## **C+** Escribe

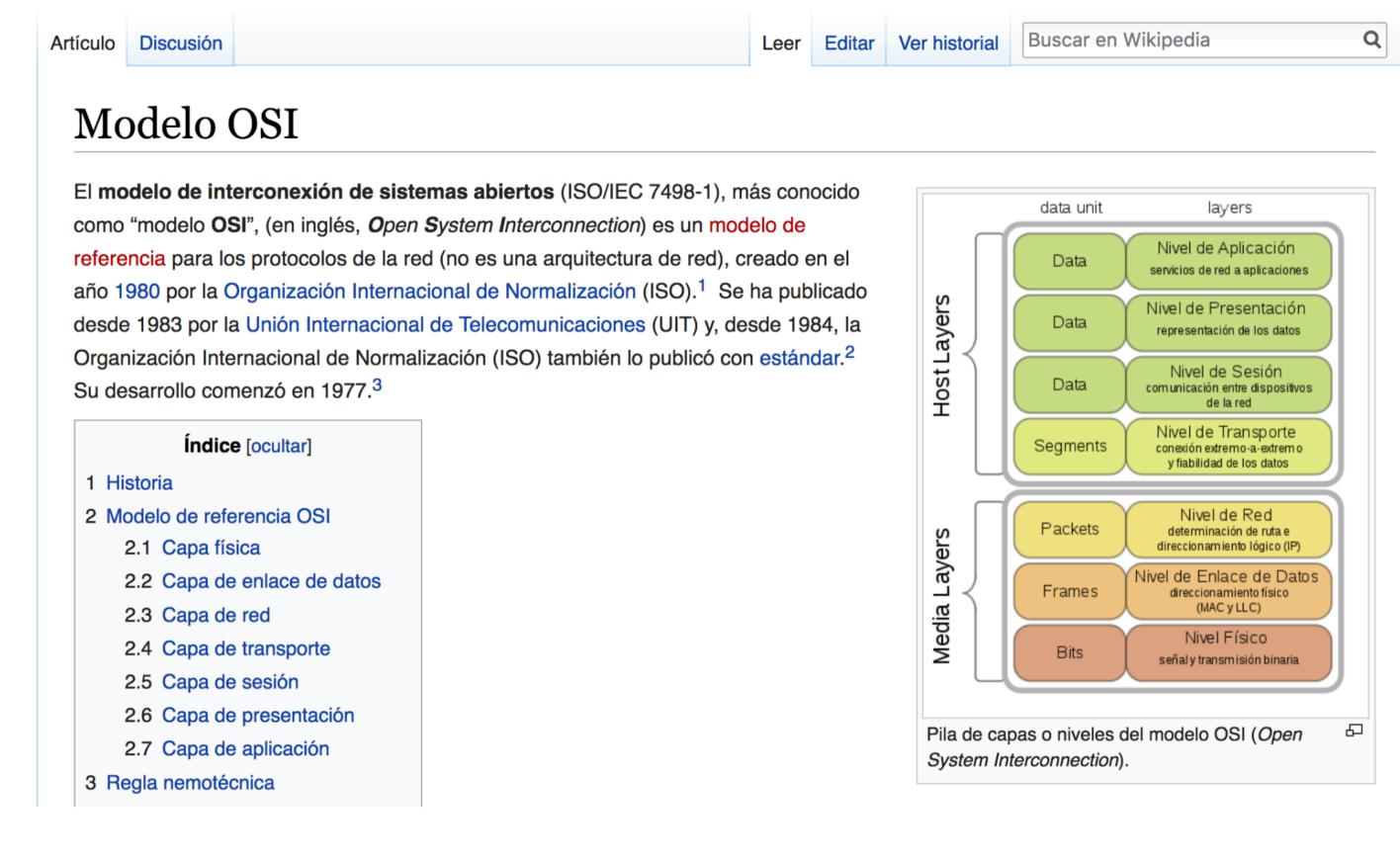
## T0x02 El bibliotecario

En este capítulo voy a presentarte la biblioteca en línea del diccionario órganico y de uso de la lengua española.

Primero te pondré en contexto. Querré justificar de forma indirecta, contándote una historia, que puedo hacer lo que pretendo. Que es abrir en enero del 2020 las puertas de la biblioteca en línea.

## Primera parte: puedo hacerlo

obra.



pondrá en contacto con la oficina de soporte. Explicará su problema. Para esos momentos deberá encontrarse todo en su sitio y listo para reaccionar. El problema será estudiado e identificado. Se buscará un

LLegará un momento en que alguien que está usando *el software* se

incidencia puede pasar al departamento de satisfacción al cliente. La colaboración del usuario a la hora de aplicar las actualizaciones expandirá la mejora por toda la plataforma y ese

workaround, un hotfix, una versión nueva que no contenga

notificará al usuario que el problema se resolvió y la

breaking changes. Un patch. Se aplicará al sistema y se le

error quedará subsanado. Pero falta bastante para ese punto. En este primer paso no hay nada preparado y todos se encuentran listos para iniciar la

El investigador que presentó el reto la semana pasada y su contexto. Así podría titularse lo que a continuación explicaré.

Tras un primer ciclo scrum, un primer año completo, el equipo

funciona en cinco columnas que marchan ora independientes ora

plegadas por los campos de batalla. La idea es que se ha

completado la primera versión beta, iteración a iteración,

backlog a la vista de las cinco columnas.

sprint a sprint. El sistema ya ha tomado su forma germinal. Se ha pasado, al menos una vez, por todos y cada uno de los casos de uso; y todas las features y las issues han sido eliminadas a la columna de closed en el tablero que todos estos meses ha soportado el

En los sprints 1-2 se ha generado via el swagger-editor una especificación yaml que cumple la especificación OpenApi y que se exporta a typescript para los sprint 3-4 en que se consolida la REST API.

Esto de arriba ocurrió en backend. Mientras en frontend se tomó

otra exportación typescript y con ella se agregó a una Angular 7 un share/models seguido de un lib/services que soluciona el CRUD sobre las entidades principales y rellena las constantes y las enumeraciones. Esto lo hicieron a partir del sprint 3-4, ya que en el sprint 1-

2 estaban creando la lib/ui como un paquete node-module donde se

aprovechando que iban creando la lib/features donde se almacenan

los ng-modules con los comunes de la app: desde la página about

componentes y sus directivas: botones, formularios, etcétera,

encuentran las abstracciones del framework ui con los

hasta la página de contact, desde la página de bienvenida hasta la página de configuración de la cuenta. Todos los test unitarios de estos componentes fueron clonados de repositorios de oss y los patterns fueron respetados sin miramientos. A pesar de ser de los primeros en la historia en usar Angular 7, no es necesario, y es importante no hacerlo,

reinventar la rueda cada vez. En el sprint 5-6 las cinco columnas se unieron en la retaguardia y estuvieron todos acampados. Concentrados en la wiki. Tanto la documentación swagger en línea como los manuales de uso de cada elemento en la carpeta lib (services, ui, features) fueron ultimados. Los creadores de los componentes hicieron charlas y

en conjunto se solventaron casos de usos no previstos o

potencialmente útiles.

tests asociados.

las primeras orientaciones de la estructura general de la obra. Explicaron, en varias meetings y sesiones de live-coding cómo se iban a usar todos esos componentes y servicios y features base para solventar las necesidades que los product-owner habían demandado a los scrum-masters. Los product-owners y los scrum masters convocaron la última

Durante el sprint 7 los scrum masters hicieron repaso y dieron

estaba lleno con la primera tanda de tareas. La versión 0.1.0, al final de los sprints, debía ser un hecho y sería reto y meta. En el sprint 10-11 solamente trabajaron los testers. La columna entera. Tanto los de backend como los de frontend. El resto de pesonal en las otras cuatro columas se disolvió. Una a una, la

que se quedó, tomó las tareas del backlog y fueron creando sus

reunión de preparación el primer día del sprint 9. El backlog ya

En paralelo, fuera del proyecto, en otra sede, otra gente, los admins (dos columnas, mitad sys admin, un cuarto webmaster y un cuarto help-desk) conectaron en la integración continua tanto la pila de tests unitarios como la de end-to-end. De los sprints 12 a 20 los testers tuvieron vacaciones. Tanto en

frontend como en backend tenían conectada una luz que cambia de

verde a rojo en las notas del tablero de tareas. Vía Travis cada desarrollador permanecía atento a las reuniones de planificación, a las diarias y a las de review siempre con un ojo en que todo su código tuviera las notas asociadas en complete y con el chivato de los tests en verde. En los sprints 12-13 la columna 1 y la columna 3 (íntegras de

frontend) trabajaron conjuntamente en los *reducers* para la *store* 

Redux. Fueron mapeando la carpeta shared/models y conectando los

estados de los *services*, *ui* y *features*. Mientras, la columna 2

(íntegra de backend) levantaba la base de datos ejecutando los

scripts de inicialización y enviaba a los admin notificación para agregar en la integración continua tanto la base de datos como el frontal web. La columna 4 (híbrida) en estos dos sprints completó los sistemas de autorización y notificación de errores mapeando los mensajes y errores que la API responde en metadatos en un servicio de popups y alerts. En los sprints 14-15-16 se inflaron las principales features. Tanto en backend como en frontend. Los mocks de datos fueron

entrando en el sistema y circulando en las vistas con sus

listas y contenedores de datos.

formularios y los desarrolladores haciendo pruebas manuales a

medida que completaban la creación. Los juegos de prueba fueron

importados en el background desde donde efundían a través de la

API al frontend y discurrían desde los servicios con RxJs a las

En el sprint 18 solo trabajó la columna 2 y los miembros background de la columna 4. Tuvieron la oportunidad de dedicar un sprint a refactorizar haciendo reuniones de retrospective diarias y manteniendo un hilo del sistema de mensajerías en tiempo real. Los servicios pudieron reiniciarse. La base de datos pudo vaciarse y recrearse. Nadie usaba el frontend y por tanto los admins pudieron reiniciar el cloud.

En el sprint 20 backend y frontend estuvieron de vacaciones. La columna de testers hizo la primera lista de RQs.

En los sprints 21 a 24 los admins dieron por finalizada la

de bugs en la integración continua también quedó en

prosiguió liquidando tareas y completando user stories.

En el sprint 19 background estuvo fuera de la oficina y frontend

funcionamiento. Durante esos sprints los testers mantuvieron viva la lista de

instalación del frontal help desk; y la conexión vía el frontal

RQs y las cuatro columnas de coders mantuvieron vivos los historiales de commits haciendo fixing y bug solving. Ya estas listas de RQs y todos los chivatos de Travis en verde, durante los sprints 25 y 26 solamente los product-owner y los

scrum-master trabajaron. Reunidos mañana y tarde. Esquemas y

listas de hitos tachados y archivados.

columnas de nuevo acampadas dedicas a labores de gestión y administración del entorno de desarrollo. Había que compactar carpetas de trabajo y eliminar copias de respaldo y PoCs deshechadas. Ese día era el día. Demo y release.

El primer día del sprint 27 había reunión grupal con todas las

Pero backend tuvo problemas con el cloud y no logró tener lista la presentación. A la hora indicada el scrum-master que hacía de maestro de ceromonias dijo lo de: "And now, we load the app, and you can see…" pero no se cargó la app y nadie pudo ver nada. El segundo día, misma hora, mismo lugar, mismos asistentes, todo

fue como la seda. El mc pudo completar el show y todos

de la demo y en pocas horas llegó el certificado firmado de que la release 0.1.0 ya estaba en línea. Todo esto había pasado hasta la fecha. Durante el primer año. Espero que los avispados hayan dividido las semanas que tiene un

año entre el número de sprints y conozcan entonces cuánto duraba

suspiramos victoriosos. Los product-owner aplaudieron al final

de la obra.

Literalmente, las cinco columnas fueron ágiles en el desarrollo

Encargado del contenido online de la nonata editorial e-Editar Artesanía

## ← Thinkers vs Artificers

cada uno.

Introduce aquí tu comentario...

Buscar ...

Responder

Archivos • <u>junio 2019</u> mayo 2019

Sitio web ofrecido por WordPress.com.

Personalizar Editar

Estadísticas •••