

# Архитектура программных систем

## Курсовая работа

### Разработка архитектуры распределенной программной системы

#### Цель работы

Изучить модели и методы проектирования и документирования архитектуры программных систем, получить навыки разработки архитектуры распределенной программной системы на примере учебного проекта.

#### Теоретические сведения

Архитектура ПО обычно содержит несколько видов, которые аналогичны различным типам чертежей в строительстве зданий. В онтологии, установленной ANSI / IEEE 1471—2000, виды являются экземплярами точки зрения, где точка зрения существует для описания архитектуры с точки зрения заданного множества заинтересованных лиц.

Примеры видов:

- Функциональный/логический вид
- Вид код/модуль
- Вид разработки (development)/структурный
- Вид параллельности выполнения/процесс/поток
- Физический вид/вид развертывания
- Вид с точки зрения действий пользователя
- Вид с точки зрения данных

Хотя было разработано несколько языков для описания архитектуры программного обеспечения, в настоящий момент нет согласия по поводу того, какой набор видов должен быть принят в качестве эталона. В качестве стандарта «для моделирования программных систем» был создан язык UML.

Существуют следующие базовые фреймворки (англ. software architecture frameworks), относящиеся к области архитектуры ПО:

- Модель представления архитектуры «4+1» Филиппа Крутхена

- RM-ODP (Reference Model of Open Distributed Processing)
- Service-Oriented Modeling Framework (SOMF)

### **Задание**

Сформулировать техническое задание на разработку программной системы, разработать документ «Архитектура программной системы» в соответствии с моделью «4+1» и с использованием диаграмм UML, ER.

В курсовой работе ДОЛЖНА рассматриваться архитектура распределенной программной системой, включающей, как минимум, 2 независимо функционирующих процесса (приложения).

(Рекомендуется взять за основу тему НИРС или дипломного проекта).

### **Содержание отчета**

1. Титульный лист.
2. Цель работы, задание с указанием выбранной предметной области.
3. Содержание.
4. Введение *(с обоснованием цели и актуальности поставленных задач)*.
5. Техническое задание на программную систему.

(краткое изложение сути проекта, автоматизируемых процессов, основных пользователей, модели вариантов использования, функций системы)

#### 6. Архитектура системы

##### 6.1. Введение\*

(общее описание концепции архитектуры системы, дающее понимание «замысла» архитектуры)

##### 6.2. Логическое представление\*

(декомпозиция системы на модули, решающие функциональные задачи пользователей, и логика взаимодействия этих модулей)

##### 6.3. Представление разработки\*

(декомпозиция системы на независимо разрабатываемые компоненты и взаимодействие этих компонентов, определение технологии реализации компонентов, интерфейсов/протоколов взаимодействия)

#### 6.4.Процессное представление\*

(декомпозиция системы на параллельные процессы и технологии взаимодействия этих процессов)

#### 6.5.Представление развертывания\*

(распределение модулей системы по узлам сети, аппаратные и программные платформы для узлов сети, технологии/протоколы взаимодействия этих узлов)

#### 6.6.Представление данных

(концептуальная модель данных, представленная диаграммой «сущность-связь»)

#### 6.7.Ключевые сценарии\*

(описание реализации ключевых вариантов использования в терминах компонентов, определенных в п.4.1-4.6)

7. Заключение *(последовательное, логически стройное изложение полученных итогов и их соотношение с общей целью и конкретными задачами, поставленными во введении).*
8. Список использованных источников.
9. Приложения *(при наличии).*

В описание архитектуры должны входить обязательные разделы и представления, отмеченные «\*». Остальные представления могут быть включены в зависимости от специфики системы.

Для каждого представления архитектуры документируется структурный аспект с использованием подходящей диаграммы UML (классов, компонентов, развертывания и т.д.) и, при необходимости пояснения сложных моментов,

поведенческий аспект с использованием подходящих диаграмм взаимодействия (последовательности, коммуникации и т.д.).

### **Контрольные вопросы**

1.

### **Литература**

1. Денис Иванов, Федор Новиков Документирование архитектуры программных систем средствами UML / [http://uml3.ru/library/architecture\\_via\\_uml.pdf](http://uml3.ru/library/architecture_via_uml.pdf)
2. Э. Браудэ Технология разработки программного обеспечения / Э. Браудэ. – СПб.: Питер, 2004. – 655 с.
3. P. Clements Documenting Software Architectures. Views and Beyond. Second Edition / Paul Clements, Felix Bachmann, Len Bass, David Garlan, James Ivers, Reed Little, Paulo Merson, Robert Nord, Judith Stafford. – Addison Wesley, 2010. – 540 с.

### **Варианты**

Выбираются и предлагаются студентами самостоятельно исходя из темы НИРС или дипломного проекта.