

# Вопросы к зачёту «Проектирование и архитектура программного обеспечения»

Юрий Литвинов  
yurii.litvinov@gmail.com

1. Понятие архитектуры, профессия «Архитектор».
2. Архитектурные виды.
3. Роль архитектуры в жизненном цикле программного обеспечения.
4. Понятие декомпозиции. Модульность, связность, сопряжение, сложность.
5. Понятия класса и объекта, абстракция, инкапсуляция, наследование.
6. Принципы выделения абстракций предметной области.
7. Принципы SOLID.
8. Закон Деметры. Некоторые принципы хорошего объектно-ориентированного кода.
9. Моделирование, визуальные модели, виды моделей.
10. Язык UML. Проектирование структуры системы, диаграммы классов.
11. Диаграммы объектов, диаграммы пакетов UML.
12. Диаграммы компонентов, диаграммы развёртывания UML.
13. Диаграмма случаев использования UML.
14. Диаграмма активностей UML.
15. Диаграммы конечных автоматов UML.
16. Диаграммы последовательностей UML.
17. Диаграммы «Сущность-связь».
18. Понятие архитектурного стиля, трёхзвенная архитектура.
19. Model-View-Controller, Sense-Compute-Control.
20. Слоистый стиль, «Клиент-сервер».
21. Гексагональная архитектура, луковая архитектура.
22. Чистая архитектура.
23. Пакетная обработка, каналы и фильтры.
24. Стил Blackboard.

25. Событийно-ориентированные стили, Publish-Subscribe, событийная шина.
26. Понятие Domain-Driven Design, единый язык.
27. Изоляция предметной области в DDD, антипаттерн «Умный GUI».
28. DDD, основные структурные элементы модели предметной области.
29. DDD, паттерн «Агрегат».
30. DDD, паттерны «Фабрика», «Репозиторий».
31. DDD, паттерн «Спецификация».
32. Паттерн «Компоновщик».
33. Паттерн «Декоратор».
34. Паттерн «Стратегия».
35. Паттерн «Адаптер».
36. Паттерн «Заместитель».
37. Паттерн «Фасад».
38. Паттерн «Мост».
39. Паттерн «Приспособленец».
40. Паттерн «Фабричный метод».
41. Паттерн «Абстрактная фабрика».
42. Паттерн «Одиночка».
43. Паттерн «Прототип».
44. Паттерн «Строитель».
45. Паттерн «Наблюдатель».
46. Паттерн «Шаблонный метод».
47. Паттерн «Посредник».
48. Паттерн «Команда».
49. Паттерн «Цепочка ответственности».
50. Паттерн «Состояние».
51. Паттерн «Посетитель».
52. Паттерн «Хранитель».

53. Паттерн «Интерпретатор».
54. Паттерн «Итератор».
55. Антипаттерны «Круговая зависимость», «Последовательная связность».
56. Антипаттерны «Вызов предка», «Проблема Йо-Йо».
57. Антипаттерны «Активное ожидание», «Скрытие ошибки».
58. Антипаттерны «Магические числа», «Магические строки».
59. Антипаттерн «Божественный объект».
60. Антипаттерн «Поток лавы».
61. Антипаттерн «Функциональная декомпозиция».
62. Антипаттерн «Золотой молоток».
63. Понятие распределённой системы, заблуждения при проектировании распределённых систем.
64. RPC, RMI. Пример: gRPC.
65. Веб-сервисы, SOAP. WCF.
66. Веб-сервисы, REST. ASP.NET Web APIs.
67. Архитектурные стили распределённых приложений: Big Compute, Big Data.
68. Web-queue-worker, N-звенная архитектура.
69. Микросервисная архитектура.
70. Дизайн REST-интерфейса.
71. Принципы дизайна распределённых приложений: самовосстановление.
72. Паттерн «Circuit Breaker».
73. Принципы дизайна распределённых приложений: избыточность.
74. Принципы дизайна распределённых приложений: минимизация координации.
75. Command and Query Responsibility Segregation.
76. CAP-теорема, модели ACID и BASE.
77. Принципы дизайна распределённых приложений: проектирование для обслуживания.