



**MANUAL TÉCNICO DEL SISTEMA**



## CONTENIDO

	Pág.
PRESENTACIÓN .....	5
RESUMEN.....	6
OBJETIVO.....	7
FINALIDAD DEL MANUAL .....	7
INTRODUCCIÓN.....	8
1. ASPECTOS TÉCNICOS.....	9
1.1. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO .....	9
1.1.1. SublimeText .....	9
1.1.2. GitHub.....	9
1.1.3. Docker.....	10
2. DIAGRAMAS DE MODELAMIENTO .....	11
2.1. DIAGRAMA DE CLASES.....	11
2.2. DIAGRAMA DE CASOS DE USO.....	12
2.3. DICCIONARIO DE DATOS.....	13
3. ASPECTO TÉCNICO DEL DESARROLLO DEL SISTEMA .....	14
3.1. MODIFICACIÓN LOCAL.....	14
4. REQUERIMIENTOS DEL SOFTWARE .....	21
4.1. REQUISITOS MÍNIMOS.....	21
BIBLIOGRAFÍA.....	22

## LISTA DE FIGURAS

	Pág.
Figura 1. Diagrama de modelado de clases.....	11
Figura 2. Diagrama de casos de uso .....	12
Figura 3. Página web de descarga Python .....	14
Figura 4. Página web de descarga Sublime Text.....	15
Figura 5. Página web de descarga GitHub .....	15
Figura 6. Clonación repositorio de GitHub .....	16
Figura 7. Ventana de cambios del repositorio .....	16
Figura 8. Validación del archivo requirements.txt.....	17
Figura 9. Ejecución de comando para instalar paquetes necesarios.....	17
Figura 10. Creación de súper usuario para la administración de Django. ....	18
Figura 11. Inicialización del servidor local .....	18
Figura 12. Ingreso a la administración de Django .....	19
Figura 13. Administración de Django .....	19
Figura 14. Cambios repositorio en GitHub Desktop .....	20



## LISTA DE TABLAS

	Pág.
Tabla 1. Diccionario de datos modelo usuarios.....	13
Tabla 2. Diccionario de datos modelo niveles .....	13
Tabla 3. Diccionario de datos modelo actividades .....	13



Manual técnico del sistema

## **PRESENTACIÓN**

El siguiente manual se ha desarrollado con la finalidad de dar a conocer la información necesaria para realizar mantenimiento, instalación y exploración del software funnyYD, el cual consta de diferentes actividades para el mejoramiento de los procesos de psicomotricidad gruesa y concentración en los niños con síndrome de Down del Colegio Espíritu Santo Marianista.

El manual ofrece la información necesaria de ¿cómo está realizado el software? para que la persona (Desarrollador en el framework DJANGO) que quiera editar el software lo haga de una manera apropiada, dando a conocer la estructura del desarrollo del aplicativo.



Manual técnico del sistema

## RESUMEN

El manual detalla los aspectos técnicos e informáticos del software funnYD con la finalidad de explicar la estructura del aplicativo al personal que quiera administrarlo, editarlo o configurarlo. La siguiente guía se encuentra dividida en las herramientas que se usaron para la creación del software con una breve explicación paso a paso, El aplicativo web maneja diferentes funcionalidades el cual requieren de hardware y software el cual se explicará que funcionamiento realiza cada uno de ellos, dando sugerencias para el debido uso del sistema de información.



Manual técnico del sistema

## **OBJETIVO**

Dar a conocer el uso adecuado del software funnYD en aspectos técnicos de manera descriptiva e ilustrada sobre los componentes y funcionalidades que conforman el buen funcionamiento del sistema de información.

## **FINALIDAD DEL MANUAL**

La finalidad de éste manual técnico es instruir a la persona que quiera administrar, editar o configurar el software funnYD usando las debidas herramientas.



## INTRODUCCIÓN

El manual se realiza con el fin de detallar el software funnYD en términos técnicos para que la persona que vaya a administrar, editar o configurar el aplicativo lo haga de una manera apropiada. El documento se encuentra dividido en las siguientes secciones:

- **ASPECTOS TEÓRICOS:** Se darán a conocer conceptos, definiciones y explicaciones de los componentes del aplicativo desde un punto de vista teórico para mayor entendimiento por parte del lector sobre el funcionamiento del sistema de información e herramientas.
- **DIAGRAMAS DE MODELAMIENTO:** Se compone por diagramas e ilustraciones alusivos al funcionamiento del aplicativo.
- **ASPECTO TÉCNICO DEL DESARROLLO DEL SISTEMA:** Corresponde a la instrucción al lector sobre los componentes del aplicativo desde una perspectiva técnica en los aspectos de almacenamiento de datos, estructura del desarrollo y recomendaciones del uso debido del aplicativo.
- **REQUERIMIENTOS DEL SOFTWARE:** Detalla los requerimientos básicos necesarios para el funcionamiento del software.





## 1. ASPECTOS TÉCNICOS

El aplicativo funnYD tiene la finalidad de mejorar los procesos de psicomotricidad gruesa y concentración en los niños con síndrome de Down por medio de actividades didácticas. Se recomienda que el siguiente manual sea manipulado únicamente por la persona que quiera administrar, editar o configurar el software funnYD para velar por la seguridad de los datos que se almacenan en la base de datos ya que pueden ser usados para otros fines.

### 1.1. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO

Es ésta sección se procede a explicar las herramientas informáticas empleadas para el desarrollo del aplicativo:

#### 1.1.1. SublimeText

SublimeText es un editor de texto el cual es caracterizado por sus amplias funcionalidades a la hora de desarrollar software, tiene incorporado la estructura en la sintaxis de diferentes lenguajes de programación con una interfaz amigable, en el desarrollo de funnYD se trabajó con esta herramienta de marcado ya que indica en donde está ubicado los errores de sintaxis y la codificación se maneja de forma dinámica teniendo en cuenta que se requieren de paquetes adicionales.

Sublime Text está disponible para OS X, Windows y Linux. Una licencia es todo lo que necesita para utilizar Sublime Text en todos los equipos de su propiedad, sin importar el sistema operativo que utiliza, pero también ofrece una licencia gratuita de prueba. Usa un conjunto de herramientas de interfaz de usuario personalizada, optimizado para la velocidad y la belleza, aprovechando al mismo tiempo la funcionalidad nativa en cada plataforma.

La versión usada para el desarrollo de funnYD es la versión gratuita, el cual funciona con una publicidad de compra del producto al momento de guardar los cambios realizados en el código, por lo tanto, no es una limitante para el desarrollo del aplicativo.

#### 1.1.2. GitHub

GitHub es una plataforma de desarrollo colaborativo de software para alojar proyectos utilizando el sistema de control de versiones Git. GitHub aloja tu repositorio de código y te brinda herramientas muy útiles para el trabajo en equipo,



## Manual técnico del sistema

dentro de un proyecto. Además de eso, puedes contribuir a mejorar el software de los demás. Para poder alcanzar esta meta, GitHub provee de funcionalidades para hacer un fork y solicitar pulls. Realizar un fork es simplemente clonar un repositorio ajeno (genera una copia en tu cuenta), para eliminar algún bug o modificar cosas de él. Una vez realizadas tus modificaciones puedes enviar un pull al dueño del proyecto. Éste podrá analizar los cambios que has realizado fácilmente, y si considera interesante tu contribución, adjuntarlo con el repositorio original. (Luciano Castillo, 2012)

Para funnYD, se maneja un repositorio privado el cual se contribuye diferentes commits con ajustes o cambios que se realizan en el software, haciendo así un trabajo en equipo en el desarrollo del aplicativo.

### 1.1.3. Docker

Envuelven una pieza de software en un sistema de archivos completo que contiene todo lo necesario para funcionar: código, tiempo de ejecución, herramientas del sistema, las bibliotecas del sistema - cualquier cosa que se puede instalar en un servidor. Esto garantiza que el software se ejecutará siempre la misma, independientemente de su medio ambiente. (Docker, s.f.)

Se usa Docker para poder desarrollar de una manera más cómoda, ya que este software contiene millones de librerías y herramientas para poder codificar.

### 1.1.4. Heroku

Heroku es una plataforma en la nube que permite a las empresas a construir, entregar, supervisar y aplicaciones escala - somos la manera más rápida para ir de la idea al URL, sin pasar por todos esos problemas de infraestructura. (Heroku, s.f.)

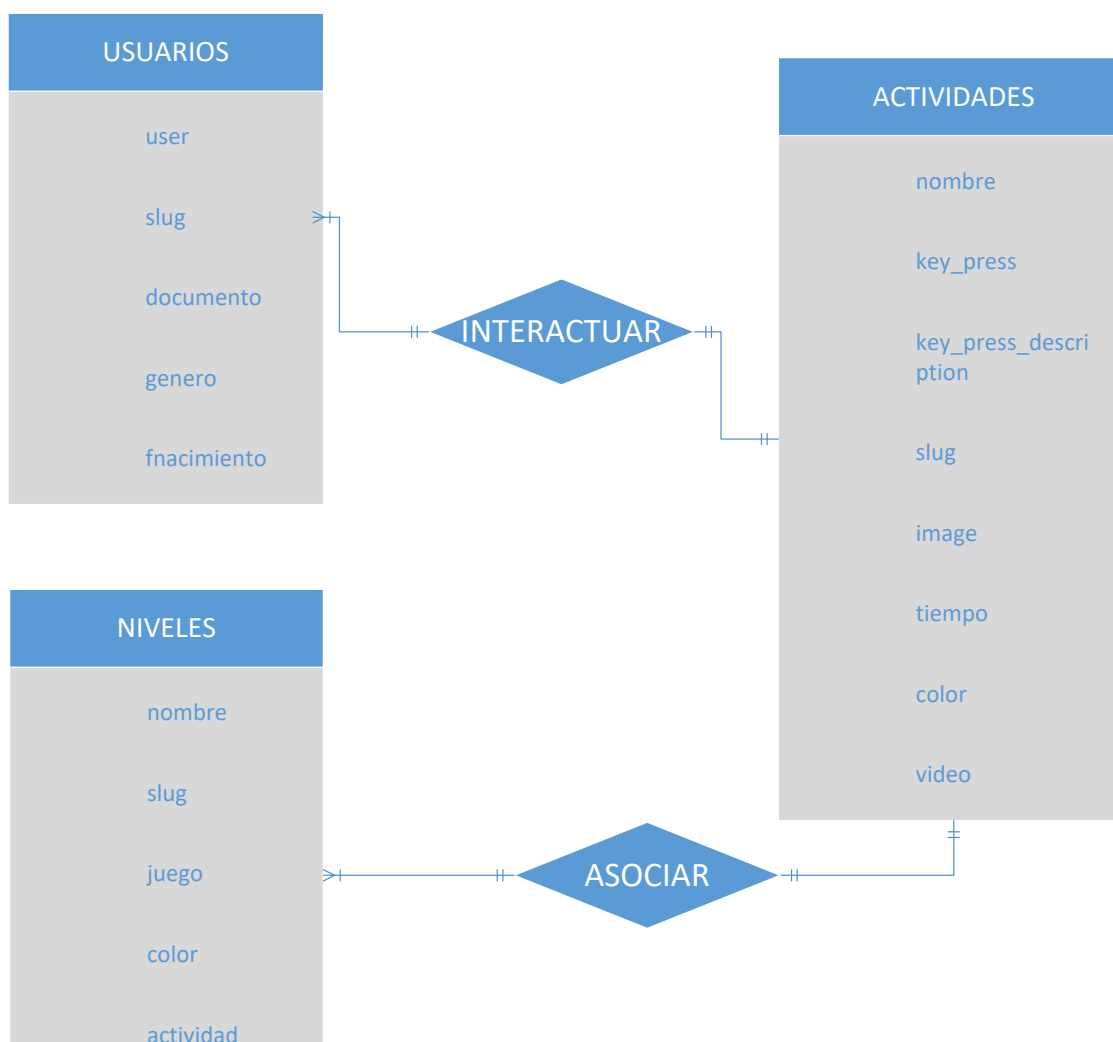
En heroku el software está desplegado conectado con GitHub, el cual se usa una cuenta free que el proveedor otorga para que el aplicativo pueda estar en la web con una infraestructura estable.

## 2. DIAGRAMAS DE MODELAMIENTO

### 2.1. DIAGRAMA DE CLASES

El diagrama de clases está compuesto de las entidades y atributos que se crearon para el almacenamiento de datos del software.

Figura 1. Diagrama de modelado de clases



Fuente: Por los autores.

En la figura 1, se detalla cada una de las entidades (modelos) usadas para el almacenamiento de datos del aplicativo, en el cual cada una de ellas realiza las siguientes funciones:

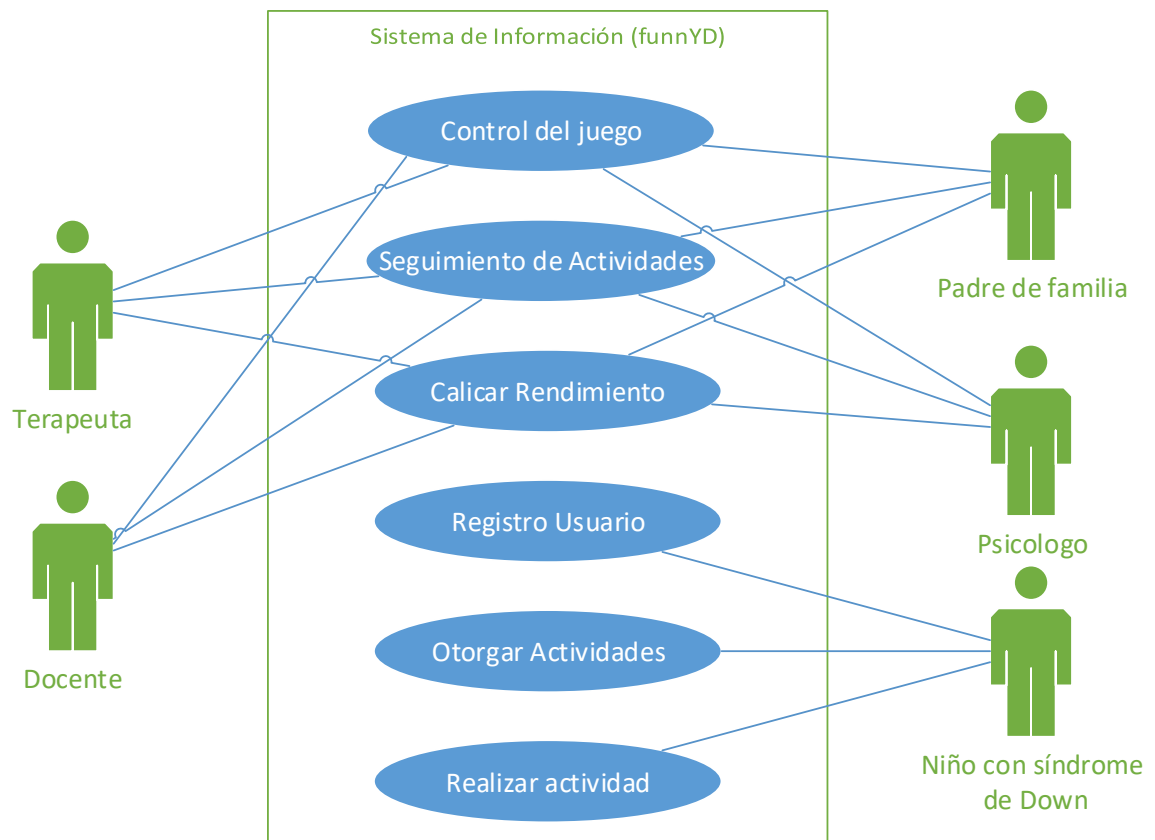
## Manual técnico del sistema

- **USUARIOS:** Almacena los datos de los usuarios que se registran para poder jugar, en este caso el niño con síndrome de Down.
- **ACTIVIDADES:** Almacena la composición y funcionalidades de cada una de las actividades que se encuentran en el software.
- **NIVELES:** Almacena cada uno de los niveles que posee cada actividad del aplicativo.

### 2.2. DIAGRAMA DE CASOS DE USO

En el diagrama de caso de uso se detalla el papel a desempeñar en relación con el aplicativo por parte de las personas relacionadas, en este caso el actor principal es el niño con síndrome de Down, y sus administradores o acompañantes son: El terapeuta, docente, padre de familia o psicólogo.

Figura 2. Diagrama de casos de uso



Fuente: Por los autores.

### 2.3. DICCIONARIO DE DATOS

Para el almacenamiento de datos del software, se definen los campos necesarios para cada una de las entidades relacionadas con el aplicativo.

Tabla 1. Diccionario de datos modelo usuarios

USUARIOS			
NOMBRE CAMPO	TIPO DATO	TAMAÑO	DESCRIPCIÓN
user	OneToOneField		Nombre usuario.
slug	SlugField		Identificador de usuario.
documento	PositiveIntegerField		Documento del niño.
genero	CharField	2	Genero del niño
fnacimiento	DateField		Edad del niño

Fuente: Por los autores.

Tabla 2. Diccionario de datos modelo niveles

NIVELES			
NOMBRE CAMPO	TIPO DATO	TAMAÑO	DESCRIPCIÓN
nombre	CharField	100	Nombre del nivel.
slug	SlugField		Identificador del nivel.
juego	CharField	50	Nombre del juego.
color	CharField	50	Color del nivel.
actividad	ForeignKey		Llave foránea con la actividad.
image	ImageField		Edad del niño

Fuente: Por los autores.

Tabla 3. Diccionario de datos modelo actividades

ACTIVIDADES			
NOMBRE CAMPO	TIPO DATO	TAMAÑO	DESCRIPCIÓN
nombre	CharField	100	Nombre de la actividad.
key_press	CharField	3	Nombre de la tecla a presionar.
key_press_description	CharField	100	Descripción de la tecla a presionar.
slug	SlugField		Identificador de la actividad.
image	ImageField		Ubicación imagen de la actividad.
tiempo	IntegerField		Almacena el tiempo de la actividad.
color	CharField	50	Color de la actividad.
video	CharField	200	Almacena de video de la actividad.

Fuente: Por los autores.

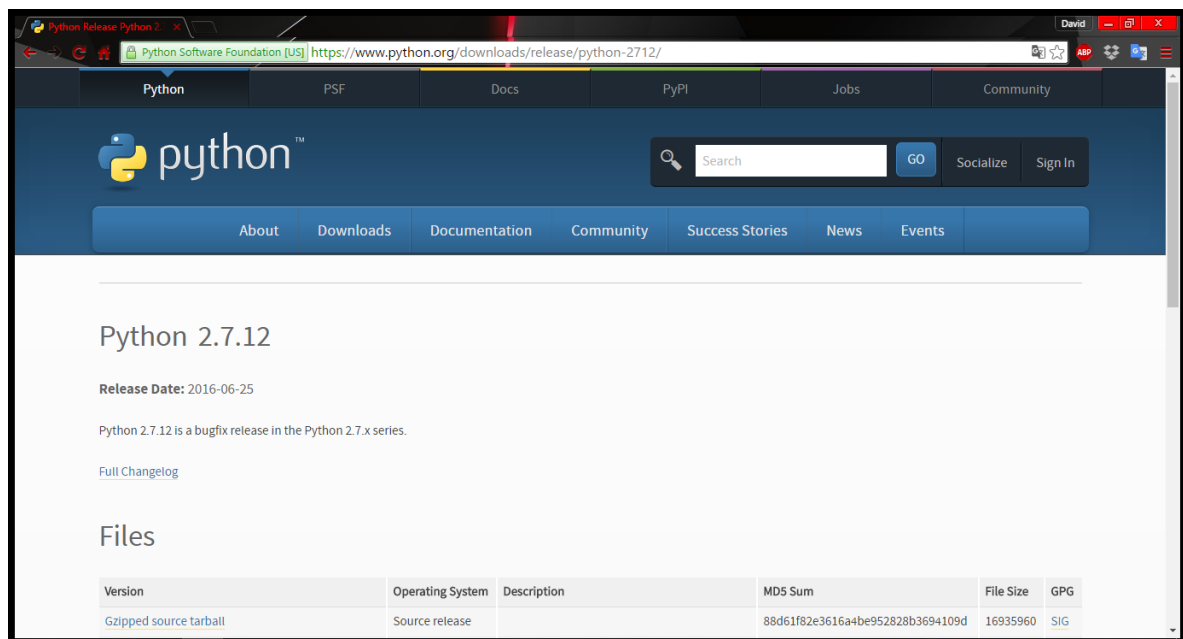
### 3. ASPECTO TÉCNICO DEL DESARROLLO DEL SISTEMA

En la siguiente sección se procede a realizar una descripción detallada sobre los aspectos técnicos del aplicativo, relacionado con la instalación de las herramientas necesarias para realizar modificaciones requeridas de manera ordenada.

#### 3.1. MODIFICACIÓN LOCAL

Si el desarrollador quiere realizar modificaciones del software de manera local, tendrá que realizar la instalación de componentes adicionales, para empezar se debe instalar Python versión 2.7.12, el cual se consigue de manera gratuita en la página [www.python.org](http://www.python.org), consiguiendo se debe de descargar pip (Sistema de gestión de paquetes de Python) para poder instalar los requerimientos del software.

Figura 3. Página web de descarga Python

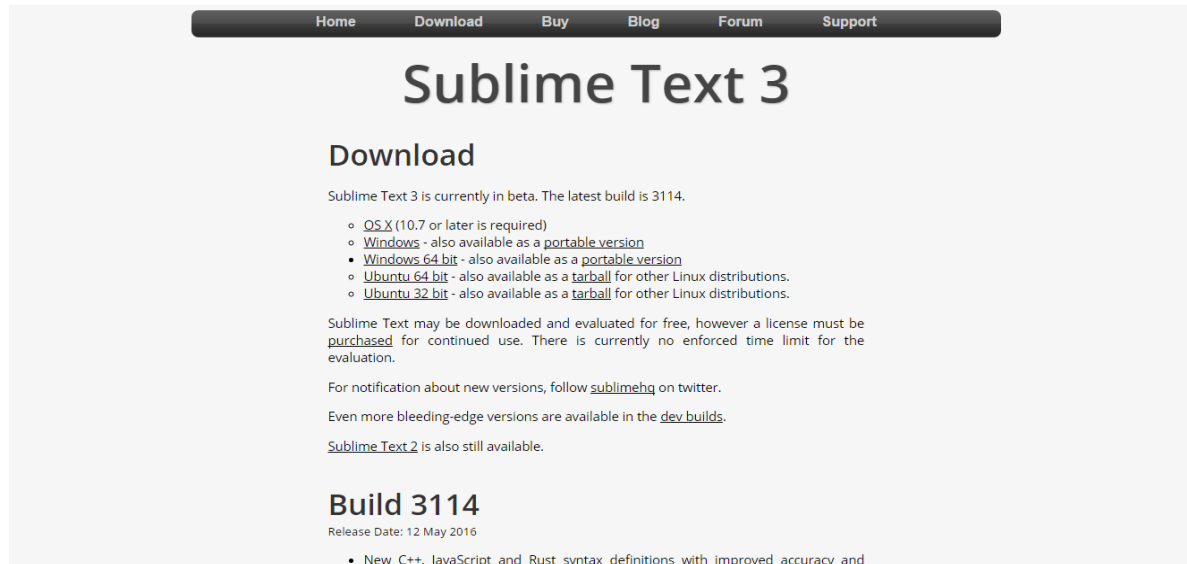


Fuente: (python, s.f.)

Al instalar PYTHON, procedemos a instalar el editor de texto para código Sublime Text, descargamos la última versión en la página oficial, esta descarga es con licencia gratuita.

## Manual técnico del sistema

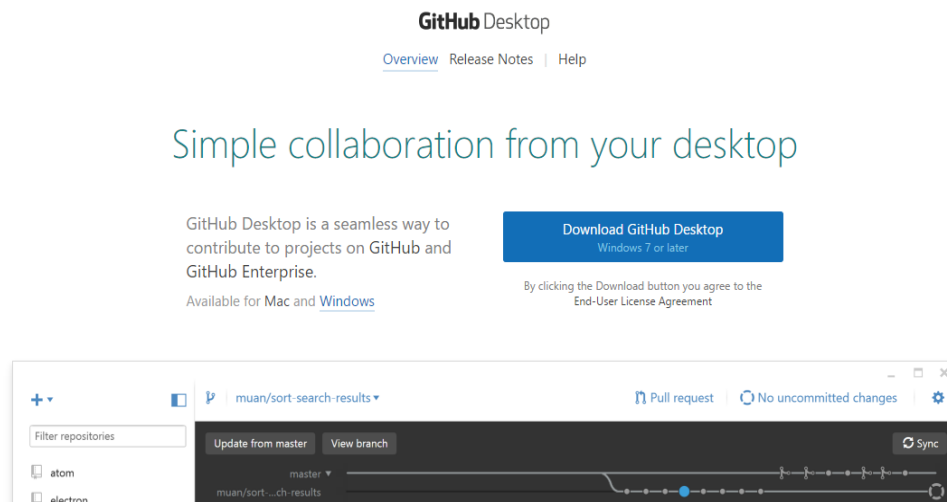
Figura 4. Página web de descarga Sublime Text



Fuente: (Sublime Text, s.f.)

Se continúa descargando el aplicativo GitHub para escritorio, se realiza desde la página oficial, con el fin de poder obtener el repositorio y realizarle cambios al software llevando un historial.

Figura 5. Página web de descarga GitHub

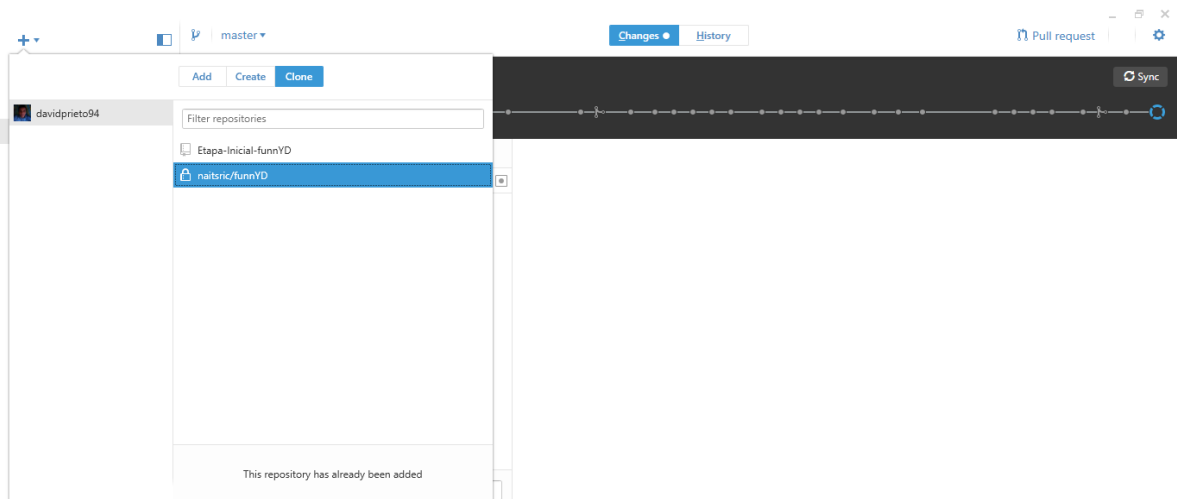


Fuente: (GitHub, s.f.)

Abrir el aplicativo GitHub después de instalado, dirigirse al logo de adición y dar clic en Clone, esa función clona el repositorio en un lugar indicado por el usuario. Se valida que quede en la venta de cambios sin que haya generado error alguno.

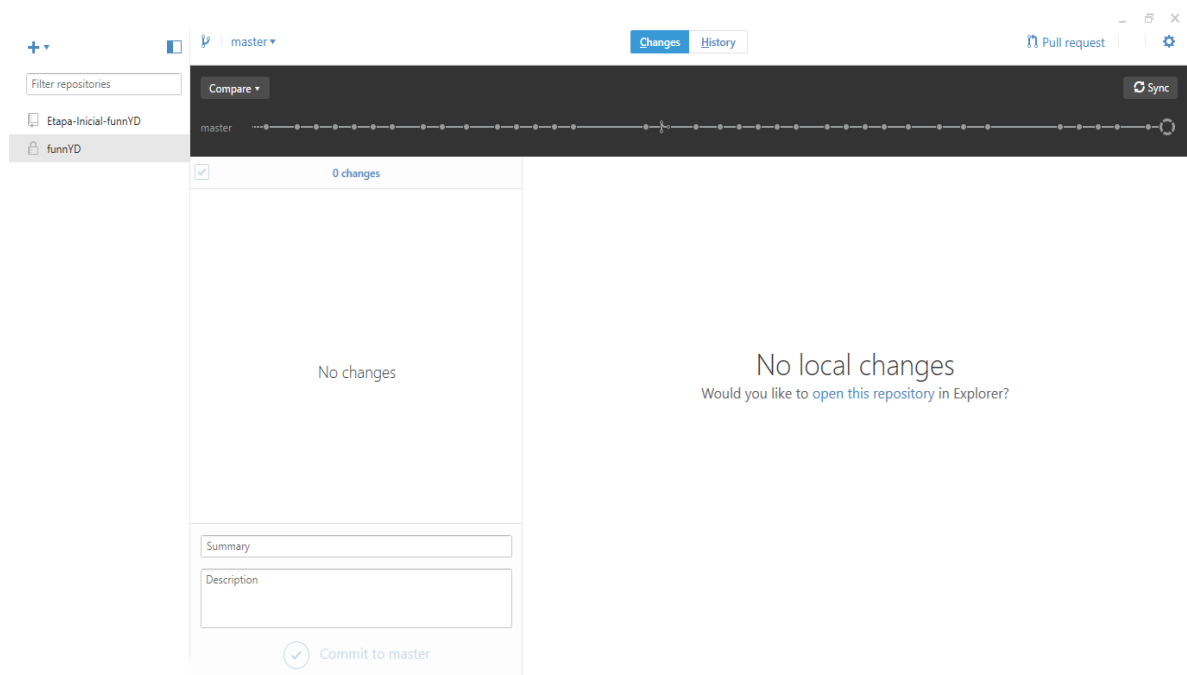
## Manual técnico del sistema

Figura 6. Clonación repositorio de GitHub



Fuente: Por los autores.

Figura 7. Ventana de cambios del repositorio



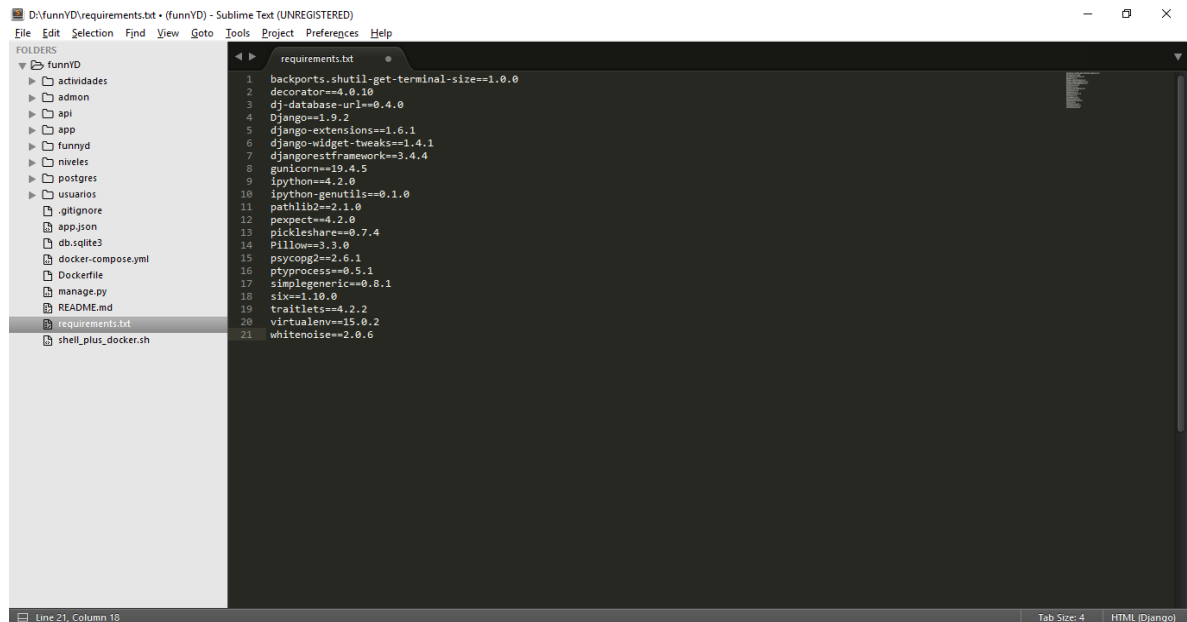
Fuente: Por los autores.

Al instalar Sublime Text, abrir la aplicación y arrastrar la carpeta donde está ubicado el proyecto creado en DJANGO, para poder visualizar los archivos que contiene. Abrimos el archivo requirements.txt para validar que estén los paquetes que se requieren para poder iniciar con las modificaciones del software.



## Manual técnico del sistema

Figura 8. Validación del archivo requirements.txt

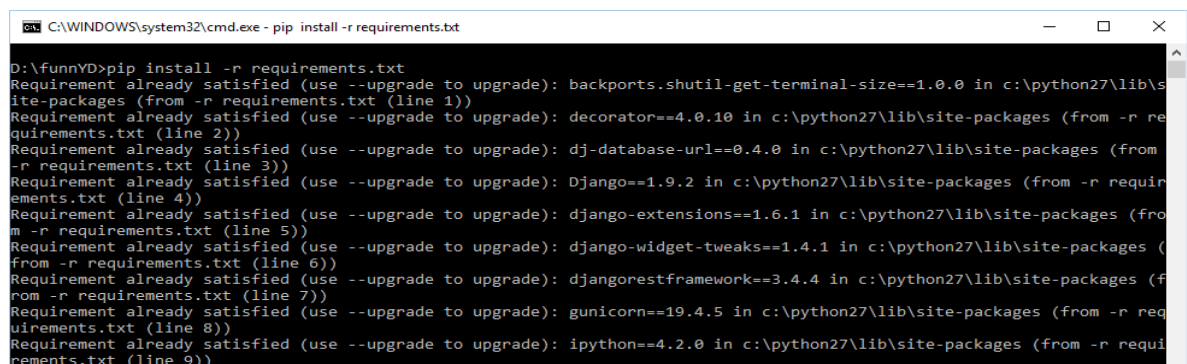


```
D:\funnyYD\requirements.txt • (funnyYD) - Sublime Text (UNREGISTERED)
File Edit Selection Fjnd View Goto Tools Project Preferences Help
FOLDERS
  funnyYD
    actividades
    admon
    api
    app
    funnyd
    niveles
    postgres
    usuarios
    .gitignore
    appjson
    db.sqlite3
    docker-compose.yml
    Dockerfile
    manage.py
    README.md
    requirements.txt
    shell_plus_docker.sh
requirements.txt
1 backports.shutil-get-terminal-size==1.0.0
2 decorator==4.0.10
3 dj-database-url==0.4.0
4 Django==1.9.2
5 django-extensions==1.6.1
6 django-widget-tweaks==1.4.1
7 djangorestframework==3.4.4
8 gunicorn==19.4.5
9 ipython==4.2.0
10 ipython-genutils==0.1.0
11 pathlib2==2.1.0
12 pexpect==4.2.0
13 pickleshare==0.7.4
14 pillow==3.3.0
15 psycopg2==2.6.1
16 ptyprocess==0.5.1
17 simplegeneric==0.8.1
18 six==1.10.0
19 traitlets==4.2.2
20 virtualenv==15.0.2
21 whitenoise==2.0.6
```

Fuente: Por los autores.

Al validar los requerimientos, procedemos a abrir el símbolo del sistema (cmd) para ejecutar el comando `pip install -r requirements.txt` con el fin de instalar los paquetes necesarios para poder inicializar el servidor local de DJANGO.

Figura 9. Ejecución de comando para instalar paquetes necesarios



```
C:\WINDOWS\system32\cmd.exe - pip install -r requirements.txt
D:\funnyYD>pip install -r requirements.txt
Requirement already satisfied (use --upgrade to upgrade): backports.shutil-get-terminal-size==1.0.0 in c:\python27\lib\site-packages (from -r requirements.txt (line 1))
Requirement already satisfied (use --upgrade to upgrade): decorator==4.0.10 in c:\python27\lib\site-packages (from -r requirements.txt (line 2))
Requirement already satisfied (use --upgrade to upgrade): dj-database-url==0.4.0 in c:\python27\lib\site-packages (from -r requirements.txt (line 3))
Requirement already satisfied (use --upgrade to upgrade): Django==1.9.2 in c:\python27\lib\site-packages (from -r requirements.txt (line 4))
Requirement already satisfied (use --upgrade to upgrade): django-extensions==1.6.1 in c:\python27\lib\site-packages (from -r requirements.txt (line 5))
Requirement already satisfied (use --upgrade to upgrade): django-widget-tweaks==1.4.1 in c:\python27\lib\site-packages (from -r requirements.txt (line 6))
Requirement already satisfied (use --upgrade to upgrade): djangorestframework==3.4.4 in c:\python27\lib\site-packages (from -r requirements.txt (line 7))
Requirement already satisfied (use --upgrade to upgrade): gunicorn==19.4.5 in c:\python27\lib\site-packages (from -r requirements.txt (line 8))
Requirement already satisfied (use --upgrade to upgrade): ipython==4.2.0 in c:\python27\lib\site-packages (from -r requirements.txt (line 9))
```

Fuente: Por los autores.

Si el administrador o desarrollador desea crear un usuario para poder ver la administración de Django, lo realiza con el comando `python manage.py createsuperuser`

## Manual técnico del sistema

Figura 10. Creación de súper usuario para la administración de Django.

```
C:\WINDOWS\system32\cmd.exe

D:\funnyD>python manage.py createsuperuser
D:\funnyD\api\urls.py:7: RemovedInDjango110Warning: django.conf.urls.patterns() is deprecated and will be removed in Django 1.10. Update your urlpatterns to be a list of django.conf.urls.url() instances instead.
  url(r'^niveles/', include('niveles.urls', namespace='niveles')),

Username (leave blank to use 'david'): administrador
Email address:
Password:
Password (again):
Superuser created successfully.
```

Fuente: Por los autores.

Para iniciar el servidor de Django y realizar cambios localmente, se ejecuta el comando `python manage.py runserver`, se debe tener claro que no debe haber errores en el código.

Figura 11. Inicialización del servidor local

```
C:\WINDOWS\system32\cmd.exe - python manage.py runserver

D:\funnyD>python manage.py runserver
Performing system checks...

D:\funnyD\api\urls.py:7: RemovedInDjango110Warning: django.conf.urls.patterns() is deprecated and will be removed in Django 1.10. Update your urlpatterns to be a list of django.conf.urls.url() instances instead.
  url(r'^niveles/', include('niveles.urls', namespace='niveles')),

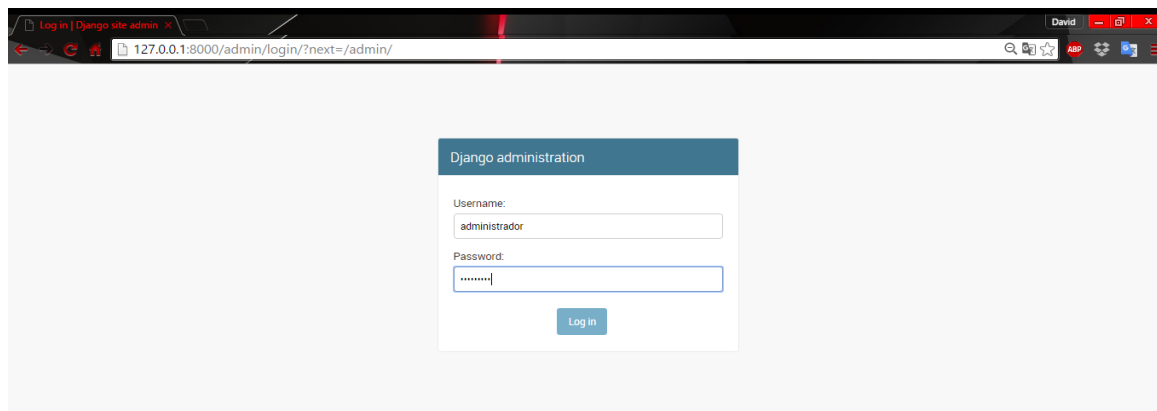
System check identified no issues (0 silenced).
August 31, 2016 - 16:48:56
Django version 1.9.2, using settings 'funnyd.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Fuente: Por los autores.

Ir a la URL **localhost:8000/admin** para abrir la administración de Django, después se ingresa el usuario creado anteriormente, para el software se crea un usuario por defecto el cual es **Administrador** y su contraseña es **Adm1n1\$tr4d0r**

## Manual técnico del sistema

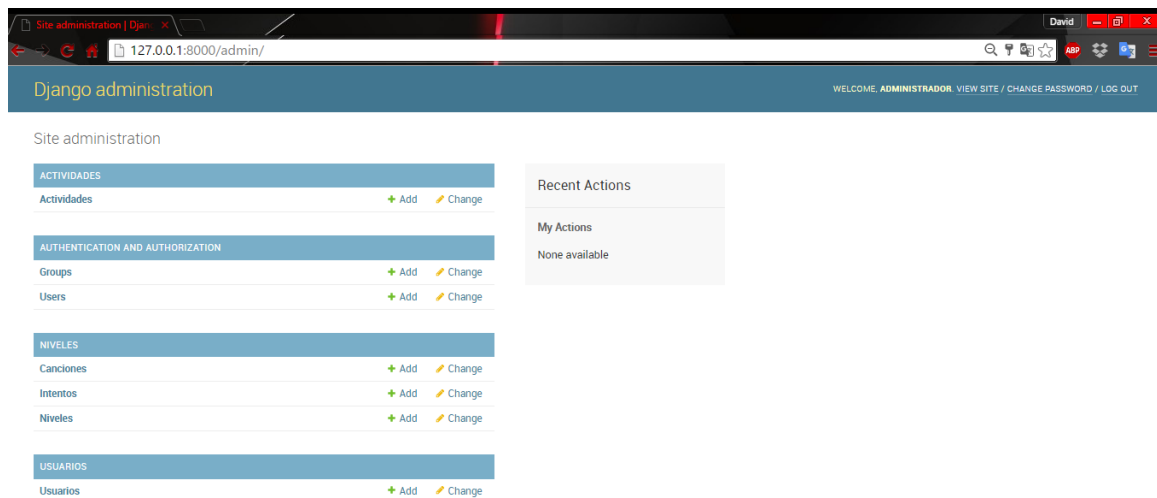
Figura 12. Ingreso a la administración de Django



Fuente: Por los autores.

Al ingresar a la administración de Django, si se requiere agregar, quitar o modificar datos manualmente a los modelos lo pueden realizar en esta sección.

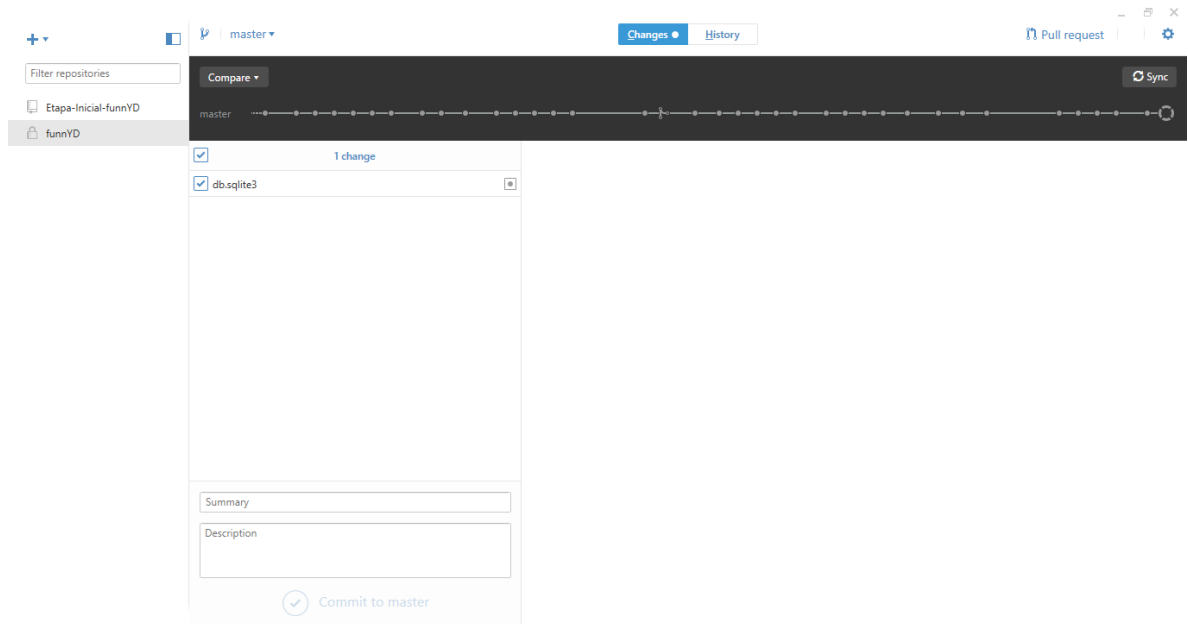
Figura 13. Administración de Django



Fuente: Por los autores.

Si se realizan cambios de código, base de datos o alguna modificación del repositorio descargado, el aplicativo GitHub Desktop reconoce las modificaciones y da la opción de ingresar Summary (Resumen) y Description (Descripción), al momento de llenar los dos campos, se realiza un Commit (Envío de cambios) al repositorio.

Figura 14. Cambios repositorio en GitHub Desktop



Fuente: Por los autores.



Manual técnico del sistema

## 4. REQUERIMIENTOS DEL SOFTWARE

En esta sección se detallará los requisitos mínimos del sistema para poder ejecutar los aplicativos usados para modificar el software funnyYD.

### 4.1. REQUISITOS MÍNIMOS

- Sistema Operativo: Windows 7
- Procesador: Intel Core Celeron
- Memoria RAM: 1GB
- Disco Duro: 1GB
- Resolución de pantalla: 1280 x 720 pixeles
- Periféricos: Teclado, ratón, Bocinas (Opcional)



## BIBLIOGRAFÍA

- Docker. (s.f.). *What is Docker?* Retrieved from <https://www.docker.com/what-docker>
- GitHub. (s.f.). *GitHub Desktop*. Retrieved from <https://desktop.github.com/>
- Heroku. (s.f.). *Lo que es Heroku*. Retrieved from <https://www.heroku.com/what>
- Luciano Castillo. (2012). *Conociendo GitHub*. Retrieved from <http://conociendogithub.readthedocs.io/en/latest/data/introduccion/>
- python. (s.f.). *Python*. Retrieved from <https://www.python.org/downloads/release/python-2712/>
- Sublime Text. (s.f.). *Sublime Text - Download*. Retrieved from <https://www.sublimetext.com/3>