

FRAMEWORKS DE DESARROLLO

Juan Diego Solís Martínez - 23720

Nils Muralles Morales - 23727

Víctor Manuel Pérez Chávez - 23731

Diego Oswaldo Flores Rivas - 23714

Isabella Recinos Rodríguez- 23003



VueJS



Contextualización

Vue.js es un framework progresivo de JavaScript diseñado para construir interfaces de usuario de manera eficiente.

Se centra en mejorar la experiencia del desarrollo front-end al facilitar la creación de aplicaciones web interactivas.



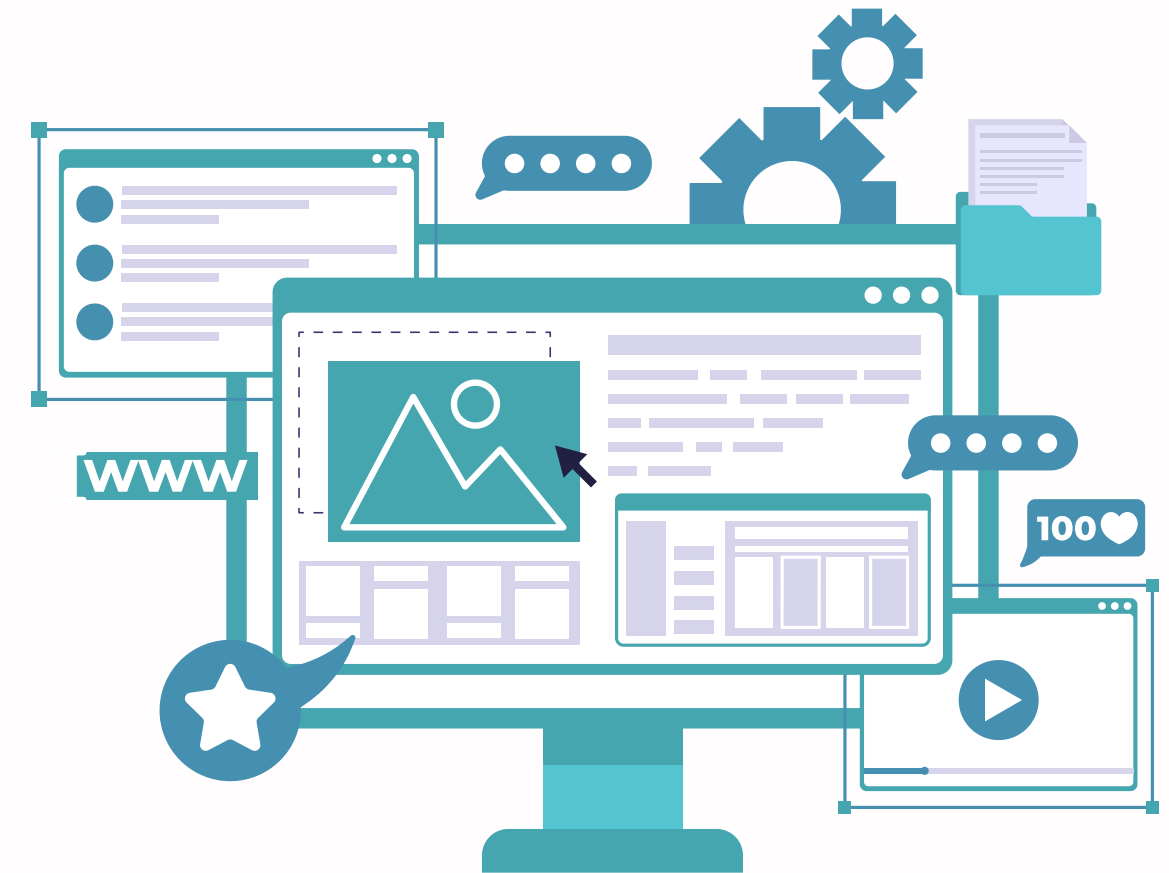
Principios de funcionamiento y componentes:

Reactividad

Enlace de datos

Basado en
componentes

Sintaxis declarativa



Especializado para el desarrollo de:

Front-end

Crea interfaces de usuario y maneja el DOM eficientemente.

Reactividad y gestión de estados

Actualiza automáticamente la UI cuando cambia el estado.

Routing y navegación

Gestiona la navegación con Vue Router.

Integración de APIs

Facilita la conexión con APIs REST o GraphQL usando Axios.

SSR y Mobile

Soporta SSR con Nuxt.js y aplicaciones móviles con Quasar o Ionic Vue.

Combinado con otras tecnologías:

Para Backend: Puede combinarse con Node.js, Django, Laravel u otras tecnologías que manejen la lógica de servidor y la base de datos.

Para Bases de Datos: Se usa con MySQL, PostgreSQL, Firebase, MongoDB, etc.

Para estilos: Puede usarse con CSS puro, Tailwind CSS, Bootstrap o Vuetify.

Patrones de diseño y arquitectónicos que implementa:

Patrones de diseño: Observador, composición y proxy

Patrones arquitectónicos: Basado en componentes, MVC/MVVM y SPA

Recomendado para:

- Aplicaciones de una sola página.
- Dashboards e interfaces interactivas.
- Sistemas de gestión de contenido para crear interfaces actualizables.
- Integrarlo en proyectos existentes.
- Aplicaciones móviles y de escritorio.



Semejanzas y diferencias con otros frameworks:

- Vue, React y Svelte tienen arquitecturas basadas en componentes
- Vue y React utilizan un DOM virtual, mientras que Angular utiliza un DOM incremental y Svelte compila el código sin necesidad de DOM virtual
- Manejo de componentes de manera reactiva.
- En comparación con Angular, React y Svelte, todos estos frameworks permiten tanto SPA como SSR



Ruby on Rails

Descripción

Es un framework del lado del servidor para desarrollo de aplicaciones web, escrito en Ruby.

Cuenta con una amplia biblioteca de gemas (paquetes reutilizables), lo que permite a los desarrolladores construir aplicaciones web robustas y escalables de manera rápida y eficiente.



Principios de funcionamiento:

Model-View-Controller (MVC)

Utiliza el patrón MVC para dividir la aplicación en capas, mejorando la escalabilidad.

Don't Repeat Yourself (DRY)

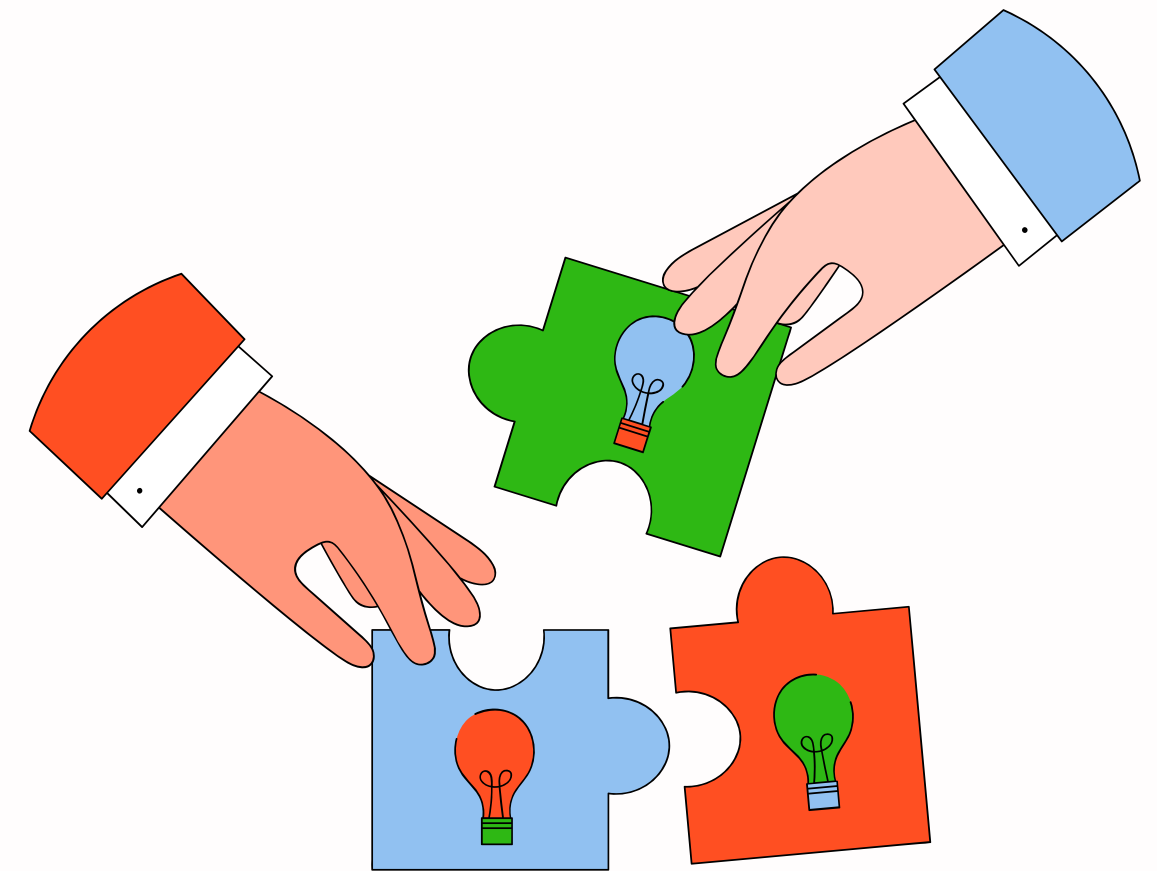
Evitar la repetición para un código mantenible, extensible y libre de bugs.

Convention over Configuration (CoC)

El framework incluye una serie de convenciones por defecto, para que el desarrollador no tenga que definirlas.

Tipo de framework y proceso que cubre.

Framework de desarrollo web full-stack basado en el lenguaje Ruby. Cubre todo el proceso de desarrollo web, desde la gestión de la base de datos hasta la presentación en la interfaz de usuario.

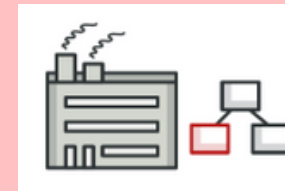


Patrones de diseño y arquitectónicos que implementa:

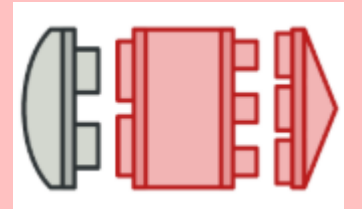
Singleton



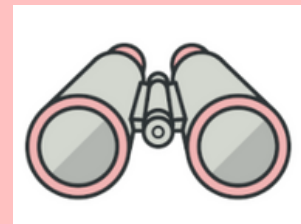
Factory Method



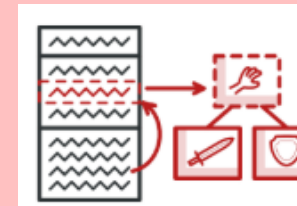
Adapter



Observer



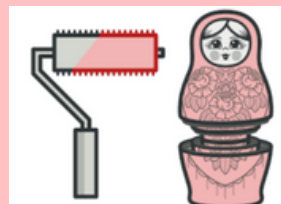
Strategy



Template Method



Decorator

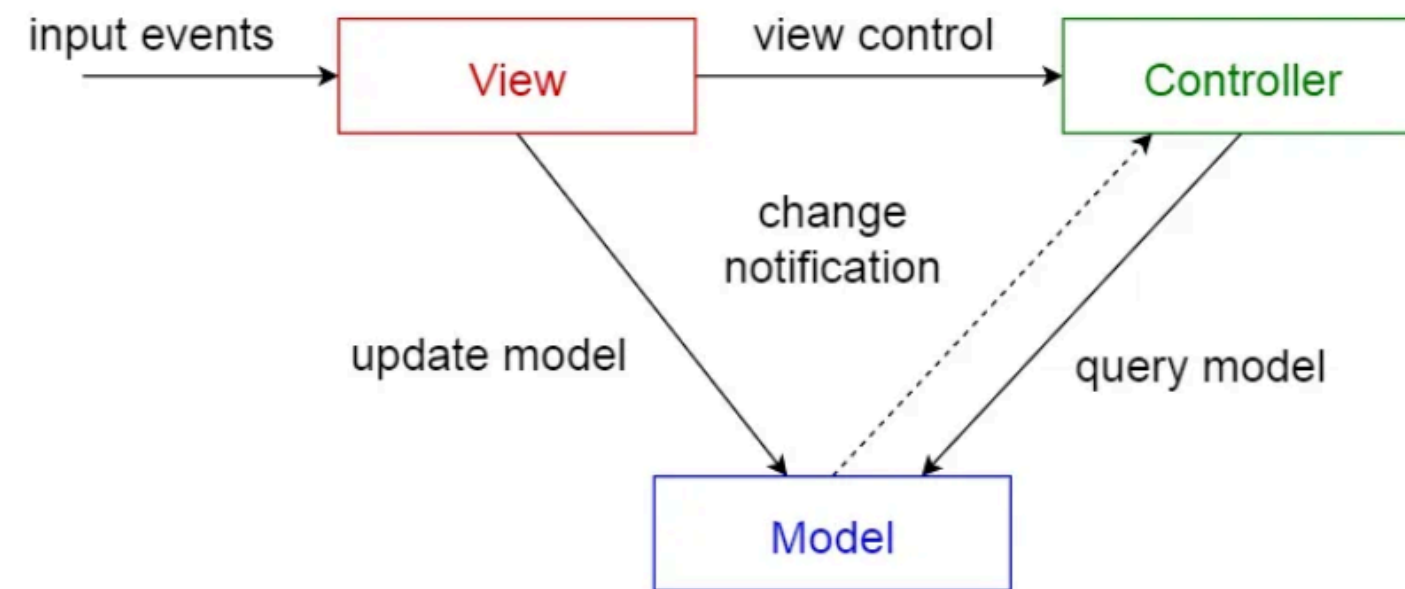


Command

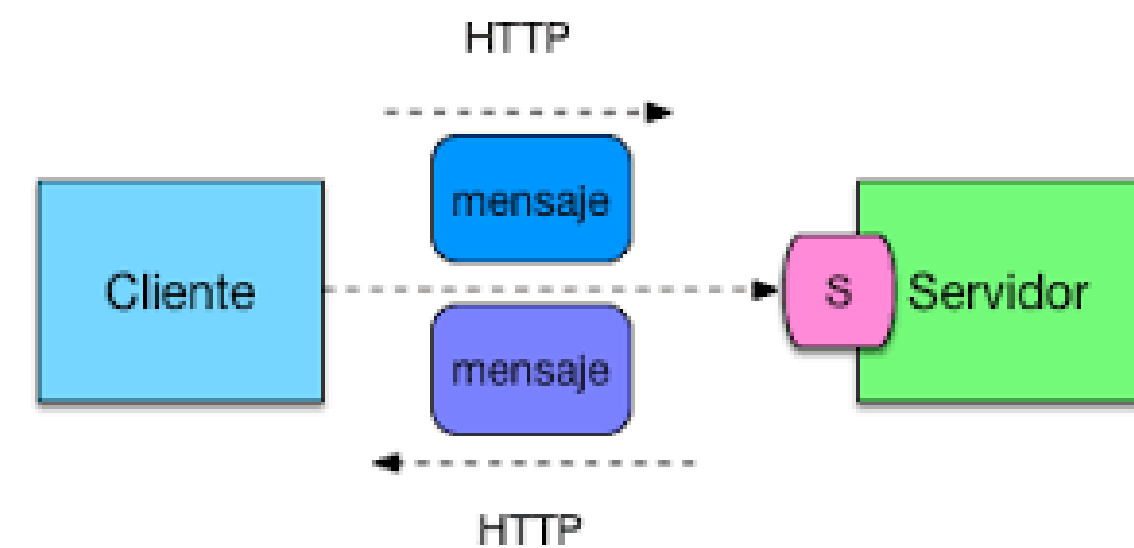


Patrones de diseño y arquitectónicos que implementa:

MVC



RESTful



Situaciones recomendadas para su uso

- APIs RESTful
- Prototipos rápidos
- Aplicaciones web



Semejanzas y diferencias con otros frameworks:

- Arquitectura Similar: Frameworks como Django y Laravel siguen el patrón MVC, separando la lógica de negocio, la interfaz de usuario y el control de flujo.
- Lenguaje de programación: Ruby on Rails utiliza Ruby, conocido por su legibilidad, mientras que Django utiliza Python y Laravel utiliza PHP.
- Ecosistema: Cada framework tiene su propio conjunto de bibliotecas y herramientas. Ruby on Rails utiliza "gems" para extender funcionalidades, mientras que Django utiliza "packages" y Laravel utiliza "packages" o "composer packages".

CONCLUSIONES

- Vue.js y Ruby on Rails están diseñados para optimizar el desarrollo de aplicaciones, ofreciendo estructuras organizadas y componentes reutilizables que mejoran la eficiencia.
- Mientras Vue.js se enfoca en la construcción de interfaces dinámicas y reactivas, Ruby on Rails cubre todo el proceso de desarrollo backend con una arquitectura MVC bien definida.
- Ambos frameworks aplican patrones de diseño y arquitecturas que favorecen la escalabilidad y el mantenimiento, permitiendo desarrollar aplicaciones robustas y eficientes.

Muchas
GRACIAS