1.1 Problem Statement

What problem is your project trying to solve? Use non-technical jargon as much as possible.

To create a realtime video compression and decompression system while keeping minimal loss in order to keep memory to a minimum.

1.2 Requirements & Constraints

List all requirements for your project . This includes functional requirements (specification), resource requirements, qualitative aesthetics requirements, economic/market requirements, environmental requirements, UI requirements, performance requirements, legal requirements, maintainability requirements, testing requirements and any others relevant to your project. When a requirement is also a quantitative constraint, either separate it into a list of constraints, or annotate at the end of requirement as "**(constraint)**". Other requirements can be a single list or can be broken out into multiple lists based on the category.

- Part 1: Software Implementation
    - Benchmark tradeoffs for various lightweight compression algorithms in Python
        - Considerations:
        - Latency VS Compression Amount
        - How much data is lost in each lossy compression algorithm?
- Part 2: Hardware Implementation
    - Minimize cost and complexity of FPGA
    - Maintain near zero latency of compression and decompression
    - The FPGA should be fully-pipelineable
- Part 3: Demo
    - Have a video (prior to compression) being streamed and an additional video being streamed from the FPGA showcasing video after being compressed and decompressed

1.3 Engineering Standards

What Engineering standards are likely to apply to your project? Some standards might be built into your requirements (Use 802.11 ac wifi standard) and many others might fall out of design. For each standard listed, also provide a brief justification.

HDMI for transmitting the video from input and to output.
Compression algorithms such as M-JPEG, MPEG-4, and H. 264 for the video compression systems.
python for basic prototyping of the compression algorithms, in order to find the best compression system for the job.

1.4 Intended Users and Uses

Who benefits from the results of your project? Who cares that it exists? How will they use it? Enumerating as many "use cases" as possible also helps you make sure that your requirements are complete (each use case may give rise to its own set of requirements).

John Deere will use this project inside their equipment to reduce the cost they have to spend on memory. John Deere wants to use the system to make their computer vision systems more efficient. Reducing the memory usage could also have the factor of reducing processing time of their machine learning algorithms.