# **DIA Organics: Automated Garden**

## **Overview**

This project implements a Smart Garden Simulation that allows users to create and manage a virtual garden. The system features interactive plant selection, environmental effects such as rain and temperature changes, and parasite interactions. The simulation is controlled via a graphical user interface (GUI) or through direct API calls for automation.

## **Features**

* Graphical User Interface (GUI) for intuitive garden management.
* Plant selection and management via dropdown menus.
* Environmental simulation including rain, temperature changes, and parasites.
* Real-time animations for rain, sun, and plant health updates.
* Event-driven system for handling weather and parasite interactions.
* Simulation logs stored in GardenUIcontroller.log for review.

## **Setup & Installation**

### **Prerequisites**

* Java (JDK 11 or higher)
* JavaFX for GUI support

### **Steps**

1. Extract the project folder to your preferred directory.
2. Open the project in an IDE (e.g., IntelliJ, Eclipse, or VS Code).
3. Ensure JavaFX is configured properly in your IDE.
4. Run the application:
   * Open HelloApplication.java
5. Click Run or execute- maven javafx:run

### **Running the Program**

#### **With GUI:**

1. Once you start the HelloApplication.java
2. The GUI will appear
3. Select a plant/vegetable/tree from the dropdown bar.
4. The plant will be added to the grid.
5. To stop the program, stop the HelloApplication.java

**Without GUI:**

Steps for the system to run without UI.

* There is a method named runAPIScheduledTasksWithoutJavaFX which makes the system work without the UI.
* In order to test it, just run the system with following code where the argument ‘no ui’ will make sure that it works without UI.
* **java com.example.ooad\_project.HelloApplication --no-ui**

**Log**:

There is separate log for each system along with the entire activity log.

If you are running with GUI, GardenUIController.log will be the appropriate log.

Else if you are using API, GardenAPI.log will be the appropriate log