Plan and Design: Final Check List

This checklist outlines the criteria I will use to review your Plan and Design section.

Please ensure that your work clearly addresses **each point** before submitting.

1. **Purpose of the System**:

   - Explain **what is the purpose** of your system and what **environmental issue** it addresses.

2. **Design Objectives:**

Design Objectives: *A* **list** of **things** your **system must** be **able to do**, one by one. Each **objective** should describe a specific action your system will perform. It should begin: *'I will …..*

● **Important:** Design Objectives must be specific to **your own risk model**. Generic or copied data/examples **will not** be accepted.

I will use environmental input data from **[sensor(s) used]** and produce **[output used]** on the Micro:bit, operating automatically once started.

I will collect and store environmental data related to **[specific forest condition]** in a **[file format]** file for later analysis.

I will simulate the forest-related risk or process **[name your risk/process]** over time by changing **[what value changes]** as conditions **[describe how they persist or change]**.

I will use Python to model changes in **[your chosen risk]** using data from **[your embedded system inputs]** and **[any simulated or external data you chose, if applicable]**.

I will test the model using what-if scenarios by changing **[specific variable(s)]** (e.g. **[your example]**) and observing how **[model output / risk level]** changes.

I will ensure the model updates automatically by **[describe how updating occurs]**, with outputs **[describe what changes]** as new data is read.

3. **Project Options:** Describe **two different build options:** Here are some possible examples:

● **Important:** Options must relate to **your own project**.

Data collection approach:	Sensor setup (single sensor vs multiple sensors)
	Data logging method (continuous logging vs threshold-based/event logging)
Modelling approach:	Rules-based model vs decision-tree style logic
Risk scoring method:	Points-based scoring vs weighted scoring
Feedback / output method:	Digital output (icons, numbers, text) vs analogue output (sound pitch, brightness)
Where data is processed	On the embedded device (micro:bit) vs in Python on a computer

For each option

Explain briefly **how the option would work** (project-specific)
Identify **one advantage**
Identify **one limitation**

5. Justification of Design Choice

● **Important:** Justification must be made on **your own project**.

Clearly state **which option was chosen**
Explain **why it better meets the Design Objectives**
Explain **why it improves modelling or simulation over time**
Refer to **one practical benefit** (e.g. clarity, reliability, manageable data)

6. Stakeholders: Stakeholders are people or groups who **benefit from or are affected by the system**, EVEN if they **do not use** it directly.

Examples: Environmental or conservation organisations - Teachers - Community groups - **STEM** education

For stakeholders, briefly state: **what** they **need**
why they **need** it
how your system helps **meet** that **need**

End Users: End users are the people who **directly use or operate the system**.

Examples: Student – Teacher – Learning testing 'what-if' scenarios.

For end users, explain: **what** they need from the system
how your system supports them through clear **feedback** and/or **modelling**

Note: In this project, all end users **are** stakeholders, but **not all** stakeholders directly use the system.

7. Technologies:

Embedded system: Micro:bit with environmental sensors and output devices.
Software: MakeCode / MicroPython for Micro:bit programming. Python for data analysis, modelling, and what-if testing.
Data storage: Environmental data stored in CSV format.
Modelling: Simple rules-based risk model implemented in Python.

8. System Architecture: Flowcharts {Ongoing}

Note 1: Flowcharts can't be completed until you've completed the 'build'.

Note 2: **Once we are working on our model, I will then address what you specifically need for:**

- **Flowcharts**
- **Testing**
- **Logs**
- **Problem**
- **Important Piece of Code**