

SpaceBoard

Santiago Salvador

Jovani San Saturnino

Resumen:

Este proyecto trata de abarcar un nuevo enfoque sobre el trabajo en plataformas web mediante el uso de un espacio de trabajo personalizable en el cual el usuario podrá ejecutar, distribuir, interactuar y almacenar contenedores similares a ventanas propias de un escritorio.

La mecánica de trabajo se realizará por medio de iframes, unos elementos que permiten obtener recursos de distintas páginas webs, los cuales el usuario podrá distribuir por la pantalla como desee, y luego almacenar su estado para verlo desde otro dispositivo o en otro momento.

La aplicación a su vez cuenta con un sistema para gestionar distintos usuarios, además de poder personalizar el espacio de trabajo al gusto de cada uno.

Abstract:

In this project I'm trying to implement a new idea for web users to improve spending time on webpages. This project is based on a content = box display where the user can access multiple content simultaneously and resize or/adjust it to their personal taste.

The idea is also that users can log into the page to save their workspace and display it on other devices or in any other moment. With this feature, users can save the properties of the boxes they've created and the content within them, to re-display them later.

Índice de capítulos:

1. Introducción.
2. Objetivos.
3. Estado actual del problema.
4. Solución propuesta.
 - 4.1. Análisis de requisitos.
 - 4.2. Tecnologías que se usan.
 - 4.3. Modelos.
 - 4.4. Experimentación.
 - 4.5. Requisitos de funcionamiento
 - 4.6. Conclusiones.
5. Estudio de mercado
 - 5.1. Fuentes de financiación
 - 5.2. Marketing mix
6. Trabajos futuros.
7. Bibliografía.

Índice de Tablas:

- Objetivos (Página 7)
- DAFO (Página 17)

Indice de Imagenes

- Diagrama Entidad - Relación (Página 10)

- Diagrama de casos de uso (Página 10)

- Diagramas de Flujo:
 - Crear Caja (Página 11)

 - Guardar Cajas (Página 11)

 - Cargar Cajas (Página 12)

 - Iniciar Sesión (Página 12)

 - Crear Usuario (Página 13)

Introducción:

Esta memoria detalla el trabajo realizado por Santiago Salvador Jovani San Saturnino así como sus implicaciones para la asignatura de Proyecto de Fin de Ciclo del Grado superior de Desarrollo de Aplicaciones Web.

Certifico que todo el código así como esta memoria ha sido desarrollado por mi, con ayuda de los manuales pertinentes y Jose Luís, que ha tenido que soportar casi dos meses mi cabezonería y mis continuos correos pidiendo ayuda.

En esta memoria describiré el trabajo realizado así como los pensamientos y enfoques que me hicieron llegar a este resultado final. Dejo en constancia que pese a que han sido varios meses de trabajo, el proyecto presenta claros fallos y pequeñas cosas que por falta de tiempo no he podido dejar del todo a mi gusto.

Por favor, sed indulgentes y que todo este trabajo haya servido al final para algo.

Objetivos:

General

Esta página permite la creación de elementos dinámicos que por medio de distintos botones sumados a la URL o al código <iframe> que pegues en estos, puedan disponerse sobre un fondo, y puedas interactuar, redimensionar y almacenar la posición de los mismos.

Concretos

1. Presentar las páginas en el "Corcho" y que se puedan mover y redimensionar sobre este.
2. Hacer un sistema de usuarios funcional desde el que puedas acceder en equipos distintos.
3. Posibilidad de almacenar dentro del usuario el "Escritorio virtual" y poder volver a cargarlo si se requiere del mismo.

| Conseguidos | No conseguidos |
|------------------------|--------------------------|
| Cajas redimensionables | Favoritos |
| Cajas móviles | Ventanas navegador |
| Sesiones de usuario | Soporte varias pantallas |
| Almacenar pizarra | |

Estado actual del problema:

La navegación web lleva estancada en una única página visualizada al mismo tiempo desde hace casi 30 años. Este enfoque es poco práctico, pues la solución que se presentó para ver varias páginas web fue la implementación de pestañas. Es una solución inteligente, pero no arregla el problema realmente del visionado simultáneo de distintos contenidos.

En mi antiguo trabajo la gran mayoría de usuarios trabajaba desde terminales “tontos”. Unos pequeños ordenadores que contaban con una pantalla, periféricos y la CPU que únicamente se conectaba con el servidor y daba acceso a las aplicaciones web de la empresa.

Personalmente estos aparatos me parecían horribles, pues como pasa cuando te acostumbras a trabajar con varias pantallas simultáneamente y de repente utilizas una sola, la experiencia se vuelve exasperante, te falta espacio por todos lados y le coges manía a tener que cambiar de pestaña todo el rato.

Si a esto además le quitamos la opción base de poder trabajar con varias ventanas y distribuirlas en el (pequeño) espacio de una pantalla, bueno, ojalá no lo tengais que experimentar.

Solución propuesta:

La idea de mi proyecto viene de ahí. Pretendo de alguna manera dar más libertad y personalización del espacio para los usuarios por medio de la carga de contenidos en algo similar a ventanas pero en un entorno web.

La solución que presento, pasa por la utilización de la página web que he desarrollado como “lienzo” y cargar los contenidos de las demás por encima.

Para esto simplemente sería necesario que el usuario contara con un ordenador conectado a internet y un navegador de los últimos 3 años.

Las tecnologías utilizadas en este proyecto, son:

- Angular: Framework de aplicaciones web destinado para la creación y el mantenimiento de proyectos web de una sola página.
- Firebase: Plataforma de desarrollo de aplicaciones web destinada al guardado de datos.
- JQuery: Librería multiplataforma basada en JavaScript usada para simplificar el manejo del Document Object Model (DOM), la interfaz cara al público de la web.

Modelos:

Diagrama Entidad - Relación:

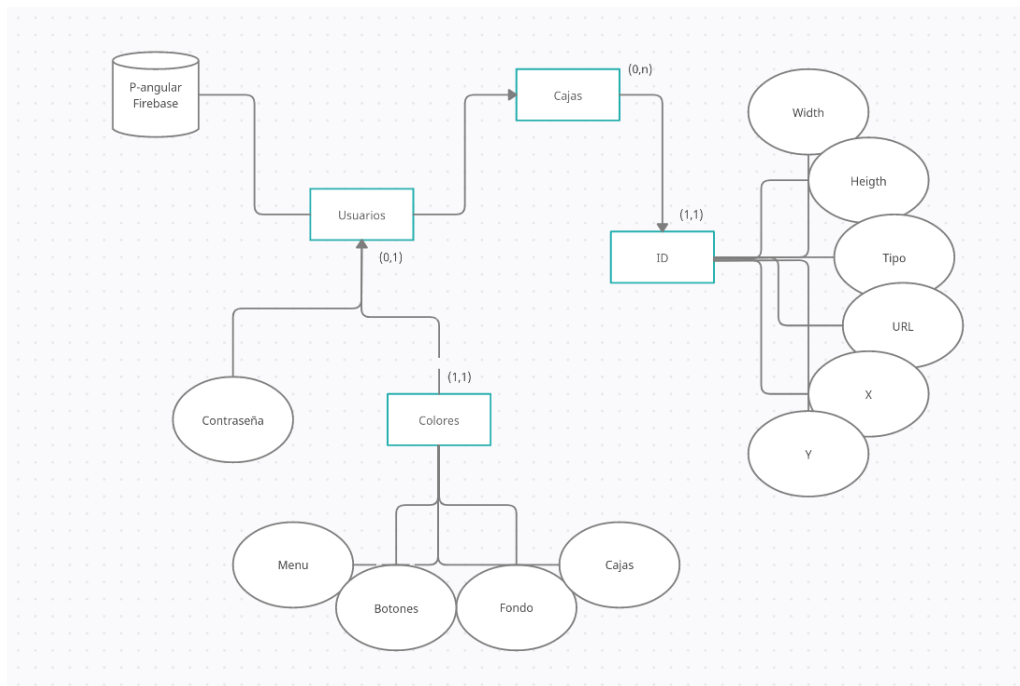
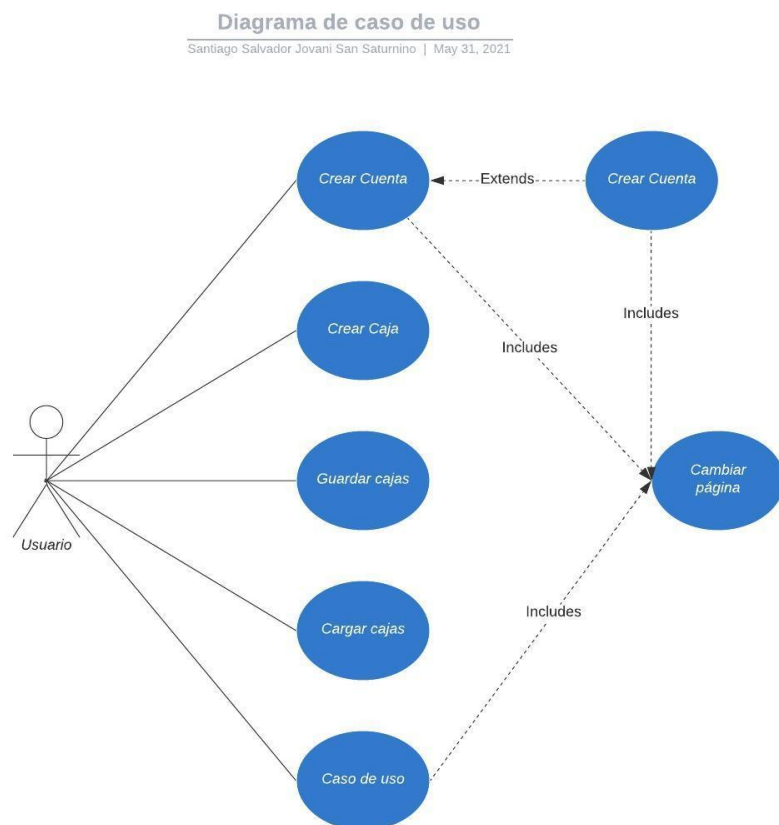
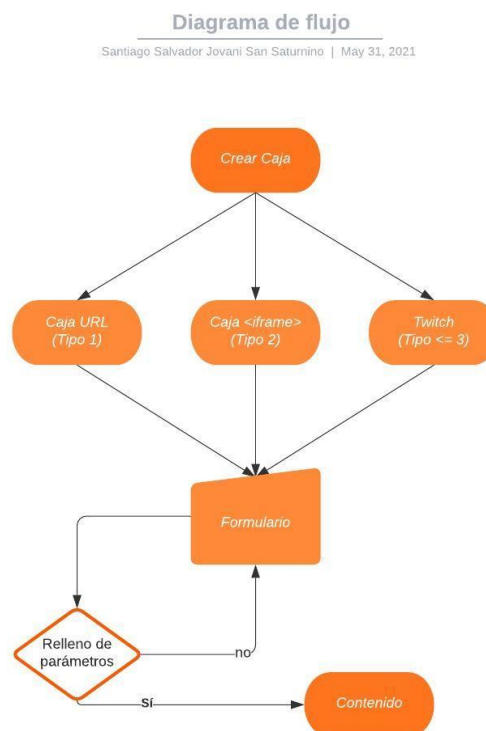


Diagrama de casos de Uso:

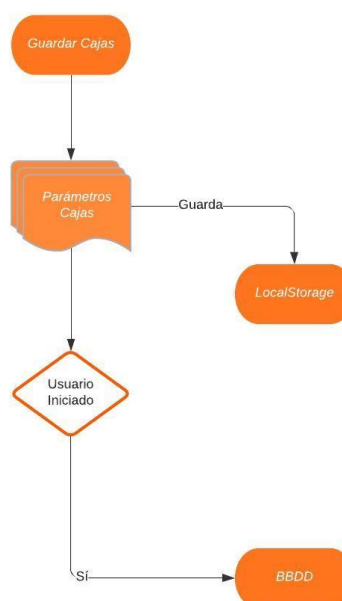


Diagramas de flujo:

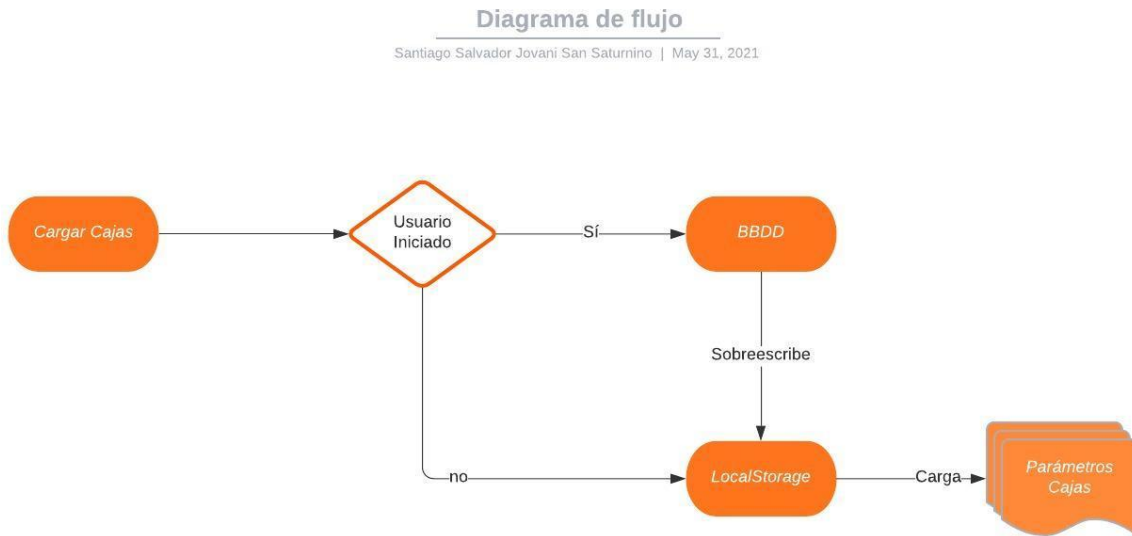
- Crear caja



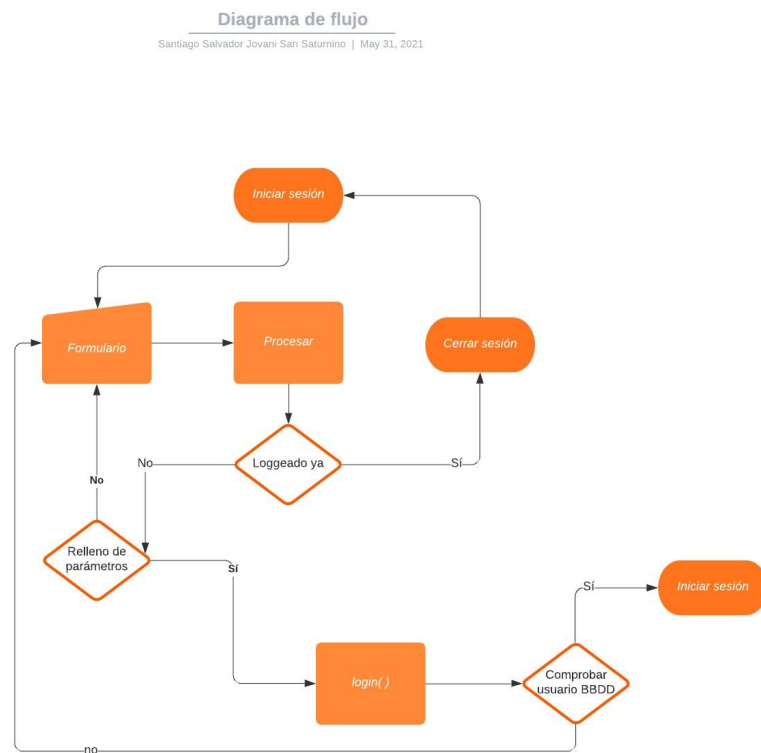
- Guardar cajas:



- Cargar cajas:



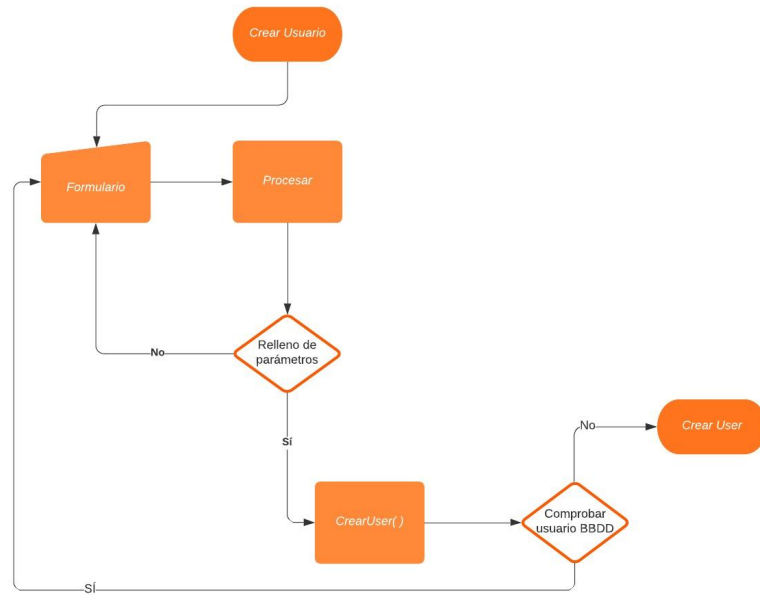
- Iniciar sesión:



- Crear usuario:

Diagrama de flujo

Santiago Salvador Jovani San Saturnino | May 31, 2021



Experimentación:

En este apartado explicaré mi proceso creativo por medio de los commits existentes en gitHub:

1. **Initial Commit:** Este fué el inicio del proyecto, cuando ya tuve claro que quería hacer, hice un par de pruebas con JavaScript + JQuery sin mucho éxito y antes de empezar a marear a Jose Luis, decidí que los fallos que tenía eran por el refresco de la página, por tanto me decidí a utilizar el framework de Angular.
2. **A:** Fué un commit de prueba cuando añadí la extensión a visual studio code.
3. **Error Crear caja:** En este commit añadí el grueso de la creación inicial de las cajas junto con un css vago para hacer pruebas.
4. **Interfaz:** En este commit cree los componentes maestros del proyecto, Board y Menú y decidí darle una aproximación vertical al componente menú como había visto en uno de los placeholders de Ionic.
5. **Crear Frames:** En este commit, conseguí hacer la primera aproximación de lo que luego sería el código del componente (Crear frame) en el que maqueto la entrada del usuario y creo un contenido de tipo iframe con los datos.
6. **Ayuda 1:** Con este commit, empecé a darme cuenta gracias a los insistentes comentarios de Jose Luís que realizar esto con JavaScript iba a ser complicado, pero como cabezón soy un rato, seguí intentando hacer que funcionara.
7. **Cajas Funcionan:** Conseguí hacer que las cajas crearán el contenido que se les pasaba por form.
8. **Lightmode:** Empecé a plantearme que el diseño del menú quedaba algo vacío y pensé en rellenarlo con un pequeño metodo para cambiar de color.
9. **Responsive:** Creación de falsas media queries para mover el contenido de MenuComponent de la posición vertical a horizontal para pantallas más pequeñas y tablets.

10. **Ayuda 2:** Me doy cuenta que JQuery no tiene soporte para inputs de tipo touch y que tengo que instalar una librería extra, pero no me va porque hay un problema de rutas.
11. **Caja component:** Después de la insistencia de Jose Luís al ver que no era buena práctica utilizar Angular para crear contenido dentro del DOM, acabo por hacerle caso y dedico el fin de semana completo a traducirlo a componentes particulares.
12. **Cajas funcionales:** Ahora las cajas crean bien el contenido dentro de ellas y la app empieza a tomar forma.
13. **Routing:** Pienso que tendría que hacer una página de manual así como la futura login y la pagina de cambio de color. Y acabo por hacer routing.
14. **LoadCajas a medias:** Primer intento de método para capturar las cajas, guardarlas y volver a crearlas.
15. **LoadCajas bien y 404:** Método funcional y componente 404.
16. **Message component:** Creación de la página de info.
17. **Login init:** Primer intento de login después de haber desistido de realizar las cosas con PHP.
18. **Final GetCajas:** GetCajas por fin escribe bien en la BBDD.
19. **Final:** LoadCajas funciona correctamente y el proyecto está terminado.
20. **F:** Cambios menores en css.

Requisitos de funcionamiento:

Tener instalado NPM y Angular.

Instalación de npm para windows:

1. Nos dirigimos a la página de descarga de NodeJs y lo descargamos e instalamos
 - a. <https://nodejs.org/es/>
 - b. Cuando lo tengamos instalado desde un terminal podemos comprobar la versión con:
 - i. `node -v`
 - ii. `npm -v`
2. Ahora desde una terminal escribiremos los siguientes comandos:
 - a. `npm remove -g @angular/cli` (solo si había una versión instalada previamente)
 - b. `npm install -g @angular/cli`
 - c. Podremos comprobar la versión con:
 - i. `ng --version`
3. Descargamos el instalador de Git desde su página.
 - a. <https://git-scm.com/downloads>
 - b. Lo instalamos
 - c. Iniciamos sesión con los siguientes comandos:
 - i. `git config --global user.name "nombre de usuario"`
 - ii. `git config --global user.email "tu email"`
4. Con esto lo único que tendremos que hacer es clonar el repositorio:
 - a. `git clone https://github.com/DrackPyros/P-Angular.git`
5. Cuando hayamos hecho este último paso, tendremos que en el directorio en el que hayamos importado el repositorio, hacer:
 - a. `ng install`
6. Y por último para mostrar la página:
 - a. `ng serve -o`

Estudio del Mercado:

Viendo la aplicación desde el punto de vista de cara al mercado, lo que proporcionamos es un servicio al consumidor el cual pueda acceder de manera libre para mejorar su ritmo de trabajo o su ocio. Como tal, la aplicación pretende llegar al mayor número de usuarios posible, pero es necesario un mantenimiento así como pagar el alojamiento web y varios servicios que gastamos en la aplicación son de prueba.

Por lo tanto, propongo la idea de pago por suscripción al servicio, en el cual, en caso de que se llevara al mercado, se plantearía añadido a por ejemplo, un sistema de almacenamiento en la nube.

| Fortalezas | Debilidades |
|-----------------------------|--------------------------------------|
| Multiplataforma | Contenido dependiente de web maestra |
| Personalización del entorno | Slots de guardado |
| Multiusuario | Ausencia de navegador |
| Oportunidades | Amenazas |
| No competidores | Ancho de Banda |
| Innovación | Popularidad / Publicidad |
| | Competidores |

Fuentes de financiación y estudio de gastos:

Como fuentes de financiación inicialmente, solicitaría un préstamo de 10000€ a BBVA a un interés del 5% a devolver en 24 meses para pagar los costes iniciales del hosting, el plan inicial de iframely y Firebase, e iniciar una campaña de publicidad dirigida a grandes y pequeñas empresas y autónomos.

El cálculo ha sido avalado por la tía de Gerard Mirallles que ha trabajado en una sucursal de este banco durante 20 años

| Activo | Pasivo |
|---|--|
| Activo no corriente: • Mobiliario 0€ | Patrimonio neto: • Capital 1000€ |
| Activo corriente: • Bancos 0€ | Pasivo corriente: • Préstamo 10000€ |
| TOTAL ACTIVO: 0€ | TOTAL PASIVO: 11000€ |

Gastos ordinarios primer año:

- Hosting + Domino: 6 meses = 1€/mes luego 3€/mes = 24€ Ionos
- Plan inicial iframely API: 29\$/mes = 384\$ * 0.82 = 314.88€
- Plan inicial Firestorm: 12\$/mes = 144\$ * 0.82 = 118.08€
- Cuota autónomos: 60€

Total consumos de explotación : 516.96€

Total de RRHH: Beneficios variables

Total de servicios externos:

- Préstamo: 2000€

Total gastos ordinarios IVA incluido: 2516.96€

Ingresos previstos:

Beneficios variables - total gastos ordinarios

Marketing Mix:

Producto:

Nuestra aplicación cumple con los estándares de producto de la siguiente manera

- Producto Básico:
Satisface la necesidad de la visualización simultánea de contenido web.
- Producto Formal o Añadido:
Como es un servicio, la duración del producto se basa en el uso que se quiera dar a la plataforma, nuestra marca, únicamente se compone de esta aplicación. Y como características técnicas adicionales, ofrecemos una personalización del entorno y la posibilidad de utilizar este servicio en distintos dispositivos.
- Producto Ampliado:
Como servicio adicional, ofrecemos mantenimiento y un correo para dudas y gestión de errores.

El producto se encuentra en fase de introducción (Obviamente), por el bajo volumen de ventas y la gran inversión tanto técnica, como comercial y de comunicación.

Precio

Como hemos explicado en el punto anterior, nos saldrían unos gastos de unos 210€ mensuales. Por lo tanto, teniendo en cuenta que dependemos de la publicidad para atraer posible clientela a la página. El precio de la cuota mensual, lo pondremos orientativamente a 19.95€/mensualidad por cuenta, con planes especializados según paquetes de licencias.

Entonces a partir de este precio orientativo, hemos calculado:

$$\text{precio venta} = \text{coste total mensual} / \text{suscripciones mensuales}$$

$$19.95 = 210 / s$$

$$s = 210 / 19.95$$

$$s = 10.5 \rightarrow 11 \text{ usuarios}$$

A partir de 11 usuarios por mes, no tendríamos pérdidas. *Que suerte que no tengamos competencia.*

Distribución

Como canales de distribución contaremos con un canal directo puesto que nos encargaremos de dar las licencias directamente. La web estará hospedada en un servidor, en internet, por lo tanto, cualquier persona podrá contratar nuestros servicios en cualquier momento.

Publicidad

Para publicitar la página web contactaremos con una agencia publicitaria y crearemos una serie de cuentas en redes sociales, en las que mostraremos nuestra página y los servicios que ofrecemos a nuestro público objetivo, enlazaremos nuestra página de información en la que se detallará todo lo importante y haremos periodos de prueba para que la gente pueda iniciarse en la aplicación gratuitamente.

Además nos comunicaremos directamente con empresas y ofreceremos nuestros servicios.

Trabajos futuros

1. Terminar el componente de cambiar colores:
 - a. Guardando los resultados en BBDD.
 - b. Añadiendo un cuadro de canvas unico para la selección de color.
 - c. Cambiando la disposición de la ventana.
2. Borrar las URL de las cajas.
3. Cambiar el tipo de resposive según el contenido que se crea.
4. Añadir un scroll dentro de las cajas si el contenido sobresale.

Bibliografía

- Diagramas de LucidChart: <https://www.lucidchart.com/>
- Repositorio con el código: <https://github.com/DrackPyros/P-Angular>
- Información de Autónomos: <https://www.agenciatributaria.es/>
- Firebase: <https://firebase.google.com/?hl=es>
- Angular: <https://angular.io/docs>
- JQuery: <https://api.jquery.com/>
- CSS Tricks: <https://css-tricks.com/>