Государственное автономное профессиональное образовательное учреждение

Пензенской области «Пензенский колледж информационных и промышленных

технологий (ИТ-колледж)

**Практикум по дисциплине**

**«Технология разработки программного обеспечения»**

**Лабораторная работа №2**

**«Разработка спецификации программы»**

Выполнила студентка группы 23ит35

Широкова Алина Сергеевна

Проверил преподаватель:

Сазонова А. Ю.

Пенза, 2024

Содержание

1. Введение
2. Цели
3. Соглашения о терминах
4. Предполагаемая аудитория и последовательность восприятия
5. Масштаб проекта
6. Ссылки на источники
7. Общее описание
8. Видение продукта
9. Функциональность продукта
10. Классы и характеристики пользователей
11. Среда функционирования продукта (операционная среда)
12. Рамки, ограничения, правила и стандарты
13. Документация для пользователей
14. Допущения и зависимости
15. Функциональность системы
16. Причинно-следственные связи, алгоритмы (движение процессов, workflows)
17. Функциональные требования
18. Требования к внешним интерфейсам

* Интерфейсы пользователя (UX)
* Программные интерфейсы
* Интерфейсы оборудования
* Интерфейсы связи и коммуникации

1. Нефункциональные требования

* Требования к производительности
* Требования к сохранности (данных)
* Требования к качеству программного обеспечения
* Требования к безопасности системы
* Требования на интеллектуальную собственность

1. Прочее

Приложение А: Глоссарий

Приложение Б: Модели процессов и предметной области и другие диаграммы

Приложение В: Список ключевых задач

**1. Введение**

Данный документ представляет собой спецификацию для разработки и внедрения программного обеспечения "ГАИ г. Пенза", предназначенного для автоматизации процессов работы Государственной Автомобильной Инспекции в Пензе.

**2. Цели**

Разработать приложение ГАИ г. Пенза, предназначенного для автоматизации процессов работы Государственной Автомобильной Инспекции в Пензе.

**3. Соглашения о терминах**

* «ГАИ»: Государственная Автомобильная Инспекция;
* «Пользователь»: Работник ГАИ, который будет взаимодействовать с системой;
* «Система»: Программное обеспечение ГАИ г. Пенза;

**4. Предполагаемая аудитория и последовательность восприятия**

Пользователи:

* сотрудники ГАИ;
* администраторы системы;

Последовательность восприятия: Пользователи будут проходить обучение и получать доступ к системе в зависимости от их уровня привилегий и необходимых функций.

**5. Масштаб проекта**

Проект включает в себя:

* разработку модулей приложения;
* создание базы данных для хранения информации;
* интерфейс для взаимодействия с данными;

**6. Ссылки на источники**

* <https://metanit.com/sharp/tutorial/1.1.php>
* <https://learn.microsoft.com/ru-ru/dotnet/csharp/>

**7. Общее описание**

Программа "ГАИ г. Пенза" представляет собой комплексное решение, позволяющее автоматизировать операции, связанные с оформлением ДТП, регистрацией транспортных средств и регистрацией автовладельцев.

**8. Видение продукта**

Видение продукта необходимо чтобы создать интуитивно понятную, надежную и безопасную систему, которая будет способствовать улучшению работы ГАИ. Программа должна стать основным инструментом для работы сотрудников ГАИ и повысить эффективность их деятельности.

**9. Функциональность продукта**

* обработка данных о транспортных средствах (создание новой записи о транспортном средстве, изменение существующей информации о транспортном средстве);
* оформление нарушений (введение новых нарушений, расчет штрафа, изменение данных о нарушении);
* обработка данных о владельце транспортного средства (создание новой записи о владельце транспортного средства, изменение существующей информации о владельце транспортного средства);

**10. Классы и характеристики пользователей**

* сотрудники ГАИ - эти пользователи будут основными операторами системы, выполняя такие функции, как внесение данных, проверка информации о транспортных средствах, обработка заявлений и обращений граждан;
* администраторы системы - пользователи, ответственные за управление, настройку и поддержку приложения, а также за обеспечение его безопасности;

**11. Среда функционирования продукта (операционная среда)**

Система должна функционировать на:

* на ОС не менее Windows 7;
* базе данных SQL Server;
* среда Visual Studio 2022;

**12. Рамки, ограничения, правила и стандарты**

* соблюдение законодательства о защите данных;
* стандарты кодирования для обеспечения безопасной разработки;
* использование современных технологий для обеспечения совместимости;

**13. Документация для пользователей**

Включает в себя:

* руководства пользователя;

**14. Допущения и зависимости**

* система будет интегрирована с существующими ресурсами и системами ГАИ;
* пользователи будут обучены работе с системой;

**15. Функциональность системы**

Функциональный блок 1: выбор необходимого окна для взаимодействия с данными.

Описание: Позволяет пользователю выбрать необходимое окно «Автомобиль», «Штраф», «Автовладелец».

Приоритет: Высокий.

Функциональный блок 2: работа с данными об автомобиле.

Описание: система должна предоставить доступ к информации из базы данных, пользователь может удалить, добавить и обновить информацию об автомобиле.

Приоритет: высокий.

Функциональный блок 3: работа с данными об автовладельце.

Описание: система должна предоставить доступ к информации из базы данных, пользователь может удалить, добавить и обновить информацию об автовладельце.

Приоритет: высокий.

Функциональный блок 4: работа с данными об штрафе.

Описание: система должна предоставить доступ к информации из базы данных, пользователь может удалить, добавить и обновить информацию об штрафе.

Приоритет: высокий.

**17. Причинно-следственные связи, алгоритмы (движение процессов, workflows)**

1. Выдача штрафов:
   * Причина: Нарушение зафиксировано и подтверждено.
   * Следствие: Автомобилисту направляется уведомление о штрафе, и информация передается в систему учёта штрафов.
2. Сбор статистических данных:
   * Причина: Необходимость анализа данных для улучшения работы парков и увеличения безопасности на дорогах.
   * Следствие: Формируются отчеты, основанные на собранной информации.
3. Добавление автомобиля:
   * Причина: Необходимость фиксация автомобилей.
   * Следствие: В базе данных появляется запись об автомобиле.
4. Добавление автовладельца:
   * Причина: Необходимость фиксация автовладельцев.
   * Следствие: В базе данных появляется запись об автовладельце.

Алгоритмы (Workflows):

1. Алгоритм регистрации нарушения:
   * Ввод данных о нарушении (время, место, детали).
   * Подтверждение информации (например, с использованием видеозаписей).
   * Сохранение данных в базу данных.
   * Выписка штрафа.

2. Алгоритм выдачи и управления штрафами:

* + Создание записи о штрафе в системе.
  + Выдача уведомления водителю.
  + Обработка платежей и напоминаний о несчастных случаях.

**18. Функциональные требования**

* управление информацией о транспортных средствах (создание новых записей о транспортных средствах, обновление существующих данных о транспортных средствах);
* регистрация правонарушений (добавление новых правонарушений, расчет штрафных санкций, изменение информации о зарегистрированных правонарушениях);
* работа с данными о владельцах транспортных средств (создание новых записей о владельцах, обновление имеющейся информации о владельцах транспортных средств);

**19. Требования к внешним интерфейсам:**

* интерфейсы пользователя (UX) - интуитивно понятный интерфейс для пользователей с высоким уровнем удобства;

Программные интерфейсы:

* JSON API для интеграции с сторонними системами;
* RESTful API для доступа к функциональности системы программным образом;

Интерфейсы оборудования:

* поддержка подключения принтеров и сканеров для распечатки документов и прочих необходимых материалов;

Интерфейсы связи и коммуникации:

* передача данных в базу данных;

**23. Нефункциональные требования**

Требования к производительности:

* время реакции системы не должно превышать 2 секунд для большинства действий, таких как поиск информации и открытие окон. Система должна демонстрировать стабильную работу даже при увеличении нагрузки и не проявлять сбоев при резком увеличении числа запросов;

Требования к сохранности (данных):

* регулярное резервное копирование данных;
* шифрование персональных данных;

Требования к качеству программного обеспечения:

* должна быть обеспечена высокая степень надежности и стабильности работы системы (Uptime более 99.5%);

Требования к безопасности системы:

* защита от современных интернет-угроз (SQL-инъекции, XSS);

Требования на интеллектуальную собственность:

* код программы должен быть защищен авторскими правами;
* все используемые библиотеки и компоненты должны иметь соответствующие лицензии;

**24. Прочее**

Обучение пользователей допустимо проводить как в очном формате, так и удаленно.

Приложение А: Глоссарий

- API: Интерфейс программирования приложений, позволяющий различным программам взаимодействовать друг с другом.

Приложение Б: Модели процессов и предметной области и другие диаграммы

Диаграммы процессов обработки заявок, управления данными, взаимодействия пользователей и системы.

Приложение В: Список ключевых задач

1. Анализ требований и проектирование системы.

2. Разработка ядра приложения и баз данных.

3. Тестирование функциональности и исправление ошибок.

4. Обучение пользователей и внедрение системы.