Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ**

на тему:

«Информационная система для автошколы»

Выполнила: студентка учебной группы

ИСПк-402-52-00

Суслов Павел Андреевич

Руководитель УП.03

Долженкова Мария Львовна

Киров, 2024

**СОДЕРЖАНИЕ**

[1 Инсталляция программы 3](#_Toc182780447)

[1.1 Пользовательский вариант 3](#_Toc182780448)

[1.2 Вариант поддержания и модернизации 3](#_Toc182780449)

[2 Проверка работоспособности программы 5](#_Toc182780450)

[3 Поддержка программы 7](#_Toc182780451)

[4 Модернизация 9](#_Toc182780452)

# Инсталляция программы

## Пользовательский вариант

* Перейдите на репозиторий программы по ссылке: «<https://github.com/PapaSavage/drivingschool>».
* Скачайте все файлы, необходимые для работы программы. Репозиторий содержит основной код программы.
* Убедитесь, что на вашем компьютере установлены Python версии 3.8 или выше, а также система управления базами данных MySQL, которая используется для хранения данных учеников, инструкторов, расписания и результатов экзаменов.
* Выполните настройку подключения к базе данных, чтобы программа могла взаимодействовать с MySQL.
* Запустите основной файл программы, чтобы проверить корректность установки и работоспособность приложения. Убедитесь, что все отображается корректно и все элементы интерфейса работают как ожидается.

## Вариант поддержания и модернизации

* Убедитесь, что на вашем компьютере установлены все необходимые инструменты для разработки, такие как Python и библиотека PyQt для создания графического интерфейса, так как данное приложение использует графический интерфейс для взаимодействия с пользователем.
* Склонируйте репозиторий программы с GitHub с помощью команды:

« git clone https://github.com/PapaSavage/drivingschool ».

Это создаст локальную копию проекта на вашем компьютере, что позволит вам вносить изменения, тестировать и модернизировать код.

* Перейдите в папку проекта, созданную после клонирования репозитория: «cd drivingschool».
* Запустите основной файл программы с помощью следующей команды, чтобы проверить успешность установки: «pip install -r requirements.txt» и «python main.py».

Это позволит проверить работоспособность программы. Убедитесь, что программа может взаимодействовать с базой данных MySQL и корректно извлекает необходимые данные.

Для модернизации и внесения изменений в код:

* Откройте проект в текстовом редакторе или IDE VS Code.
* Внесите необходимые изменения в код.
* Тестируйте изменения, чтобы убедиться в их корректности, и обновите локальную базу данных, если структура данных была изменена.
* После успешного тестирования создайте коммит и отправьте изменения в репозиторий:

«git add .

git commit -m "Описание изменений"

git push origin main»

# Проверка работоспособности программы

Для проверки работоспособности программы ИС «Автошкола» необходимо выполнить несколько тестов и убедиться в корректной работе каждого из компонентов:

Запустите приложение и создайте тестовую учетную запись

* Проверьте возможность регистрации новых пользователей, включая учеников, инструкторов и администраторов.
* Убедитесь, что все поля формы регистрации заполняются корректно и система выводит соответствующие сообщения об ошибках при введении некорректных данных.
* Убедитесь, что авторизация зарегистрированных пользователей проходит без ошибок, и система предоставляет доступ к нужным функциям на основе роли пользователя.

Запись на занятия и управление расписанием

* Попробуйте записать тестового ученика на теоретические и практические занятия, убедившись, что запись выполняется корректно и отображается в расписании.
* Проверьте, что расписание обновляется после каждого внесенного изменения и отображается корректно для учеников и инструкторов.
* Убедитесь, что назначения инструкторов и автомобилей для практических занятий сохраняются корректно и отображаются в соответствующих расписаниях.

Отображение таблиц с информацией о пользователях и занятиях

* Проверьте корректность отображения таблиц, содержащих информацию о зарегистрированных учениках, инструкторах, а также записях на занятия.
* Убедитесь, что данные загружаются из базы данных корректно и обновляются в режиме реального времени при изменении информации (например, при добавлении нового ученика или обновлении информации о занятиях).

Проверка функциональности основных элементов управления

* Убедитесь, что все кнопки и элементы управления работают корректно, включая кнопки для регистрации, записи на занятия, создания расписания и управления экзаменами.
* Протестируйте основные функции программы, такие как добавление учеников, планирование занятий и проверка прогресса учеников, чтобы убедиться, что приложение реагирует на действия пользователя как ожидается.
* Проверьте, что завершение курсов (например, после сдачи экзаменов) отображается корректно и данные об итогах обучения сохраняются в базе данных.

Эти тесты помогут удостовериться в стабильной работе каждого компонента ИС «Автошкола» и корректной интеграции с базой данных.

# Поддержка программы

Чтобы ИС «Автошкола» оставалась надежной и актуальной, необходимо регулярно выполнять следующие действия:

Обновление библиотек и зависимостей

* Регулярно проверяйте и обновляйте используемые библиотеки и зависимости программы. Это включает:
* Обновление Python до последней стабильной версии, чтобы избежать устаревания функций и уязвимостей.
* Обновление библиотек для графического интерфейса (например, PyQt или Kivy) и других используемых пакетов.

Используйте команды:

* pip list --outdated
* pip install --upgrade <package\_name>

Мониторинг отзывов пользователей

Регулярно проверяйте репозиторий программы на GitHub на наличие:

* Новых сообщений об ошибках или запросов на улучшение от пользователей.
* Уведомлений о найденных уязвимостях или проблемах производительности.
* Уделяйте внимание отзывам, анализируйте их и оперативно устраняйте выявленные проблемы.
* Для повышения удобства использования можно внедрять предложенные пользователями улучшения.

Резервное копирование данных базы данных

* Настройте регулярное резервное копирование данных, чтобы избежать их потери при технических сбоях. Для MySQL можно использовать команды:

«mysqldump -u [username] -p[password] [database\_name] > backup.sql»

* Автоматизируйте процесс резервного копирования с помощью скриптов или встроенных средств, чтобы создавать копии базы данных ежедневно или еженедельно.
* Храните резервные копии в защищенных местах, например, в облачных хранилищах или на внешних носителях.

Эти действия помогут поддерживать стабильность, актуальность и безопасность программы.

# Модернизация

Для улучшения и адаптации ИС «Автошкола» к новым требованиям рекомендуется выполнять следующие действия:

Сбор обратной связи от пользователей

* Проводите опросы, анализируйте отзывы и собирайте предложения от учеников, инструкторов и администраторов.
* Выявляйте приоритетные области для улучшения, такие как добавление новых функций, оптимизация интерфейса или улучшение производительности.

Создание новой ветки для разработки

* Для работы над новыми функциями создавайте отдельные ветки, чтобы изолировать разработку от основной версии программы: «git checkout -b feature-branch».
* Это предотвратит конфликты кода и упростит слияние изменений после завершения разработки.

Разработка новых функций

* Реализуйте новые возможности, используя Python для основной логики и PyQt или Kivy для интерфейса.

Примеры функций для модернизации:

* Добавление автоматического уведомления учеников о предстоящих занятиях.
* Создание функции мониторинга прогресса обучения в реальном времени.
* Оптимизация работы базы данных для ускорения загрузки расписания и информации о пользователях.

Тестирование новых функций

Прежде чем интегрировать изменения в основную версию программы, проведите тщательное тестирование:

* Используйте инструменты для модульного тестирования, такие как unittest или pytest.
* Проверяйте корректность взаимодействия с базой данных MySQL, избегая сбоев при работе с реальными данными.
* Убедитесь, что новые функции не нарушают стабильность существующих.

Обновление документации

* После добавления новых функций внесите изменения в руководство пользователя, чтобы отразить актуальную информацию о возможностях программы.
* Создайте описание новых функций и их использования, добавьте примеры и обновите список требований.
* Зафиксируйте изменения в репозитории: «git push origin feature-branch».

Регулярное обслуживание программы

* Оптимизируйте код для улучшения производительности и читаемости.
* Пересматривайте структуру базы данных, чтобы адаптировать её к растущим объемам информации.
* Обновляйте зависимости и используемые библиотеки для предотвращения уязвимостей и повышения стабильности.

Эти шаги помогут программе оставаться актуальной, надежной и соответствовать потребностям пользователей.