



UNIVERSITAT
POLITÀCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Web de suscripción mensual para bot de Instagram

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: Miguel Ángel Cerver Copoví

Tutor: Manuela Albert Albiol
María Victoria Torres Bosch

2020-2021

*“Dedicado a mi madre,
por todo lo que me ha dado en esta vida”*

Resumen

El presente proyecto de emprendimiento consiste en el análisis y el desarrollo de una idea de negocio basada en la oferta de un servicio de automatización de interacciones en Instagram. El objetivo de la automatización de interacciones es el crecimiento (en número de seguidores y actividad) orgánico de la cuenta de Instagram del usuario.

Para lograrlo se ha hecho un estudio de mercado analizando posibles competidores, se han utilizado distintas metodologías que han ayudado a trazar el plan de empresa y finalmente se han materializado las conclusiones obtenidas en dicho estudio en una versión preliminar completamente funcional del producto final.

El producto es una aplicación web mediante la cual los usuarios se registran y contratan un plan de suscripción, el cual les da acceso al servicio que realiza las interacciones. Se les da acceso a un panel de configuración, donde pueden parametrizar el comportamiento del servicio de automatización para ajustarlo a las particularidades de sus cuentas.

Este TFG se ha desarrollado en el marco de Start.inf, el espacio de emprendimiento de la ETSInf.

Palabras clave: emprendimiento, Instagram, bot, web, seguidores

Abstract

This entrepreneurship project consists of the analysis and development of a business idea based on the offer of an interaction automation service on Instagram. The goal of the automation of interactions is the organic growth (in number of followers and activity) of the user's Instagram account.

To achieve this, a market study has been carried out analyzing possible competitors, different methodologies have been used to help to draw up the business plan and finally the conclusions obtained in said study have been materialized in a fully functional preliminary version of the final product.

The product is a web application through which users register and contract a subscription plan, which gives them access to the service that performs the interactions. They are given access to a configuration panel, where they can parameterize the behavior of the automation service to adjust it to the particularities of their accounts.

This TFG has been developed within the framework of Start.inf, the entrepreneurship space of the ETSInf.

Keywords: entrepreneurship, Instagram, bot, web, followers

Tabla de contenidos

1. Introducción.....	6
1.1 Motivación.....	6
1.2 Objetivos.....	6
1.3 Impacto esperado.....	7
1.4 Estructura	7
2. Evaluación de la idea de negocio	9
2.1 Estudio de mercado.....	9
2.1.1 Clientes objetivo.....	9
2.1.2 Productos competidores	9
2.2 Análisis DAFO	12
2.3 Modelo de negocio y proyección de ingresos y gastos.....	13
2.4 Lean Canvas	16
2.5 Conclusiones de la evaluación.....	18
3. Metodología y tecnologías aplicadas.....	19
3.1 Metodología seguida	19
3.2 Tecnologías y lenguajes utilizados	20
3.2.1 PhpStorm	20
3.2.2 Adobe Illustrator.....	20
3.2.3 Adobe Photoshop	21
4. Análisis de requisitos.....	22
4.1 MVP I.....	22
4.2 Cronología del proyecto	23
5. Diseño y arquitectura del sistema.....	25
5.1 Diseño de la interfaz gráfica.....	25
5.2 Modelo de datos	29
5.3 Arquitectura software.....	30
6. Implementación.....	32
6.1 Implementación del front-end.....	56
6.1.1 Login/Registro.....	36
6.1.2 Recuperar contraseña	38
6.1.3 Mis cuentas.....	40
6.1.4 Centro de operaciones	42
6.1.5 Configuración.....	44
6.1.6 Suscripciones y pagos.....	47

6.1.7	Ajustes	49
6.1.8	Landing Page.....	49
6.2	Implementación del back-end	56
6.2.1	Creación/actualización de las configuraciones	56
6.2.2	Asignación o eliminación de configuración para un día de la semana.....	58
6.2.3	Desactivación de usuario cuando vence suscripción	60
6.2.4	Activación de usuario cuando reactiva suscripción	61
6.4	Pasarela de pago	62
6.4.1	Webhook.....	62
6.5	Desafíos técnicos	64
6.5.1	Inestabilidad en el bot de Instagram	64
6.5.2	Problemas para obtener datos de Instagram.....	65
7	Validación.....	67
8	Conclusiones.....	72
9	Bibliografía	73



1. Introducción

En esta sección se va a exponer la motivación que ha llevado a la realización del presente TFG, los objetivos que se desean alcanzar con el desarrollo del mismo y el impacto que se espera que tenga en el usuario que utilice el producto una vez desarrollado.

Para concluir, se va a comentar las diferentes secciones que componen el TFG y sus principales puntos.

1.1 Motivación

El presente proyecto se ha ideado fruto de la necesidad que tiene un negocio de crear o ampliar la cartera de clientes en un mercado donde la competencia es muy alta. Cada vez es más complicado dar a conocer los productos o servicios que se ofrecen sin gastar ingentes cantidades de dinero y obtener una buena tasa de conversión con la que salga rentable dicha inversión en marketing.

Las redes sociales son el escaparate perfecto para cualquier empresa o negocio. Cuando un usuario sigue la cuenta de una empresa, los contenidos que ésta publique aparecerán en el muro del usuario consiguiendo de esta forma publicidad completamente gratuita. Si la empresa cuenta con un gran número de seguidores, el impacto de esta publicidad puede suponer un incremento significativo en las ventas. Otra forma de publicidad muy utilizada en las redes sociales es la basada en la figura del *Influencer*, una persona, normalmente especializada en un sector concreto (belleza, moda, motor, etc.), que tiene una gran cantidad de seguidores los cuales confían en su criterio. Muchas empresas hacen contratos publicitarios con ellos para que recomienden sus productos.

El reto para un negocio en las redes sociales es conseguir una comunidad amplia y sólida de seguidores, ya que, aunque se tenga un contenido de calidad, muchas veces se necesitan fuertes inversiones en publicidad para lograrlo.

Con el producto que se ha desarrollado en este trabajo final de grado (TFG), tanto empresas como usuarios tienen la posibilidad de hacer crecer sus cuentas de Instagram de forma significativa y a un bajo coste. Es la herramienta perfecta, y a nuestro juicio imprescindible, para poder crear una gran comunidad de seguidores o ampliar la que ya se tiene.

1.2 Objetivos

El objetivo del proyecto de emprendimiento en el que se enmarca este TFG es la creación de una empresa que mediante una aplicación web oferte a los usuarios un servicio de crecimiento autónomo para sus cuentas de Instagram.

Para ello, hay que hacer un estudio de la viabilidad de la idea de negocio en el que se compruebe que el proyecto es viable económicamente y que aporta valor al

mercado para permitir posteriormente sentar las bases del mismo y trazar el plan de empresa.

Una vez completado todo el estudio de la idea de negocio y en caso de obtener una confirmación positiva del mismo respecto a su viabilidad, proceder a su desarrollo completo culminando con lanzamiento del mismo al mercado.

Sin embargo, el tiempo de realización de un TFG está comprendido entre 300 y 400 horas aproximadamente. Este tiempo es insuficiente para completar el proyecto de emprendimiento, que abarca desde el análisis de la idea hasta el lanzamiento final al mercado. Así que el TFG comprende desde la fase de análisis hasta el desarrollo de un producto preliminar funcional, aunque con limitaciones e incompleto.

Por tanto, el presente TFG tiene una doble finalidad.

En primer lugar, estudiar de la viabilidad comercial y el plan de empresa de una idea de negocio basada en la oferta de una aplicación que hará crecer las cuentas de Instagram de los clientes de manera autónoma mediante interacciones automatizadas.

En segundo lugar, desarrollar una versión preliminar completamente funcional de la aplicación.

1.3 Impacto esperado

Mediante el uso de la aplicación propuesta en este TFG el usuario experimentará un notable incremento en el número de seguidores de su cuenta de Instagram, así como en las interacciones que recibirá en la misma (por ejemplo me gustas, comentarios, seguimientos, etc.). En el caso de un negocio, resultará en un incremento de la captación de potenciales clientes y por tanto en un aumento de las ventas generadas.

Sin embargo, la calidad del contenido de la cuenta está estrechamente relacionada con el grado de impacto que tendrá la aplicación. Cuanto más trabajada esté la cuenta y su contenido, más interés despertará en los usuarios con los que interactúa y por ende la tasa de conversión será mayor.

1.4 Estructura

El presente TFG se ha estructurado como se explica a continuación:

En el primer capítulo, tal y como se ha podido observar, se explica la motivación que ha llevado al desarrollo de este proyecto, los objetivos que se han perseguido y el impacto que tendrá el producto resultante en el usuario que lo utilice.

En el segundo capítulo se evalúa la idea de negocio estudiando su viabilidad en el mercado mediante el análisis de la competencia existente, análisis DAFO, haciendo una proyección de ingresos y gastos y completando el modelo Lean Canvas. El objetivo principal es comprobar a priori que es viable el desarrollo del proyecto y que puede lograr posicionarse en el mercado.



El tercer capítulo expone la metodología utilizada en el desarrollo del proyecto y las herramientas que se han utilizado a lo largo del mismo.

El cuarto capítulo muestra el resultado del primer Mínimo Producto Viable (MVP) comentando sus principales características y también detalla la cronología del proyecto.

El quinto capítulo expone toda la etapa de diseño previa al desarrollo del producto, que engloba desde el diseño de la interfaz gráfica al modelo de datos pasando por la arquitectura software.

El sexto capítulo corresponde a la etapa de implementación. En él, se detalla la implementación de la interfaz gráfica, el front-end, el servidor del bot y la integración con la pasarela de pago. Además, se muestran los principales desafíos técnicos con los que se ha tenido que lidiar durante esta fase de desarrollo.

En el séptimo capítulo se muestra la etapa de validación. Se ha elegido a un grupo de usuarios para que prueben el producto y se les ha dado un cuestionario para que opinen sobre él. Posteriormente se han extraído conclusiones en base a las respuestas de los usuarios.

En el octavo capítulo se exponen las conclusiones a las que se ha llegado tras finalizar el presente TFG.

Por último, en el noveno capítulo se muestra la bibliografía que se ha consultado.

2. Evaluación de la idea de negocio

En primer lugar, se va a estudiar la viabilidad de la idea de negocio y a trazar el plan de empresa. Con este estudio se pretende analizar si la idea aporta valor al mercado y valorar si es factible su desarrollo. Para ello nos serviremos de algunas herramientas ampliamente utilizadas en el ámbito de la planificación empresarial como el Lean Canvas o el análisis de la competencia.

2.1 Estudio de mercado

En el estudio de mercado se han definido los clientes objetivo, esto es importante para poder enfocar el producto hacia ese nicho de mercado. Se han estudiado los productos competidores y sus principales características para generar una visión del mercado en el que se va a situar el producto. Además, se ha hecho un análisis DAFO que permite optimizar la estrategia que se va a tomar. A continuación, se ha definido una proyección de ingresos y gastos para confirmar la viabilidad económica del proyecto. También se ha completado el modelo Lean Canvas para analizar el modelo de negocio. Por último, se han extraído conclusiones en base al estudio realizado.

2.1.1 Clientes objetivo

El cliente objetivo es toda aquella persona o negocio que quiera hacer crecer su comunidad de seguidores en Instagram. El servicio encaja tanto con *influencers* como con empresas que quieran aumentar el tráfico en su cuenta. En la fase inicial de lanzamiento, el cliente objetivo es hispanohablante, ya que se va a ofrecer la web en español. Posteriormente se ampliará con más idiomas.

Instagram cuenta con más de 1000 millones de usuarios activos por mes de los cuales 130 millones entran al menos una vez por mes en posts de productos. Además, el 83% de los usuarios han descubierto nuevos productos o servicios a través de la red social [1].

Con estos datos, podemos afirmar sin ninguna duda que Instagram es un gran escaparate para los negocios, por lo que el número de usuarios potenciales de nuestro servicio es enorme y está en constante crecimiento.

2.1.2 Productos competidores

El estudio de las características de los servicios que ofrece la competencia nos ayudará a determinar si nuestro proyecto aporta un valor real al mercado, y tiene posibilidades de acaparar parte del nicho de mercado existente.

Se han utilizado durante varios días los servicios que ofrecen los competidores con mayor cuota en el mercado: Social Captain, SocialSteeze, Instazood, Inflou, Instaboom y Boostagram. Y se han anotado las características de cada uno de ellos.

En la Tabla 1 se muestran todas las características desglosadas por competidor.

COMPETIDOR	Social Captain	SocialSteeze	Instazood	Inflou	Instaboom	Boostgram
CONFIGURACIÓN						
Mensaje directo automático	✓	—	✓ ¹	—	—	—
Follow automático	✓	✓	✓	✓	✓	✓
Me gusta automático	✓	✓	✓	✓	✓	✓
Comentarios automáticos	✓	✓	✓	✓	✓	—
Unfollow automático	✓	✓	✓	✓	✓	✓
Elección de genero/edad/tipo de post (foto, video, álbum, etc.)	✓	—	—	—	—	—
Elección de rango (Min/Max following/followers, Min/Max me gustas/comentarios en una foto)	✓	—	—	—	—	⊖ ²
Interactuar por usuarios	✓	✓	✓	✓	✓	✓
Interactuar por hashtags	✓	✓	✓	✓	✓	✓
Interactuar por ubicación	✓	✓	✓	✓	✓	✓
Comentarios personalizados	✓	✓	✓ ³	✓	✓	—

¹ Permite hacer difusión.

² Permite uso solo del rango de me gustas.

³ Generador de comentarios con expresiones regulares.

Configuración ciega (Usuario elige los parámetros, pero no controla ni cuándo ni cuántas interacciones se producen)	✓ ⁴	✓	✓ ⁵	✓	✓	⊖ ⁶
ESTADÍSTICAS						
Evolución del número de seguidores	✓	⊖	⊖	✓	⊖	✓
Últimas acciones del bot	✓	⊖	⊖	✓	⊖	⊖ ⁷
Muestra cuántos seguidores ha generado cada tipo de configuración (Hashtags, ubicaciones...)	✓	⊖	✓	⊖	⊖	⊖
OTROS						
Idiomas						
Programar posts	⊖	⊖	⊖	⊖	⊖	✓
Reportes por e-mail	⊖	⊖	⊖	⊖	⊖	✓
Venta de me gustas	⊖	⊖	✓	⊖	⊖	⊖
Blog	✓	⊖	✓ ⁸	⊖	✓ ⁹	⊖
Tutoriales, centro de ayuda, etc.	✓	✓	✓	⊖ ¹⁰	⊖	✓
Estado del sistema (si está caído, etc.)	✓	⊖	⊖	⊖	⊖	⊖

⁴ Con diferentes velocidades (baja, media, alta, turbo)

⁵ Aunque se pueden elegir los límites diarios, en teoría el usuario controla las interacciones al día

⁶ El usuario elige la configuración concreta con la que funcionara el bot (tags, seguidores de otra cuenta o ubicaciones). Además, puede guardar varias configuraciones, pero solo puede elegir una a la vez.

⁷ Sólo el número de acciones.

⁸ Para atraer clientes, solo tiene información general de Instagram y noticias.

⁹ En él dan trucos y consejos para crecer en Instagram.

¹⁰ Preguntas frecuentes.

Inteligencia artificial	✓ ¹¹	—	—	—	✓	—
Prueba gratuita	✓ 12 horas	—	✓ 3 días	✓ 7 días	—	✓ 3 días
Programa referidos	✓ 30% Mes	✓ 30% Mes	✓ 15% primer pago y 10% Mes	✓ 20% Mes	—	✓ ¹²
Permite introducción de proxy	—	—	✓ ¹³	—	—	—

Tabla 1: Competidores

2.2 Análisis DAFO

A continuación, se presenta el análisis DAFO. Éste es un tipo de análisis empresarial que se basa en el entorno (interno y externo) en el que se establecen las debilidades, fortalezas, amenazas y oportunidades de negocio con el objeto de mejorar la estrategia en el mercado [2].

- **Debilidades:** Se cuenta con recursos de inversión escasos y con un presupuesto limitado.
- **Fortalezas:** Implementación de funciones que no tiene la competencia como es el control total del comportamiento del bot al poder configurar muchos más parámetros y poder crear varias configuraciones y asignarlas a los días de la semana que se deseen. Además, la aplicación ofrece una interfaz sencilla e intuitiva.
- **Amenazas:** Instagram puede implementar mecanismos de detección de bots o modificar su web, lo que afectaría al funcionamiento del servicio pudiendo llegar incluso a inutilizarlo completamente.
- **Oportunidades:** Se pueden ofrecer precios más bajos que la competencia y ganar así cuota de mercado.

¹¹ Sólo en el modo turbo.

¹² No se especifica el porcentaje de comisión.

¹³ Permite introducir uno propio o que se ejecute un software en el pc del usuario y que éste haga de proxy.

2.3 Modelo de negocio y proyección de ingresos y gastos

El modelo de negocio escogido es el de pago por suscripción. El usuario se suscribe al servicio para poder usarlo. Cuando finalice el periodo de suscripción se renovará automáticamente hasta que el usuario cancele la renovación.

Se ha elegido el modelo de monetización por suscripción debido a que por la naturaleza del servicio su uso es prolongado en el tiempo y por tanto un modelo de pago recurrente como es la suscripción es el que mejor encaja.

Se ha hecho una estimación de ingresos y gastos en Excel para comprobar la rentabilidad del proyecto. Se muestra en la Figura 1.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	
1	Número de Trimestre																				
2	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
3	Tipos de suscripción acumuladas																				
4	Año 1 / T1	Año 1 / T2	Año 1 / T3	Año 1 / T4	Año 2 / T1	Año 2 / T2	Año 2 / T3	Año 2 / T4	Año 3 / T1	Año 3 / T2	Año 3 / T3	Año 3 / T4	Año 4 / T1	Año 4 / T2	Año 4 / T3	Año 4 / T4	Año 5 / T1	Año 5 / T2	Año 5 / T3	Año 5 / T4	
5	Suscripciones nuevas	0	70	105	138	236	354	532	797	1037	1140	1254	1380	1518	1669	1836	2020	2222	2444	2689	2957
6	Suscripciones renovadas después de un trimestre	0	0	35	70	114	175	265	398	598	817	979	1116	1248	1383	1526	1681	1851	2036	2240	2464
7	Total de suscripciones nuevas y renovadas	0	70	140	228	350	529	796	1195	1634	1957	2233	2496	2766	3052	3362	3701	4072	4480	4929	5422
8	Ingresos Trimestrales																				
10	(Suscripciones nuevas + Renovaciones) * Precio	0 €	5.250 €	10.500 €	17.063 €	26.250 €	39.703 €	59.719 €	89.660 €	122.571 €	146.801 €	167.467 €	187.207 €	207.424 €	228.915 €	252.180 €	277.585 €	305.437 €	336.028 €	369.654 €	406.631 €
12	Total Ingresos	0 €	5.250 €	10.500 €	17.063 €	26.250 €	39.703 €	59.719 €	89.660 €	122.571 €	146.801 €	167.467 €	187.207 €	207.424 €	228.915 €	252.180 €	277.585 €	305.437 €	336.028 €	369.654 €	406.631 €
13	Gastos Anuales																				
15	Infraestructura Cloud	1.000 €	1.000 €	1.000 €	1.000 €	1.000 €	1.000 €	1.000 €	1.000 €	2.000 €	2.000 €	2.000 €	2.000 €	2.000 €	2.000 €	3.000 €	3.000 €	3.000 €	3.000 €	3.000 €	3.000 €
16	Ordenadores e impresoras	4.000 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	2.000 €	4.000 €	4.000 €	4.000 €	4.000 €	2.000 €	2.000 €	2.000 €	2.000 €	0 €	0 €
17	Muebles oficina e instalaciones	3.000 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	2.000 €	0 €	2.000 €	0 €	2.000 €	0 €	2.000 €	0 €	0 €	0 €
18	Marketing	5.000 €	0 €	5.000 €	5.000 €	5.000 €	5.000 €	5.000 €	5.000 €	7.000 €	7.000 €	7.000 €	7.000 €	7.000 €	7.000 €	9.000 €	9.000 €	9.000 €	9.000 €	9.000 €	9.000 €
19	Alquiler oficina	3.000 €	3.000 €	3.000 €	3.000 €	3.000 €	3.000 €	3.000 €	3.000 €	3.000 €	3.000 €	3.000 €	4.000 €	4.000 €	4.000 €	4.000 €	4.000 €	4.000 €	4.000 €	4.000 €	4.000 €
20	Internet, electricidad, agua, teléfono, etc.	1.500 €	1.500 €	1.500 €	1.500 €	1.500 €	1.500 €	1.500 €	1.500 €	1.500 €	1.500 €	1.500 €	2.000 €	2.000 €	2.000 €	2.000 €	2.000 €	2.000 €	2.000 €	2.000 €	2.000 €
21	Gestoría	600 €	600 €	600 €	600 €	600 €	600 €	600 €	600 €	600 €	600 €	600 €	800 €	800 €	800 €	800 €	800 €	800 €	800 €	800 €	800 €
22	CEO - Director Ejecutivo	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €
23	CTO - Director Técnico	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €
24	CMO - Director de Marketing	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €	12.000 €
25	CFO - Director Financiero	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €
26	Desarrolladores Senior	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	10.000 €	20.000 €	20.000 €	20.000 €	20.000 €	30.000 €	30.000 €	30.000 €	30.000 €	44.000 €
27	Técnicos de soporte	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	7.000 €	7.000 €	7.000 €	7.000 €	14.000 €	14.000 €	14.000 €	14.000 €	21.000 €
28	Administrativo	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	0 €	6.000 €	6.000 €	6.000 €	6.000 €	6.000 €	6.000 €	6.000 €	6.000 €	6.000 €	6.000 €
29	Total Gastos	48.100 €	36.100 €	41.100 €	41.100 €	41.100 €	41.100 €	41.100 €	41.100 €	41.100 €	44.100 €	52.100 €	85.100 €	84.800 €	98.800 €	94.800 €	108.800 €	118.800 €	154.800 €	155.800 €	151.800 €
30	Resultado Trimestral	-48.100 €	-30.850 €	-30.600 €	-24.038 €	-14.850 €	-1.397 €	18.619 €	48.560 €	78.471 €	94.701 €	82.367 €	102.407 €	108.624 €	134.115 €	143.380 €	158.785 €	150.637 €	180.228 €	217.854 €	254.831 €
31	Resultado Trimestral Acumulado	-48.100 €	-78.950 €	-109.550 €	-133.588 €	-148.438 €	-149.834 €	-131.216 €	-82.655 €	-4.194 €	90.516 €	172.883 €	275.290 €	383.914 €	518.029 €	661.409 €	820.194 €	970.832 €	1.151.060 €	1.368.914 €	1.623.745 €
32	Personal contratado																				
39	Año 1 / T1	Año 1 / T2	Año 1 / T3	Año 1 / T4	Año 2 / T1	Año 2 / T2	Año 2 / T3	Año 2 / T4	Año 3 / T1	Año 3 / T2	Año 3 / T3	Año 3 / T4	Año 4 / T1	Año 4 / T2	Año 4 / T3	Año 4 / T4	Año 5 / T1	Año 5 / T2	Año 5 / T3	Año 5 / T4	
40	CEO - Director Ejecutivo	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
41	CTO - Director Técnico	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	
42	CMO - Director de Marketing	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
43	CFO - Director Financiero	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	
44	Desarrolladores Senior	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	3	3	3	3	4
45	Técnicos de soporte	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	3	3	3	3	4
46	Administrativo	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	
47	Total empleados	4	4	4	4	4	4	4	4	4	5	6	8	8	10	10	10	11	13	13	
48	Inversión mínima solicitada																				
50	Inversión mínima solicitada	149.834 €																			
51	Punto de equilibrio alcanzado en trimestre	7																			
52	Recuperación de inversión en trimestre	10																			

Figura 1: Proyección de ingresos y gastos en Excel

Además, se ha representado los resultados de la estimación en gráficas para observar de una forma más visual la proyección.



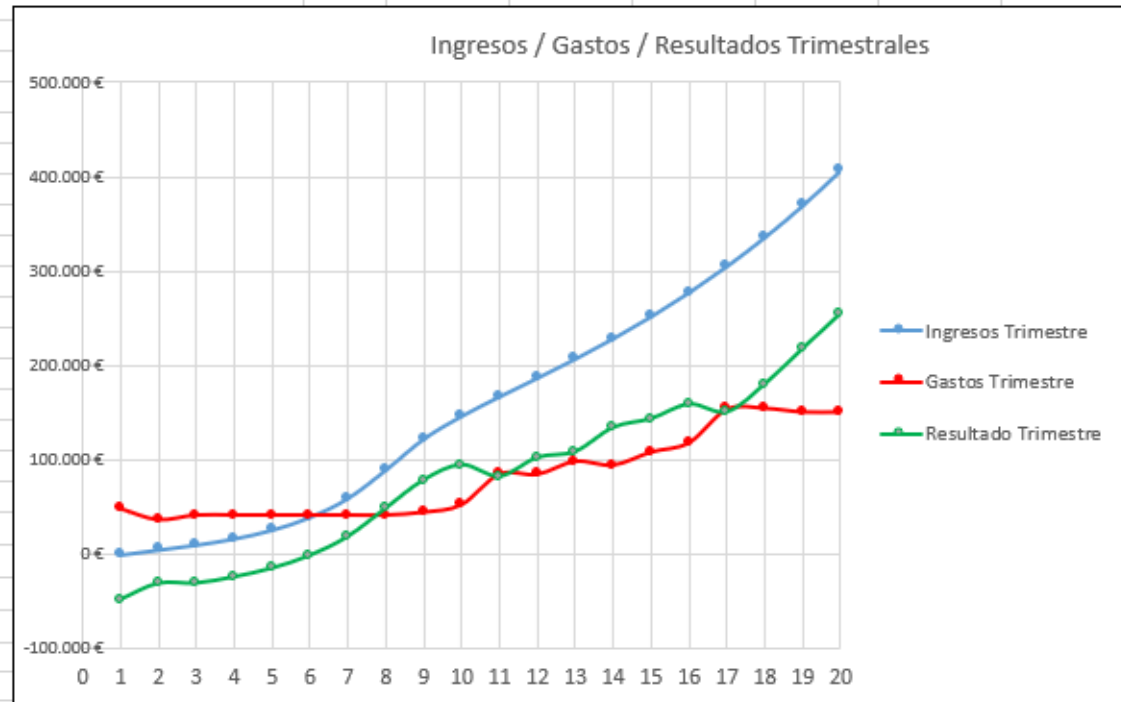


Figura 2: Gráfica Ingresos/Gastos/Resultados Trimestrales

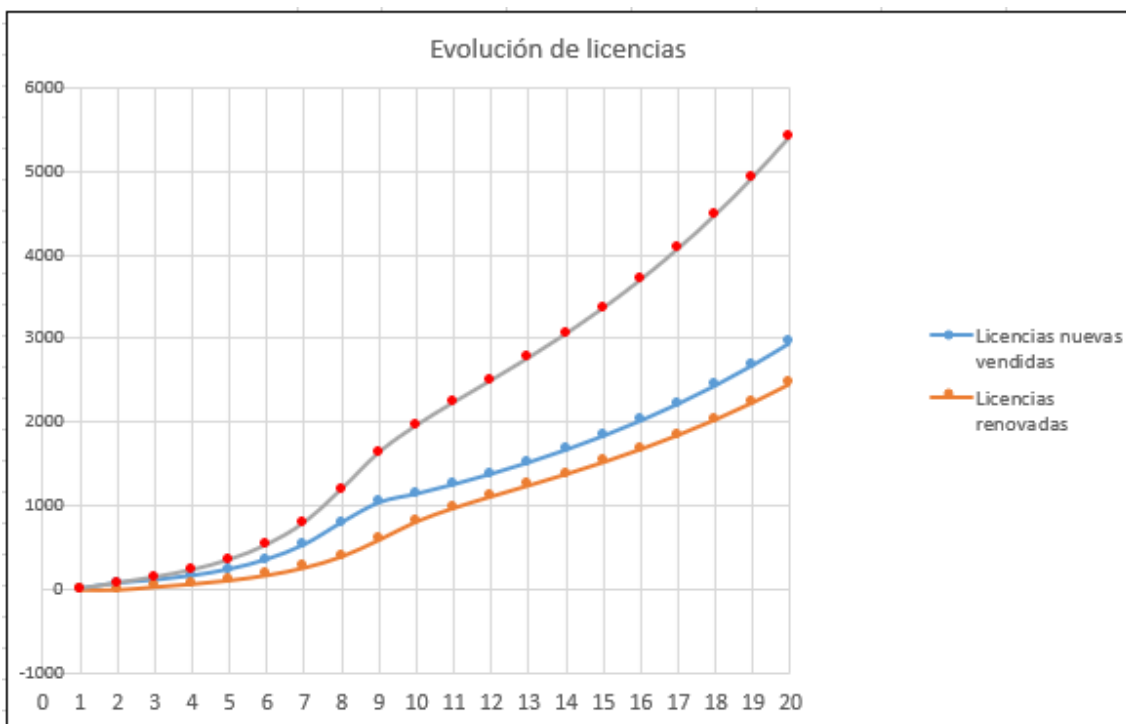


Figura 3: Gráfica Evolución de licencias

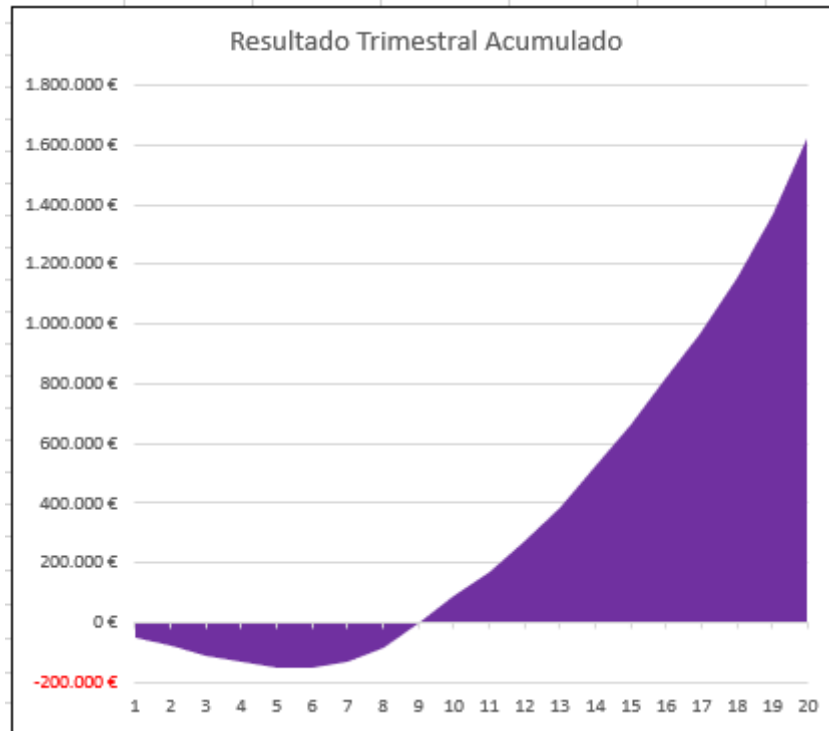


Figura 4: Gráfica Resultado Trimestral Acumulado

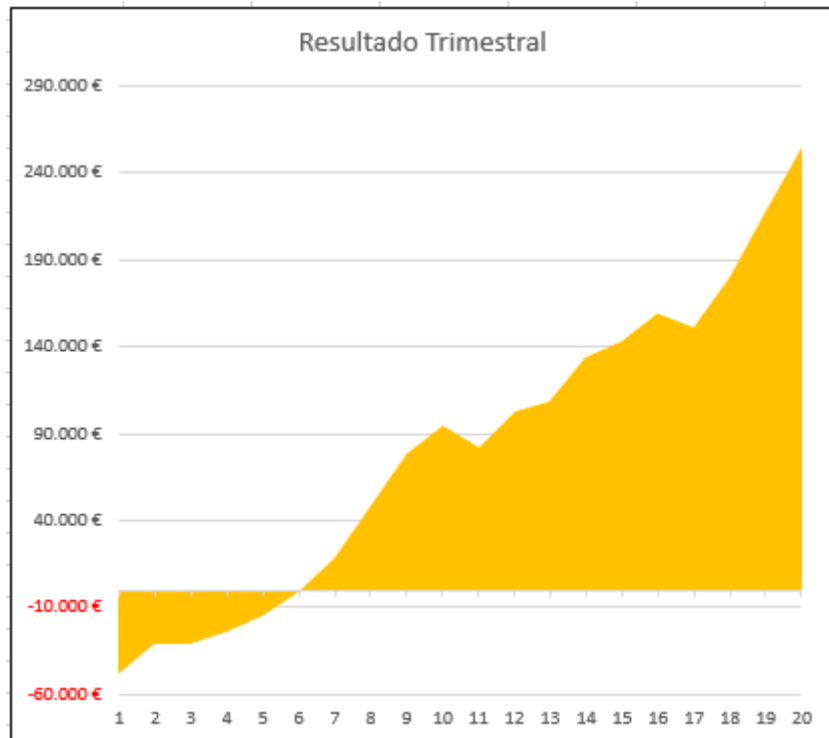


Figura 5: Gráfica Resultado Trimestral

2.4 Lean Canvas

“El Lean Canvas es una herramienta de visualización de modelos de negocio pensada para empresas incipientes. Se enfoca en la metodología lean startup, centrada en generar nuevas ideas y llevarlas a prototipado para verificar si el mercado está en búsqueda de las soluciones que usted está proponiendo” [3].

En la Figura 6 se muestran los componentes del modelo.

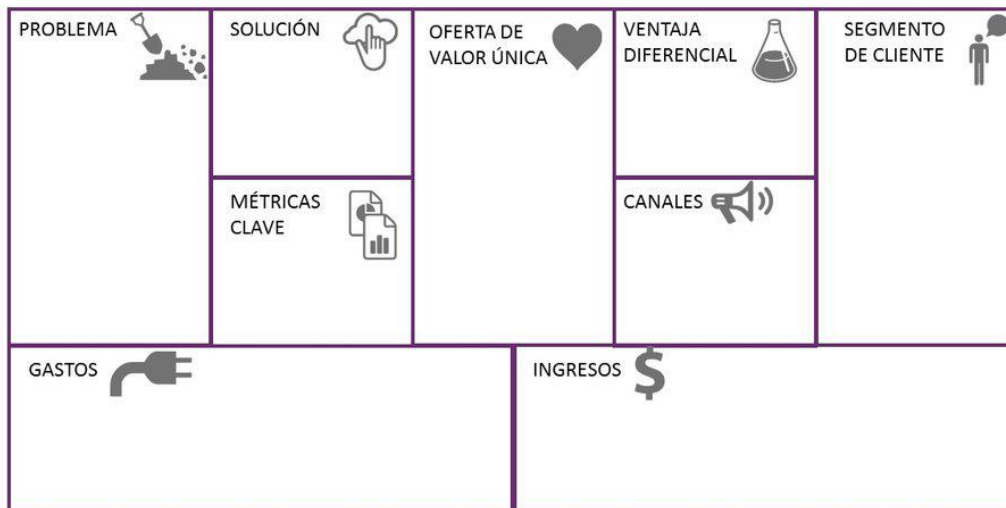


Figura 6: Lienzo Lean Canvas

A continuación, se presenta el resultado obtenido tras su realización, donde se detalla cada una de las casillas definidas en la plantilla.

- **Problema**

Resuelve la necesidad de una empresa o individuo que posee una cuenta en Instagram de llegar a un mayor número de usuarios/potenciales clientes y aumentar su credibilidad frente a ellos.

Como competidores, existen muchas webs que ofrecen un servicio de bot: SocialCaptain, Instaboom, Ingramer, Inflou, Gramista, Boostgram, Instazood, Combin(Programa de PC), ViralSoda, Bigbangram, Instamacro, Instavast, Archie.co, Instagrambotfollower.com, Ploxia, Argenbot, SocialSteeze, Kicksta, Upleap, RiseSocial, MagicSocial, Instato.io, Socialupgrade.com, MegaFollow, FalconSocial.co, Instavast .

Características de la competencia necesarias en nuestro producto: Follows automáticos, me gustas automáticos, comentarios automáticos, unfollows automáticos, interactuar por usuarios, interactuar por hashtags, interactuar por ubicación, comentarios, evolución de seguidores, acciones del bot, asignación de asesor profesional.

- **Solución**

La principal revolución es la completa configuración por parte del usuario (otorgándole así poder sobre el comportamiento del bot). Podrá elegir que se hace cada día con un set de configuraciones que irá creando y tendrá control total de la configuración. Podrían incorporarse sets predefinidos según áreas (deportes, música, etc).

Un aspecto profesional de la web además de un precio competitivo (alrededor de 25€/mes), muchas interacciones al día (unos 600 follows), una IP para cada 2 cuentas o para cada una, interfaz amigable y sencilla a la par que completa. Será una web muy cercana al cliente.

Además, se contempla ofrecer la posibilidad de tener un asesor personal a cambio de un coste adicional. La aplicación propuesta, focalizada en el mercado español puede hacerse un buen hueco, el inglés está muy saturado y hay buenas opciones (SocialCaptain) pero el español no es tan rico en opciones.

- **Métricas**

Las métricas que indicarán el crecimiento del negocio serán el número de clientes que tengan una suscripción vigente. También será importante a qué plan están suscritos dichos clientes, ya que habrá varios con diferente coste.

- **Proposición de valor**

Se propone la siguiente lista de proposiciones de valor de nuestro servicio:

- Sé una referencia en Instagram
- Revoluciona tu alcance y visibilidad
- Empieza a crecer en Instagram

- **Ventaja competitiva**

La competencia desconoce el código del bot que estamos usando.

- **Canales**

Los clientes conocerán el producto a través de Google Ads, de nuestra presencia en redes (nuestro perfil en Instagram), del programa de embajadores y referidos, del SEO y de posibles colaboraciones con otras webs relacionadas con redes sociales, publicidad en medios de la UPV, posible compra de publicidad en Instagram y contacto con medios de comunicación.

- **Clientes**

Se ha identificado tres tipos diferentes de clientes:

- **Empresas:** Startups, empresas con pocos seguidores en Instagram que quieren ganar presencia, básicamente pequeñas o medianas empresas con deseos de darse a conocer en las redes sociales.



- Particulares: Músicos, entrenadores personales, usuarios que desean aumentar su popularidad mediante el aumento de seguidores, *influencers*, etc.
- *Early adopters*: Actuales clientes, amigos, compañeros de trabajo, etc.
- **Costos**

A continuación, se detallan los costos agrupados por concepto:

- Servidores (tanto servidor web como el que ejecuta el bot)
 - Proxys (una IP por cada 2/3 usuarios).
 - Publicidad (Ads, Instagram, etc.).
 - Personal de marketing y soporte técnico para el cliente.
 - Perfiles técnicos para el desarrollo y mantenimiento de la web como futuras mejoras en el bot.
- **Ingresos**

Los ingresos corresponden únicamente al cobro de las suscripciones mensuales abonadas por los clientes.

2.5 Conclusiones de la evaluación

Una vez concluido el estudio de mercado, se puede observar que el producto posee distintivos que ofrecen ventajas competitivas respecto a los productos similares que hay en el mercado. Los puntos fuertes de la aplicación son el amplio abanico de configuraciones que se ofrecen para el bot y el precio, que es inferior a la media.

Además, la proyección económica resulta muy atractiva, empezando a dar beneficios en el segundo trimestre del segundo año y arrojando un saldo positivo acumulado a partir del segundo trimestre del tercer año.

Por tanto, tras comprobar que el proyecto es viable tanto económica como competitivamente se procede a su desarrollo.

3. Metodología y tecnologías aplicadas

En esta sección se expone la metodología seguida durante el desarrollo del producto además de las principales tecnologías y los lenguajes aplicados.

3.1 Metodología seguida

Para la implementación del producto se ha seguido un modelo evolutivo iterativo basado en la lista de características que tendrá el producto final (Backlog).

En primer lugar, se ha diseñado el Backlog. Partiendo de él, se ha hecho una selección de características escogiendo solo las más importantes, que serán las que formen parte de la primera iteración del desarrollo.

Una vez completado el desarrollo de la primera iteración, se procederá a validar el resultado obtenido con usuarios que probarán el producto. Se obtendrá feedback de los usuarios y se harán cambios en el Backlog si así se estima oportuno. A continuación, se repetirá este proceso seleccionando otras características pendientes y procediendo a su implementación y validación. Se puede observar el ciclo de desarrollo expuesto en la Figura 7.

Se deberán realizar tantas iteraciones como sean necesarias hasta obtener un producto completo que contenga todas las características que se han especificado en el Backlog.

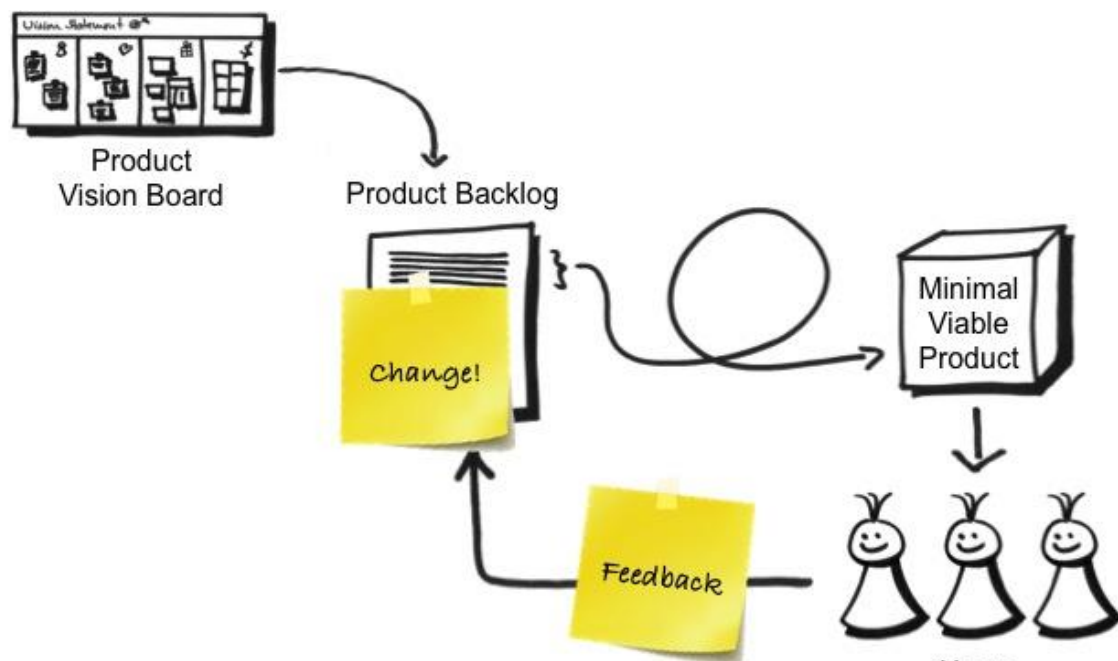


Figura 7: Ciclo de desarrollo del producto

3.2 Tecnologías y lenguajes utilizados

A continuación, se van a presentar brevemente las tecnologías y lenguajes utilizados durante el desarrollo del presente TFG.

3.2.1 PhpStorm

PhpStorm ha sido el IDE escogido para todo el desarrollo del front-end (HTML, CSS y Javascript) y back-end (PHP) de la web. Es una herramienta muy potente y considerado uno de los mejores si no el mejor en su campo.

Ofrece potentes funciones de autocompletado para facilitar la programación y la detección de errores antes de ejecutar el código.

Además, una función muy interesante que ofrece este IDE es la de poder editar directamente los ficheros que están ubicados en un servidor externo. Esto se consigue mediante una conexión SSH con el servidor. Esta función es muy importante para el proyecto porque ha permitido desarrollar directamente en los VPS en lugar de hacerlo primero en local.

3.2.2 Adobe Illustrator

Adobe Illustrator es el software de gráficos vectoriales estándar del sector. Se ha utilizado para modificar algunas imágenes vectoriales que se han incluido en la web. Las imágenes se han descargado de Freepik.com pero se requería de su modificación ya que la mayoría venían con fondos y elementos innecesarios que se han eliminado.

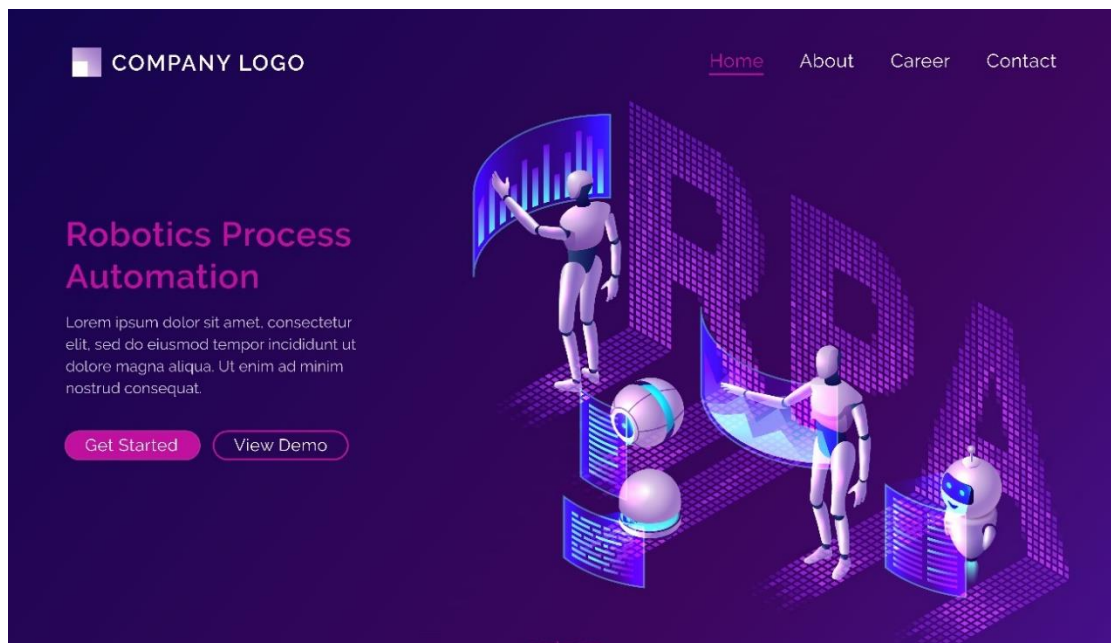


Figura 8: Vector antes de ser modificado

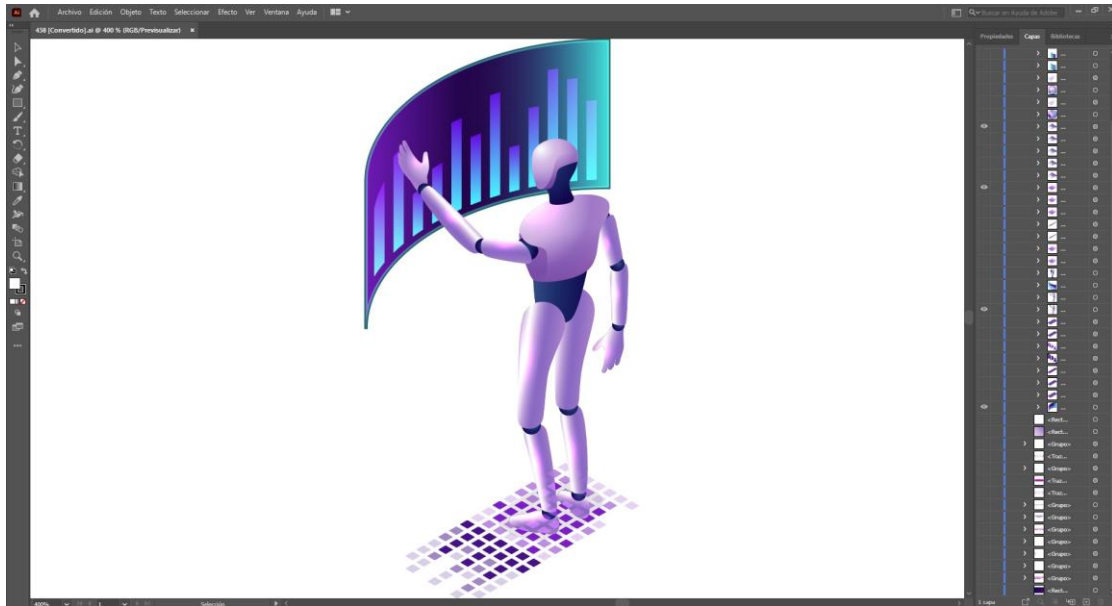


Figura 9: Vector después de ser modificado

3.2.3 Adobe Photoshop

Se ha utilizado Adobe Photoshop para la creación del logo. En primer lugar, se buscó una imagen de un robot en Freepik.com que diese personalidad a la página. Se editó con Adobe Illustrator para eliminar elementos innecesarios. Una vez completada la edición, se buscó un tipo de letra en Dafont.com adecuado al estilo de la web y se le aplicaron efectos. Finalmente se combinó el logo del robot con el texto para culminar la creación del logo.



Figura 10: Imagen del logo antes de ser modificada



Figura 11: Logo final

4. Análisis de requisitos

El primer paso para el desarrollo es establecer el Backlog. El Backlog es la lista de todas las características que tendrá nuestro producto.

A continuación, se listan todas las características que forman parte del Backlog:

- Dashboard
- Aceptar pagos con tarjeta de crédito
- Página configuración del bot
- Interacción por hashtags
- Landing Page
- Unfollow
- Interacción por ubicaciones
- Interacción por seguidores de otro usuario
- Centro de ayuda
- Blog
- Multi-idioma

Para el desarrollo se ha elegido un modelo evolutivo basado en el Mínimo Producto Viable (MVP). Un producto con una funcionalidad básica e incompleta, pero útil para poder hacer estudios con usuarios que prueben el producto y den feedback. Para ello se realizarán varios MVP, cada uno partiendo del anterior y añadiéndole más características hasta llegar a la versión completa deseada que tenga todas las características del Backlog.

Debido a que el primer MVP (MVP I) que se va a desarrollar tiene una gran carga de trabajo asociada, para el presente TFG sólo se llevará a cabo el desarrollo de esta primera iteración.

A continuación, se describe cómo se ha organizado, desarrollado y evaluado la lista de características que forman en Backlog en el MVP.

4.1 MVP I

Para el primer MVP se han definido los siguientes requisitos:

- Dashboard
- Aceptar pagos con tarjeta de crédito
- Página configuración del bot
- Interacción por hashtags
- Landing Page
- Unfollow

Para este MVP se ha decidido incluir un gran número de características, para que los *early adopters* puedan probar un producto lo más cercano posible al producto final.

Para desarrollar el proyecto se ha utilizado Instapy¹⁴, un bot open source para Instagram escrito en Python que simula el comportamiento de un usuario real y automatiza las interacciones para conseguir un aumento en el número de seguidores e interacciones que tiene una cuenta en la red social. Debido a que su instalación requiere conocimientos técnicos de programación y administración de sistemas Linux además de un servidor, el producto que se va a desarrollar es un portal donde el usuario se registra y tras pagar una suscripción tiene acceso al servicio de interacciones automatizadas, que está alojado en un servidor propio. Mediante una interfaz gráfica puede configurarlo de forma sencilla y comprobar su efectividad mediante datos estadísticos y gráficas.

4.2 Cronología del proyecto

A continuación, en la Figura 12 se muestra un cronograma donde aparecen los principales hitos del proyecto junto a la fecha en la que se llevaron a cabo y una breve descripción de cada uno de ellos.

¹⁴ <https://github.com/timgrossmann/InstaPy>

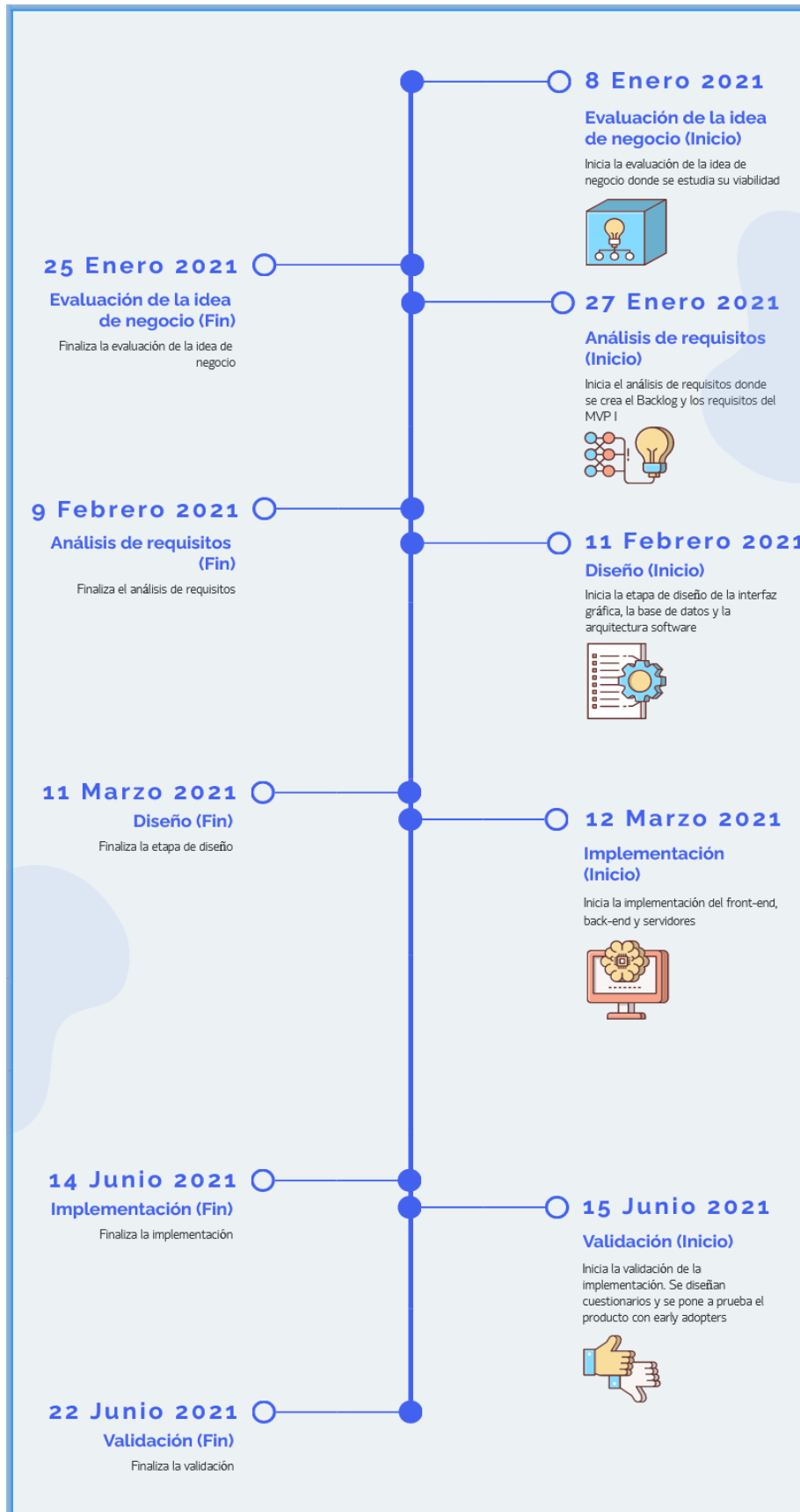


Figura 10: Cronograma

5. Diseño y arquitectura del sistema

En esta sección se ha expone toda la etapa de diseño. Se engloba aquí todo el diseño de la interfaz gráfica de la web, el diseño del modelo de datos para la base de datos y la arquitectura software compuesta por los servidores de la web y el bot.

5.1 Diseño de la interfaz gráfica

El primer paso del desarrollo de la web ha sido hacer *wireframes* a modo de bocetos informales en papel de las principales páginas de la web (panel de inicio, dashboard/centro de operaciones, panel de configuración, etc.).

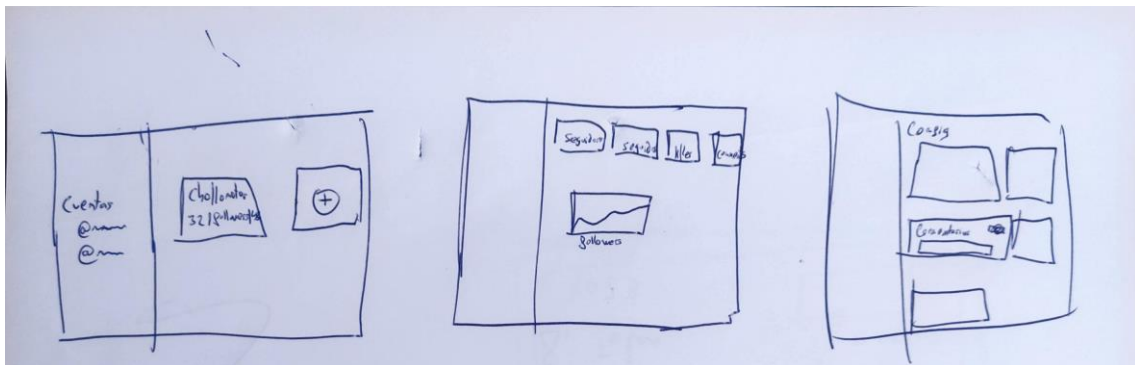


Figura 11: Wireframe 1

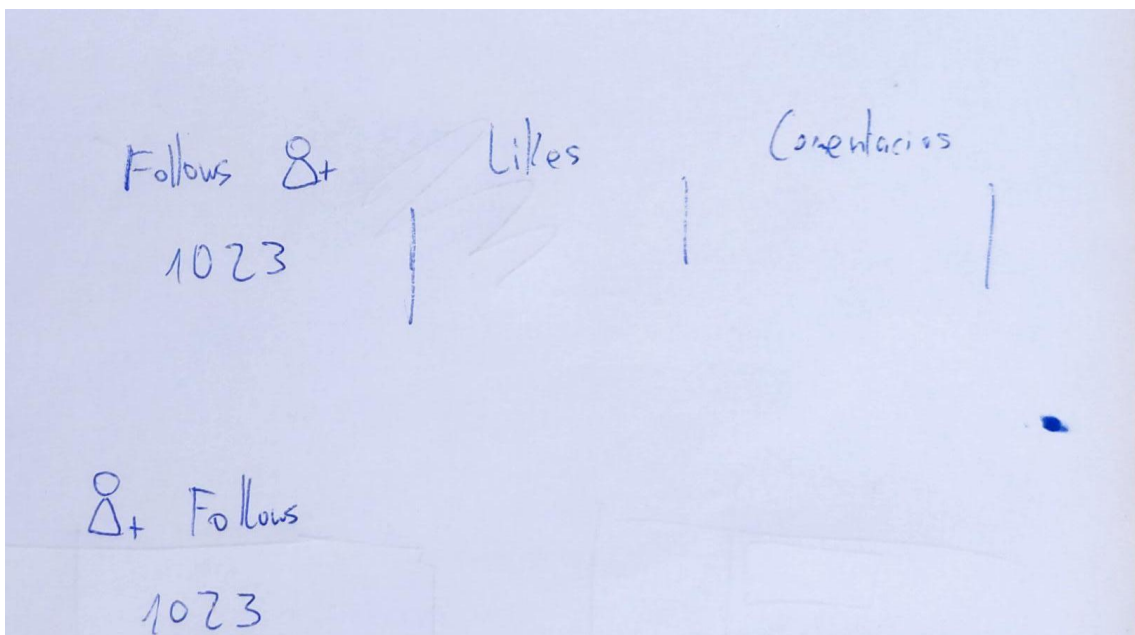


Figura 12: Wireframe 2

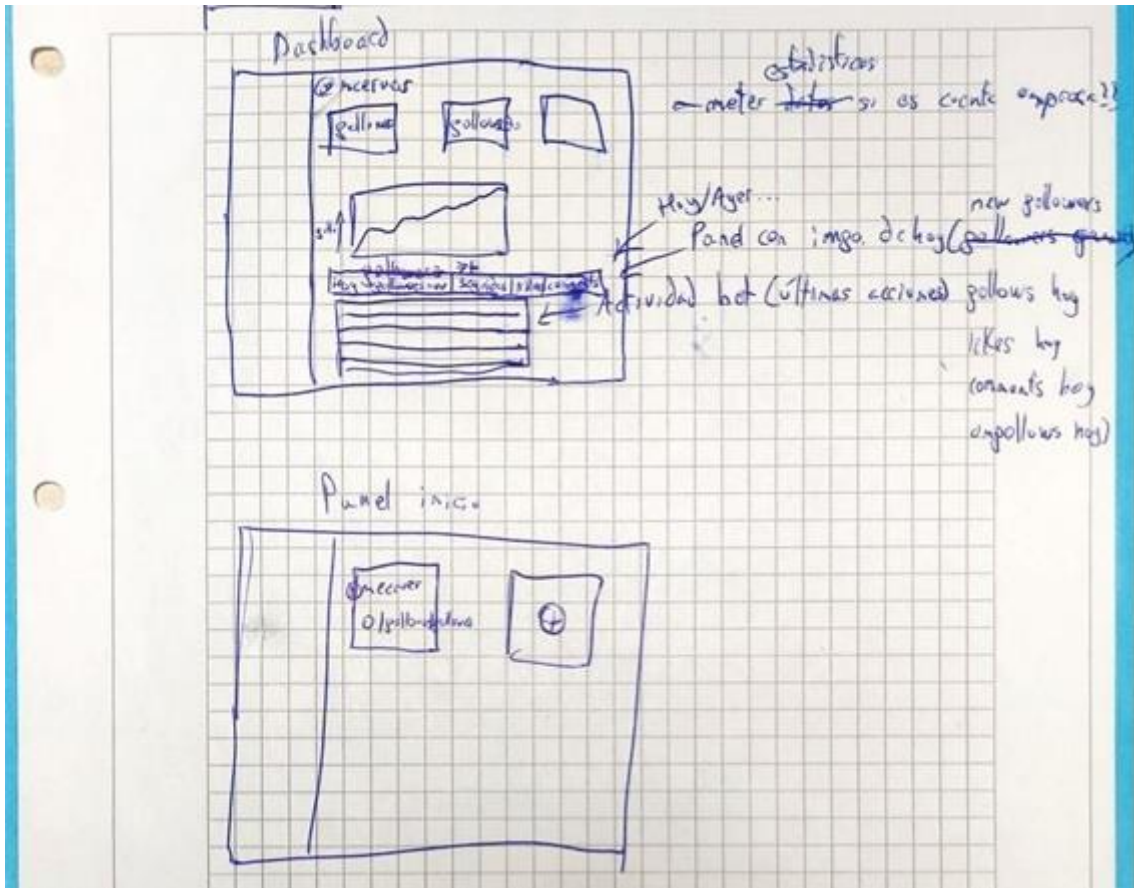


Figura 13: Wireframe 3

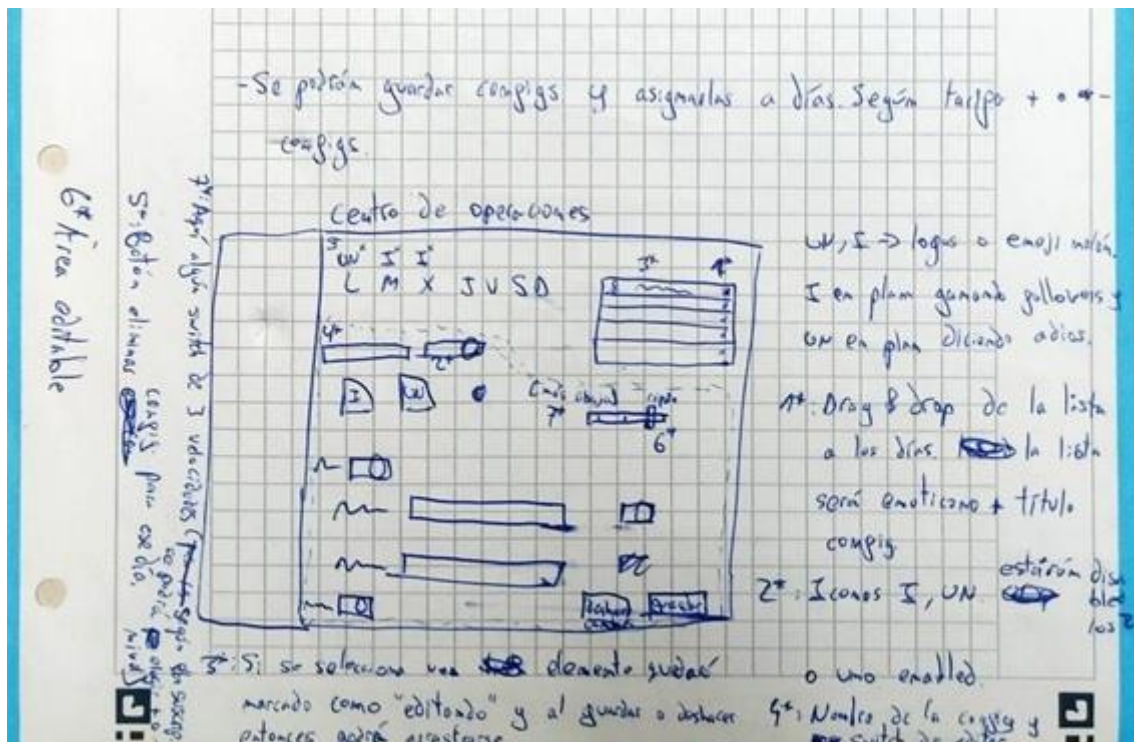


Figura 14: Wireframe 4

A la entrada, el área editable estará vacía, y la lista habrá solo 1 "+" para añadir una nueva configuración.

Figura 15: Wireframe 5

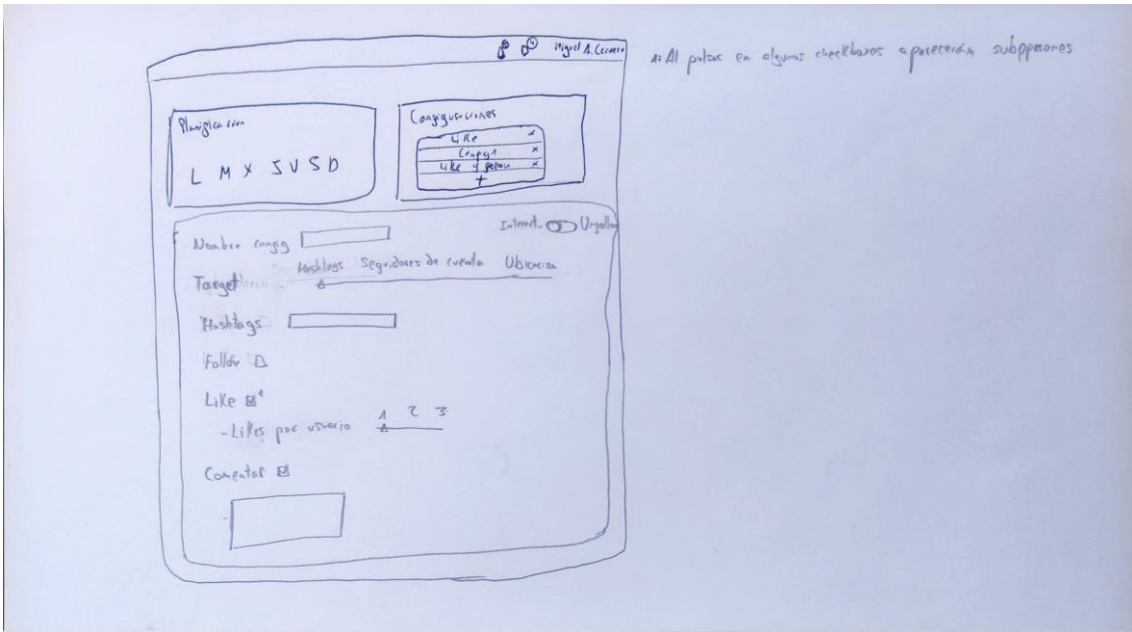


Figura 16: Wireframe 6

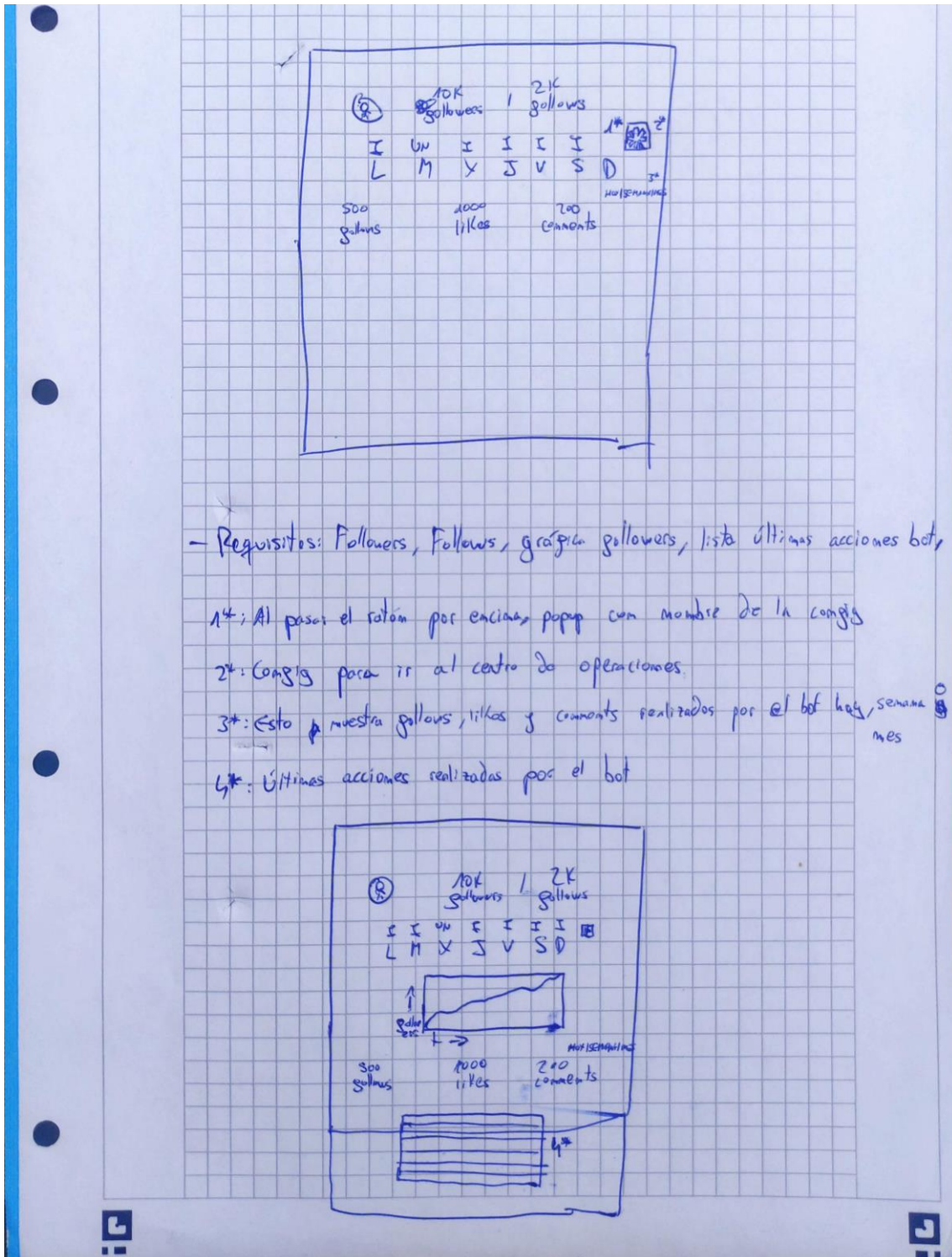


Figura 17: Wireframe 7

Aunque son bocetos hechos a mano alzada y sujetos a cambios, dan una idea muy clara de la base de cada página.

5.2 Modelo de datos

La base de datos MySQL está alojada en el servidor de la web. Para su diseño se ha hecho un estudio de los distintos objetos que necesita la aplicación web y sus propiedades.

Se ha utilizado la utilidad MySQL Workbench para la representación del modelo de datos de datos [4]. En la Figura 20 se muestra el modelo de datos de la aplicación como un diagrama entidad-relación.

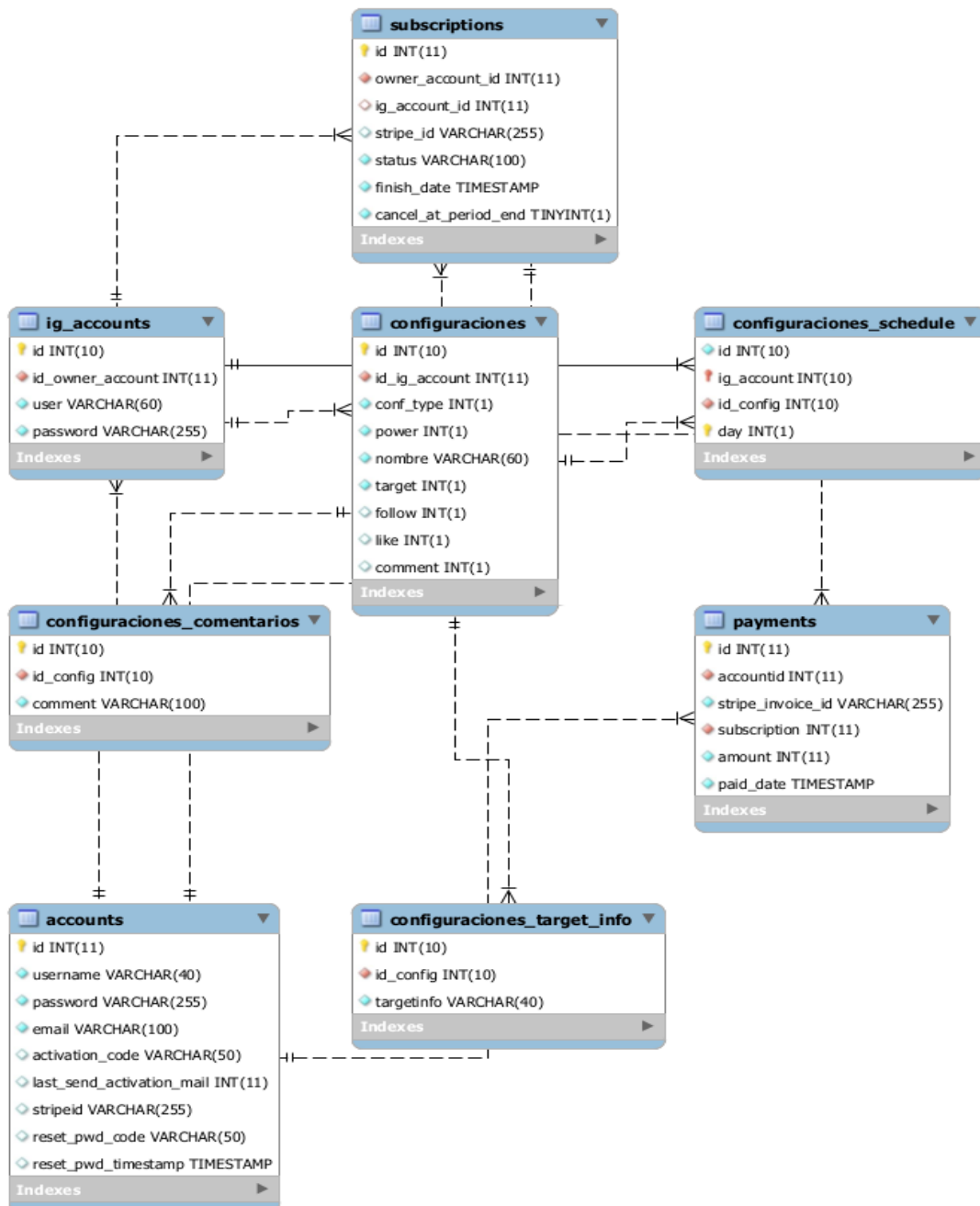


Figura 18: Modelo de datos de la BD

5.3 Arquitectura software

La necesidad de dos servidores era fundamental para poder dividir la carga de procesamiento. Además, aporta mayor seguridad al estar la web y el bot en entornos aislados.

Para empezar en un entorno de pruebas y ahorrar costes, tras comparar precios se ha optado por el alquiler de máquinas virtuales (VPS) las cuales ejercerán de servidores brindando el mismo servicio, aunque con menos potencia de cómputo que un servidor dedicado.

Por tanto, se contrató dos VPS a un proveedor. Una aloja la web, mientras que la otra aloja todas las instancias del bot (una por usuario de Instagram). Para ambas VPS se eligió CentOs como sistema operativo por ser uno de los más utilizados y con mayor soporte. También se registró el dominio de la web (triumfalo.es).

En las VPS se ha instalado fail2ban para bloquear varios intentos fallidos de login por SSH desde la misma IP. Esto se ha hecho porque los VPS de proveedores reconocidos son el objetivo de redes de bots que por fuerza bruta intentan conseguir acceso SSH. Se ha establecido un bloqueo permanente de acceso por SSH desde una IP si se falla 3 veces el login.

Esto se ha hecho porque se observó que cada VPS cada día tenía cientos o miles de intentos fallidos de login. Con esta medida se han reducido a unas decenas. La medida que se contempla una vez lanzado el producto a producción para erradicar este riesgo es permitir el acceso únicamente con clave RSA mediante la herramienta ssh-keygen.

También se ha creado un script en Bash en los VPS para que cada vez que se conecte un usuario por SSH pueda comprobar de un vistazo la carga de CPU, memoria ram y otros datos de relevancia respecto al estado del servidor.

```
miguel@miguel-DESKTOP:~/Projects$ ./connectWebTF6.sh
Last failed login: Sun Jun 13 21:08:36 CEST 2021 from 134.122.126.108 on ssh:notty
There were 2055 failed login attempts since the last successful login.
Last login: Wed Mar 3 00:11:57 2021 from 2.59.232.4

##### ESTADO DEL SISTEMA #####
Uptime: 336 days, 6:43

##### MEMORIA RAM #####
Memoria RAM total: 1,7G
% de uso: 42 %

##### CPU #####
Procesadores: 1
Carga CPU: 0,03, 0,05, 0,05

##### ALMACENAMIENTO #####
Total: 20G
Usado: 8,1G
% de uso: 44%
#
```

Figura 19: Script que muestra el estado del servidor

Por un lado, en el VPS de la web se ha instalado un servidor apache con php y un servidor mysql, además de phpmyadmin para la administración. También se ha instalado un servidor de correo Postfix.

Por otro lado, en el VPS del bot se ha instalado el bot junto con todos los paquetes necesarios para su funcionamiento.

Se ha registrado el dominio de la web (triumfalo.es) y se han añadido entradas en los DNS del dominio para vincularlo con su VPS.

ENTRADA	TIPO	VALOR	UTILIDAD
www.triumfalo.es	A	51.75.141.129	Conectar el dominio con el VPS de la web.
triumfalo.es	A	51.75.141.129	Conectar el dominio con el VPS de la web.
mail.triumfalo.es	A	51.75.141.129	Subdominio del servidor de correo.
triumfalo.es	MX	mail.triumfalo.es.	Indica a un cliente de correo la ubicación del servidor de correo.
triumfalo.es	TXT	v=spf1 a:mail.triumfalo.es -all	Registro SPF: Garantiza al receptor que el email que ha recibido proviene del remitente que se indica (evita suplantaciones de identidad)
default._domainkey.triumfalo.es	TXT	v=DKIM1; k=rsa; t=s; s=email; p=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ...	Registro DKIM: Garantiza que no se ha modificado el contenido del mensaje durante la transmisión.

Tabla 2: Entradas DNS

En la aplicación es muy importante la comunicación entre el servidor web y el servidor del bot. Para lograrlo, desde PHP se establecen conexiones SSH al servidor del bot y se ejecutan scripts en Bash que se han creado para cada una de las tareas o órdenes que se envían remotamente.



6. Implementación

En esta sección se describen los detalles de la implementación de la aplicación.

La aplicación se compone de:

- La interfaz de usuario, o front-end de la aplicación. La interfaz gráfica implementa los *wireframes* realizados en la etapa de diseño. En muchas ocasiones durante la implementación se han modificado partes de la estructura que se habían diseñado durante la etapa de diseño, ya que durante la implementación se tiene una visión distinta y van surgiendo mejoras y cambios.
- El back-end de la aplicación, con la implementación de la parte funcional y dinámica de la página. Esto incluye PHP, Javascript y Bash.

A continuación, se describe cómo se han implementado estas dos partes.

6.1 Implementación del front-end

Para iniciar la fase de implementación del front-end, se ha buscado plantillas dashboard que sirvan como base para la interfaz gráfica de usuario. Ya que hay un gran repertorio en internet con un gran nivel de diseño y facilita mucho el desarrollo además de reducir los costes de tiempo.

El objetivo para la interfaz es tener un diseño amigable y ligeramente desenfocado, pero siempre cuidando la imagen de profesionalidad que el producto debe dar. Así, se ha elegido la plantilla Admindek de colorlib.com. La plantilla incluye varios tipos de contenedores y widgets que han servido como base para el desarrollo de los elementos del front-end que se muestran en la Figura 22.

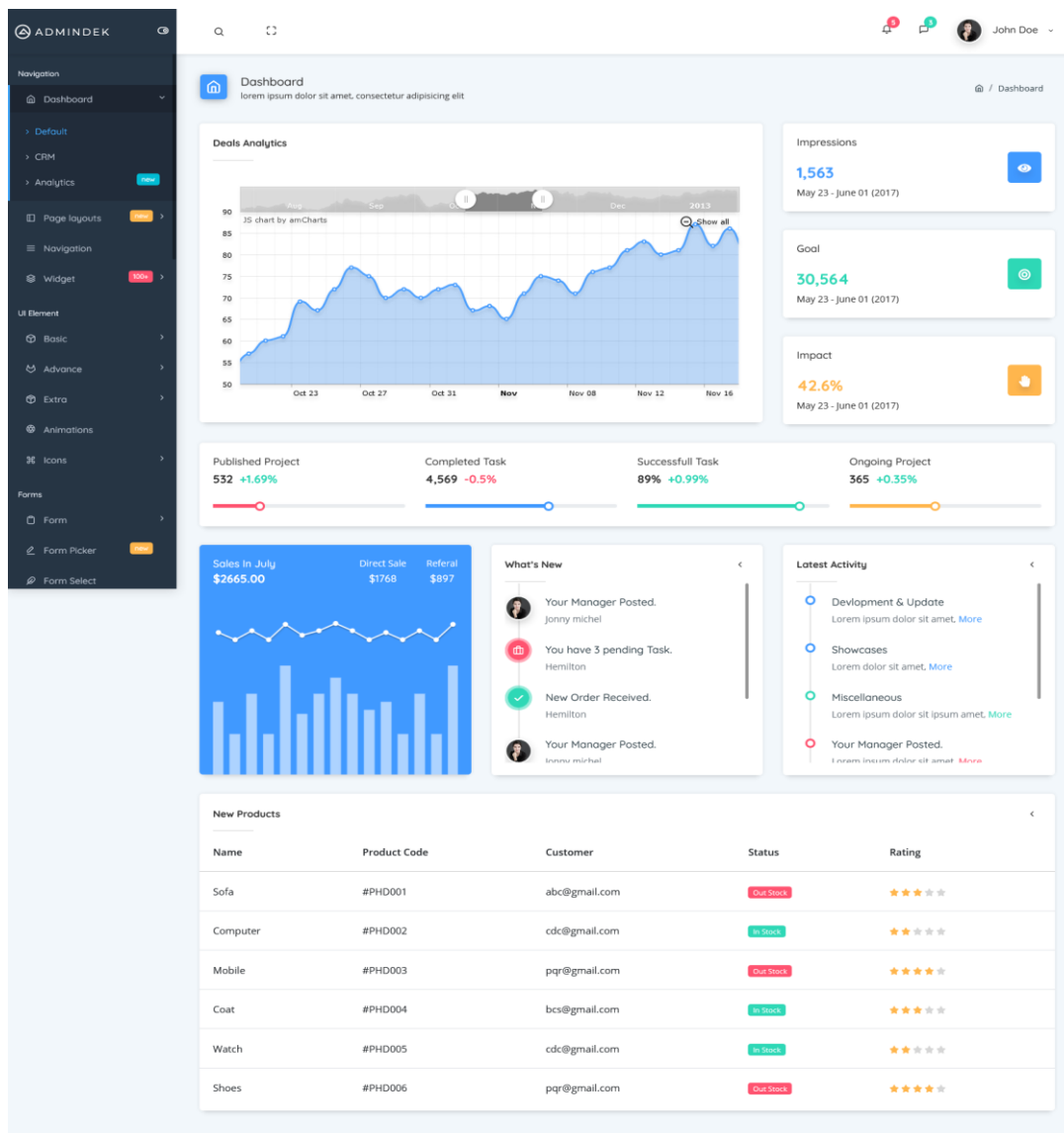


Figura 22: Plantilla Dashboard 1

Una vez seleccionada la plantilla, ya ha podido empezar a desarrollar.

Se han creado ficheros reutilizables en PHP para secciones como la cabecera o el pie de página, se incluyen mediante la función `include` en cada una de las páginas de la web. Ya que es un código que se repite en cada una de las páginas es mucho mejor unificarlo en un solo archivo. Es más fácil de mantener ya que si se quiere modificar solo habría que hacerlo una vez, ahorrando tiempo y posibles errores. Además, el resto de las páginas quedan limpias de código redundante ciñéndose únicamente al contenido concreto de las mismas.



```

84 </head>
85 <title><?= isset($pageTitle) ? $pageTitle : "Triunfalo"?></title>
86
87 <!-- Google Tag Manager -->
88 <script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
89 new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
90 j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=true;j.src=
91 'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);
92 })(window,document,'script','dataLayer','GTM-T72H3B8');</script>
93 <!-- End Google Tag Manager -->
94
95 <!-- HTML5 Shim and Respond.js IE10 support of HTML5 elements and media queries -->
96 <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
97 <!--[if lt IE 10]>
98 <script src="/js/old-browser-support/html5shiv.min.js"></script>
99 <script src="/js/old-browser-support/respond.min.js"></script>
100 <![endif]-->
101
102
103 <!-- Meta -->
104 <meta charset="utf-8">
105 <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=0, minimal-ui">
106 <meta http-equiv="X-UA-Compatible" content="IE=edge" />
107 <meta name="description" content="Crecimiento organico automatizado para tus cuentas de Instagram" />
108 <meta name="keywords" content="instagram, grow, bot, influencer, increase, automate, like, follow, comment">
109 <meta name="author" content="trunfalo" />
110 <!-- Favicon icon -->
111 <link rel="icon" href="/assets/img/favicon.ico" type="image/x-icon">
112 <!-- Google font-->
113 <link href="https://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700,800" rel="stylesheet">
114 <link href="https://fonts.googleapis.com/css?family=Quicksand:500,700" rel="stylesheet">
115 <link href="/assets/lib/feather-icons/feather.css" rel="stylesheet">
116 <?php if (function_exists( function: 'customCssTags')){
117     customCssTags();
118 }?>
119
120 </head>
121 <body>

```

Figura 23: Código header-logged.php

Estos ficheros reutilizables también contienen las llamadas a los ficheros Javascript y CSS. Como la mayoría de las páginas tienen ficheros Javascript y CSS únicos, se ha creado una función para que desde cada una de las páginas se les pase a los ficheros de cabecera y pie los ficheros Javascript y CSS únicos para que se incluyan donde corresponde. Se ha hecho lo mismo para el título de la página.

En cuanto a librerías principales, se ha utilizado Bootstrap para CSS y JQuery para Javascript. Además, se han usado otras librerías Javascript para optimizar el desarrollo como `validatejs` o `jscharts`, entre otras.

```

1 <?php
2 $pageTitle="Triunfalo | Configuracion";
3
4 function customCssTags(){?>
5     <!--Jquery nice select-->
6     <link rel="stylesheet" href="./assets/lib/jquery-nice-select/jquery-nice-select.css">
7     <!--Bootstrap-tagsinput-->
8     <link rel="stylesheet" href="./assets/lib/bootstrap-tagsinput/bootstrap-tagsinput.css">
9     <link rel="stylesheet" type="text/css" href="./css/widget.css">
10 <?php }
11
12 function customScriptTags(){?>
13
14     <!--Jquery nice select-->
15     <script src="./assets/lib/jquery-nice-select/jquery-nice-select.min.js"></script>
16     <!--Bootstrap-tagsinput-->
17     <script src="./assets/lib/bootstrap-tagsinput/bootstrap-tagsinput.min.js"></script>
18     <script src="./assets/lib/sweetalert-js/sweetalert.min.js"></script>
19     <!-- My js -->
20     <script src="./js/bot-config.js"></script>
21
22 <?php }
23 //if(isset($_GET['igid'])){ //Comprobamos que esta la variable con la id de cuenta de ig
24 // $igid=$_GET['igid'];
25 //}else{ //Si no tiene puesta la variable de ig lo devolvemos a seleccion de cuenta de ig
26 // header('Location: ig-account-select.php');
27 // exit();
28 //
29 //}
30
31 include_once('header-logged.php');

```

Figura 24: Código bot-config.php Uso de tags personalizados

También se han creado ficheros reutilizables para la comprobación de sesión y las conexiones a la base de datos.

```

13 if ( !isset($_POST['username'], $_POST['password']) ) {
14     // Could not get the data that should have been sent.
15     die ('Username and/or password does not exist!');
16 }
17 // Prepare our SQL
18 if ($stmt = $con->prepare( query: 'SELECT id, password, activation_code FROM accounts WHERE username = ?' )) {
19     // Bind parameters (s = string, i = int, b = blob, etc), hash the password using the PHP password_hash function.
20     $stmt->bind_param( types: 's', &var: $_POST['username'] );
21     $stmt->execute();
22     $stmt->store_result();
23     // Store the result so we can check if the account exists in the database.
24     if ($stmt->num_rows > 0) {
25         $stmt->bind_result( &var: $id, &var: $password, &var: $activation_code );
26         $stmt->fetch();
27         // $stmt->close();
28
29         // Account exists, now we verify the password.
30         if (password_verify($_POST['password'], $password)) {
31
32             if ($activation_code == "activated"){
33                 // Verificación completada
34                 $_SESSION['loggedin'] = TRUE;
35                 $_SESSION['name'] = $_POST['username'];
36                 $_SESSION['id'] = $id;
37                 $stmt->close();
38                 echo '1';

```

Figura 25: Código authenticate.php

Para el control de la sesión se utilizan cookies. Cuando el usuario se loguea se crean tres cookies. Una indica que esta logueado, otra indica el nombre de usuario y otra el id.

Estas cookies se consultan en la mayoría de las acciones que realiza el usuario para comprobar que está autorizado a realizarlas.

6.1.1 Login/Registro

La página de login presenta una estructura típica de usuario y contraseña. También tiene links para recuperar la contraseña en caso de olvido y para registrarse.

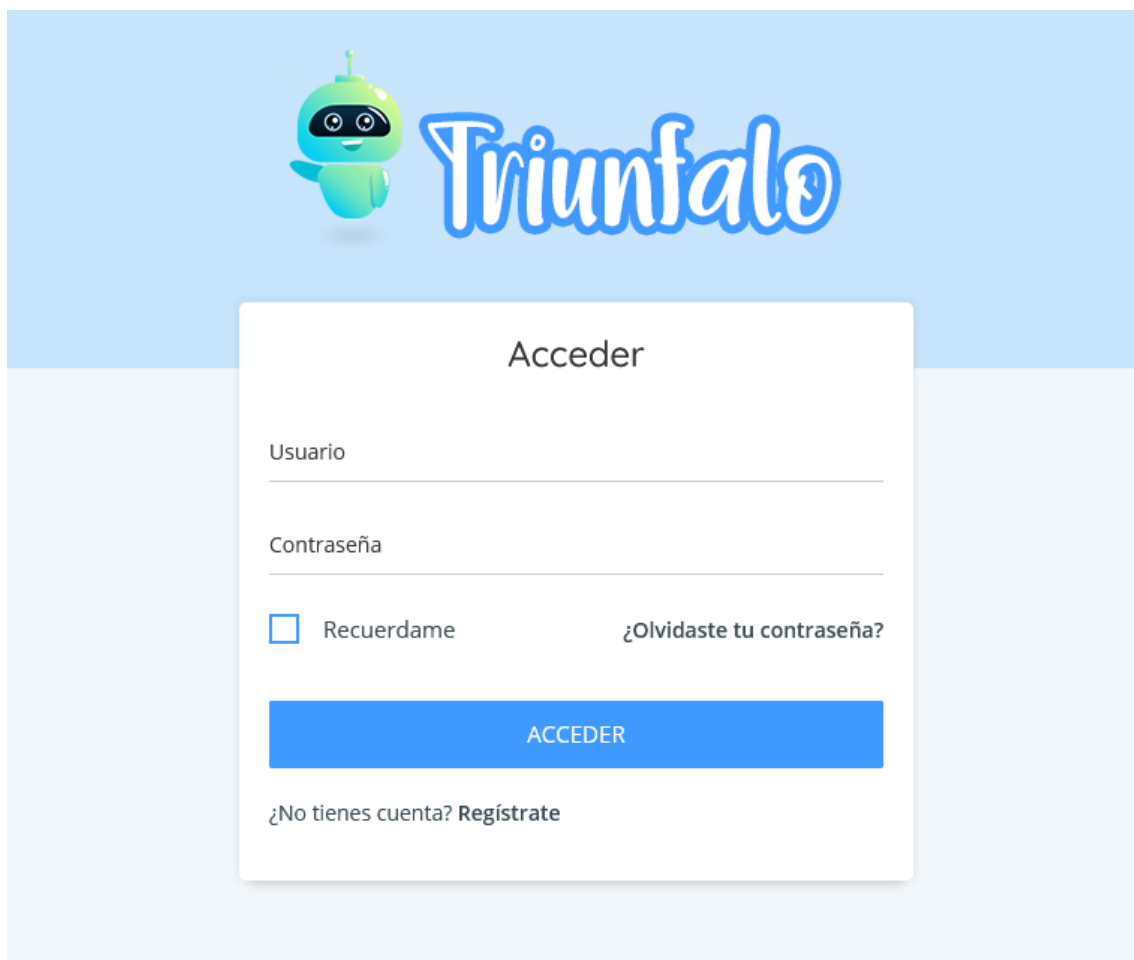
The image shows a login form for a website called 'Triunfalo'. At the top left is a green robot icon. To its right is the word 'Triunfalo' in a blue, bubbly font. Below this is a white rectangular box with a light blue border. Inside the box, the title 'Acceder' is centered at the top. Below the title are two input fields: 'Usuario' and 'Contraseña', each with a horizontal line underneath. Below the 'Contraseña' field is a checkbox labeled 'Recuérdame' and a link that says '¿Olvidaste tu contraseña?'. At the bottom of the box is a blue button with the text 'ACCEDER' in white. Below the button is a link that says '¿No tienes cuenta? Regístrate'.

Figura 26: Login

En la Figura 27 se muestra la página de registro.





Crea tu cuenta

Nombre de usuario

Correo electronico

Contraseña

Confirma contraseña

He leído y acepto los Terminos & Condiciones

REGÍSTRATE

Figura 27: Registro

Tanto el login como el registro utilizan la librería `validatejs` para validar los campos antes de enviar y mostrar mensajes personalizados.

Además, la página de registro comprueba mediante Ajax en los campos usuario y email si éstos ya existen cada vez que el usuario escribe un carácter. Si el usuario o email ya existen, muestra un mensaje advirtiéndolo sin necesidad de que pulse en enviar.

Estado	Método	Dominio	Archivo
200	GET	trunfalo.es	check-user-exists.php?username=test3
200	GET	trunfalo.es	check-email-exists.php?email=fccuenta@gmail.com
200	GET	trunfalo.es	check-email-exists.php?email=fccuenta@d
200	GET	trunfalo.es	check-email-exists.php?email=fccuenta@g
200	GET	trunfalo.es	check-email-exists.php?email=fccuenta@gm

Figura 28: Peticiones ajax registro

6.1.2 Recuperar contraseña

Recuperar la contraseña consta de varias páginas. La primera es el formulario que se enlaza desde la página de login.

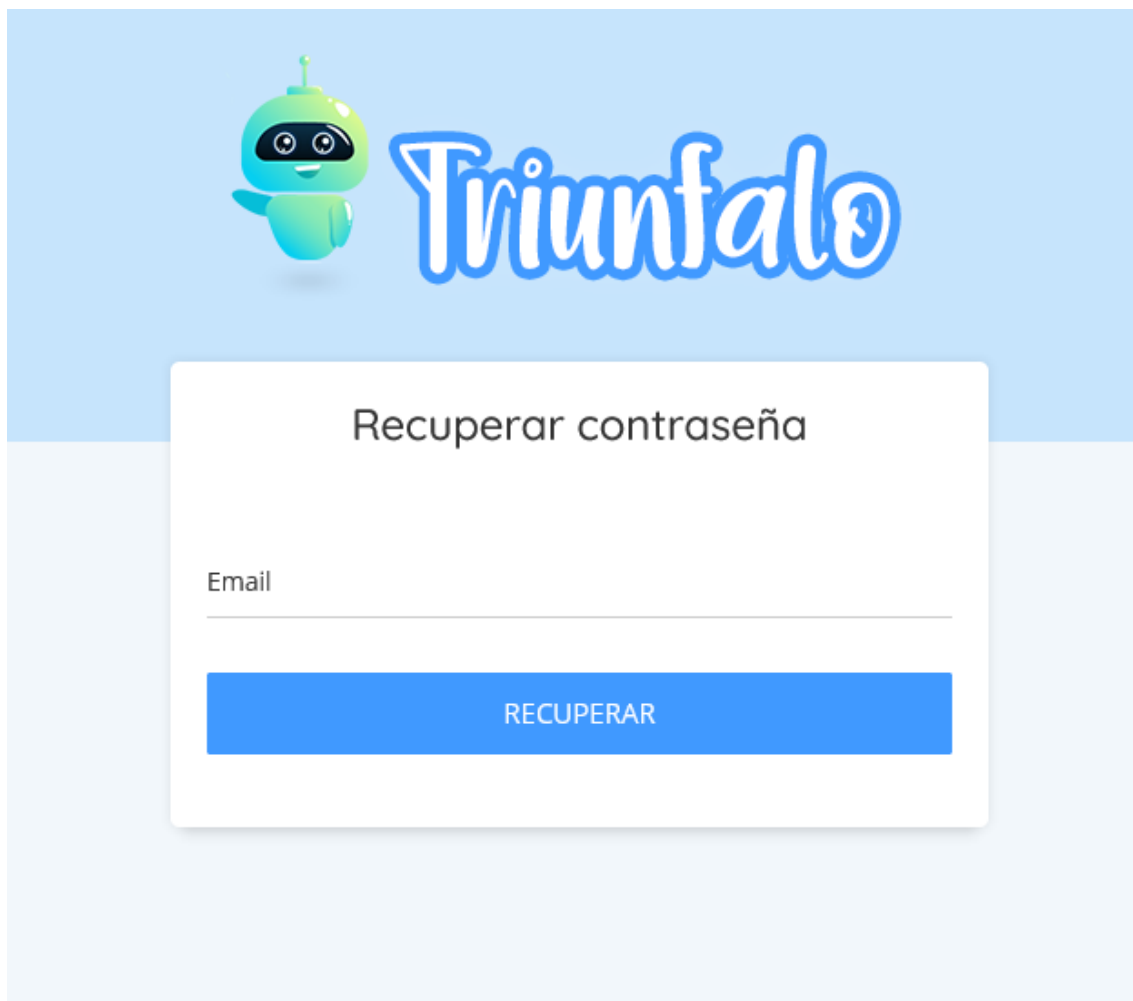


Figura 29: Recuperar contraseña

El formulario lo que hace es generar un token aleatorio con una fecha de caducidad de 5 minutos y lo añade a la entrada del usuario al que pertenece el email en la base de datos. A continuación, envía un email al usuario usando el servidor Postfix instalado en el VPS. El email tiene un enlace a <https://triumfalo.es/webTFG/recover-password-change.php?email=usuario@email.es&code=60c3101c729d2> donde se pasan el email del usuario y el token por GET.

Una vez el usuario accede al enlace recibido por email, el servidor comprueba en la base de datos que el token se ha creado hace menos de 5 minutos y que es correcto para el email dado. Si es así, le muestra los campos para cambiar la contraseña.

Si el token no es válido o ha expirado se indica a través de la interfaz.

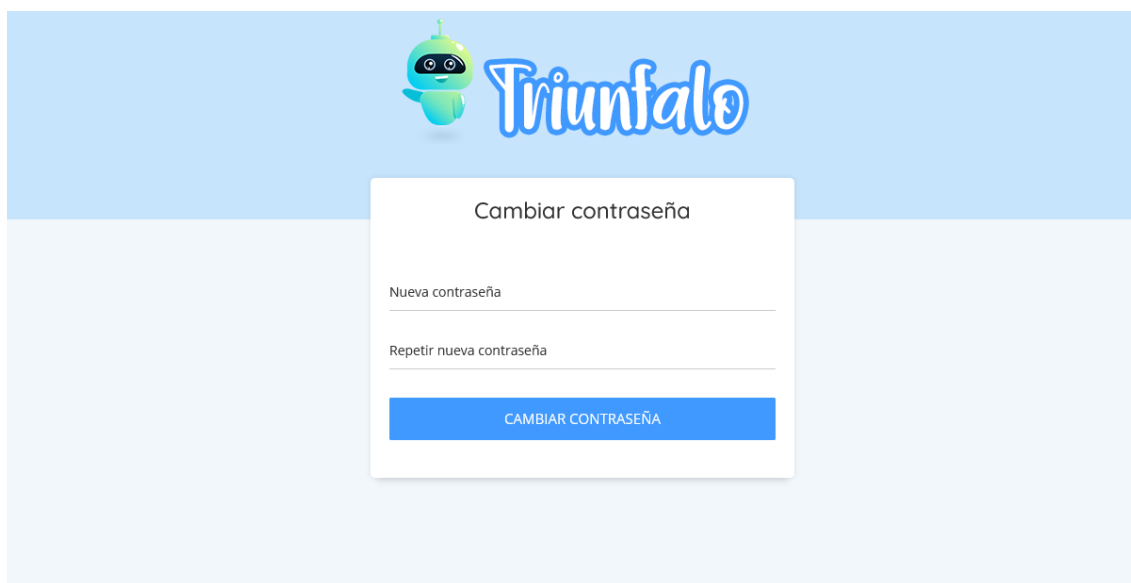


Figura 30: Cambiar contraseña

6.1.3 Mis cuentas

La página de mis cuentas es la primera página que ve el usuario una vez entra en la aplicación. Desde esta página el usuario tiene acceso a sus cuentas de Instagram registradas en el bot (@davidmarti63 y @recambios_fran en la Figura 31), pudiendo además registrar nuevas (botón “+” en la Figura 31) e incluso eliminar las existentes.

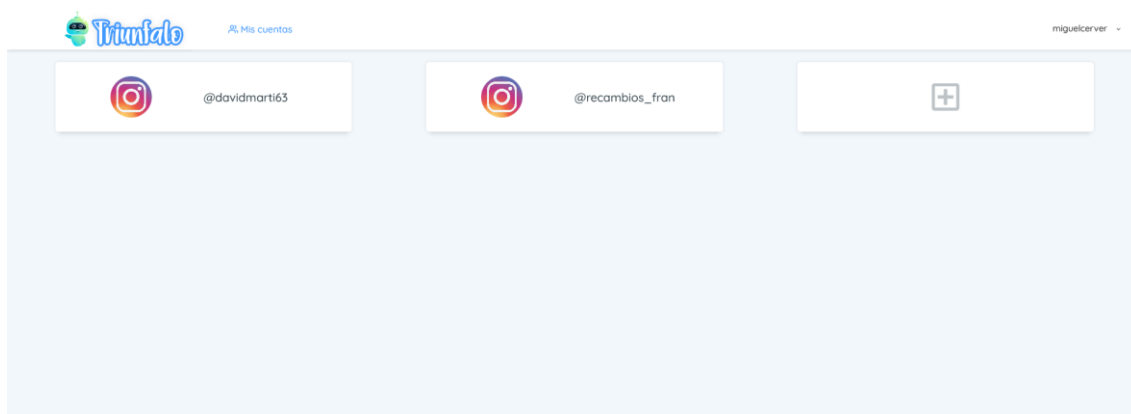


Figura 31: Mis cuentas

Un usuario puede tener múltiples suscripciones, aunque (cada suscripción deberá asociarse a una cuenta diferente de Instagram). Una vez se realice esta asociación entre cuenta de Instagram y suscripción esta última se podrá activar y a partir de este momento se podrá usar el bot.

Para asociar una nueva cuenta de Instagram a una suscripción basta con pulsar en el botón “+” (ver Figura 31) e introducir las credenciales de Instagram (ver Figura 32). Si el usuario tiene una suscripción libre (por ejemplo, una usada con una cuenta de

Instagram que se ha eliminado de nuestra aplicación), el sistema directamente abre un pop-up para registrar la cuenta.

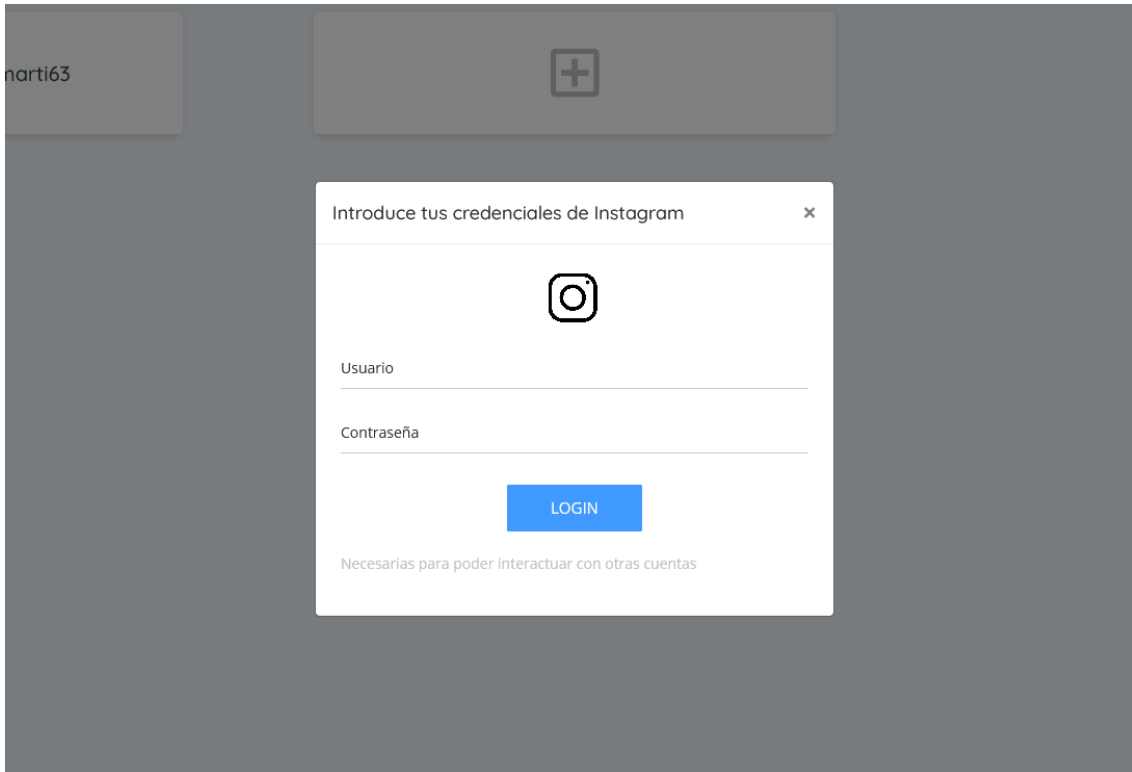


Figura 32 :Login Instagram en Mis Cuentas

En caso contrario, es decir, cuando el usuario no dispone de ninguna suscripción activa, el sistema redirigirá al usuario a la página de compra de suscripción (ver Figura 33).

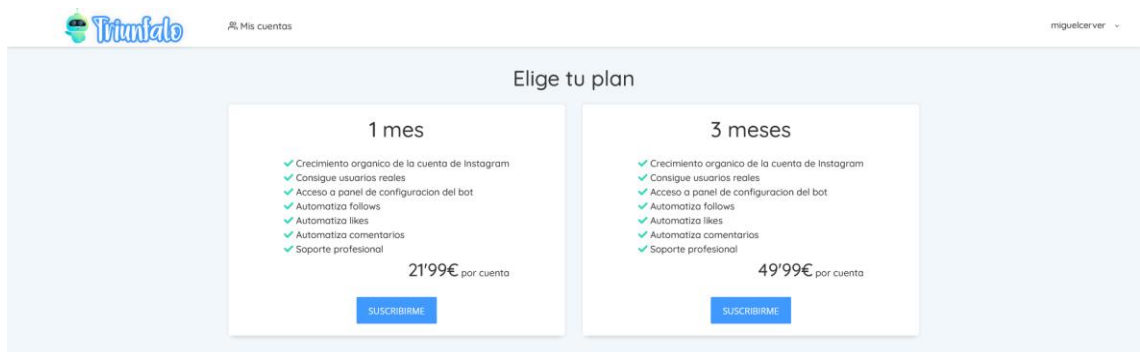


Figura 33: Selección de plan

Una vez completado el pago, el sistema redirige al usuario a la página de “Mis cuentas” y abre automáticamente el pop-up de registro de cuenta. Para ello se le pasa por GET el parámetro “successpay” con el valor “true” a la página de “Mis cuentas”:

<https://triumfalo.es/webTFG/ig-account-select.php?successpay=true>

Cada vez que se ejecuta esta página, pide por REST el estado de las suscripciones asociadas a las cuentas. En caso de no tener o de estar caducada, se muestra un botón para activar la cuenta(en realidad lo que se activa es la suscripción) y el bot está desactivado para la cuenta de Instagram en cuestión(la desactivación se produce en el mismo momento en el que expira la suscripción, para más detalles consultar la sección de suscripciones).

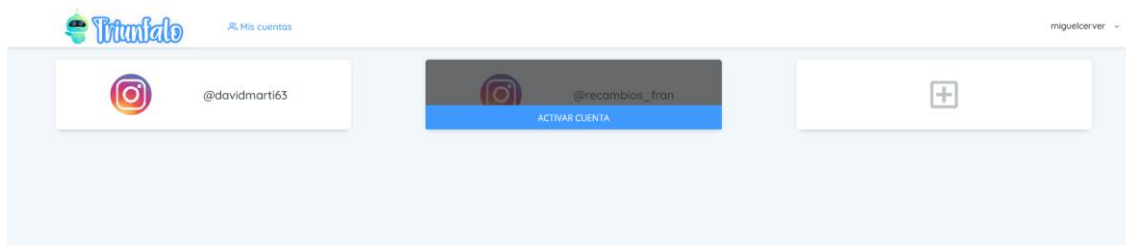


Figura 34: Activar cuenta

Si se pulsa en activar, se comprueba por REST si existe una suscripción activa libre. En caso afirmativo se reactiva la cuenta y se asocia con ella. En caso negativo se dirige a la página de compra de suscripción.

Mediante un argumento que se le pasa a la suscripción que se ha creado en Stripe, se le asigna a la suscripción el identificador de la cuenta de Instagram que se quiere reactivar. Así, una vez completado el pago se asociará automáticamente la nueva suscripción con la cuenta de Instagram que se quiere reactivar.

6.1.4 Centro de operaciones

El centro de operaciones es la página que se accede cuando se pulsa en una cuenta. El propósito de esta página es tener una visión general de la cuenta y su progreso, estadísticas de interacción y qué configuración del bot tiene asignada para cada día de la semana.

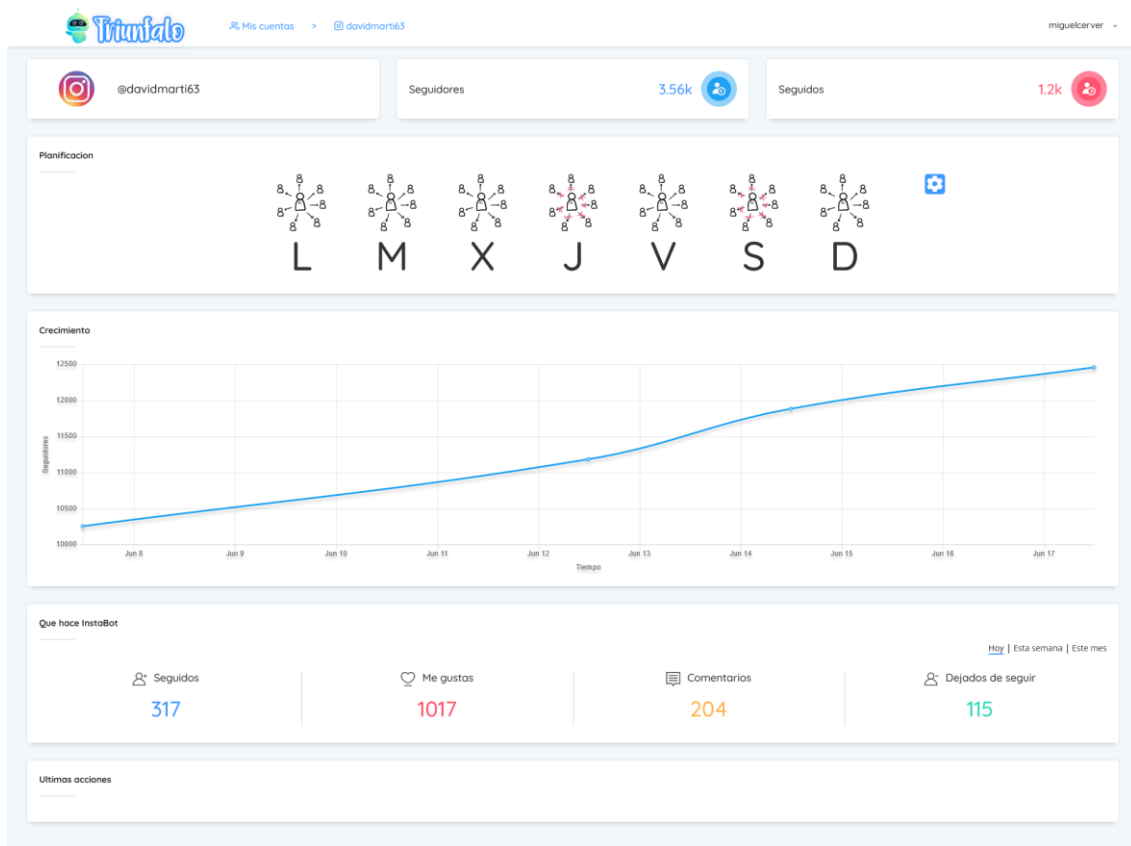


Figura 35: Centro de operaciones

Actualmente, la funcionalidad que permite mostrar los datos estadísticos de una cuenta de Instagram no está desarrollada y los datos que se muestran en la página son fijos (ver bloque “Que hace InstaBot” en la Figura 35). Sin embargo, el contenedor de planificación sí que ha sido desarrollado (ver bloque “Planificación” en la Figura 35). Este muestra las configuraciones asignadas a cada día de la semana. Al poner el puntero encima de cada una de ellas se muestra un pop-up con el nombre de la configuración tal y como se muestra en la Figura 36.

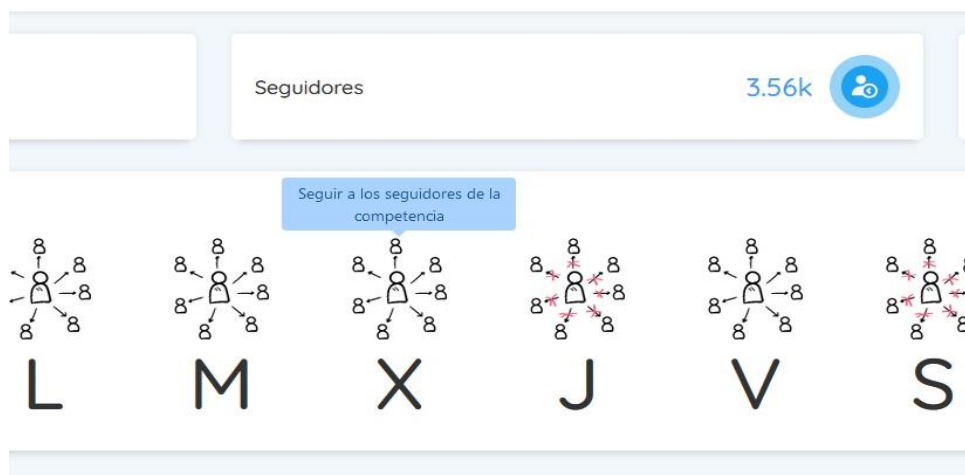


Figura 36: Pop-up en Centro de operaciones

El sistema permite definir dos tipos de configuración:

- Interacción con otros usuarios. Esta es la interacción principal e incluye cualquier tipo de interacción con otras cuentas (seguir, comentar o dar me gusta). Se representa en la planificación tal y como se muestra en la Figura 37.
- Interacción para dejar de seguir. Este tipo de interacción es secundaria ya que no sirve para hacer crecer la cuenta de Instagram. Sin embargo, es necesaria porque como el bot sigue a muchos usuarios, la cuenta de Instagram que utilice nuestra aplicación terminaría siguiendo a un enorme número de usuarios. Esto restaría credibilidad de cara a los usuarios y acabaría repercutiendo en el crecimiento. Se representa en la planificación tal y como se muestra en la Figura 38.

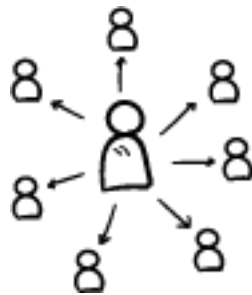


Figura 37: Imágen interacción



Figura 38: Imagen dejar de seguir

Por último, pulsando en el engranaje ubicado arriba a la derecha en el contenedor de Planificación se accede a la configuración del bot.

6.1.5 Configuración

Las configuraciones del bot están accesibles desde el bloque de “Configuraciones” que aparece en la Figura 39. Desde ahí el usuario puede especificar los días de la semana que deben ejecutarse. Para crear una configuración el usuario debe pulsar en el botón “+” y completar el formulario que contiene todos los parámetros (ver bloque “Configurar” en la Figura 39). Entre la información que el usuario debe proporcionar se encuentra:

- Decidir si se desea interactuar o hacer unfollow usando el switch de arriba a la izquierda.
- En el caso de interactuar, rellenar la configuración según los intereses y el público objetivo de la cuenta de Instagram. Esto incluye indicar el método para crear la interacción (Hashtags, seguidores de cuentas o ubicaciones), indicar si se desea hacer follow, dar “me gusta” o comentar (creando una batería de comentarios genéricos en el campo correspondiente) y finalmente indicar el nivel de potencia de las interacciones (a más potencia, más número de interacciones por día).

- En el caso de unfollow, tan solo indicar a qué usuarios se dejará de seguir (Usuarios que ha seguido Triunfalo, todos los usuarios que se siguen o todos los usuarios que se siguen que no siguen a la cuenta del usuario).

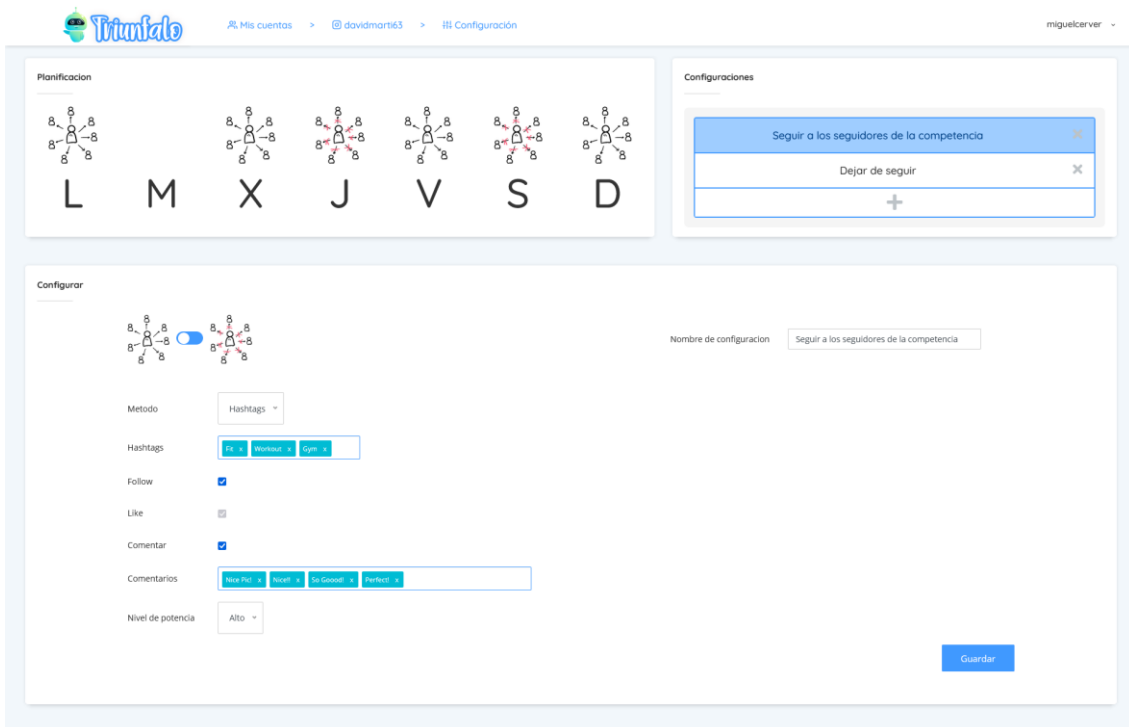


Figura 39: Configuración 1

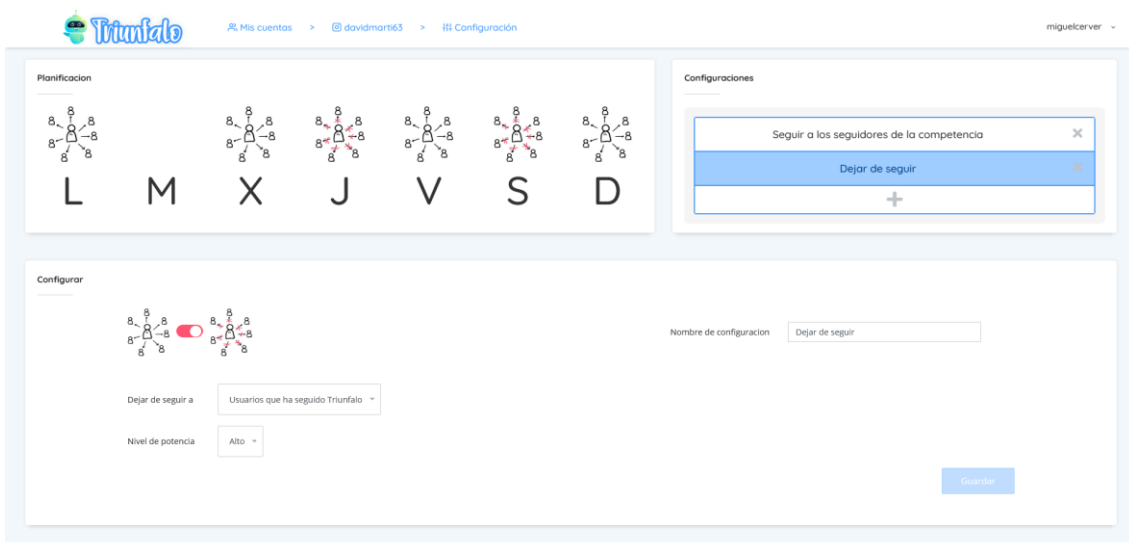


Figura 40: Configuración 2

Cuando se crea una configuración, se crea en la base de datos una entrada con todos los parámetros seleccionados. Así mismo, se crea un ejecutable del bot en el VPS del bot para el usuario con todos los parámetros que ha elegido.

Si se modifica una configuración existente se modifica tanto la entrada de la base de datos como el ejecutable correspondiente en el VPS del bot. De igual modo, al eliminarla se elimina en ambos sitios. La consistencia es absoluta.

Una vez creadas las configuraciones que se necesiten, hay que asignarlas a un día de la semana. Sólo puede haber una configuración por día, y la misma configuración puede usarse en muchos días.

Para asignarlas simplemente hay que arrastrarlas a los días deseados. Para la función de drag&drop se ha usado la librería jQueryUI, concretamente las funciones droppable y draggable.

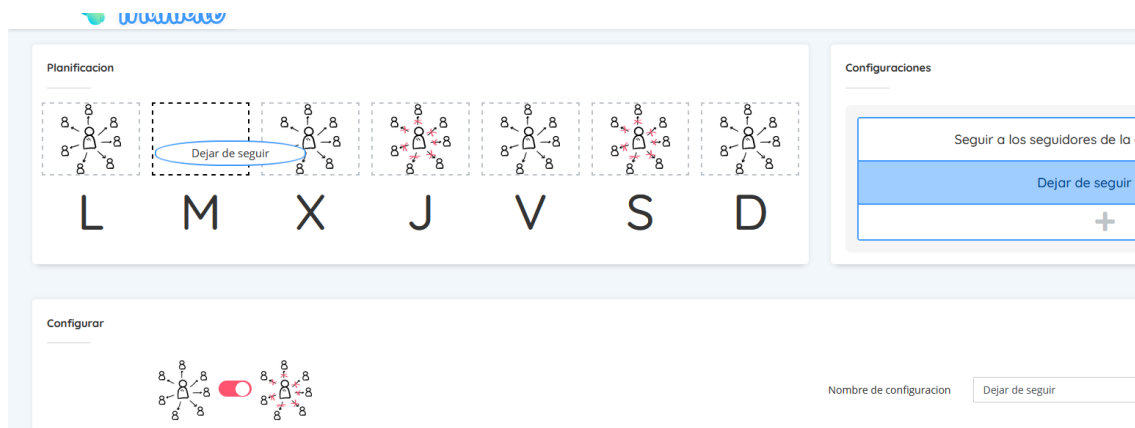


Figura 41: Asignación de configuración a día de la semana

Una vez se arrastra, se hace una llamada REST que asigna la configuración al día deseado en la base de datos. Esto se hace para poderlo mostrar en la interfaz cuando el usuario vuelva a entrar en otro momento y mantener un estado consistente.

Además, en el VPS del bot se crea una entrada en el crontab con hora y minuto aleatorios en el día de la semana indicado en la que se ejecutará una instancia del bot con la configuración creada.

Para eliminar una configuración de un día de la semana simplemente hay que posicionar el ratón encima de ésta y pulsar en la cruz que aparece.

ficacion

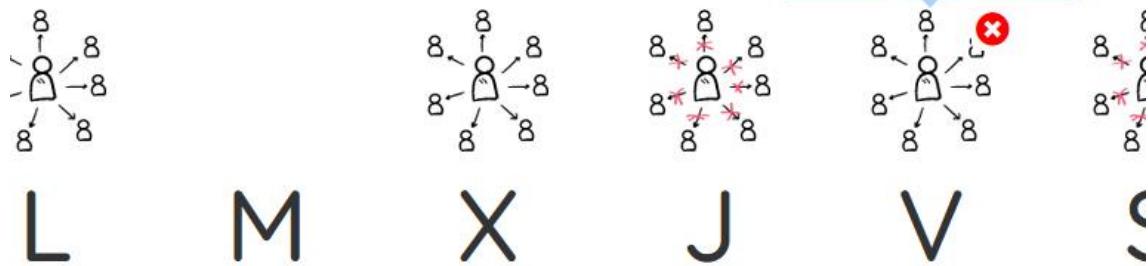


Figura 42: Desasignación de configuración

Cuando se pulsa en eliminar se elimina de la base de datos y también del crontab asociado.

6.1.6 Suscripciones y pagos

En la barra superior de la web, en la parte derecha se despliega un submenú al hacer click en el nombre de usuario. Desde ahí se puede acceder a la página “Suscripciones y pagos”.

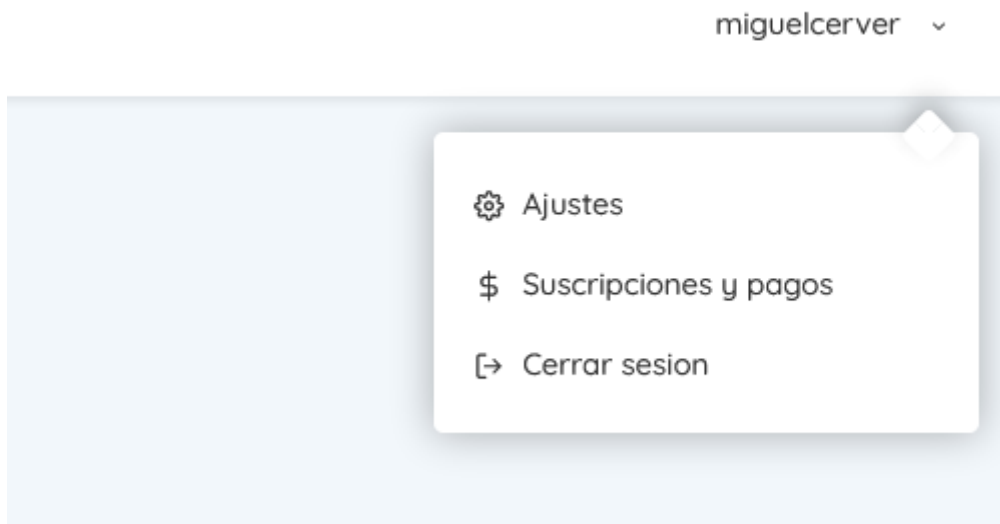


Figura 43: Menú desplegable Suscripciones y pagos

La misión de esta página es el control de todas las suscripciones activas, donde se puede ver a qué cuenta están asociadas, cuando finalizan y gestionar la renovación

automática. Todo esto se hace desde la misma web usando el API de Stripe para modificar el estado de la renovación automática y la base de datos de la web para el resto de los datos.

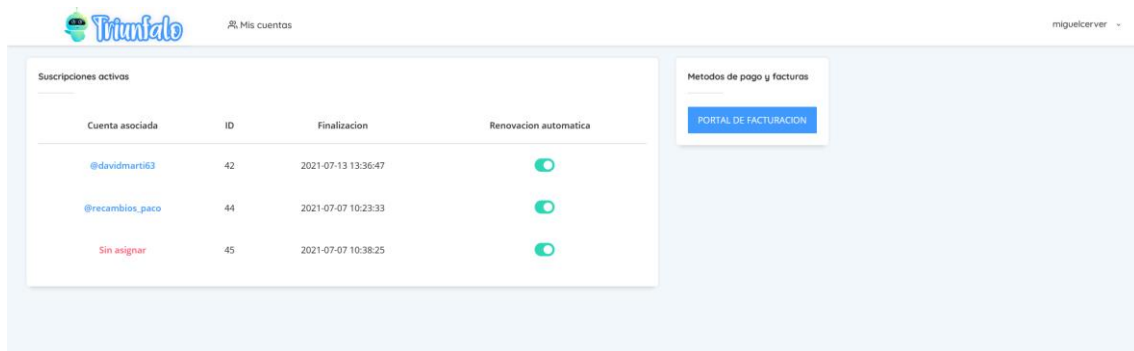


Figura 44: Suscripciones y pagos

Además, pulsando el botón de “Portal de Facturación” nos lleva al portal de facturación de Stripe donde se puede añadir métodos de pago y asignárselos a las suscripciones, consultar los métodos de pago o consultar el historial de facturación y descargar facturas.

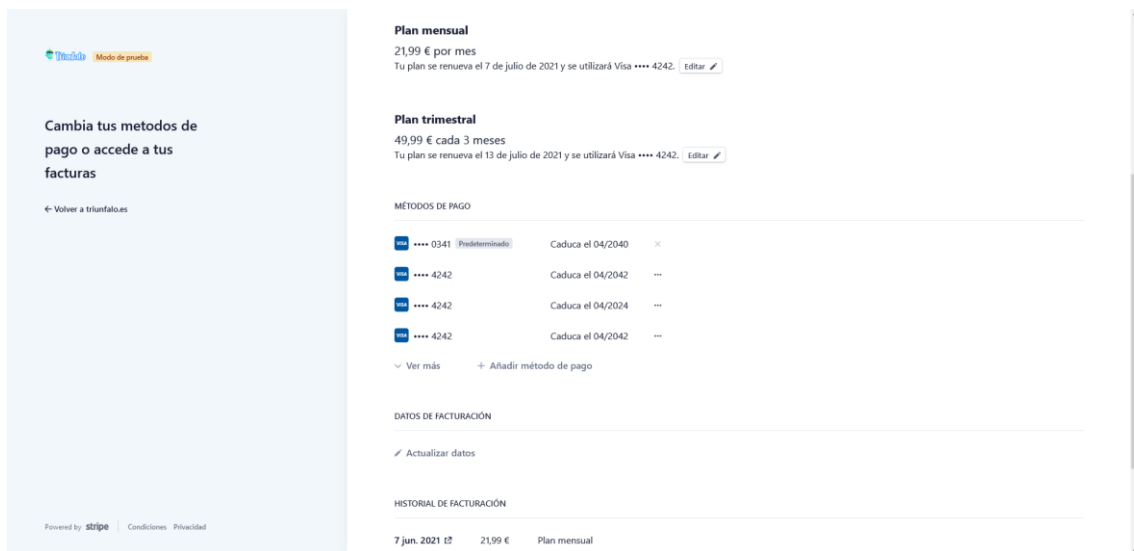


Figura 45: Portal de facturación

6.1.7 Ajustes

Desde la página de ajustes de la cuenta, accesible desde el menú desplegable ubicado en la esquina superior derecha, el usuario puede cambiar datos básicos de su cuenta principal de la aplicación web como son el email y la contraseña.

Mis cuentas

Configurar cuenta

Cambiar Email

Cambiar contraseña

Nueva contraseña

Repetir contraseña

Figura 46: Ajustes de cuenta

6.1.8 Landing Page

La landing page es una parte muy importante de cualquier negocio web. Es la página principal a la que accede un visitante y su objetivo es convertirlo en cliente. Por ello es muy importante que se cuiden todos los detalles y darle al visitante la información que necesita para entender qué es lo que se ofrece.

Además, se le debe dar de forma concisa, visual y amigable para facilitar la lectura y evitar textos pesados que disuadan al visitante.

Para el desarrollo de la landing page se hizo una búsqueda entre plantillas Bootstrap con el objetivo de encontrar una que encajase en la filosofía y estilo del proyecto.

Finalmente se escogió Rapid de bootstrapmade.com (<https://bootstrapmade.com/rapid-multipurpose-bootstrap-business-template/>)

En las Figuras 47, 48 y 49 se muestran algunas instantáneas de la plantilla sin modificar.

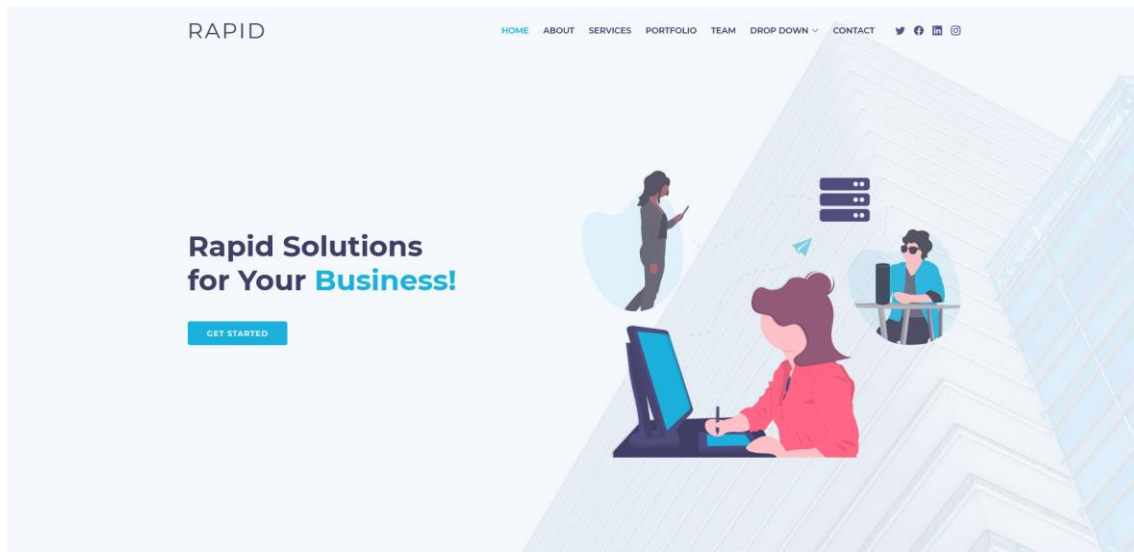


Figura 47: Plantilla Landing 1

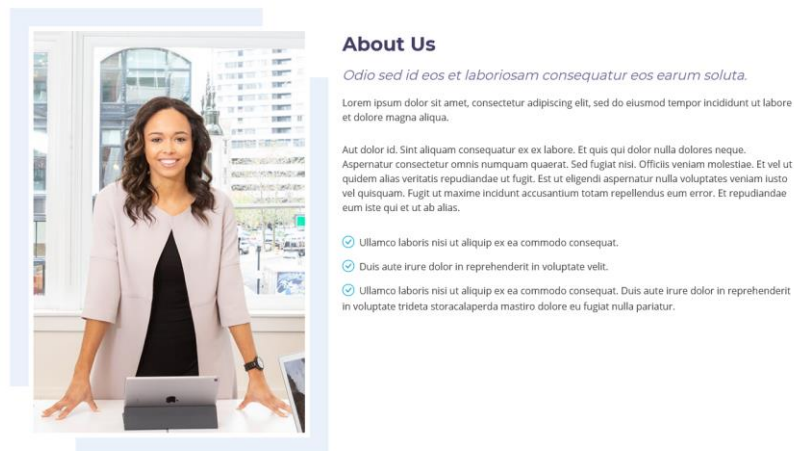


Figura 48: Plantilla Landing 2

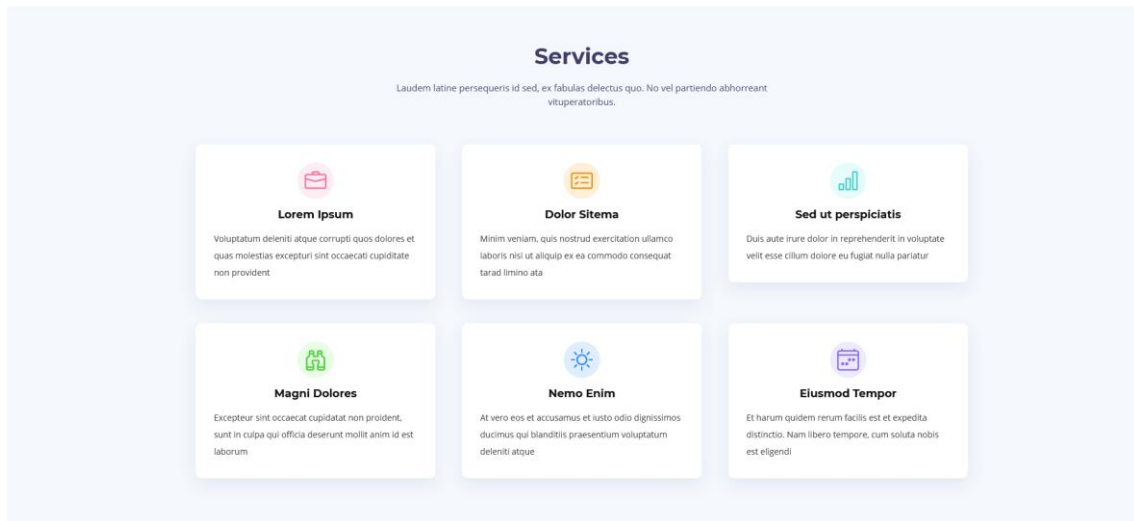


Figura 49: Plantilla Landing 3

Se ha usado como base para la landing. Se han buscado imágenes que representen nuestro producto. Para ello se ha usado el repositorio de imágenes y vectores Freepik.es.

Se han tenido que modificar usando Adobe Illustrator para eliminar los fondos y algún que otro elemento.

Además, se ha modificado el CSS para mostrar las nuevas imágenes en un tamaño adecuado y que se redimensionen correctamente para la versión móvil.

También se han hecho cambios en la barra de menú ya que el logo de la plantilla es texto y había que adaptarlo para que redimensione bien el logo de Triunfalo que es una imagen.

A continuación, se muestra el resultado final del landing page.



Figura 50: Landing Page 1



¿Qué te ofrecemos?

La solución más completa para el crecimiento orgánico en Instagram

Todos los días, nuestro bot busca automáticamente usuarios potencialmente afines a la temática de tu cuenta e interactúa con ellos. Esta interacción automatizada se traduce en una **constante entrada de seguidores activos**.

Tenemos la clave del éxito para:

- ✓ Negocios que quieran ampliar su cartera de clientes y ventas
- ✓ Influencers que buscan más rentabilidad por su contenido
- ✓ Cuentas tanto pequeñas como grandes que quieren posicionarse como merecen

Figura 51: Landing Page 2

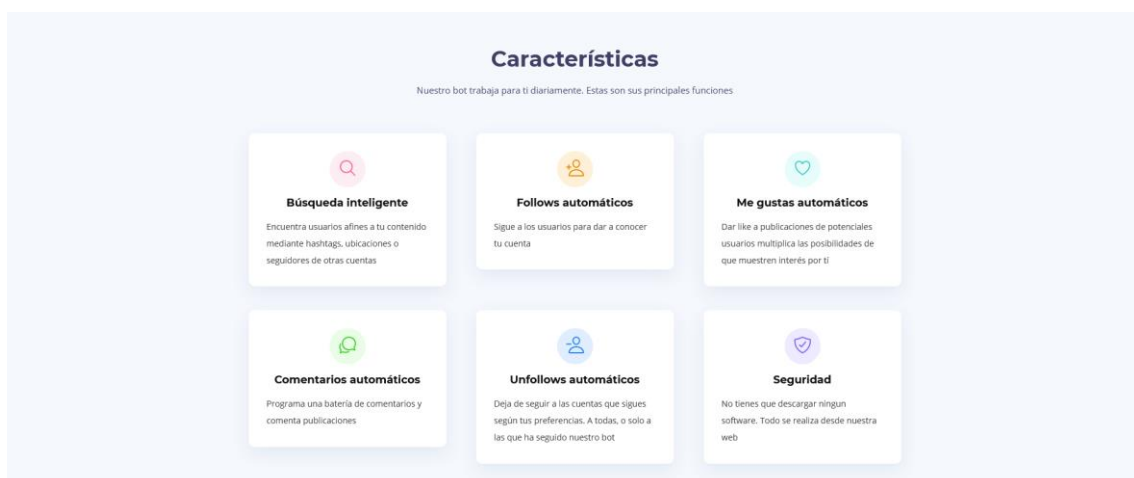
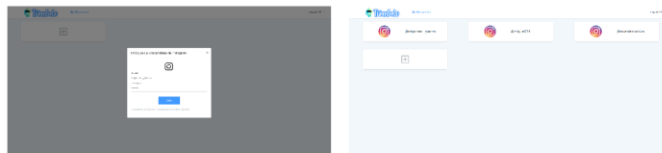


Figura 52: Landing Page 3

¿Cómo funciona?

1 Añade tu cuenta de Instagram

Puedes añadir múltiples cuentas. Tantas como suscripciones poseas



2 Crea configuraciones

El control total que poseen nuestros clientes nos hace únicos

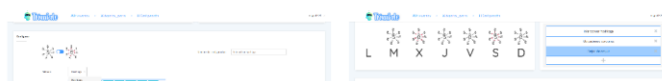


Figura 53: Landing Page 4

Planes

Elige el plan que prefieras. Puedes cancelarlo en cualquier momento

Principiante

€21⁹⁹ /mes

- Acceso al panel de configuración
- Búsqueda inteligente
- Follows automáticos
- Me gustas automáticos
- Comentarios automáticos
- Unfollows automáticos
- Soporte profesional

REGÍSTRATE

Profesional

€49⁹⁹ /trimestre

25% descuento

- Acceso al panel de configuración
- Búsqueda inteligente
- Follows automáticos
- Me gustas automáticos
- Comentarios automáticos
- Unfollows automáticos
- Soporte profesional

REGÍSTRATE

Figura 54: Landing Page 5

Preguntas frecuentes

¿Es seguro para mi cuenta de Instagram?

Totalmente seguro. Simulamos patrones de comportamiento de una persona real mediante inteligencia artificial cuando interactuamos con otras cuentas. Por eso Instagram percibe que eres tú el que está realizando las acciones.

¿Puedo cancelar la suscripción en cualquier momento?

¿Es difícil configurar la herramienta?



Figura 55: Landing Page 6

La landing es completamente responsive y se adapta a cualquier dispositivo ya sea tablet o móvil.



Figura 56: Landing Page Móvil 1



Figura 57: Landing Page Móvil 2

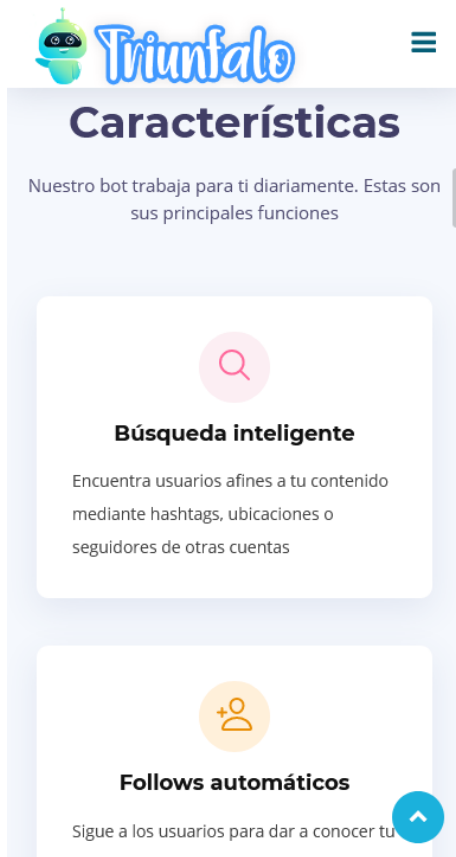


Figura 58: Landing Page Móvil 3

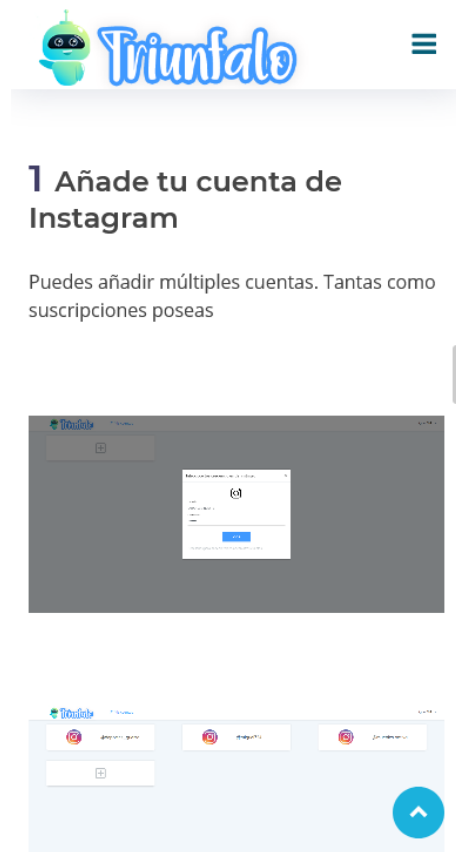


Figura 59: Landing Page Móvil 4



Figura 60: Landing Page Móvil 5

6.2 Implementación del back-end

En cuanto al back-end, en el servidor del bot se ejecuta una instancia del bot por cada cuenta de Instagram configurada que tenga una suscripción vigente asociada. Una instancia es un ejecutable Python parametrizado con los datos de la cuenta de Instagram y el método de interacción elegido.

Las instancias se ejecutan todas las semanas los días de la semana configurados. Se crean entradas en crontab cuando un usuario asigna una configuración a un día de la semana. La hora y minuto de ejecución de cada instancia son aleatorios para distribuir lo máximo posible la carga dentro del servidor.

Para todas las acciones de cierta complejidad que se ordenan desde el servidor web se ha creado un script en Bash. Así el servidor web solo tiene que llamar al script pasándole los parámetros necesarios mediante una conexión SSH.

6.2.1 Creación/actualización de las configuraciones

Cuando un usuario crea una configuración para una cuenta en la web, se hace una llamada REST desde bot-config.php utilizando AJAX a update-config.php donde se pasan todos los datos del formulario de configuración.

```

132 //Enviamos el json con todas las opciones
133 $.ajax({
134   url: 'rest/update-config.php',
135   type: 'POST',
136   data: json,
137   success: function(response){
138     swal("Configuración guardada correctamente", "", "success");
139
140     //Actualizamos el nombre de la config
141     $('[data-conf-id="'+idconfig+''] > h5).text(confname);
142
143     //Actualizamos el atributo id de la config
144     var nuevoconfid = response.configid;
145     $(".selected-config").attr("data-conf-id",nuevoconfid);
146
147     //Actualizamos el atributo conf-type
148     if(esinteract) {
149       $(".selected-config").attr("data-conf-type", 1);
150     }else{
151       $(".selected-config").attr("data-conf-type", 2);
152     }
153     //Si era una nueva config la que se ha guardado, reactivamos el boton mas
154     if(idconfig==""){
155       activaBotonMasOnClick();
156       $(".new-config-box").removeClass('no-click');
157     }
158   },
159   error: function () {
160   }
161 }
162
163 });

```

Figura 61: Webhook.php Creación/Actualización de configuración

Update-config.php comprueba si lo que está haciendo el usuario es actualizar una configuración existente o creando una nueva y actúa en consecuencia.

```
66 //Fin validación: Ya sabemos que todos los parametros tienen valores correctos
67
68 //Primero comprobamos que esta cuenta de ig es del usuario
69 $stmt = $con->prepare( query: 'select count(*) from ig_accounts where id = ? AND id_owner_account = ?');
70 $stmt->bind_param( types: 'ii', &var1: $igid, &var2: $userid);
71 if(!$stmt->execute()){ //Fallo en la ejecución de la query
72     http_response_code( response_code: 500);
73     exit();
74 }
75 $stmt->bind_result( &var1: $countResult); //Asignamos el valor de count(*) a la variable countResult
76 $stmt->fetch();
77 $stmt->close();
78 if($countResult==0){ // 0 no existe la cuenta de ig o no pertenece al usuario, forbidden
79     http_response_code( response_code: 403);
80     exit();
81 }
82
83 //Aqui obtendremos el numero de configuraciones para dicha cuenta de ig, para asegurarnos que no supera los limites
84
85 $stmt = $con->prepare( query: 'select count(*) from configuraciones where id_ig_account = ?');
86 $stmt->bind_param( types: 'i', &var1: $igid);
87 if(!$stmt->execute()){ //Fallo en la ejecución de la query
88     http_response_code( response_code: 500);
89     exit();
90 }
91 $stmt->bind_result( &var1: $countResult); //Asignamos el valor de count(*) a la variable countResult
92 $stmt->fetch();
93 $stmt->close();
94
95 //Creamos parametro conf_type segun sea interact o unfollow
96 if($_POST['esinteract']=='true'){ //interact
97     $conftype = 1;
98 }else{ //UnFollow
99     $conftype = 2;
100 }
101
102 if(empty($idconfig)){//Es una nueva configuracion (INSERT)
103     if($countResult<5) { //Comprobamos que no hay mas de x configs antes de crear una nueva
```

Figura 62: Bot-config.js

Una vez validados todos los parámetros, comprobado si se está actualizando una configuración o creando una nueva y insertado o actualizado en la base de datos la configuración, se ejecuta por SSH el script del servidor del bot pasándole todos los datos para que cree el ejecutable del bot.

```
ssh -o StrictHostKeyChecking=no -o UserKnownHostsFile=/dev/null -o LogLevel=ERROR -o ConnectTimeout=10 -o ControlMaster=no -o ControlPersist=10m -o ProxyCommand='ssh -W %h:%p root@192.168.1.100' root@192.168.1.100 'python3 /usr/bin/bot-config.js --igid=1 --userid=1 --esinteract=true --conftype=1 --idconfig=1 --targetid=1'
```

Figura 63: Bot-config.js Ejecución del script por SSH

El script responsable de la creación de las configuraciones del bot está escrito en Bash. Utiliza un fichero independiente que es una plantilla del ejecutable del bot (escrito en Python) donde se han incluido palabras clave que se utilizan como variables para que sean sustituidas por los valores que ha indicado el usuario. En la Figura 64 se muestra una parte del mismo.

```

botenemo boolean del dofollow
if [ $dofollow == 1 ]; then
    dofollow_boolean="True"
else
    dofollow_boolean="False"
fi

#Calculamos el parametro amount segun el nivel de potencia elegido y el numero de targets que hayan
if [ $sesinteract == 1 ]; then #Es interact
    num_targetinfo=${#array_targetinfo[@]}
    if [ $power == 1 ];then
        real_amount=$(( low_power / num_targetinfo ))
    elif [ $power == 2 ];then
        real_amount=$(( medium_power / num_targetinfo ))
    else
        real_amount=$(( high_power / num_targetinfo ))
    fi
else #Es unfollow
    if [ $power == 1 ];then
        real_amount=low_power
    elif [ $power == 2 ];then
        real_amount=medium_power
    else
        real_amount=high_power
    fi
fi

#Formatamos el array de targetinfos para que tenga el formato de string que necesita instapy 'targetinfo1',targetinfo2' ...
targetinfo_formatted=""
for var in $(array_targetinfo[@]); do
    targetinfo_formatted+="$var"
done
targetinfo_formatted=${targetinfo_formatted%}, #Eliminamos la coma final

#Probamos cual es el target para elegir la plantilla adecuada
if [ $sesinteract == 1 ]; then #Es interact
    if [ $target == 1 ]; then #Es hashtags
        template_selected=hashtag_template
    fi
else #Es unfollow
    if [ $target == 1 ]; then #Hacer unfollow a los que ha seguido instapy
        template_selected=unfollow_instapy_followed_template
    fi
fi

#Procedemos a escribir el archivo. Mandamos variables a la plantilla y creamos el archivo de configuracion final
sed -e "/\${iguser}/s/user/" -e "/\${igpass}/s/pass/" -e "/\${igcomments}/s/comments formatted/" -e "/\${igposts}/s/posts/" -e "/\${dofollow}/s/dofollow boolean/" -e "/\${targetinfo}/s/targetinfo formatted/" -e "/\${amount}/s/real_amount/" $template_selected > $conf_file

```

Figura 64: Crea_config_bot.sh

Una vez ejecutado el script, se crea en la carpeta de configuraciones del bot en el servidor un archivo de configuración cuyo nombre es {idinstagram}_{idconfiguracion}.py.

Con este nombre se facilita la posterior creación de entradas en el crontab para crear una ejecución de la configuración programada para el día de la semana que asigne el usuario.

```

[root@botVPS webTFG]# ls
31 219.py 31 221.py 31 228.py 33 222.py 34 223.py 34 224.py 34 225.py 35 226.py 36 227.py 37 229.py

```

Figura 65: Archivos de configuración creados

6.2.2 Asignación o eliminación de configuración para un día de la semana

En la configuración, cuando un usuario asigna una configuración a un día de la semana, se hace una llamada por AJAX a insert-config-day.php en la que se le pasa el id de la configuración (idconfig), el id de la cuenta de Instagram (igid) y el día de la semana al que se desea asignar (day).

A continuación, el PHP se conecta por SSH al servidor del bot y ejecuta edit_crontab_bot.sh pasándolo como parámetros el id de la cuenta de Instagram, el id de la configuración y el día de la semana para el que se va a programar.

Este script crea o elimina entradas en el crontab del servidor.

```

#!/bin/bash
#Funcionamiento: Programa en crontab un archivo de Instapy para ser ejecutado. Tambien elimina una programacion existente. Si para borrar solo se pasa como argumento una cuenta de Instagram, borra todos los crontabs asociados a ella. Si s
olo se pasa cuenta de ig y config, borra todos los crontabs asociados a esa config.
#Usage: edit_crontab_bot.sh [add/delete] (igid) (idconfig) (day)
#Ejemplo: edit_crontab_bot.sh 1 5 107 3
# ARGUMENTOS
# add/delete : 1->add 0->delete
# igid : Entero con el id de la cuenta de Instagram en 80
# idconfig : Entero con el id de la config o 0 si la config es nueva
# day : Entero con el dia. Dejar en blanco para eliminar todos los dias de la config dada

#Constantes
dir_configs="/root/ejecutablesInstaPy/webTFG/" #Directorio almacena configs

#variables
adddelete=1
igid=2
idconfig=3
day=4

if [ $adddelete == 1 ]; then #Añadimos nueva linea al cron
  dir_config_final=$(dir_configs${igid} ${idconfig}.py"
  linescron=$(($RANDOM % 20)) $(($RANDOM % 12)) * * $(day) python3.8 ${dir_config_final}"
  { crontab -l; echo "linescron:"; } | crontab -
else #Eliminar
  if [ -z "$4" ] && [ -z "$3" ];then #No hay argumento dia ni idconfig, se borran todos los crontabs para la cuenta de ig dada
    crontab -l | grep -v "python3.8 ${dir_configs}${igid}." | crontab -
  else
    dir_config_final=$(dir_configs${igid} ${idconfig}.py"
    [ -z "$4" ];then #No hay argumento dia, eliminar crontab de la config para cualquier dia
      crontab -l | grep -v "python3.8 ${dir_config_final}" | crontab -
    else #Eliminar crontab para un dia dado
      crontab -l | grep -v "$(day) python3.8 ${dir_config_final}" | crontab -
    fi
  fi
fi

```

Figura 66: Edit-crontab-bot.sh

A continuación, se muestra el resultado de las configuraciones programadas en el crontab del servidor del bot. La hora y el minuto de la ejecución se definen aleatoriamente para distribuir en la medida de lo posible la carga en el servidor una vez se ejecuten.

```

# For details see man 4 crontabs

# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name  command to be executed

#Clientes

37 4 * * 1 python3.8 /root/ejecutablesInstaPy/webTFG/31_219.py
8 7 * * 3 python3.8 /root/ejecutablesInstaPy/webTFG/31_219.py
0 7 * * 5 python3.8 /root/ejecutablesInstaPy/webTFG/31_219.py
23 3 * * 7 python3.8 /root/ejecutablesInstaPy/webTFG/31_219.py
25 10 * * 4 python3.8 /root/ejecutablesInstaPy/webTFG/31_221.py
2 20 * * 6 python3.8 /root/ejecutablesInstaPy/webTFG/31_221.py
44 17 * * 4 python3.8 /root/ejecutablesInstaPy/webTFG/33_222.py
54 13 * * 1 python3.8 /root/ejecutablesInstaPy/webTFG/33_222.py
28 13 * * 5 python3.8 /root/ejecutablesInstaPy/webTFG/33_222.py
10 4 * * 1 python3.8 /root/ejecutablesInstaPy/webTFG/34_223.py
17 8 * * 3 python3.8 /root/ejecutablesInstaPy/webTFG/34_224.py
4 10 * * 5 python3.8 /root/ejecutablesInstaPy/webTFG/34_225.py
40 16 * * 2 python3.8 /root/ejecutablesInstaPy/webTFG/31_228.py

```

Figura 67: Crontab



6.2.3 Desactivación de usuario cuando vence suscripción

Cuando la suscripción de un usuario asociada a una cuenta de Instagram vence, se debe desactivar tanto el acceso a la cuenta dentro de la aplicación web como la ejecución de las configuraciones del bot que hayan sido programadas.

Para llevar a cabo esta tarea, Stripe se conecta al webhook (se detalla su funcionamiento en el apartado 5.7.1) y comunica la expiración de la suscripción con el evento “customer.subscription.deleted”. El webhook recibe este evento y actúa en consecuencia actualizando el estado de la suscripción en la base de datos para bloquear el acceso a la misma desde la aplicación web.

```

187 case 'customer.subscription.deleted': //Una suscripción ha sido borrada(ha vencido)
188
189     $subscription = $event['data']['object'];
190     //Actualizamos status en BD
191     $stmt = $con->prepare( query: 'UPDATE subscriptions SET `status` = ? WHERE stripe_id = ? ');
192     $stmt->bind_param( types: 'ss', &var1: $subscription->status, &var2: $subscription->id);
193     $stmt->execute();
194     $stmt->close();
195
196     //Obtenemos igid de la cuenta asociada a esta suscripcion(puede ser nulo si no hay cuenta de ig asociada a la suscripcion)
197     if ($stmt = $con->prepare( query: 'SELECT ig_account_id FROM subscriptions WHERE stripe_id = ?')) {
198         $stmt->bind_param( types: 's', &var1: $subscription->id);
199         $stmt->execute();
200         $stmt->bind_result( &var1: $igid); //Asignamos los valores
201         $stmt->fetch();
202         $stmt->close();
203     }
204
205     if(!is_null($igid)){ //No es nulo, hay una cuenta de ig asociada a esta suscripcion
206         //Anulamos(comentamos) crontab de la cuenta de ig asociada a esta suscripcion
207         $sshcommand="ssh root@51.77.159.72 /bin/bot_comment_crontab.sh \"$1\" \"$igid.\"";
208         shell_exec($sshcommand);
209
210         //Eliminamos la asociacion de esta suscripcion a la cuenta de instagram
211         $stmt = $con->prepare( query: 'UPDATE `subscriptions` SET ig_account_id = NULL WHERE stripe_id = ? ');
212         $stmt->bind_param( types: 's', &var1: $subscription->id);
213         $stmt->execute();
214         $stmt->close();
215     }
216     break;

```

Figura 68: Stripe-webhook.php

Además, ejecuta por SSH el script del servidor del bot “bot_comment_crontab.sh” al que le pasa como parámetro el id de la cuenta de Instagram asociada a dicha suscripción.

Este script busca en el crontab las entradas para el id de la cuenta de Instagram dada y los comenta para desactivar su ejecución programada. Se comentan en lugar de eliminarse para que en caso de que el usuario posteriormente renueve la suscripción se pueda reactivar la ejecución de sus configuraciones.

```

#!/bin/bash
#Funcionamiento: Comenta o descomenta en crontab todas las líneas que ejecutan el bot para una cuenta de ig dada. Esto se hace para cuando a un user se le acaba la suscripcion, paralizamos el bot para su cuenta comentando las líneas. Si se
#olve a activar su suscripcion, las descomentamos.
#Usage: bot_comment_crontab.sh [comment/uncomment] [igid]
#Ejemplo: bot_comment_crontab.sh 1 1
# ARGUMENTOS
# comment/uncomment : 1 ->Comment 0->Uncomment
# igid : Entero con el id de la cuenta de Instagram en BD
#Variables
comment=1
igid=42

if [ $comment == 1 ]; then #Comentamos líneas en cron que pertenezcan a la cuenta de ig dada
    crontab -l | sed '/\/\*\*sigid\*\*/\*/\*/\*/g' | crontab -
else #Descomentamos líneas en cron que pertenezcan a la cuenta de ig dada
    crontab -l | sed '/\/\*\*sigid\*\*/\*/\*/\*/g' | crontab -
fi

```

Figura 69: Bot_comment_crontab.sh

6.2.4 Activación de usuario cuando reactiva suscripción

De forma análoga a lo expuesto en el apartado anterior, una vez el usuario reactiva la suscripción para una cuenta de Instagram, el webhook del servidor de la web recibe el evento “invoice.paid” con la nueva suscripción al que se le ha añadido como parámetro adicional usando la propiedad “metadata” del objeto “subscription” en el sistema de Stripe el id de la cuenta de Instagram que ha reactivado.

Una vez recibido, se da de alta la nueva suscripción en la base de datos y se le asigna la cuenta de Instagram que ha reactivado.

```

31 case 'invoice.paid': //Un pago acaba de ser realizado
32     $invoice = $event['data']['object'];
33
34     //Obtenemos id de stripe del usuario a partir de su customerid de stripe
35     if ($stmt = $con->prepare(query: 'SELECT id FROM accounts WHERE stripeid = ?')) {
36         $stmt->bind_param(types: 's', &var1: $invoice->customer);
37         $stmt->execute();
38         $stmt->bind_result(&var1: $accountid); //Asignamos los valores
39         $stmt->fetch();
40         $stmt->close();
41     }
42
43     //Obtenemos fecha fin de la suscripcion
44     $subscription_finish_date = $invoice->lines->data[0]->period->end;
45
46     //Comprobamos si la suscripcion existe
47
48     if ($stmt = $con->prepare(query: 'SELECT * FROM subscriptions WHERE stripe_id = ?')) {
49         $stmt->bind_param(types: 's', &var1: $invoice->subscription);
50         $stmt->execute();
51         // Store the result so we can check if the account exists in the database.
52         $stmt->store_result();
53         $filas_select = $stmt->num_rows;
54         $stmt->close();
55         if ($filas_select > 0) { //La suscripcion existe, ha sido renovada
56
57             //Obtenemos id de la suscripcion a partir de su stripeid. Servira para anadir el payment a BD y actualizar su finish_date
58             if ($stmt = $con->prepare(query: 'SELECT id FROM subscriptions WHERE stripeid = ?')) {
59                 $stmt->bind_param(types: 's', &var1: $invoice->subscription);
60                 $stmt->execute();
61                 $stmt->bind_result(&var1: $subscriptionid); //Asignamos los valores
62                 $stmt->fetch();
63                 $stmt->close();
64             }
65
66             //Actualizamos finish_date en BD
67             $stmt = $con->prepare(query: 'UPDATE subscriptions SET finish_date = FROM_UNIXTIME(?) WHERE id = ?');
68             $stmt->bind_param(types: 'ii', &var1: $subscription_finish_date, &var2: $subscriptionid);
69

```

Figura 70: Stripe-webhook.php

A continuación, mediante una llamada por SSH al script del bot “bot_comment_crontab.sh” indicando mediante un parámetro que la acción a realizar es



la de descomentar, se buscan las entradas en crontab correspondientes a la cuenta de Instagram en cuestión y se descomenta eliminando los corchetes al principio de línea (Ver Figura 69).

6.4 Pasarela de pago

Stripe es una empresa de procesamiento de pagos online que soporta tarjetas de todo el mundo [5]. Además, entre otros métodos soporta Google Pay y Apple Pay lo que lo hace muy interesante para facilitar los pagos desde dispositivos móviles.

Antes de escoger Stripe como pasarela de pagos se hizo una exhaustiva investigación [6] comparando las pasarelas más populares. Se optó por Stripe por sus bajas comisiones respecto a sus competidores (1'4%+0'25€ para tarjetas europeas y 2'9% + 0'25€ para tarjetas no europeas) y por su API, la cual tiene mucha documentación y es relativamente sencilla de implementar.

El trabajo de integración con Stripe requiere un trabajo previo de aprendizaje leyendo la documentación para entender como funciona internamente y en especial como funcionan las suscripciones en Stripe y cual es su ciclo de vida.

Una de las ventajas de Stripe es que permite almacenar muchos datos relativos a los pagos, suscripciones, tarjetas, etc. dentro de sus servidores, lo que aporta mayor seguridad para el comprador y tranquilidad para los administradores de la aplicación al estar exentos del control de estos datos.

El ciclo de vida de las suscripciones en Triunfalo es sencillo: Primero el usuario crea una suscripción cuando elige un plan y paga. Su tarjeta queda almacenada en los servidores de Stripe y la renovación de la suscripción es automática cada mes o cada tres meses según haya elegido el plan mensual o el trimestral. Si no se ha podido cobrar la suscripción por problemas en el método de pago como puede ser falta de fondos, que ha caducado, etc. se avisa al usuario por email en el que se le envía un enlace para que actualice su método de pago y finalice el mismo. Si no se finaliza el pago de renovación la suscripción expirará y pasará a estado cancelada. Esta suscripción habrá acabado su ciclo de vida y nunca más podrá volver a activarse. Si el usuario quiere volver a suscribirse más adelante, se crea una nueva.

Además, el usuario puede cancelar la renovación usando la página "Suscripciones y pagos" de nuestra web. Si la renovación está cancelada, no se harán nuevos intentos de cobro al usuario y por tanto cuando finalice su periodo pasará su estado a cancelada. Mientras la suscripción siga activa puede revertir la cancelación de la renovación en cualquier momento desde la misma página.

6.4.1 Webhook

Muchas de las comunicaciones entre Stripe y una aplicación web son síncronas. Como puede ser el proceso de creación y pago de una suscripción. Es un proceso donde mientras el usuario realiza determinada acción en la web, ésta se comunica con Stripe para trasladársela y Stripe le responde mediante JSON o códigos de error HTTP.

Pero hay otro tipo de comunicaciones que se deben producir y son asíncronas, es decir, el usuario puede no estar haciendo una acción para que se produzcan. Son eventos asíncronos por ejemplo el fin de una suscripción (se debe desactivar tanto el acceso como el bot) o el fallo en el cobro de la renovación de una suscripción.

Para poder gestionar este tipo de comunicaciones necesarias Stripe pone a disposición del desarrollador una herramienta muy potente: los webhooks.

Citando a Stripe, los webhooks son “la combinación de elementos que colectivamente crean un sistema de notificación y reacción dentro de una integración mayor” [7].

En esencia un webhook es un PHP en nuestro servidor donde Stripe se conectará y enviará información sobre un evento que acaba de ocurrir. Cuando ocurra un tipo de evento que se haya configurado para el webhook, un disparador en Stripe enviará por JSON al webhook información relativa al evento. Entonces el webhook procesará dicho evento y actuará en consecuencia.

En este proyecto es muy importante el evento que se dispara cuando finaliza una suscripción. Si tan sólo hubiese que restringir el acceso a la web y no importase no ser consistente entre la base de datos de la web y Stripe, no haría falta el Webhook, ya que simplemente habría que comprobar mediante la API de Stripe si la suscripción sigue vigente cada vez que se inicia sesión. En caso afirmativo se le permite el uso de la web y en caso negativo se le redirigiría a la pasarela de pago para renovar la suscripción.

No obstante, en este proyecto hay un bot independiente a la web que se está ejecutando recurrentemente todos los días de la semana que se hayan configurado. Sin webhooks, si el usuario no se loguease una vez su suscripción hubiese expirado, el bot seguiría ejecutándose indefinidamente incluso si la suscripción del usuario hubiese expirado. Sería así hasta que el usuario se loguease, en ese momento la web detectaría que la suscripción habría expirado y sería entonces cuando se comunicaría con el servidor del bot y daría la orden de borrar o comentar el crontab para que se detenga la ejecución recurrente.

Por eso los webhooks son vitales para el correcto funcionamiento de la aplicación y un control consistente de las suscripciones.



```

1  <?php
2
3  require_once './stripe-shared.php';
4
5  $event = null;
6
7  try {
8      // Make sure the event is coming from Stripe by checking the signature header
9      $event = \Stripe\Webhook::constructEvent($input, $_SERVER['HTTP_STRIPE_SIGNATURE'], $config['stripe_webhook_secret']);
10 }
11 catch (Exception $e) {
12     http_response_code( response_code: 403);
13     echo json_encode([ 'error' => $e->getMessage() ]);
14     exit;
15 }
16
17 $details = '';
18
19 $type = $event['type'];
20 $object = $event['data']['object'];
21
22 //if($type == 'checkout.session.completed') {
23 //    error_log(' Checkout Session was completed!');
24 //} else {
25 //    error_log(' Other webhook received! ' . $type);
26 //}
27
28 include 'mysql-connection.php'; //Conectamos con BD
29
30 switch($event['type']){
31     case 'invoice.paid': //Un pago acaba de ser realizado
32         $invoice = $event['data']['object'];
33
34         //Obtenemos id de stripe del usuario a partir de su customerid de stripe
35         if ($stmt = $con->prepare( query: 'SELECT id FROM accounts WHERE stripeid = ?')) {
36             $stmt->bind_param( types: 's', &var1: $invoice->customer);
37             $stmt->execute();
38             $stmt->bind_result( &var1: $accountid); //Asignamos los valores
39             $stmt->fetch();

```

Figura 71: Stripe-webhook.php

6.5 Desafíos técnicos

Durante el desarrollo de la aplicación se ha tenido que lidiar con diversos desafíos técnicos. A continuación, se muestran los principales:

6.5.1 Inestabilidad en el bot de Instagram

El bot usado es de código abierto y tiene una amplia comunidad de desarrolladores que colaboran en él. Así las actualizaciones son muy frecuentes y también las mejoras que se van desarrollando.

El bot lo que hace es conectarse mediante navegador web a Instagram, y mediante web scrapping localiza los botones y enlaces que necesita para acceder al contenido y simular el comportamiento de un usuario real. Para la localización de estos elementos utiliza desde las clases de las etiquetas html hasta la posición de estas (p.ej. para localizar un botón busca en el tercer div dentro del body).

Por tanto, cualquier cambio mínimo en la estructura de la web por parte de Instagram supone la inutilización completa del bot. No obstante, gracias a su activa comunidad se parchea rápidamente y aunque no esté el parche aplicado en la rama principal del proyecto, buscando en la sección de “Issues” en GitHub se suele encontrar la solución incluso en el mismo día de la aparición del problema.

Los cambios en la estructura, aunque no son muy frecuentes, ocurren de vez en cuando. Es por ello por lo que requiere que se preste atención al correcto funcionamiento del bot para detectar un mal funcionamiento rápidamente y parchearlo lo antes posible.

Además, Instagram puede implementar mecanismos de detección de bots que reduzcan la efectividad de estos o incluso los inutilicen por completo.

En un primer momento, cuando se empezó a usar el bot antes de iniciar el desarrollo de la presente aplicación, este funcionaba perfectamente. No obstante, en el momento en el que se escriben estas líneas, Instagram está combatiendo activamente los bots con inteligencia artificial que detecta patrones de uso sospechosos en la web y bloquea temporalmente las interacciones durante cierto periodo de tiempo.

Es posible que se solucione pronto por parte de la comunidad de desarrolladores del bot, cambiando los patrones de comportamiento del bot.

Aun así, se expone una de las principales debilidades de este tipo de desarrollos: El correcto funcionamiento no depende sólo del trabajo propio, también depende de la web o servicio con el que se trabaja. Éste puede poner trabas e implementar mecanismos para evitar que se utilice desde un servicio de terceros (en este caso el nuestro) llegando a poder inutilizarlo completamente o requiriendo un mantenimiento constante para que siga funcionando.

6.5.2 Problemas para obtener datos de Instagram

Para el dashboard principal de la aplicación web (llamado “Centro de operaciones”) es interesante obtener de Instagram datos de los usuarios como la foto de perfil, los usuarios a los que sigue y los usuarios que le siguen.

Se ha intentado obtener estos datos mediante código PHP. Por ejemplo, analizando las peticiones que hace Instagram cuando se visita la página de un usuario se observó que mediante una petición GET a `https://www.instagram.com/usuarioinstagram/?__a=1` se obtienen estos datos. Es por ello por lo que se implementó un script en PHP para extraer estos datos.

```
1 <?php
2
3 $urlinstagram = 'https://www.instagram.com/iguser/?__a=1';
4 $raw = file_get_contents($urlinstagram);
5 preg_match( pattern: '/\\"edge_followed_by\\":\s?\{\{"count\\":\s?([0-9]+)\}/', $raw, &matches $m); //Obtenemos en $m[1] los followers
6 echo "Followers: ".intval($m[1]);
```

Figura 72: Código extracción datos Instagram

Sin embargo, aunque las primeras veces funciona correctamente y se obtienen los datos deseados, cuando se hacen peticiones de forma sistemática a esta url los

servidores de Instagram detectan un comportamiento sospechoso y dejan de mandar los datos JSON del usuario.

Como se puede observar, los mecanismos de Instagram para bloquear su uso desde terceras aplicaciones surten efecto impidiendo un uso estable y prolongado de esta técnica.

Se ha intentado obtener esta información de distintas formas, pero siempre con el mismo resultado. Al principio funciona bien, pero una vez los servidores de Instagram marcan la IP como sospechosa deja de mandar la información durante un periodo de tiempo de bloqueo.

7 Validación

A lo largo del proceso de desarrollo se han realizado manualmente pruebas en el producto. Después de cada modificación se accedía a la web, se comprobaba su correcto funcionamiento y se probaba la nueva implementación de todas las formas posibles buscando bugs o errores. Por ejemplo, en el formulario de registro se introducían caracteres no permitidos, se intentaba registrar usuarios ya registrados, se redimensionaba la ventana para comprobar la correcta implementación responsive, etc.

Una vez completado el primer MVP, se propone hacer un experimento con un grupo reducido de 6 *early adopters* a los que se les dejará usar el producto sin darles ninguna indicación respecto a su funcionamiento y posteriormente se les pasará un cuestionario para que dejen su feedback sobre el producto.

En la Figura 73 se muestra el cuestionario que se ha diseñado para los *early adopters*.

Cuestionario de uso Triunfalo.es

Por favor pedimos sinceridad ante todo, tus respuestas ayudarán a mejorar nuestro producto.

PREGUNTA	NO	POCO	REGULAR	BASTANTE	SI
¿Has entendido en qué consiste el producto y lo que te ofrece después de leer la página principal?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
¿Consideras visualmente atractiva la web?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
¿Ha sido sencillo configurar el bot?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
¿Crees conveniente añadir un videotutorial?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
¿Crees que te suscribirías al producto si tuvieses un negocio que tuviese cuenta en Instagram?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
¿Consideras que son justos los precios ofrecidos?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
¿Preferirías pagar un poco más y que un profesional te lo configure?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

OBSERVACIONES/SUGERENCIAS (Puede continuar en el reverso si es necesario)

Figura 73: Cuestionario MVP1

Con los *early adopters* que han podido se les ha hecho el experimento in-situ. Con los que por motivos de tiempo no, se les ha dado en papel el cuestionario y han hecho la prueba ellos solos.

Ha habido un total de 6 *early adopters* entre los que se incluyen perfiles muy variados con distintos niveles de estudios. Son 3 chicos y 3 chicas con edades comprendidas entre los 22 y los 26 años. Es interesante este análisis para poder comprobar que el nivel de educación no supone un impedimento para entender la mecánica de la aplicación y que se puedan desenvolver correctamente con ella personas de perfiles diversos. Se han seleccionado personas que van desde un alto nivel de estudios (grado, máster con un doctorado en curso) a personas sin estudios.

Después de la realización de la prueba, a continuación se detallan los resultados obtenidos, los cuales se muestran en la Figura 74.

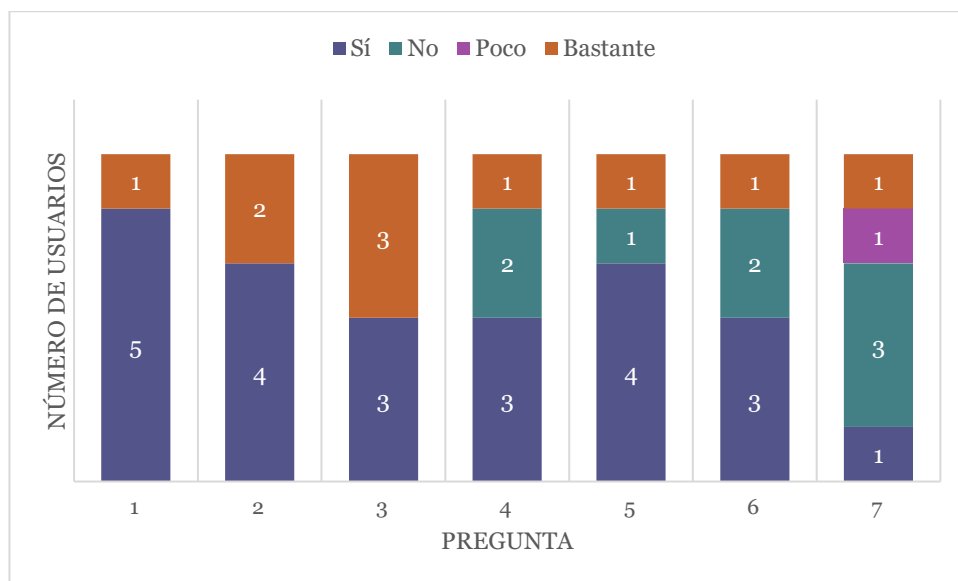


Figura 74: Resultado de la evaluación con *early adopters*

La primera pregunta “¿Has entendido en qué consiste el producto y lo que te ofrece después de leer la página principal?” tenía por objeto comprobar si la página principal transmite la misión de la aplicación web y su funcionamiento con suficiente claridad. Tal y como muestra la Figura 74, los valores obtenidos para esa pregunta (Sí: 5, Bastante: 1) nos permiten concluir que el contenido de la página principal cumple con su objetivo y no requiere modificaciones.

La segunda pregunta “¿Consideras visualmente atractiva la web?” estaba enfocada a la vistosidad de la interfaz de usuario. La respuesta (Sí: 4, Bastante: 2) es muy positiva y cumple con las expectativas, ya que proporcionar una interfaz agradable es muy importante para que los usuarios se sientan cómodos usando la aplicación.

La tercera pregunta era “¿Ha sido sencillo configurar el bot?”. Las respuestas (Sí: 3, Bastante: 3) han sido más positivas de lo esperado, ya que, aunque la interfaz se ha desarrollado pensando en brindar al usuario una experiencia lo más intuitiva posible, se suponía que un usuario que nunca ha utilizado una aplicación similar requeriría

una pequeña formación en forma de tutorial o videotutorial para poder usarla de forma sencilla.

La cuarta pregunta “¿Crees conveniente añadir un videotutorial?” está estrechamente relacionada con la pregunta anterior. Está en los planes de desarrollo la inclusión de un videotutorial para facilitar el uso de la aplicación. Aunque hay cierta discrepancia (Sí: 3, Bastante: 1, No: 2). predomina la afirmación de la conveniencia de un videotutorial. Y teniendo en cuenta que este es un recurso que solo será utilizado en la medida que el usuario lo necesite se sigue adelante con los planes de su incorporación a la web.

La quinta pregunta era “¿Crees que te suscribirías al producto si tuvieses un negocio que tuviese cuenta en Instagram?”. El feedback recibido (Sí: 4, Bastante: 1, No: 1) ha sido muy positivo teniendo solo un usuario de los 6 encuestados que no se suscribiría al servicio.

La sexta pregunta “¿Consideras que son justos los precios ofrecidos?” tenía el objeto de conocer la impresión que generan los precios que se ofrecen en la web. Hay que tener en cuenta para esta pregunta que se le ha permitido a los encuestados consultar los precios que ofrecen otros productos competidores. Las impresiones han sido muy positivas (Sí: 3, Bastante: 2, No: 1). Teniendo además en cuenta que el precio del servicio está muy por debajo de la media, se decide mantenerlo.

La séptima y última pregunta “¿Preferirías pagar un poco más y que un profesional te lo configure?” tenía por objeto analizar la preferencia de los usuarios de dejar en manos de un profesional la configuración del servicio para sus cuentas, ahorrándoles tiempo y asegurándose una configuración adecuada, a cambio de un pequeño incremento en las cuotas. Los resultados han sido discrepantes, aunque la tendencia se acerca al No (Sí: 1, Bastante: 1, Poco: 1, No: 3). Está en los planes de mejora de la aplicación implementar esta opción para satisfacer a los usuarios que prefieran una configuración profesional y sin complicaciones, sin embargo, la prioridad para la implementación de este servicio es baja ya que se considera un servicio adicional. Si los encuestados hubiese creído muy necesario este servicio se hubiese incrementado la prioridad de su desarrollo. Sin embargo, una vez analizados los resultados se mantiene como característica secundaria en los planes de desarrollo.

En el apartado final los encuestados podían añadir cualquier tipo de sugerencia o observación que creyeran conveniente. Mediante este feedback se ha identificado los siguientes problemas:

- Se ha detectado en primer lugar un bug en el proceso de registro donde los usuarios no recibían ningún mensaje de confirmación una vez pulsaban en el botón “Registrarse” tras haber cumplimentado los datos del formulario. Dos usuarios han reportado este bug y se ha corregido tan pronto como han hecho llegar sus encuestas.
- Además, dos usuarios han recalcado la necesidad de la creación de un videotutorial ya que no han conseguido asignar las configuraciones a los días de la semana. Esta acción se realiza mediante drag&drop y se suponía que iba a ser poco intuitivo para un usuario nuevo ya que sólo se informa de esto en la página principal. Se reafirma la necesidad del videotutorial y además se va a añadir un pequeño mensaje en la página

de configuración indicando que se arrastren las configuraciones a los días de la semana para que sean asignados a ellos.

- Respecto a los planes de suscripción, un usuario ha indicado “No llego a comprender la diferencia de precio ente principiante y profesional con la descripción adjunta”. La única diferencia es que el plan “Principiante” es mensual y el “Profesional” es trimestral y este último tiene un ahorro del 25%. Esto se indica en la descripción de los planes, pero se estudiará una modificación en su presentación para que la diferencia de precio entre ambos se observe de un sólo vistazo por ejemplo añadiendo una etiqueta de “-25%” en la parte superior del contenedor del plan profesional.
- Otro usuario ha recomendado la inclusión de más métodos de pago como PayPal o Bizum. Esta mejora ya se ha estudiado, pero tiene carácter secundario ya que el método de pago más utilizado es la tarjeta de crédito. Aun así, se contempla en el plan de desarrollo una vez la aplicación sea lanzada al mercado.



8 Conclusiones

Este proyecto me ha brindado la oportunidad de sumergirme en el mundo empresarial y conseguir transformar lo que un día fue una simple idea a un servicio web real, completamente funcional y con perspectivas de ser lanzado al mercado en los meses siguientes a esta presentación.

Se han cumplido los dos objetivos propuestos. En primer lugar, el estudio de la viabilidad comercial me ayudó a descartar algunas funciones que estaban planeadas implementar en el servicio y añadí otras que solo después de dicho estudio entendí que eran muy relevantes. Por otro lado, el desarrollo de la versión preliminar del servicio se ha completado, siendo completamente funcional.

Cabe destacar que dicho desarrollo me ha ayudado a ampliar notablemente mi conocimiento sobre el desarrollo web y la administración de servidores Linux. También he aprendido a desarrollar scripts en Bash. Pero lo más importante que me ha dado este proyecto son las ganas de superarme a mí mismo y seguir creciendo profesionalmente.

Desde bien pequeño he tenido el sueño de emprender, de ser mi propio jefe y de aportar algo al mundo. De crear soluciones a problemas o mejorar las ya existentes.

El éxito en el mundo del emprendimiento está reservado para un pequeño porcentaje de todos los valientes que se atreven a adentrarse en él. Sin embargo, eso no me frena. Al contrario, me da alas para seguir esforzándome y luchando por lo que quiero.

Como dijo Jean Paul Marat “No existe el fracaso, salvo cuando dejamos de esforzarnos”.

9 Bibliografía

1. SMPERTH (2020). *Facts & Figures // Instagram Statistics for 2021*. Recuperado el 14 de enero de 2021, de SMPERTH:
<https://www.smpertth.com/resources/instagram/instagram-statistics/>
2. Martín, J. (2019). *Claves para hacer un buen DAFO*. Recuperado el 18 de enero de 2021, de CEREM:
<https://www.cerem.es/blog/claves-para-hacer-un-buen-dafo-o-foda>
3. INCAE (2018). *Lean Canvas un lienzo para emprendedores*. Recuperado el 18 de enero de 2021, de INCAE Business School:
<https://www.incae.edu/es/blog/2018/05/21/lean-canvas-un-lienzo-para-emprendedores.html>
4. Kononow, P. (2018). *How to create ER diagram for existing MySQL database*. Recuperado el 23 de febrero de 2021, de Dataedo:
<https://dataedo.com/kb/tools/mysql-workbench/create-database-diagram>
5. Navarro, G. *Stripe : la pasarela de pago*. Recuperado el 1 de junio de 2021, de GonzaloNavarro:
<https://gonzalonavarro.es/blog/198-que-es-stripe/>
6. MyChoice2Pay (2019). *Guía para saber elegir una pasarela de pago*. Recuperado el 1 de junio de 2021, de MyChoice2Pay:
<https://www.mychoice2pay.com/es/blog/elegir-una-pasarela-de-pago>
7. Stripe. *Webhooks*. Recuperado el 2 de junio de 2021, de Stripe:
<https://stripe.com/docs/webhooks>
8. Zarrelli, G. *Mastering Bash*. Packt Publishing, 2017.
9. Boduch A., Chaffer J., Swedberg K. *Learning jQuery 3*. Packt Publishing, 2017.
10. Helmke M., Sobell M. *A practical guide to Linux commands, editors, and shell programming*. Addison-Wesley Professional, 2017.