Федеральное агентство по образованию РФ

ГОУ ВПО Нижегородский государственный университет им. Н.И. Лобачевского

Институт информационных технологий, математики и механики

УЧЕБНЫЙ КУРС

**«Объектно-ориентированный анализ и проектирование»**

для подготовки по направлению «Инженерия программного обеспечения»

СТРУКТУРА ПРОЕКТА

“АВТОСЕРВИС”

Выполнили:

студенты группы 381706-1м

Гладков Дмитрий

Пипикин Олег

Флегонтов Александр

Шептунов ВладиславРецензент

Нижний Новгород  
2018

###### Содержание

[1. Рамки проекта 3](#_Toc516212405)

[1.1. Матрица компромиссов проекта 3](#_Toc516212406)

[1.2. Вехи проекта 3](#_Toc516212407)

[1.3. Сметы проекта 5](#_Toc516212408)

[1.4. План-график проекта 5](#_Toc516212409)

[2. Роли и ответственности 5](#_Toc516212410)

[2.1. Знания, умения и навыки 5](#_Toc516212411)

[2.2. Структура команды 6](#_Toc516212412)

[3. Протоколы проекта 6](#_Toc516212413)

[3.1. Управление конфигурацией 6](#_Toc516212414)

[3.2. Управление изменениями 7](#_Toc516212415)

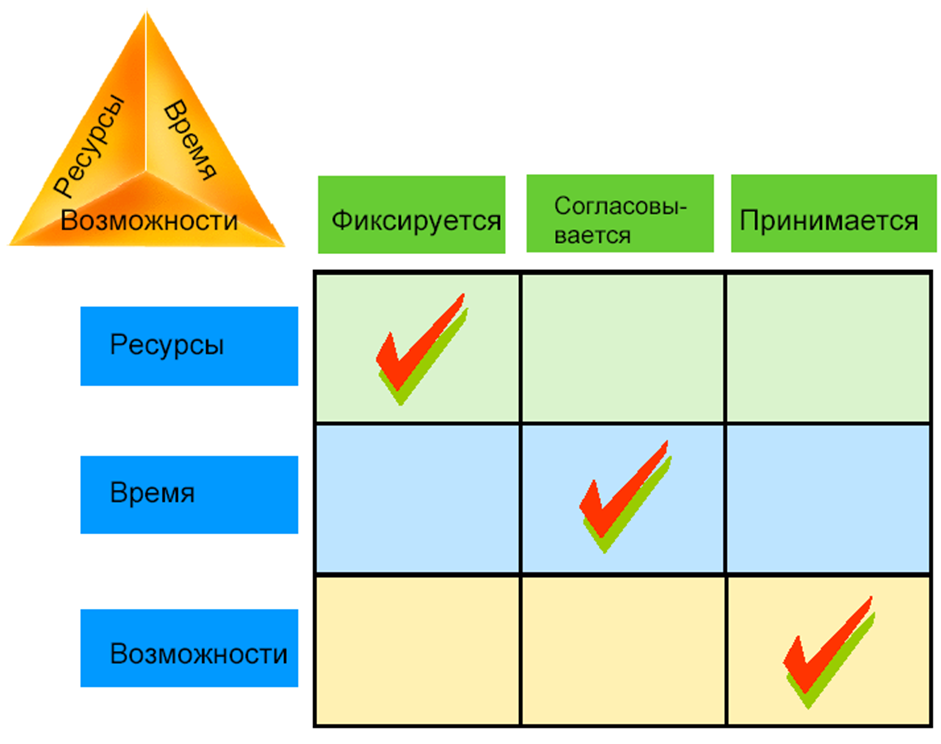
[3.3. Управление внедрениями 7](#_Toc516212416)

[3.4. Достижение качества проекта 7](#_Toc516212417)

[3.5. Рабочая среда проекта 7](#_Toc516212418)

# Рамки проекта

## Матрица компромиссов проекта



1. Параметр «Время» согласуется. Срок выполнения проекта – с 01.04.18 по 31.05.18. и может быть изменен при согласовании с заказчиком.
2. Параметр «Возможности» принимаются. Будет реализован функционал, описанный в документе «Концепция проекта» в разделе 3.1
3. Параметр «Ресурсы» фиксируется. Имеется команда, состоящая из 4 человек, которая работает над реализацией проекта.

## Вехи проекта

1. Написание и согласование с заказчиком следующей документации:
   1. “Концепции проекта”,
   2. “Структуры проекта”,
   3. “Функциональной спецификации”.
2. Утверждение планов проекта:
   1. Формирование и согласование план-графика;
   2. Формирование и согласование бюджета;
3. Разработка проекта:
   1. Клиентское приложение
   2. Менеджерское приложение
   3. Серверная часть
   4. Тестирование
4. Сдача проекта
   1. Конечная сдача проекта заказчику

## Сметы проекта

|  |  |  |  |
| --- | --- | --- | --- |
| Ресурс | Требуемое количество | Тариф | Общая стоимость |
| Лицензионная IDE для разработки кода | 4 | 5000 руб. за штуку | 20000 руб. |
| Оплата труда разработчиков | 4 человека | Оценка “Отлично” | 4 оценки “Отлично” |
| Расход на питание | 4 человека  х2 месяца | 15000 руб. в месяц на человека | 120000 руб. |
| Расход на канцелярию | - | 250 руб. в месяц на человека | 2000 руб. |
| Расход на бензин для встречи с заказчиком | 50 л. | 40 руб. за 1л. | 2000 руб. |
| Сервер | 1 | 100000 руб. за штуку | 100000 руб. |
| **Итог:** | 244000 руб. + 4 оценки “Отлично” | | | |

## План-график проекта

[План-график проекта.](https://goo.gl/vAVNCZ)

# Роли и ответственности

## Знания, умения и навыки

Для успешного выполнения данного проекта необходимо обладать следующими навыками и умениями:

* Уверенное владение языком программирования Java.
* Понимание работы протоколов TCP/IP, Socket.
* Знание паттернов проектирования.
* Опыт проектирования клиент-серверных приложений.
* Владение английским на уровне чтения технической документации.
* Знание основ базы данных, SQL.
* Опыт работы с распределенными системами контроля версий (Git).
* Опыт написания модульных тестов (JUnit).

## Структура команды

|  |  |
| --- | --- |
| **Участник команды** | **Роль** |
| **Флегонтов Александр** | Менеджер проекта |
| **Гладков Дмитрий** | Архитектор |
| **Шептунов Владислав** | Старший программист |
| **Пипикин Олег** | Старший тестировщик |

Обязанности менеджера проекта:

* Согласование требований к проекту с заказчиком.
* Организация работ по проекту в соответствии с планом.
* Планирование и проведение совещаний.

Обязанности архитектора системы:

* Разработка ключевых сценариев взаимодействия компонентов.
* Создание модели данных.
* Определение интерфейсов.
* Определение технических средств и паттернов для реализации компонентов.

Обязанности старшего программиста:

* Распределение работ по программной реализации проекта между членами команды.
* Разработка серверной части
* Разработка базы данных и интерфейса взаимодействия с ней.
* Разработка программной логики компонент проекта.

Обязанности тестировщика:

* Разработка сценариев тестирования (как для модульного тестирования, так и user-case).
* Разработка как позитивных, так и негативных тест кейсов.

Контроль за прохождением модульных тестов.

# Протоколы проекта

## Управление конфигурацией

Стратегия управления конфигурацией включает в себя:

1. Исходный код

- Git - выбранная система контроля версий на основе платформы GitHub.

- Для внесения изменений используются pull request'ы.

- Окончательное решение о принятии изменений принимает менеджер проекта.

1. Документация

- Разработка документации и обмен между членами команды осуществляется с помощью сервиса Google Docs.

1. Команда

- Каждый участник должен следовать Git Workflow.

- При командной разработке коммуникация осуществляется с помощью Skype.

## Управление изменениями

Изменения, связанные с программной и документной частью проекта, могут вноситься всеми участниками проекта, без ограничений, кроме заказчика.

Если заказчик собирается внести какие-либо изменения, это должно быть согласовано с командой на общем митинге в устной форме, причем от команды должно присутствовать не менее двух человек.

## Управление внедрениями

Процесс внедрения:

1. В фирме устанавливается купленный сервер. Срок установки – до начала стадии «Полное тестирование системы» (п. 1.4)
2. На сервер устанавливается серверная часть системы. На тестовый компьютер устанавливаются клиентские части. Происходит тестирование системы
3. В начале стадии «Завершение проекта» на компьютеры сотрудников и начальника фирмы устанавливаются их клиентские части. Заказчику передается также приложение для работы с клиентами фирмы
4. Проводятся необходимые тренинги для сотрудников фирмы по работе с разработанной системой

## Достижение качества проекта

Стратегия по достижению требуемого качества проекта включает в себя:

1. Следование принципам SOLID и Clean Code
2. Стадия Code review перед принятием изменений в коде проекта
3. Регулярное взаимодействие с заказчиком, позволяющее корректировать работу команды
4. Использование определенных архитектурных стилей (клиент-сервер) и паттернов (MVC)
5. Следование методологии управления проектами Scrum
6. Использование системы контроля версий Git, с которой команда проекта уже работала в прошлом

## Рабочая среда проекта

Разработка проекта должна происходить с использованием собственного оборудования членов команды для снижения стоимости проекта. Предполагается наличие оборудования, отвечающего следующим требованиям:

* аппаратные характеристики:
  + ОЗУ 4 Гб;
  + Процессор не ниже архитекруты Intel haswell и линейки Intel i5.
* ОС - Windows 10.
* Software:
  + Netbeans;
  + Git client for Windows;
  + JDK 8.xx;
  + Microsoft Office 2013.
  + Текстовый редактор (Notepad++ или Sublime Text)
* Доступ в интернет.

Так как разработка проекта ведется из дома, то каждый из членов команды самостоятельно отвечает за обеспечение себя необходимым офисным оборудованием в виде стула, стола итд.

Все члены команды должны быть обеспечены канцелярскими товарами согласно смете проекта.