Work Breakdown Structure (WBS)

This Work Breakdown Structure (WBS) serves as a comprehensive decomposition of the total scope of the GrowHub project. Its primary purposes are to:

- Provide a clear, hierarchical representation of all work required to complete the project
- Break down complex project activities into smaller, more manageable components
- Facilitate accurate estimation of time, cost, and resources and to establish a foundation for project scheduling, budgeting, and risk management

Version: 001
Created: 7. 7. 2024 by Filip Vallo (Project Manager)
Approved: 7. 7. 2024

1. GENERAL PROJECT INFORMATION

Project name:
GrowHub

Project Manager: Filip Vallo
Email: filip.vallo@gmail.com

Expected start date: 8. 7. 2024
Expected completion date: 1. 9. 2024

2. PROJECT OVERVIEW

2.1 Description

GrowHub is an MVP IoT product designed to help indoor growers monitor and control their growing environments. The system will collect real-time data from various sensors and provide users with a web-based dashboard for monitoring and managing growing conditions. The product will include a set of fully programmable peripheral devices for irrigation scheduling, fertilizer injection, and power control. The dashboard will also allow to set alerts with notifications if any of the measured parameters exceeds set thresholds.

2.2 Purpose and Justification

The purpose of the GrowHub project is to develop an MVP IoT solution that provides hobbyists and professional growers with the tools to monitor and control their indoor growing environments. This solution aims to enhance plant growth by maintaining optimal conditions through automated and precise control of environmental factors.

2.3 Problem Statement

Indoor growers face challenges in maintaining optimal growing conditions for their plants. The lack of precise and automated control over environmental factors can lead to suboptimal plant growth and reduced yields.

2.4 Business Case

Developing the GrowHub MVP will demonstrate the feasibility of an IoT-based solution for indoor growing environments. This project has the potential to attract commercial interest, leading to further development and potential market entry. Successful implementation of the MVP will showcase the value of precise environmental control in improving plant health and yields.

2.5 Vision

Revolution of indoor growing through an innovative IoT solution that empowers both hobbyists and professionals to achieve optimal plant health and yields, by providing precise, automated environmental control and real-time monitoring.

2.6 Goals

- Manufacture a functional prototype of the monitoring IoT system using Raspberry Pi with integrated sensors to measure atmospheric, soil, and water conditions
- Manufacture connected programmable peripheral IoT devices for irrigation control, fertilizer injection, and power control
- Develop a fully operational website with dashboard for live and historical data visualization with notification system and environmental control system for managing growing conditions
- Manufacture embedded PCB that integrates all electronic circuits needed to process digital signals from all connected sensors and to control all custom IoT peripheral devices
- Test the MVP's commercial potential

2.7 Scope Statement

The GrowHub project encompasses the development, testing, and deployment of a fully functional MVP IoT system for indoor growing environment monitoring and management.

2.8 Functional Requirements

2.8.1 IoT device

- Real-time data collection from sensors connected to IoT compute unit: soil (moisture), atmospheric (temperature, humidity, CO2), and water (pH, conductivity)
- Fully programable peripheral IoT devices for irrigation control, fertilizer injection, and power control
- Embedded PCB with custom integrated electronic circuit that processes digital signals from all connected sensors and controls all peripheral devices

2.8.2 Web-based dashboard

- Real-time data collection from IoT device
- Historical data tracking and visualization
- Notification system for alerts when measured parameters exceed thresholds
- Environmental control system for managing growing conditions through peripheral devices

2.9 Non-functional Requirements

- High reliability and accuracy of sensor data
- Secure and responsive web dashboard
- User-friendly interface for both novice and experienced growers
- Scalability to handle multiple users and devices in the future

2.10 Outside of Scope

- Full commercialization and mass production (beyond MVP)
- User registration/authentication and user management system
- Advanced features not essential for the MVP
- Casing and product design

2.11 Risks

- Technical challenges in integrating sensors and developing the dashboard
- Limited budget and resources
- Delays in hardware procurement
- Delays in custom PCB manufacturing from the external vendor
- Quality issues with the manufactured PCB

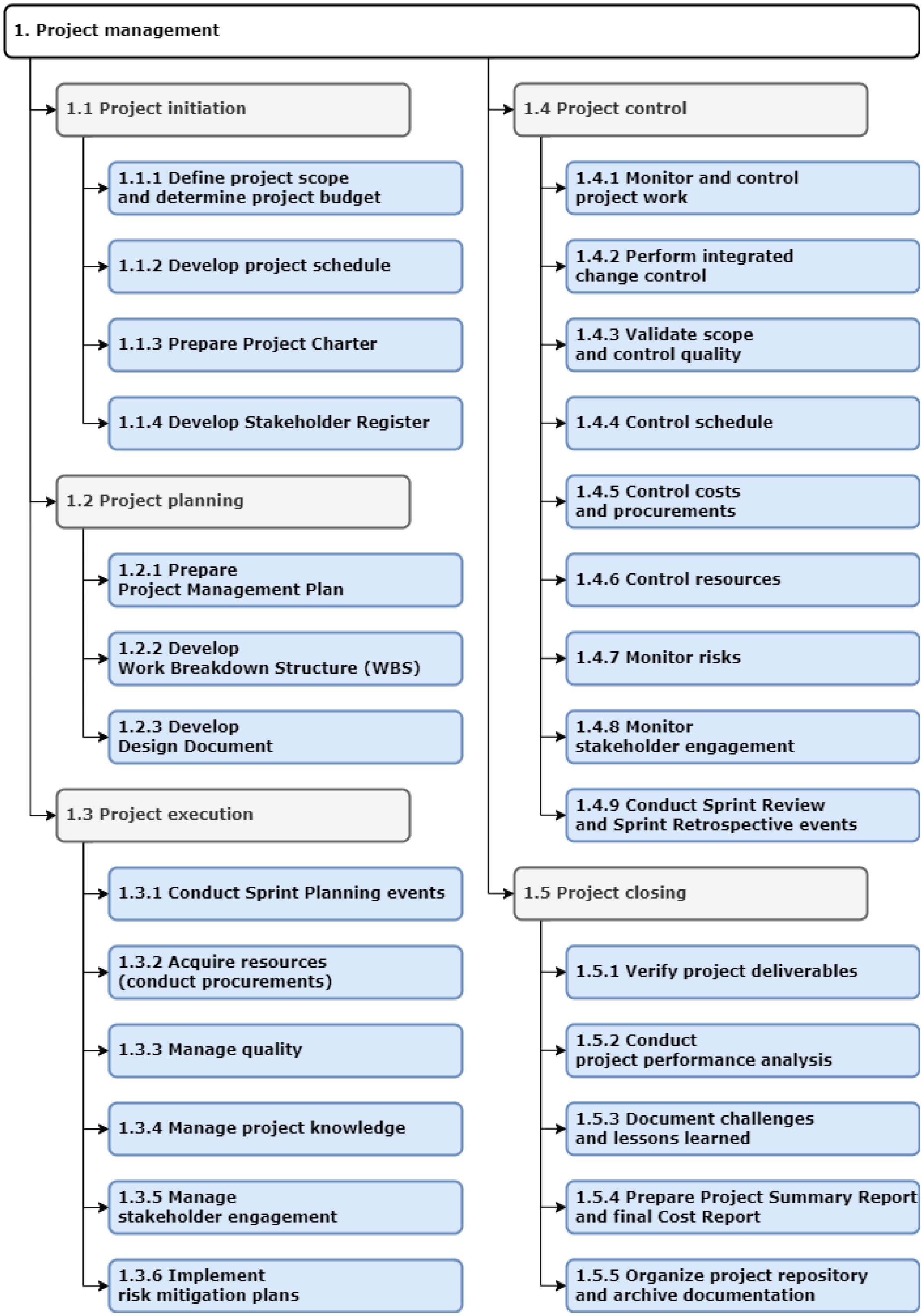
2.12 Constraints

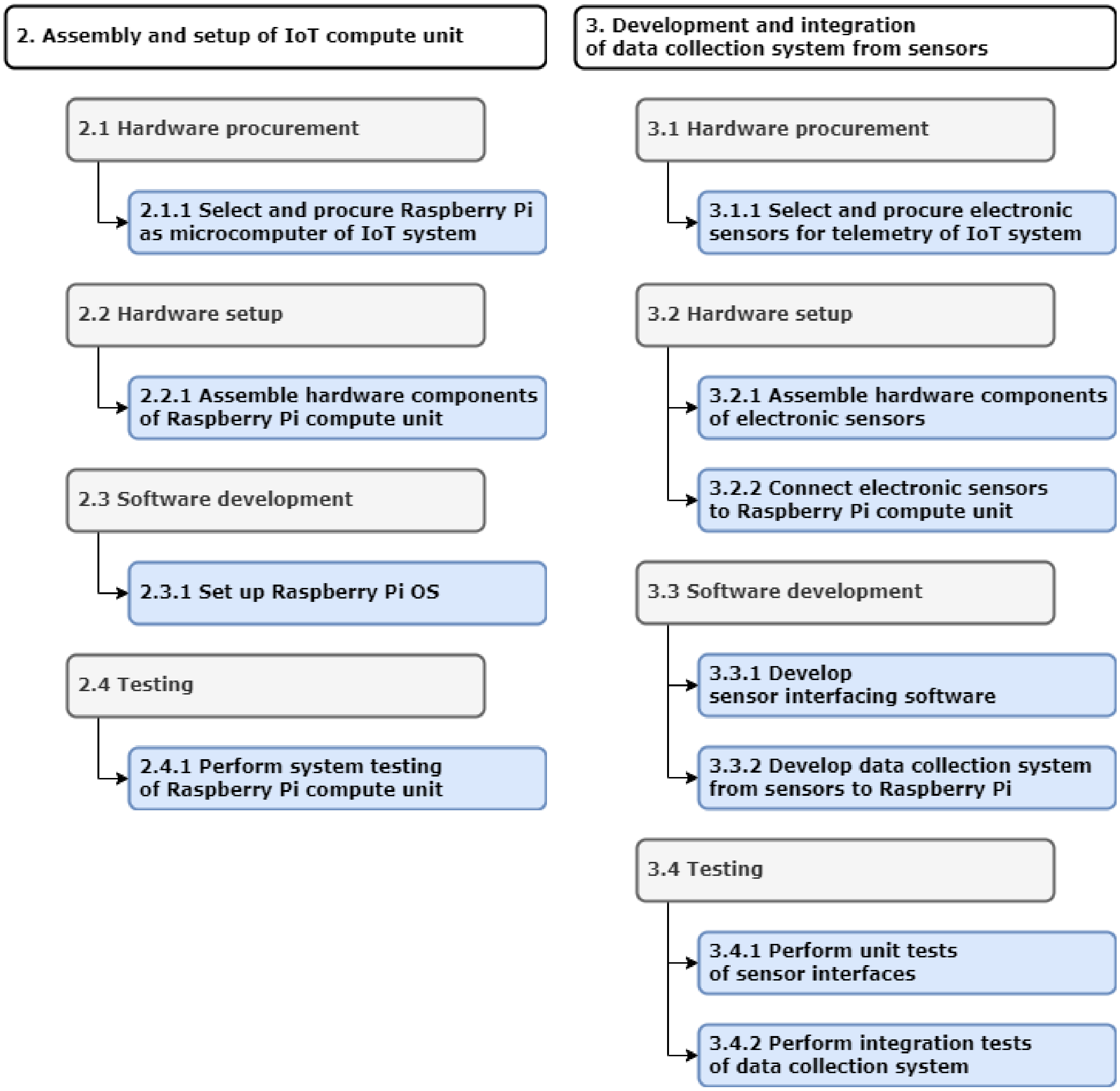
- Limited budget for hardware procurement and development
- Tight timeline to complete the MVP within the four planned phases
- Dependence on third-party services for cloud hosting and data storage
- Dependence on external vendor timelines for custom PCB manufacturing

2.13 Assumptions

- Availability of necessary hardware components
- Access to development tools and platforms for building the website
- The external vendor will meet the specified quality and timeline requirements for the custom PCB circuit
- Initial funding is sufficient to develop the MVP

3. WBS DIAGRAM


1. Project management
1.1 Project initiation
1.1.1 Define project scope and determine project budget
1.1.2 Develop project schedule
1.1.3 Prepare Project Charter
1.1.4 Develop Stakeholder Register
1.2 Project planning
1.2.1 Prepare Project Management Plan
1.2.2 Develop Work Breakdown Structure (WBS)
1.2.3 Develop Design Document
1.3 Project execution
1.3.1 Conduct Sprint Planning events
1.3.2 Acquire resources (conduct procurements)
1.3.3 Manage quality
1.3.4 Manage project knowledge
1.3.5 Manage stakeholder engagement
1.3.6 Implement risk mitigation plans
1.4 Project control
1.4.1 Monitor and control project work
1.4.2 Perform integrated change control
1.4.3 Validate scope and control quality
1.4.4 Control schedule
1.4.5 Control costs and procurements
1.4.6 Control resources
1.4.7 Monitor risks
1.4.8 Monitor stakeholder engagement
1.4.9 Conduct Sprint Review and Sprint Retrospective events
1.5 Project closing
1.5.1 Verify project deliverables
1.5.2 Conduct project performance analysis
1.5.3 Document challenges and lessons learned
1.5.4 Prepare Project Summary Report and final Cost Report
1.5.5 Organize project repository and archive documentation

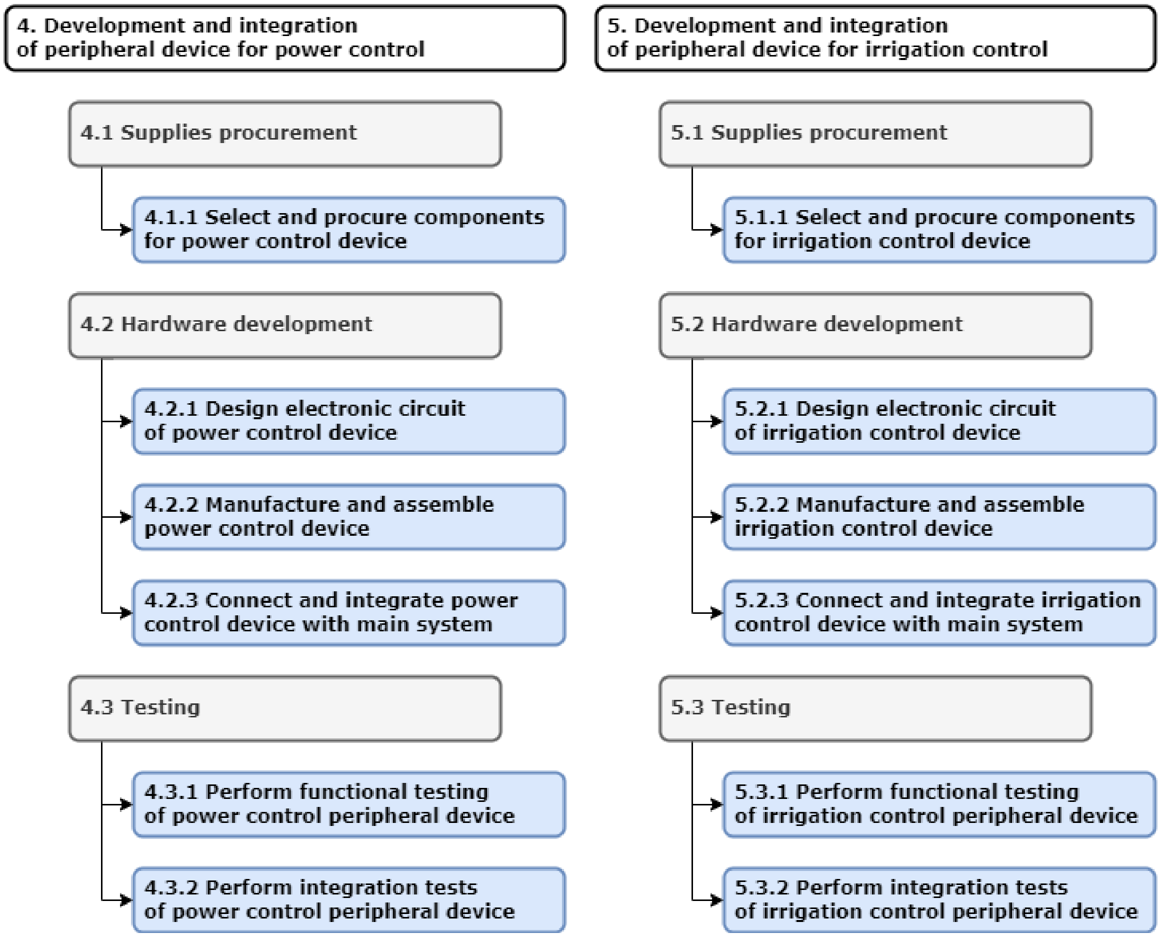


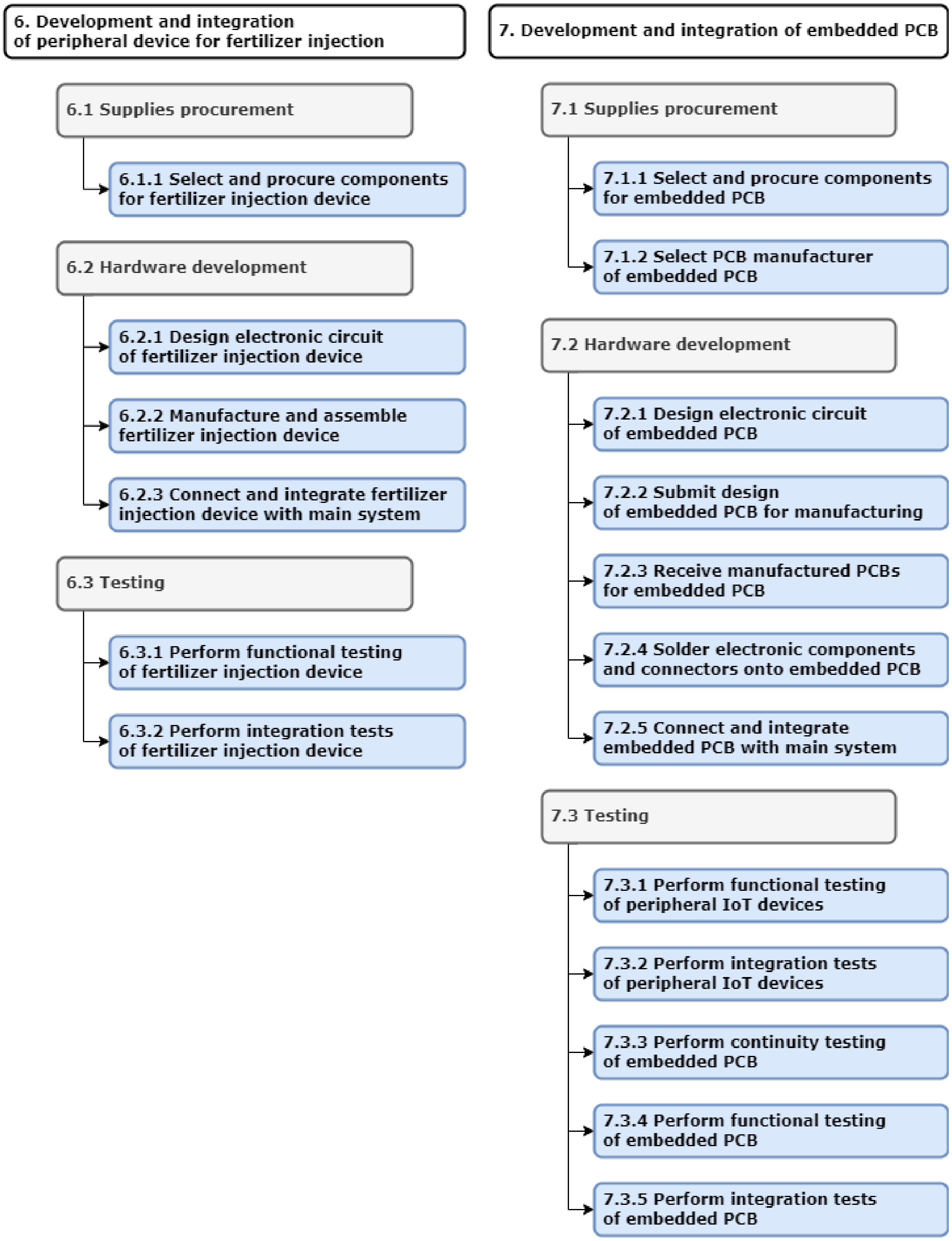
2. Assembly and setup of IoT compute unit

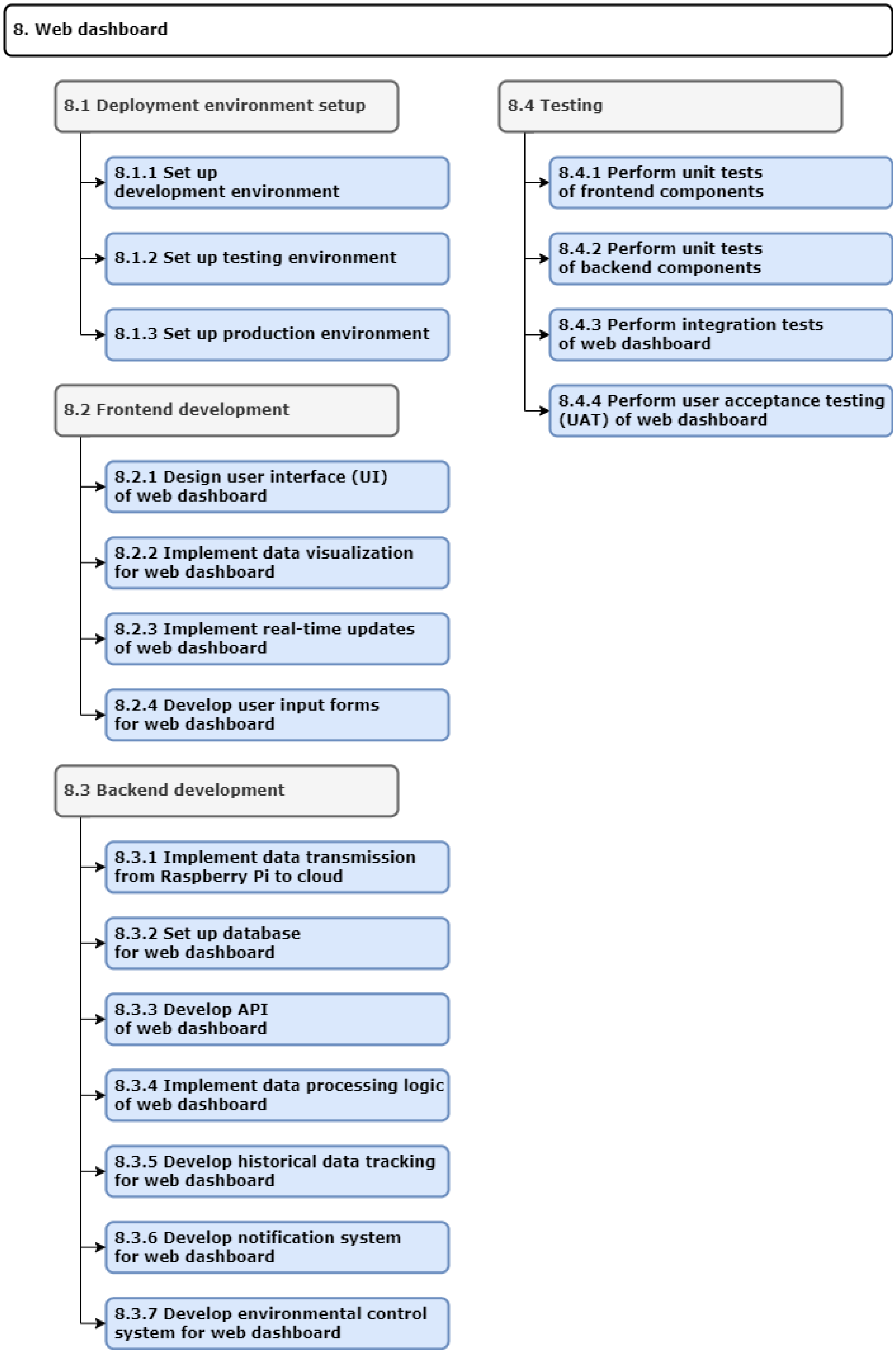
- 2.1 Hardware procurement
  - 2.1.1 Select and procure Raspberry Pi as microcomputer of IoT system
- 2.2 Hardware setup
  - 2.2.1 Assemble hardware components of Raspberry Pi compute unit
- 2.3 Software development
  - 2.3.1 Set up Raspberry Pi OS
- 2.4 Testing
  - 2.4.1 Perform system testing of Raspberry Pi compute unit

3. Development and integration of data collection system from sensors

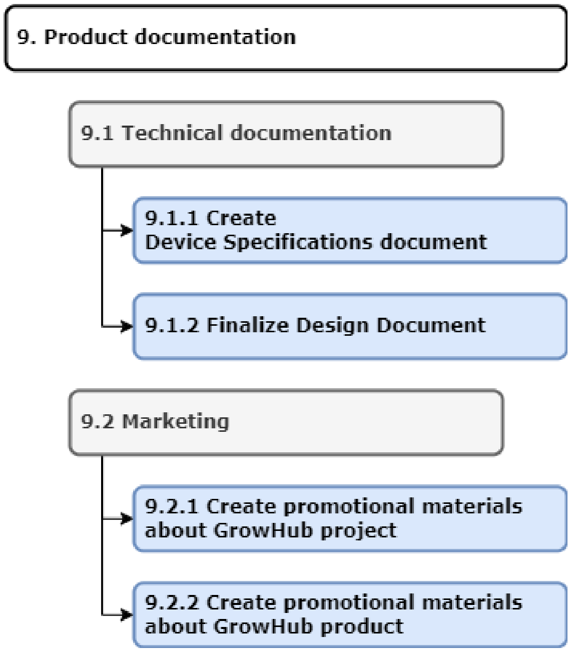
- 3.1 Hardware procurement
  - 3.1.1 Select and procure electronic sensors for telemetry of IoT system
- 3.2 Hardware setup
  - 3.2.1 Assemble hardware components of electronic sensors
  - 3.2.2 Connect electronic sensors to Raspberry Pi compute unit
- 3.3 Software development
  - 3.3.1 Develop sensor interfacing software
  - 3.3.2 Develop data collection system from sensors to Raspberry Pi
- 3.4 Testing
  - 3.4.1 Perform unit tests of sensor interfaces
  - 3.4.2 Perform integration tests of data collection system


4. Development and integration of peripheral device for power control
4.1 Supplies procurement
4.1.1 Select and procure components for power control device
4.2 Hardware development
4.2.1 Design electronic circuit of power control device
4.2.2 Manufacture and assemble power control device
4.2.3 Connect and integrate power control device with main system
4.3 Testing
4.3.1 Perform functional testing of power control peripheral device
4.3.2 Perform integration tests of power control peripheral device
5. Development and integration of peripheral device for irrigation control
5.1 Supplies procurement
5.1.1 Select and procure components for irrigation control device
5.2 Hardware development
5.2.1 Design electronic circuit of irrigation control device
5.2.2 Manufacture and assemble irrigation control device
5.2.3 Connect and integrate irrigation control device with main system
5.3 Testing
5.3.1 Perform functional testing of irrigation control peripheral device
5.3.2 Perform integration tests of irrigation control peripheral device


6. Development and integration of peripheral device for fertilizer injection
6.1 Supplies procurement
6.1.1 Select and procure components for fertilizer injection device
6.2 Hardware development
6.2.1 Design electronic circuit of fertilizer injection device
6.2.2 Manufacture and assemble fertilizer injection device
6.2.3 Connect and integrate fertilizer injection device with main system
6.3 Testing
6.3.1 Perform functional testing of fertilizer injection device
6.3.2 Perform integration tests of fertilizer injection device
7. Development and integration of embedded PCB
7.1 Supplies procurement
7.1.1 Select and procure components for embedded PCB
7.1.2 Select PCB manufacturer of embedded PCB
7.2 Hardware development
7.2.1 Design electronic circuit of embedded PCB
7.2.2 Submit design of embedded PCB for manufacturing
7.2.3 Receive manufactured PCBs for embedded PCB
7.2.4 Solder electronic components and connectors onto embedded PCB
7.2.5 Connect and integrate embedded PCB with main system
7.3 Testing
7.3.1 Perform functional testing of peripheral IoT devices
7.3.2 Perform integration tests of peripheral IoT devices
7.3.3 Perform continuity testing of embedded PCB
7.3.4 Perform functional testing of embedded PCB
7.3.5 Perform integration tests of embedded PCB



- 8. Web dashboard
  - 8.1 Deployment environment setup
    - 8.1.1 Set up development environment
    - 8.1.2 Set up testing environment
    - 8.1.3 Set up production environment
  - 8.2 Frontend development
    - 8.2.1 Design user interface (UI) of web dashboard
    - 8.2.2 Implement data visualization for web dashboard
    - 8.2.3 Implement real-time updates of web dashboard
    - 8.2.4 Develop user input forms for web dashboard
  - 8.3 Backend development
    - 8.3.1 Implement data transmission from Raspberry Pi to cloud
    - 8.3.2 Set up database for web dashboard
    - 8.3.3 Develop API of web dashboard
    - 8.3.4 Implement data processing logic of web dashboard
    - 8.3.5 Develop historical data tracking for web dashboard
    - 8.3.6 Develop notification system for web dashboard
    - 8.3.7 Develop environmental control system for web dashboard
  - 8.4 Testing
    - 8.4.1 Perform unit tests of frontend components
    - 8.4.2 Perform unit tests of backend components
    - 8.4.3 Perform integration tests of web dashboard
    - 8.4.4 Perform user acceptance testing (UAT) of web dashboard


9. Product documentation
9.1 Technical documentation
9.1.1 Create Device Specifications document
9.1.2 Finalize Design Document
9.2 Marketing
9.2.1 Create promotional materials about GrowHub project
9.2.2 Create promotional materials about GrowHub product

4. GLOSSARY OF TERMS

- **Application Programming Interface** (**API**): A set of protocols, routines, and tools for building software applications. An API specifies how software components should interact and allows different software systems to communicate with each other.
- **Configuration Item** (**CI**): A uniquely identifiable element within a configuration management system, which can be managed and controlled through the configuration management process. CIs can be components of hardware, software, documentation, or any other essential part of a project that requires management to ensure consistency and quality throughout its lifecycle.
- **Daily Scrum** (Daily Stand-up): Formal short daily event of Scrum framework (typically 15 minutes) held every day of the Sprint. It is a key event in the Scrum framework where the Development Team synchronizes their work and plans for the next 24 hours.
- **Internet of Things** (**IoT**): The network of physical objects embedded with sensors, software, and other technologies that connect and exchange data with other devices and systems over the internet. These objects, or "things," can range from everyday household items to sophisticated industrial tools.
- **Minimum Viable Product** (**MVP**): The most basic version of a product that is still functional and usable by early adopters. It includes only the essential features needed to meet the initial user needs and provide feedback for future development.
- **Printed Circuit Board** (**PCB**) - Flat board made of insulating material, typically fiberglass, with conductive pathways or "traces" etched or printed onto it. These traces connect various electronic components such as resistors, capacitors, and integrated circuits, which are soldered onto the board.

- **Raspberry Pi**: A series of small single-board computers developed in the UK by the Raspberry Pi Foundation. It's widely used in IoT projects due to its low cost, modularity, and open architecture.
- **Sensor**: A device that detects and responds to some type of input from the physical environment. The output is generally a signal that is converted to human-readable display at the sensor location or transmitted electronically over a network for reading or further processing.
- **Scrum**: An agile framework for managing complex projects, most commonly used in software development. It is designed to help teams work together, encouraging collaboration, accountability, and iterative progress toward a well-defined goal. Scrum enables teams to deliver high-value products by dividing the work into short, manageable cycles called Sprints.
- **Sprint**: Fixed time-boxed period (usually 1-4 weeks) of Scrum framework during which a Scrum team works to complete a specific set of work items from the Product Backlog. Each Sprint is a mini-project aimed at creating a usable and potentially releasable increment of the product.
- **Sprint Planning**: Formal event of Scrum framework that kicks off the Sprint. During this meeting, the team defines the work to be completed during the upcoming Sprint.
- **Sprint Retrospective**: Formal event of Scrum framework held after the Sprint Review and before the next Sprint Planning, where the Scrum team reflects on the past Sprint to identify improvements for future Sprints.
- **Sprint Review**: Formal event of Scrum framework held at the end of each Sprint, where the Scrum team and stakeholders inspect the increment of the product developed during the Sprint and discuss feedback and future work.
- **User Interface** (**UI**): The point of interaction between a user and a computer system, typically through a set of visual elements like buttons, menus, and icons. It encompasses both the visual design and the interactive elements of a software application.