

Документы ЖЦ программного проекта в ВКРБ

Раздел 1. Бизнес-анализ и моделирование предметной области (анализ предметной области)

Исходные данные к проекту. Постановка задачи. История вопроса. Описание окружения. Основные требования (неформальное изложение)

1.1 Анализ предметной области (аналитический обзор).

Первичный сбор материалов (литература, интернет-источники) по всем этапам разработки, ключевым моментам, используемым средствам, существующим решениям. Классификация материала по темам: **существующие аналоги (прототипы), средства разработки, библиотеки, стандарты, проблемы, графический интерфейс, алгоритмы** (в пределах 10 рубрик по 5-15 ссылок в каждой).

Анализ существующих решений и прототипов: выбор классификационных признаков, например, виды поддерживаемого функционала, наличие открытого кода, свободное/коммерческое ПО, объем разработки, год выпуска, стоимость. Краткая аннотация по каждому прототипу. Сводная таблица характеристик. Определение характеристик собственной разработки («экологическая ниша», преимущества).

Сюда же: анализ средств разработки, готовых частных решений, фреймворков по той же схеме: аннотированный обзор, классификационные признаки, сводная таблица характеристик, обоснование выбора. Анализ должен производиться с точки зрения основных требований со стороны системы разработки (например, платформенная независимость, совместимость).

1.2 Системная инженерия (системный анализ)

Краткое описание системы и ее окружения. Список заинтересованных лиц (стейкхолдеров), в т.ч. разработчики, службы сопровождения и эксплуатации, заказчик, пользователи, руководитель ВКРБ, секретарь ГЭК (приемка), «злоумышленники». Описание их интересов в системе и целей, из которых следуют эти интересы. Определение основных **архитектурных видов (аспектов) и архитектурных точек зрения** (см. 6.1. Системная инженерия в папке «Программная инженерия»).

1.3 Глоссарий

Содержательная (не масло масляное) описание **наиболее важных** элементов функционала, архитектуры, реализации, особых или жаргонных терминов, используемых в разработке.

1.4 Бизнес-анализ

- 1. бизнес-требования** – обоснование полезности проекта, особенности проекта, обеспечивающие его привлекательность, предполагаемые отличия от аналогов, проблемы предметной области и способы их решения в проекте, возможности коммерческого использования, способы монетизации.
- 2. границы проекта** – перечень бизнес-процессов, поддерживаемых и не поддерживаемых системой.
- 3. перечень пользователей проекта**, их квалификация, типовые задачи, которые ими решаются, специфические требования

4. словесное описание бизнес-процессов или использование диаграмм BPMN (см. папку «Бизнес-анализ» в программной инженерии)

1.5 Видение проекта

Основные данные фазы исследования, включенные в видение:

1. название проекта
2. цели проекта
3. результаты проекта
4. допущения и ограничения
5. ключевые участники и заинтересованные стороны
6. ресурсы проекта
7. сроки
8. риски
9. критерии приемки
10. обоснование полезности проекта

1.6 Моделирование предметной области

1. Модель предметной области как таковой (если проект встраивается в существующие бизнес-процессы)
2. Представление предметной области в программной системе
3. Классы бизнес-объектов на этапе проектирования

Раздел 2. Системная аналитика и архитектурное проектирование (структурное проектирование)

2.1 Анализ требований

1. Диаграммы прецедентов
2. Ключевые сценарии
3. Иерархический список требований, в т.ч. вносимых на более поздних этапах проектирования (GUI, архитектура, системные требования, форматы)

2.2 Графический интерфейс

1. Подробная диаграмма экранов, количественные данные в таблице (ссылки, активные элементы (списки, кнопки, поля, события))
2. Графический дизайн
3. Подсказки, справки в программе, ориентация на различных пользователей, программная документация
4. Обсуждение вопросов поиска, навигации, визуализации данных.

2.3 Архитектура

1. Ключевые вопросы архитектурного решения (основные идеи)
 2. Компоненты архитектуры: слои, интерфейсы, протоколы, сервисы, подсистемы
 3. Конкретные решения, паттерны, стили, фреймворки (и реализуемые ими перечисленные выше компоненты)
 4. стандарты представления и обмена данными (форматы, протоколы)
 5. Описание поведения в наиболее существенных или критических состояниях или сценариях.
- Описание архитектуры согласно определения:

6. **значимые решения** по поводу организации ПС
7. **структурные элементы и их интерфейсы**, при помощи которых компонуется система
8. **поведение** - взаимодействие между этими элементами
9. **компоновка элементов** в иерархию подсистем
10. **стиль архитектуры** который направляет эту организацию

Краткое описание специфики архитектуры с каждой их сторон ее представления (модель 4+1): функциональное, логическое, процедурное, реализация, развертывание.

Раздел 3. Детальное проектирование и конструирование (функциональное проектирование)

3.1 Детальное проектирование

Архитектурные решения по перечисленному списку ключевых моментов проектирования:

1. ключевые идеи и концепции: абстрагирование, связность и соединение, модульность и декомпозиция, инкапсуляция, разделение интерфейса и реализации
2. параллелизм
3. контроль и обработка событий
4. распределение компонентов
5. обработка ошибок и исключительных ситуаций и обеспечение отказоустойчивости;
6. взаимодействие и представление (Model-View-Controller - MVC)
7. сохраняемость данных (доступность «долгоживущих» данных)
8. виды представления архитектуры - структурное, поведенческое, логическое, физическое, реализация кода
9. архитектурные стили
10. шаблоны проектирования
11. методы проектирования: нисходящее проектирование, модульное, абстрагирование, итеративность, функционально-ориентированное или структурное проектирование, объектно-ориентированное проектирование, проектирование на основе структур данных, компонентное проектирование

Архитектурные классы и диаграммы классов: архитектурные слои: представление, контроллеры, бизнес-модель, DAO (ORM), БД, коммуникации (сеть, протоколы), валидность, исключения.

Основные документы по проектированию:

1. Структура БД
2. ORM- или DAO-объекты
3. Основные бизнес-объекты
4. Форматы файлов (сохранение данных, импорт/экспорт, конфигурирование и настройка)
5. Описание протоколов: базовый протокол (web-сокеты, http, tcp/ip), форматы сообщений, процедуры обмена, синхронный/асинхронный, симметричный/не-симметричный, буферизация и мультиплексирование данных, параллелизм, синхронизация.
6. Форматы собственных API проекта
7. Диаграммы состояний – автоматные модели для описания поведения сущностей системы, состояния протокола и т.п..

3.2 Конструирование

Прототип программного проекта в среде разработки в виде набора классов-пустышек. Пакеты: базовые интерфейсы и сервисы, паттерны (параллелизм, исключения, валидность данных), файлы и сериализация, коммуникация к БД, DAO (ORM-классы), бизнес-модель,

контроллеры, представление (View), коммуникации с сетью (линейные и протокольные компоненты). Определяются абстракции (абстрактные БД и интерфейсы), а также классы-наследники с соответствующим наполнением. Класс содержит основные данные (свойства), а также заголовки наиболее существенных методов. Ориентировочно в прототипе должно быть порядка 50-70 классов и интерфейсов и порядка 10 пакетов.

По прототипу должна быть справка: количество пакетов, классов, свойств, методов. Эти данные используются при оценке проекта. Данные можно представить в табличной форме

Раздел 4. Тестирование, сопровождение, управление проектом

4.1 Тестирование

1. Модульное тестирование (примеры модульных тестов, возможно в табличной форме)
2. Интеграционное тестирование (описание тестовых сценариев, тестовых наборов, результаты)
3. Специфические тестирование (например, производительности)

4.2. Документация по установке и использованию

1. Руководство по сборке проекта
2. Руководство по установке, конфигурированию, настройке, в т.ч описание конфигурационных и настроечных файлов и процедур
3. Краткое руководство пользованием, примеры базовых сценариев

4.3. Управление проектом

1. Оценка трудоемкости и сроков исполнения проекта по COCOMO и по «смете». Во втором случае из проектной документации берутся данные о количестве компонент разработки (таблиц БД, полей, запросов, окон, событий) и берется **примерная** оценка трудоемкости на единицу
2. Список исполнителей проекта, иерархическая структура работ, расписание