

Trabajo Practico 2: DriveHub

Algoritmos y Programación I

29 de junio de 2021

1. Introducción

La cátedra de Diego Acosta de la asignatura de Algoritmos y Programación I ha solicitado la creación de un nuevo sistema para organizar la corrección de evaluaciones de la asignatura frente a la gran cantidad de quejas presentadas por sus estudiantes con respecto a la poca organización de la misma. En esta primera presentación se ha pedido la elaboración de un prototipo de sistema al estilo [GitHub](#) con ciertas funcionalidades y criterios que deberemos respetar.

2. Requisitos fundamentales

- El sistema deberá estar desarrollado en el lenguaje [Python](#) entre las versiones 3.6 a 3.9.
- Se deberá respetar las buenas practicas de programación.
- Se deberá desarrollar en equipo con la participación equitativa de cada miembro.
- Se deberá realizar un sistema de versionado en la plataforma [GitHub](#).
- Se deberá cumplir con los requerimientos solicitados en las siguientes secciones.
- Se deberá utilizar la API denominada [Drive API](#) desarrollada por la empresa [Google](#).
- Se deberá utilizar la API denominada [Gmail API](#) desarrollada por la empresa [Google](#).

3. Requerimientos

3.1. Menú principal

Se deberá crear un menú por consola de comandos el cual servirá como interfaz para que el usuario pueda interaccionar con el sistema. El mismo deberá ser amigable al usuario, claro en cada una de sus opciones, permitir validar los ingresos del usuario y dar una opción de salida del programa.

3.1.1. Opciones del menú

1. *Listar archivos de la carpeta actual.*

Deberá listar todos los archivos que se encuentren en la carpeta actual y en las subcarpetas de la misma tanto a nivel local como remoto.

2. *Crear un archivo.*

Deberá permitir crear un archivo o carpeta en la carpeta actual tanto a nivel local como remoto.

3. *Subir un archivo.*

Deberá subir un archivo que se encuentren en la carpeta local a la misma carpeta en remoto o a otra ubicación especificada por el usuario.

4. *Descargar un archivo.*

Deberá descargar un archivo que se encuentren en la carpeta remota a la misma carpeta en local o a otra ubicación especificada por el usuario.

5. *Sincronizar.*

Deberá sincronizar las carpetas y archivos conservando los últimos que hayan sufrido modificaciones.

6. *Generar carpetas de una evaluación.*

Deberá recibir vía mail la información necesaria para crear las carpetas tanto en local como en remoto.

7. *Actualizar entregas de alumnos vía mail.*

Deberá subir los archivos que haya recibido via mail a la carpeta correspondiente del alumno.

8. *Salir.*

Deberá salir del programa.

3.2. Listado de archivos

Se deberá permitirle al usuario ver por pantalla un listado de todos los archivos que se encuentren disponibles.

3.2.1. En local

Deberá mostrar los archivos que se encuentren en la carpeta principal del usuario así como las subcarpetas permitiendo navegar entre ellas.

3.2.2. En remoto

Deberá mostrar los archivos así como las subcarpetas que se encuentren en el [Drive](#) del usuario permitiendo navegar entre ellas.

Documentación sobre búsqueda de archivos disponible en este [link](#).

Documentación sobre búsqueda por campos especificaos de archivos disponible en este [link](#).

3.3. Creación de archivos

Se deberá crear una funcionalidad que permita crear un nuevo archivo entre una lista de posibles extensiones. La creación del archivo debe reflejarse tanto en local como en remoto.

3.4. Subir archivos

Deberá permitir al usuario subir un archivo desde su equipo local a su [Drive](#).

Documentación sobre subir archivos disponible en este [link](#).

3.5. Descargar archivos

Deberá permitir al usuario descargar un archivo o carpeta desde su [Drive](#) a su equipo local.

Documentación sobre descargar archivos disponible en este [link](#).

3.6. Sincronización

Deberá permitir al usuario actualizar los archivos tanto locales como remotos a las versiones que sufrieron la ultima modificación. Es decir, si un archivo está presente en local y en remoto y tiene fechas de modificación diferentes, el más antiguo debe modificarse por el más actual.

3.7. Sistema de carpetas

Esta funcionalidad deberá permitir crear un sistema de carpetas con 3 niveles de anidación. El primero sera de la evaluación, el segundo de los docentes y el tercero de los alumnos. Con lo cual cada evaluación sera una carpeta que contenga diferentes carpetas de cada docente que a su vez contendrán carpetas con el nombre de los alumnos que deben corregir. El nombre de la carpeta de evaluación vendrá dado en un asunto del mail que se recibirá. Luego, los nombres de los docentes, alumnos y relación alumnos - docente vendrán dadas en un archivo comprimido el cual contendrá 3 archivos de '.csv', uno para cada especificación. Si existieran docentes sin alumnos asignados o alumnos sin docentes asignados estos igual deberán tener su carpeta correspondiente.

3.7.1. docentes.csv

El archivo de docentes contendrá el siguiente formato de información:
Formato: nombre del docente, mail del docente.

3.7.2. alumnos.csv

El archivo de alumnos contendrá el siguiente formato de información:
Formato: nombre del alumno, padrón, mail

3.7.3. docente-alumnos.csv

El archivo de docente-alumnos contendrá el siguiente formato de información:
Formato: nombre del docente, nombre del alumno
Documentación sobre crear carpetas disponible en este [link](#).

3.8. Asignación de archivos

Esta funcionalidad permitirá cargar en la carpeta de cada alumno el/los archivos que este envíe en un mail con el siguiente formato: Asunto del mail: padrón Archivo adjunto: archivo comprimido
Documentación sobre crear archivos disponible en este [link](#).

3.9. Recepción de entregas

Esta funcionalidad permitirá enviar un mail al alumno que realizo una entrega informando si la misma fue exitosa o si presentó algún tipo de inconveniente (padrón incorrecto, nombre no coincidente con padrón, etc).

4. Archivos necesarios

Se brindaran dos archivos que configuran la conexión con las dos API's necesarias.
Archivo para API Drive en este [link](#).
Archivo para API Gmail en este [link](#).

5. Diagrama de flujo posible

Aquí se enseñara un posible diagrama de flujo para organizar el comienzo del TP2. Este esquema no es obligatorio y cada grupo podrá este o cualquier otro esquema que considere oportuno.

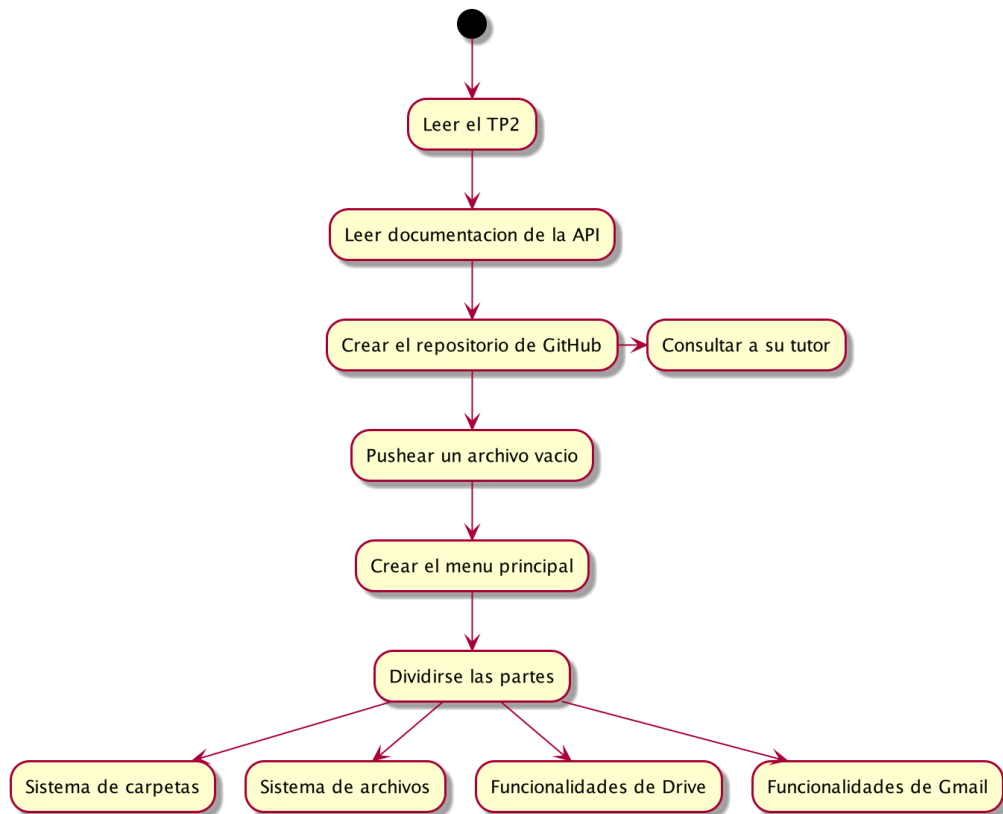


Figura 1: Esquema central.

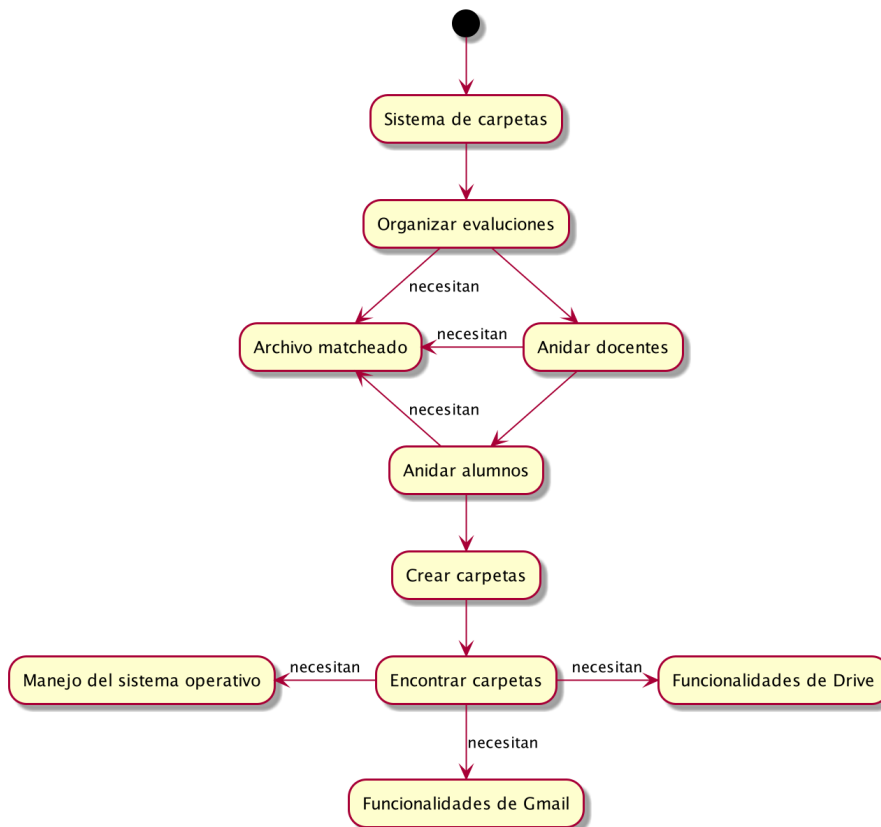


Figura 2: Esquema del sistema de carpetas.

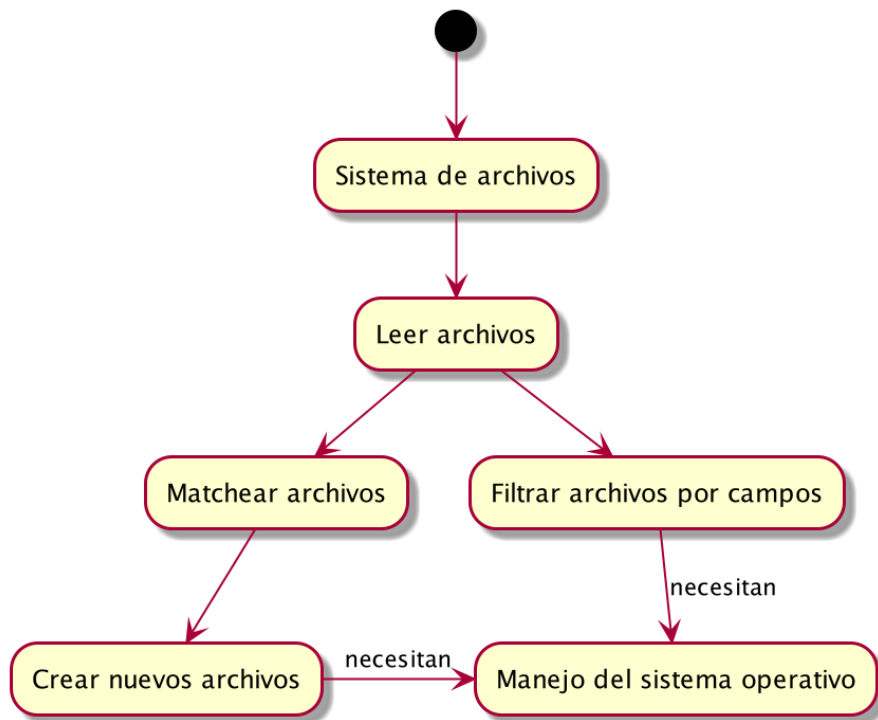


Figura 3: Esquema del sistema de archivos.

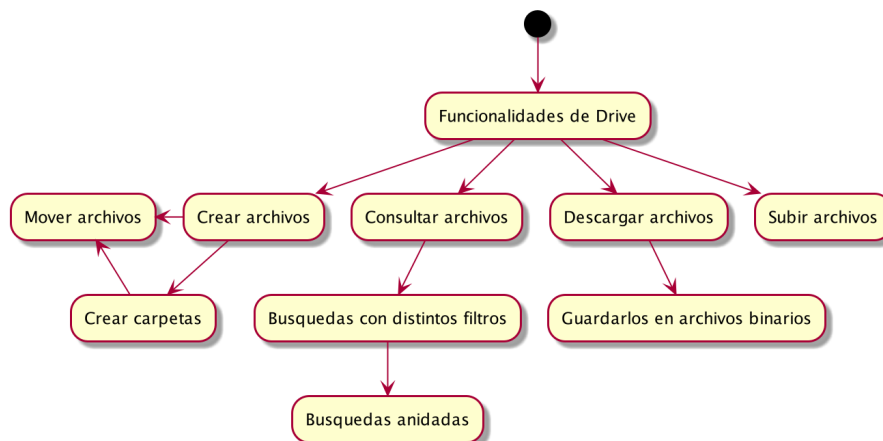


Figura 4: Esquema de funcionalidades de Drive.

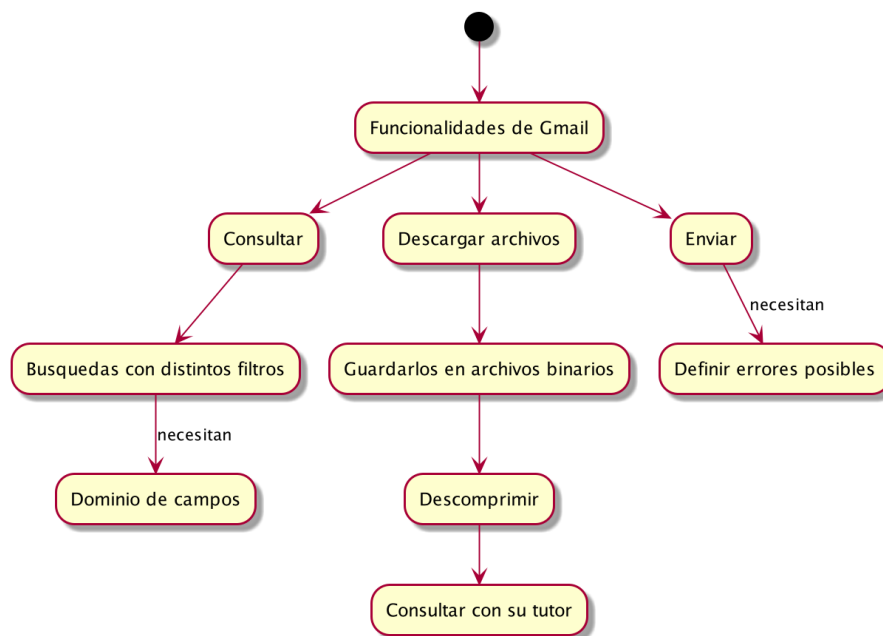


Figura 5: Esquema de funcionalidades de Gmail.