**Руководство программиста**

**«**Информационной системы для кофейни»

1. **Инсталляция программы:**
   1. **Пользовательский вариант**
      1. Скачать файлы для работоспособности программы, которые находится по ссылке <https://github.com/Duudarra/Leyla-Darya/tree/main/program>
      2. Для программы требуется файл tovari3.
      3. Проверить приложение.
   2. **Вариант поддержания и модернизации**
2. Убедитесь, что на вашем компьютере установлен Python, с библиотекой PyQt5.
3. Склонируйте репозиторий программы с GitHub с помощью команды git clone <https://github.com/Duudarra/Leyla-Darya/tree/main/program>
4. Перейдите в папку вашего проекта с помощью команды cd repository.
5. Установите зависимости проекта, выполнив команду yarn install.
6. Запустите программу для проверки успешности установки командой yarn start.
7. Запустите программу.
8. **Проверка работоспособности ИС**

Для проверки работоспособности требуется запустить программу, проверить отображение таблиц и корректную работу кнопок приложения, а также генерацию отчета по продажам.

1. **Поддержка программы**

* **Обновление**. Регулярно обновляйте библиотеки и зависимости программы (**Python** и **PyQt5)** до актуальных версий. Это поможет избежать проблем совместимости и обеспечит защиту от уязвимостей.
* **Отслеживание ошибок**. Анализируйте сообщения об ошибках от пользователей и своевременно предоставляйте поддержку, исправляя выявленные проблемы. Это поможет сделать программу более стабильной и удобной для использования.
* **Уведомление пользователей**. При добавлении новых функций или изменений, уведомляйте пользователей, публикуя сообщения на GitHub.
* **Документация**: Обновляйте документацию программы, включая инструкции по устранению проблем, часто задаваемые вопросы (FAQ) и храните ее в репозитории.
* **Производительность и резервирование**. Проводите оптимизацию кода и базы данных по мере необходимости, и регулярно создавайте резервные копии данных, чтобы предотвратить потерю важной информации в случае технических сбоев или сбоев системы.

1. **Модернизация**

* **Сбор обратной связи**. Анализируйте отзывы пользователей для выявления возможностей улучшения программы. Это помогает выявлять новые требования и определять приоритеты для добавления новых функций или улучшения существующих.
* **Создание новой ветки**. Создайте ветку для разработки новых функций: git checkout -b feature-branch. Это позволит вам работать над новыми возможностями, не мешая основной версии программы, и поможет избежать конфликтов кода при слиянии.
* **Разработка функций**. Реализуйте новые функции, используя **Python** и **PyQt5** для логики программы, а также его интерфейса.
* **Тестирование**. Перед выпуском протестируйте новые функции на наличие ошибок. Используйте инструменты для тестирования Python-кода.
* **Обновление документации**. Обновите документацию, добавив описание новых функций и изменений, чтобы отразить актуальную версию программы.
* **Отправка изменений в репозиторий**: git push origin feature-branch.
* **Обслуживание**. Регулярно выполняйте обновления зависимостей с помощью команды: yarn upgrade.