**Этапы:**

1. Импорт БД/ Создание БД
2. Настройка сервера
3. Настройка клиента
4. Запуск сервера
5. Запуск клиента
6. Работа в клиенте
7. Юнит тест

**1) Импорт БД**

Прежде чем начать, необходимо установить PostgreSQL.

Развёртывание БД производится при помощи команд:

|  |
| --- |
| \Posgtresql\12\bin\psql.exe -U postgres -d postgres -c "CREATE DATABASE meetings"  \Posgtresql\12\bin \psql.exe -U postgres meetings < meetings.sql |

Альтернативным способом может служить создание новой БД, (но тогда юнит тест будет выдавать ошибки, из-за уже имеющихся сохранённых данных). Создание и открытие БД:

|  |
| --- |
| \Posgtresql\12\bin\psql.exe -U postgres -d postgres -c "CREATE DATABASE meetings"  \Posgtresql\12\bin\psql.exe meetings |

Создание таблиц:

|  |
| --- |
| CREATE TABLE meeting(  id serial PRIMARY KEY,  name VARCHAR(30) NOT NULL,  date DATE NOT NULL,  time TIME NOT NULL  );  CREATE TABLE participant(  id serial PRIMARY KEY,  name VARCHAR(30) NOT NULL,  surname VARCHAR(30) NOT NULL,  email VARCHAR(30) NOT NULL,  meetingID INTEGER REFERENCES meeting(id) NOT NULL  ); |

При возникновении проблем с доступом, можно в файле pg\_hba.conf задать метод доступа trust. Очевидно, что не стоит хостить БД с таким типом доступа.

**2) Настройка сервера**

У сервера необходимо задать название БД, имя пользователя БД, пароль БД, адрес, где она находится, порт доступа к БД (строки 5-6, server.py):

|  |
| --- |
| conn = psycopg2.connect(dbname='meetings', user='Nikita',  password='', host='localhost', port="5432") |

А так же сами параметры сервера. Адрес и порт доступа находятся на 148 строке server.py

|  |
| --- |
| def run(server\_class=HTTPServer, handler\_class=AvitoMeetingsServer, addr="localhost", port=8000): |

**3) Настройка клиента**

У клиента аналогичным образом необходимо указать адрес и порт (138 строка client.py)

|  |
| --- |
| myurl = "http://localhost:8000" |

**4) Запуск сервера**

Запуск сервера производится при помощи команды:

|  |
| --- |
| python server.py |

**5) Запуск клиента**

Запуск клиента производится при помощи команды в новом терминале:

|  |
| --- |
| python client.py |

**6) Работа в клиенте**

Управление клиентом производится при помощи ввода кодов команд и подтверждения клавишей Enter. Затем, могут потребоваться дополнительные данные. Клавиши основных команд

1. Назначить встречу. Необходимо ввести название, дату и время.
2. Отменить встречу. Необходимо ввести название встречи.
3. Пригласить участника на встречу. Необходимо ввести имя, фамилию, электронную почту и название встречи.
4. Отменить приглашение. Необходимо ввести имя, фамилию и название встречи.
5. Показать все встречи и участников.
6. Завершить работу.

Любая другая команда приведёт к выводу справки.

**7) Юнит тест**

Для проверки работоспособности и отсутствия ошибок проверяется следующий сценарий (файл testclient.py):

1. Начало работы.
2. Добавление новой встречи с проверкой успешности.
3. Добавление трёх участников с проверкой успешности.
4. Вывод всех встреч и сверка с ожиданием.
5. Отмена приглашения одного участника с проверкой успешности.
6. Вывод всех встреч и сверка с ожиданием.
7. Отмена встречи с проверкой успешности.
8. Вывод всех встреч и сверка с ожиданием.

Юнит тест проходит без ошибок, что при дальнейшем тестировании может послужить доказательством работоспособности программного продукта.