**Montaje automático de videos**

**(Moovid)**

<Logo del Proyecto>

Arquitectura e Integración de Sistemas Software

Grado de Ingeniería del Software

Curso 2º IS

Daniel Caro Olmedo ([danielcaroolmedo2@gmail.com](mailto:danielcaroolmedo2@gmail.com))

Nicolás De Ory Carmona ([deorynicolas@gmail.com](mailto:deorynicolas@gmail.com))

Antonio González Gómez ([antoniogg696@gmail.com](mailto:antoniogg696@gmail.com))

Ignacio Navarro Blázquez ([inavarroblazquez@gmail.com](mailto:inavarroblazquez@gmail.com))

Tutor: Javier Troya Castilla

Número de grupo:

Enlace de la aplicación: <Enlace de la aplicación en AppEngine>

Enlace de proyecto en GitHub: <https://github.com/nicolasdeory/moovid>

Historial de versiones

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión | Detalles | Participantes |
| 07/03/2020 | 1.0 | - Incluye introducción, prototipos de las interfaces de usuario y diagramas UML de componentes y despliegue y secuencia de alto nivel. | Daniel Caro  Nicolás De Ory  Antonio González  Ignacio Navarro |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Índice

[1 Introducción 3](#_Toc34908517)

[1.1 Aplicaciones integradas 3](#_Toc34908518)

[2 Prototipos de interfaz de usuario 4](#_Toc34908519)

[2.1 Vista x 4](#_Toc34908520)

[2.2 Vista y 4](#_Toc34908521)

[3 Arquitectura 4](#_Toc34908522)

[3.1 Diagrama de componentes 4](#_Toc34908523)

[3.2 Diagrama de despliegue 4](#_Toc34908524)

[3.3 Diagrama de secuencia de alto nivel 4](#_Toc34908525)

# Introducción

Moovid busca integrar Google Photos junto con Google Cognitive Services y Language Understanding Intelligent Service (LUIS) para crear automáticamente fotomontajes personalizados. De manera fácil y natural es posible informar que restricciones queremos en nuestro montaje, desde la fecha hasta el lugar, pasando por elementos dentro de la propia imagen, y todo mediante sencillas frases interpretadas por LUIS.

(Presentar y motivar el problema que se va a resolver con la aplicación. ¿Por qué es un problema importante? Describir el mashup desarrollado y como éste le da solución al problema mencionando anteriormente. ¿Por qué es un buen mashup?)

## Aplicaciones integradas

Google Photos: esta API se encargará de analizar cada foto determinando que elementos contienen, la fecha en la que se ha tomado o se ha recibido o el lugar geográfico en el que la foto toma lugar. Aplicará las restricciones recibidas desde las otras dos APIs.

LUIS: recibe los mensajes del usuario y es el encargado de interpretar los elementos clave del mensaje, como la restricción de fechas o que clase de fotos busca el usuario así como de confirmar o anular ordenes según el usuario. Debido a la gran variedad de posibles formatos de un mensaje, debe ser capaz de hallar las restricciones independientemente de dicha variedad, ignorando aquellos mensajes que no contengan información relevante.

|  |  |
| --- | --- |
| Nombre aplicación | URL documentación API |
| Google Photos | <https://developers.google.com/photos> |
| Google Cognitive Services | <https://cloud.google.com/vision?hl=es> |
| LUIS (Language Understanding Intelligent Service) | <https://www.luis.ai/home> |

Tabla 1. Aplicación integradas

# Prototipos de interfaz de usuario

## Vista x



## Vista y



# Arquitectura

## Diagrama de componentes

V

## Diagrama de despliegue

V

## Diagrama de secuencia de alto nivel

v