



**Universidad**  
Internacional  
de Valencia

## Product backlog y Sprint backlog

**Titulación:**

Máster Universitario en Desarrollo de  
Aplicaciones y Servicios Web

**Curso académico:** 2022-2023

**Alumno/a:** Gómez Olivencia, Rubén

**D.N.I.:** 78910013-A

**Asignatura:** Gestión de proyectos en  
entornos ágiles

## Índice general

<b>1. Introducción</b>	<b>3</b>
<b>2. Historias de usuario</b>	<b>3</b>
2.1. Resultado del análisis realizado	4
2.2. Mapa de las historias de usuario	8
2.3. Sprint backlog	9
<b>3. Gestión de tareas en Trello</b>	<b>9</b>
3.1. Avance de tareas	12
<b>4. Conclusiones</b>	<b>14</b>

# 1. Introducción

Hoy en día es conveniente hacer uso de las **metodologías ágiles** cuando realizamos la gestión de un proyecto, ya que cuentan con una serie de ventajas que nos van a permitir dar una respuesta más rápida y flexible ante posibles cambios durante la vida de desarrollo del producto.

Entre las ventajas que pueden aportar las metodologías ágiles, podemos destacar:

- Entregar resultados funcionales al cliente cada poco tiempo y así obtener *feedback* de lo realizado.
- Ser capaces de dar una respuesta ágil y flexible ante posibles cambios que puedan surgir (cambios en los requisitos, ante bajas de personal asignado al proyecto, dificultades durante el desarrollo...).
- Tratar de evitar burocracia innecesaria, para centrarnos de esta manera en el producto y sus funcionalidades.
- Se trata de involucrar al cliente y colaborar con él para conseguir un mayor valor al producto.

Con todo ello, se va a realizar la gestión de un proyecto que dará como resultado un gestor de contraseñas y documentación sensible basado en una plataforma web creada con [Angular](#) y que hará uso como *backend* del sistema [Vault](#) de la empresa [Hashicorp](#).

# 2. Historias de usuario

Para comenzar con la gestión del proyecto debemos obtener los requisitos del cliente y transformarlos en las conocidas como “**historias de usuario**”.

En ellas plasmaremos, en un lenguaje que pueda entender el cliente, las funcionalidades que el producto debe tener, a las que asociaremos otros datos como son:

- **Identificador único** para poder diferenciarlas del resto.
- Una **estimación de tiempos**
- **Prioridad** inicial para poder ordenarlas entre ellas.
- Se asignará a un desarrollador.
- Se añadirán posibles dependencias de otras historias de usuario.
- Una **descripción** de lo que se quiere conseguir.
- Unos **criterios de validación** por parte del cliente.

## 2.1. Resultado del análisis realizado

A continuación se van a plasmar las distintas “historias de usuario” obtenidas:

ID	Estimación	Prioridad	Asignación	Dependencias
1	15 horas	1	Ariane Olivencia	
<b>Desarrollar “esqueleto” del interfaz web</b>				
<b>Descripción:</b> Como usuario necesito un interfaz web para poder acceder a la aplicación y hacer uso de ella. Es la base del proyecto.				
<b>Criterios de validación:</b> El interfaz es <i>responsive</i> , visualmente atractivo y funcional.				

ID	Estimación	Prioridad	Asignación	Dependencias
2	5 horas	1	Rubén Gómez	1
<b>Crear “login” de usuario</b>				
<b>Descripción:</b> Como usuario quiero poder acceder a la aplicación utilizando como sistema de autenticación un usuario y contraseña.				
<b>Criterios de validación:</b> Los valores introducidos serán autenticados contra el <i>backend</i> que el cliente decida (base de datos o LDAP/Active Directory).				

ID	Estimación	Prioridad	Asignación	Dependencias
3	30 horas	1	Rafael Gómez	2
<b>Listado de secretos</b>				
<b>Descripción:</b> Como usuario quiero poder obtener un listado de los secretos existentes para poder navegar por ellos y seleccionar el que me interese.				
<b>Criterios de validación:</b> <ul style="list-style-type: none"> <li>• El listado completo se obtiene mediante un árbol jerarquizado.</li> <li>• Se puede navegar por el árbol seleccionando “directorios” que desplegarán a su vez su contenido.</li> <li>• También se puede acceder como si fuera un explorador de archivos.</li> <li>• Se puede buscar por el nombre de los secretos.</li> </ul>				

ID	Estimación	Prioridad	Asignación	Dependencias
4	10 horas	2	Carmen Olivencia	2
<b>Visualizar un secreto</b>				
<b>Descripción:</b> Como usuario quiero acceder a un secreto para visualizar su contenido.				
<b>Criterios de validación:</b> <ul style="list-style-type: none"> <li>• La visualización mostrará el secreto en formato HTML.</li> <li>• Si un usuario accede a un secreto “bloqueado” lo debe saber y no se le permitirá editarlo.</li> <li>• Al visualizar el secreto aparecerán botones con distintas acciones que se podrán realizar sobre él (editar, imprimir, ver históricos, borrar).</li> </ul>				

ID	Estimación	Prioridad	Asignación	Dependencias
5	25 horas	2	Rubén Gómez	2
<b>Modificar un secreto</b>				
<b>Descripción:</b> Como usuario quiero poder editar un secreto para realizar modificaciones.				
<b>Criterios de validación:</b> <ul style="list-style-type: none"> <li>• La modificación se realizará a través de un editor <b>WYSIWYG</b>.</li> <li>• El editor permitirá realizar modificaciones en formato <u>Markdown</u>.</li> <li>• El editor permitirá guardar el secreto y al hacerlo volveremos a la visualización del mismo.</li> <li>• Al comenzar la modificación el secreto se pondrá en estado “bloqueado”.</li> </ul>				

ID	Estimación	Prioridad	Asignación	Dependencias
6	10 horas	3	Asier Gómez	4
<b>Borrado de secretos</b>				
<b>Descripción:</b> Como usuario quiero poder borrar secretos para que desaparezcan.				
<b>Criterios de validación:</b> Antes de borrar debe existir un mensaje de confirmación.				

ID	Estimación	Prioridad	Asignación	Dependencias
7	15 horas	3	Ariane Olivencia	2
<b>Permitir gestionar ficheros como secretos</b>				
<b>Descripción:</b> Como usuario quiero poder subir ficheros al sistema para guardarlos cifrados.				
<b>Criterios de validación:</b> Se permite subir cualquier tipo de fichero y se podrá descargar. Se debe poder visualizar los ficheros más habituales en la propia web (imágenes y ficheros PDF) sin necesidad de descargarlos.				

ID	Estimación	Prioridad	Asignación	Dependencias
8	20 horas	4	Rafael Gómez	4
<b>Versionado de secretos</b>				
<b>Descripción:</b> Como usuario quiero poder ver las veces que el fichero ha sido modificado para poder ver los cambios realizados.				
<b>Criterios de validación:</b>				
<ul style="list-style-type: none"> <li>• Poder ver el número de versiones que tiene un secreto.</li> <li>• Poder ver una versión concreta del secreto.</li> <li>• Poder ver las diferencias entre versiones.</li> <li>• Poder restaurar una versión concreta del secreto.</li> </ul>				

ID	Estimación	Prioridad	Asignación	Dependencias
9	5 horas	4	Asier Gómez	2
<b>Imprimir secretos</b>				
<b>Descripción:</b> Como usuario quiero poder imprimir un secreto.				
<b>Criterios de validación:</b> Existe un botón para imprimir un secreto.				

ID	Estimación	Prioridad	Asignación	Dependencias
10	35 horas	2	Rubén Gómez	1
<b>Puesta en producción segura</b>				
<b>Descripción:</b> Como usuario quiero poder acceder a la plataforma de manera segura para realizar todo lo comentado hasta ahora				
<b>Criterios de validación:</b>				
<ul style="list-style-type: none"> <li>• El acceso web tiene que ser por HTTPS (certificado válido).</li> <li>• El servidor debe estar securizado ante accesos externos.</li> <li>• Los servicios deben auto-arrancarse en caso de caída.</li> <li>• Existe un registro de lo sucedido a cada secreto (auditoría).</li> </ul>				

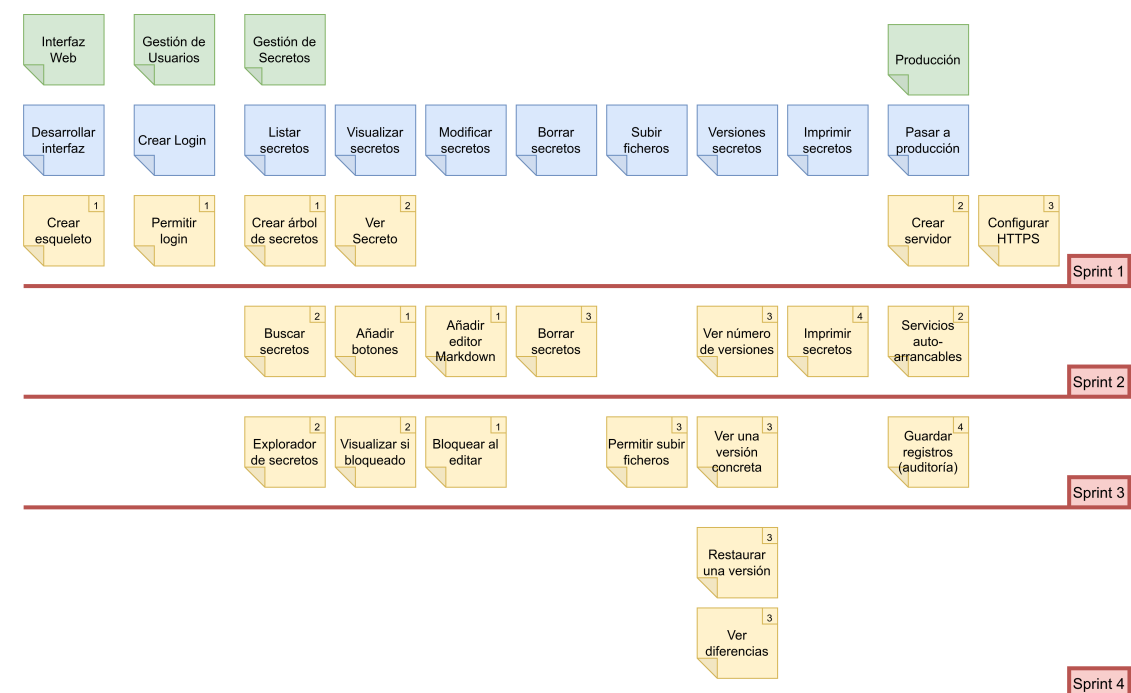
## 2.2. Mapa de las historias de usuario

Dado que las historias de usuario pueden suponer la agrupación de distintas tareas técnicas que son más pequeñas, es conveniente crear un mapa que englobe todas las tareas a realizar.

Para la organización de este mapa se pueden identificar los siguientes apartados:

- **Backbone** o temas, que son las características a alto nivel que va a tener la aplicación. En el mapa se ha diferenciado en color verde.
- Los **epics**, en color azul, son características que normalmente no pueden realizarse en un *sprint* y que también pueden ser divididas en distintas historias de usuarios que están relacionadas entre sí.
- Las **tareas**, en este caso en amarillo, son las unidades concretas de trabajo, tratando que sean lo más cortas posibles y que a nivel técnico sean auto-contenidas y aisladas.

El mapa resultante es el siguiente:



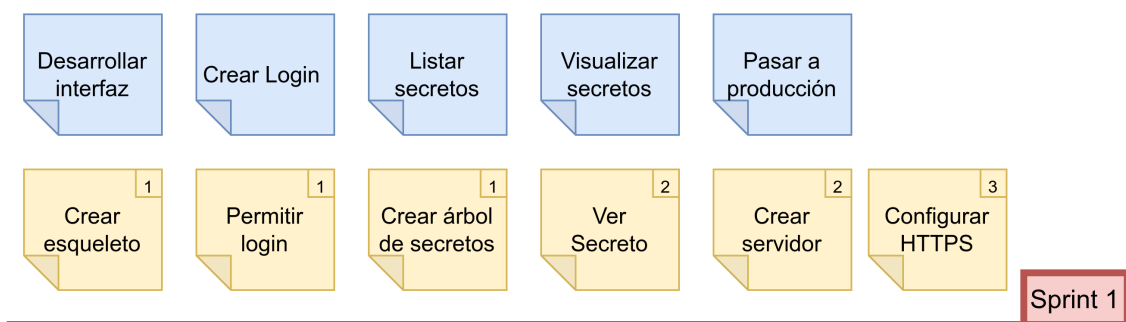


## 2.3. Sprint backlog

A la par que se ha ido creando el mapa anterior, se han diferenciado diferentes *sprints* con las tareas que deberían completarse para cada uno de ellos.

Bien es cierto, que aunque esa primera estimación haya sido realizada, se podrían efectuar modificaciones en caso de que fuese necesario.

Para el primer *sprint* se han identificado las siguientes tareas que deben realizarse:



Tal como se puede ver, las tareas están representadas con su prioridad dentro del *sprint*, para así también tener claro qué tareas deben tratar de terminarse antes que otras.

Al terminar este primer *sprint*, tendremos la base del proyecto terminada, y podremos mostrarle al cliente una primera versión con unas funcionalidades básicas. De esta manera, el cliente nos puede dar *feedback* de lo realizado.

Ese *feedback* podremos utilizarlo como retroalimentación para el siguiente *sprint*, ya sea para realizar modificaciones de lo ya realizado (hacer cambios nuevos que el cliente pida) o para utilizarlo para las tareas ya planificadas.

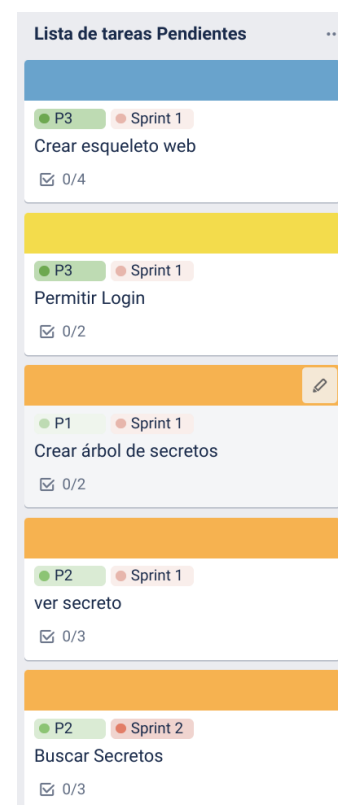
## 3. Gestión de tareas en Trello

[Trello](#) es una aplicación web, creada por la empresa [Atlassian](#), que nos permite tener un tablero [Kanban](#) donde podremos añadir las tareas de nuestro proyecto, e ir moviéndolas entre distintas etapas.

Aunque cuenta con una versión de suscripción, la versión gratuita tiene las características suficientes como para poder ser utilizado para la gestión de proyectos sin ningún problema.

Una vez definidas las tareas, tal como se ha visto antes, se han añadido a Trello, en el que se han creado las siguientes columnas, o fases de desarrollo:

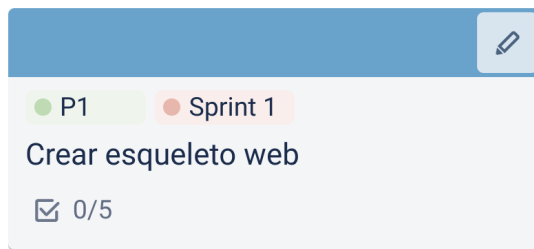
- **Tareas pendientes:** donde situaremos las tareas que se deben realizar para llevar a cabo el proyecto.
- **Desarrollo:** Son las tareas que se han empezado a desarrollar.
- **Testing:** Son las tareas que se dan por terminadas en el desarrollo y que pasan a una fase de testeo. En algunos casos puede ser directamente testado por el cliente para dar su feedback (por ejemplo para el interfaz web), aunque lo habitual es que el testeo lo realice otra persona del proyecto (que no haya sido quien ha hecho el desarrollo).  
  
En caso de que una tarea no pase esta fase, se anotará los inconvenientes y volverá a la columna de **desarrollo**.
- **Producción:** Una vez el *testing* ha terminado, se puede dar por terminada la tarea y debe ser incluida en producción.



*Tareas pendientes en Trello*

Tal como se puede ver en la imagen, cada tarea se representa como una pequeña tarjeta, apareciendo como un listado de todas ellas en la columna correspondiente en las que están situadas (en este caso, todavía en la tarea de pendientes).

A cada una de las tareas se les ha asignado un color y varias etiquetas que se van a detallar a continuación. De esta manera, Trello nos permitirá realizar búsquedas o visualizar las tareas con la etiqueta que nos interese.



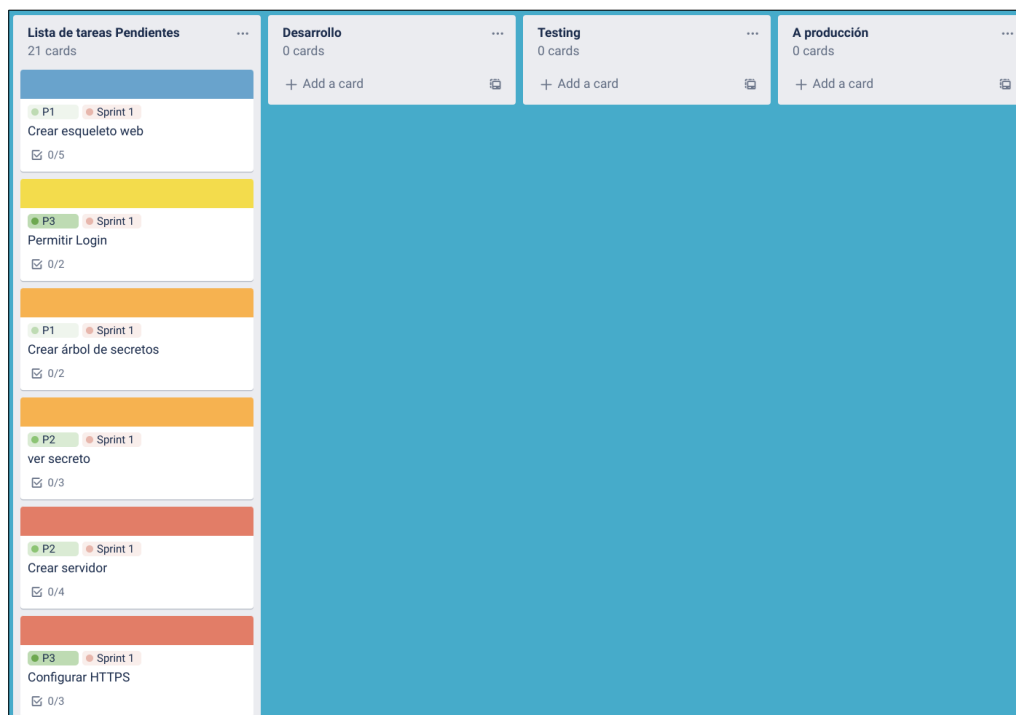
Detalle de tarea

A cada tarjeta se le ha dado un color, dependiendo del tema al que esté más orientada la tarea.

Otro aspecto son las etiquetas asignadas. En este caso cuenta con “P1” que marca la prioridad, siendo esta la más alta, y la segunda etiqueta referenciando a qué *sprint* pertenece.

Por último, cada tarea puede tener pequeños ítems. Estos han sido detallados para que el desarrollador tenga en cuenta qué es lo que debe hacer para dar por finalizada la tarea.

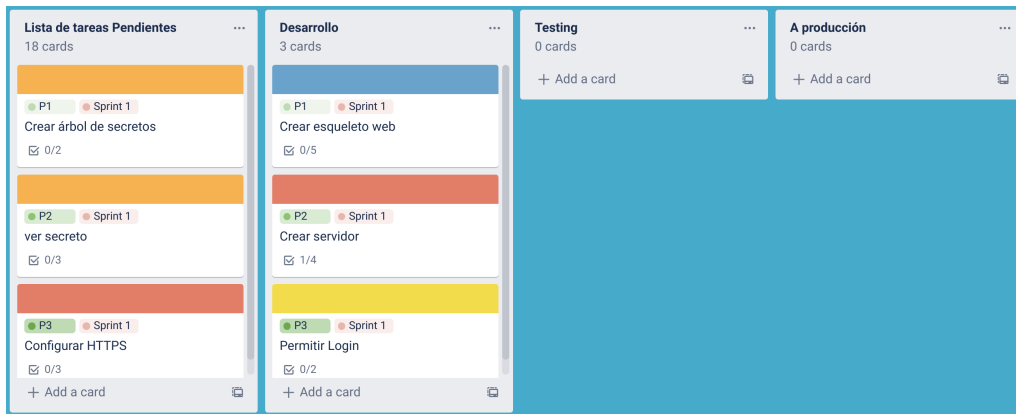
Tal como se ha dicho previamente, se pueden hacer filtros por las etiquetas, por lo que dependiendo del *sprint* en el que nos encontremos, podremos filtrar por dicha etiqueta. El panel principal de Trello, al inicio del primer *sprint*, queda de la siguiente manera:



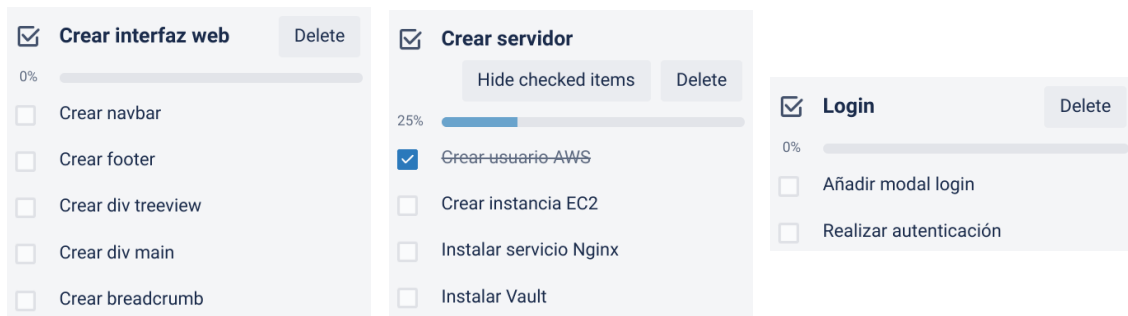
Este es el punto de partida y las tareas serán repartidas a sus correspondientes desarrolladores, que empezarán con ellas en cuanto se termine la asignación.

## 3.1. Avance de tareas

Tras asignar las primeras tareas, estas son movidas a la columna “desarrollo”.

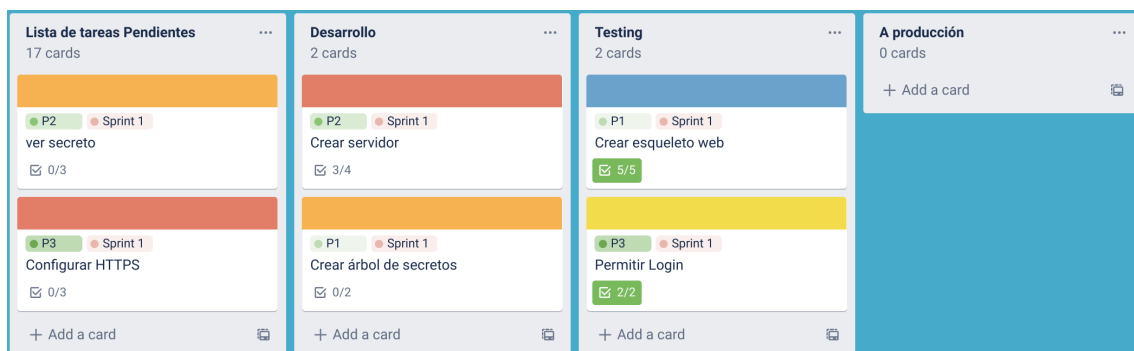



De esta manera, los desarrolladores comenzarán a realizar los ítems que tiene sus tareas. A continuación se muestran los ítems de las primeras tres tareas:



Como se puede ver, en la tarea de “Crear servidor” ya ha habido un pequeño avance porque se ha creado el usuario en el servicio [AWS](#) de Amazon.

Si dejamos avanzar un poco el tiempo, el panel avanzará a tener el siguiente aspecto:

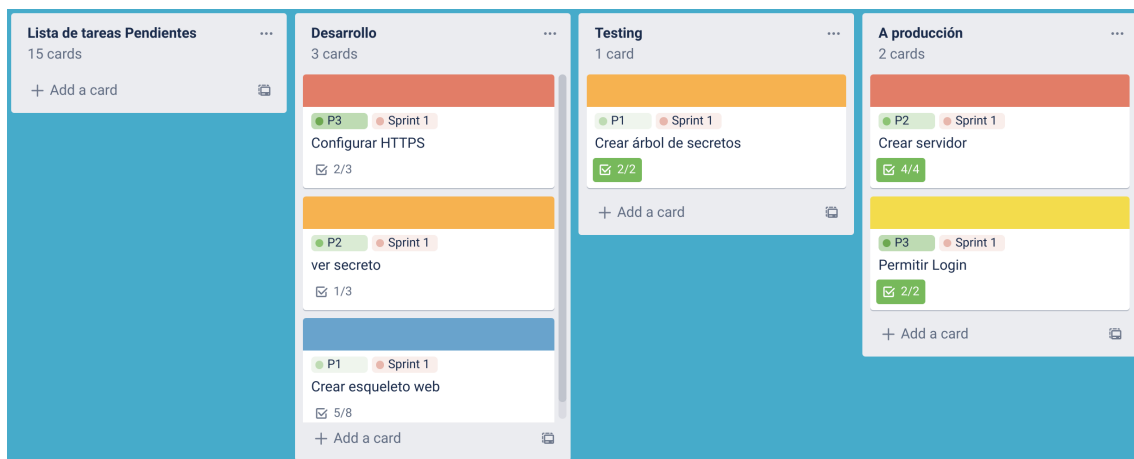


En este punto en el que nos encontramos, se puede ver que ha habido un avance de dos tareas a la columna “**Testing**”. Esto significa que se han completado todos los ítems  en cada tarea :

- **Crear esqueleto web** es la tarea inicial que crea la web para el desarrollo del proyecto. En este caso el testing va a ser llevado por parte del cliente para que ofrezca su *feedback* de lo realizado y por si quiere realizar algún tipo de cambio en colores o maquetación.
- **Permitir login** es la otra tarea completada y que por tanto debe testearse. Esta tarea debe testearse de manera correcta ya que es importante de cara al resto del proyecto.

En desarrollo sigue la tarea “crear servidor”, aunque ahora mismo se encuentra al 75 % y se ha comenzado a desarrollar la tarea “crear árbol de secretos”.

Si dejamos avanzar un poco más el tiempo el panel llega a tener este aspecto:



Uno de los aspectos principales que podemos ver es que en la columna de tareas pendientes ya no tenemos ninguna del *sprint 1* (sigue habiendo 15 tareas, pero ocultas, que son de los siguientes *sprints*).

Por otro lado, vemos que la tarea “crear esqueleto web” ha vuelto a la columna “desarrollo”, ya que el cliente nos ha otorgado *feedback* y se han añadido nuevos *ítems* (se ha pasado de 5 a 8), y se han identificado como:

- El cliente pide: modificar color del background.

- El cliente nos da un logotipo nuevo.
- El cliente pide cambiar el footer.

Estas modificaciones no se habían visto contempladas porque el cliente ha decidido hacer un *rebranding* de su marca y logotipo tras comenzar con el desarrollo de la aplicación, por lo que hay que adaptarse a ello.

Por otro lado, se ha pasado a producción dos tareas (“crear servidor” y “permitir login”), y otra que estaba previamente en desarrollo se ha dado por finalizada comenzando su testeo.

## 4. Conclusiones

Tal como se ha podido ver a lo largo del documento, antes de comenzar con el desarrollo de un proyecto hay un trabajo previo que debe realizarse, y que es muy importante como es el de crear las historias de usuario.

Con ellas podremos comenzar a realizar el **product backlog**, que posteriormente nos permitirá distinguir las distintas tareas y representarlas en los distintos *sprints* que llevaremos a cabo.

Tras esto, herramientas como [Trello](#) nos van a ayudar para que los desarrolladores puedan ver las tareas que deben realizar y cómo se encuentra el estado del *sprint* actual.