INTROUCTION TO PROBLEM SOLVING AND PROGRAMMING

2025-2026

CROP PROFIT CALCULATOR

SUBMITTED BY
ASHISH SINGH RAJPUT

SUBMITTED TO
MOSES DIAN

Introduction

To assist farmers, agriculture students, and analysts in determining the anticipated profit or loss from cultivating a specific crop, the Crop Profit Prediction System is a straightforward but effective tool. The system forecasts financial results by using important agricultural inputs, including land area, yield per acre, market price, and cultivation cost. This helps users make informed farming and investment decisions.

Problem Statement

Farmers frequently make decisions based on inaccurate profitability data. Many rely on presumptions or conventional wisdom, which could result in monetary losses. Before cultivation starts, there is a need for an easy-to-use tool that can forecast profit or loss.

Functional Needs

User input must be accepted by the system:

- Name of crop
- Area of land (acres)
- Yield anticipated per acre (kg)
- Market value in kilograms (₹)
- Cost of cultivation per acre (₹)

Numerical inputs must be validated.

Calculation is required:

- Overall yield
- Total earnings
- Total expense
- Gain or loss

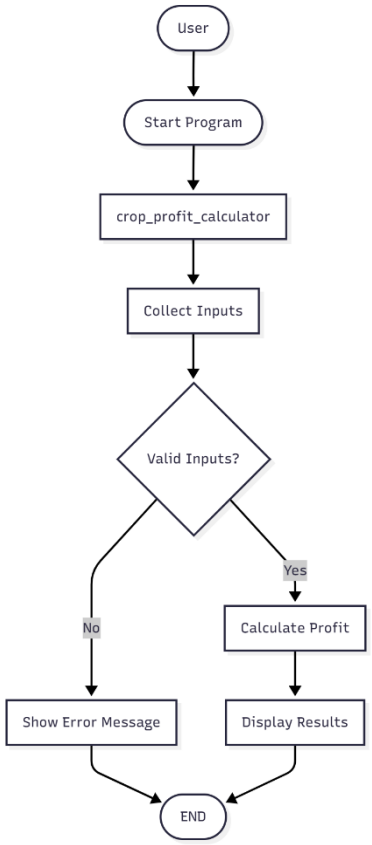
Results must be presented to the user in an understandable manner.

Non-functional prerequisites

Usability: Even for novices, it should be easy to use.
Reliability: Every calculation should be precise.
Efficiency: Should produce outcomes right away.
Maintainability: Code should be simple to expand or change.
Portability: It should function on any system that has Python installed.

Architecture of the System

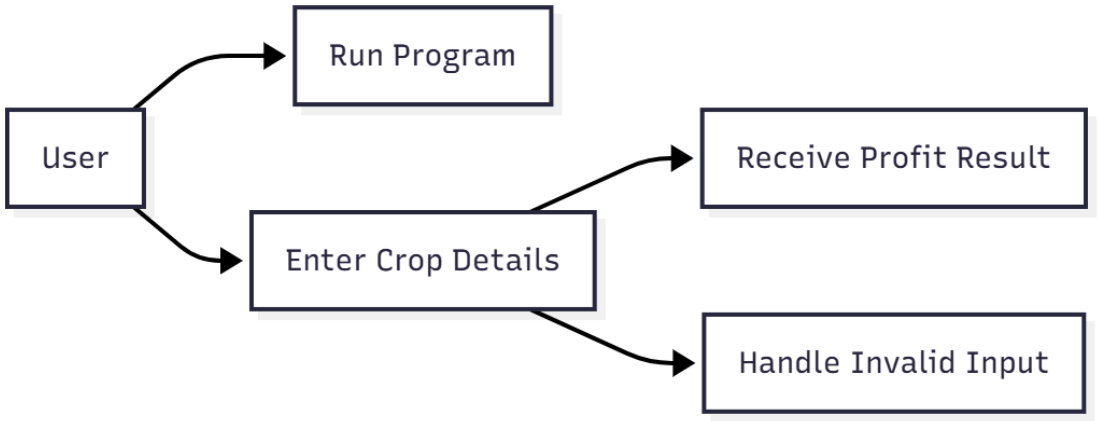
An architecture of straightforward linear flow:



Every step serves a distinct function to guarantee precise outcomes.

Design Diagrams

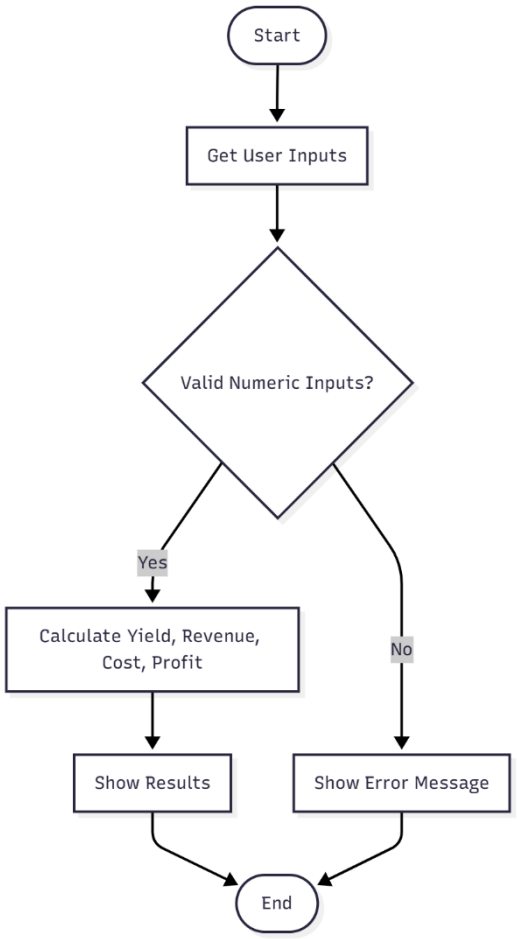
a. Use Case Diagram



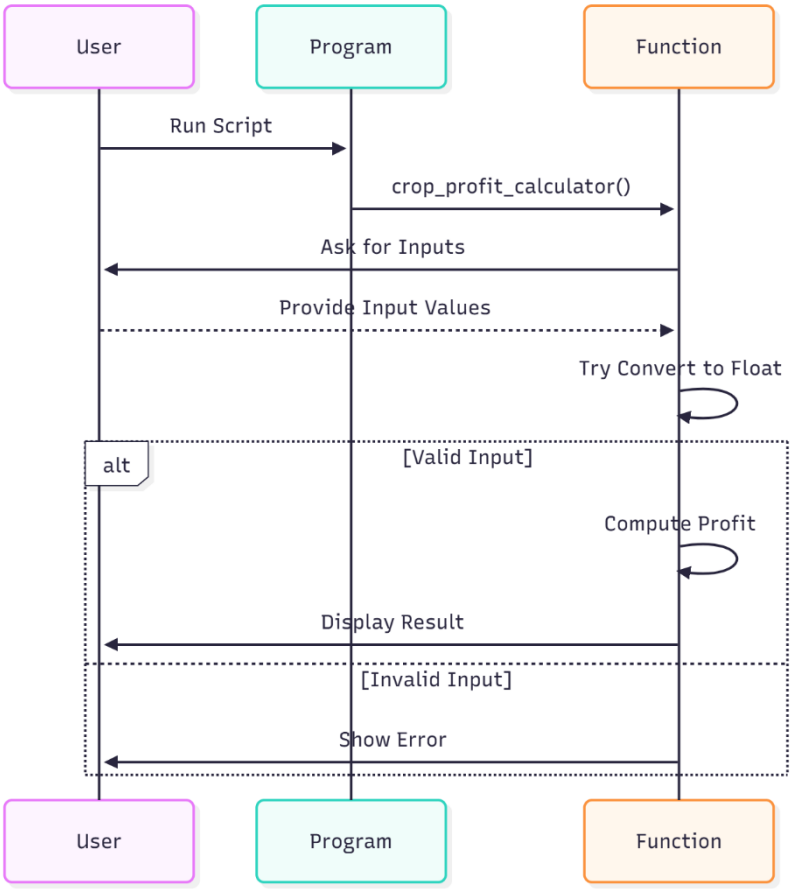
b. Workflow Diagram:

Step-by-step flow:

1. Start
2. Input values
3. Validate
4. Calculate
5. Display result
6. End



c. Sequence Diagram



Design Decisions and Their Rationale

- We chose Python for its simplicity and readability.
- We chose a command line interface because it is lightweight and easier for beginners to start using.
- We chose to use a modular design so, if the program ever grew, like to add a GUI or a database, it would be simple and easy.
- We lowered the number of dependencies to keep the program portable.

Implementation Details

The code uses:

- Python input/output
- try/except for error handling
- basic math to calculate profit
- a single main function called crop_profit_calculator()
- formatted print to have a human-readable output

Screenshots of source code:

Input

```python
def crop_profit_calculator():
    print("crop profit prediction calculator")
    crop_name = input("enter the name of crop: ")
    try:
        land_area = float(input("enter area of land: "))
        yield_per_acre = float(input("enter expected yeild per acre(kg): "))
        market_price = float(input("enter expected price of market per kg: "))
        cost_per_acre = float(input("enter total cultivation cost per acre: "))
    except ValueError:
        print("invalid input! enter numeric values for area, yield, price, and cost")
        return
    total_yield = land_area * yield_per_acre
    total_revenue = total_yield * market_price
    total_cost = land_area * cost_per_acre
    profit = total_revenue - total_cost
    print("crop profit prediction result")
    print(f"crop:{crop_name}")
    print(f"total expected yield: {total_yield:.2f}kg")
    print(f"total expected revenue: in rupees {total_revenue:.2f}")
    print(f"total cultivation cost: in rupees {total_cost:.2f}")
    if profit > 0:
        print(f"estimate profit: in rupees {profit:.2f}")
    elif profit == 0:
        print("no profit and no loss")
    else:
        print(f"estimated loss: in rupees{profit:.2f}")

if __name__ == "__main__":
    crop_profit_calculator()

```

Output:

```
crop profit prediction calculator
enter the name of crop: Wheat
enter area of land: 30
enter expected yeild per acre(kg): 1000
enter expected price of market per kg: 10
enter total cultivation cost per acre: 1500
crop profit prediction result
crop:Wheat
total expected yield: 30000.00kg
total expected revenue: in rupees 300000.00
total cultivation cost: in rupees 45000.00
estimate profit: in rupees 255000.00
```

Testing Approach

We tested:

Input Validation Testing (invalid numbers, blank inputs)

Boundary Testing (very small/large values)

Functional Testing (different crop scenarios

Error Handling Testing (invalid entries)

Challenges Faced

Ensuring robust input validation

Ensuring output is clean and easy to interpret

Constructing a simple structure that works well.

Negative profit and zero profit scenarios to consider.

Learnings & Key Takeaways

- A better understanding of how Python input and error handling works
- Surface-level appreciation for clean UI/UX design, even in terminal applications
- Learned about simpler designs for small systems complete with proper requirements
- Experience modularizing code and documentation

Future Enhancements

- Add a GUI with Tkinter / PyQt
- Add a database to hold user predictions
- Including more variables, such as
  1) Fertilizer cost
  2) Weather impact
  3) Seed variety factors
- Add charts and graphs for an enhanced output
- Convert to a mobile application for farmers

References

- Python Official Documentation
- Web information and learning material around Crop Economics
- Basic measure tutorials around software design & architecture
- Farmer.gov.in for basic agricultural data