

Лаб 1. Инверсия контроля

Принцип инверсии контроля (Inversion of control), проблематика, решаемые задачи. Внедрение зависимостей (Dependency injection) как средство реализации инверсии контроля. Преимущества и недостатки построения приложений на основе этого принципа. Фреймворк Spring как средство реализации внедрения зависимостей, аналогичные фреймворки. Конфигурирование внедрения зависимости в Spring. Внешние конфигурации. Автоматическое связывание. Виды автоматического связывания (по типу, по имени). Связывание, основанное на метаданных (аннотациях). Смешанные виды связывания. Инверсия контроля и тестирование приложения.

Лаб 2. Компонентное(сервисное) построение web-приложений.

Принцип внедрения зависимости и бизнес-логика приложения. Применение внедрения зависимостей при создании web-приложений на Java. Внедрение зависимостей и разделение приложения на модель-представление-контроллер. Spring MVC: Параметры контроллера, передача данных во view, контроллеры без view. Формы и валидация. Аспектно-ориентированное программирование. Aspect (аспект), Join point (точка соединения), Advice (совет), Pointcut (срез).

Лаб 3. Динамическая подгрузка модулей приложения

Способы уменьшения связности в компонентных приложениях. Различные этапы связывания компонентов при разработке приложений. Технологии создания плагинов к приложениям. Горячая замена кода. Зависимости между подключаемыми модулями. Конфликты зависимостей в подключаемых модулях и способы их разрешения. ClassLoader-ы в Java. ServiceLoader. OSGI. Динамическая подгрузка модулей в веб-приложениях.

Лаб 4. Событийные распределенные архитектуры

Асинхронные(неблокирующие) операции ввода-вывода, особенности реализации, преимущества и недостатки. Промежуточное ПО для обработки сообщений, брокеры сообщений (AMQP, JMS), темы (topic) сообщений, очереди сообщений. Гарантированная доставка сообщений. Распределение нагрузки при помощи брокеров сообщений. Реализация полнодуплексного соединения в web. Websockets.