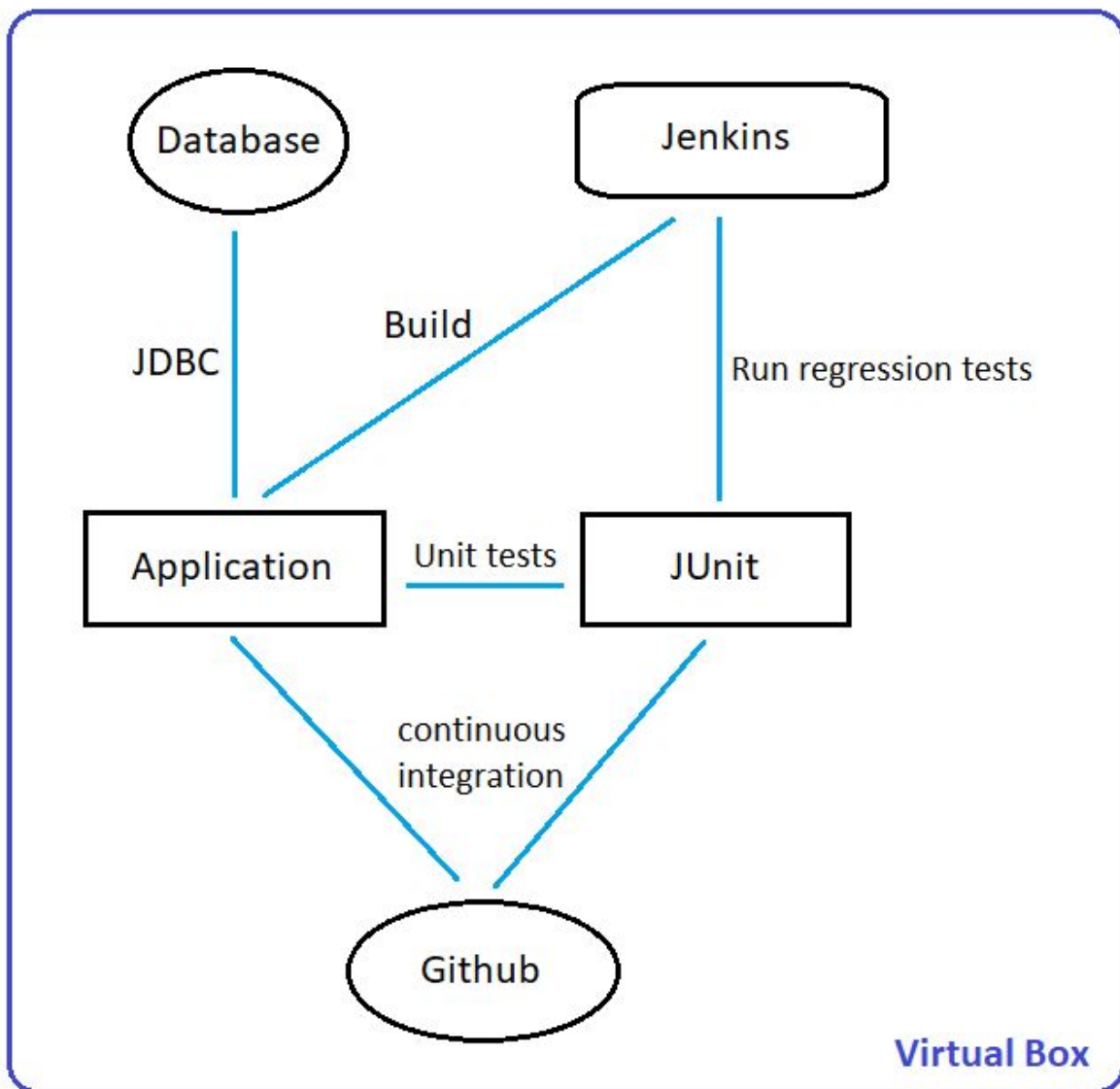


Dokumentacja



Projekt składa się z:

- Baza danych MySql
- Jenkins
- Aplikacji napisanej w języku Java:
 - Spark (<http://sparkjava.com/>)
 - GSON (serializacja/deserializacja danych)
 - junit (testy jednostkowe)
 - json (json parser)
 - log4j (logger)

- Maven

- System kontroli wersji GIT
- IDE IntelliJ Ultimate
- System operacyjny Ubuntu
- Virtual Box

Opis:

Wszystkie komponenty połączone są jak na schemacie wyżej. Komunikacja jest dwustronna. Aplikacja to kantor, który łączy się do bazy danych. Oprócz tego aplikacja umożliwia wykonywanie transakcji między kontami w oparciu o zapytania REST. Do testowania oprogramowania użyta została biblioteka JUnit (testy jednostkowe). Wszystko jest oparte o tzw. "Continuous Integration", czyli po każdym commicie uruchamiane są testy jednostkowe w celu sprawdzenia, czy wszystko dobrze się buduje. Cały projekt oparty jest o system kontroli wersji GIT. Cały projekt został stworzony na systemie operacyjnym Ubuntu. Sam system operacyjny został zainstalowany jako obraz pod Virtual Boxa.

Na komputerze należy mieć:
JDK 8, Maven, IntelliJ Ultimate

Kompilacja aplikacji:
mvn clean install

Uruchomienie aplikacji po zaimportowaniu do IntelliJ Ultimate:
src->main->java->com->moneyapp->utils->Application->Run "Application.main()"

Instalacja Jenkins:
<https://linuxize.com/post/how-to-install-jenkins-on-ubuntu-18-04/>

Następnie:
localhost:8080

Następnie należy ustawić webhook:
<https://www.youtube.com/watch?v=Z3S2gMBUkBo>

Komendy git -> commit na repo:
git add *
git commit -m "treść commita"
git push

Wejście do MySQL:
sudo mysql -u root -p

Podgląd dostępnych baz danych:
show databases;

Użycie konkretnej bazy:

```
use Bank;
```

Podgląd tabel w bazie danych (będąc już podpiętym do konkretnej bazy):

```
show tables;
```

Podgląd tabeli:

```
select * from Cantor;
```

SQL:

```
CREATE DATABASE Bank;
```

```
use Bank;
```

```
CREATE TABLE Concurrency (  
    id int,  
    name varchar(255),  
    converterToDolar float,  
    code varchar(255)  
);
```

```
INSERT INTO Concurrency (id, name, converterToDolar, code)  
VALUES (1, EURO, 1.15, EUR);
```

```
INSERT INTO Concurrency (id, name, converterToDolar, code)  
VALUES (2, DOLAR, 0, DOL);
```

```
INSERT INTO Concurrency (id, name, converterToDolar, code)  
VALUES (3, POLISH_ZLOTY, 0.27, PLN);
```

```
INSERT INTO Concurrency (id, name, converterToDolar, code)  
VALUES (4, BRITISH_POUND, 1.29, GBP);
```

VirtualBox:

Wersja 5.2.12 r122591 (Qt5.6.2)