ПWS

ПОИТ-4

1. Лекции 36 час.
2. Лабораторные работы 32 час.

**Сервис-ориентированная архитектура приложения**

1. **SOA: Service-oriented architecture**, **сервис-ориентированная архитектура** – парадигма проектирования и разработки программного обеспечения, основанная на применении распределенных слабосвязанных компонентов, обеспечивающих стандартные интерфейсы (открытые протоколы).
2. **SOA:** стиль архитектуры информационной системы, который позволяет создавать приложения путем комбинации слабосвязанных распределенных компонент.
3. **SOA:** программная платформа, позволяющая разрабатывать приложения, состоящие из слабо связанных распределенных компонент (AWS, Azure).
4. **SOA:** компоненты распределенной системы SOA – узлы - **сервисы**.
5. **SOA: сервис –** логическое представление реальной программы, базы данных, бизнес-процесса и т.п., определенное в терминах функциональности.
6. **SOA: основные свойства**

* независимость от аппаратной реализации узлов;
* независимость от операционной системы в узлах;
* независимость от языка программирования разработки сервиса;
* масштабируемость.

1. **SOA:** **сервис** – это видимый ресурс, выполняющий повторяющуюся задачу и описанный внешней инструкцией.
2. **SOA:** свойства сервиса

* сервис ориентирован на бизнес;
* сервис автономен;
* повторное использование;
* четко описанная инструкция в терминах интерфейса (SLA);
* сервис виден (доступен).

1. **SOA: SLA – System Level Agreement**  - формальный документ, описывающий перечень услуг и обязательства, а также уровень качества предоставления услуг.

* сроки действия соглашения;
* дни и часы действия сервиса;
* правила безопасности;
* гарантированные характеристики (среднее время отклика, уровень доступности, средняя пропускная способность, … )
* порядок получения отчетов (протоколов);
* процедуры оформления ошибок;
* процедуры оформления заявок на доработки;
* процедуры оплаты;
* и т.д.

1. **SOA: причины**

* распределенный доступ к сервисам:
* снижение затрат на разработку ПО;
* снижение затрат на владение;
* масштабирование.

1. **SOA:**



1. **SOA: ESB – Enterprise Service Bus:** программный компонент обеспечивающий обмен сообщениями между различными информационными системами, имеющих сервис-ориентированную структуру.
2. **SOA: ESB**

* синхронный и асинхронный вызов сервисов;
* гарантированная доставка сообщений;
* поддержка транзакций;
* маршрутизация сообщений;
* мониторинг, аудит и протоколирование.



1. **SOA: специальные компоненты** (как правило часть ESB)

* SOA Registry;
* Workflow Engine;
* Service Broker;
* SOA Supervisor.

1. **SOA:** **SOA Registry –** реестр сервисов, информация о сервисах и их интерфейсах. Gateway Service



1. **SOA:** **Workflow Engine –** программный компонент, предназначенный для построить (построить модель бизнес процесса) и выполнить бизнес процесс на основе группы сервисов. Другими словами разработать новый сервис на основе последовательного выполнения нескольких сервисов.



1. **SOA: Service broker** –программный компонент, позволяющий трансформировать запрос пользователя в системе в запуск и координированную работу.



1. **SOA: SOA Supervisor** – служебный сервис, предназначенный для управления и мониторинга других служебных сервисов.
2. **SOA: Health Check Service** – служебный сервис для проверки работоспособности других сервисов, часто в составе Supervisor; **Health Check API:** API для проверки работоспособности. Системы мониторинга серверов: Nagios, Zabbiks, Icinga (Nagios Core), dotcom.monitor.
3. **SOA: Identity Service -**  сервис аутентификации, предназначен для реализации SSO (Single Sign On), протоколы токен-авторизации (JWT), OpenID Connect, OAuth2. IDaaS. (AWS IAM).
4. **SOA: интерфейсы – REST, SOAP, JSON-RPC**
5. **SOA: способы клиент-серверного взаимодействия**











1. **SOA:** основные стандарты W3С: XML**, SOAP, WSDL, UDDI.**
2. **SOA:** спецификации второго уровня W3C (WS\*):

* WS-Policy, WS-PolicyAttachment, WS-PolicyAssertion (описание политик web-сервиса);
* WS-Addressing (механизм адресации web-cthdbcf);
* WS-Security (целостность и конфиденциальность web-сервисов);
* WS-Trust (механизм получения маркеров защиты);
* WS-SecureConversion (создание безопасной сессии обмена сообщениями);
* WS-SecurityPolicy (определяет набор утверждений политики безопасности);
* WS-Federation (объединение защищенных доменов);
* WS-Transfer (механизм обновления, создания и удаления ресурсов);
* WS-ResourceTransfer, WS-Fragment (обеспечивает частичный доступ к ресурсам);
* WS-MetadataExchange (механизм получения метаданных);
* WS-Enumeration (механизм получения данных больших размеров);
* WS-Eventing (механизм уведомления о событиях web-сервисов);
* WS-Management (SOAP-управление системами);
* WS-Discovery (механизмы публикации и поиска web-сервисов);
* WS-ReliableMessaging, WS-ReliableMessagingPolicy (механизмы надежной передачи сообщений между web-сервисами);
* WS-MakeCpnnection (установка соединения с сервисом не имеющего доступный адрес);
* WS-Coordination (механизмы взаимодействия web-сервисов);
* WS-AtomicTransaction (поддержка транзакций web-сервисов);
* WS-BusinessActivity (координация бизнес-взаимодействия web-сервисов).

1. **Паттерны микросервисов.**
2. **SOA: оркестровка**
3. **SOA: микросервисы**