

Федеральное агентство по образованию РФ
ГОУ ВПО Нижегородский государственный университет им. Н.И. Лобачевского
Факультет Вычислительной математики и кибернетики
Кафедра Математического обеспечения ЭВМ

УЧЕБНЫЙ КУРС
«Объектно-ориентированный анализ и проектирование»
для подготовки по направлению «Информационные технологии»
проект «Система бронирования такси»

Выполнили:

Студент группы 85М21 Лазарев Антон
Студент группы 85М21 Погорельский Илья
Студент группы 85М22 Васильев Евгений
Студент группы 85М22 Патрушев Андрей

Нижний Новгород
2015

Оглавление

- [1. Необходимость проекта](#)
 - [1.1. Обоснование необходимости](#)
 - [1.2. Видение проекта](#)
 - [1.3. Анализ выгод](#)
- [2. Концепция решения](#)
 - [2.1. Цели и Задачи](#)
 - [2.2. Предположения и Ограничения](#)
 - [2.3. Анализ использования](#)
 - [2.3.1. Пользователи](#)
 - [2.3.2. Сценарии использования](#)
 - [2.4. Требования](#)
 - [2.4.1. Требования пользователей](#)
 - [2.4.2. Системные требования](#)
- [3. Рамки](#)
 - [3.1. Функциональность решения](#)
 - [3.2. За рамками решения](#)
 - [3.3. Критерии одобрения решения](#)
- [4. Стратегии дизайна решения](#)
 - [4.1. Стратегия архитектурного дизайна](#)
 - [4.2. Стратегия технологического дизайна](#)

1. Необходимость проекта

1.1. Обоснование необходимости

Основной проблемой частного извоза является поиск клиентов, обычно таксисты стоят в опытном путем определенных местах и ждут заказов, или же полагаются на диспетчера принимающего и передающего заказы вручную. Процесс передачи заказов достаточно легко автоматизируется, в связи с этим нашей проектной группе было предложено реализовать систему бронирования такси.

1.2. Видение проекта

Подразумеваемая система бронирования должна обеспечить клиентов фирмы возможностью бронирования такси в своем городе. Успешным будет считаться обеспечение клиентов этой возможностью. Проект планируется закончить до конца мая месяца.

1.3. Анализ выгод

Данное приложение позволит снизить нагрузку на живых диспетчеров, которые определяют такси клиентам или же вовсе от них отказаться. Облегчит водителям поиск клиентов, также как клиентам водителей. Все это в целом повысит популярность и конкурентоспособность фирмы на рынке, что приведет к увеличению прибыли.

Команда разработчиков свою очередь получит опыт разработки, дополнительные листы в портфолио и возможную премию при выполнении проекта.

2. Концепция решения

2.1. Цели и Задачи

Основной целью проекта является создание приложения для фирмы, предоставляющей услуги такси.

Предполагаемое приложение будет состоять из двух частей. Первая часть предназначена для водителей, водители должны иметь возможность регистрироваться в системе и предавать ей данные о своем состоянии и местоположении, а также получать заказы на

перевозки. Вторая часть предназначена для клиентов, при помощи приложения пользователь должен иметь возможность заказа или бронирования такси в определенное место (с возможностью указания на координатной сетке) и время.

«YourNewTaxi». В рамках этой цели сформулированы следующие задачи:

- Создать и поддерживать базу водителей и, возможно, клиентов
- Создать интерфейс позволяющий водителям взаимодействовать с системой (передача статуса и местоположения, получение заказов)
- Создать интерфейс позволяющий пользователям бронировать такси (указание времени и места начала маршрута, места окончания маршрута, личных предпочтений)

2.2. Предположения и Ограничения

Предполагается что у фирмы заказчика есть карта местности в векторном формате содержащая все необходимые данные о местности в т.ч. географические координаты объектов на карте. Фирма в состоянии обеспечить своих водителей необходимыми устройствами для взаимодействия с сетью (например, портативный компьютер с доступом в сеть интернет или к другой коммуникационной линии способной обеспечить связь с ядром системы).

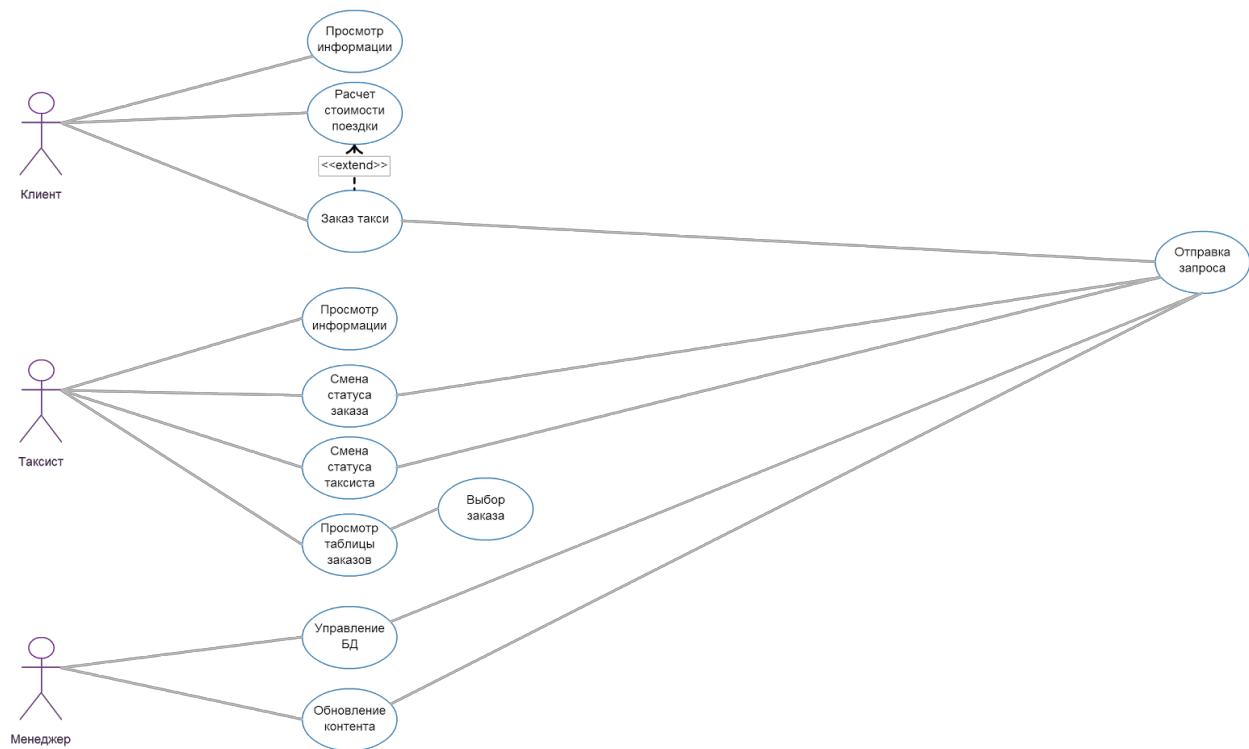
Фирма заказчик не выдвинула иных ограничений на работу системы за исключением описанного выше видения.

2.3. Анализ использования

2.3.1. Пользователи

- Клиент - ознакомление с информацией, отправка заявки на заказ такси.
- Таксист - личный кабинет, таблица заказов, уведомления.
- Менеджер - администрирование веб-сайта: база данных, разграничение прав доступа, наполнение.

2.3.2. Сценарии использования



2.4. Требования

2.4.1. Требования пользователей

I. Клиент

- 1) Просмотр контента сайта (новости, акции, информация о компании)
- 2) Расчет стоимости поездки
 - a) Место отправки, место назначения, автоподбор в полях ввода
 - b) Визуализация маршрута на карте
 - c) Отображение прайса вышеперечисленных услуг
 - d) Расчет итоговой стоимости
- 3) Оформление заказа
 - a) Расчет стоимости поездки
 - b) Выставление времени
 - c) Отправка заявки
 - d) Получение номера заказа

II. Таксист

- 1) Просмотр таблицы заказов
- 2) Применение фильтров для поиска заказа (тип, расстояние, цена)
- 3) Предоставление подробной информации по выбранному заказу
- 4) Взять заказ
- 5) Изменить статус заказа
- 6) Показать историю выполненных заказов
- 7) Изменение личного статуса

III. Менеджер

- 1) Управление БД исполнителей
 - a. Добавить/удалить таксиста.
 - b. Добавить/изменить/удалить информацию о таксисте.
 - c. Добавить/удалить заказ.
 - d. Добавить/изменить/удалить информацию о заказе.
- 2) Обновление контента
 - a. Изменение содержания страниц новости, акции и других информационных областей.

2.4.2. Системные требования

- Обеспечение постоянной работы сервера, высокая пропускная способность.
- Обеспечение работоспособности и поддержки базы данных.
- Разработка под ОС Windows NT/XP/7.

3. Рамки

3.1. Функциональность решения

Условно функциональные решения можно разделить на три группы:

а) Решения для клиентского приложения:

- Оформление заказа такси при помощи формы на веб-сайте, форма должна обеспечивать возможность заказа при помощи адреса заказчика или географических координат;
- Возможность редактирования уже созданного заказа при помощи формы на веб-сайте (всё, кроме адреса или координат);
- Проверка корректности вводимых данных (как минимум телефон).

б) Сервер получения и обработки заказов:

- Проверка положения на карте и статусов всех записей в базе таксистов;

- Обновление информации в базе таксистов на основании полученной информации;
- Запись принятого заказа в базу данных;
- На основании полученного адреса/координат поиск ближайшего свободного такси и назначение его на заказ. При этом система меняет статус назначенного такси и уведомляет клиента о назначенном такси;
- Уведомление таксиста о назначенном заказе;
- Уведомление пользователя о назначенном водителе;
- Обновление статуса заказа при получении информации о завершении его выполнения.

в) Решения для водительского приложения:

- Возможность регистрироваться в системе в роли водителя;
- Возможность смены статуса водителя на «Занят»;
- Возможность ставить отметку о выполненном заказе.

3.2. За рамками решения

Решено вынести за рамки проектируемой системы возможность редактировать множество состояний таксиста. Данное решение было принято в связи с признанием данной функциональности избыточной. Указанное множество статусов водителя (свободен, занят, выполняю заказ) обеспечивает покрытие всех сценариев использования нашей системы в начальной постановке задачи.

3.3. Критерии одобрения решения

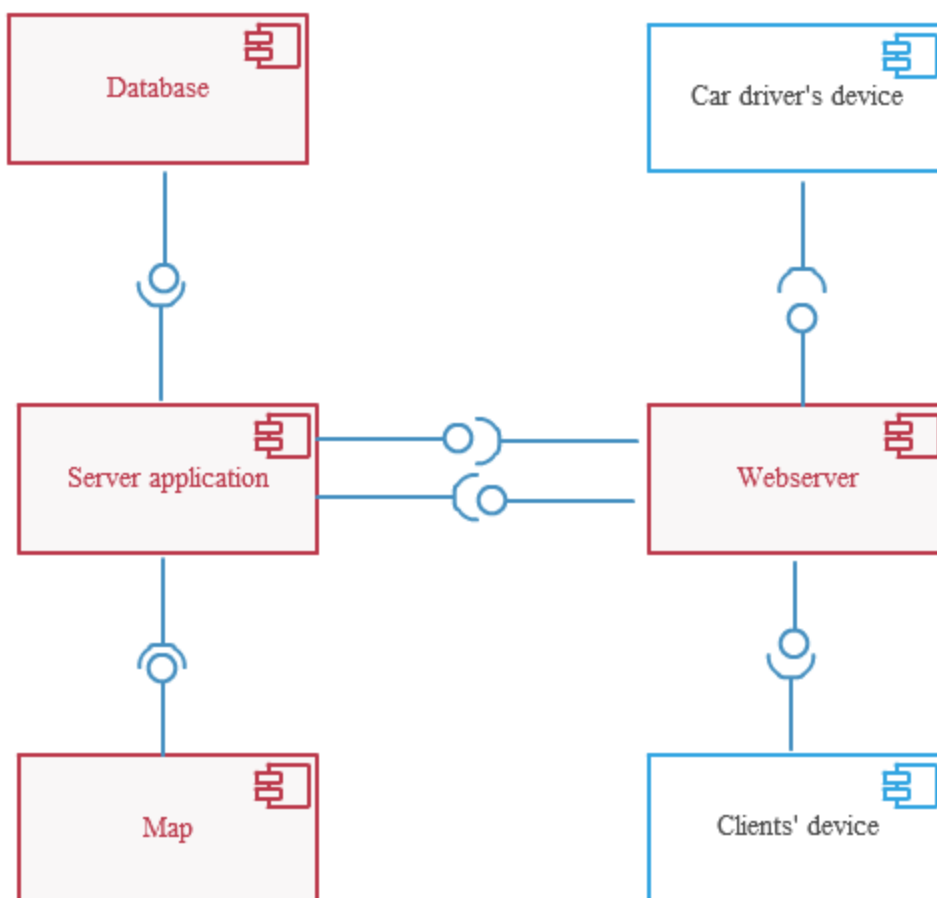
В перечень данных критериев входит:

- Положение и статусы водителей обновляются раз в 5 минут;
- Максимальное время от поступления заказа в систему до назначения её на водителя должно составлять не более минуты;
- Уведомления клиенты и водителю о состоянии заказа должны отправляться в течении 30 секунд после изменения заказа;
- Местоположение водителя и клиента определяются с точностью до 75 метров.

4. Стратегии дизайна решения

4.1. Стратегия архитектурного дизайна

Основная часть программного комплекса - серверное приложение, которое хранит список заказов и список водителей. Серверное приложение имеет интерфейс к карте. Серверное приложение соединено интерфейсами с модулем водителя и веб-браузерами клиентов.



4.2. Стратегия технологического дизайна

Для разработки выбран язык программирования C#, по следующим причинам:

- Уже известен разработчикам;
- Интуитивно понятный;
- Кроссплатформенный.

В качестве IDE планируется использование Visual Studio 2012. Данная IDE выбрана по многочисленным причинам:

- Уже установлена у разработчиков;
- Разработчики имеют большой опыт работы с данной версией;
- Удобный редактор кода.

В качестве репозитория исходного кода программ планируется использование удаленного репозитория <https://github.com/>. Его выбор объясняется несколькими причинами:

- Бесплатный репозиторий;
- Высокая популярность;
- Минимальное время настройки;
- Отсутствие ограничений на количество разработчиков.

В качестве системы управления версиями файлов планируется использование Git с графической оболочкой TortoiseGit. Выбор данной связки основан на следующих параметрах:

- Бесплатный;
- Имеется графическая оболочка;

В проекте будет использоваться технология ASP.NET для создания веб-служб. Основные преимущества ASP.NET:

- Богатый функционал;
- Поставляется вместе с Visual Studio.