# Group Proposal: Deep Learning for Trash Classification using TrashNet

## Problem Selection

We decided to tackle trash classification using images because it feels like a project that could actually help the planet. Sorting trash automatically could cut down on mistakes at recycling plants, speed things up, and save some money—pretty neat, right? Plus, we've been learning that image classification is a sweet spot for convolutional neural networks (CNNs), so it's a perfect chance to use deep learning from this course..

## Dataset

https://www.kaggle.com/datasets/feyzazkefe/trashnet/data

We will use the TrashNet dataset by Feyza Ozkefe from Kaggle. We're going with the TrashNet dataset by Feyza Ozkefe from Kaggle. It's got over 2,500 labeled images of trash like paper, cardboard, glass, metal, plastic, and non-recyclable trash. The classes are decently balanced, which is great since it means we've got enough data to train a solid CNN without hunting down extra images.

## Network Architecture

Our plan is to build a Convolutional Neural Network (CNN) for this. We'll start with something basic like LeNet to get a feel for it, then maybe level up to ResNet or MobileNet if we need more power. We're thinking of tweaking it with dropout and batch normalization to keep it from overfitting, and we'll play around with ReLU or softmax for the activation functions—whatever works best

## Framework

We're sticking with TensorFlow and Keras because they're super easy to use and have killer documentation. The GPU support is a bonus, and it ties into what we covered in lecture 5 with the Sequential and Functional APIs. It just feels like the right fit for this project.

## Reference Materials

Our reference materials include:
- Lecture slides on CNNs and TensorFlow (Lectures 4–7).
- TrashNet dataset documentation and Kaggle page.
- TensorFlow documentation and Keras tutorials.

## Performance Evaluation

We will evaluate the model using standard classification metrics:
- Accuracy: Primary metric for overall performance.
- Precision, Recall, and F1-Score: For class-specific insights.
- Confusion Matrix: To understand misclassifications.

We will also visualize the training process using learning curves and analyze overfitting using validation loss trends.