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»

**СТРУКТУРА ПРОЕКТА**

Нижний Новгород  
2006

# Содержание

[1. Рамки проекта](#h.w21o9kpow859)

[1.1. Матрица компромиссов проекта](#h.lkqt9h208poz)

[1.2. Вехи проекта](#h.dccu3lh79n41)

[1.3. Сметы проекта](#h.mecplrg37pqx)

[1.4. План-график проекта](#h.9eajxgao7m70)

[2. Роли и ответственности](#h.ljxq07nte8w5)

[2.1. Знания, умения и навыки](#h.9saygmm6085b)

[2.2. Структура команды](#h.bjem7493iapk)

[3. Протоколы проекта](#h.2nno2wl10oi0)

[3.1. Управление конфигурацией](#h.8jdo5axcopx3)

[3.2. Управление изменениями](#h.yybhhvkl8as)

[3.3. Управление внедрениями](#h.7t7xmi5dnu7t)

[3.4. Достижение качества проекта](#h.73wns0eqd59i)

[3.5. Рабочая среда проекта](#h.vf3uz9dijiru)

###### 

Документ “Структура проекта” включает в себя информацию об организации проектной группы, персонификации ролей и ответственности. Также документ разъясняет схемы взаимодействия проектной группы с заказчиком и заказчика – с проектной группой.

# 1. Рамки проекта

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 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 определить и фиксировать рамки решения и проекта, используя треугольник компромиссов и матрицу компромиссов проекта.

## 1.1. Матрица компромиссов проекта

Хорошо известна взаимозависимость между ресурсами проекта (людскими и финансовыми), его календарным графиком (временем) и реализуемыми возможностями (рамками). Эти три переменные образуют треугольник компромисов (tradeoff triangle), показанный на рис. 1.

После достижения равновесия в этом треугольнике изменение на любой из его сторон для поддержания баланса требует модификаций на другой (двух других) сторонах и/или на изначально измененной стороне.



Рисунок 1. Треугольник компромиссов\*

Нахождение верного баланса между ресурсами, временем разработки и возможностями – ключевой момент в построении решения, должным образом отвечающего нуждам заказчика.

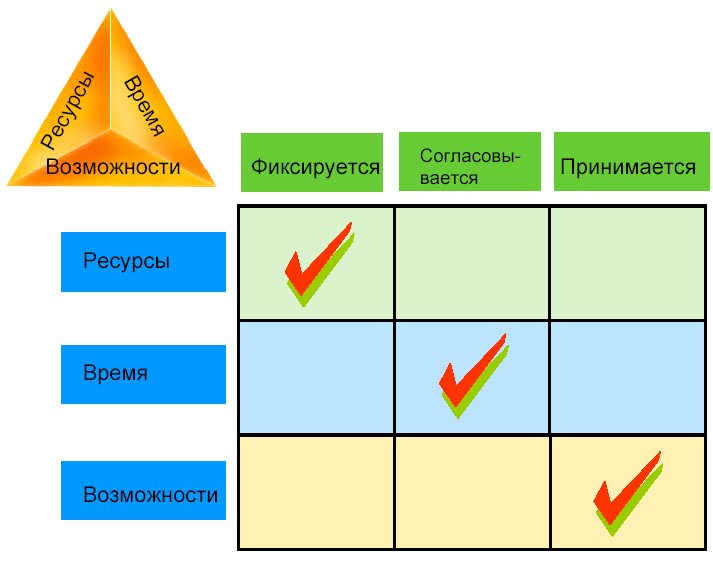


Рисунок 2. Матрица компромиссов\*

Другое весьма полезное средство для управления проектными компромиссами – матрица компромиссов проекта (project tradeoff matrix), показанная на

рис. 2. Она отражает достигнутое на ранних этапах проекта соглашение между проектной группой и заказчиком о выборе приоритетов в возможных в будущем компромиссных решениях. В определенных случаях из этой приоритезации могут делаться исключения, но в целом следование ей облегчает достижение соглашений по спорным вопросам.

Рис. 2 показывает матрицу компромиссов проекта, используемую обычно проектными группами Майкрософт. Она помогает обозначить проектное ограничение, воздействие на которое практически невозможно (колонка “Фиксируется”), фактор, являющийся в проекте приоритетным (колонка “Согласовывается”), и третий параметр, значение которого должно быть принято в соответствии с установленными значениями первых двух величин (колонка “Принимается”).

Принципиально важно наличие у проектной группы и заказчика единого, однозначного взгляда на матрицу компромиссов проекта.

Укажите здесь оценки (в выбранных вами единицах) параметров треугольника компромиссов: ресурсы, которыми располагает ваш проект; имеющееся для реализации проекта время; возможности решения, которые, согласно описанию в документе “Концепция проекта”, будут вами реализованы. Расставьте приоритеты и постройте на их основе матрицу компромиссов.

## 1.2. Вехи проекта

Вехи проекта – это существенные события в жизни проекта. На фазе выработки концепции обычно “выставляются” внешние вехи, которые показывают достижение видимых целей проекта. На самом верхнем уровне в качестве внешних вех могут рассматриваться окончания фаз выполнения проекта. В зависимости от природы проекта вехи могут быть основаны на финансовых затратах, на прогрессе в проекте, на временных интервалах и т.д.

Раннее определение вех позволяет установить точки на временном графике проекта, по которым заинтересованные стороны и проектная группа будут судить о ходе его выполнения.

Выберите и аргументируйте подход к определению вех в вашем проекте. Определите и опишите здесь вехи, на которые вы будете ориентироваться в ходе выполнения проекта.

Вехи проекта выбраны согласно каскадной модели разработки, в силу трудностей возникающих со спиральной, итерационной моделью: не возможностью тестирования продукта, обладающего неполной функциональностью, сложностью в организации работы команды.

* Определение концепции проекта
* Разработка проектной документации
  + Концепция проекта
  + Структура проекта
  + Функциональная спецификация
  + Спецификации и сценарии тестов
  + Тестирование и отчеты об ошибках
  + Постпроектный анализ
* Разработка прототипа
  + Настройка окружения разработчика
  + Создание простого web-сервера, в котором используется база данных, отображается web-страница с кнопкой, и реализована функция, принимающая запрос по нажатию кнопки.
  + Дополнение простого web-сервера расширенным набором html-шаблонов для регистрации, авторизации, отображения списка алгоритмов, добавления и тестирования алгоритмов. Подразумевается, что на данном этапе такие страницы содержат минимум функционала.
* Разработка независимых компонент
  + Разработка детальных html-шаблонов
  + Обработка запросов с html страниц
  + Проработка архитектуры базы данных
  + Реализация компоненты, отвечающей за построение алгоритмов
    - построение алгоритмов, написанных на C/C++
    - построение алгоритмов, написанных на C#
    - построение алгоритмов, написанных на Pascal
  + Реализация компоненты, отвечающей за тестирование алгоритмов
* Интеграция компонент
* Разработка тестов и тестирование

## 1.3. Сметы проекта

Выполнение проекта предполагает использование ресурсов и подразумевает наличие затрат. Ресурсы включают в себя людей, оборудование, различные расходные материалы и т.д. Затраты рассчитываются на основе тарифов (расценок) на каждый вид требуемого ресурса. В данном разделе должна быть представлена информация о:

§ списке видов ресурсов,

§ требуемом количестве каждого ресурса,

§ тарифе на каждый вид ресурса,

§ общей стоимости каждого ресурса,

§ общей стоимости всех ресурсов, необходимых проектной группе.

На основе информации данного раздела должен рассчитываться бюджет проекта. Также этот этап – хорошая возможность идентифицировать специфические ресурсы, которые могут потребоваться для выполнения проекта.

Укажите, какие виды ресурсов необходимы вашей проектной группе для выполнения проекта, определите в выбранных единицах количество каждого ресурса, оцените общие ожидаемые затраты по ресурсам.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Вид ресурса** | **Наименование** | **Стоимость \ Тариф** | **Количество** | **Общая стоимость** |
| Человек | Разработчик | 10 у.е. | 4 | 40 у.е. |
| Аппаратное обеспечение | Сервер | 7 у.е. | 1 | 7 у.е. |
| Аппаратное обеспечение | ПК | 5 у.е. | 4 | 20 у.е. |
| Программное обеспечение | OS Windows 7/8/8.1 | 2 у.е. | 5 | 10 у.е. |
| Программное обеспечение | Python 2.7 | 0 у.е. | 5 | 0 у.е. |
| Программное обеспечение | Django 1.7 | 0 у.е. | 5 | 0 у.е. |
| Программное обеспечение | SQLite | 0 у.е. | 5 | 0 у.е. |
| Программное обеспечение | Компиляторы C/C++, C#, Pascal | 1 у.е. | 5 | 5 у.е. |
| Программное обеспечение | Notepad++ | 0 у.е. | 4 | 0 у.е. |
| Программное обеспечение | Google Chrome | 0 у.е. | 4 | 0 у.е. |
| Программное обеспечение | Skype | 0 у.е. | 4 | 0 у.е. |

Итого общая стоимость проекта: 82 у.е..

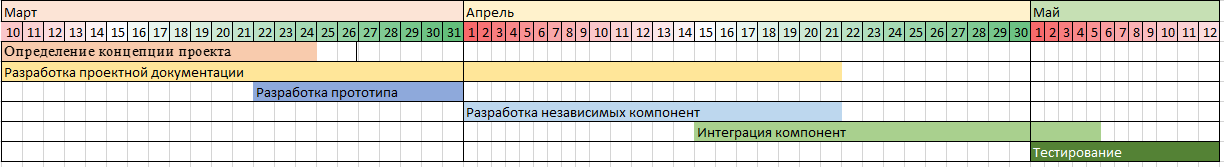
## 1.4. План-график проекта [TB AR: сделать отдельный док с мелкими вехами]

На этом этапе строится первый вариант графика выполнения проекта на основе выделенных вех и уже сформулированных задач, для каждой из которых задаются даты начала и окончания.

Процесс построения графика проекта итеративен. На фазе выработки концепции график строится на основных вехах проекта. На фазе планирования график становится более детальным в процессе выделения отдельных задач проекта.

На основе выделенных выше вех, а также сформулированных целей и задач определите ожидаемую временную протяженность проекта и постройте предварительный график выполнения проекта.

|  |  |  |
| --- | --- | --- |
| **№** | **Наименование работ** | **Сроки** |
| 1 | Определение концепции проекта | 10.03.2015 - 24.03.2015 |
| 2 | Разработка проектной документации | 10.03.2015 - 21.04.2015 |
| 3 | Разработка прототипа | 22.03.2015 - 31.03.2015 |
| 4 | Разработка независимых компонент | 01.04.2015 - 21.04.2015 |
| 5 | Интеграция компонент | 15.04.2015 - 05.05.2015 |
| 6 | Разработка тестов, тестирование, исправление ошибок | 01.05.2015 - 12.05.2015 |



2. Роли и ответственности

В данном разделе описывается организация проектной группы. Четкие требования к квалификации, ролям и ответственностям членов проектной группы позволяют менеджеру проекта правильно подобрать команду и дают каждому понимание его личного вклада в общий успех проекта.

## 2.1. Знания, умения и навыки

Участники проектной группы должны удовлетворять определенным требованиям для успешного выполнения проекта. Эти требования выражаются в терминах знаний, умений и навыков и должны включать технические, управленческие и иные возможности.

На основе имеющейся к данному моменту информации о разрабатываемом решении и способе его реализации сформулируйте требования к знаниям, умениям и навыкам, которые необходимы участникам проектной группы. В процессе формулирования учитывайте, что требования должны касаться всех ролевых кластеров команды.

|  |  |
| --- | --- |
| **Имя** | **Навыки** |
| Бышева Татьяна | Python, Django, C/C++, JavaScript, MySQL |
| Бобыльков Федор | C/C++, C#, Java, JavaScript |
| Лялюшкин Николай | Python, C/C++, Java, MySQL |
| Краснояров Никита | C/C++, Java, MySQL |

## 2.2. Структура команды

Структура команды определяет организационные единицы (менеджер проекта, спонсоры, лидеры команд, и т.д.), задает отношения между ними и зоны их ответственности.

Распределите роли, в вашей команде и опишите здесь это распределение.

|  |  |
| --- | --- |
| **Имя** | **Роль** |
| Бышева Татьяна | Архитектор, Разработчик |
| Бобыльков Федор | Разработчик, Тестировщик |
| Лялюшкин Николай | Разработчик, Менеджер |
| Краснояров Никита | Разработчик, |

# 3. Протоколы проекта

Протоколы проекта – это набор описаний процессов в проекте, которые должны быть утверждены, чтобы все участники команды действовали в одинаковом ключе. Стандартизация повышает производительность и облегчает общение как внутри проектной группы так и с заинтересованными сторонами.

## 3.1. Управление конфигурацией

Многие составляющие решения в процессе работы над проектом претерпевают неоднократные изменения, следовательно, проект нуждается в некоторой стратегии управления конфигурацией, определяющей выбранные методы отслеживания изменений, ведения отчетности и др. Управление конфигурацией должно охватывать проектную документацию, среды разработки и тестирования и любые изменения в проектной среде в целом.

Данный раздел включает:

§ описание методов и средств управления конфигурацией;

§ описание шагов по запросу и принятию изменений в конфигурации;

§ роли и ответственности в управлении конфигурацией;

§ выбор и/или описание требований к системе контроля версий.

Опишите стратегию управления конфигурацией в проекте в соответствии с представленными выше пунктами.

* В качестве системы контроля версий для хранения документации и исходного кода используется система GitHub
* Запросы по внесению изменений принимаются в виде Pull request на GitHub
* Вся команда обязуется перед вливанием исходного кода в основную ветку проверять корректность своих изменений
* Одобрение Pull request производит менеджер проекта после обсуждения с заказчиком и командой
* Ответственность за конкретные проблемы в коде лежит на ответственном за данную компоненту разработчике
* Система контроля версий должна позволять вести разработку в отдельных ветках с возможностью объединения этих веток
* Перед вливанием кода в общую ветку, автор изменения должен пройти review и получить одобрение от двух участников команды

## 3.2. Управление изменениями

Одна из существенных идей, лежащих в основе объектно-ориентированного подхода – ориентированность на изменения. В то же время порядок внесения предложений, утверждения и реализации изменений и отслеживания результатов должен быть определен и зафиксирован. Кроме того, все изменения в проекте должны проводиться в соответствии с принятыми и утвержденными рамками решения и рамками проекта. В данном разделе должны быть описаны:

§ процесс управления изменениями;

§ форма запроса на изменение;

§ роли и ответственности в процессе управления изменениями;

§ действия в случае, если предложенные изменения могут отразиться на контракте с заказчиком (в том числе, если эти действия были инициированы самим заказчиком, см. треугольник компромиссов).

Опишите стратегию управления изменениями в проекте в соответствии с представленными выше пунктами.

Заказчик может запросить новые требования к продукту с помощью Pull Request в GitHub. После обсуждения с менеджером команды, командой и заказчиком принимается решение о принятии или отклонении данного запроса. В процессе обсуждения решение принимается на основе описанных в проектной документации рамок проекта, компромиссов и имеющихся ресурсов.

## 3.3. Управление внедрениями

Работа над решением не заканчивается с окончанием этапа разработки. Важным аспектом, требующим внимания, является и процесс внедрения как пилотных так и финальных версий решения. В данном разделе должны быть описаны способы и средства по подготовке релизов и управлению их внедрением, как в тестовые, так и в производственные окружения.

Сформулируйте и опишите здесь мероприятия, необходимые для будущего внедрения решения, которое будет разработано в ходе проекта.

Продукт поставляется в виде исходного кода. Инсталляция продукта производится путем запуска установочного bat-файла, разворачивающего окружение (веб-сервер, база данных, зависимости).

Описать процесс релиза

## 3.4. Достижение качества проекта

Данный раздел описывает, каким образом в ходе выполнения проекта будут удовлетворяться ожидания заказчика и будущих пользователей к качеству создаваемого решения, как с точки зрения разработчиков, так и со стороны управления. Достижение требуемого качества решения должно быть описано через:

§ ожидания к качеству решения;

§ процесс проверки качества;

§ процесс управления достижением качества;

§ роли и ответственности в процессе достижения качества.

Сформулируйте и опишите стратегию по достижению требуемого качества проекта.

* CleanCode
* Code Review
* Refactoring
* Post-commit testing
* Разработчик, ответственный за тестирование, после внедрения каждой новой функциональности обязан написать тест на эту функциональность
* Каждый разработчик, добавляющий новую функциональность, ответственен за её корректность

## 3.5. Рабочая среда проекта

В данном разделе должен быть описан подход к созданию рабочей среды проекта, определяющий как организационные требования (размещение членов команды, комнаты для митингов и т.д.), так и требования к оборудованию (компьютеры, столы, телефоны и т.д.). Здесь же должны определяться требования к инструментам и системам, таким как системы контроля версий, среды разработки, инструменты тестирования и др.

* Сформулируйте и опишите требования к рабочей среде проекта.
* Члены команды размещаются каждый у себя дома
* Командные собрания проиходят в режиме онлайн посредством Skype-конференции
* Очные командные собрания проходят в случайных местах
* Персональные компьютеры должны удовлетворять требованиям, описанным в документе “Концепции проекта”, п.2.4.2.
* Требования к программным средствам:
  + Skype
  + Notepad++
  + Django 1.6 (1.7)
  + SQLite
  + MS VC 2010
  + C# scs.exe (.NET 4.\*)
  + Free Pascal
  + Google Chrome
  + Tortoise Git / Git Bash
* Наличие высокоскоростного доступа в Интернет

[\*] В документе использованы материалы белых книг (white papers) “MSF Process Model”, “MSF Risk Management Discipline”, “MSF Team Model” (<http://www.microsoft.com/msf>), их переводов “Модель процессов MSF”, “Дисциплина управления рисками MSF”, “Модель проектной группы MSF” выполненных в 2003 году корпораций eLine Software ([http://www.elinesoftware.com](http://www.elinesoftware.com/)), а также официальных курсов Microsoft 2710B и 1846A.

\* Рисунок заимствован из белой книги (white paper) “Модель процессов MSF” в переводе корпорации eLine Software 2003 года.