Spring

**Framework -** готовый набор инструментов, который помогает разработчику быстро создать продукт.

**Spring –** это Framework который на самом деле состоит из 21 различных активных проектов.

**Проекты:**

**Spring Boot** - это часть фреймворка Spring, которая представляет собой набор инструментов и библиотек для создания приложений корпоративного уровня. Spring Boot упрощает создание автономных приложений производственного уровня, которые можно легко развернуть и запустить с минимальной конфигурацией.

**Spring Framework** - обеспечивает базовую поддержку внедрения зависимостей, управления транзакциями, веб-приложений, доступа к данным, обмена сообщениями и многого другого.

**Spring Data** - обеспечивает согласованный подход к доступу к данным - реляционный, нереляционный, сокращающий отображение и другие.

**Spring Cloud** - Предоставляет набор инструментов для работы с общими шаблонами в распределенных системах. Полезен для создания и развертывания микросервисов.

**Spring Cloud Data Flow** - это облачный набор инструментов для создания конвейеров данных в реальном времени и пакетных процессов. Spring Cloud Data Flow готов для использования в различных случаях использования обработки данных, таких как простой импорт / экспорт, обработка ETL, потоковая передача событий и прогнозная аналитика.

**Spring Security** - защищает ваше приложение благодаря всесторонней и расширяемой поддержке аутентификации и авторизации.

**Spring Authorization Server** - это решение авторизации и аутентификации для небольших проектов с любой архитектурой: монолиты, микро сервисы, который можно гибко настроить под собственные нужны.

**Spring for GraphQL** - это язык запросов с открытым исходным кодом, созданный Facebook в качестве альтернативы распространенной архитектуре REST. Он позволяет пользователям запрашивать конкретные данные и таким образом избежать классической проблемы избыточности данных в ответах, связанной с REST.

**Spring Session** - предоставляет API и реализации для управления информацией о сеансе пользователя.

**Spring Integration** - предназначен для разработки интеграционных решений с событийно-управляемой архитектурой, либо с архитектурой основанной на обмене сообщениями.

**Spring HATEOAS** - это библиотека API-интерфейсов, которую мы можем использовать для простого создания представлений REST, соответствующих принципу HATEOAS (гипертекст как механизм состояния приложения). Вообще говоря, принцип подразумевает, что API должен направлять клиента через приложение, возвращая соответствующую информацию о следующих возможных шагах вместе с каждым ответом.

**Spring REST Docs** - создает документацию для служб RESTful, которая является точной и удобочитаемой. Он сочетает рукописную документацию с автоматически генерируемыми фрагментами документов, созданными с помощью тестов Spring.

**Spring Batch** - это среда, созданная для пакетной обработки больших объёмов данных. В основе Spring Batch лежит понятие задания (Job). Каждое задание может состоять из нескольких этапов (Step). Этап в свою очередь может быть либо фрагментом произвольного кода (Tasklet), либо иметь более сложную структуру, состоящую из считывателей элементов (ItemReader), обработчиков элементов (ItemProcessor) и записывателей элементов (ItemWriter).

**Spring AMQP** - проект Spring AMQP применяет основные концепции Spring к разработке решений для обмена сообщениями на основе AMQP. Мы предоставляем “шаблон” как высокоуровневую абстракцию для отправки и получения сообщений.

**Spring Flo** - это библиотека JavaScript, которая предлагает базовый встраиваемый визуальный конструктор HTML5 для конвейеров и простых графиков. Эта библиотека используется в качестве основы построителя потоков в Spring Cloud Data Flow.

**Spring for Apache Kafka** - применяет основные концепции Spring к разработке решений для обмена сообщениями на основе Kafka. Он предоставляет "шаблон" в качестве высокоуровневой абстракции для отправки сообщений.

**Spring LDAP** - это java-библиотека, которая упрощает операции по протоколу LDAP и основана на шаблоне Spring JdbcTemplate.

**Spring Shell** – позволяет легко создать полнофункциональное приложение shell (оно же командная строка), используя jar-файлы Spring Shell и добавляя свои собственные команды (которые поставляются в виде методов в spring beans). Создание приложения командной строки может быть полезно, например, для взаимодействия с REST API вашего проекта или для работы с содержимым локального файла.

**Spring Statemachine** - это платформа, позволяющая разработчикам приложений использовать концепции конечного автомата в приложениях Spring.

**Spring Web Flow** - позволяет реализовать "потоки" веб-приложения. Поток инкапсулирует последовательность шагов, которые направляют пользователя через выполнение некоторой бизнес-задачи. Он охватывает несколько HTTP-запросов, имеет состояние, обрабатывает транзакционные данные, может использоваться повторно и может быть динамичным и длительным по своей природе.

**Spring Web Services** - это продукт Spring-сообщества, основной целью которого является создание документоориентированных web-сервисов.