**Resumen Detallado del PDF: Componentes y Vue CLI**

**1. Componentes**

1.1. **Definición de Componente:**

* Un componente es una parte reutilizable de la interfaz de usuario (UI) que encapsula HTML, CSS y JavaScript.
* Se utiliza para separar la funcionalidad y estructurar la lógica del proyecto.
* Características principales:
  + Reusabilidad.
  + Capacidad de combinarse con otros componentes.
  + Independencia (cada componente tiene su propia estructura y funcionalidad).

1.2. **Props:**

* Permiten pasar parámetros de un componente padre a uno hijo.
* Se definen como atributos en la etiqueta del componente y se configuran con la directiva v-bind.
* Ejemplo:
* <mi-componente attr1="25" :attr2="var1" />

**2. Single File Components (SFC)**

2.1. **Ventajas:**

* Organización del código en un único archivo .vue con secciones separadas:
  + <template>: HTML del componente.
  + <script>: Lógica en JavaScript.
  + <style>: Estilos CSS.

2.2. **Ciclo de Vida del Componente:**

* Estados y hooks disponibles en Vue.js:
  + **beforeCreate**: Antes de inicializar.
  + **created**: Después de inicializar la instancia.
  + **beforeMount**: Antes de montar en el DOM.
  + **mounted**: Después de montar en el DOM.
  + **beforeUpdate**: Antes de actualizar el DOM.
  + **updated**: Después de actualizar el DOM.
  + **beforeUnmount**: Antes de desmontar.
  + **unmounted**: Después de desmontar.

2.3. **Estructura del Proyecto:**

* Cada componente debe tener su propio archivo .vue.
* Componentes principales como App.vue contienen la estructura base.

**3. Vue CLI**

3.1. **Definición:**

* Herramienta de línea de comandos para Vue.js que automatiza la creación y configuración de proyectos.
* Incluye herramientas como Webpack, Babel y Uglify para optimizar y empaquetar el código.

3.2. **Instalación y Uso:**

* Requiere **Node.js** y se instala globalmente con npm install -g @vue/cli.
* Comando para crear proyectos: vue create nombre\_proyecto.
* Ofrece opciones de configuración manual o predefinida.

3.3. **Contenido Generado por Vue CLI:**

* Carpeta public: Contiene index.html.
* Carpeta src:
  + main.js: Archivo principal de configuración.
  + App.vue: Componente raíz.
  + assets: Archivos estáticos.
* Archivos de configuración como package.json y .gitignore.

3.4. **Instalación de Plugins:**

* Plugins como bootstrap-vue se añaden con vue add nombrePlugin.

**4. Comunicación entre Componentes**

4.1. **Custom Events:**

* Un componente hijo puede emitir eventos personalizados al padre usando $emit.
* El padre captura estos eventos mediante atributos en la etiqueta del componente hijo.

4.2. **Otras Opciones de Comunicación:**

* Uso de variables globales (no recomendado).
* Librerías como **Mitt** para Vue 3 o el patrón de estado global con **Pinia**.

**5. Despliegue en Producción**

* Compilar el proyecto con npm run build, generando una carpeta dist lista para producción.
* Herramienta recomendada para despliegue: **Netlify**.
* Opciones:
  + Subir la carpeta dist manualmente.
  + Vincular el proyecto a un repositorio de GitHub.

**Ejercicios Propuestos:** El PDF incluye diversos ejercicios prácticos enfocados en:

* Crear proyectos con Vue CLI.
* Configurar comunicación entre componentes.
* Usar props y custom events.
* Personalizar estilos y desplegar proyectos en producción.

¿Quieres un resumen aún más condensado o detalles sobre algún tema específico?