Generate Image Descriptions

Background:
We're building a product where our clients can generate images for their websites and other marketing purposes. Often the prompts used for generation do not describe the image and objects in it correctly. We need good image descriptions for various purposes - search, categorization, etc.

Task:
Create a proof of concept service which, given an image, generates an image description.

Details:
- Input should be an image (jpg or other format) with a size up to 1024x1024.
- Output should be text - a one-paragraph description of the objects in the image in detail.
- The descriptions don't need to be perfect - they need to be good enough to serve the purposes of searching and categorization.
- The service can be a web application, console application or whatever is most convenient for you. It should not reload the model on each image description generation.
- The service should be run locally, do not send the image to external services like openAI.
- We do not have a labeled image dataset for evaluation.
- Code should be written in Python.
- Any open source libraries can be used.

Evaluation Criteria:
- Our main motivation is to evaluate the design, code quality, and analytical approach to the problem.
- The task is vague in order to give you options to focus on what you find interesting.
- Such a task can take much more time if each component is researched and developed in depth. That is why we expect a proof of concept but it's important to acknowledge what you focus on and sketch "next steps" for the things which are skipped. We want to see how thoroughly you explored the problem.
- As the actual criteria of the description are not well defined, we expect the provided solution to be flexible and allow easy change in the output format - for example add colors to the description, describe the composition, generate tags, etc.
- Code quality is important for us.

Things which will be considered a plus:
- Thorough research of possible solutions.
- Quality evaluation of the provided solution.
- Experiments with multiple algorithms or models.
- Performance evaluation.
- Usage of modern web app framework.
- Exposing the service in more than one way - cli, REST API, etc.