|  |
| --- |
| Fornite |
| For epic gamers |
| Diego Cuenca Arriols |

Contenido

[Naturaleza del proyecto 3](#_Toc10441812)

[Descripción del proyecto 3](#_Toc10441813)

[Justificación 3](#_Toc10441814)

[Beneficiarios 3](#_Toc10441815)

[Definición de los Objetivos 4](#_Toc10441816)

[Tareas 4](#_Toc10441817)

[Diagrama Temporal 5](#_Toc10441818)

[Desarrollo de la Aplicación 6](#_Toc10441819)

[Tipos de Usuario 6](#_Toc10441820)

[Tecnologías empleadas 6](#_Toc10441821)

[Django 6](#_Toc10441822)

[Python 6](#_Toc10441823)

[MySQL 6](#_Toc10441824)

[Javascript 7](#_Toc10441825)

[Bootstrap 7](#_Toc10441826)

[ChartJS 7](#_Toc10441827)

[Análisis 8](#_Toc10441828)

[Diagrama E/R 8](#_Toc10441829)

[Casos de Uso 8](#_Toc10441830)

[Estructura de datos 8](#_Toc10441831)

[Diagrama de colaboración 8](#_Toc10441832)

[Diagrama navegación 8](#_Toc10441833)

[Diseño 8](#_Toc10441834)

[Estructura de archivos 8](#_Toc10441835)

[Implementación 8](#_Toc10441836)

[Back-End 8](#_Toc10441837)

[Front-End 8](#_Toc10441838)

[Resultados Obtenidos 9](#_Toc10441839)

[Conclusiones y Propuestas 10](#_Toc10441840)

[Anexo 11](#_Toc10441841)

[Estadísticas: Una introducción 11](#_Toc10441842)

[Bibliografía 12](#_Toc10441843)

# Naturaleza del proyecto

## Descripción del proyecto

Nuestra aplicación consiste en un gestor de estadísticas deportivas para equipos de baloncesto.

Extrae estadísticas básicas de otras aplicaciones, almacena esos datos y calcula estadísticas avanzadas de la temporada o partidos concretos. Presenta luego toda esta información de manera simple y sencilla de entender.

De esta manera, luego el equipo técnico (entrenadores, delegados...) de cada equipo puede emplear esta información para tomar decisiones estratégicas, pudiendo exportar estos datos a formatos como PDF.

## Justificación

Los orígenes de esta aplicación surgen de observar la manera de trabajar de mi padre, el cual como afición trabaja como delegado para un equipo de baloncesto llevando estadísticas y otras gestiones. Toma estadísticas durante los partidos y luego mediante hojas Excel y revisando el partido en video, elabora informes con estadísticas avanzadas para el entrenador. También sube esta información a un blog Wordpress donde los jugadores y sus compañeros del equipo técnico pueden consultar el rendimiento del equipo.

Esta tarea es costosa y consume mucho tiempo por lo que la aplicación busca simplificar este proceso. Con tan solo mandar a la aplicación las estadísticas básicas que se toman en el partido, esta genera las estadísticas avanzadas, crea informes y presenta la información en un sitio web donde los jugadores y demás miembros del equipo pueden consultarlo.

## Beneficiarios

Aparte de los beneficiarios de donde surgió la idea de la aplicación (mi padre y el equipo Nacional Femenino Corazonistas del que forma parte) La aplicación puede ser empleada por otros equipos de la categoría, una categoría de alto rendimiento pero con presupuestos modestos que no pueden permitirse pagar a una gran empresa para que gestione sus estadísticas (como hacen grandes equipos de la ACB).

Nuestra aplicación es por tanto una solución económica y simple para equipos más modestos y con una necesidad de conocer el rendimiento estadístico del equipo a un nivel medio-alto.

# Definición de los Objetivos

Contamos con un único objetivo general:

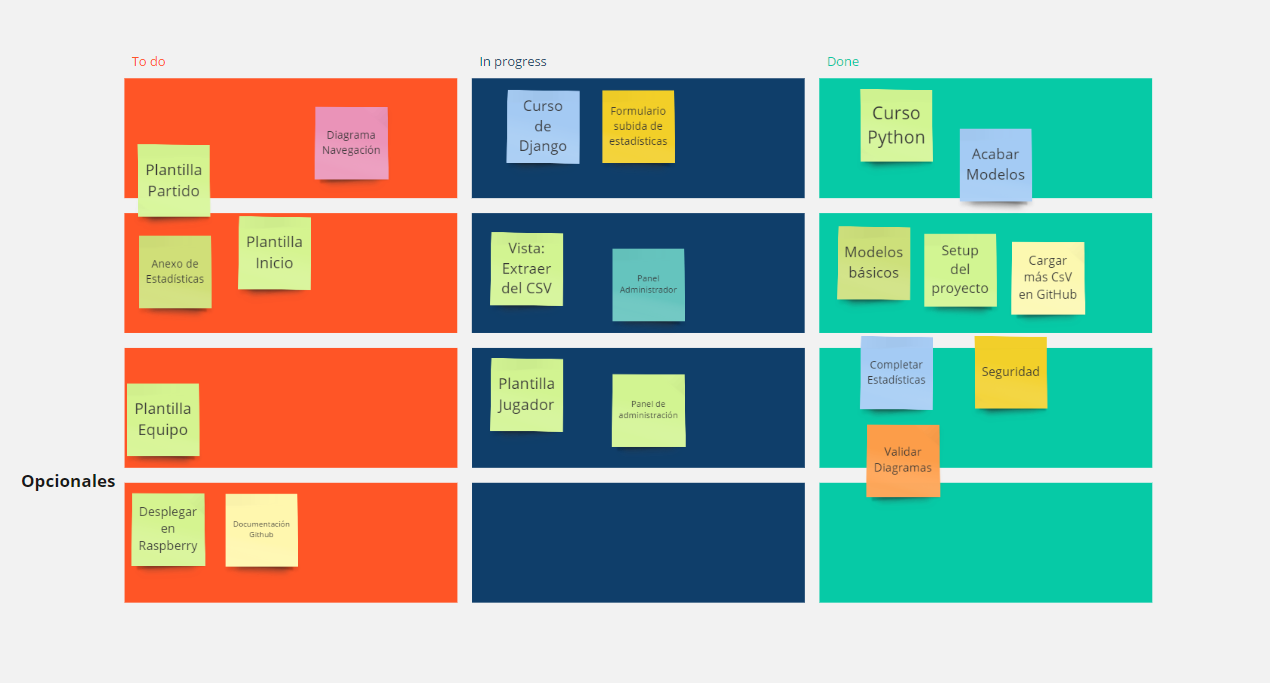
* Calcular estadísticas avanzadas y mostrarlas de manera atractiva: El cual podemos dividir en varios objetivos específicos
  + Extraer la información de nuestras fuentes y guardarla en bases de datos.
  + Calcular estadísticas avanzadas
  + Construir una interfaz para mostrar la información
  + Añadir un criterio de búsqueda para navegar
  + Construir gráficas que faciliten el entendimiento de la información.
  + Exportar la información a otros formatos.

# Tareas

Podemos dividir las tareas a realizar en los siguientes grandes grupos:

* Formación: Adquirir familiaridad con las tecnologías implementadas en el proyecto mediante cursos, documentación…
* Documentación: Elaboración del presente documento.
* Backend: Construir la parte Back de nuestra aplicación, concretamente nuestros Modelos y Controladores (aunque más adelante veremos que Django emplea una nomenclatura distinta para esta estructura).
* Extraer la información de los .CSV que recibimos.
* Frontend: Elaborar las Vistas con las que interactuarán nuestros usuarios, esto incluye la construcción de gráficas a la hora de presentar la información.

Como anotación, se ha empleado la aplicación “Miro” para organizar las tareas. Se trata de una aplicación que simula un tablero Kanban, pudiendo indicar tiempos y otras opciones más avanzadas que dada la envergadura del proyecto no se han usado.



Estado del tablero en un momento indeterminado del proyecto

# Diagrama Temporal

|  |  |  |
| --- | --- | --- |
| Tarea | T. Estimado | T. Real |
| Formación y documentación | 15h | 18h |
| Codificación: Backend | 20h | 22h |
| Codificación: Frontend | 15h | 20h |
| Pruebas | 8h | 9h |
| Despliegue | 2h | 3h |
| TOTAL | 60h | 72h |

# Desarrollo de la Aplicación

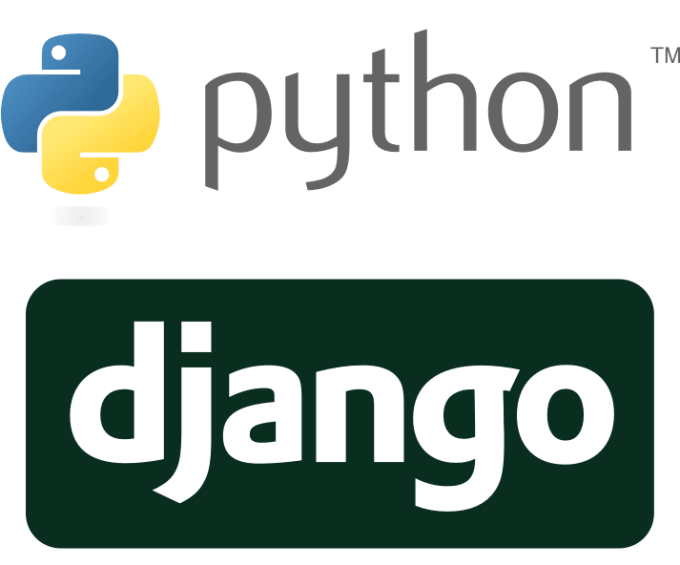
## Tipos de Usuario

Contamos con dos tipos de usuario:

* Jugadores y equipo-técnico: Equivalen al usuario estándar de nuestra aplicación, podrán usar todas las funcionalidades de la aplicación: consulta y exportación de datos.
* Delegado: Equiparable al administrador de la aplicación, aparte de contar con las mismas funcionalidades de un jugador podrá acceder al panel de administrador de la aplicación. Lugar donde se puede editar el contenido de la base de datos manualmente, aparte podrá agregar información de partidos subiendo un fichero .csv

## Tecnologías empleadas

Se han empleado las siguientes tecnologías:



### Django

Se trata de un potente framework de código abierto para Python. Su principal meta es facilitar la creación de sitios web complejos poniendo un énfasis en la reutilización de componentes.

Django sigue el patrón de diseño MVT (Modelo, vista, template). Donde los modelos son una abstracción de los datos de la aplicación web. La vista encapsula toda la lógica responsable de controlar las peticiones de los usuarios y las templates renderizan la información que se presenta al usuario. Es un patrón similar al MVC intercambiando las vistas por templates y el controlador por la vista.

### Python

Python es un lenguaje de alto nivel y multiparadigma. Creado en los años 90 ha crecido exponencialmente su popularidad al ser bastante potente y fácil de aprender e interpretar.

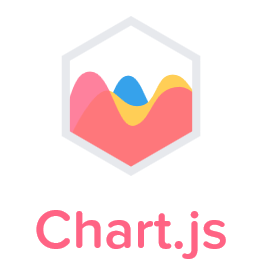
### MySQL

Es un sistema de base de datos relacional propiedad de Oracle. Django no cuenta con soporte nativo para esta base de datos por lo que hemos empleado un plugin llamado “mysqlclient” de Pypi.

### https://syndicode.com/wp-content/uploads/2018/01/Bootrap4-arrived-e1516612818493.pngJavascript

Para añadir funcionalidad a las templates (escritas en HTML5) se ha empleado Javascript, lenguaje de programación interpretado y enfocado al Frontend.

### Bootstrap

Es un framework orientado al Frontend. Facilita el diseño de la interfaz al incorporar multiples plantillas, menús de navegación y estilos.

### ChartJS

Se trata de una libreria de Javascript que permite la construcción de gráficas dinámicas y responsive junto con HTML5.



### GitHub

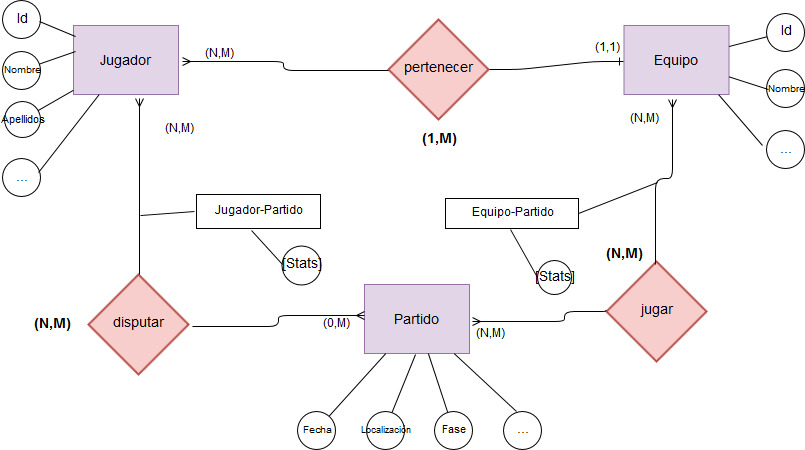
Se ha empleado GitHub y su aplicación para escritorio “GitHub Desktop” para alojar el proyecto. GitHub es una página para alojar proyectos colaborativos de programación empleando el sistema de control de versiones Git. En la bibliografía se incluye un enlace a la página de GitHub donde se encuentra alojado el proyecto.

## 

## Análisis

### Diagrama E/R

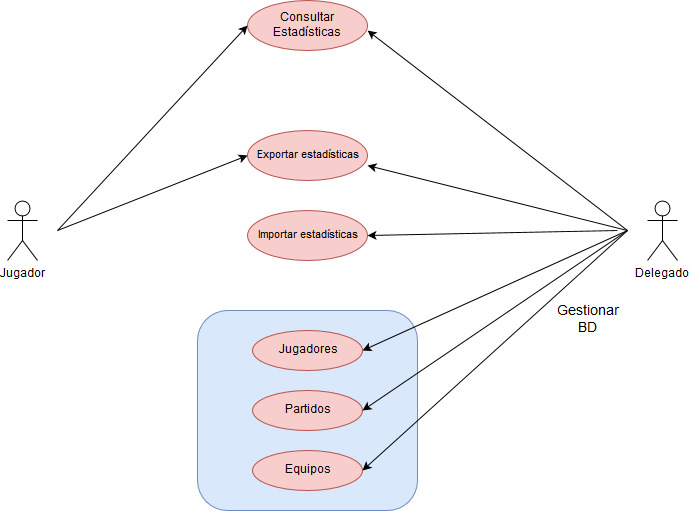
* Jugador: Los jugadores que participan en la competición, posee unos atributos básicos (nombre, apellidos, dorsal…), el equipo al que pertenecen (FK de Equipo) y todas las estadísticas acumuladas de la temporada (faltas que ha realizado, tiros libres que ha anotado...)
* Equipo: Los equipos de la competición, al igual que con el jugador, almacenamos una información básica sobre los mismos (nombre, sede…) y las estadísticas acumuladas para el equipo (puntos en la temporada, porcentaje de tiros de 3 del equipo…)
* Partido: La información de cada uno de los partidos que se juegan, posee dos FK de los equipos participantes, un listado de los jugadores que han sido convocados y otra información como la fecha o el lugar donde ha tenido lugar.
* Jugador-Partido: Tabla relacional que almacena el desempeño del jugador para un partido en concreto.
* Equipo-Partido: Tabla relacional que almacena el rendimiento del equipo durante un partido concreto.



### Casos de Uso

En el diagrama de casos de uso se indican las distintas funciones de la aplicación, se ha obviado el login para simplificar el diagrama. Existen los siguientes usos:

* Consultar estadísticas: Se engloban el consultar estadísticas de un partido concreto (un único jugador o el equipo en general) las estadísticas a lo largo de la temporada de un jugador o las del equipo.
* Exportar estadísticas: Se pueden exportar las estadísticas que estemos viendo a formato pdf.
* Importar estadísticas: Es la manera en la que se agrega información, el delegado agrega un nuevo partido (adjuntando un fichero .csv con la información de un partido). La aplicación extrae esta información y actualiza la base de datos.
* Gestionar BD: El delegado, como administrador, puede acceder y modificar directamente toda la información almacenada en la base de datos.

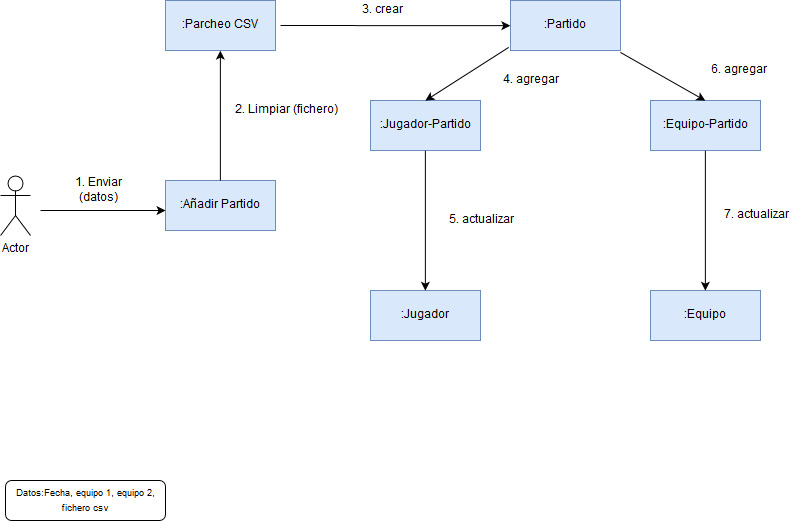


|  |  |
| --- | --- |
| Consultar Estadísticas | |
| Descripción | Un usuario accede a las estadísticas de un jugador. |
| Actores | Jugadores, Delegados |
| Precondiciones | Estar logueado |
| Curso Normal | 1. El usuario accede a un equipo 2. En el listado de jugadores selecciona uno 3. Se le muestra una página con las estadísticas del jugador |

|  |  |
| --- | --- |
| Importar Estadísticas | |
| Descripción | El delegado agrega un nuevo partido |
| Actores | Delegados |
| Precondiciones | Estar logueado |
| Curso Normal | 1. El delegado accede al formulario para agregar partidos. 2. Indica la fecha, equipos y adjunta el CSV 3. La aplicación extrae información del CSV 4. Crea un partido nuevo. 5. Actualiza las estadísticas acumuladas de Jugador y Equipo |
| Alternativas | 2. El fichero adjuntado no es un CSV  3. La aplicación devuelve un error y redirige al formulario. |

### Diagrama de colaboración

Se añade el diagrama de colaboración del importar estadísticas, al tratarse de la función principal de la aplicación.



### Diagrama navegación

## Diseño

### Estructura de archivos

## Implementación

### Back-End

### Front-End

# Resultados Obtenidos

# Conclusiones y Propuestas

# Anexo

## Estadísticas: Una introducción

# Bibliografía

### Estadísticas Baloncesto

* Mi santo padre

### Análisis

* Libro Entornos de Desarrollo

### Backend

* Documentación Django <https://docs.djangoproject.com/en/2.2/>
* Curso Python 3 desde 0
* Curso Django
* Python for Data Science
* <https://simpleisbetterthancomplex.com/> Blog sobre Django
* <https://simpleisbetterthancomplex.com/archive/>
* <https://ccbv.co.uk/> Django Class Based Views
* <https://pypi.org/project/django-chartjs/>

### Frontend

* ChartJs documentación
* Documentación Bootstrap