Graduation Project Proposal

Project Information		
Project Title	**Department/Faculty/University**	**Project Field/Discipline**
Agri Lens	Computer Science/ Faculty of Computer and Information Sciences/ Mansoura University	Artificial Intelligence (AI), Agriculture, Robotics
Advisors' Names	**Advisors' Mobile Numbers**	**Advisors' Email Addresses**
Samir El mougy	**01011531792**	**mougy@mans.edu.eg**
May Mohamed Beder	**01025154591**	**maybedeer@mans.edu.eg**
Students' Names	**Students' Mobile Numbers**	**Students' Email Addresses**
Ibrahim Mohamed Ibrahim Alsayed Hegazi	**01551711251**	**ihegaziwork@gmail.com**
Ibrahim Mohamed Yousef Yousef Al-Banawy	**01066250312**	**albanawyibrahim@gmail.com**
Basel Mohamed Salah Elazaly	**01009276101**	**basel.elazaly2003@gmail.com**
Ahmed Khalf Attia Abo Almaati	**01220725319**	**cra19245@gmail.com**
Ahmed Ashraf Mohamed Kamel	**01019721138**	**ahmed_kamel2003@yahoo.com**
Abdel Aleem Mohamed Abdel Aleem Mohamed Elsherbiny	**01552475851**	**abdelaleem.mohammed2150203@gmail.com**

Motivation
Please write why you chose this project idea, explaining clearly the problem that the project is addressing

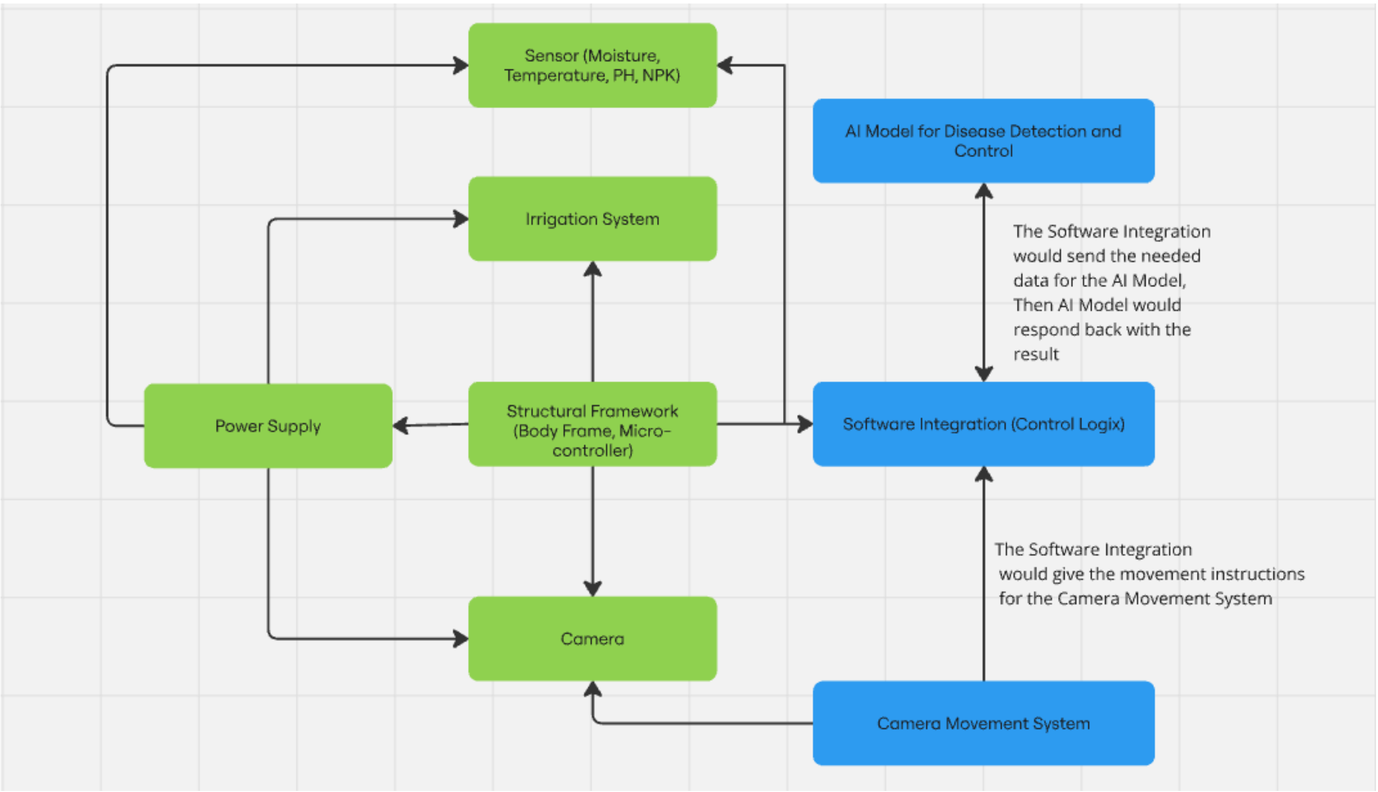
This project was chosen to address specific agricultural challenges faced in Egypt, including:

1. **Late Detection of Plant Diseases**: In Egypt, plant diseases often go undetected until they spread, leading to significant crop losses. This project uses AI-powered disease detection through a smart camera system to ensure early identification and intervention.
2. **Inefficient Resource Usage**: Over-irrigation and excessive use of fertilizers are common, contributing to water wastage and soil degradation. The project's precise irrigation system and environmental sensors optimize resource use, reducing waste and increasing sustainability.
3. **Lack of Accessible Smart Farming Solutions**: Advanced agricultural technologies are often costly or inaccessible for small and medium-scale farmers in Egypt. This modular and cost-effective system provides a practical and affordable entry point to smart farming.
4. **Desertification and Agricultural Land Loss**: Egypt faces increasing desertification, limiting arable land and threatening food security. This project promotes controlled-environment agriculture, enabling farming in non-arable areas by creating self-contained systems that mitigate land loss.
5. **Lack of Automation in Farming**: Egyptian agriculture heavily relies on manual labor, which can be inefficient and error-prone. The integration of automated disease detection, irrigation, and environmental control in this project addresses this gap, enhancing productivity and reducing dependency on human intervention.
6. **Scalability for Large-Scale Farming**: Current solutions often lack scalability for broader applications. This project starts with a small-scale 3×1 box but is designed to be expandable for large-scale farming, making it adaptable to Egypt’s needs for mass agricultural production.

Why do you think your project should be funded? for which the applicants write in a few lines where the help statement should be "Explain in no more than 3 lines the new and innovative aspects in your project that make it worthy of funding."
Our project introduces a cost-effective, scalable, and modular smart farming system tailored to address Egypt's agricultural challenges, including early disease detection, optimized resource usage, and farming in desertified areas. By combining AI-powered automation, precise irrigation, and environmental control, it offers an innovative solution to enhance productivity and sustainability in agriculture. This accessible system empowers farmers while promoting eco-friendly practices.

Block Diagram

Please insert the project detailed block diagram below, (Please highlight the parts that will be implemented in different colors than the parts that will be purchased)



Prototype Description and Specifications

Please note that ITAC only funds projects that result in a prototype. Include a clear description of how the prototype will operate, explaining a scenario/use case of the operation. Also include the performance metrics you target in the prototype.

The proposed prototype aims to address several critical agricultural challenges in Egypt through the integration of smart farming technologies. The prototype will consist of a modular, scalable system designed to help detect plant diseases, optimize resource use, and automate farming processes, contributing to improved crop yield and sustainability.

Key Components:

1. **Smart Camera System for Disease Detection:**
    - **Technology: AI-powered image recognition.**
    - **Function: Detects early signs of plant diseases by analyzing images captured by cameras placed in the agricultural area.**
    - **Scenario: The camera takes periodic images of the plants and processes them through a machine learning model to detect any visible signs of diseases or pests. If an issue is detected, an alert is sent to the farmer for immediate intervention.**
    - **Performance Metrics: 95% detection accuracy.**
2. **Precision Irrigation System:**
    - **Technology: Automated irrigation controlled by environmental sensors (soil moisture, temperature, humidity).**
    - **Function: Optimizes water usage by only irrigating when necessary, reducing water wastage.**
    - **Scenario: Based on real-time data from the soil moisture sensors, the system activates irrigation only in areas where moisture levels fall below a pre-set threshold.**
    - **Performance Metrics: 30% reduction in water usage compared to conventional irrigation systems.**
3. **Environmental Control and Fertilizer Optimization:**
    - **Technology: Environmental sensors and fertilizer suggestion system.**
    - **Function: Monitors environmental conditions (e.g., temperature, humidity) and suggests the needed fertilizer based on crop requirements.**
    - **Scenario: The system will gather data on temperature, humidity, and soil nutrients to identify trends and suggest the right fertilizer usage. The system will not automatically adjust fertilizer levels but will provide data that will later be analyzed for trends and patterns.**
    - **Data Collected: Temperature, humidity, soil moisture, and nutrient levels.**
    - **Use for Future Analysis: The data collected will be analyzed later to evaluate the correlation between environmental factors and optimal fertilizer usage, allowing for more accurate future recommendations.**
4. **Controlled-Environment Agriculture Box (3×1 unit):**
    - **Technology: Modular self-contained system featuring automated irrigation, and disease detection.**
    - **Function: Provides a environment for plants, enabling farming in desertified or non-arable areas.**
    - **Scenario: The prototype will operate in a 3×1 meter area, with sensors collecting data on key environmental factors (e.g., temperature, humidity). The system does not adjust these factors automatically but provides data for later analysis.**
    - **Data Collected: Temperature, humidity, soil moisture, and crop growth parameters.**
    - **Use for Future Analysis: The collected data will be analyzed to identify patterns that optimize crop growth in non-arable conditions, helping to refine controlled-environment strategies for larger-scale applications.**
5. **Automation of Manual Farming Tasks:**
    - **Technology: Robotics and AI systems for monitoring and control.**
    - **Function: Automates key tasks such as disease detection, and irrigation, reducing reliance on human labor.**
    - **Scenario: Automated sensors monitor the crop growth and environmental factors. When necessary, the system adjusts conditions or alerts the operator for manual intervention.**
    - **Performance Metrics: 50% reduction in manual labor requirements, with 10% improvement in overall productivity.**

Project Plan

Please define the approach and phases to deliver the intended project outcome.

Phase 1: Planning and Requirement Gathering
Objective: Define the project scope, gather requirements, and plan the design for a smart farming system.

1. **Task 1: Define Project Scope**
    - **Identify the purpose and goals of the smart farming system.**
    - **Outline the functional requirements:**
        - **Hardware: sensors, actuators, camera system, IoT integration, and automation features.**
        - **Software: Mobile app, web app, backend system, and AI functionalities (e.g., disease detection, prediction models).**
    - **List non-functional requirements: scalability, energy efficiency, reliability, and ease of deployment.**
2. **Task 2: Research and Feasibility Study**
    - **Research sensor specifications (soil pH, soil moisture, NPK levels, humidity, temperature, and light intensity).**
    - **Study IoT-enabled data transmission and remote monitoring techniques.**
    - **Review disease detection techniques using a camera system (e.g., image processing or machine learning).**
    - **Explore automated irrigation and nutrient delivery systems.**
    - **Research frameworks for:**
        - **Mobile app (e.g., Flutter, React Native)**
        - **Web app (e.g., React.js, Angular)**
        - **Backend (e.g., Django, .Net Framework)**
        - **AI models for disease detection and environmental prediction**
    - **Explore cloud integration for IoT and AI data processing**
3. **Task 3: Create a High-Level Architecture**
    - **Draft a system architecture for hardware, mobile app, web app, backend, and AI modules.**
    - **Decide on microcontroller/processor (e.g., Arduino, Raspberry Pi, ESP32) based on computational and connectivity requirements.**
    - **Plan API integration to connect mobile/web applications with the backend and IoT devices.**
    - **Plan the IoT architecture, including cloud or local storage options for data analysis.**
4. **Deliverables**
    - **Project requirement document.**
    - **High-level system architecture diagram (including hardware, software, and cloud integration).**
    - **Component list and budget estimation.**
    - **Technology stack proposal.**

Phase 2: System Design
Objective: Create detailed designs for both hardware and software components.

1. **Task 1: Hardware Design**
    - **Design the sensor placement layout (moisture, humidity, and temperature sensors in fixed positions).**
    - **Design the movement system for the NPK, pH sensors and camera (e.g., servo motor or stepper motor with rail systems).**
2. **Task 2: Circuit Design**
    - **Use Proteus to create a schematic diagram.**
    - **Include power supply circuits, sensor connections, motor driver circuits, and a microcontroller.**
    - **Simulate the circuit to ensure functionality.**
3. **Task 3: Software Design**
    - **Develop a flowchart for system operation, including:**
        - **Data acquisition from sensors.**
        - **Movement system control.**
        - **Disease detection using the camera.**

- Outline data logging and processing requirements.
- Define communication protocols (e.g., I2C, UART, or SPI).

4. Task 4: Backend and API design
   - Design a REST API or GraphQL backend to:
     - Handle real-time sensor data from IoT devices.
     - Facilitate communication between mobile/web applications and the database
     - Integrate AI models for disease detection and data insights.
   - Plan database schema for storing sensor data, user activity, and system logs.
5. Task 5: Mobile and Web App Design
   - Design user interfaces for:
     - Mobile app: Real-time monitoring, notifications, and control (e.g., irrigation settings).
     - Web app: Advanced data analytics, AI-driven recommendations, and system settings.
   - Create wireframes for a seamless user experience.
6. Task 6: AI Model Design
   - Plan AI workflow for:
     - Disease detection (image processing via camera data).
     - Predictive analytics for irrigation and nutrient delivery.
   - Define data processing and model training pipelines.
7. Deliverables:
   - Finalized circuit schematic.
   - Movement system design and control algorithm.
   - Software design document:
     - Backend architecture and database schema.
     - Mobile and web app wireframes.
     - AI model design and data pipelines.

Phase 3: Prototyping and Simulation

Objective: Build a working prototype and validate the design.

1. Task 1: Prototype the Circuit
   - Assemble the circuit on a breadboard or test rig based on the Proteus schematic.
   - Test individual components (e.g., sensors, motors) for expected performance.
2. Task 2: Simulate in Proteus
   - Simulate the entire circuit, including sensor inputs, motor movement, and microcontroller logic.
   - Debug any issues in the design or logic.
3. Task 3: Test Movement System
   - Build a basic mockup of the rail system for the NPK sensor and camera.
   - Test precision, speed, and reliability of the movement system.
4. Task 4: Backend Prototyping (NEEDS REVISION)
   - Develop basic backend services for:
     - Sensor data logging.
     - API endpoints for mobile/web applications.
5. Task 5: Mobile and Web App Prototyping
   - Create a minimum viable product (MVP) for:
     - Mobile app: Sensor data visualization and basic controls.
     - Web app: Sensor data analytics dashboard.

6. **Task 6: AI Model Prototyping**
   - Train and test a prototype AI model for disease detection using mock data.
7. **Task 7: Cloud Integration and Simulation**
   - Simulate sensor data transmission and verify cloud data storage and analytics
   - Test real-time data monitoring and control through cloud and dashboards
   - Debug any issues in cloud connectivity and data processing
8. **Deliverables:**
   - Functional prototype (circuit + movement system).
   - Simulation results and debugging logs.
   - Backend MVP (Minimum Viable Product).
   - Mobile and web app prototypes.
   - Initial AI model.
   - Simulation results, debugging logs, and cloud integration test results.

Phase 4: PCB Design and Fabrication
Objective: Design and fabricate a compact and robust PCB for the system.

1. **Task 1: Design the PCB**
   - Transfer the Proteus schematic to the ARES PCB layout tool.
   - Optimize component placement and trace routing.
   - Add mounting holes and headers for sensors and actuators.
2. **Task 2: Fabricate the PCB**
   - Generate Gerber files and send them to a PCB manufacturer.
   - Receive and inspect the fabricated PCB for quality.
3. **Deliverables:**
   - PCB layout design.
   - Manufactured PCB ready for assembly.

Phase 5: System Integration
Objective: Assemble all components and integrate hardware with software.

1. **Task 1: Assemble the System**
   - Mount sensors, actuators, and PCB onto the physical box.
   - Connect the movement system to the NPK, pH sensors and camera.
2. **Task 2: Implement and Test Software**
   - Write the microcontroller code to acquire sensor data, control the movement system, and log readings.
   - Test the system for each functionality:
     - Sensor data acquisition.
     - Motor control and positioning.
3. **Task 3: Backend Deployment (NEEDS REVISION)**
   - Deploy backend services to the cloud (e.g., AWS, Azure).
   - Set up real-time data streaming for IoT devices.
4. **Task 4: Mobile and Web App Integration**
   - Connect apps to the backend for real-time data retrieval and control.
   - Implement push notifications for critical alerts.
5. **Task 5: AI Integration**
   - Deploy trained AI models on the backend.

- **Enable real-time disease detetction and predictive analytics through APIs.**

6. **Task 6: Test the Entire System**
   - **Conduct end-to-end tests for all functionalities**
     - **Hourly data collection and cloud storage**
     - **Cloud-based remote control and monitoring**
     - **Movement system operation**
     - **Camera-based disease detection**
7. **Deliverables:**
   - **Fully integrated system with cloud support.**
   - **Test results and performance reports**

Task 6: Comprehensive System Testing

Objective: Conduct thorough end-to-end testing to validate the integration and functionality of all system components.

1. **Task 1: Prepare Testing Environment:**
   - **Set up the entire system in a controlled environment, ensuring all hardware and software components are properly connected and configured.**
   - **Ensure network connectivity for IoT devices, backend services, mobile, and web applications.**
   - **Monitor real-time performance over a set period using cloud dashboards.**
2. **Task 2: Test Core Functionalities:**
   - **Sensor Data Acquisition:**
     - **Verify that all sensors (e.g., pH, NPK, moisture, temperature, humidity) provide accurate and real-time data.**
     - **Simulate edge cases like sensor failure or extreme environmental conditions to test system resilience.**
   - **Movement System Operation:**
     - **Test the precision, speed, and accuracy of the movement system controlling the NPK sensor and camera.**
     - **Validate smooth integration between hardware (motors) and microcontroller logic.**
   - **Camera-Based Disease Detection:**
     - **Use pre-collected images of healthy and diseased plants to test the AI model's ability to detect diseases in real time.**
     - **Assess model accuracy, response time, and handling of ambiguous inputs.**
   - **Cloud Wise validation:**
     - **Analyze the accuracy and reliability of sensor data stored in the cloud.**
     - **Evaluate the efficiency of cloud-based data collection and communication.**
3. **Task 3: Verify Backend Operations:**
   - **Ensure the backend system can handle real-time data streaming from IoT devices without delays or data loss.**
   - **Test database functionality, including data logging, retrieval, and storage of historical data.**
4. **Task 4: Validate Mobile and Web Application Integration:**
   - **Test mobile and web apps for real-time data visualization and system control.**
   - **Validate push notifications for alerts like abnormal sensor readings or detected plant diseases.**
   - **Check responsiveness and usability across different devices and browsers.**
5. **Task 5: AI Model Deployment and Integration:**
   - **Test API endpoints for AI-based functionalities, including predictive analytics and disease detection.**
   - **Validate that AI predictions and recommendations are accurate and actionable.**
6. **Task 6: Stress Testing and Scalability:**
   - **Simulate a high number of sensor inputs to test system performance under heavy load.**
   - **Assess the system's ability to scale, such as adding more sensors or extending its functionality to monitor multiple fields.**
7. **Task 7: Reliability and Fault Tolerance Testing:**

- Conduct power failure simulations and check if the system can recover without data loss.
- Test fallback mechanisms in case of IoT connectivity loss or backend downtime.

8. **Task 8: Feedback Testing:**
   - Allow end users or stakeholders to operate the system in the testing environment.
   - Gather feedback on system usability, responsiveness, and reliability.

Deliverables:

- Comprehensive test reports documenting functionality, performance, and reliability results.
- Logs of detected issues and corresponding resolutions.
- A validated, fully functional, and reliable smart farming system ready for deployment.

Phase 7: Deployment and Evaluation

Objective: Deploy the system and evaluate its performance in a controlled environment.

1. Task 1: Deploy The Hardware in a Test Environment
   - Install the smart farming system in a suitable test environment (e.g., open field, greenhouse, or indoor setup).
   - Monitor real-time performance over a set period (e.g., 1-2 weeks).

2 Task 2: Deploy Applications
   - Deploy the web app on a secure hosting platform.

3. Task 3: Evaluate System Performance
   - Analyze the accuracy and reliability of sensor data (e.g., soil pH, moisture, NPK levels, temperature, and humidity).
   - Evaluate the efficiency of the IoT-enabled data collection and communication system.
   - Assess the accuracy of the disease detection system (if using image processing or machine learning).
   - Measure the effectiveness of automated irrigation and nutrient delivery mechanisms.
   - Monitor app responsiveness, data sync, and AI recommendations.

4. Task 4: Gather Feedback
   - Identify areas for improvement (e.g., sensor calibration, IoT connectivity, and software optimization).
   - Incorporate feedback from test results for future iterations.

5. Deliverables
   - Deployment report with performance evaluation.
   - Recommendations for optimization.

Phase 8: Documentation and Handover

Objective: Finalize documentation and prepare for scaling or further development.

1. Task 1: Create System Documentation
   - Document hardware connections, and user instructions.
   - Include API documentation, app user guides, and AI model workflows.
   - Include troubleshooting guides for common issues.
2. Task 2: Prepare for Handover or Scaling
   - Train users or stakeholders to operate the system.
   - Explore scalability options (e.g., expanding to more plants or automating additional tasks).
3. Deliverables:
   - Comprehensive system documentation.
   - Final project report.
   - System ready for scaling or further development.

Prototype Prospects

List the Egyptian ICT companies that may be interested in the developed prototype and the end-users/customers (name the specific class of individuals, governmental agencies, ministries … etc. that will benefit from the prototype)
List of ICT Companies: 1. **ITIDA (Information Technology Industry Development Agency)** 2. **Valeo Egypt** 3. **Systel Telecom** 4. **Si-Ware Systems** 5. **Vodafone Egypt** 6. **NTRA (National Telecom Regulatory Authority)** **Potential End-Users/ Consumers:** 1. **Agricultural Ministries and Governmental Agencies** 2. **Farmers and Agricultural Cooperatives** 3. **Agri-Tech Startups** 4. **Universities and Research Institutes** 5. **Export-Oriented Agricultural Companies** 6. **Food Security and Sustainability Organizations** 7. **Smart City Initiatives** 8. **Greenhouse Operators** 9. **Environmental Ministries and Agencies** 10. **International Development Agencies**