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»

концепция проекта

Нижний Новгород  
2006

###### Содержание[[1]](#footnote-1)

[1. Необходимость проекта 3](#_Toc147596835)

[1.1. Обоснование необходимости 3](#_Toc147596836)

[1.2. Видение проекта 3](#_Toc147596837)

[1.3. Анализ выгод 4](#_Toc147596838)

[2. Концепция решения 4](#_Toc147596839)

[2.1. Цели и Задачи 4](#_Toc147596840)

[2.2. Предположения и Ограничения 4](#_Toc147596841)

[2.3. Анализ использования 4](#_Toc147596842)

[2.3.1. Пользователи 4](#_Toc147596843)

[2.3.2. Сценарии использования 5](#_Toc147596844)

[2.4. Требования 5](#_Toc147596845)

[2.4.1. Требования пользователей 5](#_Toc147596846)

[2.4.2. Системные требования 5](#_Toc147596847)

[3. Рамки 5](#_Toc147596848)

[3.1. Функциональность решения 6](#_Toc147596849)

[3.2. За рамками решения 6](#_Toc147596850)

[3.3. Критерии одобрения решения 6](#_Toc147596851)

[4. Стратегии дизайна решения 6](#_Toc147596852)

[4.1. Стратегия архитектурного дизайна 6](#_Toc147596853)

[4.2. Стратегия технологического дизайна 6](#_Toc147596854)

Итеративный подход к процессу разработки требует использования гибкого способа ведения документации. Живые документы (living documents) должны изменяться по мере эволюции проекта. Такой подход существенно отличается от принципов ведения документации в известной каскадной модели, где процесс разработки начинается лишь после того, как готовы и зафиксированы все требования и спецификации.

Документация проектов, также как и программный код, разрабатывается итеративно. На фазе выработки концепции планы имеют форму описания высокоуровневых подходов (approaches) и по мере подготовки распространяются среди членов проектной группы и других заинтересованных лиц для получения отзывов. К примеру, подход к тестированию может быть кратко сформулирован во время фазы выработки концепции, а его превращение в план тестирования происходит на более поздних фазах. После перехода к фазе планирования документы постепенно дорабатываются, возникающие детальные планы снова поступают на проверку всем заинтересованным сторонам, и описанный процесс повторяется итеративно. Типы планов и общее количество описывающих их документов могут варьироваться от проекта к проекту.

# Необходимость проекта

## Обоснование необходимости

Заказчиком является новая компания в сфере перевозок «YourNewTaxi». Фирма занимается поиском такси и назначением такси клиентам. Для удобства водителей такси и клиентов фирма решила разработать собственное приложение бронирования.

## Видение проекта

Старания объектной группы будут направлены на разработку программного обеспечения, способствующего оптимальной взаимосвязи между «Клиентом» и «Водителем». В рамках проекта будет разработана система, которая обеспечит клиента возможностью заказа такси с дальнейшим предоставлением информации о его выполнении, а водителям позволит оперативнее реагировать на запросы клиентов по средствам обращения к серверу обработки запросов клиентов. Временные рамки сильно сужены. На написание проекта выделено два с половиной месяца.

Видение (vision) – это ничем не ограничиваемое представление о том, каким должно быть решение (solution). Видение проекта направлено на формирование у всех вовлеченных в проект сторон единого понимания его концепции. Формулировка видения (vision statement) должна быть достаточно краткой для запоминания, достаточно ясной для понимания и достаточно сильной для мотивирования. Хорошая формулировка видения ориентируется на пять SMART характеристик:

* Specific (определенность/конкретность) – видение четко указывает на то (идеальное) состояние, достижение которого является целью проекта.
* Measurable (измеримость) – дает проектной группе четкий критерий успешности проекта и достижения поставленных целей.
* Achievable (достижимость) – цели, сформулированные в видении, должны быть достижимы в рамках имеющихся ресурсов, времени и возможностей команды. Достижимость мотивирует команду на выполнение проекта.
* Relevant (обоснованность) – цели, сформулированные в видении, должны иметь существенное значение для заинтересованных сторон и напрямую быть связанными с их проблемами и/или потребностями.
* Time-based (ограниченность во времени) – видение должно четко указывать на ожидаемые временные рамки, в которые решение будет достигнуто.

Сформулируйте (максимально кратко, в одной двух фразах) видение проекта.

## Анализ выгод

Основываясь на сформулированном выше видении проекта, перечислите, какие выгоды получат заинтересованные стороны по завершении проекта (в результате внедрения и использования решения).

В результате внедрения программного обеспечения стороны получат следующие выгоды:

* Водители смогут оперативно реагировать на запросы клиентов;
* Клиенты получат возможность заказа такси в любую часть города;
* Заказчик в лице компании «YourNewTaxi» получит программный продукт, обеспечивающий связь между клиентом и водителем напрямую. Благодаря чему компания сэкономит на содержании call-центров, а также получит доступ ко многим статистическим данным для оптимизации расположения водителей;
* Разработчики программного комплекса получат не только денежное вознаграждение в соответствии со сметой, но и ценный опыт в разработке, анализе и проектировании приложений.

# Концепция решения

Концепция решения (solution concept) предоставляет общее описание подходов, которые проектная группа предполагает использовать для разрешения проблем и/или удовлетворения потребностей заинтересованных сторон.

Формирование концепции решения начинается с выяснения у заинтересованных сторон, описания и фиксации проектной группой целей проекта. Далее каждая цель разбивается на измеримые компоненты – задачи.

Во взаимодействии с заинтересованными сторонами проекта сформулируйте и утвердите цели решения, на достижение которых направлен проект. Определите задачи, из которых будет складываться достижение каждой цели.

Для организации работы водителей необходимо, что бы каждый водитель смог зарегистрироваться в системе. После регистрации приложение водителя должно периодически сообщать общей системе положение и состояние водителя (занят, свободен, выполняет заказ, отсутствует в сети).

Для удобства клиента будет разработано приложение для заказа такси. Заказ может осуществляться как посредством задания адреса, так и задания географических координат.

Для организации связи «Водитель - Клиент», а также для сбора статистических данных, разработана серверная часть, благодаря которой запрос клиента будет удовлетворён водителем, первым откликнувшимся на заказ.

## Предположения и Ограничения

В процессе формирования концепции решения проектная группа постоянно взаимодействует с заинтересованными сторонами, собирая необходимую информацию о требованиях к функциональности будущего решения. Тем не менее, неизбежная неполнота информации приводит к тому, что относительно некоторых функциональных возможностей решения могут потребоваться предположения (assumptions). Помимо функциональных требований заинтересованные стороны могут выдвигать качественные требования, задающие ограничения на создаваемое решение. Также ограничения могут порождаться средой, в которой должно будет функционировать решение после внедрения.

Определите, имеются ли в проекте требования, нуждающиеся в предположениях, если да, сформулируйте их. Определите, имеются ли ограничения на будущее решение. Если да, сформулируйте их.

При анализе сферы деятельности компании выдвинуты необходимые требование, напрямую не затронутые в техническом задании:

* Информирование клиента о примерном времени ожидания такси;
* Наличие у клиента дополнительных требований к транспортному средству: наличие подъёмника, багажника, количество мест в транспортном средстве, и т.д.;
* Наличие обратной связи;
* Организация системы наказаний для недобросовестных водителей такси;
* Наличие системы поощрения водителей на основе отзывов клиентов;
* Возможность заказа такси к определённому времени;

При анализе сферы деятельности компании выявлены следующие ограничения на решение:

* Территориальные ограничения: компания работает только в одном городе;
* Технологические ограничение: водителю необходимо иметь в автомобиле компактное устройство с доступом к интернету, в частности к серверу компании «YourNewTaxi»;
* Ограничение на число одновременно выполняющихся заказов: водитель не должен брать более одного заказа;
* Графический интерфейс программы должен быть понятен и ориентирован на конечного пользователя. При этом каждый пользователь не может иметь доступ к серверной составляющей напрямую;

## Анализ использования

Основой формулировки требований является анализ использования, включающий определение пользователей (users) и описание того, как пользователи будут взаимодействовать с решением.

### Пользователи

В разработке решения заинтересованы множество сторон, однако непосредственная работа с ним будет выполняться пользователями, поэтому прежде чем приступать к дизайну решения, необходимо определить, кто будет с ним взаимодействовать. В процессе анализа должны быть выделены группы пользователей (например, на основе областей их деятельности, в которых будет использоваться разрабатываемое решение).

Сформируйте список групп пользователей, для которых предназначено решение.

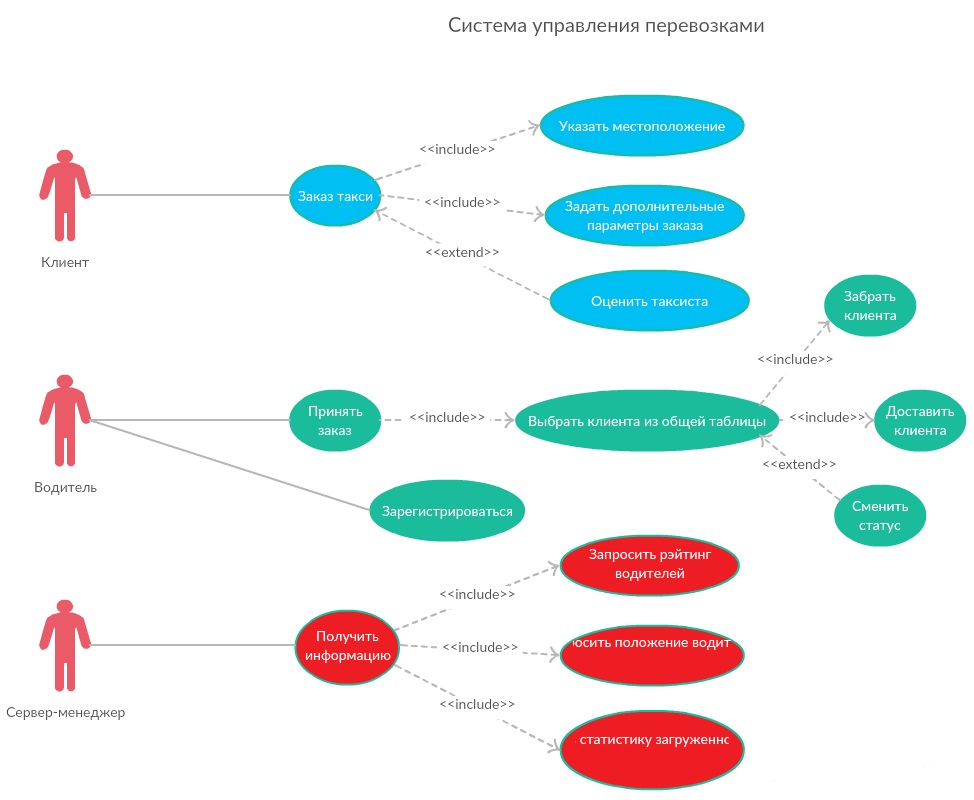
Можно выделить три группы пользователей, для которых предназначено данное решение:

* Клиенты – люди, которые заказывают такси;
* Водители, или таксисты – сотрудники компании, выполняющие заказ клиента;
* Серверные менеджеры – персонал компании, деятельность которого направлена на оптимизацию, мониторинг, управление перевозками.

### Сценарии использования

Сценарии использования (usage scenarios) определяют последовательности действий, которые пользователи выполняют при взаимодействии с решением. Один из возможных (и достаточно распространенных) вариантов – язык UML.

Для каждой выделенной на предыдущем шаге группы пользователей определите характерные способы их взаимодействия с решением и, используя необходимые диаграммы UML, опишите сценарии использования.



## Требования

Требования (requirements) определяют, что должно делать разрабатываемое решение. Требования могут выражаться в терминах функциональности или в виде правил и параметров, определяющих функциональность.

### Требования пользователей

Разобьём требования пользователей так же на несколько групп:

1. Требования клиентов: возможность заказа такси из приложения, при этом задание местоположения осуществляется вводом адреса в текстовое поле или считыванием непосредственно географических координат при нажатии на область карты;
2. Требования водителей: возможность зарегистрироваться в приложении, обеспечить доступ к базе запросов клиентов, возможность взять заказ из базы простым кликом по запросу клиента, возможность изменять свой текущий статус;
3. Требования руководства: доступ к статистическим данным: частота запросов клиентов в каждом районе, текущий статус и местоположение каждого водителя, текущий рейтинг водителей, доступ к базе данных сотрудников. Для этого необходимо разработать графический интерфейс для работы не только с базой но и с картой;
4. Помимо этого важно интерактивный графический пользовательский интерфейс. Каждый пользователь получит доступ к карте города для удобства заказа такси\выполнения заказа\мониторинга. Однако функционал разных пользователей будет различным, что связано с правами доступа к базе данных.

### Системные требования

Будущее решение должно быть адаптировано как под стационарные компьютеры, так и под мобильные устройство. Основные преимущества, связанные с кроссплатформенностью приложения:

* Доступность в любое время и в любом месте;
* Ориентированность на разнообразие рынка устройств.

Поэтому для каждой платформы стоит сформулировать свои системные требования:

* Для приложений, запускаемых на ПК необходимо наличие установленного .Net Framework версии 4.5 и выше, 256 мб оперативной памяти, процессор с тактовой частотой 1,8 и выше.
* Приложения, запускаемые на мобильных телефонах, будет работать сразу на трёх платформах: iOS, Android, Windows Phone. Поэтому для стабильной работы необходимы системы: iOS 7 и старше, Android 4.4 и выше, Win.Phone 8.1 и выше.
* Для организации работы приложения как web-сервиса, необходимо наличие предустановленного браузера с поддержкой HTML.
* Для сервера минимальные системные требования следующие: 1,3 ГГц (64-разрядный процессор) или быстрее для многоядерных процессоров, 4 ГБ при развертывании Windows Server Essentials в качестве виртуальной машины, Жесткий диск объемом 160 ГБ, системный раздел 60 ГБ, Адаптер Gigabit Ethernet (10/100/1000baseT PHY/MAC), Маршрутизатор или брандмауэр, поддерживающие преобразования сетевых адресов для IPv4 или IPv6, стабильный доступ к сети Интернет.

# Рамки

Рамки (scope) определяют пространство параметров, в котором будет создаваться решение, детализируя функциональность, определяя, что останется за рамками решения и указывая критерии, по которым заинтересованные лица будут судить о готовности решения. Рамки создаются на основе единого видения, являются результатом компромисса между сформулированными целями и условиями реальности и отражают приоритезацию заказчиком имеющихся требований к создаваемому решению. Частью процесса определения рамок проекта является вынесение менее важной функциональности из текущего проекта в планы на будущее.

Рамки решения *(*solution scope*)* определяют функциональность решения и его возможности (включая те, что не относятся к программному обеспечению). Возможность (функциональность, составляющая, feature) – это требуемый или желаемый аспект программного или аппаратного обеспечения. Например, предварительный просмотр перед печатью может быть возможностью текстового процессора; шифрование почтовых сообщений – возможностью почтовой программы. Сопроводительные руководства пользователей, интерактивные файлы помощи, операционные руководства и обучение также могут быть составляющими решения.

Рамки проекта *(*project scope*)* определяют объем работ, который должен быть выполнен проектной группой для поставки заказчику каждого из элементов, определенного рамками решения.

Управление рамками проекта критично для его успеха. Необходимо определять и фиксировать рамки решения и проекта, используя треугольник компромиссов и матрицу компромиссов проекта.

## Функциональность решения

Укажите здесь функциональность в терминах возможностей (features) и функций (functions), которая будет реализована в разрабатываемом решении.

В разрабатываемом решении будет реализована ограниченная функциональность, а именно:

* Возможность заказа такси клиентом, путем задания адреса или географических координат;
* Клиент будет иметь возможность задавать определённые параметры для транспортного средства водителя;
* Возможность водителя принимать или отклонять заказ, сообщать системе текущее местоположение, а также свой статус, регистрироваться в общей базе водителей компании;
* Обеспечение связи между клиентом и водителем на основе серверной поддержки;

Также, будет подготовлена почва для наращивания функционала приложения, а именно:

* Организация системы поощрения водителей на основе его рейтинга среди клиентов;
* Оптимальная система анализа оптимальности расположения таксистов на текущей локации;
* Механизм добавления новых карт местности в приложение, и, как следствие, возможность междугородних перевозок с заранее оговорёнными ограничениями;
* Возможность онлайн оплаты заказа клиентом;

## За рамками решения

Укажите здесь функциональность, которая имеется или предполагается в требованиях заинтересованных сторон, но не будет реализована в решении, и опишите причины вынесения данных возможностей и функций за рамки решения (используйте треугольник компромиссов).

Существует ряд функций, которые не будут реализованы в решении. К таким можно отнести реализацию следующих возможностей:

* Реализация чёрного списка для клиентов;
* Возможность клиента прервать заказ до посадки, во время посадки и после;
* Возможность оперативной связи водителя с серверным центром;
* Возможность удаления водителя из базы компании;
* Перечисление заработных плат водителям с использованием серверных возможностей.

Для обоснования вынесения данных функций за рамки решения воспользуемся треугольником компромиссов.



Учитывая то, что треугольник сбалансирован и при изменении, или усложнении требований к проекту балансировка будет нарушена. В случае реализации «чёрного списка» для клиентов нам понадобятся дополнительные технические ресурсы (дополнительная база данных), что в свою очередь удорожает проекта, и время на создания решения будет увеличено. Возможность клиента прервать заказ влечёт увеличение функциональности и увеличение времени и, как следствие, дивидендов за расширение возможностей решения. Увеличение возможностей водителя также влечёт разрастание функционала, как впрочем и перечисление им затрат. Однако помимо увеличения возможностей такой шаг влечёт увеличение затрат, ибо необходим оформлять постоянный банковский счёт для перечисления зарплат.

## Критерии одобрения решения

Список критериев, указывающий на готовность решения:

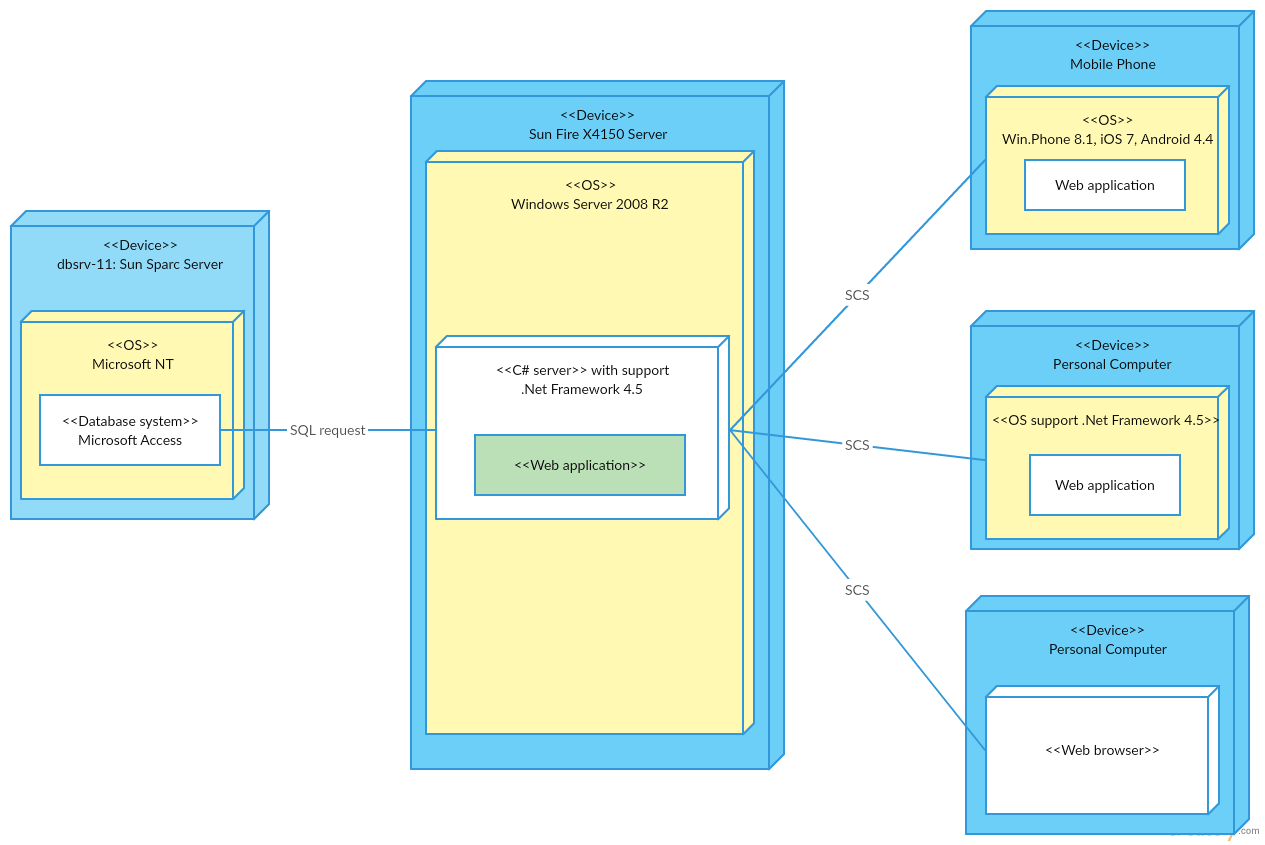
* Все требования сторон удовлетворены в рамках текущих требований функциональности;
* Удачно пройдены тесты по внедрению решения в некоторые опытные образцы устройств, на которых в дальнейшем решение и будет развёрнуто для использования;

# Стратегии дизайна решения

## Стратегия архитектурного дизайна

На основе разработанного списка возможностей и функций формируется стратегия архитектурного дизайна (architectural design strategy), описывающая решение в целом. Она определяет компоненты решения и их взаимодействие. Отличный способ описания решения на этом этапе – использование иллюстрирующих диаграмм (например, UML).

Сформируйте и опишите общий архитектурный проект решения.



## Стратегия технологического дизайна

Разработка решения потребует использования определенных продуктов и технологий. Стратегия технологического дизайна (technical design strategy) описывает, какие технологии и продукты выбраны проектной группой в качестве средства реализации решения.

Аргументировано опишите, какие технологические средства будут использованы в процессе работы над решением.

В рамках стратегии технологического дизайна в качестве средств реализации, проектной группой выбраны средства разработки платформы Java. Связано это с кроссплатформенностью разработанного решения, так как внедрение продукта будет происходить не только на персональные компьютеры, но и на мобильные устройства конечных пользователей. Программная платформа Java предоставляет возможность разработки и внедрения решения и на ПК и на мобильные устройства, важной составляющей которых является наличие виртуальной машины Java.

1. [↑](#footnote-ref-1)