Choose the Right Hardware

Proposal Template

Scenario 1: Manufacturing

Client Requirements and Potential Hardware Solution

Look through the scenario and find any relevant client requirements. Then, suggest a potential hardware type and explain how this hardware would satisfy each of the requirements.

Which hardware might be most appropriate for this scenario? **(CPU / IGPU / VPU / FPGA)**
FPGA

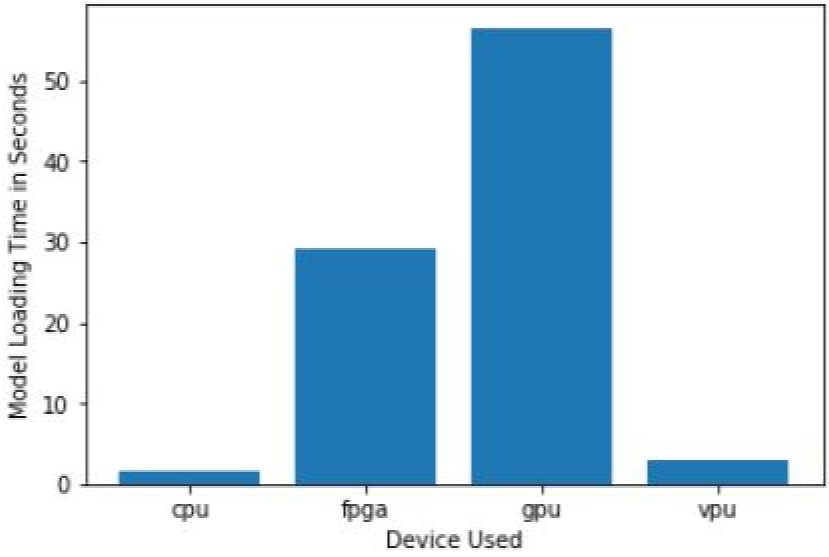
Requirement Observed **(Include at least two.)**	**How does the chosen hardware meet this requirement?**
Client is looking to run the system 24 hours a day	FPGA has100% on-time performance. i.e. it can be continuously run 24 hours a day, 7 days a week, 365 days a year
System should detect chip flaws without slowing down the packaging process and run inference on the video stream very quickly	FPGA can execute neural networks with high performance and very little latency because of running many sections in parallel and everything run inside FPGA without going off-chip. It also supports various precision - FP16, 11, 9. This allows developer to balance between speed and accuracy
System would also need to be flexible so that it can be reprogrammed and optimized to quickly detect flaws in different chip designs	FPGA is naturally flexible since it is field-programmable and bitstreams can be updated without modifying hardware.
Client has plenty of revenue to install a quality system	FPGA cost is quite higher than other devices. But it is fine since client prefer high quality system
Client would ideally like it to last for at least 5-10 years	FPGA has a long life-span. IOT Group has a guaranteed availability of 10 years, from start of production

Queue Monitoring Requirements

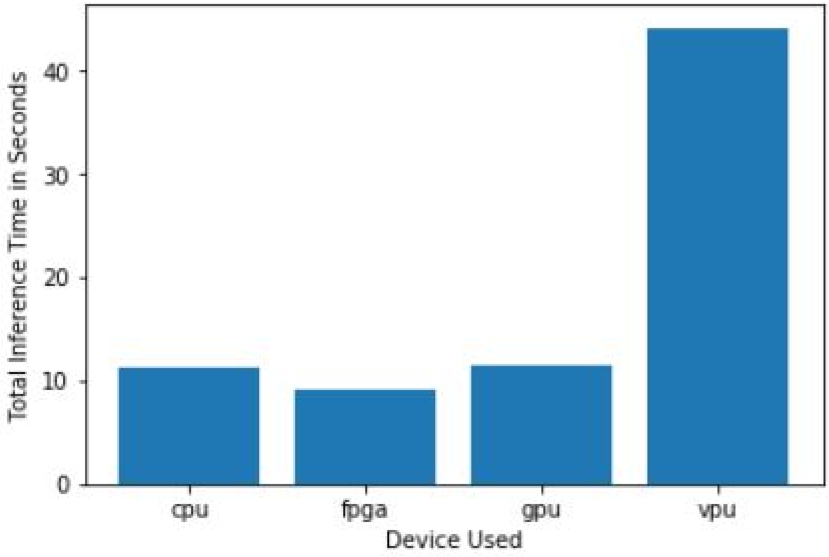
Maximum number of people in the queue	5-7
Model precision chosen (FP32, FP16, or Int8)	FP16

Test Results

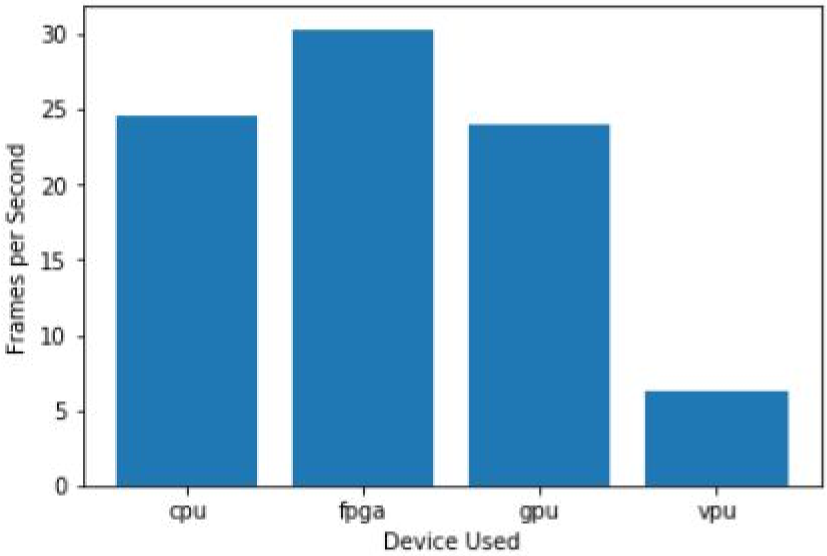
After you've tested your application on all four hardware types (CPU, IGPU, VPU, and FPGA), copy the matplotlib output showing the comparison into the spaces below. You should have three graphs (for model load time, inference time, and FPS).



Model Load Time



Inference Time



FPS

Final Hardware Recommendation

Now synthesize your points from above and provide a brief write-up describing why the chosen hardware is the best choice for this scenario. Be sure to discuss the client's requirements, the test results, and how these relate to one another (e.g., perhaps one of the devices performed better than the rest, but does not meet one of the client's requirements).

Write-up: Final Hardware Recommendation
FPGA gives best inference time and FPS compared to other devices, has potential to meet client requirement - 30-35 FPS FPGA supports 100% on-time performance, high performance & low latency, is flexible and has long life span

Scenario 2: Retail

Client Requirements and Potential Hardware Solution

Look through the scenario and find any relevant client requirements. Then, suggest a potential hardware type and explain how this hardware would satisfy each of the requirements.

Which hardware might be most appropriate for this scenario? (CPU / IGPU / VPU / FPGA)
CPU

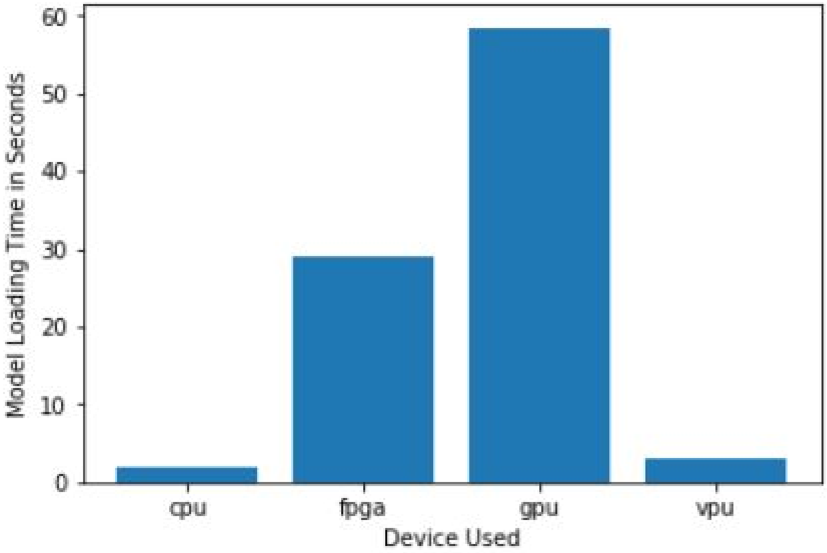
Requirement Observed (Include at least two.)	How does the chosen hardware meet this requirement?
Client stores checkout counters already have a modern computers, each of which has an Intel i7 core processor	Modern Intel i7 desktops have higher end configuration - minimum 4 Core. Can make use of existing resource
Existing computers are only used to carry out some minimal tasks that are not computationally expensive	Good to reuse the existing system since it only carries out minimal computational expensive tasks currently.
Client does not have much money to invest in additional hardware	Existing resources are utilized- no new hardware bought for this requirement. Hence, no additional cost
Client likes to save as much as possible on his electric bill	Electrical bill is saved because of utilizing available resources as well as CPU completes job as quickly as possible.

Queue Monitoring Requirements

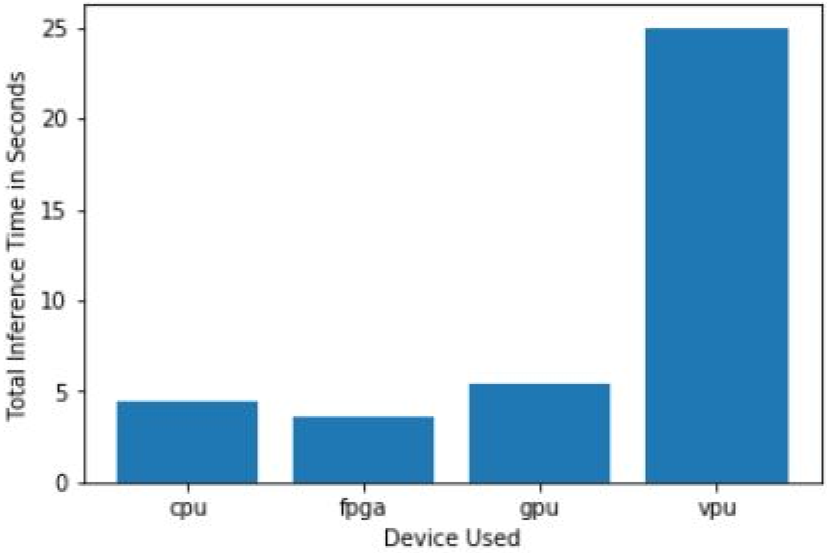
Maximum number of people in the queue	2-5
Model precision chosen (FP32, FP16, or Int8)	FP32

Test Results

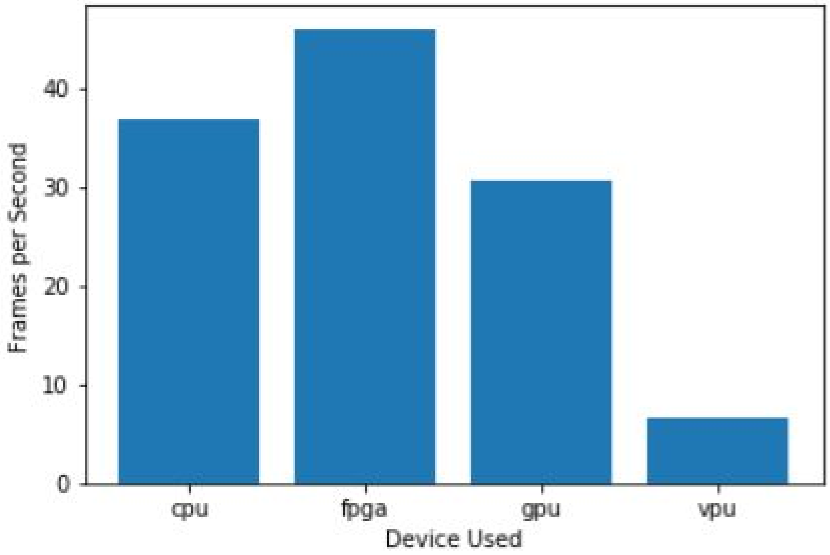
After you've tested your application on all four hardware types (CPU, IGPU, VPU, and FPGA), copy the matplotlib output showing the comparison into the spaces below. You should have three graphs (for model load time, inference time, and FPS).



Model Load Time



Inference Time



FPS

Final Hardware Recommendation

Now synthesize your points from above and provide a brief write-up describing why the chosen hardware is the best choice for this scenario. Be sure to discuss the client's requirements, the test results, and how these relate to one another (e.g., perhaps one of the devices performed better than the rest, but does not meet one of the client's requirements).

Write-up: Final Hardware Recommendation
CPU gives best model loading time, has relatively better inference time and FPS. It can effectively inference 2-5 people without any additional hardware and cost. CPU make use of existing client computers, thereby saving cost and electrical bill as much as possible. Intel i7 core processor has minimum 4 Core and supports hyperthreading. This feature helps to convert to minimum 8 virtual core and boost performance.

Scenario 3: Transportation

Client Requirements and Potential Hardware Solution

Look through the scenario and find any relevant client requirements. Then, suggest a potential hardware type and explain how this hardware would satisfy each of the requirements.

Which hardware might be most appropriate for this scenario? (CPU / IGPU / VPU / FPGA)
VPU

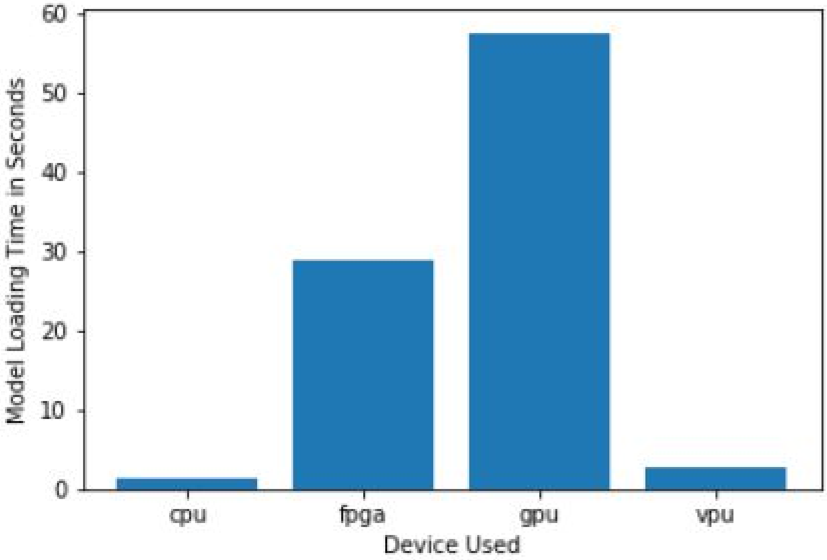
Requirement Observed (Include at least two.)	How does the chosen hardware meet this requirement?
The Client monitors the entire situation with 7 CCTV cameras on the platform. These are connected to closed All-In-One PCs that are located in a nearby security booth	NCS2 is USB3.1 plug and play removable VPU. supports all of the operating systems. Client already has PCs for security purposes. NCS2 is very much compatible to plug-in.
The CPUs in these machines are currently being used to process and view CCTV footage for security purposes and no significant additional processing power is available to run inference	NCS2 offers pre-trained models to be run on the stick with integration of OpenVino Toolkit. So It takes care of running inference in client PCs.
Client's budget allows for a maximum of $300 per machine	NCS2 is an inexpensive option compared to other devices, costing around $70 to $100, which meets client's budget
The Client would like to save as much as possible both on hardware and future power requirements.	NCS2 is meant to be a low-power device so that it can be easily deployed at the edge

Queue Monitoring Requirements

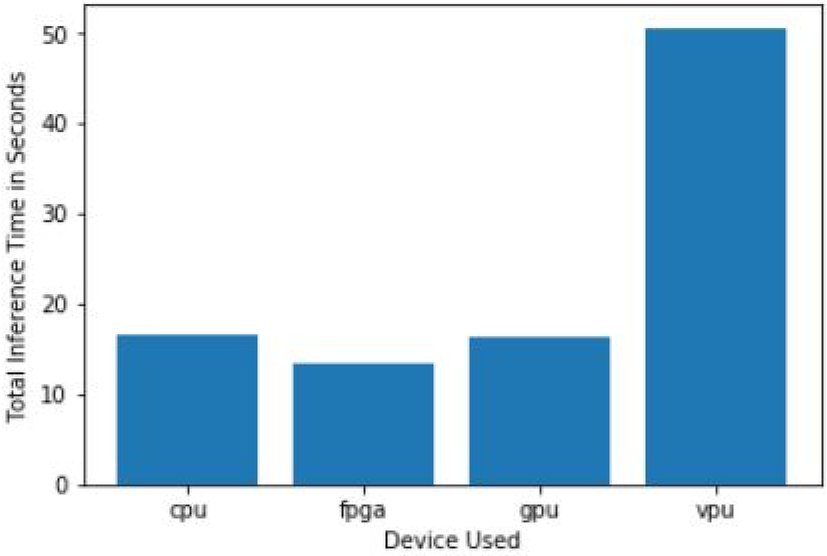
Maximum number of people in the queue	7-15
Model precision chosen (FP32, FP16, or Int8)	FP16

Test Results

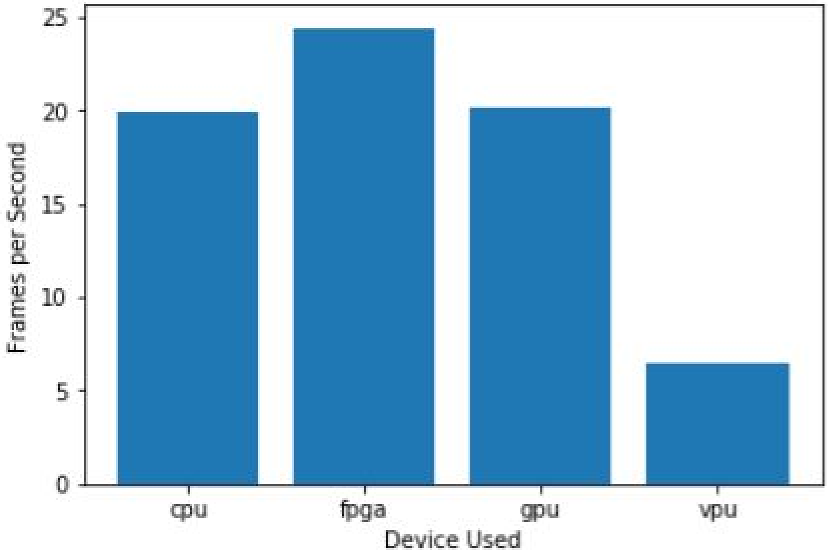
After you've tested your application on all four hardware types (CPU, IGPU, VPU, and FPGA), copy the matplotlib output showing the comparison into the spaces below. You should have three graphs (for model load time, inference time, and FPS).



Model Load Time



Inference Time



FPS

Final Hardware Recommendation

Now synthesize your points from above and provide a brief write-up describing why the chosen hardware is the best choice for this scenario. Be sure to discuss the client's requirements, the test results, and how these relate to one another (e.g., perhaps one of the devices performed better than the rest, but does not meet one of the client's requirements).

Write-up: Final Hardware Recommendation
VPU is an inexpensive, low power consuming device and can handle 7-15 people that satisfies the client's requirement and budget. Though VPU gives relatively poor inference time and FPS, it can be overcome by adding 2-3 NCS2 sticks thereby running multiple inferences in parallel. This would provide adequate performance without comprising the cost. VPU is pluggable, supports inference on pre-trained models, compatible with most of the operating systems, inexpensive and extremely low power consuming device.