**tTrainer**

**Sprint 1**

**Autor:**

Guillermo Fernández Pérez

3ºA GITT

Madrid, Noviembre 2020

Lista de Cambios

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha** | **Descripción** | **Autor/es** |
| 0.1 | 10/12/20 | Primer borrador. | Guillermo Fernández Pérez |
| 0.2 | 01/01/21 | Formateo del documento. | Guillermo Fernández Pérez |
| 0.3 | 04/01/21 | Cambio de requisitos. | Guillermo Fernández Pérez |
| 1.0 | 07/01/21 | Revisión final. | Guillermo Fernández Pérez |

Índice

[1. INTRODUCCIÓN - 5 -](#_Toc60923078)

[2. ESTUDIO DE VIABILIDAD DEL PROYECTO - 6 -](#_Toc60923079)

[2.1. Identificación de necesidades - 6 -](#_Toc60923082)

[2.1.1. Alcance del sistema - 6 -](#_Toc60923083)

[2.1.2. Objetivo del sistema - 7 -](#_Toc60923084)

[2.1.3. Antecedentes - 8 -](#_Toc60923085)

[2.1.4. Descripción de Participantes - 8 -](#_Toc60923086)

[2.1.5. Restricciones - 9 -](#_Toc60923087)

[2.1.6. Enfoques Posibles - 10 -](#_Toc60923088)

[2.2. Análisis Coste-Beneficio - 11 -](#_Toc60923089)

[2.2.1. Viabilidad Técnica-Temporal - 11 -](#_Toc60923090)

[2.2.2. Viabilidad Económica - 12 -](#_Toc60923091)

[2.3. CONCLUSIÓN - 16 -](#_Toc60923092)

[3. PLANIFICACIÓN INICIAL - 17 -](#_Toc60923093)

Lista de figuras

[Ilustración 1: Equiquetas de la planificación - 18 -](#_Toc60923137)

[Ilustración 2: Planificación Sprint 1 - 19 -](#_Toc60923138)

# INTRODUCCIÓN

Más de 60 clases semanales, más de veinte clientes, más de 20 direcciones, todos con un plan nutricional personalizado, un plan de entrenamiento personalizado, una historia deportiva y médica propia... almacene todo esto en su cabeza. Tampoco es para tanto, ¿no?

¿Y si cada cliente tiene vida, proponiendo un cambio de horario casi semanal? ¿Y si hay que hacer dos planes nutricionales mensuales? ¿Y si cierto cliente es alérgico? ¿Y si los clientes quieren entrenar al aire libre un día cualquiera?

Esto es la vida normal de un entrenador personal y sólo tiene tres opciones:

* Ser un robot eficiencia u organización.
* Contratar un secretario.
* Contar con ayuda tecnológica.

Obviamente la última es la más barata y permite al entrenador enfocarse en su negocio. Aunque actualmente hay muchas herramientas como calendarios, aplicaciones de realizaciones de dietas, la idea es agrupar todas las necesidades en una única aplicación que solucione la mayoría de las necesidades de este sector. También hay aplicaciones ya desarrolladas que lo intentan, pero el problema que tienen a juicio de muchos entrenadores es que todas se enfocan a que el cliente interactúe con la aplicación consultando allí los planes nutricionales, sus citas, etc... Realmente los clientes quieren comunicarse con el entrenador directamente y que el solucione todo, que para eso le pagan.

Por lo que el objetivo tTrainer es la aplicación para entrenadores que aglutinará todas sus necesidades tecnológicas de un entrenador personal sin que los clientes sepan que la usa, (¡que sigan pensando que el entrenador es un robot!), permitiendo unas comunicaciones cliente-entrenador más humanas y facilite el día a día de la vida de un profesional de este sector.

A continuación se presenta el estudio de viabilidad de esta aplicación.

# ESTUDIO DE VIABILIDAD DEL PROYECTO

Para la iniciar un proyecto software se deben analizar los factores que van a estar presentes en todas las fases del desarrollo. El estudio de viabilidad se centrará en el factor temporal y económico.

Aunque este proyecto supone el trabajo de una asignatura, hay que evaluar tanto el coste en el que se incurriría en caso de un desarrollo comercial, como en los posibles beneficios que originarían su venta, y los conocimientos que se han sido adquiridos en el desarrollo de la aplicación y que no pueden ser tasados económicamente.

El análisis económico se centrará en la relación costes-beneficios, el estudio de esta relación dará una visión de viabilidad del desarrollo. Con los resultados obtenidos, se decidirá realizar o no el proyecto.

Hay que recalcar que el coste, en realidad es coste de oportunidad, ya que los desarrolladores no han percibido dinero alguno por el desarrollo del proyecto.



## Identificación de necesidades

En este apartado procederemos a resumir los problemas y el objetivo que tiene la presente aplicación.

### Alcance del sistema

Tras hablar personalmente con varios entrenadores personales con mucha experiencia en su sector se han identificado los siguientes problemas reales:

* Cambios continuados de horario de las sesiones con clientes, lo que provoca conflictos de horario entre clientes y sensación de caos generalizado.
* Rigidez de los sistemas de elaboración de planes nutricionales, ya que ninguna aplicación es totalmente flexible.
* Incapacidad de gestionar un histórico de resultados eficientemente.
* Diseño de tablas de ejercicios visuales de manera rápida.
* Envío de información a un cliente de manera sencilla.
* Gestionar todo desde el móvil.

Estos problemas se pueden resolver de una manera relativamente sencilla mediante una aplicación que resuelva cada uno de estos puntos.

### Objetivo del sistema

El principal objetivo de este software es la unificación de todas las herramientas tecnológicas necesarias que podría utilizar un entrenador personal sin necesitar iteración de los clientes.

Definiendo objetivos más específicos tenemos:

* Gestión de clientes: Posibilidad de introducir información de todos los clientes, desde información de contacto a un historial médico.
* Gestión de calendario: Gestión de citas con los clientes, tanto presenciales como online. Lo ideal sería poder integrarse con las herramientas de calendario más habituales.
* Coaching Nutricional: Elaboración de distintas menús diarios para los clientes en formato de texto libre. También se podrá guardar plantillas de menús para poder elaborarlas más rápido, así cómo se permitirá ver la información relevan te del cliente simultáneamente.
* Gestión de progresión de los clientes: Se permitirá la introducción en el sistema de diversos parámetros de los clientes, cómo peso, porcentaje graso, grasa abdominal, etc. Con estos datos se elaborarán informes que se podrán enviar a los clientes.
* Macros: Elaboración de tablas de ejercicios específicas para cada cliente, pudiendo ser creadas a partir de otras ya creadas.

### Antecedentes

El cliente potencial es todo entrenador personal que busque organizar sus necesidades digitalmente sin tener que involucrar a sus clientes.

Actualmente existen dos alternativas tecnológicas para entrenadores personales:

* Herramientas específicas para cada tarea: Consistiría en usar una herramienta o aplicación diferente para cada una de las tareas. Lo común suele ser por ejemplo el calendario del correo para la gestión de las citas o apps de gestión de menús. El inconveniente principal es que habría que utilizar una aplicación para cada necesidad y el posible coste de cada una de ellas. Por ejemplo:
  + Google Calendar
  + Apps de elaboración de menús
* Suites completas: Existen aplicaciones desarrolladas que cumplen uno o varios de los objetivos expuestos, el principal problema encontrado en ellas es que suelen necesitar de que el cliente acceda a la plataforma cómo medida para intentar captar beneficios a través de ellos ya sea por publicidad o un coste directo. Algunas de las más utilizadas son:
  + <https://business.virtuagym.com/>
  + bsport
  + dudyfit

### Descripción de Participantes

Para la elaboración del proyecto en esta primera fase inicial se contará con un único programador, que contará con la ayuda en el análisis de un entrenador personal que le definirá los objetivos de la aplicación.

Cuando la aplicación esté desplegada en un entorno productivo los participantes o roles del sistema serán:

* Administrador: Encargado de administrar el sistema en caso de errores o problemas. Hasta que no se defina una función de registro de entrenadores será el encargado del alta de los mismos.
* Entrenador: Usuario principal del sistema que será el encargado de utilizar las diferentes herramientas proporcionadas por la aplicación.
* Cliente: No interactuará directamente con el sistema, pero sus datos serán administrados por los distintos entrenadores, también podrán recibir información del sistema a través de integraciones con otras plataformas.

### Restricciones

A continuación se detallan las restricciones o barreras que se tienen en la elaboración, desplegue y explotación de esta herramienta:

* Económicas: No se dispone de ninguna clase de financiación para el desarrollo de la plataforma por lo que no se dispone de ningún entorno dónde desplegar, no se permite contar con ninguna herramienta o librería con licencia de pago así como no se contará con ninguna clase de consultaría externa con la que elaborar alguna parte de la aplicación.
* Temporales: Se dispone con un tiempo limitado para el desarrollo que corresponde con los plazos de entrega de la asignatura por lo que el objetivo será desarrollar únicamente un producto mínimo viable y dejar definidos unas posibles líneas de desarrollo futuras.
* Técnicas: Actualmente el desarrollador no dispone de conocimientos suficientes para desarrollar correctamente ciertas partes necesarias del programa como son:
  + UI: Distintas interfaces gráficas de plataforma móvil y elección de un lenguaje de programación más adecuado para el frontal web como podría ser AngularJs o ReactJS.
  + Plataforma: Instalación y mantenimiento en un entorno productivo.
* Seguridad: Actualmente se carece de una formación necesaria en este campo, lo cual es completamente necesario.
* Legales: La privacidad de todos los datos que se gestionan en la aplicación tienen que cumplir toda sería de normativas de protección de datos por sé que ser requeriría ayuda legal en este aspecto.

### Enfoques Posibles

Una vez introducidos los objetivos, el entorno y los problemas encontrados en la elaboración del proyecto, es el turno de analizar los posibles enfoques de podríamos aplicar para sacar un beneficio económico de la aplicación:

* Enfoque freemium: Consistiría en ofrecer a los entrenadores gran parte de la aplicación gratuitamente, a la vez que se ofrezca un soporte premium de pago o ciertas funcionalidades extras adicionales
* Enfoque pago como servicio: Se ofrecería una funcionalidad muy limitada de la aplicación gratuitamente, cómo podría ser gestionar menos de 5 clientes y ofrecer planes de pago mensuales o anuales.
* Enfoque publicitario: Consistiría en ofrecer la aplicación gratuitamente con anuncios incorporados, obteniendo beneficios de la publicidad. A la vez se ofrecería la posibilidad de mediante un pago único eliminar los anuncios de la publicidad.

Se descarta la primera opción de inicio ya que se necesitaría un volumen de clientes muy alto o que la herramienta tuviese funcionalidades muy complejas para poder obtener beneficio.

La segunda opción es la más rentable, pero habría que ofrecer una aplicación en nivel de madurez muy avanzado para que los clientes estén dispuestos a pagar por ella.

Así que de inicio se elegirá la tercera opción, aunque a priori puede ser la menos beneficiosa, al menos mientras la aplicación no se considere que esté en un estado 100% funcional. Y más adelante, con una base de usuarios establecida y una cantidad de funcionalidad suficiente, convendría reestudiar si migrar hacia otro enfoque monetario.

## Análisis Coste-Beneficio

A continuación se procede a analizar la viabilidad del proyecto desde un punto de vista principalmente económico.

### Viabilidad Técnica-Temporal

Para la realización de cualquier proyecto se deben considerar diversos aspectos. Por una parte, es necesario realizar una planificación temporal para delimitar el tiempo disponible para la realización del proyecto, y por otra parte los recursos técnicos necesarios para desarrollarlo.

Los recursos técnicos que requiere este proyecto no son elevados, y no suponen ningún problema. Se requiere de un desarrollador que sería el alumno y un *product* *owner*, el cual sería el profesor de la asignatura.

En cuanto a la viabilidad temporal, se ha contado con fecha de entrega máxima fija a enero de 2021, teniendo que tener cubierta un mínimo de funcionalidad y documentación. Hasta el tercer sprint.

En la planificación inicial se estimó una duración de 4 *sprints* de cuatro semanas para crear el producto mínimo viable, con los siguientes objetivos en cada sprint:

* Sprint 1: Preparación del proyecto y estudio de viabilidad.
* Sprint 2: Análisis del proyecto y gestión de usuarios de la aplicación.
* Sprint 3: Diseño del proyecto y primeras funcionalidades.
* Sprint 4: Terminar las funcionalidades restantes.

### Viabilidad Económica

Es necesaria la realización de un estudio de viabilidad económica previo al desarrollo del proyecto, para poder estimar los recursos que se van a consumir y la cantidad de beneficios que éste generará.

Este estudio consiste en una evaluación de los costes de desarrollo frente a los beneficios del producto final desarrollado. La parte más importante del estudio de la viabilidad se corresponde con el análisis coste-beneficio.

* + - 1. Análisis de Costes

Se debe realizar un análisis completo de todos los costes derivados del desarrollo del proyecto, costes de personal, de recursos hardware y de las licencias software.

Esta es la estimación correspondiente a la planificación inicial.

* **Costes de personal:**

Un desarrollador en la elaboración del proyecto, aunque no dedicado a tiempo completo en todo el periodo de desarrollo. Para realizar los cálculos se parte de un sueldo base de 35 € empresa por hora, con un horario limitado de 38 horas por semana.

Para obtener el coste del personal, se tiene que tener en cuenta el tiempo de desarrollo y el porcentaje de impuestos que se aplica a la empresa por empleado, siendo en este caso un 29,7%.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Coste** | €**/hora** | **Horas / Semana** | **Semanas / Sprint** | **Sprints Proyecto** | **Nº Trabajadores** | **Coste Total** |
| Empresa | 35 | 38 | 4 | 4 sprits | 1 | 5320 |
| Empleado | 24.61 | 38 | 4 | 4 sprits | 1 | 3740,72 |

Tabla 1: Costes de personal

* **Amortización del Hardware:**

Para el desarrollo del proyecto se han utilizado diversos sistemas y componentes informáticos, detallados a continuación:

Las características principales de los equipos son:

* Equipo 1: *Monitor HP EliteDisplay E243p de 60,45cm (23,8''),IPS,Full HD, Pantalla Full HD IPS con retroiluminación LED de 60,45cm (23,8''), Interruptor de modo de privacidad, 1920 x 1080 a 60 Hz. Tiempo de respuesta 5ms (GaG), 1 DisplayPort™ 1.2 (compatible con HDCP); 1 HDMI 1.4 (compatible con HDCP); 1 VGA*
* Equipo 2: *HP ProBook 440 G6 Core i7 8565U / 1.8 GHz Win 10 Pro 64 bits 16 GB RAM 512 GB SSD NVMe, TLC 14" 1920 x 1080 (Full HD)*

*UHD Graphics 620 Wi-Fi, Bluetooth kbd: español*

*REF: 6MR16EA*

|  |  |  |  |
| --- | --- | --- | --- |
| **Equipo** | **Coste** | **Unidades** | **Total** |
| Equipo 1 | 480.37 € | 1 | 480.37 € |
| Equipo 2 | 831.72 € + 5.45 € Canon digital | 1 | 837.17 € |
|  |  |  |  |
|  |  | **TOTAL** | 1317.54 € |

**Tabla 2: Coste de equipos.**

Teniendo en cuenta el tiempo de amortización de los equipos, definido en 4 años el coste real asociado a éstos sería:

*Coste de amortización por mes* = 1317.54€ / 48 meses = 27.45€/mes

**(Ecuación 1)**

Por tanto, el coste de amortización del hardware para el presente proyecto se muestra en la Ecuación 2, siendo equivalente a multiplicar el coste de un mes por el número de meses de desarrollo del proyecto.

*Coste amortización software* = 27.45€/mes \* 8 meses = 219.60€

**(Ecuación 2)**

* **Licencias de Software:** En la tabla se muestra el precio de cada licencia de software utilizada, que no sea libre o gratuita. //TODO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Software** | **Tipo de licencia** | **Nº de licencias** | **Coste / licencia** | **Total** |
| Windows 10 Home Premium | Por equipo | 1 | 95 € | 95 € |
| IntelliJ | Por equipo | 1 | 149€ | 149€ |
| Microsoft Office Home and Business 2019 | Por equipo | 1 | 199.13€ | 199.13€ |
|  |  |  | **TOTAL** | 443.13€ |

Tabla 2: Costes licencias de software

Al igual que anteriormente, se calcula la amortización del software. Aplicando la Ecuación 3 se calcula el coste de amortización de cada mes, considerando una amortización de software a 4 años.

*Coste de amortización por mes* = 443.13 €/48 meses = 9.23 €/mes

**(Ecuación 3)**

Por tanto, el coste de amortización de licencias software para el presente proyecto se muestra en la Ecuación 4, siendo equivalente a multiplicar el coste de un mes por el número de meses de desarrollo del proyecto.

*Coste amortización software* = 9.23 €/mes \* 8 meses = 73.84 €

**(Ecuación 4)**

* **Coste total del proyecto:**

Finalmente, los costes totales de desarrollo son los siguientes:

Coste total = coste amortización software + coste amortización hardware + coste personal

*Coste total* = 73.84 + 219.60 + 5320 = 5613.44 €

**(Ecuación 5)**

Hay que tener en cuenta que estos cálculos son orientativos y no se tienen en cuenta todos los factores, pero sirven como referencia del coste total.

* + - 1. Análisis de beneficios

Se puede estimar una aproximación comercial de lo que podría generar en base a diferentes criterios:

* **Beneficios económicos**

Unos supuestos beneficios con la venta del producto, a partir de los siguientes conceptos:

* Coste total del proyecto.
* Margen comercial.

Y la siguiente fórmula:

*Coste de venta = Coste Total del Proyecto + Margen Comercial (8% del Coste Total)*

*Coste total* = 5613.44 + 0.08 \* 5613.44 = 6062,52€

**(Ecuación 6)**

Se ha decidido que el margen sea del 8% por no contar con experiencia en venta de este tipo de productos, ni tener el respaldo de una empresa o marca de prestigio, por lo que para introducirse en el mercado se contará únicamente con los beneficios de publicidad integrada en la aplicación y algún usuario que pague por la eliminación de los mismos.

* **Beneficios no económicos**

Enumerados a continuación:

* Desarrollo de una aplicación de cero por parte del alumno.
* Construcción del proyecto utilizando varias librerías gratuitas, construyendo un producto novedoso y completo.
* Desarrollo de una interfaz de usuario sofisticada.
* Aprendizaje en el proceso de desarrollo de aplicaciones utilizando tecnologías y herramientas actualmente utilizadas en equipos de desarrollo.
* Aplicación de metodologías ágiles en el desarrollo del proyecto.

## CONCLUSIÓN

Una vez presentado el plan completo de viabilidad del servicio se considera que con las restricciones actuales, principalmente al disponer de un único desarrollador junior y no disponer del respaldo de ninguna empresa o entidad, económicamente no es posible recuperar la inversión realizada.

Aun así el desarrollador considera que los beneficios no económicos podrían compensar el esfuerzo necesario para la elaboración de la aplicación ya que supondría la actividad formativa más importante realizada hasta la fecha.

Además que aunque no se considera que se sea capaz de obtener un verdadero rédito económico, con el esfuerzo suficiente sí que se considera que se podría obtener beneficios suficientes para que al menos sufragar los costes de infraestructura con la publicidad de la aplicación.

Unido esto a que tener una aplicación desplegada puede ser una distinción importante a la hora de la incorporación del desarrollador en un mercado profesional, se ha decidido poner en marcha el desarrollo de la aplicación.

# PLANIFICACIÓN INICIAL

Tal y como se ha comentado en el estudio de la viabilidad, se ha optado por un desarrollo iterativo para este proyecto apoyado en un tablero Kanban, con 4 sprints para crear un producto mínimo viable y la documentación del proyecto.

Como herramienta para esta planificación se ha utilizado la herramienta Trello. Organizando el tablero de la siguiente manera:

* Columnas:
  + Backlog: Con las tareas a realizar a lo largo de todo el proyecto. Aquí es dónde de inicio se insertan nuevas tareas.
  + Sprint Backlog: Tareas a realizar en el sprint en curso. A principio del sprint se seleccionan que tareas se van a realizar ese sprint, se analizan más en detalle y se dejan en esta columna.
  + To-Do: Tareas seleccionadas para realizar a continuación.
  + Doing: Las tareas que se están realizando en este momento, lo ideal es que haya una por desarrollador, aunque en ocasiones habrá dos que se realizarán simultáneamente.
  + Done: Tareas completamente terminadas.

En ocasiones las tareas podrán volver a estados anteriores si se encuentran problemas relacionados con estas tareas, mejoras, etc.

Además de este sistema de organización cada tarea podrá tener una serie de etiquetas para diferenciar diferentes cosas como por ejemplo el sprint en el que se planea incluirlo o el tipo (técnica, documentación...) Inicialmente se cuenta con las siguientes etiquetas:

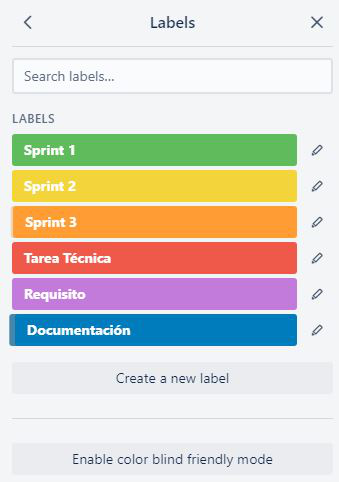


Ilustración 1: Equiquetas de la planificación

* Sprint: El sprint dónde se planea realizar la tarea.
* Tarea Técnica: Tarea relacionada con la tecnología de la aplicación que no implica cambios para el usuario. Por ejemplo requisitos no funcionales.
* Requisito: Requisito de usuario relacionada con una funcionalidad de usuario.
* Documentación: Toda tarea relacionada con la entrega de documentación del proyecto.

Teniendo esto en cuanta, la planificación inicial queda de la siguiente manera:

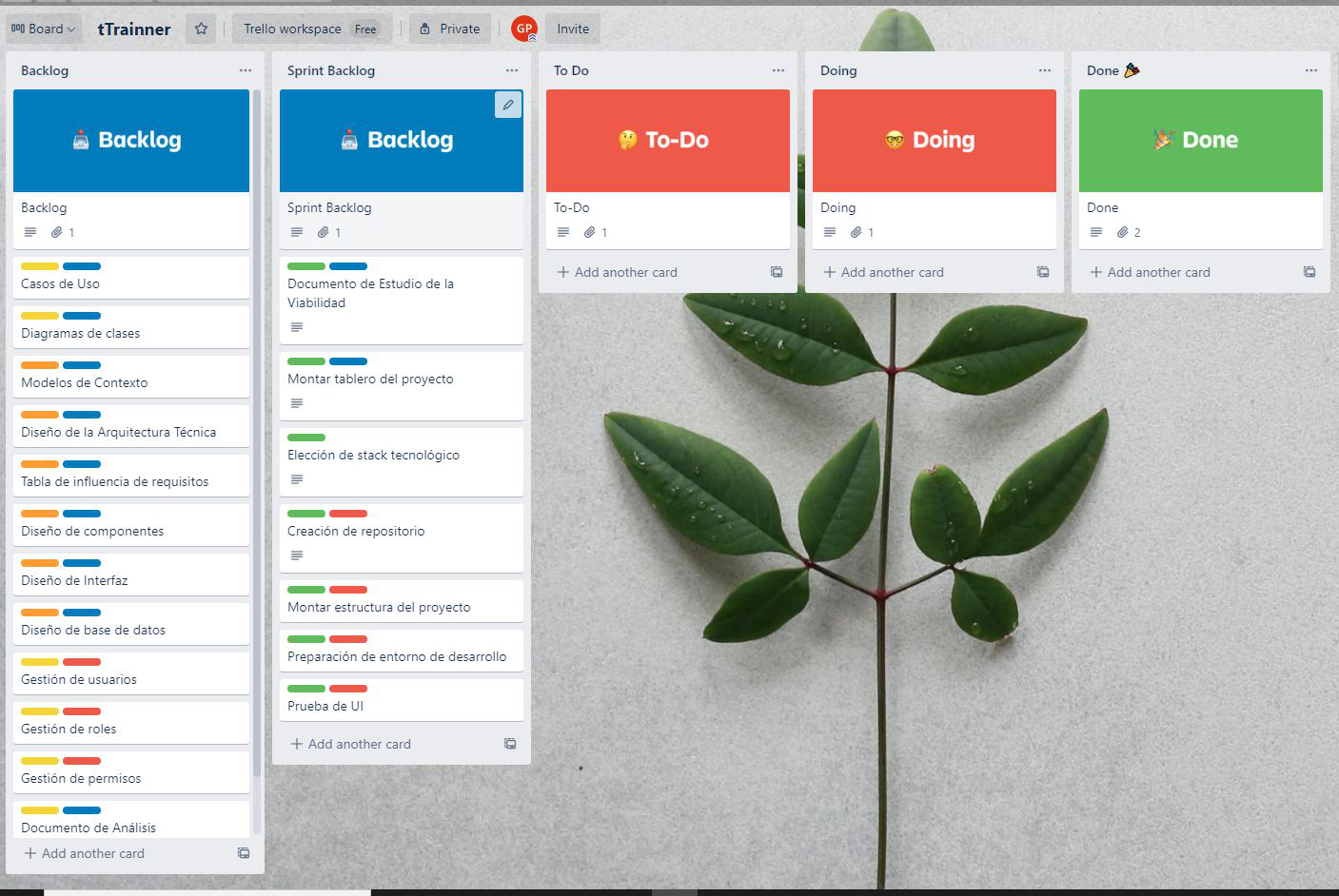


Ilustración 2: Planificación Sprint 1

De inicio se ha planificado lo siguiente:

* Sprint 1: Montar el proyecto junto a toda su infraestructura, junto con el estudio de viabilidad del proyecto.
* Sprint 2: Gestión de usuario, roles y permisos básicos de la aplicación más tareas de documentación de Análisis.
* Sprint 3: Documentos de diseño más algún requisito que salga del proceso de análisis y diseño.
* Sprint 4: Continuar realizando requisitos.