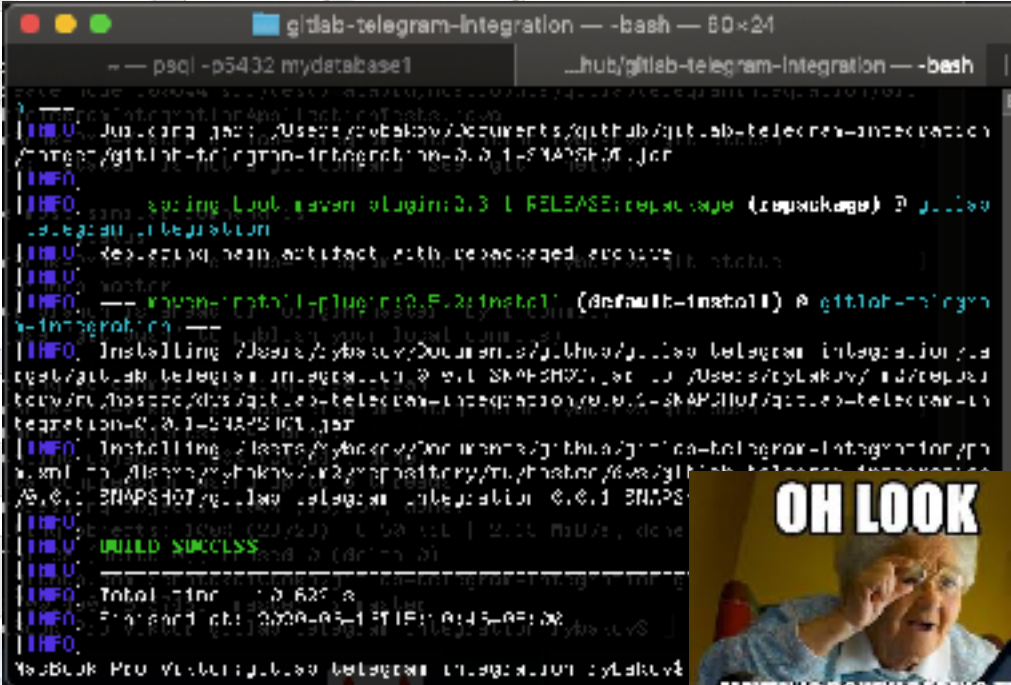
# Maven

## Автоматизатор сборок

До мавена разработчикам приходилось писать «javac -d bin src/ru/hostco/Class1.java src/ru/hostco/Class2.java src/ru/hostco/Class3.java». При мавене разработчики пишут «mvn clean install».

Ссылка для скачивания:

<https://maven.apache.org>



```
gltab-telegram-integration — bash — 80x24
-- psql -p5432 mydatabase1 _hub/gitlab-telegram-integration — bash
gitlab-telegram-integration:~$ mvn clean install
[INFO] Scanning for projects...
[INFO]
[INFO] --- maven-clean-plugin:3.1.0:clean (default-clean) @ gitlab-telegram-integration ---
[INFO] Deleting /Users/tylakov/Documents/gitlab-telegram-integration/target
[INFO]
[INFO] --- maven-resources-plugin:3.1.0:resources (default-resources) @ gitlab-telegram-integration ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] Copying 1 resource
[INFO]
[INFO] --- maven-compiler-plugin:3.8.1:compile (default-compile) @ gitlab-telegram-integration ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to /Users/tylakov/Documents/gitlab-telegram-integration/target/classes
[INFO]
[INFO] --- maven-surefire-plugin:2.22.2:test (default-test) @ gitlab-telegram-integration ---
[INFO]
[INFO] --- maven-jar-plugin:3.1.2:jar (default-jar) @ gitlab-telegram-integration ---
[INFO] Building jar: /Users/tylakov/Documents/gitlab-telegram-integration/target/gitlab-telegram-integration-0.0.1-SNAPSHOT.jar
[INFO]
[INFO] --- maven-install-plugin:3.0.1:install (default-install) @ gitlab-telegram-integration ---
[INFO] Installing /Users/tylakov/Documents/gitlab-telegram-integration/target/gitlab-telegram-integration-0.0.1-SNAPSHOT.jar to /Users/tylakov/.m2/repository/ru/hostco/gitlab-telegram-integration/0.0.1-SNAPSHOT/gitlab-telegram-integration-0.0.1-SNAPSHOT.jar
[INFO]
[INFO] --- maven-deploy-plugin:3.0.0:deploy (default-deploy) @ gitlab-telegram-integration ---
[INFO]
[INFO] Total time: 0.628 s
[INFO] Finished at: 2020-03-11T15:04:50+03:00
[INFO]
[INFO] Your build is successful
[INFO]
[INFO] Total time: 0.628 s
[INFO] Finished at: 2020-03-11T15:04:50+03:00
[INFO]
[INFO] Your build is successful
```



# Spring

## Java фреймворк

Фреймворк, предназначенный для enterprise разработки приложений. Задумывался, как альтернатива Java EE. Основной концепцией является управление зависимостями и циклом жизни некоторых объектов (бинов). Основой приложения является контейнер (по аналогии с банкой), в котором и хранятся бины (фасолины). Spring связывает их сам, на основании определения бинов.

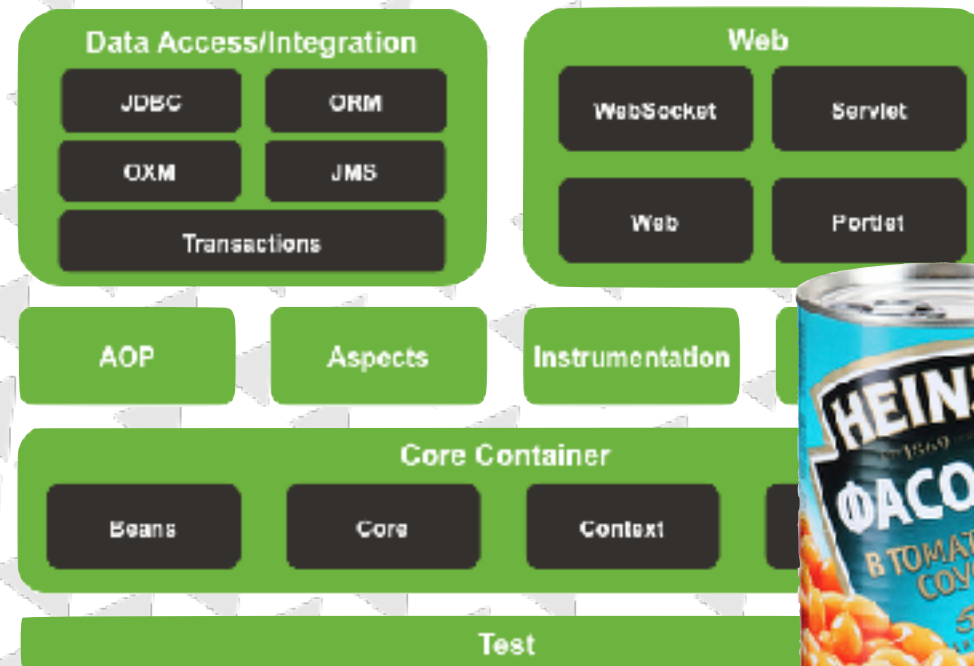
Создание проекта на основе spring:

<https://spring.io/quickstart>

<https://start.spring.io>



Spring Framework Runtime



# Spring-boot

## Фреймворк для фреймворка

А такое бывает!? Несомненно. Spring-boot является своего рода надстройкой над spring для ускорения процесса разработки и возможности атвоконфигурирования приложений. То, на что в классическом spring mvc нужно было строк 500 xml конфигурации - теперь можно не конфигурировать руками.

Создание проекта на основе spring:

<https://spring.io/quickstart>

<https://start.spring.io>

```
import ...  
  
@SpringBootApplication  
public class Reservations {  
  
    public  
    (Reservati  
}
```





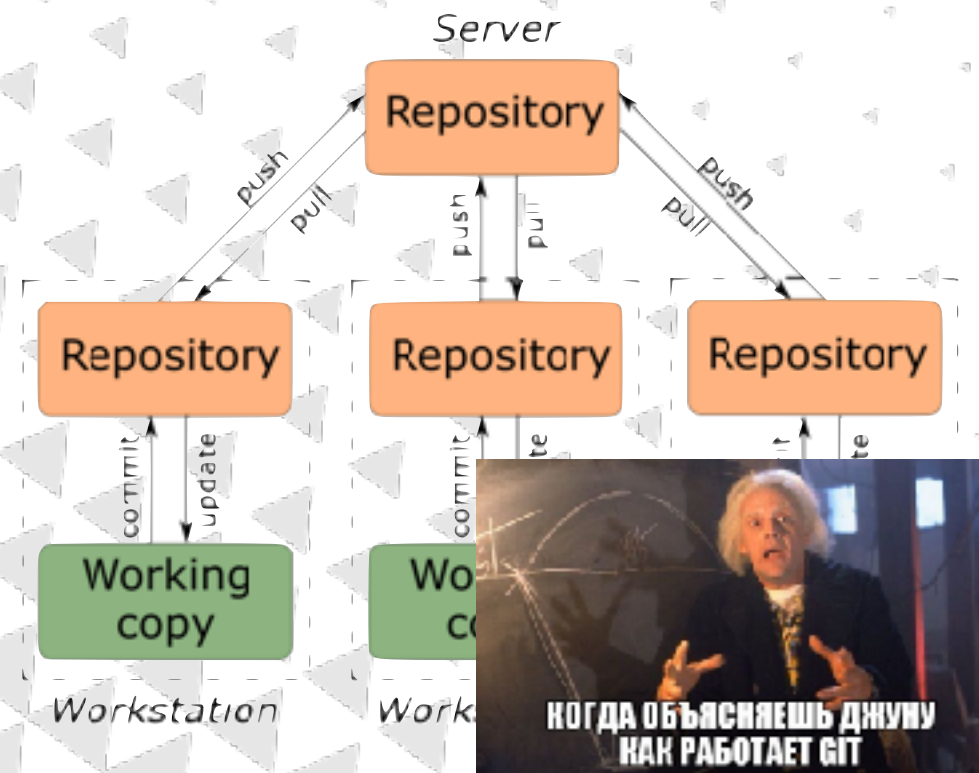
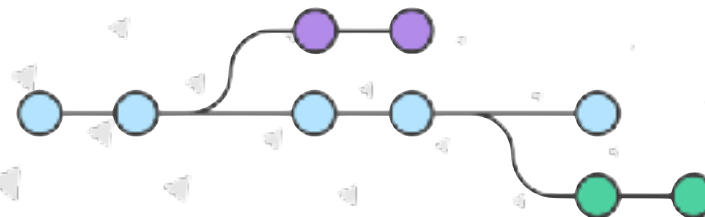
# Git

## Управлятор версиями кода

Гит следит за тем, что разработчики делают с кодом. Основа того, что разработчики делают - коммиты. Коммит - атомарный набор изменений кода. Именно по коммитам в гите можно отследить, кто написал плохой код и потыкать негодяя в плохой код. Есть ещё ветки. Ветка - указатель на коммит. Как можно понять из названия - ветки могут иметь и общие и разные наборы коммитов. Сложно? Тогда нужно просто представить деревянное дерево, ветки которого состоят из каких-то коммитов.

Ссылка для скачивания:

<https://git-scm.com>



# Postgresql

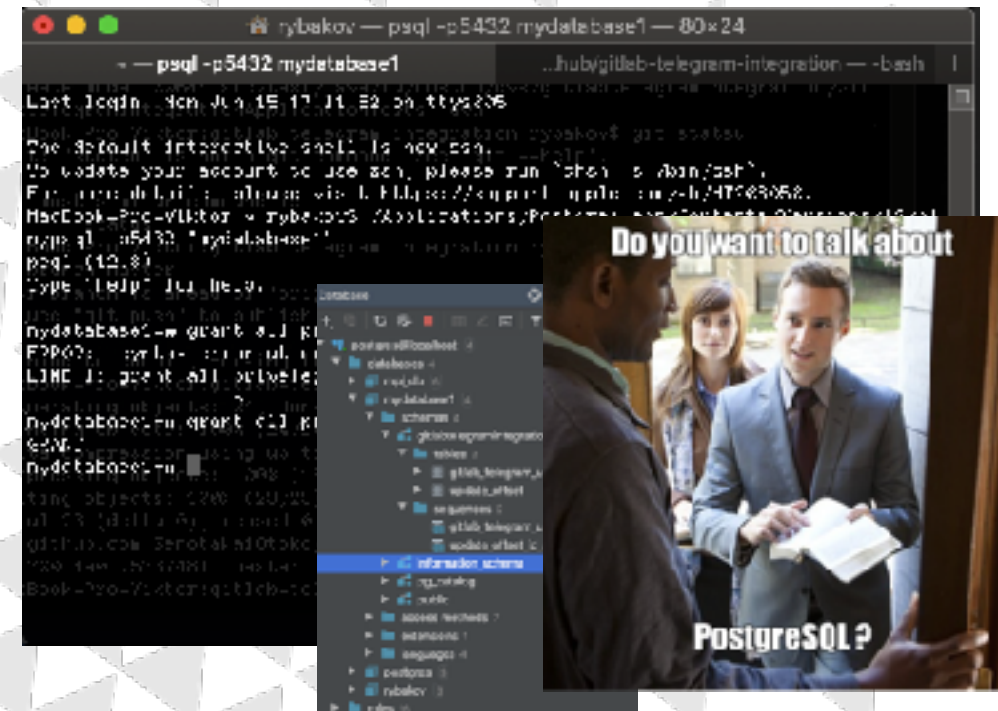
## Очередная база данных

Если не вылезать за пределы ANSI SQL, то можно никогда не узнать, чем postgresql отличается от других sql баз данных. А отличается она существенно. Множество возможностей по работе с BigData чего только стоят. Возможность написания кастомных типов данных. БЕСПЛАТНОСТЬ.

Ссылки для скачивания:

<https://www.postgresql.org/download/>

<https://sourceforge.net/projects/postgresqlportable/>

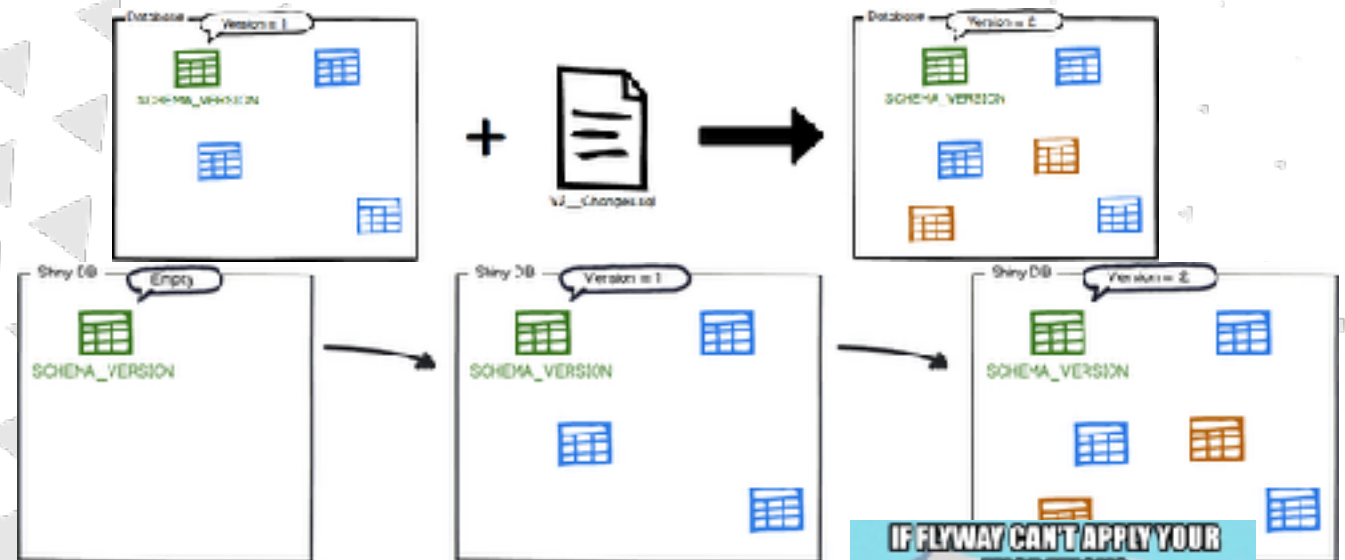


# Flyway

## Управление версиями БД

В процессе разработки у приложения бывают разные версии. И как водится, объекты БД (таблицы, функции, представления) между разными версиями приложения могут меняться. Для того, чтобы эти изменения централизованно производить и отслеживать и существует flyway.

Flyway существует как в виде самостоятельного приложения, так и в виде плагина для maven и в виде библиотеки для java приложения.

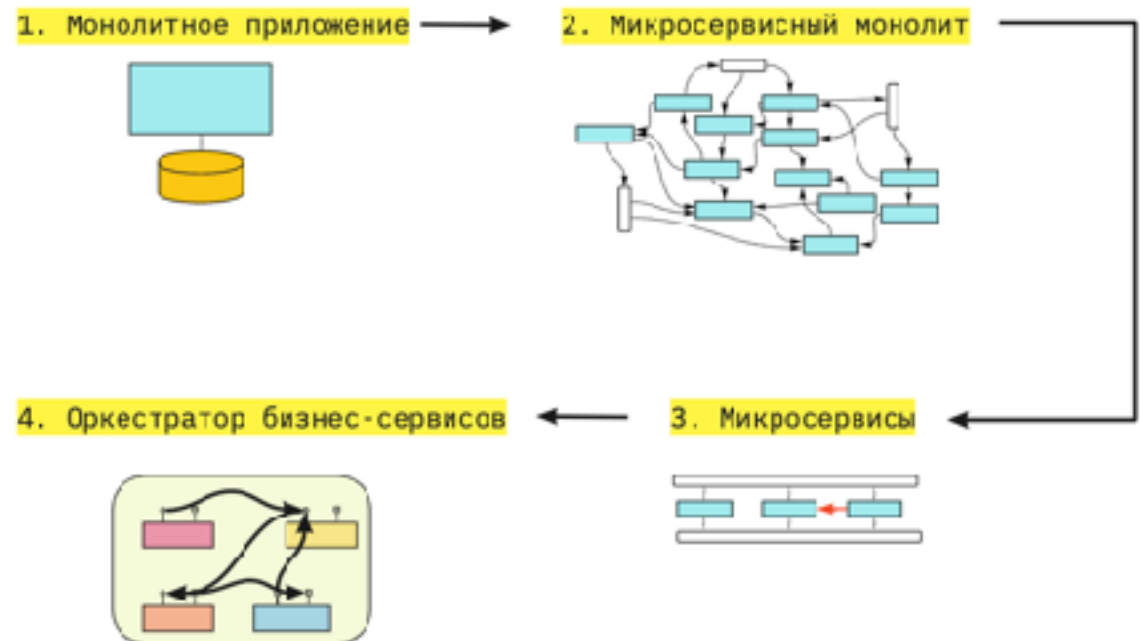




# Микросервисы

## Архитектура приложения

В общем случае микросервис - маленькое приложение, выполняющее одну функцию. Например, авторизацию пользователей или работу с сообщениями пользователей. В дальнейшем определения расходятся. Кто-то считает, что если все микросервисы работают на одной БД, то это микросервисы, а кто-то, что нет. Кто-то считает, что три микросервиса должны работать только через брокеры сообщений, а не с REST API друг друга. Архитектура хороша масштабируемостью и независимостью всех модулей от одного ЯП. Требуется чуть больших знаний при разработке, чтобы не накуролесить по сравнению с монолитом.



## Первые микросервисы



## Микросервисы



## Микросервисный монолит

