

[Return to "Intel® Edge AI for IoT Developers Nanodegree" in the classroom](#)

[DISCUSS ON STUDENT HUB](#)

Smart Queue Monitoring System

REVIEW

CODE REVIEW

HISTORY

Meets Specifications

Overall this is an excellent work!

Proposal



For each of the three scenarios, the Client Requirements and Potential Hardware Solution section is complete and does the following:

- Describes at least two relevant client requirements
- Indicates which hardware might be most appropriate for the client
- Explains how the chosen hardware would meet these requirements

You clearly clarify client requirements and provide appropriate hardware solutions.



For each of the three scenarios, the Queue Monitoring Requirements section includes the following information:

1. Maximum number of people in the queue (before the system would redirect them to another queue)

2. Correct model precision (FP32/FP16/ Int8) for the proposed hardware type

You correctly select all the model precisions!



For each of the three scenarios, the Test Results section of the proposal document includes the following graphs comparing all four hardware types:

- Model Load Time
- Inference Time
- Frames per Second (FPS)

For each of the three scenarios, the Final Hardware Recommendation section of the proposal document:

- Indicates the correct hardware type
- Uses both the client requirements and performance test results to justify the hardware choice

You make good final hardware recommendations.

Testing the Hardware



All methods in the `PersonDetect` class in the `person_detect.py` script are completed and functional:

- `load_model`
- `predict`
- `draw_outputs`
- `preprocess_outputs`
- `preprocess_inputs`

The `queue_job.sh` submission script is completed and functional.

The project runs models and performs inference on the following edge devices:

- CPU
- IGPU
- FPGA
- VPU

The submission also includes the output files generated by DevCloud after successful completion of inference jobs on DevCloud. These output files include the following:

- Output video with bounding boxes drawn on the video
- Stats file with the following stats:
 - Inference time
 - Model load time
 - Frames per Second (FPS)

Good job!



The Jupyter notebook includes the `qsub` command statements, which are used with the correct arguments to run the model on all four hardware devices.



[DOWNLOAD PROJECT](#)

[RETURN TO PATH](#)