

---

## SISTEMA DE SIMULACIÓN DE RIEGO Y FERTILIZACIÓN AUTOMATIZADO EN INVERNADEROS CON DRONES CON TDA'S

---

Carnet 202400111 – Richard Steven Arizandietta Yol

### Resumen

El presente trabajo expone el desarrollo de un sistema de simulación de riego y fertilización automatizado en invernaderos, utilizando drones como agentes principales para ejecutar planes previamente definidos. La aplicación permite cargar configuraciones desde archivos XML, procesar la asignación de drones y plantas, y generar reportes en distintos formatos, incluyendo HTML, XML y representaciones gráficas con Graphviz. Esta propuesta responde a la creciente necesidad de modernizar la agricultura a través de la automatización y el uso de tecnologías digitales, aportando un modelo didáctico que integra estructuras de datos, programación orientada a objetos y aplicaciones web.

En el contexto nacional, donde la agricultura sigue siendo un sector estratégico, la implementación de herramientas de este tipo puede optimizar recursos hídricos y fertilizantes, reduciendo costos y promoviendo prácticas sostenibles. A nivel internacional, se relaciona con las tendencias de agricultura inteligente.

### Palabras clave

Simulación, Invernadero, Riego automatizado, Drones, Agricultura inteligente

### Abstract

*This paper presents the development of a simulation system for automated irrigation and fertilization in greenhouses, using drones as the main agents to execute predefined plans. The application allows configurations to be loaded from XML files, processes the assignment of drones and plants, and generates reports in different formats, including HTML, XML, and graphical representations with Graphviz. This proposal responds to the growing need to modernize agriculture through automation and digital technologies, providing a didactic model that integrates data structures, object-oriented programming, and web applications.*

*In the national context, where agriculture remains a strategic sector, the implementation of tools of this kind can optimize water and fertilizer resources, reducing costs and promoting sustainable practices. At the international level, it aligns with current trends in smart agriculture.*

### Keywords

*Simulation, Greenhouse, Automated irrigation, Drones, Smart agriculture*

## Introducción

La automatización de procesos agrícolas se ha convertido en un eje central para enfrentar los desafíos de la producción de alimentos en el siglo XXI. El crecimiento poblacional, la escasez de recursos hídricos y la necesidad de prácticas sostenibles exigen la implementación de tecnologías innovadoras que optimicen el uso de agua y fertilizantes. En este contexto, la integración de drones y sistemas de simulación ofrece una alternativa eficiente y adaptable para mejorar el rendimiento en invernaderos.

El presente trabajo desarrolla un sistema que combina estructuras de datos avanzadas, programación orientada a objetos y aplicaciones web para modelar y ejecutar planes de riego automatizados. La propuesta busca demostrar cómo la informática aplicada puede contribuir a la agricultura inteligente, generando reportes detallados y visualizaciones útiles para la toma de decisiones. Surge entonces la pregunta: ¿pueden los drones y la simulación computacional convertirse en aliados estratégicos para una agricultura más eficiente y sostenible?

## Desarrollo del Proyecto

### Objetivos del Sistema:

El proyecto tiene como finalidad diseñar e implementar un sistema de simulación de riego y fertilización automatizado en invernaderos, utilizando drones como agentes principales para la ejecución de planes definidos por el usuario. A partir de la lectura de archivos XML de configuración, el sistema permite organizar y procesar información sobre invernaderos, plantas y drones, con el propósito de modelar un entorno agrícola controlado.

### Arquitectura del sistema:

La arquitectura del sistema se diseñó bajo un enfoque modular, integrando componentes de

programación orientada a objetos, estructuras de datos personalizadas y una interfaz web desarrollada con Flask. El núcleo del proyecto está constituido por el módulo Sistema.py, el cual funciona como controlador principal encargado de gestionar la carga de archivos XML, la creación de objetos y la coordinación de la simulación.

### Clases principales:

- **Invernadero:** representa el entorno agrícola, contiene las hileras, plantas y los drones asignados.
- **Dron:** actúa como agente ejecutor, encargado de recorrer las hileras, regar y fertilizar las plantas según el plan establecido.
- **Planta:** almacena información sobre el tipo de cultivo, litros de agua y gramos de fertilizante necesarios.
- **PlanRiego:** define la secuencia de acciones a realizar sobre las plantas de un invernadero.

### Flujo de Ejecución:

El flujo de ejecución del sistema inicia con la carga de un archivo XML, el cual contiene la descripción de los invernaderos, sus plantas, la asignación de drones y los planes de riego. Esta información es procesada por el módulo principal, que crea las estructuras de datos correspondientes para organizar los elementos del entorno. Una vez cargados los datos, el usuario selecciona un invernadero y un plan de riego específico desde la interfaz web. El sistema asigna a cada dron una hilera determinada y establece la cola de plantas que deben ser atendidas. Durante la simulación, los drones recorren sus hileras paso a paso, ejecutando

acciones como moverse, esperar, regar o aplicar fertilizante. Cada acción es registrada en una línea de tiempo que refleja el avance del proceso en intervalos de segundos. Finalmente, el sistema genera reportes de resultados en distintos formatos: un archivo HTML con el detalle de la simulación, un XML con la información procesada y gráficos en formato DOT (Graphviz) que representan la evolución de las estructuras de datos utilizadas.

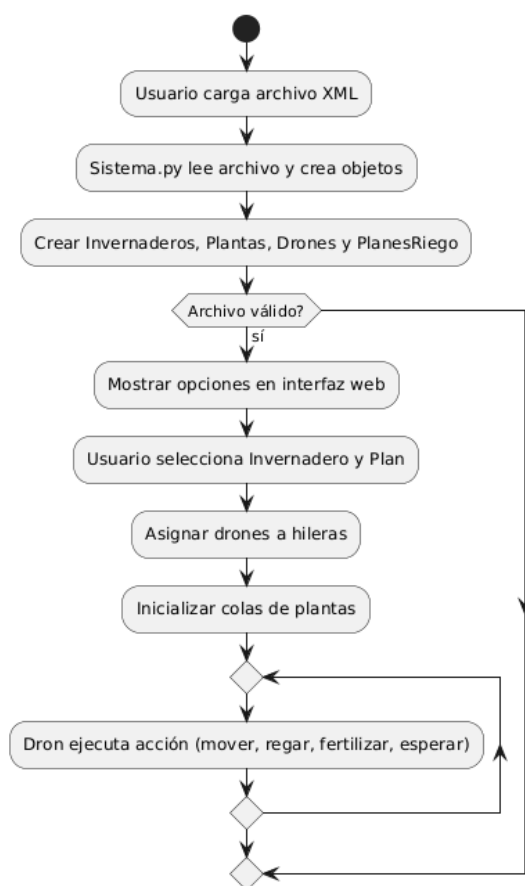


Figura 1. Flujo de Ejecución del programa.

Fuente: elaboración propia.

### Clases y estructuras principales:

El sistema se diseñó con un enfoque orientado a objetos, donde cada clase representa un elemento

clave del dominio agrícola o de la simulación. A continuación, se describen las más relevantes: Sistema: clase controladora que centraliza la ejecución. Es responsable de cargar los archivos XML de entrada, instanciar los invernaderos, drones y planes de riego, además de coordinar la simulación y generar reportes.

Invernadero: almacena la información del entorno agrícola, incluyendo el número de hileras, plantas por hilera, las instancias de Planta y los drones asignados. Además, gestiona la relación con los planes de riego.

- **Planta:** representa un cultivo dentro del invernadero. Contiene atributos como hilera, posición, litros de agua requeridos, gramos de fertilizante y el tipo de planta.
- **Dron:** actúa como agente encargado de ejecutar las acciones de riego y fertilización. Posee atributos como identificador, hilera asignada, posición actual, recursos utilizados y una cola de plantas pendientes.
- **PlanRiego:** define la secuencia de acciones que deben realizar los drones sobre las plantas de un invernadero en un tiempo determinado.
- **SimulacionRiego:** gestiona la ejecución de un plan de riego, registrando paso a paso las acciones de cada dron en una línea de tiempo y generando reportes en diferentes formatos.

### Estructuras de datos:

- **ListaSimpleEnlazada:** organiza las instancias de drones e invernaderos.
- **Matriz:** representa la disposición espacial de las plantas dentro del invernadero.

Estas clases trabajan de manera conjunta para garantizar la modularidad del sistema y la claridad en la simulación de riego automatizado.

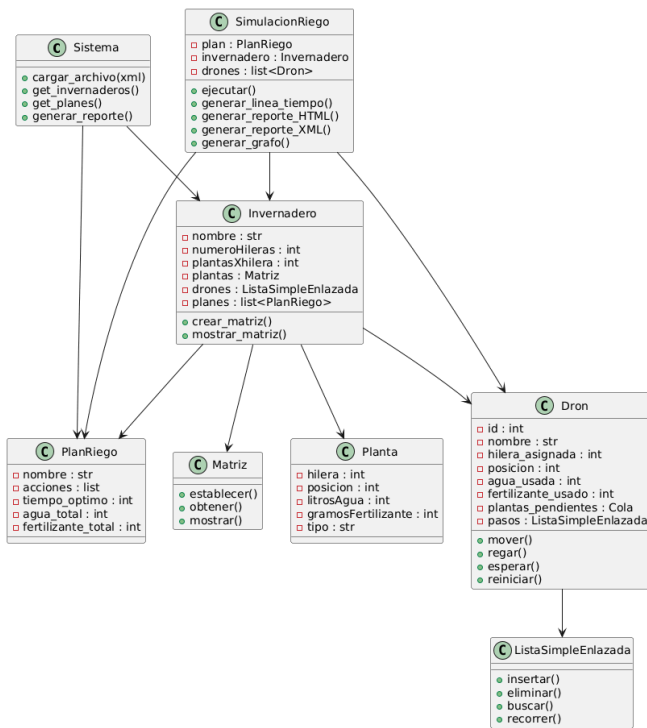


Figura 2. Diagrama de clases de la arquitectura del sistema.

Fuente: elaboración propia.

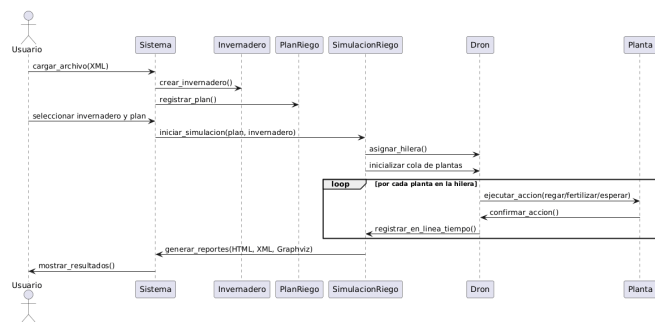


Figura 3. Diagrama de clases de la arquitectura del sistema.

Fuente: elaboración propia.

## Despliegue de componentes

El diagrama de despliegue representa la forma en que los diferentes componentes del sistema interactúan en tiempo de ejecución.

- **Usuario:** accede a la aplicación a través de un navegador web, que envía solicitudes HTTP al servidor.
- **Servidor Local:** ejecuta la aplicación web desarrollada con Flask, la cual actúa como puente entre el usuario y la lógica del sistema.
- **Aplicación Flask (app.py):** gestiona las rutas, renderiza las plantillas HTML mediante Jinja2 y enlaza con los recursos estáticos (CSS).
- **Módulos del sistema:** contienen la lógica principal de la simulación. Cada clase (Sistema, Invernadero, Dron, Planta, PlanRiego, etc.) organiza la información y ejecuta procesos.
- **Flujo general:** el navegador envía solicitudes (carga de archivos, selección de invernaderos, ejecución de simulación), Flask procesa estas acciones apoyándose en los módulos Python y devuelve al usuario reportes en HTML, XML o gráficos generados con Graphviz.

Este diseño asegura modularidad, claridad y escalabilidad, permitiendo que el sistema pueda ampliarse fácilmente o desplegarse en servidores de mayor capacidad.

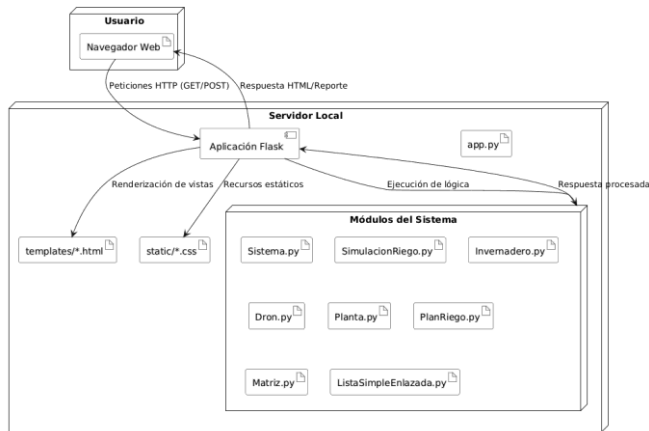


Figura 3. Diagrama de despliegue.

Fuente: elaboración propia.

## Reportes Generados

Uno de los aportes más relevantes del sistema es la capacidad de generar reportes en distintos formatos, lo que facilita la interpretación y análisis de los resultados de la simulación. Estos reportes cumplen funciones técnicas y didácticas, permitiendo evaluar el comportamiento de los drones, los tiempos de ejecución y el consumo de recursos.

- **Reporte en HTML:** contiene un resumen del invernadero y el plan ejecutado, junto con una tabla de línea de tiempo donde se registran las acciones de cada dron en intervalos de segundos. Incluye también métricas de agua y fertilizante consumidos, así como el tiempo total requerido.
- **Reporte en XML:** presenta la información estructurada de manera jerárquica, de acuerdo con el estándar utilizado para la configuración inicial. Este archivo es especialmente útil para compartir o reutilizar los resultados de la simulación en otros sistemas.
- **Reporte en Graphviz (TDA):** genera un grafo visual del estado de las estructuras de datos durante la simulación. Permite observar las colas de plantas atendidas por cada dron, así como el progreso de ejecución en diferentes momentos de tiempo (t).

La diversidad de formatos asegura que los resultados puedan ser revisados tanto por usuarios finales como por programadores interesados en el funcionamiento interno del sistema.

## Conclusiones

El desarrollo del sistema de simulación de riego y fertilización automatizado en invernaderos demostró cómo el uso de estructuras de datos, programación orientada a objetos y aplicaciones web pueden integrarse para resolver problemas del ámbito agrícola de manera eficiente y escalable. El proyecto no solo cumple con su propósito académico de aplicar conocimientos técnicos, sino que también evidencia el potencial de estas herramientas para contribuir a la agricultura inteligente y sostenible.

Entre los aportes más relevantes destacan la capacidad de modelar escenarios de riego con drones, el uso de reportes en múltiples formatos para un análisis integral y la visualización gráfica del estado de los TDAs, lo que facilita la comprensión tanto para usuarios como para programadores.

Este trabajo abre la reflexión sobre el papel de la informática en el futuro de la agricultura: ¿pueden los sistemas basados en simulación y automatización convertirse en pilares estratégicos para garantizar la seguridad alimentaria en contextos de escasez de recursos?

## Referencias bibliográficas

C. J. Date. (1991). *An Introduction to Database Systems*. Addison-Wesley Publishing Company, Inc.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms (3rd ed.)*. MIT Press.

Grinberg, M. (2018). *Flask Web Development: Developing Web Applications with Python (2nd ed.)*. O'Reilly Media.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.

Zhang, Y., Wang, G., & Li, X. (2017). *Smart Agriculture: An Approach Towards Better Agriculture Management*. *Journal of Agricultural Science and Technology*, 19(2), 1-10.