



# GRADO EN INGENIERÍA EN TECNOLOGÍAS DE LA TELECOMUNICACIÓN

Curso Académico 2018/2019

Trabajo Fin de Grado

## TORNEOS DE DR. SCRATCH

Autor : Álvaro Fernández Roig

Tutor : Dr. Gregorio Robles



# Trabajo Fin de Grado

Torneos de Dr. Scratch

**Autor :** Álvaro Fernández Roig

**Tutor :** Dr. Gregorio Robles Martínez

La defensa del presente Proyecto Fin de Carrera se realizó el día                      de  
de 2019, siendo calificada por el siguiente tribunal:

**Presidente:**

**Secretario:**

**Vocal:**

y habiendo obtenido la siguiente calificación:

**Calificación:**

Fuenlabrada, a                      de                      de 2019



*Dedicado a  
mi familia,  
por su permanente  
e incondicional sostén*



# Agradecimientos

Aquí vienen los agradecimientos... Aunque está bien acordarse de la pareja, no hay que olvidarse de dar las gracias a tu madre, que aunque a veces no lo parezca disfrutará tanto de tus logros como tú... Además, la pareja quizás no sea para siempre, pero tu madre sí.





# Resumen

Aquí viene un resumen del proyecto. Ha de constar de tres o cuatro párrafos, donde se presente de manera clara y concisa de qué va el proyecto. Han de quedar respondidas las siguientes preguntas:

- ¿De qué va este proyecto? ¿Cuál es su objetivo principal?
- ¿Cómo se ha realizado? ¿Qué tecnologías están involucradas?
- ¿En qué contexto se ha realizado el proyecto? ¿Es un proyecto dentro de un marco general?

Lo mejor es escribir el resumen al final.



# Summary

Here comes a translation of the “Resumen” into English. Please, double check it for correct grammar and spelling. As it is the translation of the “Resumen”, which is supposed to be written at the end, this as well should be filled out just before submitting.



# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Contexto . . . . .	1
1.2. Estructura de la memoria . . . . .	1
<b>2. Objetivos</b>	<b>3</b>
2.1. Objetivo general . . . . .	3
2.2. Objetivos específicos . . . . .	3
2.3. Planificación temporal . . . . .	4
<b>3. Estado del arte</b>	<b>5</b>
3.1. Python . . . . .	6
3.2. Django . . . . .	6
3.3. HTML . . . . .	6
3.4. CSS . . . . .	6
3.5. JavaScript . . . . .	6
3.6. jQuery . . . . .	6
3.7. Hairball . . . . .	6
3.8. Scratch . . . . .	6
3.9. Bootstrap . . . . .	6
<b>4. Diseño e implementación</b>	<b>7</b>
4.1. Arquitectura general . . . . .	7
4.1.1. Cliente . . . . .	8
4.1.2. Servidor . . . . .	8
4.2. Diseño e implementación por funcionalidad . . . . .	9

<b>5. Resultados</b>	<b>11</b>
<b>6. Conclusiones</b>	<b>13</b>
6.1. Consecución de objetivos . . . . .	13
6.2. Aplicación de lo aprendido . . . . .	13
6.3. Lecciones aprendidas . . . . .	13
6.4. Trabajos futuros . . . . .	14
<b>A. Manual de usuario</b>	<b>15</b>
<b>Bibliografía</b>	<b>17</b>

# Índice de figuras

4.1. Arquitectura general . . . . .	8
-------------------------------------	---





# Capítulo 1

## Introducción

El siguiente trabajo consta de la realización de un módulo independiente para la aplicación Dr. Scratch. Se trata del módulo de torneos de Dr. Scratch, que permitirá a los docentes la creación de torneos y la participación en ellos de sus alumnos.

### 1.1. Contexto

Dr. Scratch es una aplicación web que permite la evaluación de proyectos realizados a través de la herramienta Scratch. A través del análisis de los proyectos, Dr. Scratch puntúa en una escala de cero a tres una serie de parámetros relacionados con el pensamiento computacional, como el paralelismo o la abstracción, e identifica posibles errores en la realización de los proyectos, como duplicidades o código muerto.

(PENDIENTE)De manera didáctica como ayuda a los niños etc. (Bibliografía)  
(PENDIENTE)Autoevaluación, trabajo en equipo, proyectos, independencia(Bibliografía)

### 1.2. Estructura de la memoria

En esta sección se debería introducir la estructura de la memoria. Así:

- En el primer capítulo se hace una intro al proyecto.
- En el capítulo 2 (ojo, otra referencia automática) se muestran los objetivos del proyecto.
- A continuación se presenta el estado del arte.

■ ...

# Capítulo 2

## Objetivos

### 2.1. Objetivo general

La realización del módulo de torneos para Dr. Scratch surge bajo la premisa de realizar modificaciones en la aplicación de Dr. Scratch para tratar que ésta resulte más atractiva a los estudiantes que la utilicen y que esto ayude a su vez a fomentar un mayor uso de la aplicación original.

### 2.2. Objetivos específicos

- Promover el análisis de proyectos Scratch a través de Dr. Scratch a través de un apartado lúdico que motive y atraiga a los estudiantes.
- Proveer a los profesores de una aplicación administrable, que les permita tener el control de la creación, edición y borrado de las distintas entidades relacionadas con la creación de los torneos.
- Mejorar el control y la evaluación por parte de los profesores de los proyectos realizados por sus alumnos, proporcionándoles información centralizada sobre el avance y resultados de los proyectos a través de la aplicación.
- Permitir a los alumnos realizar una autoevaluación de sus proyectos Scratch mostrando información de ayuda para la mejora en los distintos parámetros evaluados.

- Realizar una aplicación intuitiva, cuya información sea comprensible tanto por profesores como por alumnos y cuyos procesos sean sencillos y tengan un aspecto visual atractivo y en consonancia la aplicación de Dr. Scratch existente.

### **2.3. Planificación temporal**

A mí me gusta que aquí pongáis una descripción de lo que os ha llevado realizar el trabajo. Hay gente que añade un diagrama de GANTT. Lo importante es que quede claro cuánto tiempo llevas (tiempo natural, p.ej., 6 meses) y a qué nivel de esfuerzo (p.ej., principalmente los fines de semana).

# Capítulo 3

## Estado del arte

Descripción de las tecnologías que utilizas en tu trabajo. Con dos o tres párrafos por cada tecnología, vale. Se supone que aquí viene todo lo que no has hecho tú.

Puedes citar libros, como el de Bonabeau et al. sobre procesos estigmérgicos [1].

También existe la posibilidad de poner notas al pie de página, por ejemplo, una para indicarte que visite la página de LibreSoft<sup>1</sup>.

---

<sup>1</sup><http://www.libresoft.es>

### **3.1. Python**

### **3.2. Django**

### **3.3. HTML**

### **3.4. CSS**

### **3.5. JavaScript**

### **3.6. jQuery**

### **3.7. Hairball**

### **3.8. Scratch**

### **3.9. Bootstrap**

# Capítulo 4

## Diseño e implementación

A continuación se detalla la aplicación torneos de Dr. Scratch, incluyendo su arquitectura y su modelo de datos.

### 4.1. Arquitectura general

La aplicación está desarrollada usando el framework Django, cuya arquitectura está basada en el modelo vista controlador MVC. Sin embargo, la terminología utilizada por Django varía con respecto a otros frameworks. En este caso, el controlador se denomina vista y la vista plantilla o template, por lo que en lugar de utilizar el acrónimo MVC se utiliza el acrónimo MTV: model, template, view<sup>1</sup>.

---

<sup>1</sup><https://docs.djangoproject.com/es/2.1/faq/general/#django-appears-to-be-a-mvc-framework-but-you-call-the-controller-the-view-and-the-view-the-template-how-come-you-don-t-use-the-standard-names>

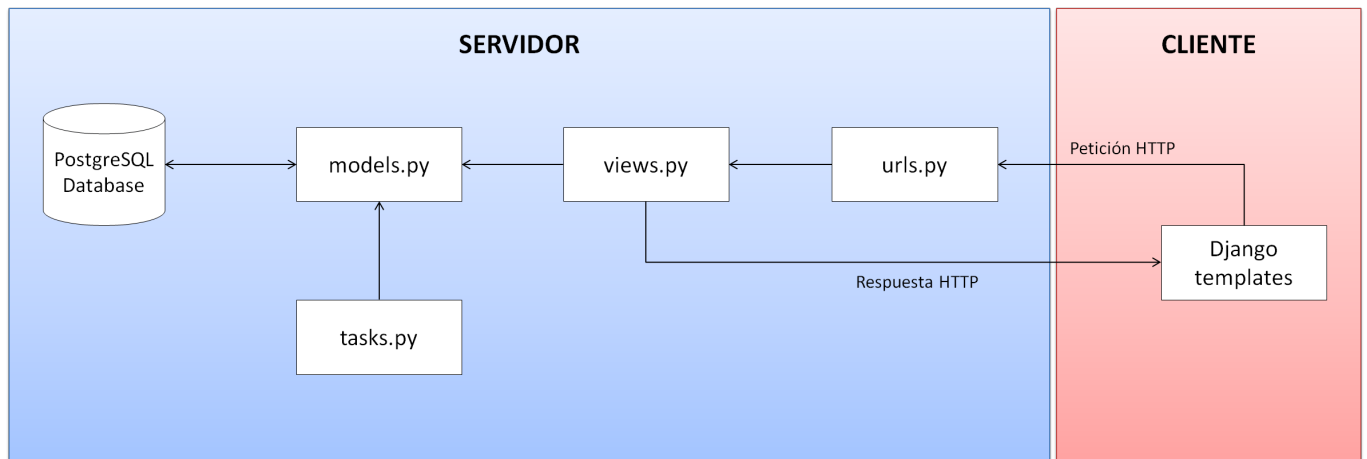


Figura 4.1: Arquitectura general

#### 4.1.1. Cliente

En el lado del cliente, se encuentran las plantillas de Django que serán visualizadas por los usuarios a través de un navegador web. Estas plantillas son archivos en formato HTML en los que se muestra la información de la respuesta recibida desde la vista. Así mismo, las plantillas dan formato a la información de respuesta usando hojas de estilo en formato CSS y scripts en lenguaje Javascript, ejecutados en el lado del cliente. En este caso, se ha utilizado el framework Bootstrap para el desarrollo del formato de las plantillas.

#### 4.1.2. Servidor

En el lado del servidor se realiza el desarrollo back-end al completo, desde la recepción de peticiones HTTP hasta el acceso a datos.

En primer lugar, las peticiones HTTP son recibidas por un despachador de URLs en el que se define un mapeo de expresiones regulares en el que cada expresión definida se enlaza con un método definido en la vista. De este modo cuando se recibe una petición HTTP que cumple una expresión regular del despachador de URLs, este invoca al método de la vista definido en el mapeo.

En la vista se encuentra la lógica de la aplicación, definiendo en ésta las operaciones a realizar y finalmente retornando la información correspondiente al cliente a través de una plantilla. En la vista se realiza también el acceso a la base de datos, este acceso se realiza a través de las



entidades existentes en el modelo. Cada entidad del modelo corresponde a una tabla existente en la base de datos y para realizar las consultas se realizan llamadas a los diferentes métodos de la API Queryset de Django. A través de estos métodos pueden realizarse tanto las consultas como la inserción, modificación o borrado de datos de la base de datos.

Cabe mencionar que en este caso se ha utilizado la librería Django Background Tasks<sup>2</sup> para la creación de procesos asíncronos. La función a ejecutar se encuentra definida en el archivo `tasks.py` que actúa como la vista, en este caso accede a la base de datos y envía un correo electrónico si se cumple una serie de condiciones.

## 4.2. Diseño e implementación por funcionalidad

---

<sup>2</sup><https://django-background-tasks.readthedocs.io/en/latest/>



# Capítulo 5

## Resultados

En este capítulo se incluyen los resultados de tu trabajo fin de grado.

Si es una herramienta de análisis lo que has realizado, aquí puedes poner ejemplos de haberla utilizado para que se vea su utilidad.



# Capítulo 6

## Conclusiones

### 6.1. Consecución de objetivos

Esta sección es la sección espejo de las dos primeras del capítulo de objetivos, donde se planteaba el objetivo general y se elaboraban los específicos.

Es aquí donde hay que debatir qué se ha conseguido y qué no. Cuando algo no se ha conseguido, se ha de justificar, en términos de qué problemas se han encontrado y qué medidas se han tomado para mitigar esos problemas.

### 6.2. Aplicación de lo aprendido

Aquí viene lo que has aprendido durante el Grado/Máster y que has aplicado en el TFG/TFM. Una buena idea es poner las asignaturas más relacionadas y comentar en un párrafo los conocimientos y habilidades puestos en práctica.

1. a

2. b

### 6.3. Lecciones aprendidas

Aquí viene lo que has aprendido en el Trabajo Fin de Grado/Máster.

1. a

2. b

## **6.4. Trabajos futuros**

Ningún software se termina, así que aquí vienen ideas y funcionalidades que estaría bien tener implementadas en el futuro.

Es un apartado que sirve para dar ideas de cara a futuros TFGs/TFM.

# **Apéndice A**

## **Manual de usuario**





# Bibliografía

- [1] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, Inc., 1999.