



# Documentación

---

EVENTOS DE ATLETISMO

Carrera: Ingeniería en Computación

Curso: Taller de Programación

Autor: Santiago Villarreal Arley

Carné: 2025120897

Profesor: William Mata

Fecha de entrega: 6 de mayo 2025

## Tabla de contenidos

<b>1. Enunciado del proyecto.....</b>	<b>3</b>
<b>2. Temas Investigados .....</b>	<b>4</b>
<b>Programación modular .....</b>	<b>4</b>
Marco teórico:.....	4
Uso en el proyecto:.....	4
Lugar de investigación: .....	4
<b>Listas anidadas .....</b>	<b>4</b>
Marco teórico:.....	4
Uso en el proyecto:.....	5
Ejemplo:.....	5
Lugar de investigación: .....	5
<b>Expresiones regulares .....</b>	<b>5</b>
Marco teórico:.....	5
Uso en el proyecto:.....	5
Lugar de investigación: .....	6
<b>Generación de PDFs con fpdf.....</b>	<b>6</b>
Marco teórico:.....	6
Uso en el proyecto:.....	6
Lugar de investigación: .....	6
<b>Envío de correos con smtplib.....</b>	<b>6</b>
Marco teórico:.....	6
Uso en el proyecto:.....	6
Lugar de investigación: .....	7
<b>Uso de herramientas de IA.....</b>	<b>7</b>
<b>Uso de herramientas de IA .....</b>	<b>7</b>
<b>3. Conclusiones del trabajo .....</b>	<b>8</b>
<b>4. Estadística de tiempos.....</b>	<b>11</b>
<b>5. Lista de revisión del proyecto .....</b>	<b>12</b>
<b>Referencias Bibliográficas .....</b>	<b>13</b>

## **1. Enunciado del proyecto**

El presente proyecto tiene como objetivo desarrollar un sistema que permita la gestión de eventos de atletismo. El sistema facilita el registro y consulta de atletas, pruebas, eventos y marcas. Además, se pueden generar reportes analíticos en formato PDF y enviarlos por correo electrónico a los participantes.

El proyecto se diseñó utilizando Python como lenguaje base, aplicando conceptos de programación modular, validaciones, estructura de datos y generación de archivos. También se consideró el uso de herramientas para documentar, automatizar y validar.

## 2. Temas Investigados

### Programación modular

#### Marco teórico:

La programación modular es un paradigma que promueve la separación del código en unidades o módulos independientes que cumplen funciones específicas. Esto permite un desarrollo más limpio, pruebas más sencillas y mejor mantenimiento del sistema.

#### Uso en el proyecto:

En el sistema de eventos de atletismo, cada módulo maneja una responsabilidad clara:

- eventos.py gestiona eventos deportivos
- pruebas.py permite definir pruebas y disciplinas
- atletas.py administra los datos personales y validaciones
- analisis.py centraliza reportes y análisis de marcas
- reportes.py y correo.py manejan funciones de salida como generación de PDFs y envío de emails

Esta estructura modular facilita el trabajo colaborativo y el reuso de componentes.

#### Lugar de investigación:

Python.org – Guía oficial de documentación

<https://docs.python.org/3/tutorial/modules.html>

### Listas anidadas

#### Marco teórico:

Las listas anidadas en Python permiten estructurar información en múltiples niveles jerárquicos. Son ideales para representar relaciones entre entidades, como eventos que contienen varias pruebas, y estas a su vez contienen marcas de atletas.

### Uso en el proyecto:

Se usaron para modelar marcas\_por\_evento, donde cada evento contiene pruebas, y cada prueba contiene tuplas con los datos de atletas y sus marcas.

### Ejemplo:

```
[  
  
    [101, ["V01", ("A001", 1, "11.23")], ["S02", ("A002", 4, "5.42")]],  
  
    [102, ["V01", ("A003", 2, "11.45")]]  
]
```

Esto permitió recorrer eficientemente los datos para validaciones y generación de reportes.

### Lugar de investigación:

Real Python – Artículo sobre estructuras de datos en Python

<https://realpython.com/python-lists-tuples/>

### Expresiones regulares

#### Marco teórico:

Las expresiones regulares (regex) son secuencias de caracteres que definen patrones de búsqueda. Son herramientas poderosas para validar y extraer información de texto.

### Uso en el proyecto:

Se usaron para validar que los correos ingresados al registrar atletas tuvieran el formato correcto:

```
patron = r'^[\w\.-]+@[\w\.-]+\.\w{2,}$'
```

Además, se implementó una validación de dominios válidos como gmail.com, tec.ac.cr, para asegurar la integridad del dato.

### Lugar de investigación:

W3Schools – Tutorial de Regex en Python

[https://www.w3schools.com/python/python\\_regex.asp](https://www.w3schools.com/python/python_regex.asp)

### Generación de PDFs con fpdf

#### Marco teórico:

La biblioteca fpdf en Python permite crear documentos en formato PDF desde cero. Admite añadir textos, celdas, encabezados y tablas con diseño personalizado.

#### Uso en el proyecto:

Se utilizaron funciones como `.add_page()`, `.cell()` y `.output()` para crear reportes como:

- Mejores marcas por prueba
- Marcas por evento
- Reportes por atleta

Cada PDF contiene información clara, formateada para impresión o envío digital, y se guarda automáticamente.

### Lugar de investigación:

PyFPDF – Documentación oficial de FPDF para Python

<https://pyfpdf.github.io/fpdf2/>

### Envío de correos con smtplib

#### Marco teórico:

smtplib es una biblioteca estándar en Python que permite enviar correos electrónicos usando el protocolo SMTP. Junto con `email.mime` se puede construir mensajes complejos con archivos adjuntos.

#### Uso en el proyecto:

Se implementó una función `enviar_reporte()` que:

- Se conecta a smtp.gmail.com
- Autentica con una contraseña de aplicación
- Adjunta el PDF generado
- Envía el correo a cada atleta si así lo solicita el usuario

Esto automatiza el proceso de entrega de resultados al finalizar un evento.

### Lugar de investigación:

GeeksForGeeks – Guía de envío de correo con Python

<https://www.geeksforgeeks.org/send-mail-using-smtp-in-python/>

### Uso de herramientas de IA

Marco teórico:

Uso de herramientas de IA es un concepto que permite...

Uso en el proyecto:

ChatGPT fue utilizado como asistente para generar y revisar código, diseñar estructura del proyecto, proponer mejoras y validaciones.

### Uso de herramientas de IA

Objetivo del uso	Herramienta utilizada	Prompt o pregunta	Respuesta	¿Cómo se usó o adaptó?	Reflexión crítica
Asistencia para diseñar estructura del código	ChatGPT	¿Cómo modularizar un sistema de eventos deportivos?	Respuesta estructurada en funciones y módulos separados	Se adaptó al contexto del proyecto en Python	Muy útil, aceleró el diseño sin errores.

Objetivo del uso	Herramienta utilizada	Prompt o pregunta	Respuesta	¿Cómo se usó o adaptó?	Reflexión crítica
Abrir un archivo en diferentes dispositivos	ChatGPT	¿Cómo abrir un archivo en diferentes sistemas operativos?	Utilizando diferentes funciones de la librería OS	Usé OS y las diferentes necesarias para que funcionara.	Muy útil, aceleró la investigación debido a que me explicó las funciones propias de OS relacionadas a abrir archivos.



### 3. Conclusiones del trabajo

El desarrollo de este proyecto representó una oportunidad valiosa para aplicar de manera integral múltiples conocimientos adquiridos a lo largo de la carrera. Se abordaron aspectos clave del desarrollo de software, desde el análisis y diseño de la solución hasta su implementación, validación y documentación. Entre los principales aprendizajes y logros destacan:

- Se aplicaron de forma práctica conocimientos de **programación modular**, estructuración de datos y validaciones robustas, lo cual permitió construir un sistema funcional, escalable y mantenible.
- El uso de **estructuras anidadas complejas** fue un reto que se logró manejar con precisión para representar de forma adecuada la relación entre eventos, pruebas y marcas.
- Se implementaron funcionalidades que van más allá del CRUD básico, como el análisis de datos, generación de reportes en PDF y envío de correos automatizados, integrando conceptos de interacción entre sistemas.
- La experiencia favoreció el fortalecimiento de habilidades de resolución de problemas, lógica algorítmica, y buenas prácticas de codificación y documentación.

### Problemas encontrados y soluciones aplicadas

- **Validación de estructuras anidadas:**  
Al trabajar con listas de listas, fue necesario diseñar recorridos eficientes que no generaran errores por índice ni omitieran información.  
**Solución:** se usaron bucles controlados y funciones auxiliares para manejar cada nivel jerárquico de forma clara y segura.
- **Bloqueo de correos por seguridad:**  
Durante el desarrollo del envío automatizado, se presentaron errores al usar contraseñas comunes debido a políticas de seguridad de Gmail.

**Solución:** se utilizó la opción de **contraseñas de aplicación** en Gmail para autenticación SMTP segura.

- **Generación de PDFs desalineados:**

En las primeras versiones, los reportes presentaban problemas de formato, márgenes y alineación de textos.

**Solución:** se ajustaron manualmente los valores de x, y, cell, y se probaron diseños hasta lograr una presentación clara y profesional.

- **Problemas con la contraseña de aplicación:**

Por temas de seguridad debí pedir una contraseña de aplicación al usuario para no vulnerar alguna de mis cuentas.

#### 4. Estadística de tiempos

Detalle de actividades realizadas y tiempo invertido:

Actividad Realizada	Horas
Análisis del problema	3
Diseño de algoritmos	2
Investigación de temas técnicos	3
Programación	3
Documentación interna	2
Pruebas	4
Elaboración del manual de usuario	2
Elaboración de documentación del proyecto	4
TOTAL	24

## 5. Lista de revisión del proyecto

Evaluación de los principales elementos del proyecto:

Elemento	% Avance	Puntos obtenidos	Análisis
CRUD completo de atletas	100%	10	Totalmente desarrollado.
Registro y validación de pruebas	100%	10	Validación cruzada con eventos incluida.
Análisis de datos	100%	10	Se generaron PDFs, ordenamientos y envío de correos.
Interfaz en consola	100%	10	Clara, validada y con menú profesional.
Documentación del proyecto	100%	10	Completa y explicativa.
Extras agregados	100%	10	Generación automática de datos, correo, análisis extendido.

Link al GitHub:

<https://github.com/Villarley/eventos-atletismo>

### Referencias Bibliográficas

1. Python Software Foundation. (2023). *The Python tutorial: Modules*. Python.org. <https://docs.python.org/3/tutorial/modules.html>
2. Real Python. (2021). *Python Lists and Tuples: A Complete Guide*. <https://realpython.com/python-lists-tuples/>
3. W3Schools. (2023). *Python RegEx*. [https://www.w3schools.com/python/python\\_regex.asp](https://www.w3schools.com/python/python_regex.asp)
4. PyFPDF. (2023). *fpdf2 documentation*. <https://pyfpdf.github.io/fpdf2/>
5. GeeksForGeeks. (2022). *Send mail using SMTP in Python*. <https://www.geeksforgeeks.org/send-mail-using-smtp-in-python/>