

Мидл системный аналитик

4 месяца

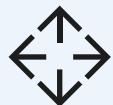
продолжительность курса

5 проектных работ

с обратной связью от эксперта



Вводная часть	Как устроен курс Входное тестирование	1 час
Модуль 1. Программная архитектура системы	Спринт 1. Программная архитектура системы	2 недели
Модуль 2. Стили архитектуры	Спринт 2. Стили архитектуры Спринт 3. Микросервисная архитектура	4 недели
Модуль 3. Системная интеграция	Спринт 4. Моделирование API Спринт 5. Проектирование REST API Спринт 6. Проектирование RPC, Query-Based, Async APIs Спринт 7. Продвинутое проектирование API	6 недель
Модуль 4. Оформление результатов проектирования	Спринт 8. Документирование результатов проектирования	1 неделя
Выпускной проект	Подготовка и защита итогового проекта	3 недели



[1 час]

Познакомитесь с форматом обучения и проверите себя с помощью тестирования.

Темы

1. Как устроен курс

2. Тестирование

Программная архитектура системы

01

[20 часов]

[1 проект]

[1 воркшоп]

В этом модуле вы научитесь выявлять характеристики архитектуры (нефункциональные требования), такие как масштабируемость, производительность, отказоустойчивость и другие. Вы также научитесь выявлять функциональные блоки системы — составлять логическую архитектуру. Разработанные требования и логические компоненты станут фундаментом для принятия решений о выборе стиля программной архитектуры системы и системной интеграции. Наконец, вы научитесь документировать принятые архитектурные решения.

Сprint 1. Программная архитектура системы

1. Измерения архитектуры

2. Характеристики архитектуры

3. Логические компоненты

4. Логическая и физическая архитектуры

5. Связность логических компонентов

6. Стили архитектуры

7. Архитектурные решения

Проектная работа 1. Разработка характеристик архитектуры и логических компонентов приложения

Вы самостоятельно выявите и задокументируете
характеристики архитектуры приложения и его логические
компоненты.

Стили архитектуры

02

[50 часов]

[1 проект]

[1 тест]

[1 воркшоп]

В этом модуле вы узнаете, какие бывают стили программной архитектуры системы. Вы познакомитесь с топологией каждого стиля и его особенностями. Вы научитесь выбирать наиболее подходящий характеристикам архитектуры и логическим компонентам стиль.

Особенно тщательно вы изучите особенности микросервисной архитектуры. Например, вы узнаете, как Domain-Driven Design связан с микросервисной архитектурой, что такое микрофронтиенды, чем хореография сервисов отличается от оркестрации, для чего нужны саги и другое. Также вы научитесь разделению монолитного приложения на микросервисы. Наконец, вы научитесь составлять диаграммы программной архитектуры и документировать принятые архитектурные решения.

Сprint 2. Стили архитектуры

1. Клиент-серверная архитектура

2. Монолитные и распределённые стили

3. Многослойная архитектура

4. Модульный монолит

5. Архитектура, основанная на сервисах

6. Событийно-ориентированная архитектура

7. Сервис-ориентированная архитектура

Тестирование

Проверите себя и закрепите теорию на практике.

Сprint 3. Микросервисная архитектура

- | | | |
|-------------------------------|------------------------------|----------------------------|
| 1. Микросервисная архитектура | 4. Микрофронтенды | 7. Масштабный куб |
| 2. Domain-Driven Design (DDD) | 5. Хореография и Оркестрация | 8. Переход на микросервисы |
| 3. Метод «Event Storming» | 6. Транзакции и Саги | |

Проектная работа 2.

Проектирование программной архитектуры

Вы выберете стиль программной архитектуры приложения на основе характеристик архитектуры. Используя выбранный стиль архитектуры и логические компоненты, вы спроектируете программную архитектуру приложения и задокументируете её. Вам также предстоит обосновать принятые архитектурных решения.

Системная интеграция

03

[85 часов]

[3 проекта]

[1 тест]

[4 воркшопа]

В этом модуле вы освоите проектирование API: от составления профиля до выбора архитектурного стиля (REST, SOAP, gRPC, GraphQL) и форматов данных. Вы научитесь проверять модели через диаграммы последовательности, проектировать REST API с учётом HTTP-протокола и работать с инструментами Swagger и Postman. Отдельно разберём Kafka и RabbitMQ и ключевые аспекты версионирования и безопасности API.

Сprint 4. Моделирование API

1. Стили интеграции

2. Принципы проектирования Web API

3. Процесс разработки Web API

4. Подход API-First

5. Процесс моделирования API

6. Профиль API

7. Диаграмма последовательности (UML Sequence Diagram)

Проектная работа 3.

Моделирование API

Вы составите профиль API на основании Job Stories для приложения. Вы последовательно преобразуете Job Stories в набор операций, ресурсов и событий, которые станут фундаментом для проектирования API.

Сprint 5. Проектирование REST API

- | | | |
|-------------------------------|------------------------------------|------------|
| 1. Стиль REST API | 4. Процесс проектирования REST API | 7. Postman |
| 2. HTTP | 5. OpenAPI Specification | |
| 3. Стили представления данных | 6. Swagger | |

Проектная работа 4. Проектирование REST API

На основании профиля API вы спроектируете REST API для приложения. Для каждой операции из профиля API вы определите пути URL-адресов ресурсов, выберите подходящие HTTP-методы и назначите коды ответов. Вы задокументируете разработанный проект REST API. Вы также поработаете с REST API в Postman.

Сprint 6. Проектирование RPC, Query-Based, Async APIs

- | | | |
|-----------------------------------|-------------------------------------|----------------------------|
| 1. Протокол gRPC | 5. SOAP API | 9. Async API Specification |
| 2. Процесс проектирования RPC API | 6. Основы обмена сообщениями | 10. Выбора стиля API |
| 3. OData | 7. Стили Async API | 11. Apache Kafka |
| 4. GraphQL | 8. Процесс проектирования Async API | 12. RabbitMQ |

Проектная работа 5. Проектирование Async API

На основании профиля API вы спроектируете Async API для приложения и поработаете с Apache Kafka.

Сprint 7. Продвинутое проектирование API

- | | | |
|---|----------------------------------|---|
| 1. Улучшение удобства использования API | 3. Стратегии версионирования API | 5. Топологии API Gateways |
| 2. Улучшение документирования API | 4. Защита API | 6. Управление идентификацией и доступом |

Тестирование

Проверите себя и закрепите теорию на практике.

Оформление результатов проектирования

04

[10 часов]

[1 воркшоп]

В этом модуле вы изучите два самых распространённых стандарта для оформления результатов проектирования, узнаете best practices по презентации результатов проделанной работы, и сможете закрепить этот навык в выпускном проекте.

Сprint 8. Документирование результатов проектирования

1. Стандарты описания программной архитектуры системы

4. Презентация результатов проектирования

Выпускной проект

05

[30 часов]

[3 недели]

С помощью итогового проекта вы подтвердите и интегрируете все приобретённые знания и умения. В нём всё будет как в реальной жизни: размытая постановка от заказчика, самостоятельная работа и строгий ревьюер.

Темы

1. Разработка требований

2. Проектирование программной архитектуры системы

3. Проектирование системных интеграций

4. Презентация результатов проектирования