

Aprovechamiento de las características de las Aplicaciones Web Progresivas en las Redes Sociales

Rocío Rodríguez, Pablo Vera, Claudia Alderete, Mariano Dogliotti

Centro de Altos Estudios en Tecnología Informática (CAETI)
Facultad de Tecnología Informática
Universidad Abierta Interamericana (UAI)
Avenida Montes de Oca 745, Ciudad Autónoma de Buenos Aires, Argentina
{RocioAndrea.Rodriguez, PabloMartin.Vera, ClaudiaGabriela.Alderete,
MarianoGaston.Dogliotti }@uai.edu.ar

Resumen. Gran parte de la población mundial ha encontrado en las redes sociales un lugar para comunicarse y difundir sus pensamientos e ideas. El acceso a las redes sociales se produce mayormente desde dispositivos móviles, los cuales poseen distintas características, sistemas operativos y capacidades. El acceso debe ser asegurado desde distintos dispositivos móviles así como desde computadoras. El desarrollar una aplicación particular para cada sistema operativo (considerando distintos versionados) y contemplando las diversas características de los mismos, resulta cada vez más complejo. Las PWA (aplicaciones web progresivas) permiten simplificar y unificar el desarrollo, con la portabilidad propia de la web, agregando características propias de las aplicaciones nativas, donde para el usuario final es indistinto si se trata de una aplicación PWA ó nativa. Este trabajo presenta un relevamiento de las principales redes sociales, analizando cuales de ellas están construidas mediante el principio de PWA y además analiza ciertas características para detectar si realmente están bien configuradas y cumplen con los lineamientos básicos y buenas prácticas de las PWA.

Palabras Clave: PWA, Redes Sociales, Service Worker, Manifiesto

1 Introducción

Las personas somos seres sociables y eso se ha reflejado a través del tiempo en el uso de distintos medios sociales en internet, “la extraordinaria capacidad de comunicación y de poner en contacto a las personas que tienen las redes ha provocado que un gran número de personas las esté utilizando con fines muy distintos” [1]. Pero la necesidad de estar conectados cobró mayor significancia en tiempos de pandemia. Y esto se vio reflejado en las redes sociales. “... Los medios sociales se han convertido en una parte indispensable de la vida cotidiana para todas las personas alrededor del mundo, el 2020 un año en el que el mundo cayó en un bloqueo, los usuarios de redes sociales crecieron a una tasa más grande que en 3 años...” [2]. "1,3 millones de nuevos usuarios se unieron a las redes sociales cada día durante 2020: 15 nuevos usuarios

cada segundo"[3]. "Esto significa que, por primera vez, más de la mitad de la población mundial ahora usa las redes sociales ... [4].

"Al avanzar la tecnología móvil poco a poco se ha ido convirtiendo en el centro de la actividad online. Hoy en día, desde prácticamente cualquier dispositivo móvil podemos acceder a Internet, ya sea bien a través de navegadores o aplicaciones instaladas" [5]. Si se toma como ejemplo la red social Facebook en Argentina, el 97,5% de los accesos se realiza desde algún dispositivo móvil, mientras que un 2,5% lo hace desde notebook o computadora de escritorio, evaluando por cada usuario esos accesos se puede determinar que el 73,5% ingresa únicamente desde un dispositivo móvil (estadísticas publicadas en [6], en Enero del 2021). Esto pone en evidencia la necesidad de contar con soluciones que sean aptas para un sinfín de dispositivos móviles con características diferentes, así como en notebook o computadoras de escritorio. En este aspecto las aplicaciones web progresivas (PWA) pueden ofrecer una buena solución.

Este artículo se encuentra estructurado de la siguiente manera: en la sección 2 se definen las Aplicaciones Web Progresivas (PWA) y se presentan sus componentes; en la sección 3 se presentan las características a analizar en una PWA; en la sección 4 se listan las redes sociales que serán consideradas para someterlas a dicho análisis, en la sección 5 se presentan los resultados obtenidos y finalmente en la sección 6 las conclusiones.

2 PWA

Las aplicaciones web progresivas (PWA) son una evolución de las aplicaciones web, que consideran las bases de las aplicaciones web adaptativas [7], incorporando la apariencia junto con algunas funcionalidades que antes eran exclusivas de las aplicaciones nativas. "Las aplicaciones web progresivas son una evolución natural de las aplicaciones web que difuminan la barrera entre la web y las aplicaciones, pudiendo realizar tareas que generalmente solo las aplicaciones nativas podían llevar a cabo. Algunos ejemplos son las notificaciones, el funcionamiento sin conexión a Internet o la posibilidad de probar una versión más ligera antes de bajarte una aplicación nativa de verdad" [8]. "La PWA se basa en los conceptos de una sola aplicación para todas las plataformas al igual que el enfoque híbrido. Sin embargo, posee distintas capacidades, como carga instantánea, notificaciones push incluso en estado fuera de línea" [9].

Los componentes principales de una PWA son: (1) Archivo de Manifiesto, (2) Service Worker, (3) Almacenamiento Local, (4) Notificaciones

Toda PWA tendrá como punto de inicio un archivo de manifiesto en el que se definen las configuraciones básicas de la aplicación y será imprescindible contar con un service worker. "Un service worker es un proceso que el navegador web ejecuta en segundo plano y está asociado a un sitio web particular. Este proceso se programa en javascript y permite capturar las peticiones que el sitio web hace a la red e interceptarlas actuando como un proxy local. El capturar esas peticiones permite que el service worker responda en lugar de la red, haciendo posible que el navegador no salga a la red sino que se le devuelvan los datos localmente" [10]. Es decir, brinda la

posibilidad de recuperar los datos sin necesidad de la red y esto se logra gracias a métodos de almacenamiento local. Por último, las PWA incorporan la capacidad de mostrar notificaciones al usuario, aun cuando el navegador no está abierto, trabajando de forma similar a una aplicación nativa.

3 Análisis

Se toman en consideración diversas características sobre un conjunto de sitios de redes sociales (los cuales se listan en la sección siguiente), el análisis comenzará por la evaluación del archivo de manifiesto (la ausencia de este archivo implica que el sitio analizado no es PWA). Dentro del archivo de manifiesto se visualizan algunas propiedades básicas si están indicadas. Luego se examina el tamaño medido en Bytes que ocupa la solución, primeramente, en cuanto a consumo de memoria, del mismo modo se analiza el serviceworker (toda aplicación PWA posee un service worker) y además se analiza si de forma adicional emplean otros mecanismos de almacenamiento como bases de datos o almacenamiento interno en el dispositivo.

A continuación, se detallan las características consideradas.

3.1 Archivo de Manifiesto y Service Worker

Como fue especificado anteriormente para que una aplicación sea PWA debe tener estos dos elementos básicos (archivos de manifiesto y service worker).

El archivo de manifiesto es un archivo JSON que permite configurar ciertas características de la aplicación. En dicho archivo se analizará la definición de las siguientes propiedades:

- **Presentación:** Se analiza que estén definidos: (1) Los colores tanto de Tema como de Fondo, (2) El modo de visualización, definido mediante la propiedad Display.
- **Iconos:** De la aplicación en distintos tamaños, con estilo enmascarable. Debido a que la PWA podrá instalarse en distintos entornos cada uno de ellos tendrá una forma diferente de mostrar los íconos (cuadrados, circulares, cuadrado con borde redondeados...), es por eso que los íconos enmascarables permiten tomar un área desde el centro que será siempre visualizable en cualquier entorno y estos serán mostrados dentro del entorno de forma tal que sean visualizados igual que una aplicación nativa. En la figura 1, se muestra uno de los íconos creados para la PWA de Youtube y como sería visto al enmascararlos en un formato circular.



Fig. 1. A la izquierda Icono de Youtube y a la derecha enmascarado en una forma circular

- Shortcuts: Se pueden definir atajos para el acceso a determinadas funcionalidades (funcionarán cuando la aplicación es instalada)
- Screenshot: Capturas de pantallas que pueden ser incorporadas para tenerlas por defecto

Existen PWA que no tienen precisiones en los parámetros de su archivo manifiesto y es por ello por lo que resulta de interés relevar la completitud de estos.

En cuanto al service worker mediante la herramienta developer tools del navegador Google Chrome es posible visualizar si un sitio determinado tiene un service worker activo, su versión y estado. A modo de ejemplo la figura 2 muestra con esta herramienta la presencia de un service worker al acceder a Twitter.

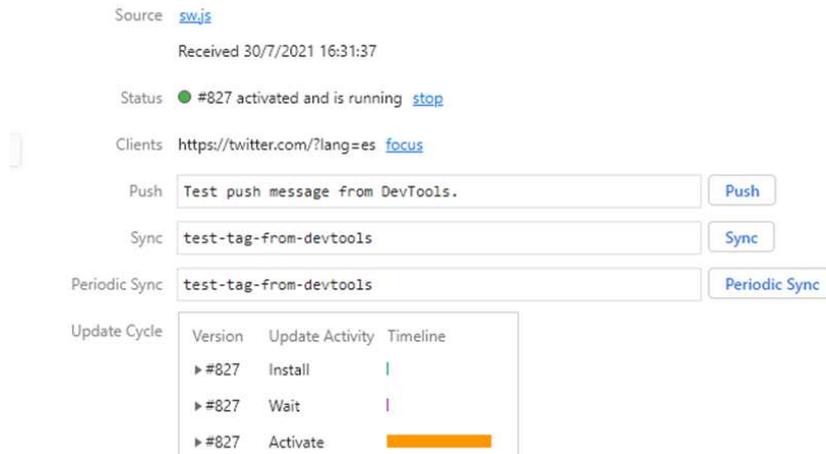


Fig. 2. Existencia de Service Worker – Ejemplo Twitter

3.2 Recursos

- Memoria: Se analiza el tamaño que ocupa en memoria la aplicación apenas el usuario se ha logueado a la misma (medido en MB).

- Service Worker: Se mide el espacio que consume el service worker (medido en KB)

A lo que se suman otros métodos de almacenamiento que pueden utilizar las aplicaciones PWA:

- IndexedDB: Base de datos local (medido en KB)
- Cache Storage: Almacenamiento local en el dispositivo (medidos en KB ó MB)

A modo de ejemplo se presenta el caso de Instagram en donde puede observarse que utiliza tanto IndexedDB como Cache Storage (en la descripción a la derecha en la figura 3) y además se observa el peso del service worker. Cabe aclarar que como todos los pesos están medidos en KB lo ocupado por Cache Storage en proporción es tan ínfimo en comparación de los restantes que no se ve en la gráfica de la izquierda.

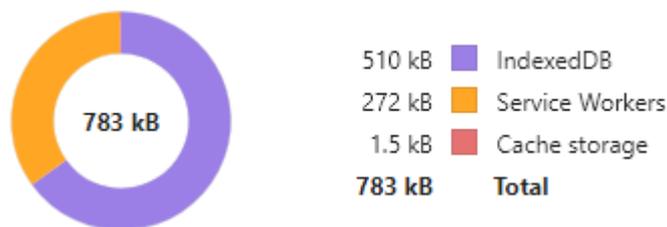


Fig. 3. Recursos – Ejemplo Instagram

3.3 PWA optimizada

Se complementa el análisis con una herramienta de código abierto “Lighthouse” [11], que puede ser ejecutada de diferentes maneras (en nuestro caso lo hemos utilizado como extensión del browser de escritorio) permitiendo a los desarrolladores analizar si la aplicación es una PWA Optimizada y el resultado que arroja es un valor del 1 al 8, según el cumplimiento de determinados parámetros:

1. Registra un service worker que controla las páginas y la url de inicio
2. Utiliza HTTPS
3. Contiene configuración para mostrar una página de inicio personalizada
4. Define un color de tema para la barra de direcciones
5. El contenido esta correctamente dimensionado al área de visualización (viewport)
6. Contiene el tag `<meta name="viewport">` configurado con un ancho o escala inicial
7. Provee un apple-touch-icon válido
8. El manifiesto contiene un ícono enmascarable

3.4 Mediciones de Tiempos

“En la actualidad vivimos en tiempos donde todo el mundo parece tener prisa, queremos todo “ya” y no deseamos perder ni tan solo un minuto. Lo mismo sucede cuando navegamos por internet, buscamos información y somos tan exigentes que tiene que ser buena y presentada rápidamente” [12]. Actualmente Google a través de la iniciativa de Web Vitals [13] propone tres métricas que permiten analizar los tiempos de una solución web. Esta iniciativa se centra en tres aspectos de la experiencia del usuario: carga, interactividad y estabilidad visual, e incluye las métricas mostradas en figura 4 se presentan los umbrales de las tres primeras métricas.

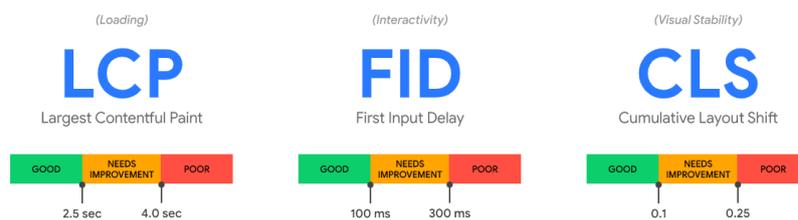


Fig. 4. Métricas consideradas

A continuación, se detallan las tres métricas propuestas:

- LCP (Largest Contentful Paint): Se refiere al tiempo para el despliegue del contenido más extenso, mide el rendimiento de la carga. Se establece que el tiempo debe producirse dentro de los 2,5 segundos desde el comienzo de carga de la página.
- FID (First Input Delay): Se refiere a la demora para la primera entrada, mide la interactividad. Para proporcionar una buena experiencia de usuario las páginas deben tener un FID menor a 100 milisegundos.
- CLS (Cumulative Layout Shift): Se refiere al cambio acumulativo en el diseño, mide la estabilidad visual. Se establece que el indicador, para proporcionar una buena experiencia de usuario, debe ser menor de 0,1.

De manera automática es posible medir el LCP y el CLS. “Las herramientas automáticas que cargan páginas en un entorno simulado sin un usuario no pueden medir FID” [13]. No obstante, muchas de ellas arrojan un valor de FID, pero el mismo no será considerado.

Por lo tanto, en este análisis se utiliza la herramienta PageSpeed Insights [14], que permite obtener tanto el LCP como el CLS.

4. Relevamiento

Se consideró una muestra de diez redes sociales, indicadas en la tabla 1 (ordenadas alfabéticamente). Las tres redes que han sido destacadas (en negrita) no se consideran

PWA por no tener un archivo manifiesto, por lo cual se someten al análisis a las siete restantes.

Tabla 1. Listado de redes sociales consideradas para el relevamiento

Nombre	Objetivo	Año de Lanzamiento
Facebook	Conectar con Personas	2004
Instagram	Compartir fotografías y videos	2010
LinkedIn	Oportunidades laborales	2003
Pinterest	Crear tableros personalizados con imágenes de interés	2010
Snapchat	Mensajería con soporte multimedia	2011
TikTok	Compartir Videos	2016
Tinder	Encuentros Online	2011
Twitch	Transmisiones en Vivo	2011
Twitter	Microblogueo	2006
Youtube	Compartir Videos	2005

5. Resultados Obtenidos

Como se mencionó previamente se descartan 3 redes sociales (Facebook, LinkedIn y Snapchat) por no ser PWA, el resto de las redes sociales han sido analizadas con el procedimiento descrito previamente.

En base al archivo de manifiesto sobresale Twitter que es la única red social (de las relevadas) que posee la descripción de todas las características analizadas, siendo la única que incorpora shortcuts y screenshots.

En cuanto al uso de memoria todas las PWA ocupan un tamaño desde 15 a 40 MB, el uso de memoria cambia ligeramente en cada ejecución de la aplicación dado que su contenido es dinámico, no obstante, resulta interesante extender el análisis pudiendo observarse que el service worker más pesado lo tiene Instagram que es la única red social (de las analizadas) que utiliza tanto indexedDB como CacheStorage. Todas las redes sociales analizadas utilizan indexedDB, siendo tan sólo 3 las que usan CacheStorage: Instagram, TikTok y Twitch.

En cuanto al puntaje de PWA extraído desde el reporte generado con Lighthouse pude observarse en la figura 5 que las redes sociales Tinder, Twitch y Twitter son las que mejor están adaptadas a PWA cumpliendo con los 8 ítems analizados por la herramienta.

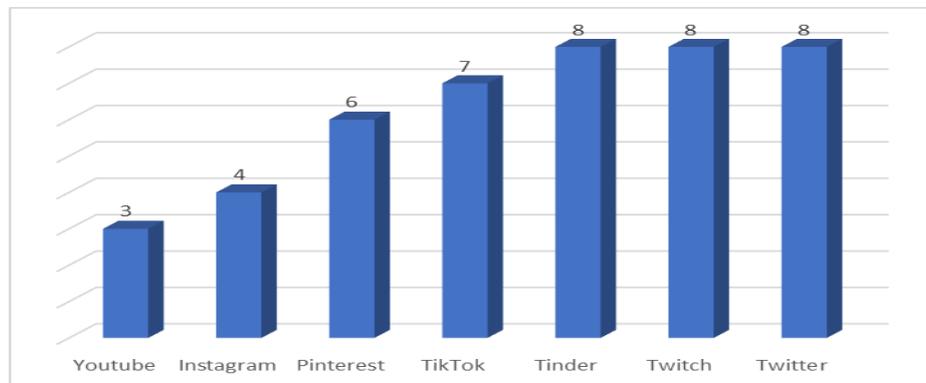


Fig. 5. Puntaje de PWA sobre un total de 8 puntos (obtenidos usando Lighthouse)

Tomando como ejemplo a las redes sociales mayormente asociadas con videos, puede advertirse las diferencias en el desarrollo posicionándose Youtube como la PWA menos optimizada (3 de 8), mientras que TikTok (7/8) está cerca de tener un desarrollo optimizado, sobresaliendo Twitch que se basa principalmente en streaming en vivo y es muy utilizada por radios y otros medios de comunicación, con una PWA optimizada (8 de 8).

En el análisis de tiempos mediante la herramienta PageSpeed Insights es posible observar que las tres redes sociales que estaban más optimizadas como PWA (Tinder, Twitch y Twitter), cumplen con los tiempos de carga tanto de FCP como de CLS. Como era de esperarse Twitch cumple con ambos parámetros mientras que Youtube sólo con 1 (CLS).

6. Conclusiones

Puede notarse como las redes sociales en general (70% de las relevadas) sacaron provecho de las posibilidades de las PWA, quedándose Facebook, LinkedIn y Snapchat rezagadas en este aspecto. De las redes sociales desarrolladas como PWA un 43% de las analizadas están optimizadas para tal fin. Sobresale el caso de Twitter que al inspeccionar el archivo manifiesto se evidencia una muy buena definición de todas las características analizadas. Como se mencionó en la sección de resultados, más allá del enfoque de la red social (se compararon tres redes sociales basadas en videos) sus desarrollos pueden ser optimizados alcanzando la mejor calidad de PWA, lo que impactará sin lugar a duda en la reducción en la descarga de datos (utilizando almacenamiento interno en el dispositivo) y usabilidad (mediante los parámetros definidos en el manifiesto). En este sentido se destaca por los resultados obtenidos en el relevamiento a Twitter.

De las redes sociales analizadas la que resultó ser la PWA más efectiva fue Twitter (por su observación de archivo de manifiesto, cumpliendo con los 8 ítems de análisis de PWA y también con los parámetros asociados a los tiempos de carga), seguida por Tinder y Twitch (ambas con un manifiesto incompleto).

Finalmente es importante concluir que si bien la mayor parte de las redes sociales han dedicado sus esfuerzos a la construcción de una PWA muchas de ellas aún les falta optimizar el uso de almacenamiento interno, tener un archivo de manifiesto más detallado y tomar en consideración cuestiones simples de configuración que permitirán tener una PWA optimizada.

Como trabajo futuro se propone analizar el parámetro FID y por otra parte indagar que optimizaciones podrían realizarse en estas PWA para que impacten mejorando en su rendimiento actual.

Referencias

1. De Haro, J. J. Redes sociales en educación. Educar para la comunicación y la cooperación social, 27, 203-216. (2010).
2. Digital 2021. We are social.
<https://wearesocial.com/digital-2021>
3. Digital Report 2021: el informe sobre las tendencias digitales, redes sociales y mobile
<https://wearesocial.com/es/blog/2021/01/digital-report-2021-el-informe-sobre-las-tendencias-digitales-redes-sociales-y-mobile>
4. Digital 2020: July Global Statshot
<https://datareportal.com/reports/digital-2020-july-global-statshot?rq=wifi>
5. Castell Ferreres, G. Desarrollo e implementación de una aplicación web progresiva (PWA) (Bachelor's thesis, Universitat Politècnica de Catalunya). (2020).
6. Branch. Estadísticas de la situación digital de Argentina en el 2020-2021
<https://branch.com.co/marketing-digital/estadisticas-de-la-situacion-digital-de-argentina-en-el-2020-2021/>
7. Rodríguez, R. A., Vera, P. M., Ramirez, M. A., Alderete, C. G., Conca, A. G., Dogliotti, M. G., & Zain, G. A. Análisis del Diseño Web Adaptativo Caso de estudio: Universidad Argentinas. Revista Abierta de Informática Aplicada (RAIA), 4(1), 51-62. (2020).
8. Ramírez Ivan. ¿Qué es una aplicación web progresiva o PWA? (2018)
<https://www.xataka.com/basics/que-es-una-aplicacion-web-progresiva-o-pwa>
9. Adetunji, O., Ajaegbu, C., Otuneme, N., & Omotosho, O. J. Dawning of Progressive Web Applications (PWA): Edging Out the Pitfalls of Traditional Mobile Development. American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS), 68(1), 85-99. (2020).
10. Vera, P., Rodriguez R., Estrategias de Manejo de Cache para Aplicaciones Web Progresivas - Presentación de un Esquema optimizado. Congreso Nacional de Ingeniería Informática. (2020)
11. Google Developers, Lighthouse
https://developers.google.com/web/tools/lighthouse/?utm_source=devtools
12. Echeverria, D. Tiempo de Respuestas y Experiencia de Usuario Estudio Experimental. Revista latinoamericana de ingeniería de software, 4(5), 231-234. (2016).
13. Philips Walton. Métricas esenciales para un sitio saludable.
<https://web.dev/vitals/>
14. Google Developers, PageSpeed Insights
<https://developers.google.com/speed/pagespeed/insights/>