Федеральное Агентство по Образованию

Государственное образовательное учреждение высшего профессионального образования

Санкт-Петербургский Государственный Электротехнический Университет «ЛЭТИ» им. В.И.Ульянова (Ленина)

(СПБГЭТУ)

Кафедра МОЭВМ

**Спецификации к проекту**

**«Система контроля и отслеживания расписания поездов»**

Выполнили : Вахтель А.В.

Дёмин А.Д.

Ужегов А.Т.

Яковлев М. А.

Факультет: КТИ

Группа: 1304

Преподаватель: к.т.н.

доц. кафедры МО ЭВМ, Спицын А.В.

Санкт-Петербург

2016

## Постановка задачи

Разрабатываемая программа должна решать проблему отсутствия единой централизованной системы контроля и отслеживания расписания поездов, которая затрудняет работу операторов и приводит к снижению спроса на услуги РЖД со стороны пользователей (по причине невозможности просмотра расписания клиентами в любое удобное для них время). Данная программа должна на основе имеющихся данных показывать карту маршрутов поездов с информацией о станциях, поездах, расписаниях и маршрутах, позволять операторам редактировать расписание поездов. Система должна выдавать на запросы пользователя содержащие пункты отправки и пункты назначения, нужные ему маршруты и поезда.

## Название программы

В качестве названия программного продукта была выбрано аббревиатура СКОРП – Система контроля расписания поездов.

## Цели и назначение программы

Цель разработки программы состоит в повышении скорости и удобства получения информации о расписании поездов для клиентов и редактирования расписания для операторов РЖД

Назначение программы – значительно упростить процесс поиска информации о маршрутах и редактирования расписания для клиентов и операторов РЖД.

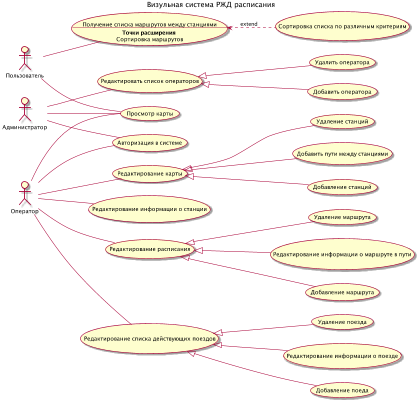
## Область применения

Программа может быть использована клиентами РЖД, которым нужно выбрать наиболее выгодный маршрут передвижения по железной дороге или операторами РЖД которым нужен инструментарий для редактирования расписаний и маршрутов

# . Описание пользователей

|  |  |  |
| --- | --- | --- |
| **Имя** | **Описание** | **Ответственность** |
| Клиент (пассажир) | Пользователь услуг, предоставляемых РЖД (транспортные перевозки и информация о движении поездов). | Использует СКОРП, для отслеживания расписания поездов |
| Оператор | Работник РЖД или программная система, осуществляющий (ая) контроль над транспортными перевозками по сети железных дорог, а также обеспечивающий (ая) связь между отдельными узлами этой сети | Редактирует расписание и карту маршрутов поездов |
| Системный администратор | Работник РЖД, ответственный за управление клиентскими базами данных, настройка и последующее сопровождение компьютерных сетей РЖД. | Создает и управляет учеными записями операторов |

1. Сценарии использования



Описание вариантов использования

*Сценарий 1.*

**Название**: «Подобрать оптимальный маршрут из точки А в точку Б».

**Роль:** Клиент РЖД

**Начальное состояние**: приложение проинициализировано.

**Основной поток событий**:

1. Пользователь просматривает карты и выбирает себе точку отправки и точку прибытия(Точки А и Б)

2. Пользователь выбирает пункт меню «составить маршрут»

3. Вводит в поле пункт отправки точку А

4. Вводит в поле пункт назначение точку Б

5. Вводит в поле время отбытия удобное уме время

6. Нажимает кнопку составить маршрут

7. Пользователь получает оптимальные маршруты

*Сценарий 2.*

**Название**: «Редактирование расписания».

**Роль:** Оператор РЖД

**Начальное состояние** приложение проинициализировано

**Основной поток событий**:

1. Оператор авторизуется.

2. Оператор нажимает кнопку редактирования расписания

3. Редактирует расписание ( изменяет время прибытия и отправки на какую либо станцию).

4. Нажимает кнопку сохранить изменения

*Сценарий 3.*

**Название**: «Редактирование карты».

**Роль:** Оператор РЖД

**Начальное состояние** приложение проинициализировано

**Основной поток событий**:

1. Оператор авторизуется.

2. Оператор нажимает кнопку редактирования карты

3. Редактирует карту ( добавляя или удаляя какие либо станции).

4. Нажимает кнопку сохранить изменения

*Сценарий 4.*

**Название**: «Редактирование списка действующих поездов».

**Роль:** Оператор РЖД

**Начальное состояние** приложение проинициализировано

**Основной поток событий**:

1. Оператор авторизуется.

2. Оператор нажимает кнопку редактирования списка поездов

3. Редактирует список поездов ( добавляя или удаляя какие либо поезда из списка).

4. Нажимает кнопку сохранить изменения

*Сценарий 5.*

**Название**: «Редактирование информации о станции».

**Роль:** Оператор РЖД

**Начальное состояние** приложение проинициализировано

**Основной поток событий**:

1. Оператор авторизуется.

2. Оператор нажимает кнопку редактирования информации о станции

3. Редактирует информацию о станции (изменение названия).

4. Нажимает кнопку сохранить изменения

*Сценарий 6.*

**Название**: «Редактирование списка операторов».

**Роль:** Администратор

**Начальное состояние** приложение проинициализировано

**Основной поток событий**:

1. Администратор авторизуется.

3. Администратор удаляет оператора или подтверждает его регистрацию

4. Нажимает кнопку сохранить изменения

1. Функциональные требования

Таблица 1. Функциональные требования

|  |  |
| --- | --- |
| № | Требование |
| 7.1 | Карта РЖД, отображающая пункты остановки поездов и пути между ними; |
| 7.2 | Возможность просмотра, редактирования (для оператора) расписания поездов для каждой станции; |
| 7.3 | Автоматическое обновление расписания при достижении поездом пункта прибытия; |
| 7.4 | Должна быть возможность формировать запросы к системе с выбором места отправления и пункта назначения. В качестве ответа на запрос система должна выдавать список рейсов удовлетворяющих заданным параметрам; |
| 7.5 | Возможность редактирования карты маршрутов поездов оператором; |
| 7.6 | Карта маршрутов поездов должна визуализировать движение поездов в реальном времени; |
| 7.7 | Наличие системы аутентификации; |

1. **Нефункциональные требования**

##### **Перспективы развития системы**

* Система должна иметь длительный жизненный цикл.
* Система должна быть построена с использованием стандартизованных и эффективно сопровождаемых решений.
* Система должна быть реализована как открытая система, и должна допускать наращивание функциональных возможностей.
* Система должна обеспечивать возможность модернизации как путем замены технического и общего программного обеспечения (ПО), так и путем совершенствования информационного обеспечения.
  1. **Показатели надёжности**

Время восстановления работоспособности системы при любых сбоях и отказах не должно превышать одного рабочего дня, исключая случаи неисправности серверного оборудования.

* 1. **Требования к надежности**
* Должно осуществляться разграничение прав доступа к системе.
* Система должна быть способной сохранять свой уровень качества функционирования в установленных условиях за определенный период времени,
* Система должна выдавать правильные результаты в ответ на запросы пользователя
* Для тестирования программы должны применяться unit-тесты, выполнение которых должно быть продемонстрировано заказчику.
* Система не должна допускать преждевременных сбоев

##### **Требования по эргономике и технической эстетике**

Интерфейс должен обеспечивать удобную навигацию в диалоге с любым пользователем, не зависимо от его уровня знаний в предметной области.

Наличие контекстно-зависимой помощи.

* 1. **Требования к производительности**

Скорость выполнения запросов должна cоставлять не больше 30 секунд на каждую 500 объектов в базе данных.

1. Требования к удобству использования

Таблица 2. Требования к удобству использования

|  |  |
| --- | --- |
| № | Требование |
| 8.1 | Число элементов на одном экране не должно превышать 4 |
| 8.2 | Время ожидания пользователя реакции системы на любое действие не должно превышать 3 секунд |
| 8.3 | В программе должна быть реализована система всплывающих подсказок при наведении на элемент |
| 8.4 | Для опытных пользователей должен быть быстрый способ работы с приложением (горячие клавиши) |
| 8.5 | Должен быть простой способ отмены действий |

1. Требования к надежности

Система должна выявлять ошибочные ситуации и сообщать о них пользователям.

Таблица 3. Ошибочные данные

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название | Причина возникновения | Сообщение об ошибке |
| 10.1 | Не задан пункт отправки | Пользователь оставил поле пункт отправки пустым | Введите пожалуйста пункт отправки |
| 10.2 | Не задан пункт прибытия | Пользователь оставил поле пункт прибытия пустым | Введите пожалуйста пункт назначения |
| 10.3 | Не задано время оправки | Пользователь оставил поле время отправки пустым | Введите пожалуйста время отправки |
| 10.4 | Не задано название станции | оператор не ввел название станции | Нельзя создать станцию без названия |
| 10.5 | Введен несуществующий пункт отправки | Пользователь ввел название станции, которая не зарегистрирована в системе | В поле «Пункт отправки» находится несуществующая станция |
| 10.6 | Введен несуществующий пункт прибытия | Пользователь ввел название станции, которая не зарегистрирована в системе | В поле «Пункт назначении» находится несуществующая станция |
| 10.7 | Введены неверные данные в поля время отправки | Пользователь ввел неверные данные в поле время отправки(символьная строка, пробелы и т.д.) | Поле «Время отправки» должно содержать данные в формате data |

1. Требования к сопровождению

Таблица 4. Требования к сопровождению

|  |  |
| --- | --- |
| № | Требование |
| 11.1 | Данные отображаемые системой должны обновляться каждый раз при внесении каких либо изменений в БД |
| 11.2 | В системе не должна быть возможно отключения обновлений |
| 11.3 | При выявлении ошибок, они должны исправляться разработчиками: критические в виде патчей к текущей версии, некритичные – должны быть исправлены в следующей версии продукта |

1. Требования к техническому обеспечению

ОС: «Microsoft Windows» версий 7, 8, 10; Mac OS X 10.11 El Captain

Свободное место на жестком диске: 30 мб.

У компьютера должна быть возможность подключения по сети.

1. Требования к информационной и программной совместимости

Таблица 5. Требования к информационной и программной совместимости

|  |  |
| --- | --- |
| № | Требование |
| 13.1 | Для работы программы будет нужен файл базы данных, в котором она содержит информацию о поездах, пользователях, станциях и маршрутах. Этот файл автоматически создаётся при установке программы либо может быть создан администратором. |
| 13.2 | Программа должна работать на следующих операционных системах Microsoft Windows» версий 7, 8, 10; Mac OS X 10.11 El Captain |
| 13.3 | Для работы программы необходима библиотека Qt5.0 или выше |

1. Требования к интерфейсу

|  |  |
| --- | --- |
| 13.1 | Окно программы должно состоять из трех областей: карты, списка станций и панели инструментов для редактирования карты. |
| 13.2 | Необходима чтобы была возможность отображения информации о станции на карте |
| 13.3 | Необходимо чтобы на карте показывалось расположение поездов в реальном времени. |
| 13.4 | В ответ на запрос пользователя должна выдаваться таблице с информацией о маршруте и поезде, который следует этому маршруту. |
| 13.5 | Необходим инструмент, который позволит добавлять станции |
| 13.6 | Необходим инструмент, который позволит добавлять переходы |
| 13.7 | Необходим инструмент, который позволит удалять станции |
| 13.8 | Необходим инструмент, который позволит удалять переходы |
| 13.9 | Форма для поиска маршрутов должна состоять из полей: Пункт Отправки, пункт назначения, время отправки |
| 13.10 | Форма авторизации, которая должна содержать поля для ввода логина и пароля |
| 13.11 | Форма регистрации, которая содержит поля для ввода логина и пароля |

## Входные/выходные данные

Входные данные для программы СКОРП представлены в таблице 7.

Таблица 7. Входные данные

| № | Входные данные | Формат данных | Вид поля на форме |
| --- | --- | --- | --- |
| 1 | Пункт отправки | Строка из выпадающего списка объектов, хранимых приложением |  |
| 2 | Пункт Прибытия | Строка из выпадающего списка объектов, хранимых приложением |  |
| 3 | Время отправки | Целое число в диапазоне [0;23] для часов и в диапазоне [0;59] для минут |  |
| 4 | Название станции | Текстовая строка до 20 символов |  |

Выходные данные: список найденных маршрутов.

Хранимые данные: база данных с пользователями, правами доступа, станциями, поездами, переходами.

1. Выбранные технологии реализации

В качестве модели жизненного цикла разрабатываемой программы была выбрана итерационная модель с перекрытиями, т.к. это позволяет наиболее эффективно выполнять этапы без задержек и исправлять найденные ошибки.

Достоинства данной модели, из-за которых она была выбрана:

1. Подходит для небольших команд разработчиков.
2. Быстрая реализация прототипа программы
3. Получение частых замечаний заказчика и ответов на вопросы при демонстрировании прототипа
4. Поддержка распараллеливания процессов
5. Первоочередность реализации решающих функций с высокой степенью риска
6. Повышение вероятности предсказуемого поведения системы с помощью уточнения поставленных целей

В качестве методологии разработки была выбрана RUP (требование заказчика).

Контроль версий.

Мы выбрали git потому что она является самой распространенной и широко используемой системой контроля версий. Благодаря тому что git является распределенной системой контроля версий можно легко найти обучающую документацию. Он позволяет избежать единой точки сбоя в лице единственного сервера хранящего все версии кода так как в git’е у каждого разработчика есть, полня версия кода со всей истории изменений в репозитории.

Также была рассмотрена система контроля версий SVN но ввиду того что она является централизованной (весь код со всеми изменениями хранится на одном сервере) и в следствие обладает свойством единой точки отказа в лице сервера который хранит все версии кода предпочтение было отдано git’у

Распределение задач

Распределение задач производится менеджером с помощью системы задач в Redmine.

Технология документирования кода

В качестве технологии документирования был выбран doxygen, т.к. он является де-факто стандартом при написании программных систем на языке программирования с/c++

Прототипирование интерфейса

Прототипирование интерфейса выполнялось в IDe qt creator ввиду того что в качестве библиотеке написания GUI был выбран Фреймворк qt

Язык программирования с++ (требование заказчика)

Технология тестирования

GTEST является де-факто стандартом для unit/интеграционных тестов программных систем реализованных на языке с++.

Среда разработки

Выбирается каждым участником индивидуально.

Коммуникация в команде

Коммуникация в команде средствами skype и с помощью комментарий в Redmine, по электронной почте и в социальных сетях.