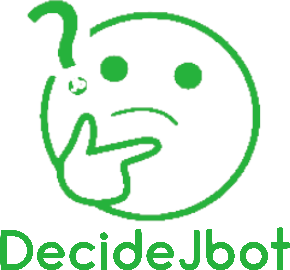


PROYECTO FIN DE CICLO



Autor

Julián Bienzobas Arnas

Tutor

Michel Cuartero Cimorra

DESARROLLO DE APLICACIONES MULTIPLATAFORMA

2021/2022

**Resumen**

Este TFG (Trabajo de Fin de Grado) ha sido realizado por Julián Bienzobas Arnas, estudiante del Grado Superior de Desarrollo de Aplicaciones Multiplataforma en el IES Tubalcain de Tarazona en el curso 2021/2022.

El proyecto titulado “DecideJbot” consta de dos aplicaciones, una aplicación móvil para Android y una aplicación Web desarrollada con el Framework Angular.

El proyecto se basa en funciones enfocadadas al azar, siendo las aplicaciones un instrumento que toma “decisiones” aleatoriamente por el usuario.

Las aplicaciones realizan funciones aleatorias como pueden ser la realización de un sorteo, el lanzamiento de un dado o una moneda, la rotación de una flecha o la generación de números aleatorios.

**Palabras clave**

Decisión, Sorteo, Participantes, Lista, Ruleta, Premios, Ganador, Actividad, Android, HTML, Javascript, Persistencia, Base de datos, Aleatorio, Historial, Resultados, Java, Azar, Rifa, Angular, Lotería, Moneda, Cara o Cruz, Dado, Flecha, Ganador, Premio, Elección, CSS, Android Studio, Visual Studio Code.

**Abstract**

This End of Degree Project has been carried out by Julián Bienzobas Arnas, student of the Higher Degree in Multiplatform Application Development at the IES Tubalcain of Tarazona in the 2021/2022 academic year.

The project entitled "DecideJbot" consists of two applications, a mobile application for Android and a Web application developed with the Angular Framework.

The project is based on randomly focused functions, with applications being an instrument that makes "decisions" randomly for the user.

Applications perform random functions such as drawing a lottery, rolling a die or coin, rotating an arrow, or generating random numbers.

**Keywords**

Decision, Draw, Participants, List, Roulette, Prizes, Winner, Activity, Android, HTML, Javascript, Persistence, Database, Random, History, Results, Java, Chance, Raffle, Angular, Lottery, Coin, Heads or Tails, Dice, Arrow, Winner, Prize, Choice , CSS, Android Studio, Visual Studio Code.

**TABLA DE CONTENIDOS**

[Descripción del proyecto 5](#_Toc105602081)

[**Contexto del proyecto** 5](#_Toc105602082)

[**Ámbito y entorno** 5](#_Toc105602083)

[**Análisis de la realidad** 6](#_Toc105602084)

[**Solución y justificación** 7](#_Toc105602085)

[**Destinatarios** 8](#_Toc105602086)

[**Objetivo del proyecto** 9](#_Toc105602087)

[**Objective of the Project** 10](#_Toc105602088)

[Acuerdo del proyecto 11](#_Toc105602089)

[**Requisitos funcionales y no funcionales** 11](#_Toc105602090)

[**Tareas** 12](#_Toc105602091)

[**Metodología a seguir para la realización del proyecto** 14](#_Toc105602092)

[**Planificación temporal de tareas** 14](#_Toc105602093)

[**Presupuesto** 15](#_Toc105602094)

[Análisis y diseño 16](#_Toc105602095)

[**Modelado de datos** 16](#_Toc105602096)

[**Análisis y diseño del sistema funcional** 16](#_Toc105602097)

[**Análisis y diseño de la interfaz de usuario** 17](#_Toc105602098)

# **Descripción del proyecto**

## **Contexto del proyecto**

La idea de DecideJbot surge a raíz de la indecisión y como un reto personal.

Debido a ciertas indecisiones, allá por el año 2019 pasó por mi cabeza la idea de tomar algunas decisiones con ayuda de “la suerte” o el azar, además, con la llegada de la informática y la programación a mi vida, con mis gustos e inquietudes, surgió la idea y el reto personal de crear mi propia aplicación que tomase decisiones por mí. Así fue como nació “DecideJbot”

## **Ámbito y entorno**

Si con entorno nos referimos a los dispositivos, este proyecto está enfocado para cualquier tipo de entorno digital ya sean dispositivos móviles con sistema operativo Android, o cualquier dispositivo que disponga de un navegador web.

Por otra parte, el proyecto y la aplicación pueden utilizarse en cualquier ámbito y entorno.

La aplicación y el proyecto “DecideJbot” surgió enfocado para ámbito personal, ya que era un proyecto y reto que quería realizar y utilizar personalmente.

Otro ámbito y entorno al que se puede dirigir y utilizar la aplicación y proyecto es el ámbito y entorno familiar, ya que por ejemplo, se podría utilizar para decidir sobre las tareas del hogar.

Otros entornos, podrían ser entornos empresariales, y escolares, ya que la aplicación podría utilizarse, por ejemplo, para que una empresa lance y realice un sorteo para sus clientes, o, en el caso de las escuelas, tomar decisiones para formar grupos, decidir un delegado…

Por último, el proyecto tiene un enfoque relacionado con el ocio, ya que la aplicación dispone de secciones que se pueden utilizar en grupos de amigos, fiestas, juegos… Como puede ser la función de girar una ruleta o lanzar un dado.

## **Análisis de la realidad**

Investigando y probando aplicaciones similares o que puedan competir con “DecideJbot” se pueden hablar de algunas:

* Sortea2: Exclusiva para navegadores, con funciones dedicadas a sorteos y permite almacenar algún dato.
* AppSorteos: Exclusiva para navegadores. Incluye varios apartados que incluye mi proyecto, y otros que se implementarán en futuras actualizaciones, además de incluir una versión de pago. Cómo punto fuerte para DecideJbot contra esta aplicación web, mi aplicación dispone de almacenamiento de datos.
* EchaloASuerte: Exclusiva para navegadores. La mayoría de funciones que ofrece se incluyen en mi aplicación. El punto fuerte de mi aplicación a comparación de esta es que la mía permite el almacenamiento de los participantes para los sorteos. El diseño de mi aplicación comenzó basándose en el diseño de EchaloASuerte.

Respecto a web, puedo destacar las funciones y diseño de esas 3 aplicaciones. Existen varias aplicaciones web que realizan funciones similares o idénticas.

Respecto a aplicaciones móviles para Android he podido investigar y probar varias, encontrando por separado aplicaciones que realizan sorteos, generan números aleatorios o distintos sorteos como podían ser de grupos, pero ninguna aplicación incluye en una sola todas las funciones que se han implementado en DecideJbot, así como no incluyen los mecanismos de guardado de datos de manera gratuita.

La realidad de este proyecto es, que pese a existir varias aplicaciones, ya sean en móvil o en navegador que realizan las funciones de DecideJbot, esta nació como una necesidad y reto personal, implementando todas las funciones que querría ver en una aplicación de “Sorteos y decisiones” en una única. Además, DecideJbot está pensada para que futuras versiones y la aplicación final implementen decisiones sobre temas específicos, ayudando al usuario a tomar decisiones en estos temas que podrían ser decidir una serie, una película, una actividad, un juego, un alimento… Además, de incluir un registro de todas las actividades del usuario, ya sean sorteos variados o decisiones con un tema específico.

Otro punto positivo de DecideJbot es la posibilidad de poder tener una sesión o no, a gusto del usuario. ¿Quieres usar datos de manera global en móvil y en la aplicación web? Entonces, conviene tener y usar una sesión. ¿Quieres darle un uso casual y rápido a la aplicación? Entonces, puedes usarla sin sesión.

Lo último a destacar, analizando la aplicación, son sus inconvenientes.

Un inconveniente puede ser la cantidad de aplicaciones y aplicaciones web que como ya se ha comentado, existen y realizan funciones similares.

Otro inconveniente, en este caso para el proyecto en si, es la realización de 2 aplicaciones por separado (Web y Android), ya que eso significa doble trabajo y se podría ahorrar y solucionar realizando una única aplicación Hibrida, aunque para el caso de este proyecto, se ha decidido trabajar con 2 aplicaciones, independientes, pero con datos comunes.

## **Solución y justificación**

Cómo se comenta anteriormente, la idea de proyecto surge a raíz de ciertas indecisiones en distintas situaciones, además de la aparición del mundo de la programación y mi interés en este mundo.

Debido a las indecisiones que se comentan, empecé a utilizar el azar para poder tomar ciertas decisiones, y, para ser exactos, comencé a utilizar una web para ayudarme a tomar estas decisiones. Siendo exactos, la web utilizada era la de “Sortea2”.

A raíz de usar esa aplicación web e ir adquiriendo más conocimientos y más gustos e inquietudes en el mundo de la programación, me surgió la idea de dejar de utilizar esa web y poder crear mi propia aplicación, con funciones similares, pero con mi propio diseño, funciones y datos.

Así pues, fue donde nació la idea de crear mi aplicación, con ese objetivo y, además, como reto personal.

Para comenzar, DecideJbot comenzó siendo una única aplicación web realizada con PHP y que simplemente realizaba “Sorteos simples” y decidía sobre algún tema específico, que fueron series y música.

Posteriormente, y ya como parte de este trabajo de Fin de Grado, me surgió la idea de trabajar la aplicación Web con Javascript, para solucionar temas de recargas de página y datos que eran un pequeño inconveniente de PHP.

Además, como parte del proyecto y necesidad para este, ya con conocimientos sobre Java y Android, nació y se comenzó a crear la aplicación móvil.

Finalmente, con conocimientos adquiridos en las prácticas y conociendo el Framework Angular, surgió la idea de adaptar la aplicación web a este Framework.

Respecto al almacenamiento de datos, estos se comenzaron almacenando en una base de datos local con MySQL, pero posteriormente, los datos se han terminado almacenando de manera local, en Android con SQLite y en web en el propio navegador con las funciones de LocalStorage. Además, implementando un almacenamiento mixto con una base de datos NoSQL de Firebase, llamada Firestore,

Con todos estos cambios y evoluciones se ha conseguido que DecideJbot sea un proyecto formado por dos aplicaciones, que realizan funciones basadas en la aleatoriedad y el azar, que pueden trabajar con sesiones de datos y trabajan con datos almacenados local y externamente.

## **Destinatarios**

DecideJbot no está destinada a un tipo de usuario específico, ya que es una aplicación que todo el mundo puede utilizar, algunos destinatarios podrían ser:

* Usuarios indecisos que pueden utilizarla para ayudarse a tomar sus decisiones.
* Usuarios que necesitan realizar un sorteo.
* Usuarios que quieren usar un elemento para obtener un valor aleatorio: Sorteo, flecha, dado, moneda…
* Usuarios que quieren realizar grupos.
* Usuarios que quieren asociar elementos.
* Usuarios que quieren disponer de un registro de actividades realizadas.

## **Objetivo del proyecto**

DecideJbot es una aplicación que toma decisiones por los usuarios, además de llevar un registro de esta.

Nos permite realizar desde un sorteo simple, indicando únicamente los participantes y los números de premios a repartir, permitiendo almacenar estos datos local, externamente o de manera conjunta para una posible reutilización, permitiéndonos visualizar los resultados de estos sorteos, además de poder almacenarlos y compartirlos.

Por otra parte, nos permite realizar sorteos más avanzados, incluyendo más opciones, como puede ser la programación de un sorteo en una fecha específica, haciendo así que los resultados de ese sorteo no se generen hasta la fecha específicada.

También, nos permite generar números aleatorios en un rango especificado, indicando un número mínimo y un máximo.

Por último, se permite realizar unas acciones específicas que harán uso del azar, como puede ser el lanzamiento de un dado, el lanzamiento de una moneda o la rotación de una flecha.

Además, la aplicación permitirá al usuario trabajar con sus datos creados por él o por la aplicación, permitiéndole crear, programarlos, borrarlos, combinarlos o visualizarlos con un uso optativo de sesión.

## **Objective of the Project**

DecideJbot is an application that makes decisions for users, in addition to keeping track of it.

It allows us to carry out from a simple draw, indicating only the participants and the numbers of prizes to be distributed, allowing us to store this data locally, externally or jointly for possible reuse, allowing us to visualize the results of these draws, in addition to being able to store and share them.

On the other hand, it allows us to make more advanced draws, including more options, such as scheduling a draw on a specific date, thus making the results of that draw not generated until the specific date.

Also, it allows us to generate random numbers in a specified range, indicating a minimum and a maximum number.

Finally, it is allowed to perform specific actions that will make use of chance, such as the throwing of a die, the throwing of a coin or the rotation of an arrow.

In addition, the application will allow the user to work with their data created by him or by the application, allowing him to create, program, delete, combine or visualize them with an optional use of session.

# **Acuerdo del proyecto**

## **Requisitos funcionales y no funcionales**

|  |
| --- |
| **Requisitos funcionales** |
| El usuario puede decidir si iniciar o no sesión |
| El usuario puede realizar un sorteo |
| El usuario puede guardar los resultados de un sorteo (Externamente) |
| El usuario puede acceder a los resultados generados de un sorteo. |
| El usuario puede compartir resultados. |
| El usuario puede añadir participantes a un sorteo. |
| El usuario puede reemplazar participantes de un sorteo. |
| El usuario puede eliminar participantes de un sorteo. |
| El usuario puede guardar un sorteo (Local, externamente o de manera conjunta) |
| El usuario puede programar la fecha de realización de un sorteo. |
| El usuario puede lanzar un dado. |
| El usuario puede generar números aleatorios. |
| El usuario puede lanzar una moneda. |
| El usuario puede girar una flecha. |
| El usuario puede gestionar sus resultados (Crear, ver, editar y eliminar) |

|  |
| --- |
| **Requisitos no funcionales** |
| La aplicación móvil es exclusiva para Android. |
| La aplicación web se puede utilizar en cualquier dispositivo con un navegador web. |
| Los datos se pueden almacenar localmente:   * Android (SQLite con Sugar ORM) * Web (LocalStorage) (Almacenamiento por navegador) |
| Los datos se pueden almacenar externamente:   * Firestore de Firebase. |
| * Las aplicaciones generan resultados aleatoriamente. |

## **Tareas**

|  |  |  |
| --- | --- | --- |
| **App Móvil** | | |
| Autenticación de usuarios | Registro | |
| Inicio de sesión | |
| Inicio de sesión con Google | |
| Sorteos | Sorteo simple | Añadir un título |
| Añadir participantes |
| Borrar todos los participantes |
| Guardar sorteo localmente |
| Ver sorteos locales |
| Cargar sorteos locales (Añadir/Reemplazar participantes) |
| Eliminar sorteos locales |
| Permitir decidir si realizar el sorteo con una cuenta atrás |
| Guardar sorteo externamente |
| Implementar sonidos |
| Generador de números aleatorios | Diseño |
| Lógica |
| Dado | |
| Flecha | |
| Cara o cruz | |
|  | Resultados | Guardar resultados (Externamente) |

|  |  |
| --- | --- |
| **App Web** | |
| Usuarios | Registro |
| Inicio de sesión |
| Inicio de sesión con Google |

|  |  |  |
| --- | --- | --- |
| Sorteos | Sorteo simple | Añadir un título |
| Añadir participantes |
| Borrar todos los participantes |
| Guardar sorteo localmente |
| Ver sorteos locales |
| Cargar sorteos locales (Añadir/Reemplazar participantes) |
| Eliminar sorteos locales |
| Permitir decidir si realizar el sorteo con una cuenta atrás |
| Guardar sorteo externamente |
| Cargar sorteos externos (Añadir/Reemplazar participantes) |
| Implementar sonidos |
| Generador de números aleatorios | Diseño |
| Lógica |
| Sorteo avanzado | Diseño de los pasos |
| Lógica |
| Programar fecha |
| Resultados | Guardar resultados (Externamente) |
| Ver resultados |
| Actualizar resultados |
| Eliminar resultados |
| Adelantar fecha de resultados a la actual |

## **Metodología a seguir para la realización del proyecto**

Para realizar el proyecto he trabajado con la metodología Kanban, realizando un tablero con las tareas que hay que realizar en todo el proyecto, las tareas en proceso y las tareas terminadas.

Utilicé la aplicación Trello para poder realizar el Kanban, creándome una plantilla con las 3 columnas (Pendientes, En proceso y Finalizadas), y una vez creadas, añadí todas las tareas y subtareas a la columna de “Pendientes”.

Analicé las tareas, para decidir cuales eran prioritarias y las moví a la columna “En proceso” para ir realizando las tareas o subtareas planteadas.

Alguna vez, se trabajaban varias tareas a la vez, ya que, al realizar dos aplicaciones, realizaba las tareas para ambas aplicaciones simultáneamente.

Una vez se termina una tarea, se índica la fecha de finalización y se mueve a la columna “Finalizada”, aunque podría darse la situación de tener que realizar una nueva subtarea sobre esa tarea y que se volviese a mover a la columna “En proceso”.

## **Planificación temporal de tareas**

Para planificar el tiempo que se le quiere dedicar a cada tarea, lo primero que hay que tener en cuenta es la fecha final de entrega de la versión del proyecto.

Siendo la fecha de entrega el día 17 de Junio de 2022, y comenzando el proyecto aproximadamente el 4 de Abril de 2022, lo primero que hay que tener en cuenta es el orden en el que se quieren realizar las tareas, ya que unas pueden depender de otras, y para esto, se utilizó la metodología.

La primera tarea, era la planificación del proyecto, teniendo una idea clara de cómo iba a ser, que funcionalidades iba a tener y cumplir…

Una vez planificado el proyecto, se pudo planificar todas las tareas, y se podía aproximar un tiempo a cada tarea. Cabe destacar, que los tiempos no iban a ser exactos, y que podrían llevar más o menos tiempo dependiendo de las complicaciones u otros aspectos que puedan afectar a la tarea.

Otra de las primeras tareas, una vez planificado el proyecto y teniéndolo en mente, era realizar un primer diseño de las aplicaciones, realizando mockups de las pantallas que tendrán las aplicaciones.

Una vez planeado el diseño de las aplicaciones, el diseño se puede comenzar a desarrollar y programar.

El planteamiento que he realizado para el el desarrollo de las tareas del proyecto ha sido realizar el diseño de una pantalla y posteriormente, planear, diseñar y programar su lógica, haciéndolo después de nuevo en la otra aplicación.

En algunas ocasiones, “pausaba” el desarrollo de una tarea y desarrollaba otra simultáneamente.

## **Presupuesto**

|  |  |  |
| --- | --- | --- |
| **DecideJbot** | | ***15/06/2022*** |
|  | | |
| **Descripción** | **Unidades** | **Ingresos** |
| Horas trabajadas | 360 | + 2880 € |
| Testers |  | -500 € |
| Diseñadores |  |  |
|  |  |  |
| Publicación de la app móvil en Google Play |  | -25 € |
| Publicación de la app Web |  | -55 € / año |
|  |  |  |
| Total |  |  |

# **Análisis y diseño**

## **Modelado de datos**

## **Análisis y diseño de la interfaz de usuario**

La primera parte del análisis y diseño de la interfaz de usuario está formada por unos mockups de las pantallas y las páginas de las aplicaciones. Los mockups son una primera idea de como van a ser estas pantallas y páginas, de que están formadas, y se acompaña de una pequeña descripción de como actuarían sus componentes, por si es necesario conocerlo para el diseño.

**Aplicación móvil:**

Al iniciar la aplicación, aparece una “Splash Screen” (Pantalla de carga) con el logo de la aplicación, que cambiará de pantalla automáticamente a los 5 segundos.

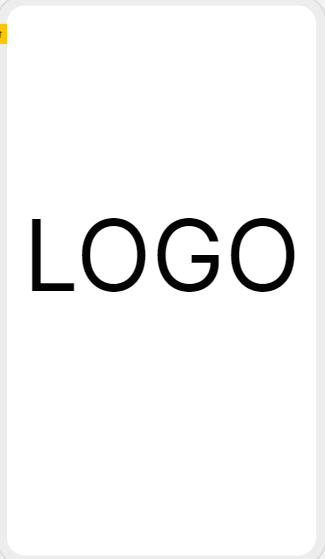
****

Ilustración 1: Splash Screen

Tras la pantalla de carga, en caso de que el usuario no tenga una sesión iniciada, aparecerá la pantalla de inicio de sesión:

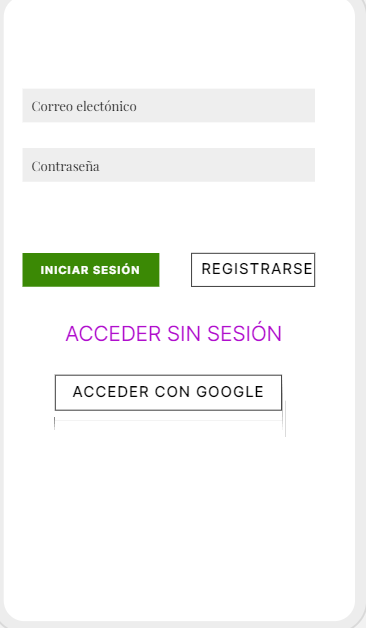


Ilustración : Pantalla de inicio de sesión

Si en la pantalla de inicio de sesión, el usuario pulsa el botón de “Registrarse”, accederá a la pantalla de registro

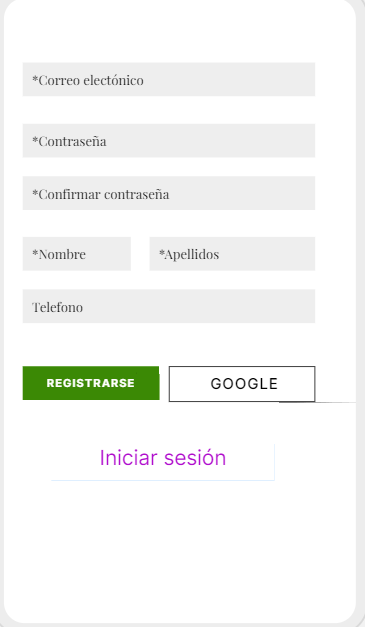


Ilustración : Pantalla de registro de usuario

Tras la pantalla de carga, en caso de que el usuario tenga iniciada sesión, o tras iniciar sesión, aparecerá la “Main Activity” o pantalla principal, con el logo de la aplicación y un menú a las secciones de la aplicación.

La sección principal o “Home” dispone de un botón en formato de carta para iniciar o cerrar sesión y otro botón para acceder a la web.



Ilustración 4: Main Activity

En caso de pulsar el botón “Sorteos” del menú de la pantalla principal, aparecerá el fragmento de la sección de sorteos.

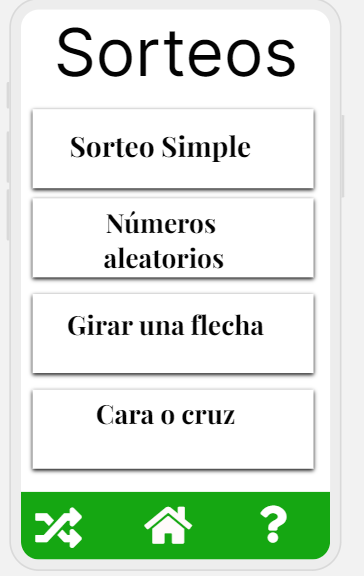


Ilustración 5: Fragmento Sorteos

En la sección de sorteos, al pulsar el botón de “Sorteo simple” nos llevará a la pantalla de Sorteo Simple.

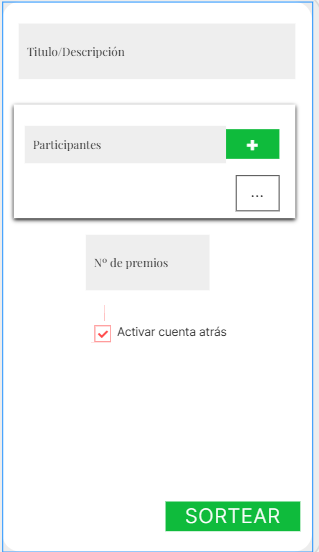


Ilustración 6: Pantalla Sorteo Simple

En el sorteo simple, si falta algún dato de completar, aparecerá una seekbar con un mensaje de error.

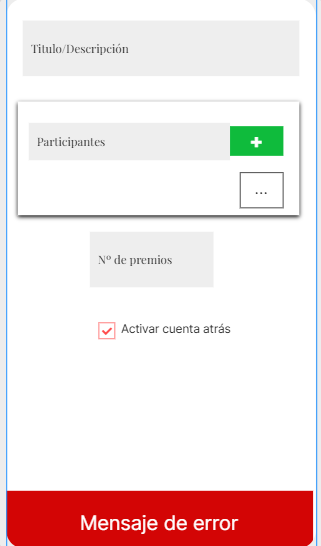


Ilustración 7: Sorteo Simple con error

En el sorteo simple, al pulsar en los tres puntitos, aparecerá un menú para poder gestionar el sorteo (Guardar, cargar o borrar)



Ilustración 8: Menú en Sorteo Simple

En el sorteo simple, si están todos los campos rellenos correctamente, al dar en el botón “Sortear” se generarán y mostrarán los resultados listados en un dialogo.



Ilustración 9: Lista de ganadores en Sorteo Simple

En el sorteo simple, al seleccionar los tres puntitos y seleccionar la opción “Guardar participantes” aparecerá un menú con el que el usuario podrá decidir si guardar los datos del sorteo localmente, externamente o de ambas maneras.

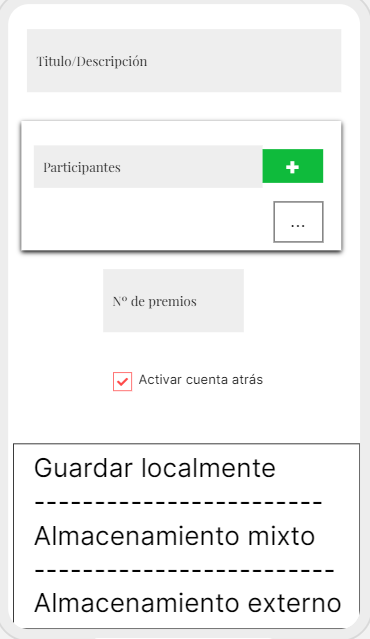


Ilustración : Menú guardar en Sorteo Simple

En el sorteo simple, al seleccionar los tres puntitos y seleccionar la opción “Cargar listas de participantes” aparecerá un cuadro de diálogo con la lista de sorteos, permitiendo al usuario cargar las listas de participantes de estos añadiéndolas a la lista actual o reemplazando la lista actual.

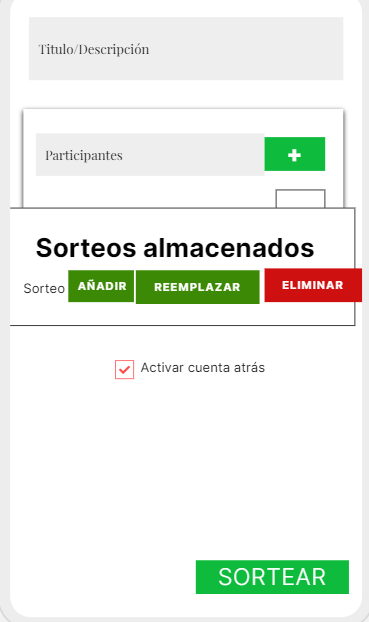


Ilustración : Lista de sorteos cargadas en Sorteo Simple

En la sección de sorteos, al pulsar en la tarjeta de “Números aleatorios” llevará al usuario a la pantalla del generador de números aleatorios

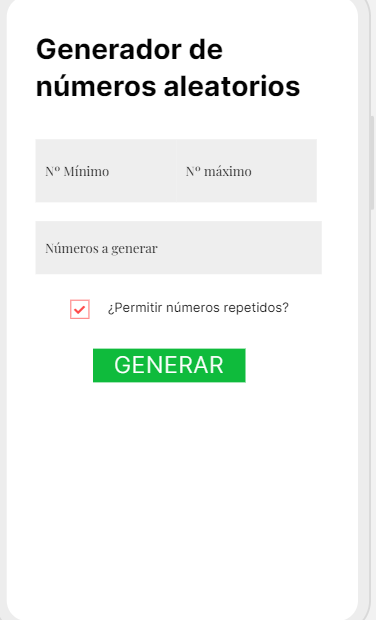


Ilustración : Generador de números aleatorios

En el generador de números aleatorios, si todos los campos están completados y correctamente, al pulsar el botón “Generar”, aparecerán los resultados en la pantalla.

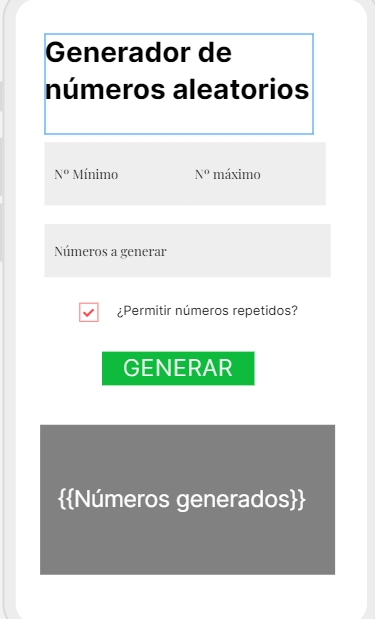


Ilustración : Generador de números aleatorios con resultados

En el generador de números aleatorios, en caso de no haber completado todos los campos o que alguno de ellos tenga un valor incorrecto, aparecerá un mensaje de error.

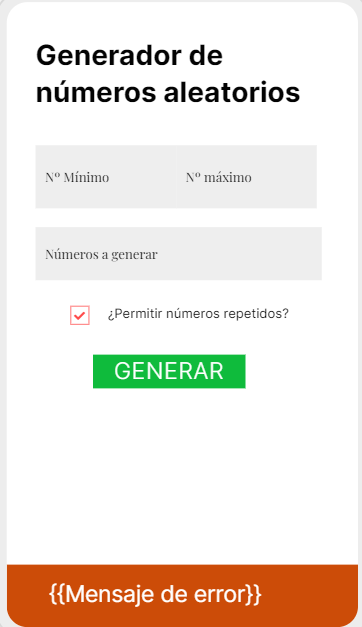


Ilustración : Generador de números aleatorios con error

En la sección de Sorteos, al pulsar en la tarjeta de girar una flecha, nos llevará a su respectiva pantalla.

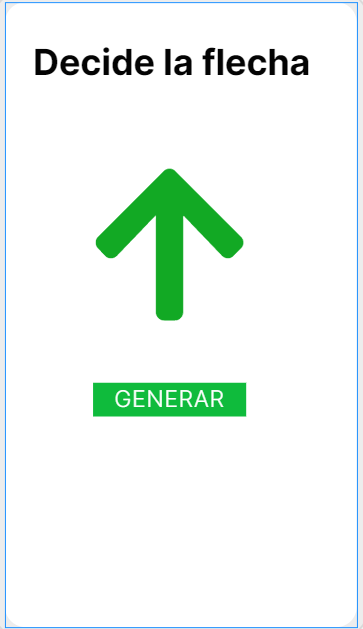


Ilustración : Pantalla "Girar una flecha"

En la sección de Sorteos, al pulsar en la tarjeta de “Lanzar un dado”, nos llevará a su respectiva pantalla.

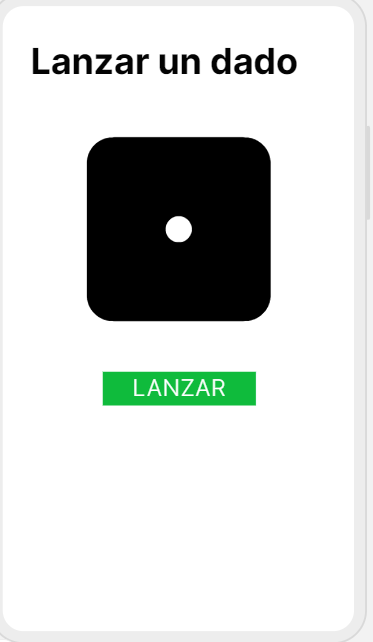


Ilustración : Pantalla "Lanzar un dado"

En la sección de Sorteos, al pulsar en la tarjeta de “Cara o cruz”, llevará al usuario a la respectiva pantalla,

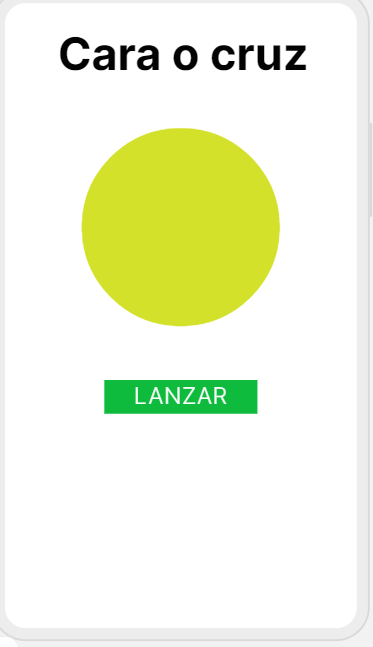


Ilustración : Pantalla "Cara o Cruz"

**Aplicación web:**

El header de la aplicación contiene el logo y nombre de la aplicación, un menú con submenú para el apartado de sorteos y sus subapartados.

En caso de no tener iniciada sesión, en el header aparecerán unos botones para iniciar sesión o registrarse.



Ilustración : Header (Sin sesión iniciada)

En caso de haber una sesión iniciada, en el header aparecerá información de usuario y un botón para cerrar la sesión.



Ilustración : Header (Con sesión iniciada)

Al acceder a la aplicación web, la página principal mostrará las secciones disponibles en formato de tarjeta.



Ilustración : Página principal de la app web

En caso de ver la página principal desde un navegador en un dispositivo móvil, la página se adaptará ya que es responsiva.

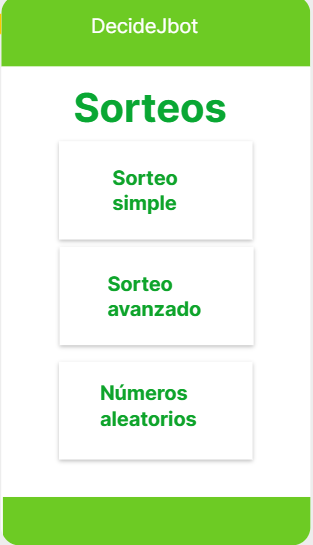


Ilustración : Página principal de la app web vista en móvil

En la página principal, en caso de tener sesión iniciada y que el usuario haya realizado sorteos y disponga de resultados guardados externamente, aparecerá la tarjeta “Historial de sorteos”

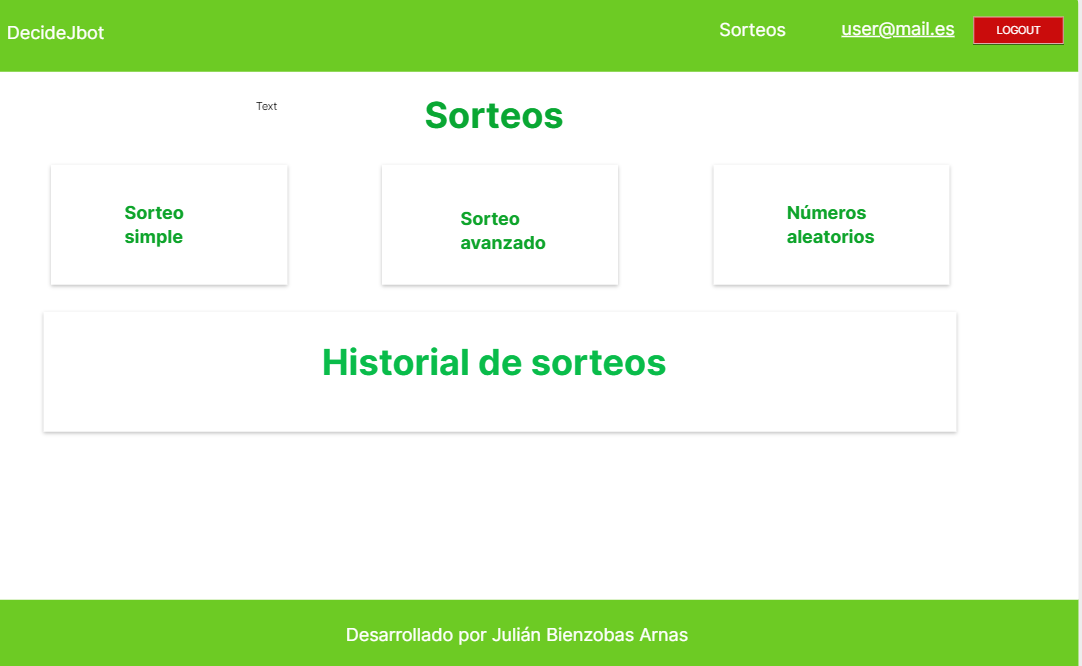


Ilustración : Pantalla principal con historial de sorteos

Si el usuario accede a la página de sorteo simple, verá el siguiente contenido:



Ilustración : Página "Sorteo Simple"

En la página del sorteo simple, en caso de que el usuario no haya completado los campos o alguno sea inválido, aparecerá una ventana modal con un mensaje de error.



Ilustración : Página "Sorteo Simple" con error

En la página de Sorteo Simple, si los campos se han completado correctamente, al pulsar el botón “Sortear” se generarán los resultados.



Ilustración : Resultados del Sorteo Simple

En la página de Sorteo Simple, si se activa el contador, al pulsar el botón “Sortear”, aparecerá un contador de 5 segundos que irá disminuyendo, acompañado de un texto de carga.



Ilustración : Resultados del Sorteo Simple

Si el usuario accede mediante el menú del header o mediante la página inicial al sorteo avanzado, lo primero que verá será lo siguiente:

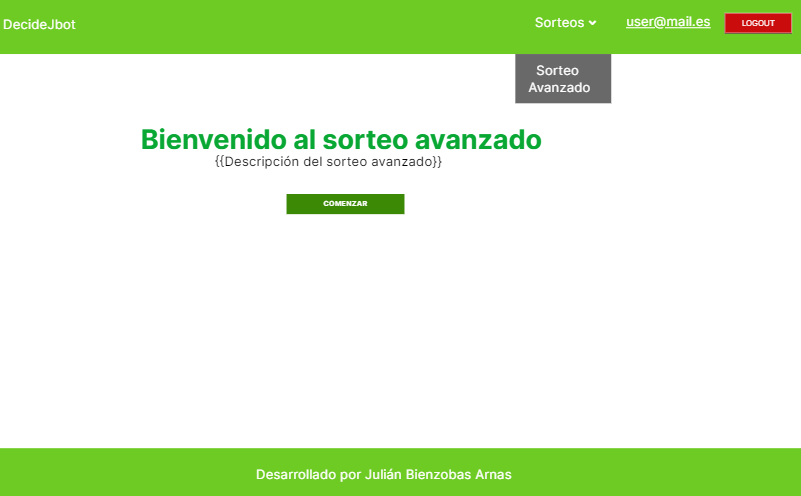


Ilustración : Página de bienvenida al Sorteo Avanzado

Cuando el usuario pulsa el botón “Comenzar” en la bienvenida del Sorteo Avanzado, aparece lo siguiente:



Ilustración : Paso 1 del Sorteo Avanzado

Cuando el usuario pulsa el botón “Participantes” en el paso 1 del Sorteo Avanzado, aparece el paso 2, correspondiente a los participantes:



Ilustración : Paso 2 del Sorteo Avanzado

Si el usuario completa correctamente el paso 2 del Sorteo Avanzado y pulsa el botón “Premios”, aparece el paso 3, correspondiente a los premios:



Ilustración : Paso 3 del Sorteo Avanzado



Ilustración : Paso 4 del Sorteo Avanzado

Cuando el usuario pulse el botón “Finalizar” del paso 4 del sorteo avanzado, aparecerán todos los datos del sorteo con botones para volver a pasos anteriores o confirmar y realizar el sorteo.



Ilustración : Detalles del Sorteo Avanzado

Cuando el usuario pulse el botón de “Sortear” en el Sorteo Avanzado aparecerá un enlace con un botón para copiar este enlace.

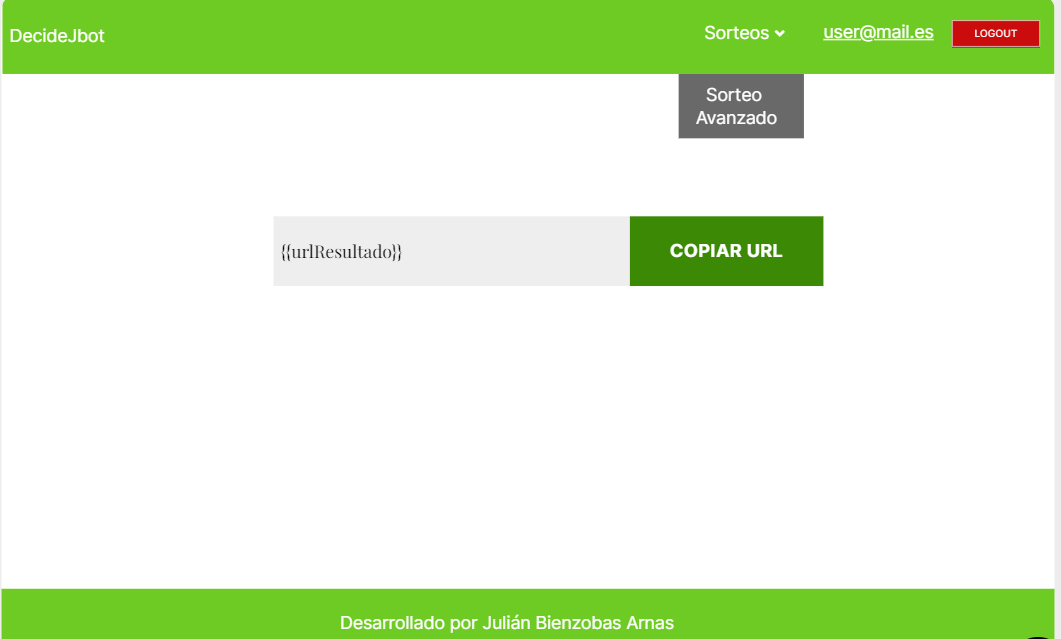


Ilustración : Enlace al resultado del Sorteo Avanzado

Cualquier usuario que acceda al enlace generado del resultado en caso de que la fecha programada no se haya alcanzado y los resultados no se hayan generado, verá lo siguiente:



Ilustración : Detalles del Sorteo Avanzado

En caso de que el usuario que acceda al enlace generado del resultado sea el usuario que creó el sorteo y no se haya alcanzado la fecha programada ni se hayan generado los resultados, se mostrará lo siguiente:

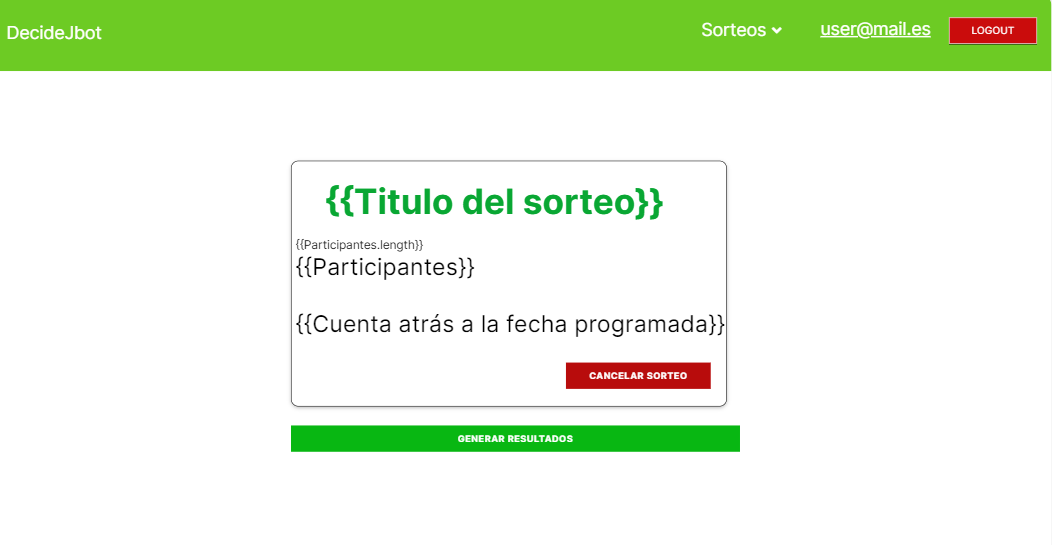


Ilustración : Sorteo programado del usuario logueado

En caso de acceder al resultado y que los resultados ya estén generados, en caso de no ser un resultado creado por el usuario logueado, se mostrará de la siguiente manera:



Ilustración : Página de un resultado

En caso de acceder a un resultado ya generado, y que pertenezca al usuario logueado, se mostrará de la siguiente manera:



Ilustración : Página de un resultado del usuario logueado

Si el usuario accede al “Historial De Sorteos” desde la página principal, si tiene algún resultado generado, podrá observar lo siguiente:



Ilustración : Página "Historial De Sorteos"

Si el usuario accede al “Generador de Números Aleatorios” verá la siguiente página:

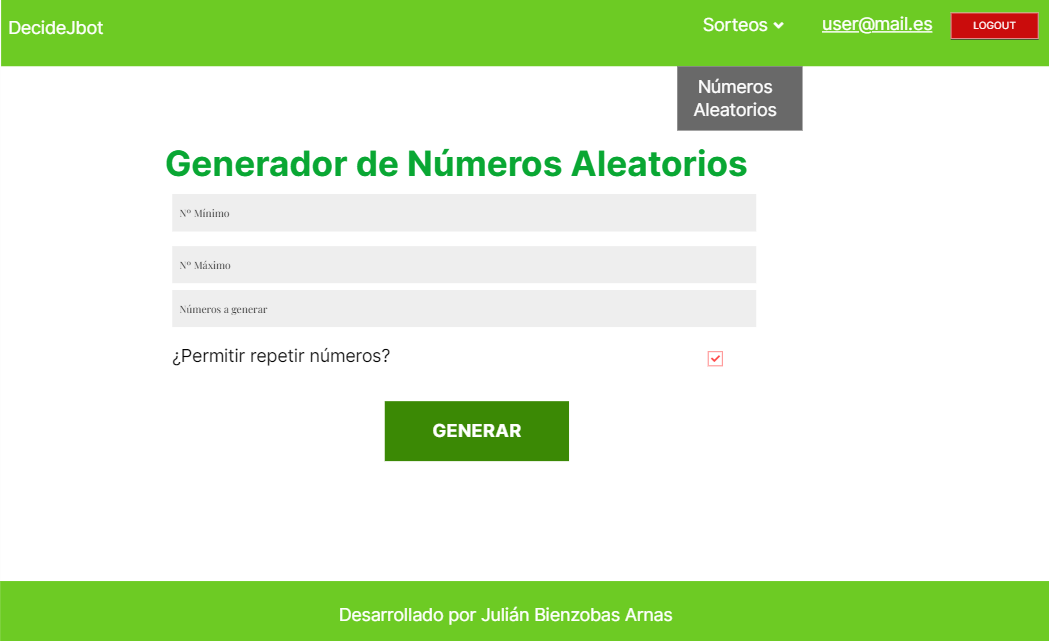


Ilustración : Página "Generador de Números Aleatorios"

Si el usuario completa correctamente todos los campos en el Generador de Números Aleatorios y pulsa el botón “Generar”, se mostrarán los números generados:

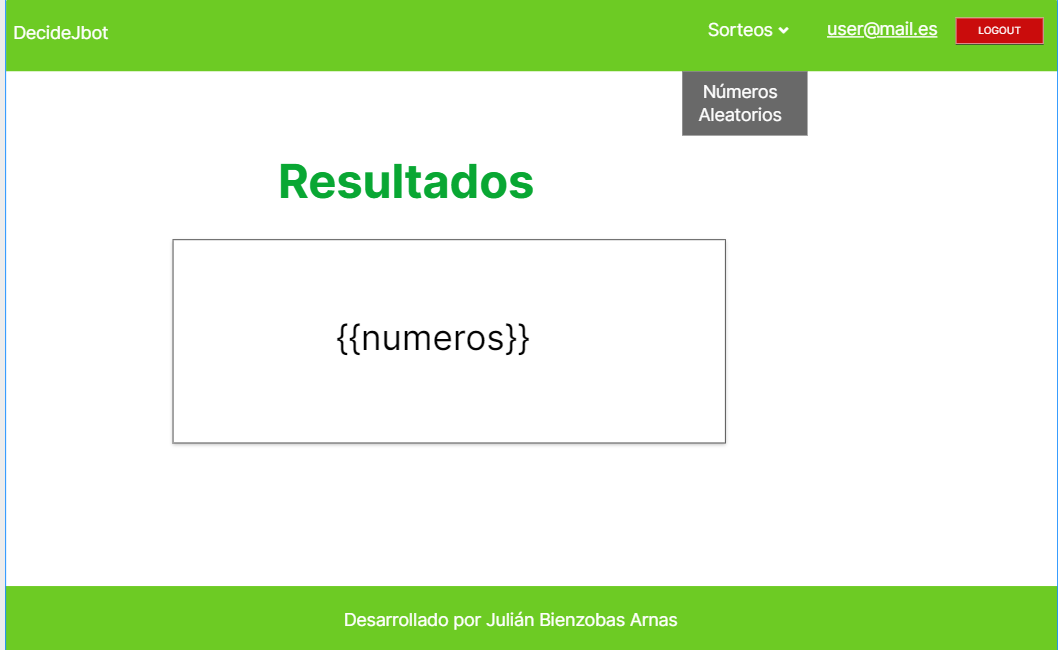


Ilustración : Resultados en la página "Generador de Números Aleatorios"

Si el usuario pulsa el botón de iniciar sesión, se le mostrará la página de inicio de sesión:

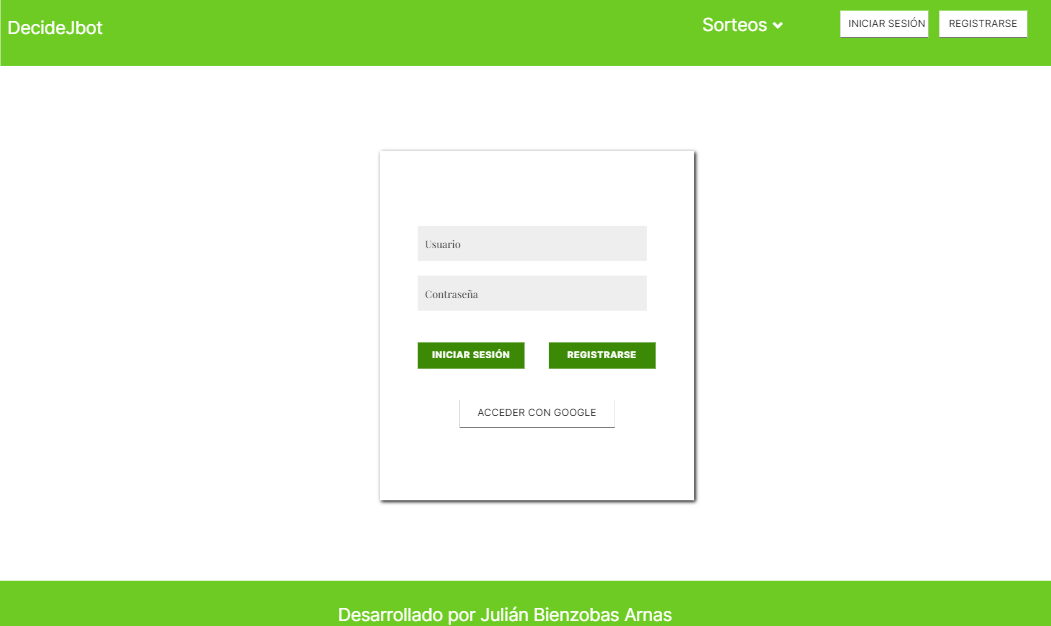


Ilustración : Página de inicio de sesión

Si el usuario pulsa el botón de registrarse, se le mostrará la página de registro:

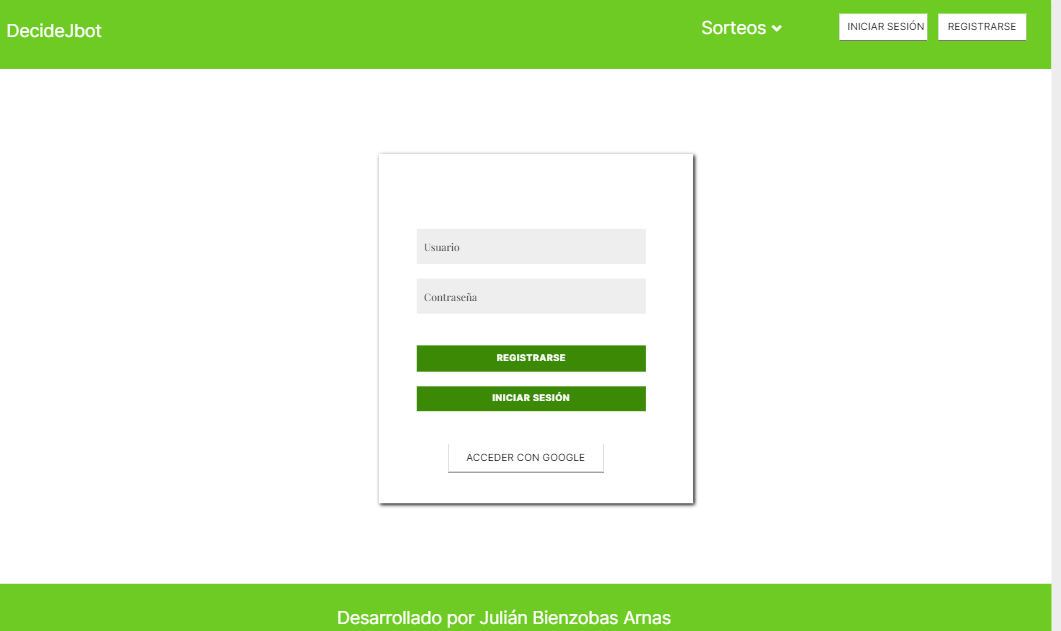


Ilustración : Página de registro de usuario

Una vez pensada la idea de cómo se quiere que sea el diseño de la interfaz de usuario de las aplicaciones, y teniendo un primer diseño en mockups, se procede al diseño real de las interfaces de usuario de las aplicaciones, quedando estas de la siguiente manera.

**Aplicación móvil:**

**Aplicación web:**

****

Ilustración : Página de inicio (Sin sesión iniciada)



Ilustración : Página de inicio (Con sesión iniciada)



Ilustración : Página "Sorteo Simple"



Ilustración : Página "Sorteo Simple" (Con datos)



Ilustración : Opciones para el Sorteo Simple



Ilustración : Opciones de importación en el Sorteo Simple



Ilustración : Opciones de exportación en el sorteo simple

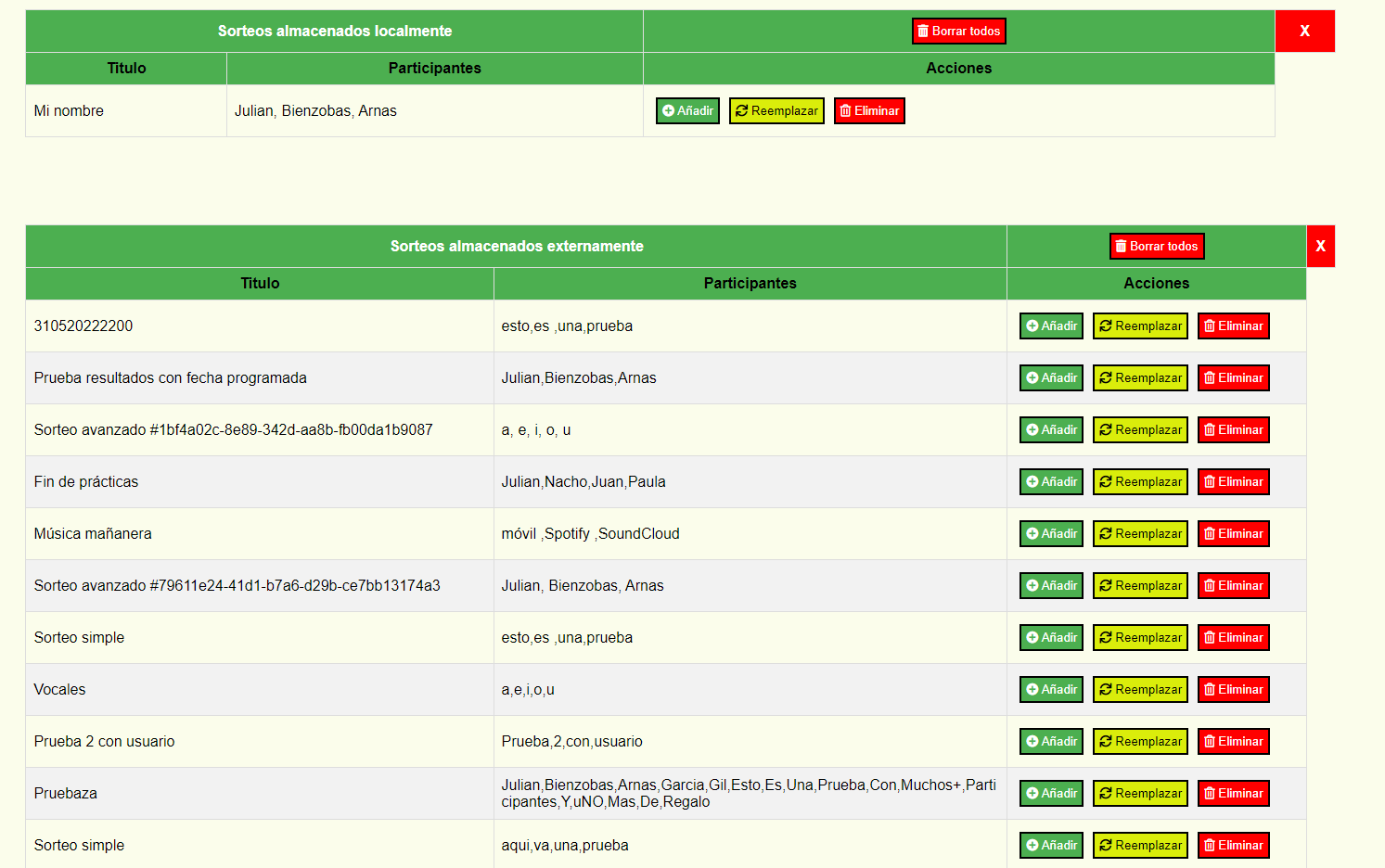


Ilustración : Importación de participantes al Sorteo Simple



Ilustración : Resultados en Sorteo Simple



Ilustración : Inicio al Sorteo Avanzado



Ilustración : Paso 1 del Sorteo Avanzado



Ilustración : Paso 2 del Sorteo Avanzado



Ilustración : Paso 3 del Sorteo Avanzado



Ilustración : Paso 4 del Sorteo Avanzado



Ilustración : Detalles del Sorteo Avanzado



Ilustración : URL a un resultado



Ilustración : Resultado programado del usuario logueado

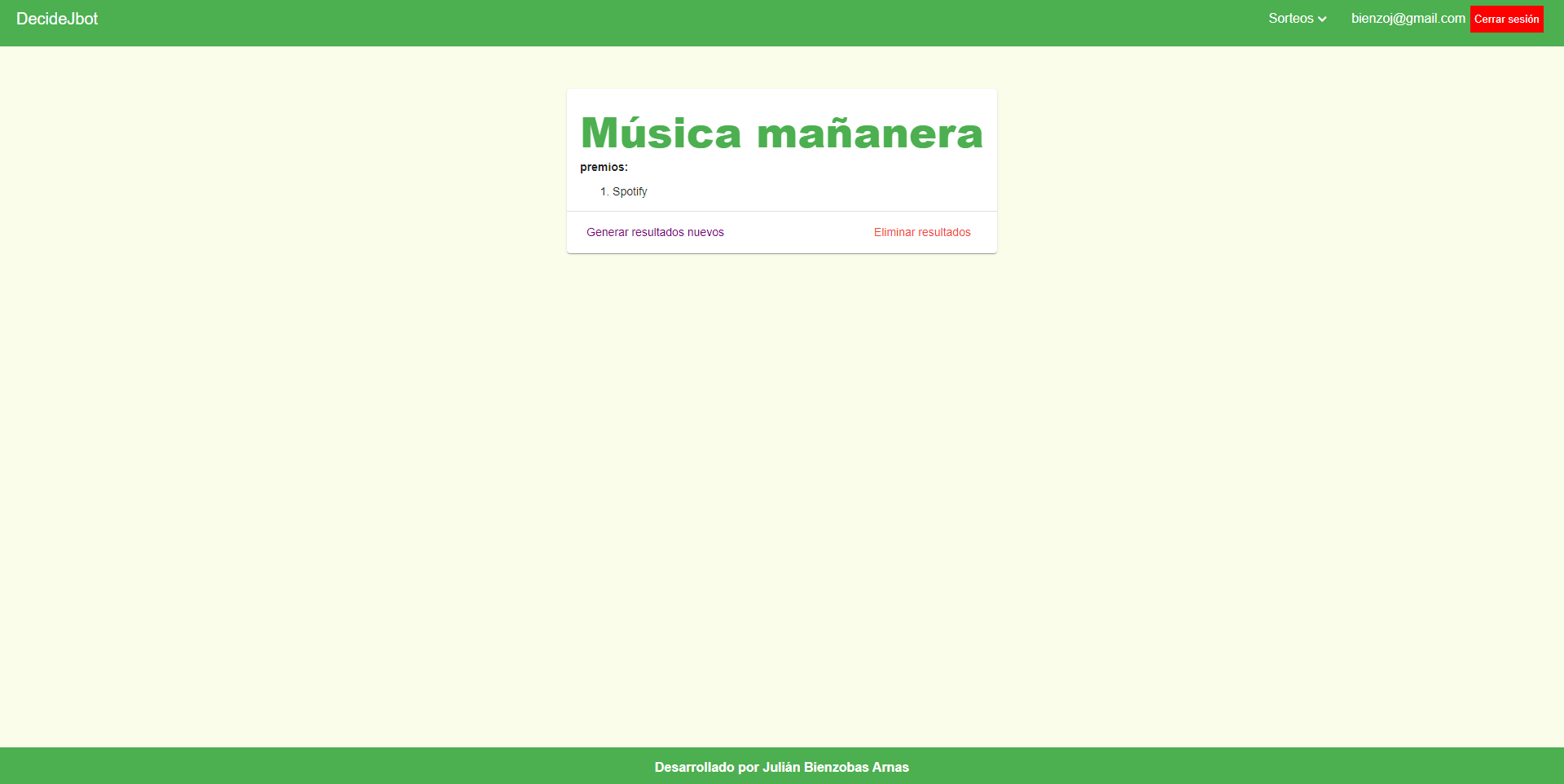


Ilustración : Resultado generado (Del usuario logueado)



Ilustración : Historial de Sorteos



Ilustración : Resultado de otro usuario



Ilustración : Generador de Números Aleatorios

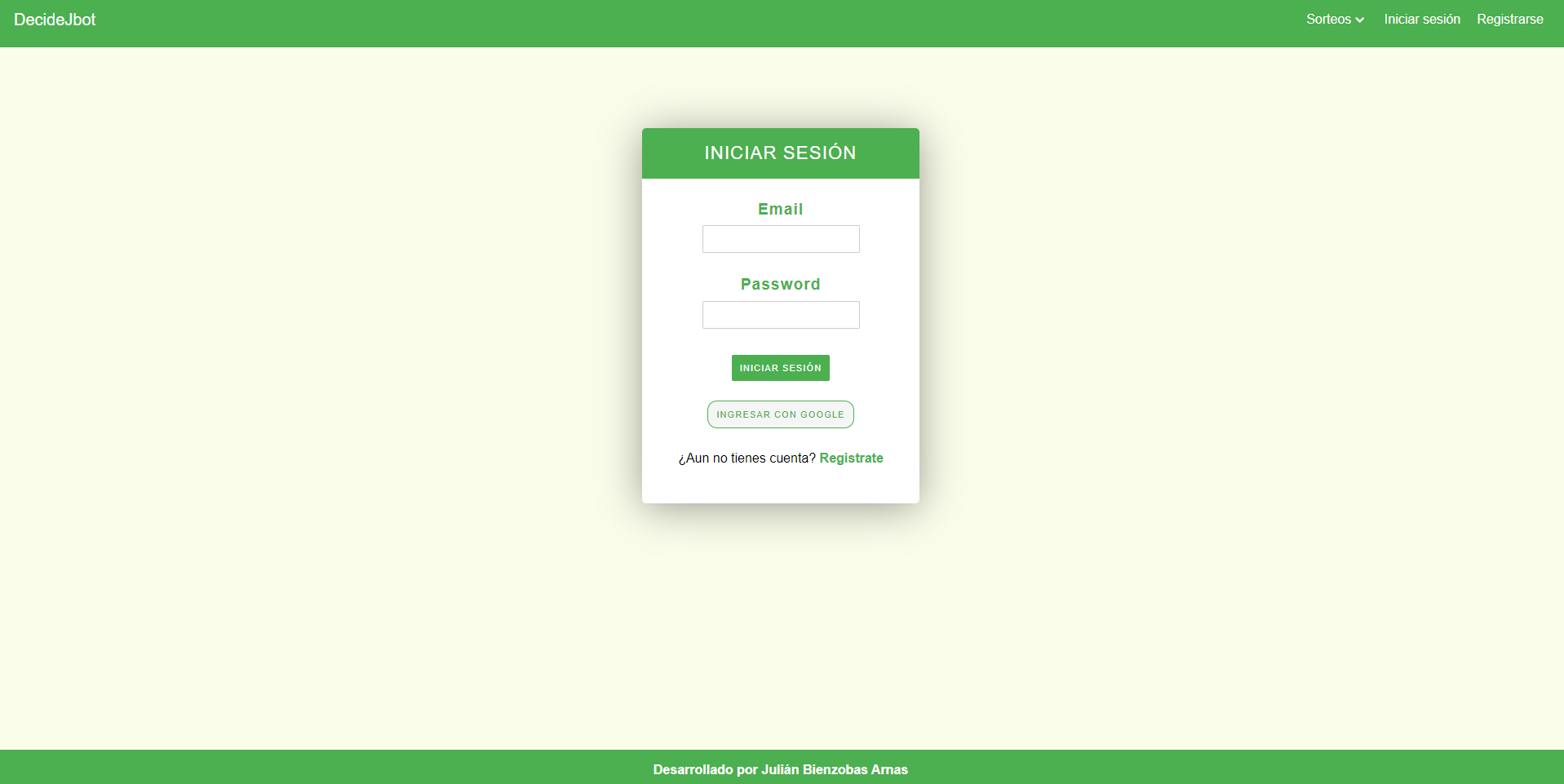


Ilustración : Página de Inicio de Sesión

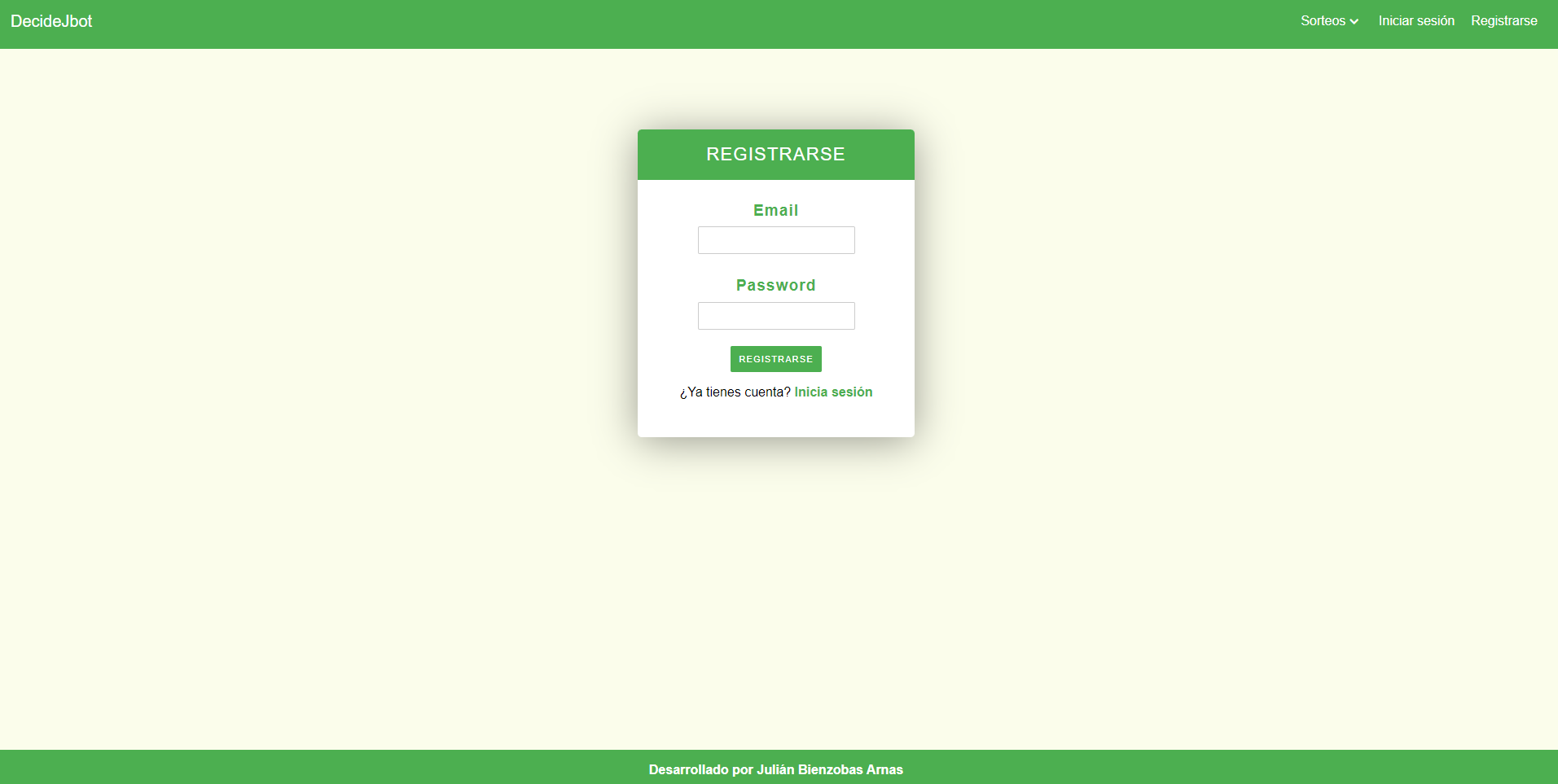


Ilustración : Página de Registro

## **Análisis y diseño del sistema funcional**

En este apartado, se describe el funcionamiento que deberá realizar cada componente de las aplicaciones, para que el Developer lo programe y haga la aplicación funcional.

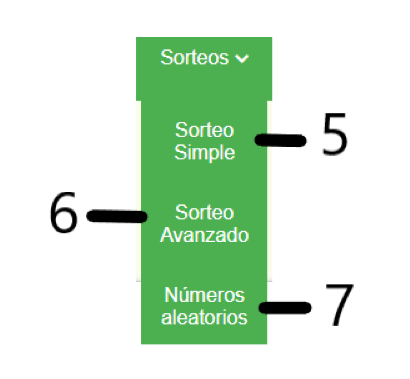
**App móvil:**

**App web:**

* Header:



1. Al pulsar el logo, se redirigirá al usuario a la pantalla principal.
2. Despliega el submenú de sorteos, mostrando los subapartados de este.
3. (Aparece en caso de no tener la sesión iniciada). Redirige al usuario al Login.
4. (Aparece en caso de no tener la sesión iniciada). Redirige al usuario a la página de registro.



1. Redirige al usuario a la página de “Sorteo Simple”
2. Redirige al usuario a la página de “Sorteo Avanzado”
3. Redirige al usuario a la página de “Números Aleatorios”



1. (Solo se muestra si hay sesión iniciada). Cierra la sesión de usuario.