

# DEVELOPER EXPERIENCE

Software Engineering Radio #524  
Abi Noda

Álvaro Davila Sampedro – [uo284548@uniovi.es](mailto:uo284548@uniovi.es)  
Andrés Ángel González Granda – [uo68216@uniovi.es](mailto:uo68216@uniovi.es)  
Gonzalo Rodríguez Rodríguez – [uo283874@uniovi.es](mailto:uo283874@uniovi.es)

## Contenido

Introducción .....	2
¿Qué es Developer Experience? .....	2
Developer Experience interno .....	2
Developer Experience externo .....	2
¿Se puede conocer de forma exacta la DX dentro de una empresa? .....	2
Ventajas y problemas que pueden aparecer .....	2
Beneficios de una buena DX .....	2
Barreras que impiden una buena DX.....	3
Problemas de una mala DX.....	3
Factores que afectan a la DX.....	3
Code Review .....	3
Entorno de desarrollo .....	4
Métricas de DX y de productividad.....	4
Malas métricas para medir DX .....	4
Buenas métricas para medir DX.....	5
Comparar la DX entre diferentes equipos.....	5
Técnicas para mejorar la DX como empresa .....	5
Técnicas para mejorar la DX como desarrollador .....	5
Referencias.....	6

# Introducción

## ¿Qué es Developer Experience?

### Developer Experience interno

La experiencia de usuario o User Experience (UX) es, según la [Wikipedia](#), “*el conjunto de factores y elementos relativos a la interacción del usuario con un entorno o dispositivo concretos*”. En resumen, la percepción de un producto por parte del usuario.

Para las empresas que desarrollan software para ser usado por desarrolladores, los desarrolladores toman el papel de usuarios del producto desarrollado. De esta forma, el usuario final pasa a ser el desarrollador y, por tanto, la User Experience pasa a ser la Developer Experience.

### Developer Experience externo

En el caso del [Episodio #524 de SE Radio](#), Abi Noda se refiere a esta última acepción y, junto con Brijesh Ammanath, definen la DX como la “*calidad de vida del desarrollador dentro de una empresa*”.

La DX es, según esta acepción, el conjunto de factores que afectan a la productividad, ausencia de distracciones y en general, a la satisfacción de los desarrolladores que trabajan dentro de un equipo o una empresa.

## ¿Se puede conocer de forma exacta la DX dentro de una empresa?

Hoy en día no existe una clasificación de empresas en cuanto a DX se refiere. Debido a esto, un candidato a desarrollador dentro de una empresa no cuenta con un lugar concreto para consultar la DX de una empresa. Sin embargo, aquellos desarrolladores que se encuentran a gusto dentro de una empresa se lo suelen hacer saber a otros desarrolladores. Por tanto, como desarrollador es muy importante estar al tanto de dónde están los buenos desarrolladores y dónde tienen mejores condiciones. Además, es necesario realizar preguntas sobre la DX durante las entrevistas con la empresa, aunque estas cuestiones se tratarán posteriormente.

## Ventajas y problemas que pueden aparecer

### Beneficios de una buena DX

La DX es uno de los factores que más estrechamente relacionados están con la productividad de una empresa. Además, las empresas que mejor DX tienen, son también, por norma general, las que mejores innovaciones traen al mercado y tienen una mejor productividad que su competencia.

## Barreras que impiden una buena DX

Uno de los factores que evitan que una empresa tenga buena DX es la falta de visibilidad de los problemas. Algunos de los motivos que fomentan una falta de visibilidad son la dificultad de medir la DX y la dificultad de los desarrolladores de expresar sus problemas. Es muy común en empresas que durante las reuniones los desarrolladores no tengan la suficiente confianza o sientan la suficiente seguridad para expresar sus problemas. Por esto, es interesante para las empresas proporcionar a los desarrolladores varias vías para expresar sus preocupaciones y problemas.

## Problemas de una mala DX

Los problemas que presenta una mala DX son muchos y muy variados dentro de las empresas. Uno de los más evidentes es la mala productividad y, por ende, una pérdida económica. Otro de los más preocupantes es la pérdida de talento dentro de la empresa cuando desarrolladores talentosos abandonan la empresa por falta de motivación o por descontento con la DX. Además, los desarrolladores que están descontentos con su entorno de trabajo suelen reducir su interés en su trabajo o incluso ralentizar los procesos del desarrollo de software.

## Factores que afectan a la DX

Existen muchos factores que pueden afectar a la DX: la complejidad del código, la facilidad para lanzar las pruebas, tener una dirección clara, unos buenos requisitos, etc.

El cómo afectan estos factores a cada desarrollador y a su DX dependen en gran medida de cada individuo.

El aspecto fundamental a la hora de enfrentarse a estos factores que pueden afectar a la DX es la experiencia profesional del desarrollador. A mayor experiencia profesional más fácil le será resolver problemas que afecten a la DX.

Otro aspecto muy importante son los intereses y las expectativas del individuo. Por ejemplo, un desarrollador que ha trabajado siempre con metodologías ágiles que se incorpore a un proyecto que utiliza una metodología clásica puede encontrar frustrante la “burocracia” del proyecto para aprobar un cambio en los requisitos. Gran parte de la DX se reduce a las percepciones y expectativas individuales de los desarrolladores.

## Code Review

Desde el punto de vista del revisado, los desarrolladores se pueden frustrar si tienen que esperar mucho tiempo para recibir realimentación o con las idas y venidas que pueden ocurrir en el proceso de revisión de código.

También puede afectar a la DX la calidad de la revisión. Comentarios muy escuetos como “OK” pueden interpretarse como falta de reconocimiento al trabajo realizado y no aportan nada a la revisión. En el extremo contrario, revisiones demasiado duras o estrictas pueden ser demasiado lentas y en ocasiones hacen a los desarrolladores perder interés. Por lo tanto, las revisiones son un ‘tradeoff’ entre calidad y rapidez y hay que intentar encontrar un balance entre ambas.

Desde el punto de vista del revisor, si el puesto o rol que ocupa en la empresa, (por ejemplo, desarrollador senior) implica realizar muchas revisiones de código también se puede frustrar y disminuir su DX. Hay que llegar a una solución de compromiso entre la calidad y exhaustividad de las revisiones de código y evitar que se conviertan en un cuello de botella.

## Entorno de desarrollo

Entre la variedad de factores que afectan la DX existe un subconjunto relacionado con la forma de organización empresarial que los desarrolladores consideran muy importantes. Aspectos como cuál es el proceso de aprobación o el de revisión de código. Cómo es la retroalimentación con los clientes.

## Métricas de DX y de productividad

Existen métricas facilitadas por el sistema automáticamente, como cuanto tiempo tardan en completarse las compilaciones o cuanto tiempo se tarda en aprobar los pullrequest. Sin embargo, los datos más interesantes son los obtenidos directamente de los desarrolladores. Si nos limitamos a las medidas obtenidas de forma automática solo se obtiene una comprensión muy limitada de la DX.

### Malas métricas para medir DX

- 1) Líneas de código. Ofrecen una vista de alto nivel del tamaño de un sistema software y cómo está evolucionando la base de código. No son una buena medida, ya que el número de líneas de código no implica mayor calidad de este. De hecho, en muchas ocasiones es preferible menos código que, en general, es más fácil de mantener.
- 2) Frecuencia de commits. Se suele premiar a los equipos con mayor número de commits. El problema con esta métrica es que no se refleja el valor real entregado al proyecto.
- 3) Recuento de pullrequest. Da una idea de la frecuencia de lanzamiento de nuevas versiones y persigue fomentar la entrega continua. No tiene en cuenta el tamaño o la dificultad de los pullrequest.

## Buenas métricas para medir DX

- 1) Tiempo de entrega
- 2) Tiempo medio entre fallos (mttr)
- 3) Frecuencia de despliegue

## Comparar la DX entre diferentes equipos

La DX se puede utilizar para comparar diferentes equipos, pero puede fomentar una dinámica de competición no saludable. Si se crean incentivos para premiar a los mejores equipos pueden darse casos de manipulación de las métricas.

## Técnicas para mejorar la DX como empresa

Actualmente las empresas están prestando mayor atención a la DX, de hecho, muchas de ellas están utilizando diferentes técnicas con el fin de mejorarla.

Una de ellas es la creación de equipos únicamente especializados en la mejora de la DX, estos equipos intentan comprender dónde los ingenieros están luchando, tienen una alta carga cognitiva o están haciendo un trabajo muy pesado dentro del proceso de desarrollo para luego crear soluciones. Por lo general, los equipos dan prioridad a las áreas que tienen el mayor impacto en la productividad de un gran número de desarrolladores.

Otra técnica muy usada es la de tratar a los desarrolladores de la empresa como si fueran clientes, preguntándoles sobre su satisfacción en la empresa y en sus diferentes entornos de trabajo, de hecho, muchas empresas están tratando de sistematizar el proceso de obtener feedback para así obtener mejores resultados.

A pesar de que existen técnicas y herramientas como las anteriores, muchas compañías se están dando cuenta que sus desarrolladores no las utilizan y esto se debe a que a estos no se atreven a dar su opinión o a participar en las reuniones. Esto se debe a que muchos ingenieros no sienten la comodidad o la seguridad para expresar su opinión en voz alta en reuniones en directo ya sea por vergüenza o por otros problemas. Por lo tanto, una buena solución como comentaba Abi Noda en el podcast sería la creación de métodos alternativos como puede ser encuestas o discusiones asíncronas que se realizan cada semana o cada mes evitando el problema anterior.

## Técnicas para mejorar la DX como desarrollador

Aunque las empresas juegan un papel importante en el aumento de la DX, los desarrolladores también deben ayudar y participar para mejorarla.

Una de las técnicas empleadas por un desarrollador para poder conocer cómo es la DX dentro de la empresa en la que quiere trabajar es a través de preguntas al entrevistador sobre diferentes aspectos.

A pesar de que existe un gran rango de factores que afecten a la experiencia de desarrollador, hay unos cuantos que son los que más destacan como puede ser el entorno de desarrollo, surgiendo así diferentes preguntas como:

- ¿Cuál es el proceso que seguir para poder trabajar, ejecutar y cambiar el código?
- ¿Cómo y cuánto se tarda en revisar y aprobar el código?
- ¿Cuál es el sistema de recompensa?

Pero también preguntando a los propios trabajadores de la empresa a través de foros o otros medios.

También existen estrategias para mejorar la DX dentro de un equipo de desarrollo siendo la más conocida “Job Crafting” que consiste en que los desarrolladores ajustan sus propios roles y descripciones de trabajo para satisfacer las demandas y lidiar con el problema que están encontrando.

Un buen ejemplo, sería el de un desarrollador senior que quiere gastar más tiempo para poder enseñar y educar a su equipo de desarrollo, pero no dispone del tiempo suficiente para ello y hable con el mánager para cambiar un porcentaje de su tiempo de revisar código en apoyar a su equipo.

## Referencias

[Episode #524: Abi Noda on developer experience](#)

[getclockwise.com: What is Developer Experience](#)

[medium.com: What is DX](#)

[usehaystack.io: Top 5 commonly misused metrics](#)

[swarmia.com: Dora Metrics](#)