

# CISC 320 Project Outline

## CleanShare

Section 3 - Group 6

David Balan

Anthony Grecu

Charlie Guarasci

Liam Harper-Mccabe

Kieron Luke

Mark Nistor

Kayetan Protas

November 14, 2025

## General Purpose

CleanShare is a desktop application designed to automatically detect and blur alcoholic beverages in photos before they are shared. The purpose is to make images safe for posting on university-affiliated social media or sharing with family without manual editing, ensuring compliance with student guidelines and community standards.

## Intended Audience

- **Primary:** University students, club social media managers, and campus event organizers who need to share photos publicly while avoiding alcohol-related content.
- **Secondary:** High-school counselors, event photographers, and anyone who needs “safe-to-share” photos in a professional or family context.

## Features

- **Image Upload & Export:** Users can submit a single image (initially) and export the blurred result in the same dimensions as the original.
- **Automatic Detection & Blur:** The system uses OpenCV for object detection, enhanced with a CNN model for higher accuracy. Any detected alcohol container, whether by branding or shape, will be blurred.
- **Manual Adjustment Tools:** Users can refine blur regions with basic brush or rectangle tools.
- **Preview:** Side-by-side before/after comparison before saving.
- **Batch Processing (Stretch Goal):** Start with single-image support; expand to batch mode later in the project.
- **Privacy by Design:** All processing is done locally on the user’s machine so no cloud uploads.

## Data Handled

- **Input:** User-submitted photos in common formats (JPEG, PNG).
- **Processing Data:** Temporary metadata (bounding boxes, masks, CNN predictions) generated during blur operations.
- **Output:** Exported images with blurred alcohol content. No personal information is stored.

## Technical Approach

- **Platform Priority:** Windows-only for grading and development.
- **Tech Stack:** C++ with Qt for the GUI, OpenCV for image handling and baseline detection, and ONNX Runtime to integrate a bundled CNN model for robust alcohol detection.
- **Model:** Pre-trained CNN weights will be packaged with the repo for local inference.
- **Dataset:** Team will use a custom image set of various alcohol bottles to test detection and accuracy.

## Relevance & Impact

CleanShare reduces the time and effort required for manual redaction of alcohol in photos. It provides a practical tool for students, clubs, and organizations to share photos responsibly, while ensuring compliance with university and community policies.

## Motivation

Many student organizations face restrictions when posting event photos that display alcohol. Manually blurring drinks is not only time-consuming but also inconsistent and prone to error, often leading to compliance risks or damaged reputations. In today's digital-first environment, where student groups and individuals are judged by what they post online, the stakes are higher than ever. A single overlooked bottle in a photo could cause disciplinary action, loss of funding for clubs, or embarrassment in a professional or family setting.

By automating this process, CleanShare addresses a problem that is both practical and reputational. It saves organizations countless hours of tedious editing while creating a uniform standard of "safe-to-share" imagery. Beyond convenience, it actively protects students, clubs, and institutions from unnecessary risks. This is not simply a matter of efficiency; it is a matter of digital responsibility, reputation management, and creating an environment where people can confidently share their experiences without fear of unintended consequences.

## Functional Requirements

The functional requirements describe what CleanShare must provide for its users:

- The system must allow users to upload images in standard formats (JPEG, PNG).
- The system must detect and blur alcoholic beverages automatically using computer vision techniques.
- The system must offer a preview mode with a side-by-side comparison before saving.
- The system must allow exporting of the processed image in the same resolution as the original.
- The system must process images locally without requiring internet access or cloud uploads.

## Non-Functional Requirements

The non-functional requirements define the quality, performance, and constraints of the system:

- **Performance:** A 1080p photo should be processed in under 10 seconds on a typical Windows laptop.
- **Usability:** The workflow should be intuitive enough that a first-time user can blur and export an image in under 30 seconds.
- **Reliability:** The system should consistently detect at least 80% of alcohol containers while keeping false positives to a minimum.
- **Security:** All processing occurs locally; no personal data or images leave the user's machine.
- **Portability:** The initial release will target Windows; later versions may support other platforms.
- **Maintainability:** The system should be built with modular components (GUI, detection, processing) to simplify future improvements.

## Evaluation and Success Criteria

We will evaluate CleanShare through a custom dataset of alcohol-containing images. Success will be measured by:

- Achieving recall above 80% (most alcoholic containers are detected).
- Maintaining low false positives (non-alcoholic items rarely blurred).
- Processing a 1080p photo in under 10 seconds on a typical Windows laptop.
- Delivering an intuitive workflow where a user can blur and export in under 30 seconds.

## Milestones / Development Plan

- Week 1–2: Develop Qt GUI framework; implement manual blur tool.
- Week 3: Add OpenCV-based heuristic detection; enable preview and save.
- Week 4: Integrate CNN detection with ONNX Runtime.
- Week 5: Testing, refinement, dataset evaluation, and documentation.

## Limitations and Risks

- False positives (e.g., soda cans blurred incorrectly).
- False negatives in low-light or crowded images.
- Dataset bias due to limited examples of certain alcohol types.
- Project timeline may restrict advanced features like batch processing.

## Future Extensions

- Batch processing for entire image folders.
- Multiple blur styles (Gaussian, pixelation, mosaic).
- Social media integration for direct posting.
- User-defined presets for sensitivity and blur strength.

## References

1. Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.
2. Qt Company. (2024). *Qt Documentation*. Retrieved from  
<https://doc.qt.io>
3. Microsoft. (2024). *ONNX Runtime Documentation*. Retrieved from  
<https://onnxruntime.ai>
4. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
5. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
6. Kingma, D. P., & Ba, J. (2015). Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations (ICLR)*.
7. Gonzalez, R. C., & Woods, R. E. (2018). *Digital Image Processing* (4th ed.). Pearson.
8. Szeliski, R. (2022). *Computer Vision: Algorithms and Applications* (2nd ed.). Springer.
9. OpenCV Team. (2024). *Image Blurring with OpenCV*. Retrieved from  
[https://docs.opencv.org/4.x/d4/d13/tutorial\\_py\\_filtering.html](https://docs.opencv.org/4.x/d4/d13/tutorial_py_filtering.html)
10. Qt Company. (2024). *Qt Widgets Tutorial*. Retrieved from  
<https://doc.qt.io/qt-6/widgets-tutorial.html>
11. ONNX. (2024). *Exporting Models to ONNX*. Retrieved from  
<https://onnx.ai/getting-started.html>
12. Hugging Face. (2024). *Hugging Face — The AI community building the future*. Retrieved from  
<https://huggingface.co/>
13. GeeksforGeeks. (2025, May 30). *Introduction to GUI Programming in C++*. Retrieved from  
<https://www.geeksforgeeks.org/cpp/cpp-gui-programming/>
14. Towards Data Science. (2024). *Convolutional Neural Networks for Beginners*. Retrieved from  
<https://towardsdatascience.com/convolutional-neural-networks-for-beginners-c1de55eee2b2/>