

You and your partner need to define the basis of a project according to these criteria. Be brief, but provide enough detail so I can make sense of what you're proposing.

1. Present the problem statement. For example, an average homeowner designing a new house wants to play with options for installing solar panels.

On most job sites PPE (personal protective equipment) is required by law or at least highly encouraged, due to the risk and danger of the work environment. Many accidents are easily avoidable by simply wearing appropriate PPE devices, but that doesn't guarantee the employees will wear them.

2. Explain who the intended user is. It can be you or a fictitious entity you will represent. For example, the user is a homeowner building a new house.

The intended users for our system would be employers and government agencies.

3. Describe why the user has this problem. For example, there are decisions to make and many options, which are difficult to manage for the average user.

Employers do not always enforce PPE usage on their employees or the employees themselves choose to not wear PPE while on the job.

4. Describe how a solution would benefit the user. For example, maximizing the solar generation while minimizing the cost saves the homeowner money. Stay within the scope of the problem; e.g., do not consider climate change because the connection is too indirect.

This will protect negligent employers from possible OSHA violations and workplace misconduct while keeping the employees safe and reducing injuries.

5. Describe the general flow for addressing the problem. The existing (or imagined) flow does not have to involve a computation solution. For example, the user defines the property and house layout and expected energy needs, then the system proposes solutions that best satisfy the criteria. This used to be done on paper by expensive expert contractors.

The device acts as a safety net over the employees to protect them by detecting and identifying faults and notifying the correct authority.

6. What is the general nature of the solution? For example, app, standalone program, website, plug-in.

A cellphone sized box with a camera and an I/O cable for power and network connection.

7. List the general software components you envision playing a role. For example, web server, database, game engine.

The program will be built using Python, Rust, Javascript, HTML, React JS, GPT3, Tensorflow/OpenCV, YOLO

8. List the general hardware components you envision playing a role. For example, drone, VR headsets, and tablets. You are responsible for your own hardware, so be reasonable.

Raspberry PI4 Model B, Camera(Specs TBD), Housing box, physical Connor server

9. Describe similar solutions, if any, and justify (or make up a justification) for why they are inadequate. For example, Project XYZ does something similar, but its cost and complexity are prohibitive. Do not get detailed with software engineering aspects. Requirements and specifications, for example, come later in the process, unless they are directly relevant to the proposal and have a justification. For example, an Android app because you have an Android device and want to become an app developer

The current alternative is to employ a 'PPE lifeguard' to watch over the job site and enforce employees to wear gear. This requires labor, time, resources, and constant vigilance, not to mention the unreliability of it if he makes friends. Having a machine watch over them would eliminate all of these human faults.

Part 1: Project Description

Develop a common vision for what your project probably looks like. Basically, revise the description so someone not on the team could fully understand what it's about from this description alone without you explaining it in person. Keep it general still, but make sure to address everything it's supposed to be about for Part 2.

To do: Write a detailed paragraph description (minimum) that fully captures the vision of the project.

We envision the product to be a self-sustaining automated safety system that will help ensure the safety of the employees on any site by keeping track if they are wearing PPE at all times.. Simply install the box at a vantage point, and the AI learning software installed inside will use a camera to identify any workers that are not wearing appropriate PPE. Once detected, a video or screenshot will be uploaded to a phone or email alerting the supervisor of the employee, allowing them to take action as they wish.

Part 2: User Stories Based on the project description, add the initial user stories of this form:

As a [type of user], I want [an action] so that [a benefit/a value results]

Make sure there's bidirectional consistency between the description and the stories. In other words, everything in the description must have one or more corresponding stories, and every story refers to something in the description. Include every user story you think applies to the project as you defined it in the description. This could be a large list. Don't filter at this point. If you think something is relevant, include it, even if it's something you expect to be fringe or possibly dropped. We'll adjust the scope after seeing what results from this.

To do: Write comprehensive user stories. You can refer to the format shared earlier.

As a **construction business/consultant** I want a **detection device** so that **our company(s) don't get sued or fined for employees not wearing PPE**.

- Detection of violations of PPE rules.
- Records of these violations, whether that be videos, descriptions, or notifications.
- Low cost for this system.
- Easy setup for these systems.
- Minimum knowledge of computer systems to get it running.
- Reliability of these systems

Part 3: Glossary

For both parts, be sure everything is defined somehow. As a team, go through each word of substance and ask yourselves whether a developer outside the project would know what it means in this context. It's really easy to overlook this already because you have some idea of where you're going and start making assumptions. But without defining these assumptions, expectations, etc., they take on a life of their own as the development process continues. This is where a lot of the problems start to creep in.

To do: Develop a glossary of terms and Assumption

PPE: Personal Protective Equipment.

Raspberry Pi: Microcontroller/Microcomputer that is the main circuit board brain that will process and calculate everything.

Camera: Device that allows the Pi to see visual data.

Housing Box: 3d printed or otherwise container/case for the Pi and the camera mount.

Server: Additional computer that the Pi will send data to, to allow the data to be seen and processed outside of the Pi itself.

I/O: Input/Output to and from the Pi.

OSHA: Occupational Safety and Health Administration. Agency that oversees worker safety.

AI/Machine Learning: Mathematical algorithm that is utilized in order to process camera data and analyze it to determine if any faults have occurred.

YOLO/OpenCV/TensorFlow: Open source libraries for computer vision.

Python/Rust/JavaScript/HTML/React: Programming languages for development of application and website.

GPT3: AI library for data generation and lookup.