



Albert Sanchez Lafuente

Grado en Ingeniería en Tecnologías Industriales

Junio 2016

Trabajo de Final de Grado
Grado en Ingeniería en Tecnologías Industriales

**Creación de un programa informático para
administrar SOS ETSEIB**

MEMORIA

Autor: Albert Sanchez Lafuente
Director: Lluís Solano Albajes
Convocatoria: Junio 2016



Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona



Resumen

El objetivo de este trabajo es realizar un software que permita automatizar, agilizar y facilitar las tareas de administración del proyecto SOS ETSEIB. Este proyecto consiste en una web dirigida por estudiantes de nuestra escuela (ETSEIB) en la cual se fomenta la compartición de apuntes entre los estudiantes de la misma. Esta compartición se realiza a través de carpetas de Dropbox, teniéndose que enviar una invitación por cada usuario y asignatura. Para saber qué asignaturas enviar a cada persona, se pide a los alumnos que rellenen vía web un formulario con estos datos

Previamente a este trabajo la compartición se realizaba manualmente, es decir descargando manualmente las inscripciones de los usuarios y sus asignaturas, y enviando las invitaciones a cada persona a cada asignatura manualmente, lo cual era un proceso muy lento.

Con este trabajo se ha automatizado este proceso, descargando las inscripciones y enviando invitaciones automáticamente, además de crear una interfaz gráfica que permita un uso intuitivo de la aplicación.

Para la realización de este software se ha empleado el lenguaje de programación Python en conjunto con diversas librerías, las que mayor importancia tienen sobre el proyecto son las siguientes:

Sqlite: ha sido utilizada para gestionar una base de datos en la que se guardan todos los datos de los usuarios, es decir todas las inscripciones.

Dropbox: se ha empleado la librería de la API de Dropbox para realizar todas las tareas de conexión a este servicio y administración de las invitaciones a las carpetas compartidas.

Kivy: se ha utilizado para crear la interfaz gráfica del software.

En este documento se pretende explicar al más mínimo detalle cómo funciona este proceso y este programa. Además se pretende que este documento sirva de guía para los futuros estudiantes que se involucren en el proyecto y quieran darle continuidad.

Sumario

RESUMEN	1
SUMARIO	3
1. GLOSARIO	5
2. PREFACIO	7
2.1. Origen del proyecto y motivación	7
2.2. Requerimientos previos	8
3. INTRODUCCIÓN	9
3.1. Objetivos del trabajo	10
3.2. Estado previo a este trabajo	10
4. DESCRIPCIÓN DEL PROBLEMA	11
4.1. Funcionamiento del proceso de administración	11
4.1.1. Descarga de datos	11
4.1.2. Procesado de datos	12
4.1.3. Registro de la persona en un Excel	13
4.1.4. Envío de invitaciones a las carpetas de Dropbox	13
4.1.5. Envío de un email	15
4.2. Objetivos de automatización	16
4.2.1. Descarga de datos	16
4.2.2. Procesado de datos	16
4.2.3. Base de datos	16
4.2.4. Envío de invitaciones a carpetas de Dropbox	17
4.2.5. Envío de emails	17
4.3. Interfaz gráfica	17
4.4. Conclusión	17
5. DESARROLLO DEL TRABAJO	19
5.1. Funcionamiento técnico de la web SOS ETSEIB	19
5.2. Base de datos SQL	20
5.2.1. Implementación de la base de datos	21
5.2.2. Estructura de la base de datos	21
5.3. Interacción con Dropbox	22
5.3.1. ¿Cómo funciona la API de Dropbox?	22

5.4. Envío de emails.....	23
5.5. Interfaz gráfica.....	23
5.5.1. Kivy	23
5.6. Funcionalidades del programa.....	24
5.6.1. Consulta de base de datos	24
5.6.2. Procesar datos de la web	27
5.6.3. Ver listas de emails	29
5.7. Estructura de archivos de la aplicación	31
5.7.1. Archivos secundarios	32
5.8. Estructura de la aplicación	32
6. COSTES Y PLANIFICACIÓN	33
6.1. Coste de personal.....	33
6.2. Coste de uso del ordenador	33
6.3. Coste total	34
6.4. Planificación	34
7. IMPACTO AMBIENTAL	35
CONCLUSIONES	37
AGRADECIMIENTOS	39
BIBLIOGRAFÍA	40
Referencias bibliográficas	40
Bibliografía complementaria.....	40

1. Glosario

Python: Lenguaje de programación.

Kivy: Librería que permite crear interfaces gráficas para programas hechos con Python.

Dropbox: Servicio de almacenamiento de archivos en la nube.

API: *Application Programming Interface*. Es el conjunto de subrutinas, funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software.

FTP: *File Transfer Protocol*. Es un protocolo de red para la transferencia de archivos.

Int: Tipo de variable utilizado en Python. Las variables de este tipo contienen números enteros.

String: Tipo de variable utilizado en Python. Las variables de este tipo contienen texto.

Tupla: Tipo de variable utilizado en Python. Las variables de este tipo contienen un conjunto de variables que pueden ser de distintos tipos, separadas por comas.

Popup: Ventana que emerge automáticamente debido a algún evento.

2. Prefacio

2.1. Origen del proyecto y motivación

SOS ETSEIB es un grupo de estudiantes que surge a partir de la idea de crear una plataforma que permita a los estudiantes de la ETSEIB compartir apuntes y material de estudio a través de internet. La idea nació en 2013 de las mentes de Mauricio Alva y Daniel Peral que en un primer momento crearon una web en forma de blog y unas carpetas de Dropbox que compartían con otros estudiantes a partir de solicitudes vía mail.

Más adelante, en 2014 me uní a esta asociación, aportando la actual web [\[1\]](#) que incorpora un formulario interactivo en el que los datos son registrados y almacenados en la propia web. Estos datos eran procesados de forma manual. En este trabajo se pretende obtener una automatización total de las tareas administrativas, aunque siempre será necesaria la supervisión de los administradores.

La idea de este trabajo surgió a mediados de 2015 ya que en esos momentos sabía que terminaría mis estudios en esta facultad y abandonaría el proyecto. Dado que yo era el único que sabía cómo gestionar la web y el proceso de administración, era necesario facilitar el proceso para que cualquier persona sin conocimientos avanzados de informática también pudiera hacerlo. Además era necesario documentarlo todo para que los estudiantes que siguiesen el proyecto pudiesen continuar trabajando en él. Fue Mauricio Alva quien me animó a contactar con el tutor, Lluís Solano y hacer que este proyecto fuese mi TFG.

Visto como ha funcionado este trabajo es posible que más miembros de SOS ETSEIB hagan su TFG sobre las distintas facetas de este proyecto, ya sea añadiendo nuevas funcionalidades al proyecto, mejorando las que ya se tienen haciéndolas más eficientes o arreglando partes del proyecto que dejen de funcionar ya que al estar basado en tecnologías web, estas pueden cambiar con el tiempo y dejar de funcionar algunas funcionalidades.

Entre otras, posibles continuaciones de este trabajo serían:

- Creación de una aplicación móvil para administrar la web
- Proyecto de expansión del proyecto SOS ETSEIB a otras universidades
- Buscar una alternativa a Dropbox (es una empresa privada que puede cambiar sus condiciones de uso en cualquier momento)

2.2. Requerimientos previos

Los requisitos impuestos a la realización de este trabajo son la necesidad de realizar el software en Python ya que es el lenguaje que se enseña en nuestra escuela y esto permite que todos los estudiantes de ella lo sepan utilizar, algo necesario para que en un futuro nuevos estudiantes puedan dar continuidad al proyecto y sepan cómo utilizar el programa de administración creado en este TFG.

Otro requerimiento es el de utilizar los datos en el formato en que son almacenados en la web (formato que será explicado más adelante), para esto se ha utilizado primero FTPLib para obtener los datos y después las propias herramientas de python para procesar estos datos.

3. Introducción

SOS ETSEIB es una web [1] administrada por estudiantes de nuestra escuela (ETSEIB), su principal objetivo es el de permitir a los estudiantes compartir apuntes de las asignaturas impartidas en nuestra escuela.

Esta compartición de apuntes se realiza a través de carpetas compartidas de Dropbox, [2] las cuales permiten almacenar una gran cantidad de archivos en internet a la vez que son privadas y solo pueden ser accedidas recibiendo una invitación. Toda persona que recibe una invitación a una carpeta, puede ver sus contenidos, modificarlos y añadir nuevos contenidos.

Desde el punto de vista del alumno el proceso de acceder a las carpetas de SOS ETSEIB es el siguiente:

1. Acceder al formulario de la web y rellenarlo. Al pulsar el botón enviar, la información escrita en el formulario queda almacenada en la web y el usuario ve un mensaje diciéndole que se ha recibido su solicitud con éxito i que en un lapso de tiempo recibirá las invitaciones a las carpetas. En caso de que algún dato introducido no sea válido el mensaje que aparece le informa de lo que ha fallado.
2. Una vez su solicitud ha sido procesada y aceptada por los administradores, el alumno recibe un email avisándole de esto y dándole información adicional sobre el proyecto. A partir de este momento este alumno ya dispone de acceso al material de las asignaturas que ha pedido.

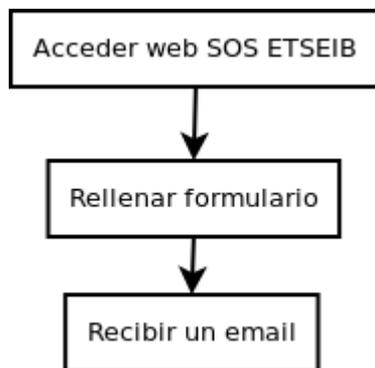


Figura 3.1: Diagrama del proceso seguido por un usuario de SOS ETSIB

Desde el punto de vista de los administradores, el proceso de administración de la web se puede resumir en unas pocas tareas que serán explicadas al detalle en apartados posteriores. Estas tareas se realizan en este orden:

- Obtener nuevos datos de la web
- Procesarlos
- Guardar una copia de los datos

- Enviar invitaciones a carpetas de Dropbox
- Enviar emails.

Desde el equipo administrativo de SOS ETSEIB se realizan más tareas como reuniones para mejorar el proyecto, adición de nuevas funcionalidades, tratamiento de casos excepcionales,... pero estas tareas se quedan fuera del alcance de este trabajo.

3.1. Objetivos del trabajo

En este trabajo se pretende crear un software que permita administrar, lo más fácil y rápidamente posible la web de SOS ETSEIB. El programa ha de encargarse de descargar los datos de la web, almacenarlos en una base de datos, enviar las correspondientes invitaciones a las carpetas de Dropbox y enviar emails. En definitiva lo que se busca es automatizar tanto el proceso como sea posible.

Todo esto debe realizarse aportando una interfaz gráfica que permita a los administradores supervisar el funcionamiento de todo el proceso, ya que es necesaria la autorización de todas las acciones por parte de los administradores.

También es necesario crear una buena documentación del programa que permita a los estudiantes que en un futuro se encargarán de administrar el proyecto, entender cómo funciona el programa y aprender a modificarlo para adaptarlo a nuevas necesidades.

3.2. Estado previo a este trabajo

En el momento de empezar este trabajo, el proceso administrativo funciona como es descrito detalladamente en el capítulo siguiente.

Para empezar el administrador descarga los datos recogidos por la web. Estos datos siguen un formato que muestra la información de un usuario distinto en cada línea. Leyendo este fichero línea a línea, se deben enviar las invitaciones a cada usuario a cada una de las carpetas que haya pedido. Este envío de invitaciones se realiza entrando a la web de Dropbox y utilizando las funcionalidades de compartición, se envían las invitaciones correspondientes. Por último enviar un email a cada usuario confirmándole que su solicitud ha sido procesada.

El problema de este proceso es que, al hacerse a mano, es muy lento. Es por eso que es de vital importancia automatizar tantas tareas como sea posible.

4. Descripción del problema

En este capítulo se hablará del proceso manual de administración de SOS ETSEIB. Se comentarán en detalle todas las peculiaridades de su funcionamiento y se describirán las necesidades que hay que cubrir para poder automatizar cada proceso.

4.1. Funcionamiento del proceso de administración

En este apartado se explicará en profundidad el proceso seguido por los administradores de SOS ETSEIB para gestionar la web, previamente a la creación del programa que surge de este TFG.

Para poder llevar a cabo el proceso de administración es necesario que nuevos haya nuevos datos de usuarios registrados en la web. En caso de que estos no existan, no tiene sentido llevar a cabo el proceso, por lo tanto supondremos que hay nuevos datos.

4.1.1. Descarga de datos

El primer paso para realizar la administración es obtener nuevos datos de la web. Esto se realiza conectándose al servidor FTP de la web y descargando el nuevo archivo de datos. Una vez descargado el archivo, este se elimina del FTP de la web.

Para realizar la conexión al FTP de la web usualmente se utiliza el programa *FileZilla*, aunque la conexión se puede realizar con cualquier cliente FTP. En la Figura 4.1 se puede ver la interfaz del programa FileZilla.

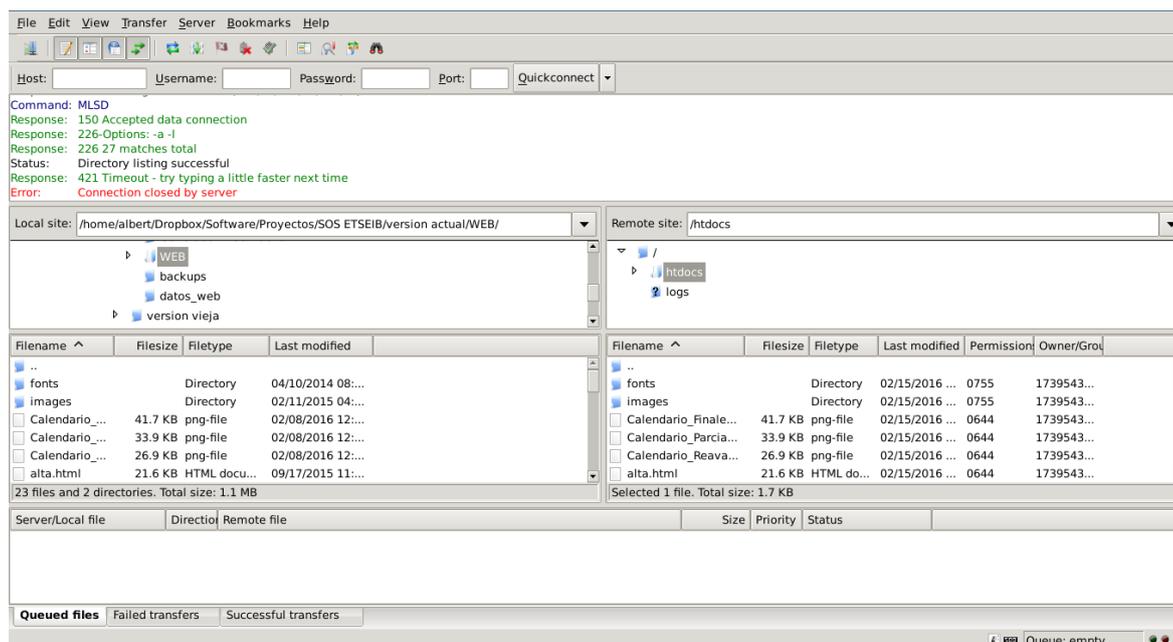


Figura 4.1: Muestra del programa FileZilla

4.1.2. Procesado de datos

Una vez descargado el archivo de datos, se abre con un editor de texto cualquiera. En el ejemplo mostrado en la Figura 4.2 se utiliza el editor Atom.

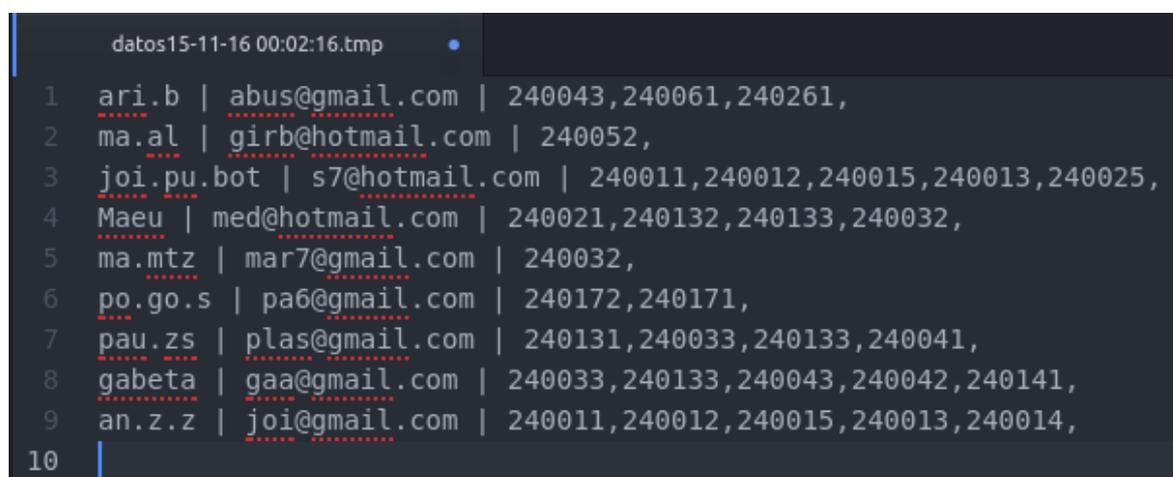


Figura 4.2: Archivo de datos abierto con Atom

Una vez abiertos estos datos, se realiza una inspección visual en busca de anomalías, como datos mal introducidos, datos duplicados, etc. En caso de encontrarse una de estas anomalías, se corrige el archivo de datos.

Cada línea del fichero corresponde a los datos del formulario enviado por una persona distinta. El siguiente paso es, línea a línea realizar las acciones necesarias para incluir a la persona en el proyecto SOS ETSEIB. Estas acciones son las descritas en los siguientes apartados.

4.1.3. Registro de la persona en un Excel

Se utiliza un archivo de Excel (Figura 4.3) para guardar todos los datos de los usuarios que se registran. Es decir, por cada línea del archivo de datos descargados, se añaden datos al archivo de Excel.

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	Código Usuario	Nombre	Correo	Acreditaciones	Primeros									
					Álgebra	Cálculo I	Cálculo II	Geometría	Mec. Fund.	Química I	Química II	Term. Fund.	Din. Sist.	Eco. Y Emp. I
1	0000	Universidad												
2	0001	SOS ETSEIB (Administrador general)	sos.etsieb@gmail.com	+5										
4	0002	SOS ETSEIB (Isa Caffeele)	sos.etsieb2@gmail.com	+5										
5	0001	SOS ETSEIB1 (Alejandro Camas)	sos.etsieb1@gmail.com	+5										
6	0007	SOS ETSEIB7 (Daniel Peral)	sos.etsieb7@gmail.com	+5										
7	0003	SOS ETSEIB3 (Mauricio Alva)	sos.etsieb3@gmail.com	+5										
8	0004													
9	0005													
10	0006													
11				# Para mirar si la gente participa. Poner +5 cuando va han aportado suficiente (para ver si se involucran suficiente la gente)										
12														
13														
14														
15	0100	Ricardo	ricardo@man.com	+5									1	1
16	0101	Isa	isa@hotmail.com	0										
17	0102	Guillermo	guillermo@hotmail.com	0										
18	0103	Arturo	arturo@hotmail.com	0										
19	0104	José	jose@hotmail.com	0										
20	0105	José	jose@hotmail.es	0									1	
21	0106	Sergio	sergio@hotmail.com	0										
22	0107	Ricardo	ricardo@hotmail.com	0										
23	0108	Silvia	silvia@telefonica.net	0										1
24	0109	Christian	christian@hotmail.com	0										
25	0110	Guillermo	guillermo@hotmail.com	0									1	1
26	0111	Luis	luis@man.com	0								1	1	
27	0112	Arturo	arturo@gmail.com	0			1	1						
28	0113	Arturo	arturo@man.com	0										
29	0114	Eduardo	eduardo@hotmail.com	0										1
30	0115	Ricardo	ricardo@hotmail.com	0										
31	0116	Sergio	sergio@estudianti.upc.edu	0									1	1
32	0117	Arturo	arturo92@gmail.com	0										
33	0118	José	jose@hotmail.com	0									1	
34	0119	Arturo	arturo@gmail.com	0										
35	0120	Ricardo	ricardo@gmail.com	0										1
36	0121	Arturo	arturo@gmail.com	0										
37	0122	Nicolas	nicolas@gmail.com	0			1	1					1	1
38	0123	Marta	marta@gmail.com	0										
39	0124	Hector	hector@gmail.com	0										
40	0125	Carlos	carlos@gmail.com	0									1	
41	0126	Marta	marta@gmail.com	0									1	1
42	0127	Marta	marta@gmail.com	0									1	1
43	0128	Arturo	arturo@gmail.com	1										1
44	0129	Arturo	arturo@gmail.com	0										
45	0130	Arturo	arturo@gmail.com	0									1	1

Figura 4.3: Archivo de Excel utilizado como base de datos

4.1.4. Envío de invitaciones a las carpetas de Dropbox

Para enviar invitaciones a las carpetas de Dropbox solicitadas por cada usuario, es necesario acceder a la web de Dropbox. En ella, accediendo a la cuenta de usuario del proyecto SOS ETSEIB, se tiene acceso a las funcionalidades de compartición de las carpetas.

Las carpetas están ordenadas por carreras y después por cuatrimestres, como se puede ver en la Figura 4.4.



The screenshot shows the Dropbox interface for a folder named 'ETSEIB INDUSTRIALES'. At the top, there are navigation icons for file operations and a search bar labeled 'Buscar'. Below this is a table listing folders Q1 through Q7. Each row includes a folder icon, the folder name, the last modification date (indicated by '--'), and the sharing status (also indicated by '--').

Nombre ▲	Última modificación	Compartido con
 Q1	--	--
 Q2	--	--
 Q3	--	--
 Q4	--	--
 Q5	--	--
 Q6	--	--
 Q7	--	--

Figura 4.4: Organización de las carpetas en Dropbox

Una vez encontrada la carpeta que se quiere compartir, se accede al menú de compartición de la carpeta (*Figura 4.5*) y se procede a enviar una invitación al usuario que la ha pedido.

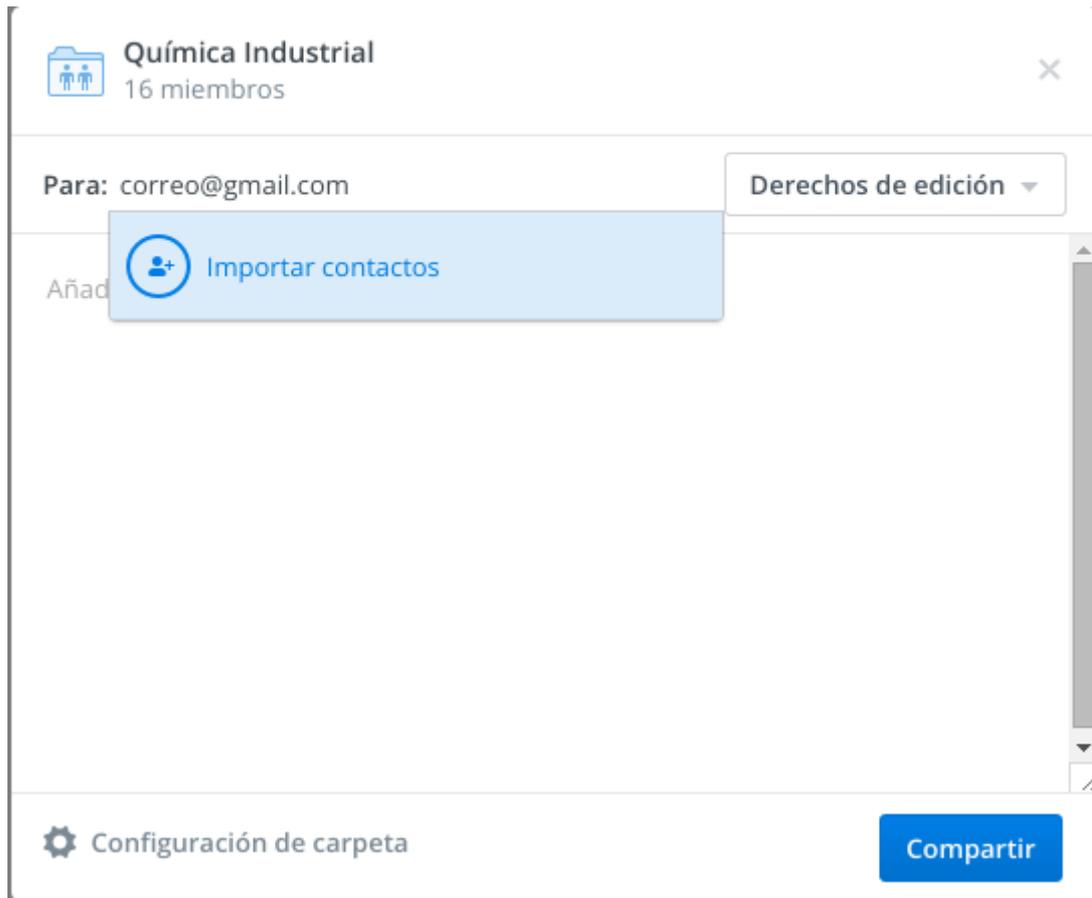


Figura 4.5: Menú de compartición de una carpeta en Dropbox

Este proceso es extremadamente lento ya que hay que hacerlo carpeta a carpeta. En un primer momento hay que buscar la carpeta, y dado que muchos alumnos piden asignaturas de varios cuatrimestres, es un proceso lento el tener que buscar cada carpeta. Además en el caso de que una carpeta tenga muchos usuarios (unos 100), abrir el menú de compartición de la carpeta puede tardar varios minutos.

4.1.5. Envío de un email

Para finalizar el proceso, se envía un email al usuario. El texto del email va cambiando según las necesidades del proyecto SOS ETSEIB, pero generalmente siempre se incluye información sobre el funcionamiento de las carpetas de Dropbox y lo que se espera que el usuario haga. También se utiliza el email para informar al usuario de su número de usuario y qué puede hacer con él. El email además sirve para avisar al usuario de que su solicitud ha sido recibida y procesada, y en caso de no haber recibido invitaciones a las carpetas pedidas, se ponga en contacto con los administradores.

4.2. Objetivos de automatización

En este apartado se explicará qué y cómo se automatizará del proceso de administración. Se seguirá la estructura del apartado anterior.

4.2.1. Descarga de datos

Desde el programa de Python será necesario descargar el archivo de datos del FTP de la web. Para la descarga de archivos de un FTP, la mejor opción trabajando con *Python* es *ftplib* que es una librería que viene incluida en el propio lenguaje *Python*, con lo que no es necesario descargarla aparte.

Automatizando este paso se ahorra el tener que abrir FileZilla, descargar y eliminar el archivo de datos. En conclusión, automatizando este paso se ahorra aproximadamente un minuto cada vez que se quiere realizar el proceso de administración.

4.2.2. Procesado de datos

En este apartado hay poco que automatizar ya que simplemente es una inspección visual. El programa se encargará de validar los datos y mostrará errores en caso de que el archivo no siga el formato deseado o haya algún dato mal introducido.

4.2.3. Base de datos

El proceso de registro en un Excel es muy lento. Se decide utilizar una base de datos para almacenar los datos de las nuevas inscripciones. El uso de una base de datos además aporta una mayor facilidad para trabajar con los datos y una muy buena integración con el lenguaje de programación.

Las bases de datos muchas veces funcionan sobre un servidor y se realizan las operaciones sobre esta en el mismo servidor. Dado que el proyecto SOS ETSEIB no puede permitirse un servidor con base de datos, la elección de base de datos queda limitada a una que pueda mantenerse en local.

Se ha encontrado el sistema de gestión de base de datos *SQLite* que almacena la base de datos en un simple archivo. Este sistema utiliza una base de datos de tipo SQL. Además una de las principales ventajas de este sistema es que viene integrado en *Python* por defecto, con lo que no será necesario instalar ninguna librería adicional para su uso.

4.2.4. Envío de invitaciones a carpetas de Dropbox

Para realizar el envío de invitaciones a las carpetas de Dropbox se utilizará la *API* del propio *Dropbox*. Esta *API*, entre muchos otros lenguajes, está disponible para *Python*, y cuenta con una muy buena documentación.

4.2.5. Envío de emails

Para realizar el envío de emails se utilizará la librería *smtplib* ya que viene incluida en *Python* sin la necesidad de instalar ninguna librería extra y permite el envío de mensajes desde una cuenta de correo Gmail que es la utilizada por el proyecto SOS ETSEIB.

4.3. Interfaz gráfica

Es necesaria la creación de una interfaz gráfica para administrar el programa para que así sea sencillo de utilizar. Esto es de primordial importancia ya que al ser este un proyecto administrado exclusivamente por estudiantes de nuestra escuela, los administradores cambian muy a menudo. Es por esto que es necesario que la curva de aprendizaje sea la menor posible para que el proyecto pueda seguir funcionando.

Ante toda la gama de librerías pensadas para crear interfaces gráficas, se escogió *Kivy*. Esta elección fue sugerida por el tutor ya que al ser una librería pensada para dar soporte a dispositivos móviles, permitiría en un futuro crear una aplicación para móvil.

4.4. Conclusión

El proceso de administración que se seguía previamente a la realización de este TFG es el que se ve en la Figura 4.6. Se trata de un proceso largo y laborioso que requiere de mucho tiempo.

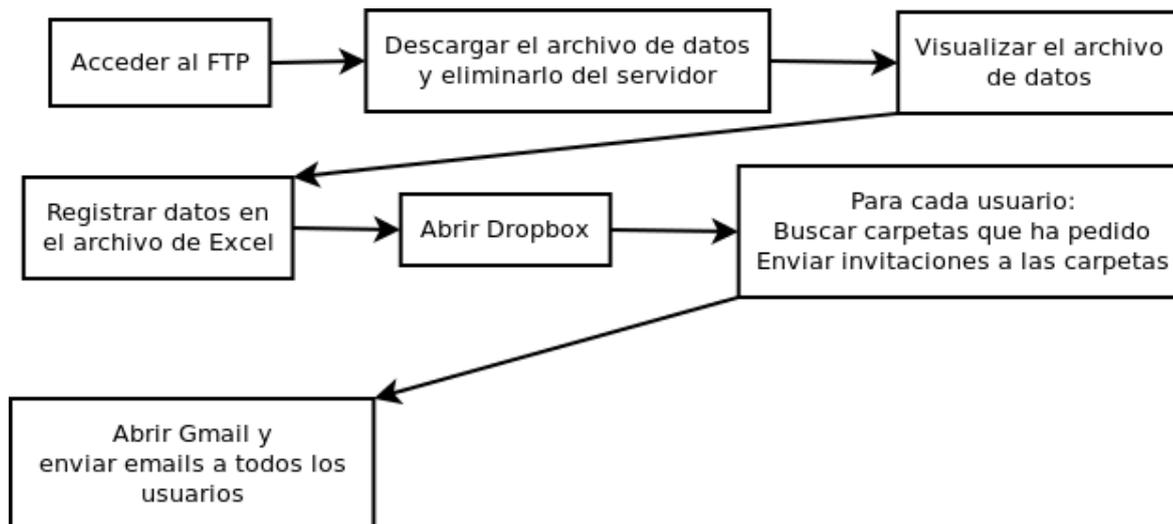


Figura 4.6: Diagrama del proceso de administración manual

Una vez realizado el TFG se pretende que el proceso de administración sea mucho más simple, como el de la Figura 4.7. Con esto se consigue ahorrar mucho tiempo al administrador.



Figura 4.7: Diagrama del proceso de administración automática

5. Desarrollo del trabajo

5.1. Funcionamiento técnico de la web SOS ETSEIB

En este trabajo tan solo vamos a centrarnos en las funcionalidades de almacenamiento de datos de usuarios de la web, por lo tanto no se hablará de la resta de funcionalidades disponibles en ella.

Los datos que se almacenan en la web provienen en su totalidad de un formulario que es rellenado por los usuarios. El formulario (Figura 5.1) recoge los datos del usuario que pide acceso a las asignaturas disponibles en SOS ETSEIB. Si estos datos son correctos, le muestra un mensaje al usuario diciéndole que todo está listo y solo debe esperar. También puede aparecer un mensaje de error en caso de que el usuario haya escrito un nombre de usuario o email invalido o de que haya pedido más de 9 asignaturas.

Código de usuario de **Atenea** (por ejemplo "nombre.apellido"):
Pon el **mismo usuario** que usas para entrar en Atenea. (Si usas los dos apellidos, pues los dos)

Email que usas para entrar al Dropbox:
Si hay cualquier problema nos pondremos en contacto contigo en ese mail.

Marca las asignaturas que quieras tener en tu Dropbox:
Cada cuatrimestre se actualizan las asignaturas. Selecciona sólo las que estás cursando actualmente o en las que quieras aportar.

Ingeniería Industrial:

Ingeniería Química:

Ingeniería de Materiales:

Q3

- Estructura y Caracterización de los Materiales
- Metalurgia Física
- Materiales Cerámicos
- Teoría de Máquinas Ing Materiales

Q4

- Materiales Plásticos

Q5

- Biomateriales

Q6

- Corrosión y Degradación de los Materiales
- Materiales Metálicos

Figura 5.1: Formulario de registro en la web de SOS ETSEIB

Los datos almacenados en la web tienen el formato siguiente:

```
Nombre | email | código_asignatura_1, código_asignatura_2, | comentario
```

Un ejemplo de archivo de datos es:

```
albert.s | alberts@hotmail.com | 250013,250011,250515,250012,250014,250016,  
juan.p | jp7@gmail.com | 240011,240012,240022,240021,240024,  
Pol.seta | PolS2@hotmail.com | 240025,
```

Este tipo de archivo, que antes era procesado manualmente, ahora se quiere procesar automáticamente. Para realizar esto se utilizara la función de python **split** para separar los distintos elementos de la lista (nombre, email, carpeta y comentario). Estos datos son añadidos a una base de datos de tipo SQL la cual se explica en el siguiente apartado. Hecho esto se procesa el envío de emails e invitaciones, cosa que será explicada en profundidad en apartados posteriores.

El archivo de datos será descargado empleando la librería de python **ftplib**.

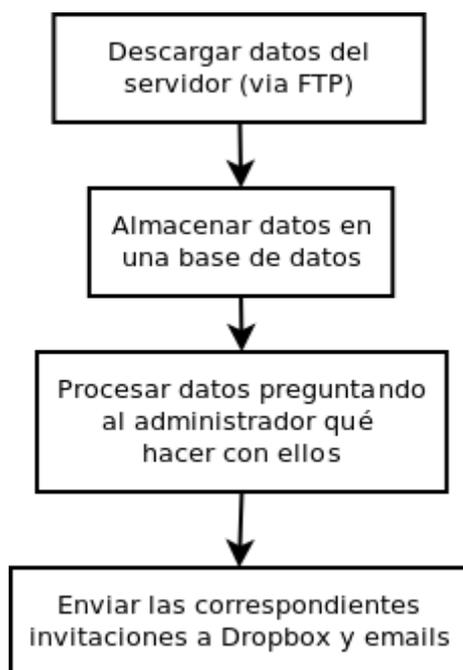


Figura 5.2: Diagrama del proceso seguido por el programa

5.2. Base de datos SQL

Para tener un registro de todas las solicitudes recibidas en la web, se ha creado una base de datos SQL, en ella se almacenan todos los usuarios y emails así como las asignaturas en las

que el usuario está dado de alta en el momento actual. Además en esta base de datos se genera un identificador único para cada usuario, este identificador es utilizado para que los usuarios suban apuntes propios e indiquen que son suyos con él.

5.2.1. Implementación de la base de datos

La base de datos se administra utilizando la librería **sqlite3** [3]. Esta librería tiene como principal ventaja que permite almacenar una base de datos SQL en un fichero local, es decir que no requiere un servidor para funcionar. Además dado que se trabaja con una base de datos muy pequeña (menos de 3000 usuarios actualmente) y no se espera un gran crecimiento en un futuro próximo, no es necesario utilizar un sistema de base de datos de alto rendimiento.

5.2.2. Estructura de la base de datos

La definición de la base de datos se puede encontrar en el archivo **funciones.py**, en la función **initDB**.

La base de datos emplea una única tabla llamada **alum**. En esta tabla se inscriben los alumnos, teniendo que rellenar 5 campos. Estos 5 campos son los que se pueden ver en la Tabla 5.1:

Campo	Tipo
Id	Numero entero único para cada usuario
code	Texto con el nombre de usuario
Comprobado	Valor binario, no se utiliza actualmente, se mantiene por razones de compatibilidad
Carpetas	Texto, lista de códigos de las asignaturas a las que está inscrito un usuario, separadas por comas
email	Texto, email del usuario

Tabla 5.1: Estructura de la base de datos

5.3. Interacción con Dropbox

El procedimiento de interacción con Dropbox puede realizarse de tres formas

- Manualmente accediendo a la web y utilizando su interfaz gráfica se pueden hacer todo tipo de gestiones. Este método es el ideal cuando lo que se quiere hacer es poco y algo puntual. En caso de tener que realizar una gran cantidad de acciones repetitivas, como es el caso de este trabajo, este es un método nada recomendable.
- Mediante la aplicación de escritorio oficial de Dropbox se puede sincronizar una carpeta del ordenador que almacene los mismos contenidos que la cuenta de Dropbox, si se modifica algo en esta carpeta, esos cambios son también hechos en Dropbox.

Se puede escribir un programa que interactúe con los archivos haciendo uso de esta característica y mantenerlos sincronizados. Para los objetivos de este trabajo esta funcionalidad no aporta nada.

- Se puede utilizar la API de Dropbox para crear una aplicación que haga las tareas necesarias, este será el camino seguido en este trabajo. El inconveniente de utilizar este método es que en cualquier momento y sin previo aviso esta API puede cambiar y en consecuencia la aplicación puede dejar de funcionar, esto ha ocurrido durante el desarrollo de este trabajo.

Una vez hecha la decisión de utilizar la API de Dropbox queda por ver cómo funciona esta, con qué lenguaje y dónde se implementa. Una posibilidad es hacerlo en la propia web de SOS ETSEIB, aunque esto queda descartado ya que dadas las condiciones económicas del proyecto no podemos permitirnos una infraestructura que nos de la confianza suficiente para estas tareas. En estos momentos se utiliza un hosting gratuito lo cual expone la web a posibles cambios de las condiciones de este, posibles pérdidas de datos en cualquier momento, etc. Por lo tanto conviene depender lo mínimo de este hosting.

Visto todo esto el proyecto será realizado en forma de aplicación de Python empleando la API de Dropbox y conexión a la web de SOS ETSEIB vía FTP.

5.3.1. ¿Cómo funciona la API de Dropbox?

Para empezar, podemos ver la documentación de esta API en [\[4\]](#)

Antes de empezar a trabajar hay que instalar la librería de Dropbox para python. Una vez

instalada la librería, debemos crear una *aplicación de Dropbox* accediendo a nuestra cuenta en la web. Este procedimiento se ha realizado en todas las cuentas de Dropbox de SOS ETSEIB. Con la aplicación creada se obtiene un *access token* (es como una contraseña) para cada cuenta.

Llegados a este punto ya tenemos todo lo necesario para iniciar la programación del software. En el programa hay que crear una conexión a Dropbox empleando el *access token* comentado previamente. Para ver todo lo que se puede realizar con la API es interesante leer la documentación completa en [\[5\]](#)

La función principal que queremos implementar es la de añadir a los alumnos a las correspondientes carpetas, para esto emplearemos las funciones del apartado *sharing* que permiten manejar los derechos de acceso a las carpetas a partir de los emails.

5.4. Envío de emails

Es necesario enviar emails a los usuarios recién registrados para que sepan que su solicitud ha sido procesada correctamente, y para informarles de algunas condiciones de uso y de su identificador de usuario.

Estos emails se quieren enviar desde la cuenta de correo principal del proyecto, sos.etsuib@gmail.com. Para realizar los envíos se emplea la librería de Python *smtplib*

5.5. Interfaz gráfica

Para que la aplicación sea fácil de utilizar se requiere implementar una interfaz gráfica intuitiva. Para la realización de la interfaz gráfica, a recomendación del tutor, Lluís Solano, se ha elegido utilizar *Kivy*.

5.5.1. Kivy

Kivy es una librería de código abierto para Python. Está destinada a la creación de aplicaciones con interfaz gráfica, con un enfoque principal en las aplicaciones para móviles. Es compatible con Android, iOS, Mac OS X, y Windows.

De cara a este proyecto, esta herramienta presenta como principales ventajas el hecho de ser multiplataforma incluyendo plataformas móviles, y facilidad de uso al separar la lógica del programa de la definición de la interfaz. Los inconvenientes de esta plataforma se han ido encontrando a medida que se realizaba este trabajo, entre otras cosas se ha encontrado a

faltar una buena documentación [6] y se han encontrado carencias en las posibilidades de esta interfaz ya que está muy enfocada a dispositivos móviles.

5.6. Funcionalidades del programa

El programa se ha diseñado de forma que se muestre un menú al abrirlo. En este menú (Figura 5.3) se encuentran disponibles todas las funcionalidades del programa y tan solo hay que elegir cual se quiere utilizar.



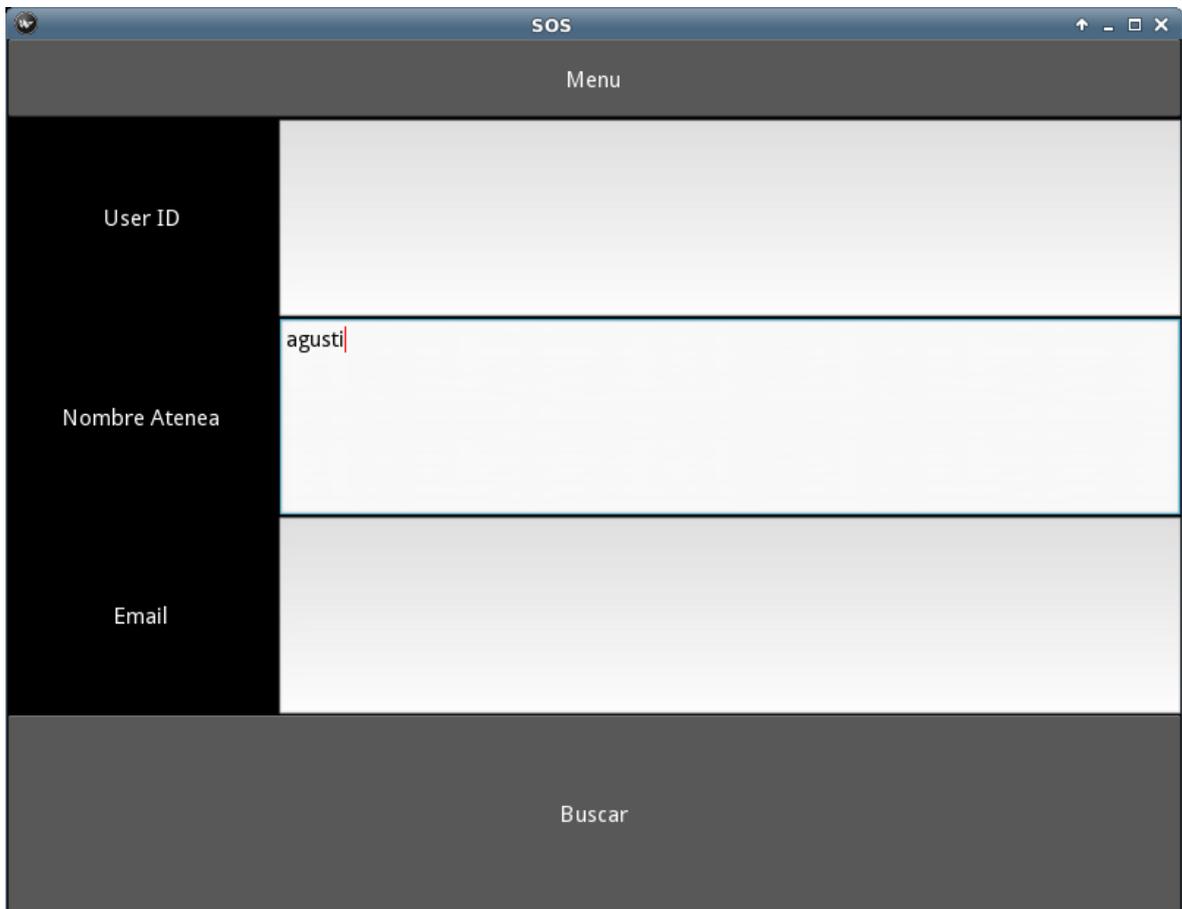
Figura 5.3: Menú principal de la aplicación

Estas funciones se dividen en tres grandes bloques que son los que se explican a continuación en los siguientes sub-apartados.

5.6.1. Consulta de base de datos

Para poder llevar un control de qué usuarios están registrados en qué asignatura es necesaria una forma fácil y rápida de buscar la información en la base de datos. Para

implementar esta funcionalidad se ha creado un buscador (Figura 5.4) que permite buscar entre los usuarios, ya sea por nombre, apellido, email o código de usuario.



The image shows a web application window titled "SOS". The window has a dark header bar with the text "Menu". Below the header, there is a dark sidebar on the left with three labels: "User ID", "Nombre Atenea", and "Email". The main content area is light gray and contains three input fields. The middle input field is active and contains the text "agusti". Below the input fields, there is a dark gray bar with the text "Buscar".

Figura 5.4: Formulario de consulta de un usuario

En el caso de la búsqueda por código de usuario, dado que este es único, se muestra el usuario correspondiente directamente. Si se busca por nombre, apellido o email, el programa muestra una lista de todos los usuarios que cumplen esa búsqueda, siendo necesario elegir cuál es el correcto (Figura 5.5).

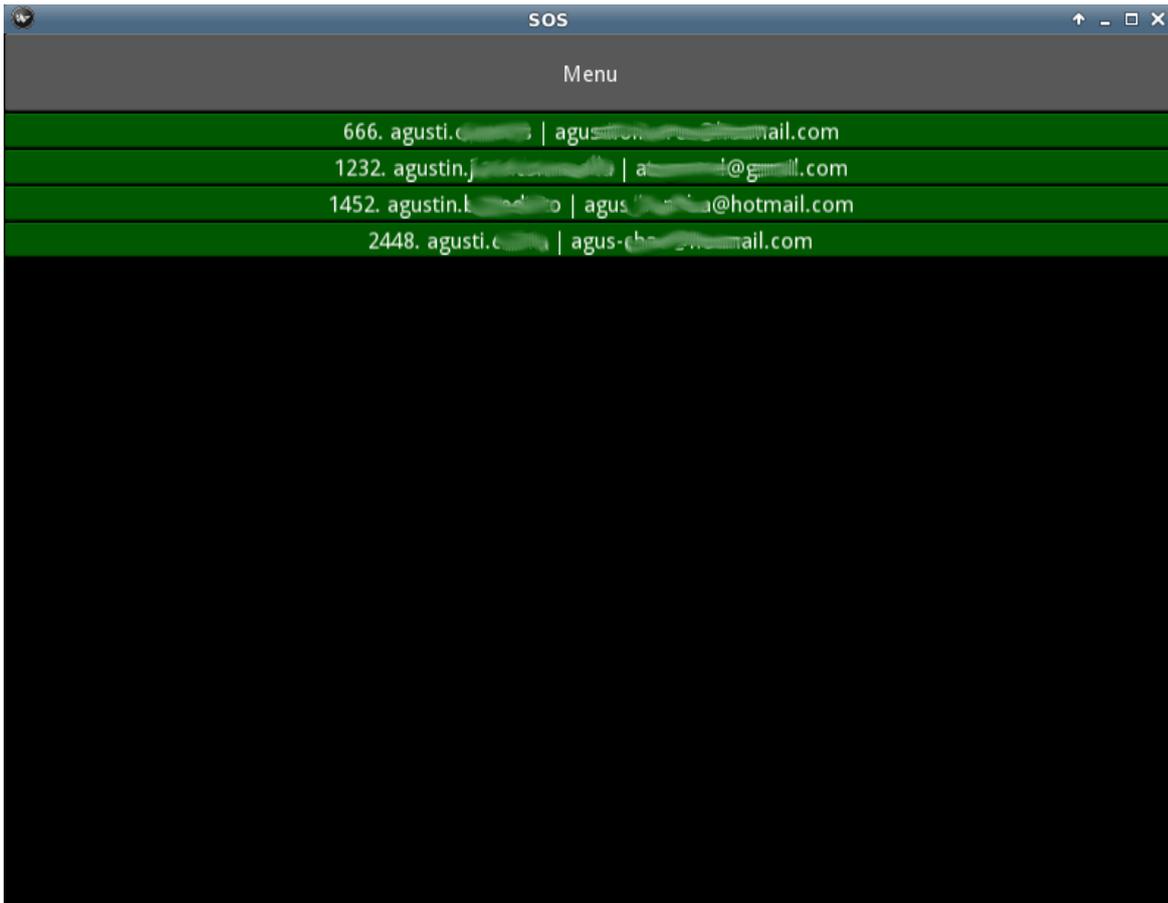
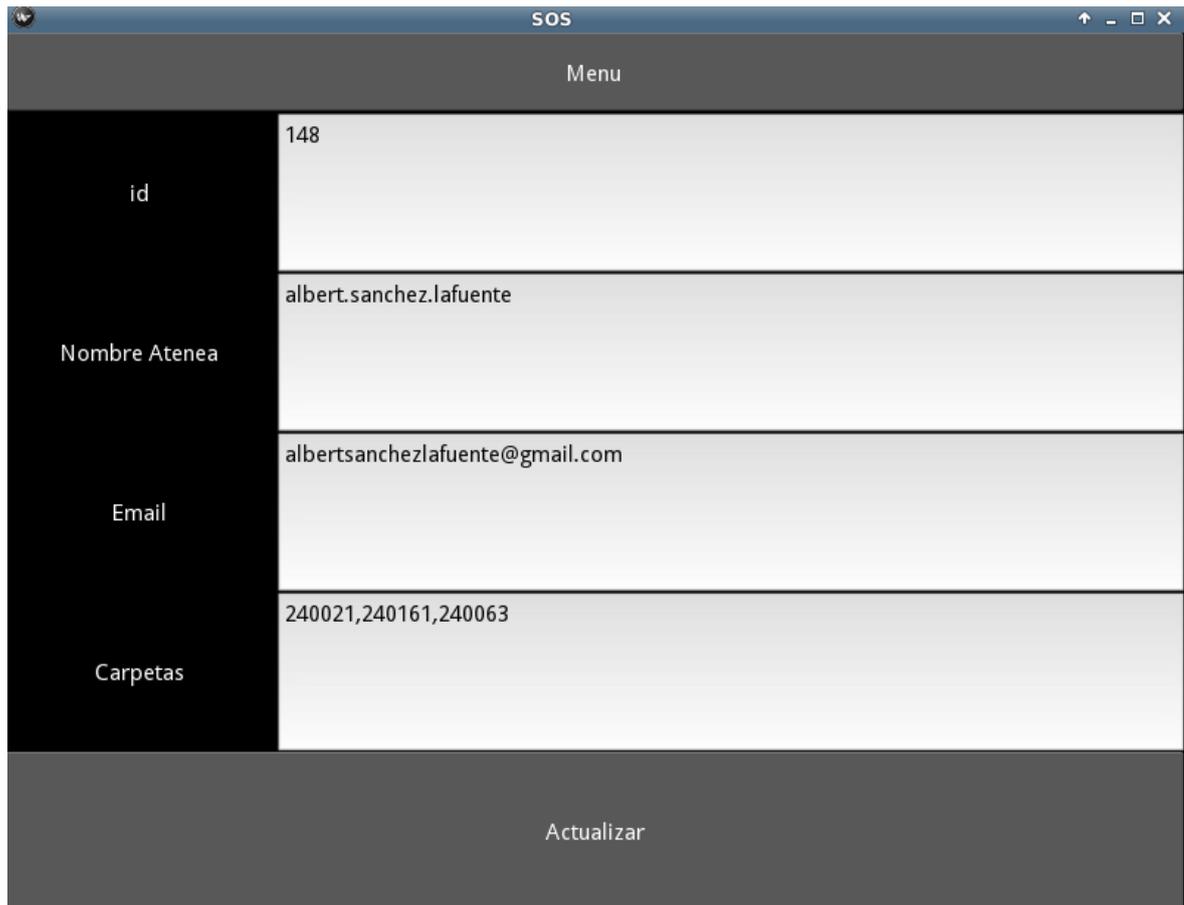


Figura 5.5: Cuando se encuentran varios usuarios, se muestra la lista para que el administrador elija

Una vez se ha accedido al usuario elegido se muestran sus datos en pantalla y se permite al administrador editar estos datos (Figura 5.6).



Menu	
id	148
Nombre Atenea	albert.sanchez.lafuente
Email	albertsanchezlafuente@gmail.com
Carpetas	240021,240161,240063
Actualizar	

Figura 5.6: Una vez seleccionado el usuario, se muestran sus datos

5.6.2. Procesar datos de la web

La función de procesamiento de datos de la web, empieza estableciendo la conexión a Dropbox, y descargando los nuevos datos de la web y uniéndolos a los antiguos. Una vez se ha hecho esto, el programa empieza a mostrar los datos de las solicitudes por pantalla (Figura 5.7). Solo se muestran los datos de una persona cada vez y se muestran en orden cronológico de relleno del formulario, es decir si fuiste el primero en rellenar el formulario web, serás el primero en ser procesado por el programa.



SOS

Menu

Nombre: prueba11

Mail: prueba@gmail.com

Carpetas: 240022,240025,240021,240024,240023

No aparece en la base de datos de la UPC

Agregar a Dropbox

No agregar a Dropbox

Figura 5.7: Ejemplo de procesamiento de datos

El programa también permite la edición de solicitudes, esto es interesante ya que por error, al rellenar el formulario, a veces se escribe algún dato mal, como por ejemplo escribir “gmal” en lugar de “gmail”. En el caso de los correos si el correo escrito no es uno de los más comunes (gmail, Hotmail, upc) se muestra resaltado en color naranja.

Los botones de la parte inferior varían según las condiciones del usuario. En el caso estándar se muestran los mismos botones que en la imagen superior. Si el usuario ya se había registrado previamente, se muestran los datos de este usuario que hay almacenados en la base de datos y los rellenados en el nuevo formulario, para que el administrador pueda compararlos. Como posibles acciones en este caso se muestran tres botones:

- **Sustituir datos:** Se envían invitaciones a las carpetas solicitadas y se sustituyen los datos de la base de datos por los del nuevo formulario. Esta opción se utiliza cuando el usuario ha cambiado el email.
- **Añadir nuevas asignaturas:** Se envían invitaciones a las carpetas solicitadas y se añaden a la base de datos las nuevas asignaturas que ha pedido el usuario. Esta

opción se utiliza cuando el usuario ha pedido nuevas asignaturas en las que no estaba registrado previamente, utilizando esta opción se consigue mantener la información que se tenía previamente y añadir la nueva.

- **Sólo enviar carpetas:** Solo se envían invitaciones a las carpetas solicitadas. Esta opción se utiliza cuando no se quiere escribir nueva información en la base de datos, generalmente se utiliza cuando el usuario ha enviado exactamente los mismos datos que la vez anterior.

En este caso de que el usuario ya estuviese registrado, se tome la acción que se tome, se pregunta al administrador si quiere enviar un email de aviso al usuario.

La lógica utilizada para decidir qué botones mostrar se puede ver en la Figura 5.8

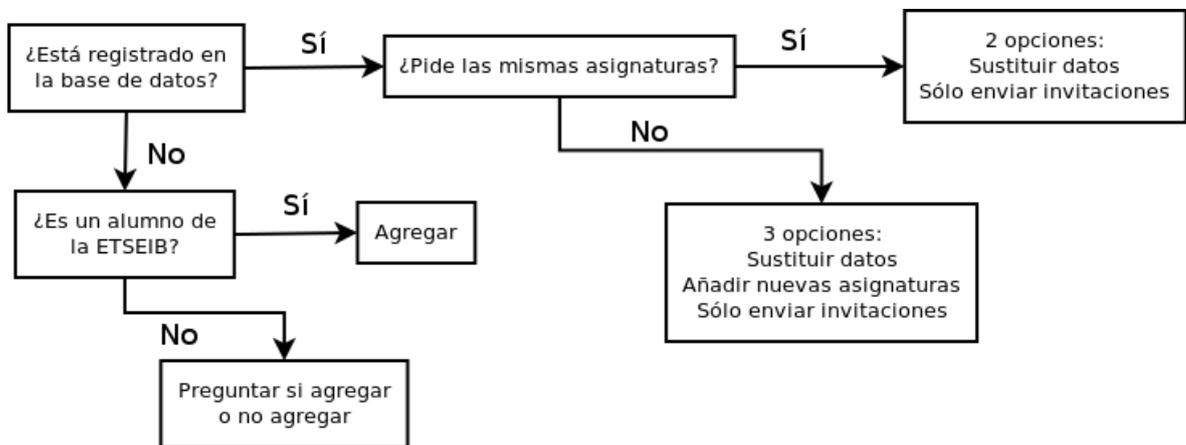


Figura 5.8: Diagrama representando la lógica utilizada para mostrar botones

5.6.3. Ver listas de emails

Para hacer difusión de información relativa al proyecto entre los usuarios, se utiliza el correo electrónico. Generalmente se realizan los envíos a todos los usuarios activos del proyecto. Por lo tanto es necesario poder obtener una lista actualizada con los emails de todos los usuarios.

Para esta tarea se han creado tres opciones en el menú del programa. Dos de ellas simplemente generan una lista de emails separados por comas y la copian al portapapeles, hecho esto simplemente se debe pegar esta información en el correspondiente cliente de correo y se puede hacer el envío masivo de un email. La diferencia entre estas dos opciones es que una genera una lista con todos los usuarios desde el inicio del proyecto, mientras que la otra solo incluye los usuarios activos en el cuadrimestre actual.

La tercera opción genera una lista de direcciones de correo separadas en bloques de 499 direcciones (Figura 5.9). Esto es útil ya que el servidor de correo Gmail que es el que se utiliza en el proyecto, tan solo permite enviar a 499 direcciones un mismo correo. Así se obtienen distintos bloques listos para copiar y enviar por separado. Además en el apartado de configuración (Figura 5.10) se permite cambiar este número ya que es posible que en un futuro este número sea cambiado por parte de Google, empresa propietaria del servicio Gmail.

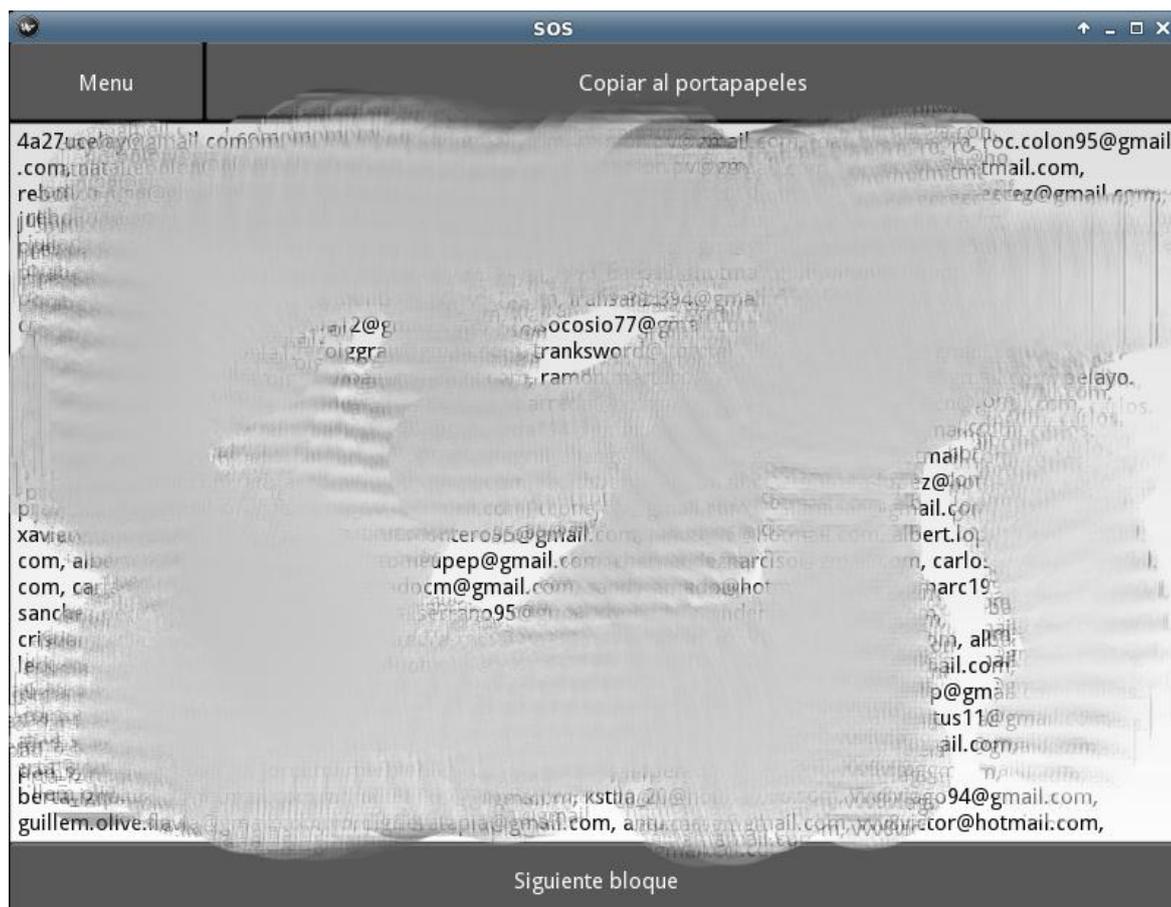


Figura 5.9: Visualizador de emails en bloques



Figura 5.10: Configuración del programa

5.7. Estructura de archivos de la aplicación

La aplicación se ha creado de tal forma que se separe la interfaz gráfica de las propias funcionalidades del programa con el objetivo de que si en un futuro es necesario se pueda cambiar la interfaz gráfica sin tener que modificar la resta de código, es decir manteniendo las mismas funciones para establecer las conexiones a Dropbox, a la web, etc.

La estructura de ficheros empleada es la siguiente:

- **funciones.py:** En este fichero están escritas todas las funciones lógicas del programa, es decir las que no se encargan de la interfaz gráfica sino de ejecutar tareas.
- **interfaz.py:** En este fichero se encuentran todas las funciones que se encargan de generar la interfaz gráfica del programa y unirla con las funciones del fichero **funciones.py**.

- **main.py:** Este fichero se encarga de lanzar la aplicación.
- **sos.kv:** En este fichero se define la interfaz gráfica, es un fichero de estilo del lenguaje Kivy.

5.7.1. Archivos secundarios

Para el correcto funcionamiento de la aplicación son necesarios los siguientes ficheros:

- **alum:** Fichero de texto con un listado de los correos oficiales de la UPC de los alumnos de la ETSEIB. Cada correo ocupa una línea del fichero.
- **correo.txt:** Fichero de texto cuyo contenido es el cuerpo del mensaje que se envía por correo electrónico a los usuarios tras rellenar el formulario.
- **datos.tmp:** Archivo temporal, puede estar vacío. En él se almacenan los datos descargados de la web. Los datos que no son procesados se mantienen en este fichero a la espera de ser procesados.
- **sos.sql:** Base de datos SQL en la que se almacenan los datos de los usuarios. Sirve tanto para leer datos antiguos como para escribir nuevos datos.

5.8. Estructura de la aplicación

En este apartado se pretende describir la estructura del funcionamiento de la aplicación, en orden cronológico.

Para empezar se abre el archivo **main.py** el cual carga el archivo **interfaz.py** y lanza la clase *SOSApp*. En ese mismo fichero también se encuentran definidas todas las ventanas de la aplicación. El lanzamiento de *SOSApp* carga la clase *SOSRoot* y la muestra por pantalla. La clase *SOSRoot* a su vez, lo que hace es mostrar en pantalla la clase *Menu* que es el menú principal que se ve una vez abierta la aplicación.

Una vez en el menú principal, entrando a cualquiera de las opciones presentes en él, se carga la clase correspondiente a esa opción y se muestra su ventana en pantalla.

Dentro de estas ventanas, mediante los botones se puede volver al menú o acceder a otras ventanas que son representadas por clases en el código. Todas estas opciones están explicadas en profundidad en el **Anexo II**.

6. Costes y planificación

Dado que se trata de un proyecto informático sus costes han sido muy reducidos, sin haberse producido la necesidad de gastar dinero directamente para realizar el proyecto propiamente dicho.

Los únicos costes que se pueden contabilizar son el tiempo de trabajo y el gasto producido debido al uso del ordenador portátil desde el que se ha hecho este trabajo.

6.1. Coste de personal

El trabajo de fin de grado tiene una carga lectiva de 12 créditos. Según la normativa de la UPC un crédito equivale a una carga de trabajo de 25 horas. Con esto se obtiene una carga de trabajo de 300 horas.

Estas 300 horas se dividen entre tareas de gestión del proyecto, llevadas a cabo por un ingeniero sénior y tareas de desarrollo del proyecto llevadas a cabo por un ingeniero junior. En total han sido necesarias 30 horas dedicadas a la gestión y las restantes 270 al desarrollo de la aplicación.

Asumiendo el salario de un ingeniero que puede ser de unos 30 euros por hora para el caso del ingeniero junior y de 45 euros por hora para el ingeniero sénior, obtenemos una inversión en tiempo valorada en 9.450 euros.

6.2. Coste de uso del ordenador

Podemos considerar dos tipos de costes relacionados al uso del ordenador, el gasto en electricidad y el gasto de amortización del ordenador.

El gasto eléctrico es muy difícil de calcular ya que al tratarse de un ordenador portátil, he realizado cargas de la batería en el tren o en la universidad, con lo cual no se puede saber con exactitud el gasto exacto. Además no se han podido encontrar datos sobre el consumo eléctrico del modelo de mi ordenador portátil.

Se aproxima el gasto energético del ordenador portátil a 0,365 kW. Multiplicando este gasto por las 300 horas de trabajo se obtiene un gasto de 109,5 kWh. El valor aproximado del kWh de energía eléctrica es de 0,1275 euros por kWh. Con lo cual el gasto total aproximado es de 14 euros.

En cuanto a los gastos de amortización, el ordenador costó en su día, hace 4 años, 400 euros. Suponiendo que el ordenador siga funcionando durante 2 años más, su vida útil habrá sido de 6 años, 72 meses.

El portátil ha sido utilizado para este fin durante 6 meses, por lo tanto el coste de amortización en este periodo es de 33,33 euros.

6.3. Coste total

Visto todo esto podemos aproximar el coste total de este trabajo en unos 9.500 euros, siendo el personal encargado de realizarlo la parte más cara del proyecto.

6.4. Planificación

La realización del proyecto se ha planificado mediante un diagrama de Gantt (Figura 6.1) en el que se han definido las tareas a realizar así como los plazos temporales para realizarlas.

	Enero	Febrero	Marzo	Abril	Mayo	Junio
Definición del trabajo	■					
Recopilación de información		■				
Lectura de información		■	■			
Programación de la aplicación		■	■	■	■	
Redacción de la memoria					■	■

Figura 6.1: Diagrama de Gantt con la planificación del proyecto

El tiempo de trabajo ha estado distribuido uniformemente durante los últimos 5 meses, siendo en el primero empleadas tan solo unas pocas horas para hacer reuniones con el tutor y buscar información.

En total el primer mes se invirtieron 10 horas de trabajo con lo que sobraron 290 en los siguientes meses. Dividiendo el trabajo uniformemente cada semana, se obtienen 14,5 horas de trabajo semanales, o 58 horas mensuales.

7. Impacto ambiental

Dado que se trata de un trabajo informático el único impacto ambiental que se ha producido es el de la electricidad consumida.

En cuanto al uso del ordenador, el gasto en electricidad es responsable de unas emisiones de CO₂. Se estima una producción de CO₂ de 302 gramos por cada kWh de electricidad consumido [7]. Dado el consumo de 109,5 kWh gastados por el portátil, se produce una emisión de 33,07 kg de CO₂

Por otro lado el uso del programa permitirá ahorrar mucho tiempo de uso de ordenadores en el futuro, así que en un futuro se podrá considerar que la realización de este trabajo ha sido beneficiosa para el medio ambiente.

Conclusiones

Con este trabajo se ha conseguido implementar una aplicación para facilitar las tareas de administración de la web SOS ETSEIB. Se han cumplido los objetivos definidos al principio del trabajo, la aplicación es intuitiva y realiza las tareas de una forma mucho más rápida que manualmente.

Para poner números a este proyecto, anteriormente cuando se administraba sin asistencia se requería un tiempo del administrador de unos 5 minutos por usuario. Ahora con el programa, tan solo son necesarios unos 10 segundos por usuario.

También se ha podido ver que Kivy no es la herramienta más adecuada para elaborar una aplicación de escritorio. Las carencias que presenta tanto en documentación como en funcionalidades hacen que a día de hoy, ésta no sea una herramienta recomendable. Sin embargo Kivy puede ser una buena herramienta si el objetivo principal del proyecto es el de crear una aplicación para móvil.

En un futuro se puede plantear como continuación de este proyecto la adaptación de esta aplicación a dispositivos móviles, cosa que no se ha hecho en este proyecto debido a la falta de tiempo.

Otras continuaciones de este proyecto podrían ser la ampliación de las funcionalidades de la aplicación según las necesidades que presente la web de SOS ETSEIB. También será necesario en un futuro ir actualizando la aplicación ya que es muy probable que el funcionamiento de los servicios de terceros que se utilizan (Dropbox, Gmail) cambien sus APIs, o sus condiciones de uso.

Agradecimientos

Me gustaría dar las gracias a todo el equipo que hace posible que SOS ETSEIB funcione, se trata de una gran labor y sin ellos este trabajo no sería posible. Me gustaría mencionar en especial a Mauricio Alva ya que fue él quien me dio el empujón necesario para elegir hacer este trabajo.

También es necesario dar las gracias al tutor Lluís Solano por toda la experiencia y consejos que ha aportado a este trabajo.

Bibliografía

Referencias bibliográficas

- [1] SOS ETSEIB *Web SOS ETSEIB* [<http://sosetseib.tk>]
- [2] DROPBOX *Web Dropbox* [<https://www.dropbox.com/>]
- [3] SQLITE *SQLite Homepage* [<https://www.sqlite.org/>, consultada febrero 2016]
- [4] DROPBOX *Dropbox for Python developers* [<https://www.dropbox.com/developers/documentation/python>, consultada febrero 2016]
- [5] DROPBOX *Dropbox for Python Documentation* [<http://dropbox-sdk-python.readthedocs.io/en/master/>, consultada febrero 2016]
- [6] KIVY *Kivy Documentation*. Release 1.9.2-dev0
- [7] GENERALITAT DE CATALUNYA *Nota informativa sobre la metodología de estimación del mix eléctrico por parte de la oficina catalana del cambio climático (occc)* [http://canviclimatic.gencat.cat/web/.content/home/reduex_emissions/factors_emissio_a_sociats_energia/160219_Nota-metodologica-mix_cast.pdf, 7 junio 2016]

Bibliografía complementaria

Para la creación de la aplicación se han utilizado como referencia las fuentes listadas en este apartado.

PHILIPS, DUSTY *Creating Apps in Kivy*. O'Reilly, 2014

VASILKOV, MARK *Kivy Blueprints*. Packt Publishing, 2015

ULLOA, ROBERTO *Kivy: Interactive applications in Python*. Packt Publishing, 2013

KIVY *Kivy documentation and API Reference*. [<https://kivy.org/docs/>, consultada febrero 2016]

PYTHONPROGRAMMING.NET *Kivy with Python tutorial for Mobile Application Development* [<https://pythonprogramming.net/kivy-application-development-tutorial/>, consultada febrero



2016]

DOWNEY, ALLEN; ELKNER, JEFFREY; MEYERS, CHRIS *Aprenda a pensar como un programador*. Green Tea Press, 2002