**JPA (Java Persistence API)**  
это спецификация Java EE и Java SE, описывающая систему управления сохранением java объектов в таблицы реляционных баз данных в удобном виде. Сама Java не содержит реализации JPA, однако есть существует много реализаций данной спецификации от разных компаний (открытых и нет). Это не единственный способ сохранения java объектов в базы данных (ORM систем), но один из самых популярных в Java мире.

**DispatcherServlet**  
Реализация шаблона проектирования Front Controller, который обрабатывает все входящие веб-запросы к приложению Spring MVC. Шаблон Front Controller (см. Шаблон проектирования корпоративных приложений ) является распространенным шаблоном в веб-приложениях, задача которого заключается в получении всего запроса и маршрутизации его к различным компонентам приложения для фактической обработки.

В случае Spring MVC DispatcherServlet веб-маршрут запрашивает контроллеры Spring MVC.

В Spring MVC DispatcherServlet используется для нахождения правильного контроллера для обработки запроса, что он делает с помощью отображения обработчика, например, @RequestMapping аннотации.

Он также отвечает за делегирование логического имени представления ViewResolver и затем отправку обработанного ответа клиенту.

**Logger (https://youtu.be/j-i3NQiKbcc?t=477)**

Поименованная штука, ее можно создать с помощью фабрики, имя у этой штуки может быть абсолютно любым, но мы используем скучное имя класса. Это удобное выражение, т.к. оно разделено точками. А точки хорошо, они устанавливают отношение «родитель-потомок» между всеми логгерами. Во главе стоит рутовый логгер. Логгер нужен для того, чтобы вызывать события. У события две характеристики: что пишем и уровень относительной ужасности.

**Appender**

Знает, куда отправлять события. Консоль, файл, Socket, DataBase, Абстрактный сферический конь в вакууме.

**Socket**

Соединение между сервером и портом. https://youtu.be/fFekJ7myksk?t=72