

Diagrama UML - Gestor de Cumpleaños

Este documento describe el diseño UML del sistema de gestión de cumpleaños implementado con Python y Streamlit. Incluye las clases principales, sus atributos, métodos y relaciones.

Diagrama UML del Gestor de Cumpleaños

```
+-----+
|  Person
+-----+
| - name: str
| - email: str
| - birthday: date
| - message_id: int
+-----+
| + __init__()
| + to_dict()
| + from_dict()
+-----+
```

▲
utiliza
▼

```
+-----+
| BirthdayManager
+-----+
| - people: List[Person]
+-----+
| + __init__()
| + load_birthdays()
| + save_birthdays()
| + add_person(person)
| + get_upcoming_birthdays()
+-----+
```

```
+-----+
|  MessageManager
+-----+
| - messages: List[str]
```

```
+-----+
| + __init__()
| + load_messages()
| + save_messages()
| + add_message(message)
| + get_message(index=None)
+-----+
```

```
+-----+
|   send_email()
+-----+
| Parámetros:
| - to_email: str
| - subject: str
| - body: str
| Funcionalidad: Enviar correo
+-----+
```

```
+-----+
|   Streamlit UI
+-----+
| - Visualiza cumpleaños
| - Permite registrar nuevos
| - Agrega mensajes
| - Simula envío automático
+-----+
```

Descripción del sistema

- Person representa a una persona con su nombre, correo, cumpleaños y un ID de mensaje personalizado opcional.
- BirthdayManager administra una lista de personas, incluyendo la carga, guardado y ordenación por cumpleaños próximo.
- MessageManager maneja una lista de mensajes de cumpleaños, permitiendo añadir y seleccionar mensajes (incluso aleatorios).
- send_email() simula el envío de un correo electrónico usando SMTP.
- Streamlit UI ofrece una interfaz de usuario para visualizar próximos cumpleaños, registrar nuevas personas y gestionar mensajes.