

Отчет по Лабораторной работе №3 ПО сервиса по ремонту техники.

Выполнили: студенты группы ПИН-37 Джалилова Ксения, Семенов Андрей, Убоженко Матвей.

Составить перечень технологий, используемых для реализации варианта задания.

Пользовательский интерфейс: HTML, CSS, JavaScript

Язык программирования: Python

Фреймворк для backend: Django

Система управления базами данных: SQLite

Обосновать выбор данных технологий и удобство их использования.

1. HTML, CSS, JavaScript

- Основные и наиболее распространённые технологии для разработки веб-интерфейсов;
- Большое количество учебных материалов, документации и примеров;
- Поддержка современными средствами разработки (IntelliJ IDEA, VS Code, Eclipse);
- Хорошая совместимость с backend-фреймворками.

2. Python + Django

- Django обеспечивает быстрое создание надёжных и безопасных веб-приложений;
- Встроенная ORM упрощает взаимодействие с базой данных;
- Чёткое разделение слоёв логики (MVC-подход);
- Большое сообщество и богатая экосистема библиотек.

3. PostgreSQL

- Надёжное и безопасное хранение данных;
- Простая установка и настройка;

- Поддержка сложных SQL-запросов, индексов, триггеров и хранимых процедур;
- Хорошая интеграция с Django.

Декомпозировать разрабатываемую систему, описать модули, необходимые для реализации ПО согласно заданию.

Управление пользователями

```
public enum Role {  
    client,  
    master,  
    admin  
}  
  
public abstract class User {  
    private int userID;  
    private String firstName;  
    private String lastName;  
    private String phoneNumber;  
    private String email;  
    private String login;  
    private String password;  
    private Role role;  
  
    public void register();  
    public void login();  
    public void logout();  
    public String getFullName();  
    public void updateContactInfo();  
}
```

```
public class Client extends User{  
    public void createOrder();  
    public void viewOrderStatus();  
    viewOrderHistory();  
}
```

```
public class Master extends User{  
    viewAssignedOrders()  
    public void takeOrder();  
    public void updateOrderStatus();  
    public Estimate createEstimate();  
}
```

```
public class Administrator extends User{  
    public void assignOrder();  
    public void manageWarehouse ();  
    public void addToWarehouse ();  
    public void assignUserRole();  
}
```

Управление заказами

```
public enum Status {  
    ОЖИДАЕТ_ПОДТВЕРЖДЕНИЯ,  
    ПРИНЯТ,  
    ДИАГНОСТИКА,  
    ГОТОВО,  
    ВЫДАНО  
}  
  
public class Order {
```

```
    private int id;
    private int Client.id;
    private int Master.id;
    private int Device.id;
    private String description;
    private Status status;
    private DateTime createdAt;
    private DateTime updatedAt;

    public void changeStatus();
    public void attachEstimate();
}

public class Device {
    private int id;
    private String type;
    private String brand;
    private String model;
    private String category;
    public void updateDeviceInfo();
}

public class SparePart {
    private int id;
    private String type;
    private String model;
    private String brand;
    private int cost;
    private int quantity;
```

```
    public void updateQuantity();  
}
```

Управление отчетностью

```
public class Estimate {  
    private int id;  
    private Order id;  
    private float totalCost;  
    private float partsCost;  
    private float laborCost;  
    private Date createdAt;  
    public void calculateTotalCost();  
}
```

Сформулировать набор задач, необходимых для реализации ПО согласно варианту задания, определить порядок выполнения и приоритет каждой из них.

Подготовительный этап:

1. Создать Django-проект и приложение (приоритет: высокий)
2. Настроить подключение к базе данных SQLite (высокий)
3. Настроить виртуальное окружение и необходимые зависимости (высокий)

Моделирование данных:

1. Создать модели
2. Настроить связи между моделями
3. Создать миграции и применить их

Основной функционал:

1. Реализовать регистрацию, вход и выход пользователей
2. Реализовать создание, просмотр и историю заказов для клиентов
3. Реализовать назначение ролей пользователям для администратора
4. Реализовать добавление устройств и деталей администратором
5. Реализовать возможность мастера брать заказ, создавать смету и менять статус заказа

Пользовательский интерфейс:

1. Главная страница и навигация
2. Разработать страницы для регистрации и входа
3. Разработать страницу создания заявки
4. Разработать страницы для просмотра заказов и истории
5. Разработать интерфейс для мастера
6. Разработать интерфейс для администратора

Дополнительный функционал

1. Модуль отчетности
2. Система уведомлений

6. Запустить Git GUI (или консоль).

Создать новый репозиторий.

```
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git init
Initialized empty Git repository in /mnt/c/Users/kseni/kpo_labs/.git/
```

Добавить в папку репозитория файлы. Зафиксировать состояние репозитория (выполнить commit).

```
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git add .
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git commit -m "Add commit"
```

```
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git add README.md
```

7. Внести изменения в файлы. Зафиксировать новое состояние репозитория.

```
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ nano README.md
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git commit -m "Add and change README.md"
[master 6791e5e] Add and change README.md
1 file changed, 3 insertions(+)
create mode 100644 README.md
```

8. Создать новую ветку 1. Внести в нее изменения и зафиксировать их.

```
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git branch branch_1
```

```
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git checkout branch_1
Switched to branch 'branch_1'
```

```
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ nano README.md
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git add .
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git commit -m "Add branch_1 and change README.md"
[branch_1 b92f797] Add branch_1 and change README.md
1 file changed, 2 insertions(+)
```

9. Переключиться на ветку мастера. Внести в нее изменения и зафиксировать их.

```
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git checkout master
Switched to branch 'master'
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ mkdir lab_3
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git add .
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git commit -m "Change in branch master and create folder lab_3"
On branch master
nothing to commit, working tree clean
```

10. Продемонстрировать слияние веток.

```
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git merge branch_1
Updating 6791e5e..b92f797
Fast-forward
 README.md | 2 ++
 1 file changed, 2 insertions(+)
```

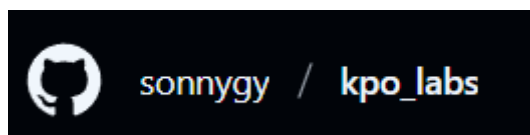
11. Просмотреть дерево изменений веток (историю).

```
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git log --oneline --all
b92f797 (HEAD -> master, branch_1) Add branch_1 and change README.md
6791e5e Add and change README.md
fd24180 Add commit
```

--oneline - показывает каждый коммит одной строкой.

--all - показывает все ветки.

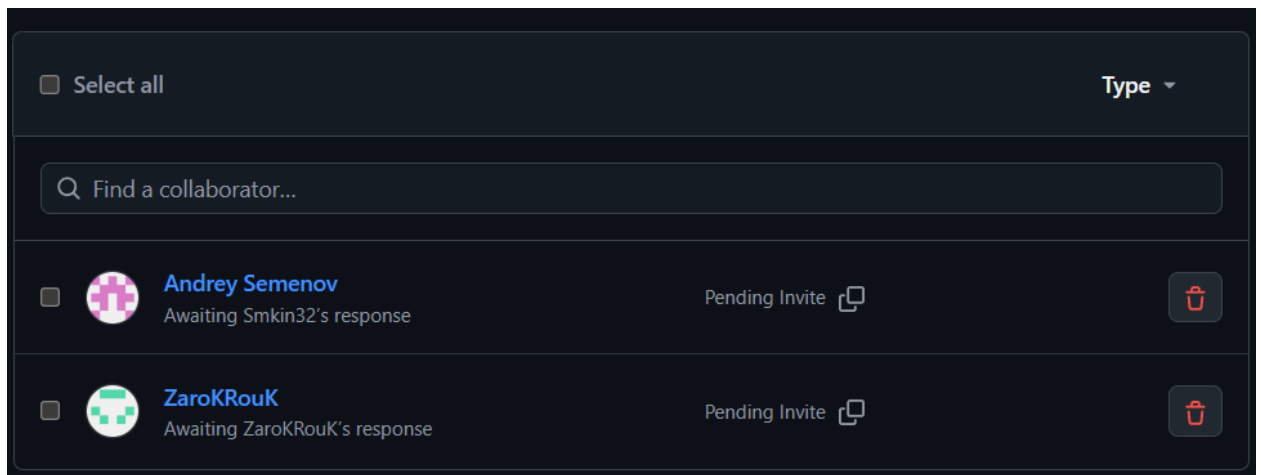
12. Создать удаленный репозиторий.



13. Загрузить на него свой проект.

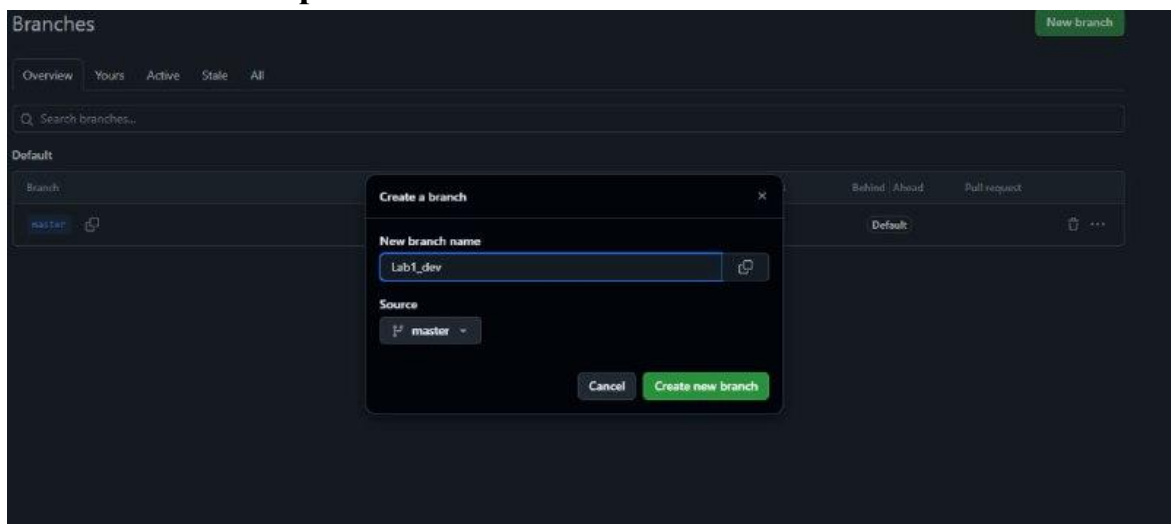
```
ksenii@LAPTOP-88IBKUQ5:/mnt/c/Users/kseni/kpo_labs$ git push -u origin master
Username for 'https://github.com': sonnygy
Password for 'https://D1%8Bsonnygy@github.com':
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 12 threads
Compressing objects: 100% (13/13), done.
Writing objects: 100% (15/15), 213.62 KiB | 3.45 MiB/s, done.
Total 15 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/sonnygy/kpo_labs.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

14. Обеспечить доступ к нему все членам команды.

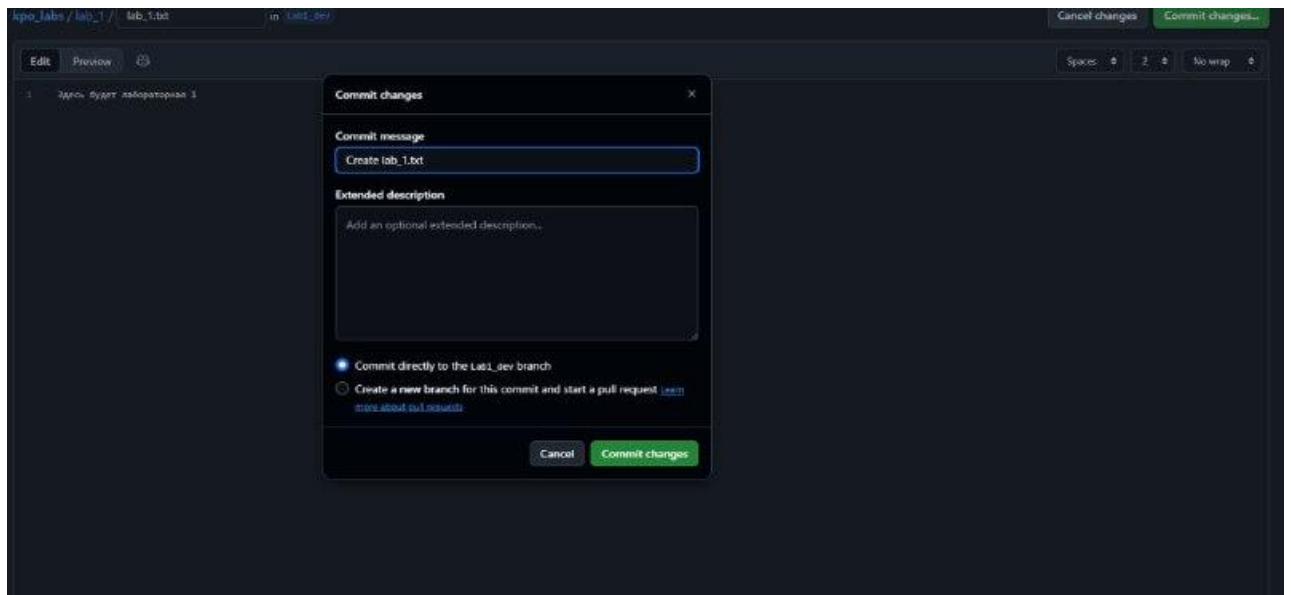


15. Скачать каждым членом команды файлы с удаленного репозитория, произвести

изменения и совершить слияние.



The screenshot shows the 'Commit changes' dialog box in VS Code. The commit message is 'Create lab_1.txt'. The extended description field is empty. The 'Commit directly to the Lab1_dev branch' option is selected. The 'Commit changes' button is highlighted in green.



Create lab_1.txt #3

Merged Smkin32 merged 1 commit into `master` from `Lab1_dev` now

Conversation 0

Commits 1

Checks 0

Files changed 1



Smkin32 commented now

Collaborator ...

No description provided.



Create lab_1.txt

Verified

68d3711



Smkin32 merged commit cea59a7 into master now

Revert



Pull request successfully merged and closed

Delete branch

You're all set — the `Lab1_dev` branch can be safely deleted.



Add a comment

Write

Preview



Add your comment here...

Markdown is supported

Paste, drop, or click to add files

Comment

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

```
zarok@localhost:~/kpo_labs$ mkdir lab_3
zarok@localhost:~/kpo_labs$ cd lab_3
zarok@localhost:~/kpo_labs/lab_3$ touch lab_3.md
zarok@localhost:~/kpo_labs/lab_3$ ls -la
total 8
drwxr-xr-x 2 zarok zarok 4096 Oct 31 13:27 .
drwxr-xr-x 5 zarok zarok 4096 Oct 31 13:26 ..
-rw-r--r-- 1 zarok zarok   0 Oct 31 13:27 lab_3.md
zarok@localhost:~/kpo_labs/lab_3$ cd ..
zarok@localhost:~/kpo_labs$ git status
On branch master
Your branch is up to date with 'origin/master'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  lab_3/

nothing added to commit but untracked files present (use "git add" to track)
zarok@localhost:~/kpo_labs$ git add lab_3/
zarok@localhost:~/kpo_labs$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   lab_3/lab_3.md

zarok@localhost:~/kpo_labs$ git commit -m "Добавлена лабораторная работа 3"
[master 9384904] Добавлена лабораторная работа 3
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 lab_3/lab_3.md
zarok@localhost:~/kpo_labs$ git push origin master
Username for 'https://github.com': ZaroKRouK
Password for 'https://ZaroKRouK@github.com':
remote: Invalid username or token. Password authentication is not supported for Git operation
fatal: Authentication failed for 'https://github.com/sonnygy/kpo_labs.git/'
zarok@localhost:~/kpo_labs$ git push origin master
Username for 'https://github.com': ZaroKRouK
Password for 'https://ZaroKRouK@github.com':
remote: Invalid username or token. Password authentication is not supported for Git operation
fatal: Authentication failed for 'https://github.com/sonnygy/kpo_labs.git/'
zarok@localhost:~/kpo_labs$ git push origin master
Username for 'https://github.com': ZaroKRouK
Password for 'https://%D0%AFZaroKRouK@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 16 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 396 bytes | 396.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/sonnygy/kpo_labs.git
 e5242e8..9384904 master -> master
```