Vaadin — свободно распространяемый фреймворк для создания веб-приложений. В отличие от библиотек на Javascript, Vaadin предлагает сервер-ориентированную архитектуру, базирующуюся на Java Enterprise Edition. Использование JEE позволяет выполнять основную часть логики приложения на стороне сервера, тогда как технология AJAX, используемая на стороне браузера, позволяет интерактивно взаимодействовать с пользователем.

**Vaadin Framework:**

Vaadin Framework предоставляет две модели разработки для веб-приложений: для клиентской части (браузер) и для серверной части.

Модель разработки на стороне сервера является более мощной и позволяет разрабатывать приложения исключительно на стороне сервера, используя основанный на AJAX Vaadin Client-Side Engine, который отображает пользовательский интерфейс в браузере. Server-side Vaadin application работает как сервлет, отвечая и принимая запросы. Сервлет принимает запросы от пользователя и интерпретирует их как события сессии пользователя. События связаны с UI компонентами и поступают в event listeners, определенные в приложении. Если UI изменяется на сервере, сервлет отправляет ответ на клиент и тот производит рендер компонентов. Движок на стороне клиента получает ответы от сервлета и изменяет страницу.

Модель на стороне клиента позволяет разрабатывать виджеты и приложения на Java, которые компилируются в JavaScript и выполняются в браузере. Для разработки на стороне клиента используется Vaadin Compiler (основанный на GWT), который позволяет компилировать Java в JavaScript.

**Основные компоненты:**

**User Interface**

UI для взаимодействия с бизнес логикой. UI класс наследует com.vaadin.ui.UI. Тема определяется как аннотация к UI. Нужен для создания пользовательского интерфейса из UI компонентов и настройки event listeners для принятия ввода пользователя. Приложение может работать в разных окнах браузера, каждое имеет уникальный экземпляр UI.

* UI

Пользовательский интерфейс представляет HTML, в котором приложение Vaadin запускается на веб-странице. Разрабатывая приложение с Vaadin Framework, вы расширете класс пользовательского интерфейса и добавлете в него контент. Обычно, когда пользователь открывает новую страницу с URL-адресом UI, для него автоматически создается новый UI (и связанный объект Page). Все они используют один и тот же пользовательский сеанс.

* Page

UI связан с объектом Page, который представляет веб-страницу, а также окно браузера, в котором работает UI.

* Vaadin Session

Объект VaadinSession представляет пользовательский сеанс с одним или несколькими UI, открытыми в приложении. Сеанс начинается, когда пользователь впервые открывает UI приложения Vaadin, и закрывается, когда истекает срок сеанса на сервере или когда он явно закрыт.

* User Interface Components

UI состоит из компонентов, которые создаются приложением. Они расположены иерархически с использованием специальных компонентов макета. Взаимодействие пользователя с компонентами вызывает события (events), связанные с компонентом, которые может обрабатывать приложение. Поля (fields) предназначены для ввода значений и могут быть напрямую связаны с данными. Можно создавать свои собственные компоненты UI с помощью наследования или композиции.

* Events and Listeners

Vaadin Framework следует парадигме программирования, управляемой событиями, в которой события и сушатели, которые обрабатывают события, являются основой для обработки взаимодействия с пользователем в приложении (хотя также возможен server pus).

* Themes

Представление и логика пользовательского интерфейса разделены. Хотя логика UI обрабатывается как код Java, представление определяется в темах как CSS или SCSS. Vaadin включает в себя несколько встроенных тем.

* Data Binding

При связывании данных любое поле (field) в Vaadin Framework может быть привязано к свойствам бизнес-объектов, таких как JavaBeans, и сгруппировано в виде форм. Компоненты могут получать свои значения напрямую, без необходимости какого-либо управляющего кода. Точно так же любой компонент select может быть связан с поставщиком данных, извлекая его элементы из коллекции Java или бэкэнда, такого как база данных SQL.

**User Interface Components/Widgets**

UI состоит из компонентов, созданных приложением. Каждый компонент на сервере имеет соответствующий компонент на клиенте, виджет, с помощью которого он рендерится в браузере и взаимодействует с пользователем. Поля, которые пользователяь может менять могут быт связаны с данными (data sources).

**Client-Side Engine**

Движок рендерит UI в браузере, используя виджеты, которые соотносятся (counterpart) компонентам на стороне сервера. Они отправляют запросы на сервер AJAX и рендерят изменения UI. User Interface Component**s** визуализируются с использованием виджетов (так они называются в GWT) на стороне клиента. Рендеринг, виджеты, а также связь со сервером обрабатываются в ApplicationConnection (Client Side Communication Manager). Соединение виджетов с User Interface Components выполняется в коннекторах. Каркас прозрачно обрабатывает сериализацию состояния компонента и включает в себя механизм RPC между двумя сторонами.

**Vaadin Servlet**

Серверная часть Vaadin работает поверх API сервлетов Java. Сервлет Vaadin, точнее класс VaadinServlet, получает запросы от разных клиентов, определяет, к какому сеансу пользователя он принадлежит, отслеживая сеансы с помощью файлов cookie, и делегирует запросы соответствующим сеансам.

**Events**

Взаимодействие с компонентами пользовательского интерфейса создает события, которые сначала обрабатываются на стороне клиента виджетами, а затем проходят через HTTP-сервер, сервлет Vaadin и компоненты пользовательского интерфейса слушателям событий, определенным в приложении.

**Server Push**

В дополнение к модели программирования, управляемой событиями, Vaadin поддерживает server push, при котором изменения пользовательского интерфейса передаются непосредственно с сервера на клиент без запроса клиента или события. Это позволяет немедленно обновлять пользовательские интерфейсы из других потоков и других пользовательских интерфейсов, не дожидаясь запроса.

**Data Binding**

В дополнение к модели пользовательского интерфейса Vaadin предоставляет API привязки данных для связи данных, представленных в полевых компонентах, таких как текстовые поля, флажки и компоненты выбора, с источником данных. Используя привязку данных, компоненты пользовательского интерфейса могут обновлять данные приложения напрямую, часто без необходимости какого-либо управляющего кода.

**Client-Side Applications**

В дополнение к серверным веб-приложениям Vaadin поддерживает клиентские прикладные модули, которые запускаются в браузере. Клиентские модули могут использовать те же виджеты, темы, что и серверные приложения Vaadin.

**GWT:**

Клиентская часть Vaadin Framework основана на GWT. Клиентские модули разрабатываются с использованием Java и компилируются в JavaScript с помощью компилятора Vaadin, который является расширением компилятора GWT.

**Vaadin Platform:**

В платформе Vaadin предоставляемые компоненты пользовательского интерфейса основаны на Web Components. Компоненты написаны с использованием библиотеки Polymer, а не GWT. Создание компонента на основе Web Components позволяет использовать его в любой современной веб-инфраструктуре, а не ограничивать его только Vaadin или GWT. Vaadin Flow – это фреймворк Java, и он полностью переписан из Vaadin Framework. Он дает гораздо лучший набор инструментов для создания любых повторно используемых компонентов UI, а не только Web Components, и не заставляет вас использовать GWT для этого.