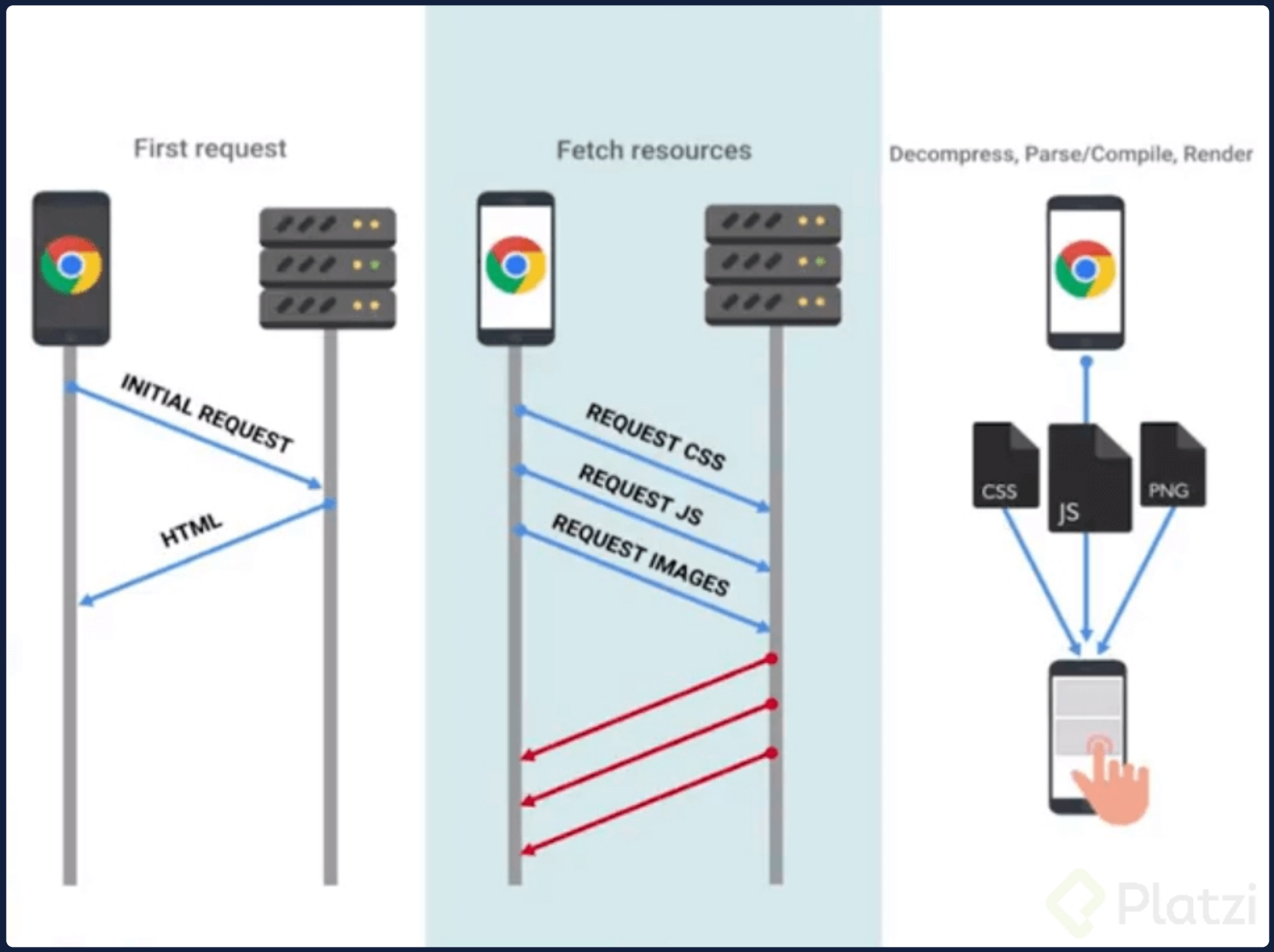
LazyLoading y CodeSplitting 10/25

RECURSOS APUNTES

Dos conceptos claves que tienes que conocer como desarrollador front-end son el *Lazy Loading* y el *CodeSplitting*. En conjunto, buscan **optimizar los tiempos de carga y rendimiento de tu aplicación**. Veamos cómo lo hacen.

Cómo es el proceso de una solicitud

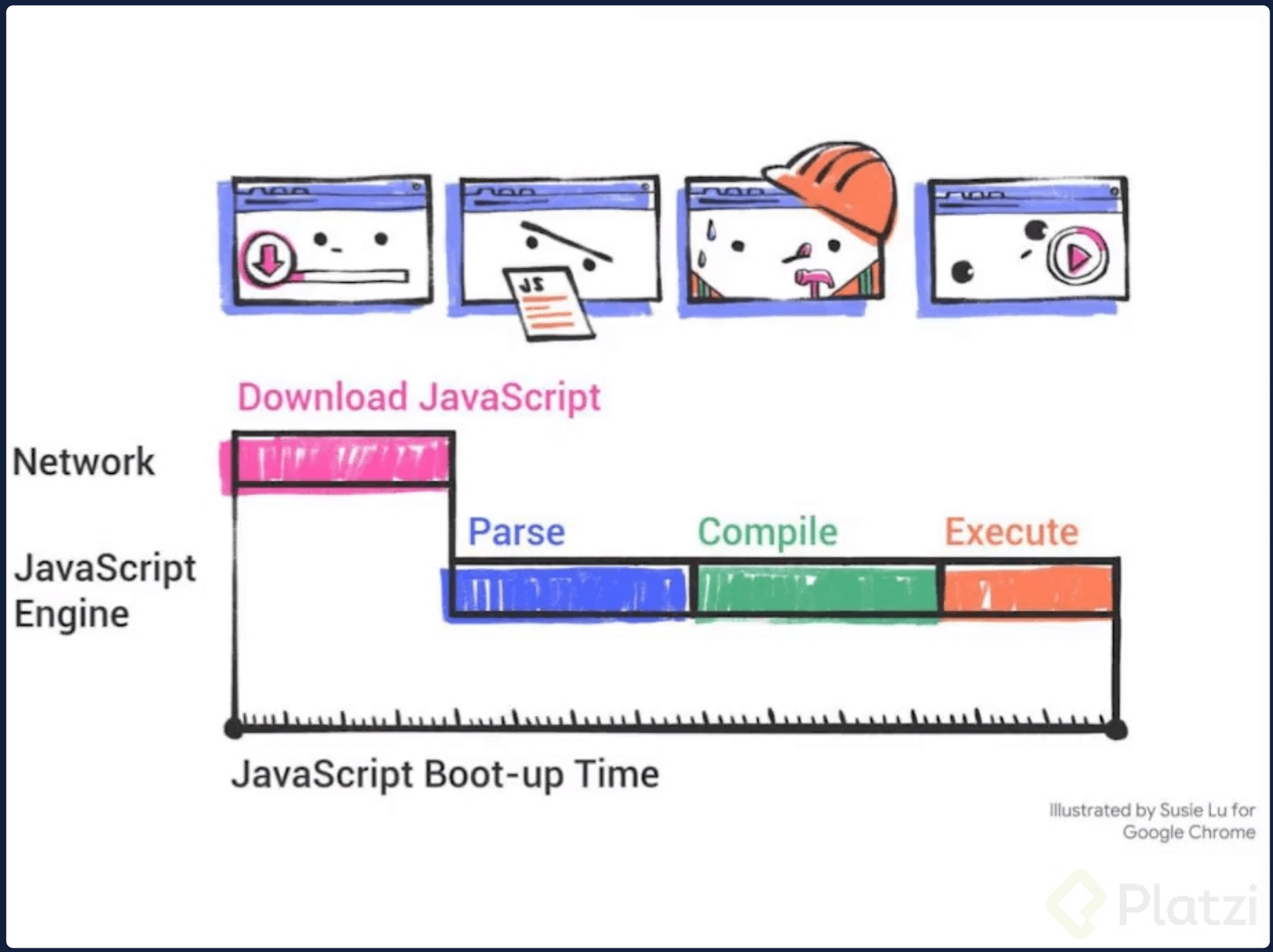
Comencemos entendiendo cómo funcionan las solicitudes de una página web a un servidor HTTP.



Paso 1: Cuando se hace la primera solicitud de un recurso a un servidor, este devuelve el archivo base de la página web (comúnmente llamado `index.html`).

Paso 2: El navegador lee este archivo, y realiza una nueva solicitud por cada recurso que encuentra. Por cada imagen, por cada hoja de estilos CSS y por cada archivo Javascript.

Paso 3: Cada archivo Javascript que es descargado, conlleva un proceso de cuatro pasos.



El archivo JS primero se descarga, se parsea, se compila y finalmente se ejecuta su código.

La descarga del archivo es el paso más importante y que más tiempo suele necesitar, ya que depende del ancho de banda de la red y del peso del mismo.

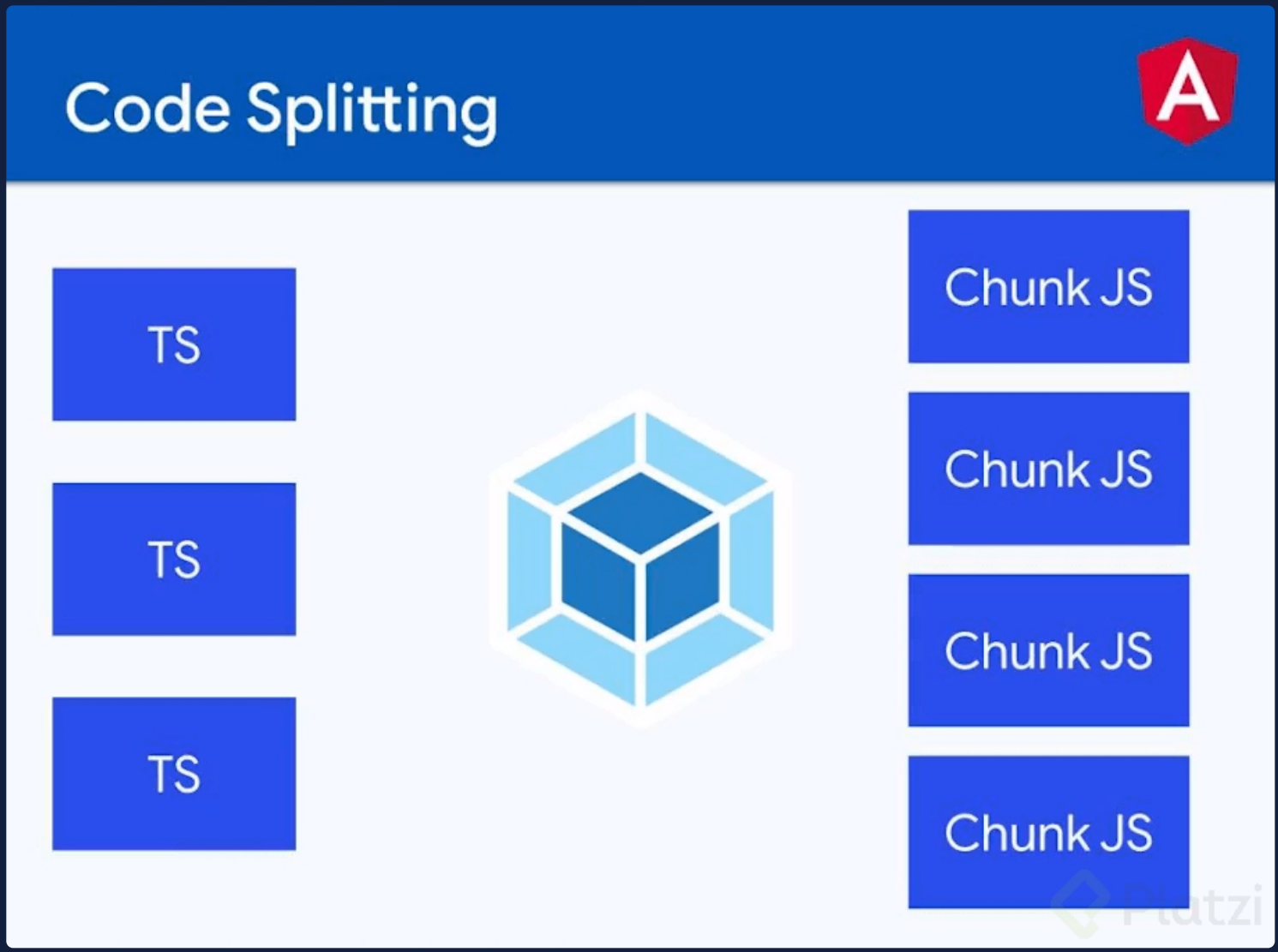
Optimización de tiempo y recursos

Para reducir el peso de las aplicaciones, aparece el concepto de *Lazy Loading* y el *CodeSplitting* que plantean la división del código fuente Javascript en pequeños módulos y solo cargar aquellos que el usuario necesite, cuando realmente los necesite.

Angular utiliza Webpack para compilar y construir la aplicación. Por defecto, genera un solo módulo denominado `main.js` . Un solo archivo con TODO el código Javascript de tu aplicación.

Esto podría ser perjudicial para el rendimiento de tu aplicación, ya que a medida que crezca, más pesada será.

Para su solución, activando la modularización de tu aplicación con *Lazy Loading*, Webpack dividirá la misma en N cantidad de Chunks (particiones) para una carga más rápida de los archivos Javascript.



Por ejemplo, si tu aplicación posee una página de Home, una página Catálogo y una página de Contacto, Webpack dividirá el build de la aplicación en tres partes para enviarle al usuario solo las páginas que necesita en ese momento.

Contribución creada por: *Kevin Fiorentino.*