

# **HIPÓTESIS E INTERPRETACIÓN DE GRÁFICAS**

**VALENTINA SILVA GARRIDO**

**FICHA: 2694667**

**PRESENTADO A INSTR. JESUS ARIEL GONZALEZ BONILLA**

**SERVICIO NACIONAL DE APRENDIZAJE (SENA)**

**TECNÓLOGO EN ANÁLISIS Y DESARROLLO DE SOFTWARE**

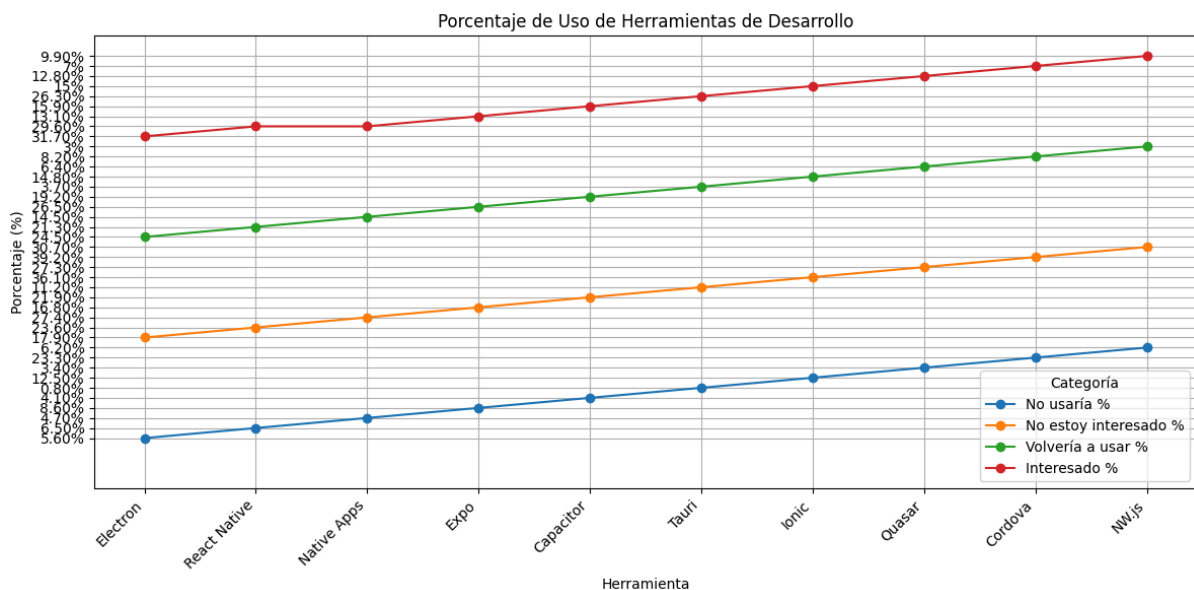
**NEIVA - HUILA**

**2024**

## Gráfica 1

### Hipótesis:

- Las herramientas con mayor porcentaje de personas interesadas o que volverían a usar son aquellas que probablemente se perciben como más modernas, útiles o con mayor soporte en la comunidad.
- Las herramientas con mayor porcentaje en la categoría "No usaría" posiblemente se perciben como obsoletas, difíciles de usar o con menos ventajas en comparación con las alternativas.
- El interés en las aplicaciones nativas podría sugerir una preferencia por el desempeño y las capacidades propias de cada plataforma, especialmente en sectores como el móvil.
- Herramientas como React Native y Tauri podrían estar ganando popularidad debido a su enfoque en desarrollo multiplataforma eficiente, mientras que las más tradicionales (como Cordova) podrían estar perdiendo interés.
- Herramientas como Ionic o Capacitor, que presentan distribución equilibrada en las diferentes categorías, podrían estar en transición o apelando a un nicho de desarrolladores específico.



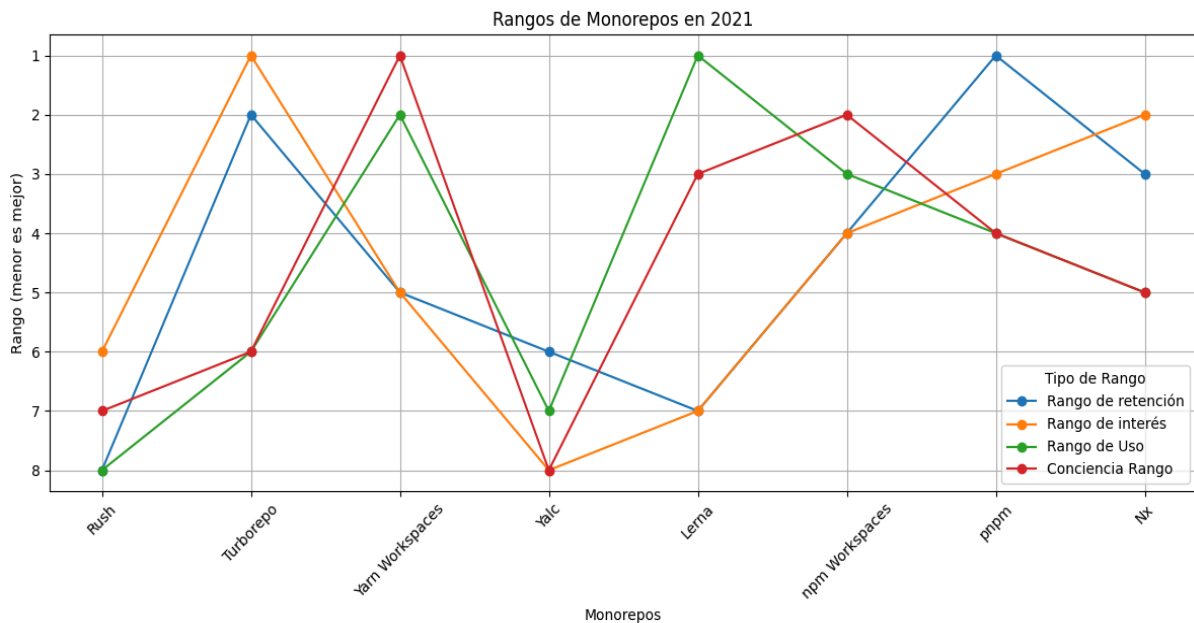
Se observa un interés creciente en las herramientas listadas de izquierda a derecha, con valores más altos para "Interesado" y "Volvería a usar" en comparación con las otras dos categorías.

Herramientas como Native Apps, Tauri, Ionic y NW.js parecen tener una mayor aceptación en comparación con las primeras como Electron y React Native, según la pendiente de las líneas.

## Gráfica 2

### Hipótesis:

La herramienta Nx parece ser la más destacada en términos de popularidad general en 2021, considerando que tiene los rangos más bajos (mejor desempeño) en retención, interés y uso. Las herramientas menos reconocidas, como Rush o Yalc, podrían tener un alcance más limitado debido a su posición más alta en varias categorías.



Nx sobresale como la mejor herramienta en el gráfico, logrando el mejor rango (más bajo) en las métricas clave, lo que sugiere una alta aceptación, retención, interés y uso entre los desarrolladores.

Turborepo y Yarn Workspaces también tienen un desempeño competitivo en términos de retención e interés, aunque no alcanzan el nivel de Nx.

Rush y Yalc están en los peores rangos en todas las categorías, lo que podría indicar menor uso y conocimiento por parte de la comunidad de desarrolladores.

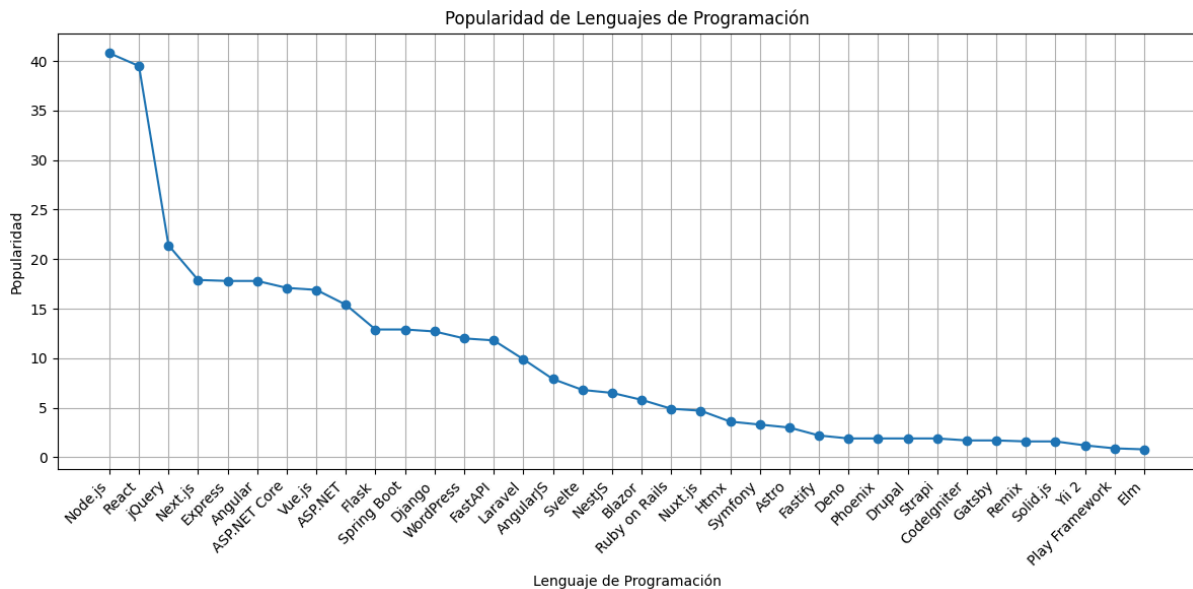
Lerna y npm Workspaces tienen un desempeño intermedio, lo que indica que están presentes en la comunidad, pero no generan tanto interés ni uso como las herramientas líderes.

En conclusión, Nx es una herramienta ampliamente utilizada y reconocida en 2021, mientras que otras herramientas tienen una popularidad más específica o limitada. Esto puede deberse a características técnicas, facilidad de uso o integración con otras tecnologías.

### Gráfica 3

#### Hipótesis:

La popularidad de los lenguajes y frameworks de programación se encuentra altamente influenciada por su facilidad de uso, amplia comunidad, y adaptabilidad en múltiples casos de uso, lo que lleva a que herramientas como Node.js, React y jQuery dominen en comparación con tecnologías más especializadas como Elm o Play Framework.

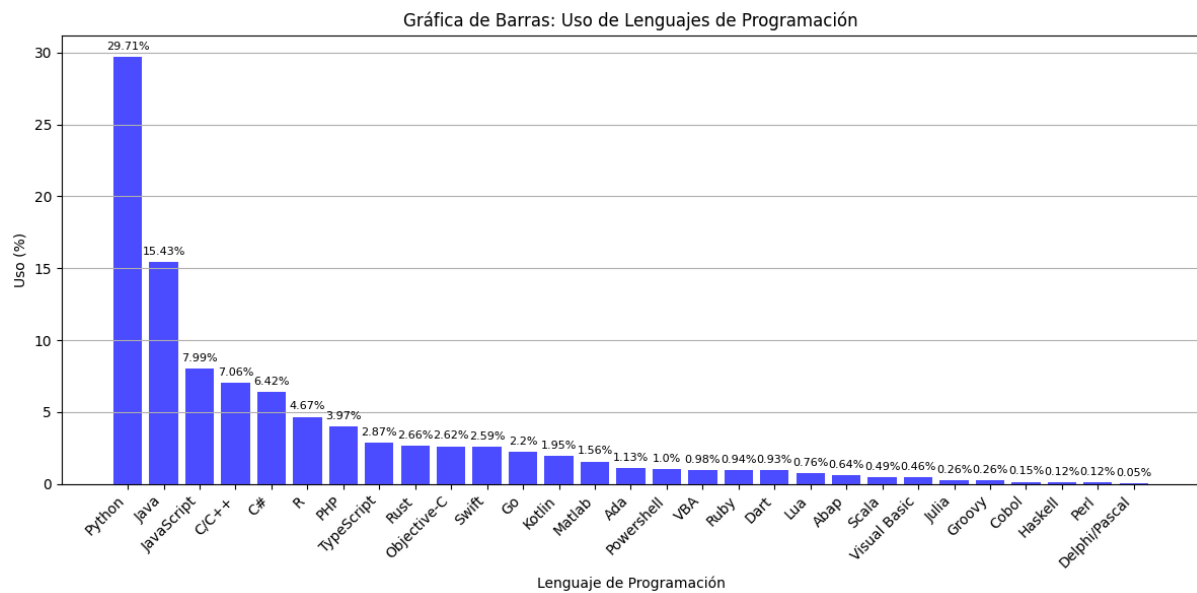


La gráfica muestra que la popularidad de los lenguajes y frameworks de programación está dominada por tecnologías ampliamente utilizadas en el desarrollo web, como Node.js, React y jQuery, debido a su versatilidad, soporte comunitario y adopción generalizada. Sin embargo, existe una caída pronunciada en la popularidad después de los primeros cinco, seguida por un largo tramo de herramientas menos conocidas como Deno o Elm, que, aunque útiles en nichos específicos, no alcanzan el nivel de adopción de las líderes. Esto sugiere que la facilidad de uso y la aplicabilidad general son factores clave en la preferencia de los desarrolladores.

### Gráfica 4

#### Hipótesis:

Python lidera el uso de lenguajes de programación debido a su versatilidad y simplicidad, lo que lo hace ideal tanto para principiantes como para profesionales en campos diversos como el desarrollo web, análisis de datos y aprendizaje automático.

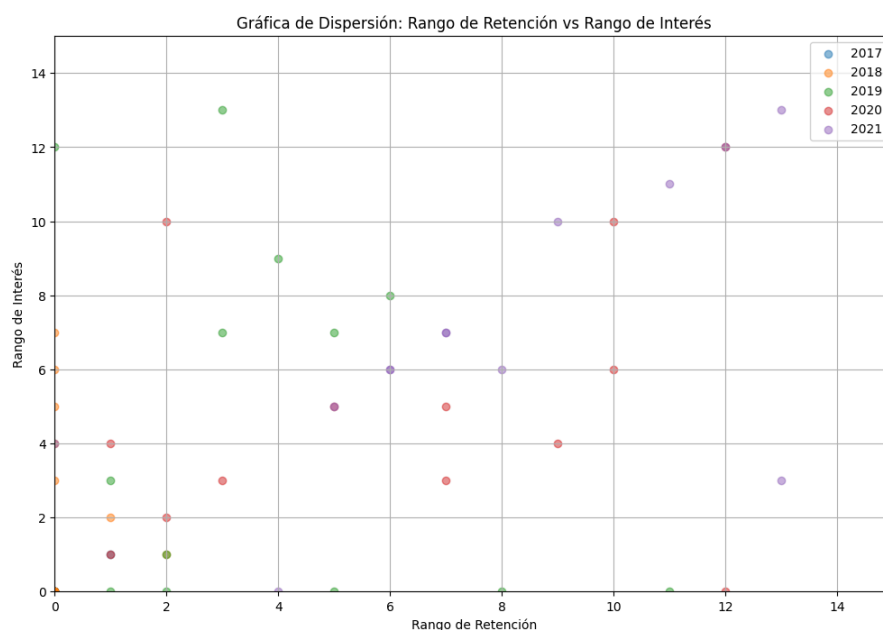


Python es el lenguaje más utilizado, con un uso significativamente mayor (29.71%) que el resto, seguido por Java (15.43%) y JavaScript (7.99%). Esto sugiere que los lenguajes dinámicos y flexibles dominan el panorama, mientras que lenguajes especializados como Delphi/Pascal tienen un uso marginal (0.05%). Esto podría reflejar las necesidades actuales del mercado tecnológico, que favorecen herramientas eficientes y de rápido desarrollo frente a lenguajes más tradicionales.

## Gráfica 5

### Hipótesis:

Existe una débil correlación positiva entre el rango de retención y el rango de interés a lo largo de los años analizados (2017-2021).

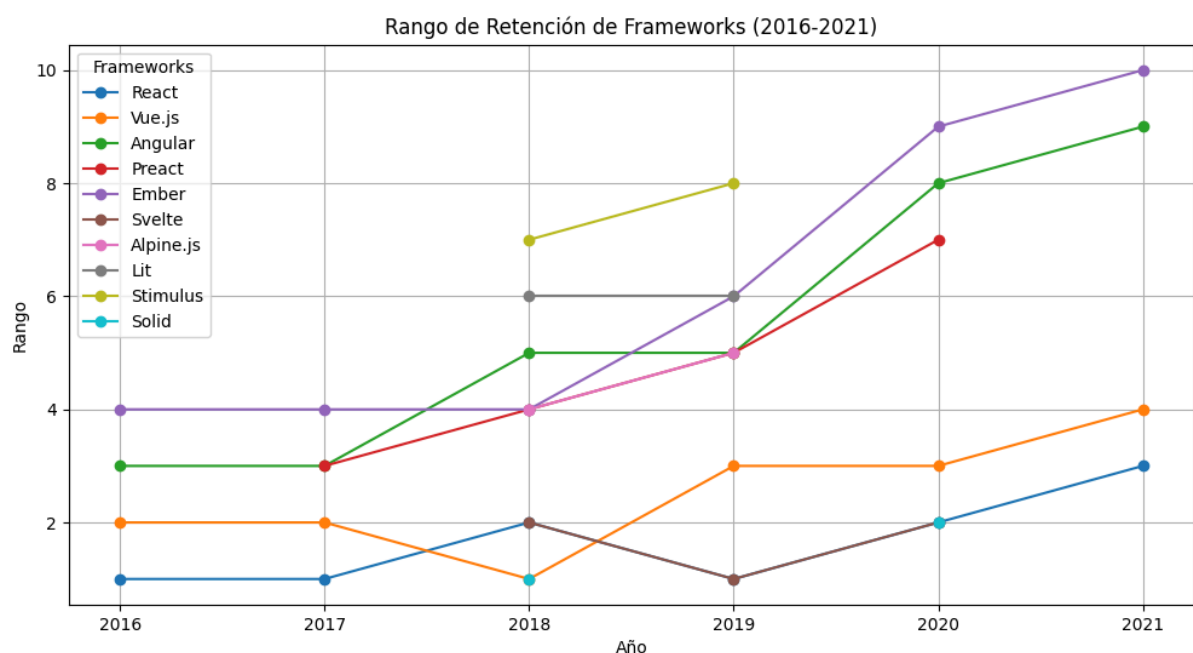


La gráfica sugiere que, en general, a medida que aumenta el rango de retención (lo cual podría interpretarse como un mayor tiempo que los usuarios permanecen en una plataforma o servicio), también tiende a aumentar ligeramente el rango de interés (lo que podría indicar un mayor engagement o nivel de satisfacción). Sin embargo, esta relación no es muy fuerte y presenta una gran dispersión de los datos, lo que indica que otros factores podrían estar influyendo en el nivel de interés de los usuarios más allá del tiempo de retención.

## Gráfica 6

### Hipótesis:

Existe una tendencia creciente en la popularidad de los frameworks React, Vue.js y Angular, con una disminución relativa en la popularidad de frameworks como Ember y Alpine.js.

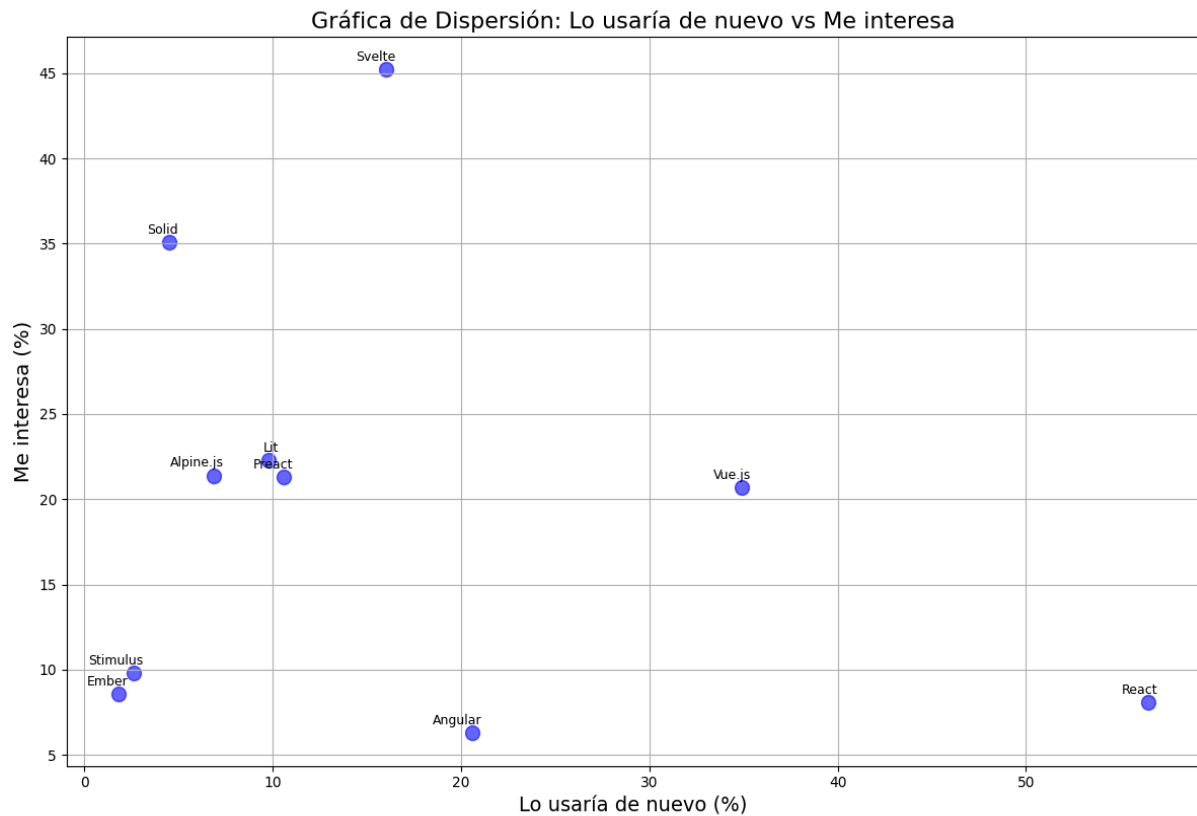


La gráfica muestra la evolución del rango de retención de varios frameworks JavaScript a lo largo de los años 2016-2021. Se observa un crecimiento constante en el rango de React, Vue.js y Angular, lo que sugiere una adopción cada vez mayor por parte de los desarrolladores. Por otro lado, frameworks como Ember y Alpine.js muestran una tendencia a la baja, indicando una posible disminución en su uso. Esta tendencia podría estar influenciada por diversos factores, como la facilidad de uso, la comunidad, el ecosistema de herramientas, y la capacidad de adaptarse a las nuevas tendencias del desarrollo web. Sin embargo, es importante notar que la popularidad de los frameworks es dinámica y puede cambiar con el tiempo, por lo que se requieren análisis más detallados para confirmar estas tendencias a largo plazo.

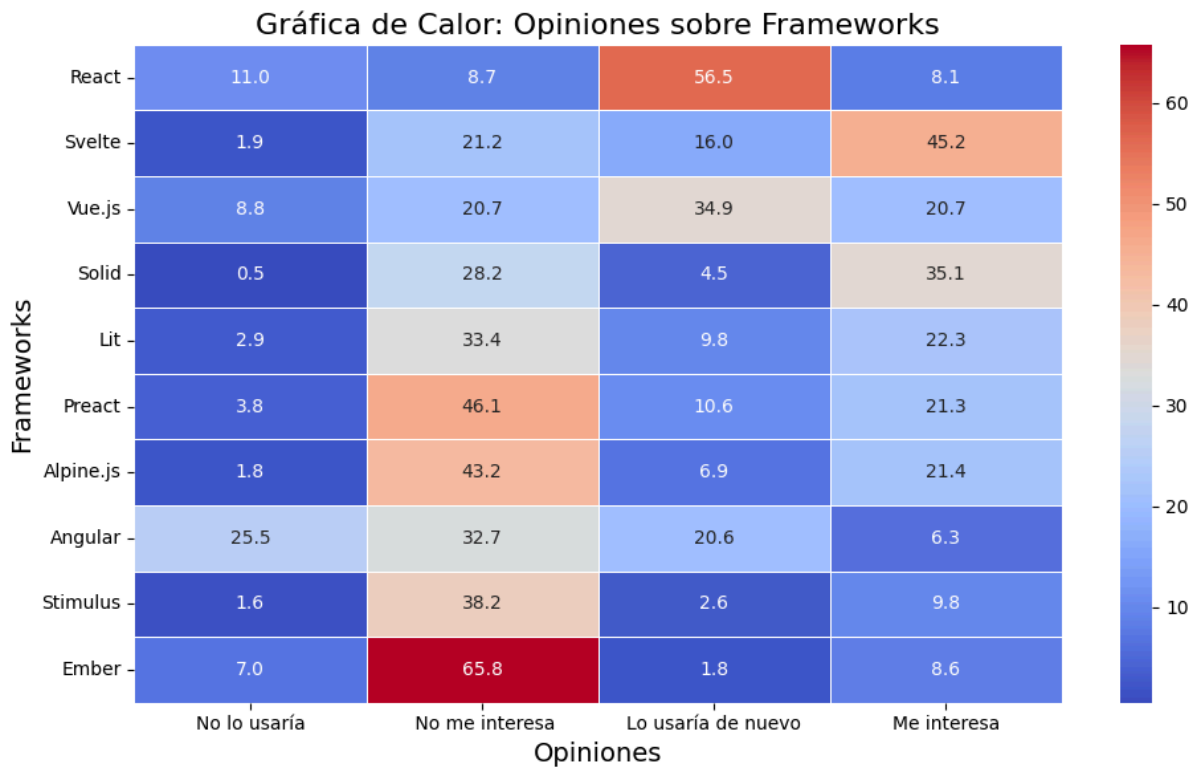
## Gráfica 7

### Hipótesis:

Existe una correlación positiva moderada entre la disposición de volver a utilizar un framework y el nivel de interés inicial en el mismo.



La gráfica de dispersión sugiere que, en general, los desarrolladores que expresan un mayor interés inicial en un framework específico también muestran una mayor probabilidad de volver a utilizarlo en el futuro. Esto indica que el interés inicial puede ser un buen predictor de la adopción a largo plazo de un framework. Sin embargo, la dispersión de los datos sugiere que otros factores, además del interés inicial, influyen en la decisión de volver a utilizar un framework, como la facilidad de uso, la documentación, la comunidad y la integración con otras herramientas. Frameworks como Svelte y Solid muestran un alto interés y una alta probabilidad de reutilización, lo que sugiere que estos frameworks están satisfaciendo las necesidades de los desarrolladores y podrían estar ganando popularidad. Por otro lado, frameworks como Ember y Stimulus muestran un interés relativamente bajo y una probabilidad de reutilización aún menor, lo que podría indicar que estos frameworks enfrentan desafíos en términos de adopción y crecimiento.

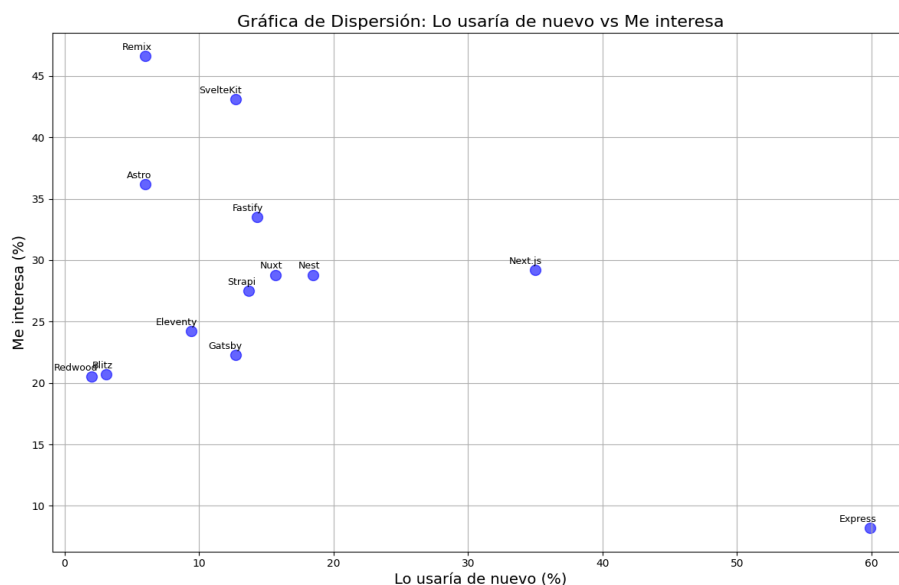


El mapa de calor nos ofrece una visión general de las opiniones de los desarrolladores sobre diferentes frameworks JavaScript. Cada celda representa el porcentaje de desarrolladores que eligieron una determinada opción (no lo usaría, no me interesa, lo usaría de nuevo, me interesa) para un framework específico. Los colores más cálidos (rojos) indican una mayor proporción de respuestas positivas, mientras que los colores más fríos (azules) indican una mayor proporción de respuestas negativas.

## Gráfica 8

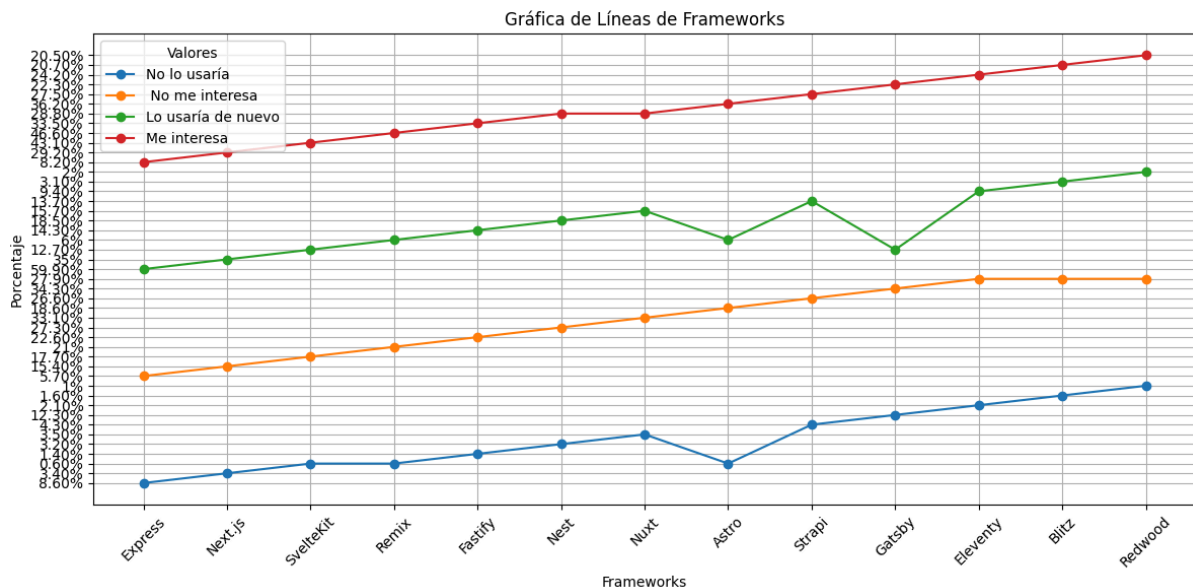
### Hipótesis:

Existe una correlación positiva moderada entre el interés general en una tecnología (representado por el eje "Me interesa") y la disposición a utilizarla nuevamente (representado por el eje "Lo usaría de nuevo").





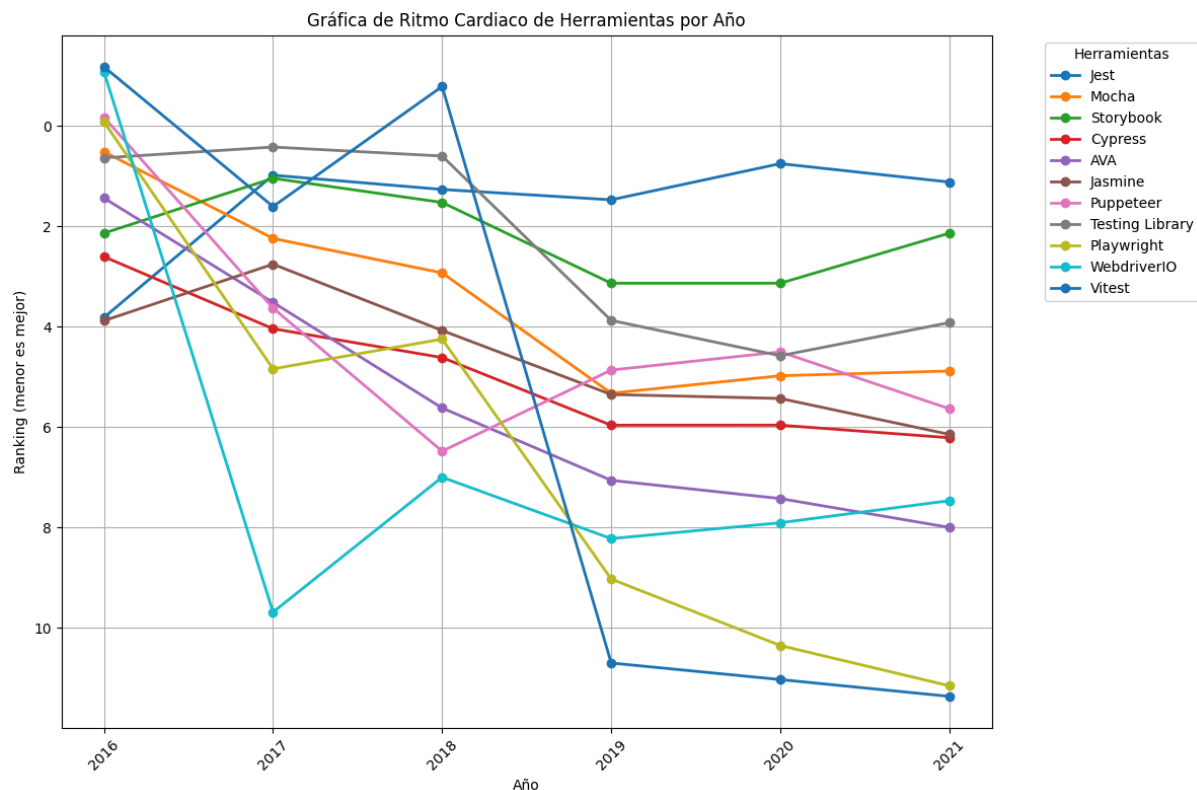
La gráfica sugiere que, en general, las tecnologías que generan un mayor interés entre los usuarios también tienen una mayor probabilidad de ser utilizadas nuevamente. Esto indica que el interés inicial en una tecnología puede ser un buen predictor de su adopción a largo plazo. Sin embargo, es importante notar que la relación no es perfecta y hay algunos puntos que se desvían de la tendencia general. Esto podría sugerir que otros factores, además del interés inicial, influyen en la decisión de volver a utilizar una tecnología.



La gráfica de líneas nos muestra una comparación de la percepción de los usuarios hacia diferentes frameworks. Podemos observar que existe una variabilidad significativa en la valoración de cada framework, lo que indica que las preferencias de los desarrolladores son diversas. En general, los frameworks tienden a obtener mejores puntuaciones en cuanto a la disposición de los usuarios a volver a utilizarlos, sugiriendo una satisfacción general con estas herramientas. Sin embargo, hay una considerable disparidad entre los frameworks en términos de interés y disposición a usarlos, lo que podría estar relacionado con factores como la complejidad, popularidad, comunidad y características específicas de cada uno.

## Gráfica 9

**Hipótesis:** La popularidad y uso de herramientas de testing ha evolucionado significativamente a lo largo de los años, con un claro predominio de ciertas herramientas en determinados períodos. Esta evolución está influenciada por factores como la madurez de las herramientas, la aparición de nuevas tecnologías y las tendencias en desarrollo de software.

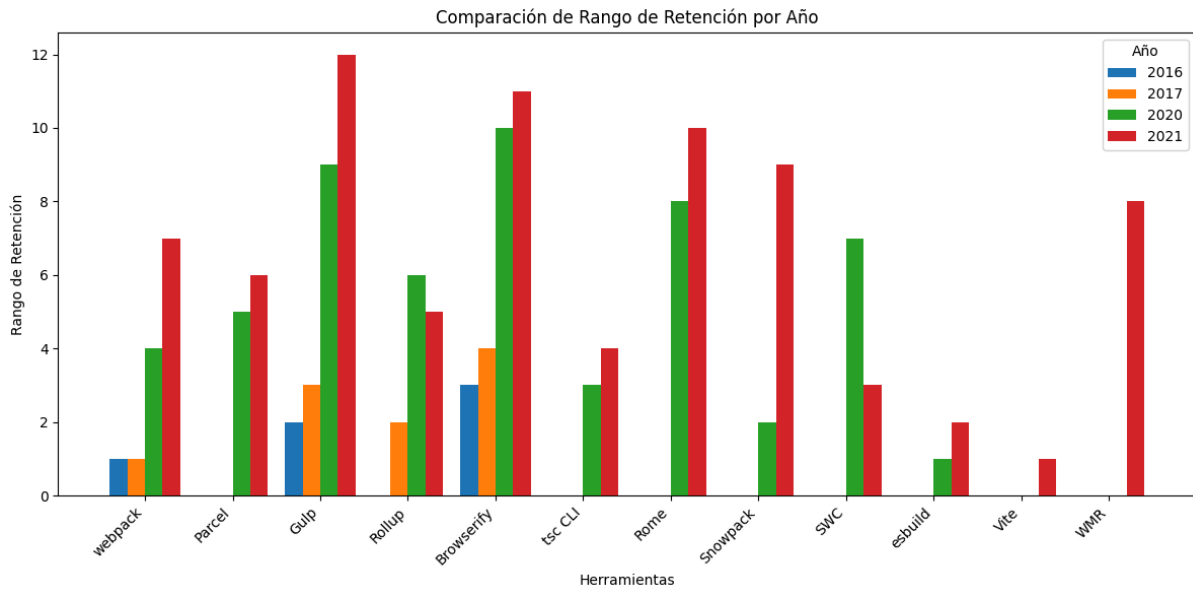


La gráfica "Ritmo Cardíaco de Herramientas por Año" presenta una dinámica interesante en el mundo del testing de software. Observamos que herramientas como Jest y Mocha han mantenido una posición de liderazgo constante a lo largo de los años, sugiriendo una sólida adopción y confianza en la comunidad de desarrolladores. Sin embargo, es notable el ascenso de herramientas más recientes como Cypress y Playwright, que han ganado terreno rápidamente, posiblemente debido a características innovadoras o una mejor adaptación a las nuevas metodologías de desarrollo. Por otro lado, herramientas como Jasmine y AVA parecen haber experimentado una disminución en su popularidad, lo que podría indicar una madurez menor o una menor inversión en su desarrollo. En general, la gráfica sugiere un panorama competitivo y en constante evolución en el ámbito de las herramientas de testing, donde la elección de una herramienta en particular depende de las necesidades específicas de cada proyecto y equipo de desarrollo.

## Gráfica 10

### Hipótesis:

La popularidad y uso de herramientas de empaquetado de módulos ha evolucionado significativamente a lo largo de los años, con un claro predominio de ciertas herramientas en determinados períodos. Esta evolución está influenciada por factores como la madurez de las herramientas, la aparición de nuevas tecnologías y las tendencias en desarrollo de aplicaciones web.



La gráfica "Comparación de Rango de Retención por Año" muestra una dinámica interesante en el mundo del empaquetado de módulos. Observamos que herramientas como Gulp y webpack han tenido una presencia constante a lo largo de los años, sugiriendo una sólida adopción y confianza en la comunidad de desarrolladores. Sin embargo, es notable el ascenso de herramientas más recientes como esbuild y Vite, que han ganado terreno rápidamente, posiblemente debido a características innovadoras como su velocidad de compilación o una mejor integración con otros elementos del ecosistema de desarrollo. Por otro lado, herramientas como Parcel parecen haber experimentado una disminución en su popularidad, lo que podría indicar una madurez menor o una menor inversión en su desarrollo. En general, la gráfica sugiere un panorama competitivo y en constante evolución en el ámbito de las herramientas de empaquetado de módulos, donde la elección de una herramienta en particular depende de las necesidades específicas de cada proyecto y equipo de desarrollo.