Grow A Garden – Final Prototype Evaluation Report

Project Description

Grow A Garden is an IoT-integrated automated sprinkler and monitoring system tailored for urban and rural gardeners who face time, labor, or technical limitations. The app is connected to a NodeMCU-based system with real-time soil monitoring and automated irrigation features. It addresses challenges such as water efficiency, plant health monitoring, and remote field management through a user-friendly mobile interface.

The app provides features like:
- Real-time data on soil moisture, temperature, and humidity
- Manual or automatic sprinkler control
- Adjustable environmental thresholds
- Visual dashboard and sensor-based automation

The target users include local farmers, home gardeners, and agriculture students looking for a reliable, low-maintenance smart gardening solution.

System Requirements

To ensure accessibility for both low-end and modern Android devices, the Grow A Garden application was designed with the following minimum and recommended specifications in mind:

Requirement Type	Specification
Compatibility	Android 7.0 and higher (Not compatible with iOS)
Required Storage	100 MB and higher
Memory (RAM)	2 GB or higher
Internet	Requires internet to access the software
Bluetooth	Not supported

These specifications ensure that the app runs smoothly even on low-end Android smartphones. The app was deliberately kept lightweight to maximize compatibility across a wider range of student and farmer devices.

Overview

To test the prototype, the team used online and on-site evaluation sessions with tools like Microsoft Forms and in-person observation. The evaluation process was split into three components:

Usability Specifications – Timed user tasks to measure efficiency and learnability
Heuristics Evaluation – Analysis of the UI design using standard usability principles
Participant Survey & Feedback – Post-task survey with Likert-scale and open-ended responses

Tasks for Evaluation

Participants completed the following tasks:
- Creating an account
- Logging in
- Viewing and interpreting sensor data
- Activating and testing automatic mode
- Adjusting moisture thresholds
- Submitting user feedback

Data Presentation

Usability Specifications

Task	Mean Time	Interpretation	Classification
Creating an Account	2m 10s	Highly Acceptable	Successful
Viewing Sensor Dashboard	1m 05s	Highly Acceptable	Successful
Activating Auto Mode	3m 45s	Acceptable	Successful
Adjusting Moisture Threshold	4m 25s	Moderately Acceptable	Neutral
Submitting Feedback	1m 30s	Highly Acceptable	Successful

Heuristic Evaluation

Most heuristics were satisfied with minor areas for improvement. Key findings:
- Visibility of System Status: Clear and real-time dashboard feedback.

- Match Between System and Real World: Easy to understand gardening terminology.
- User Control and Freedom: Users could undo or exit actions.
- Consistency and Standards: Some icon/button inconsistencies noted.
- Error Prevention: Better threshold validation needed.
- Recognition Rather Than Recall: Dashboard data always visible.
- Flexibility and Efficiency: Usable by novice and experienced users.
- Aesthetic and Minimalist Design: Clean, data-focused layout.
- Help Users Recover from Errors: Needs better plain-language error messages.
- Help and Documentation: Tooltips and guides present.

Participant Survey and Feedback

Average survey results:
- Overall Experience: 4.7 (Highly Acceptable)
- UI Design: 4.6 (Highly Acceptable)
- Task Completion Ease: 4.3 (Acceptable)
- Viewing Data: 4.6, Auto Mode: 4.1, Threshold Adjustment: 3.8, Feedback: 4.7, Navigation: 4.5

Feedback noted clear dashboard layout, but some confusion over threshold input and error prompts.

Design Implications

The prototype was successful but revealed areas needing refinement:
- Add input hints for threshold fields
- Improve error messages
- Ensure all interface elements are functional

No major flaws were discovered.

Critique and Summary

Advantages:
- Efficient feedback collection
- Strong dashboard and auto-mode reception

Disadvantages:
- Limited hardware integration during test
- UI inconsistencies in button placement

Future improvements:
- Separate pre- and post-evaluations
- Include live sensor support and more features like alerts

Final thoughts:
Grow A Garden was considered a success, balancing usability with purpose. Future iterations can further improve usability by addressing minor issues discovered during testing.