Our Idea and Why It Matters: We've noticed many financial apps struggle to sort spending properly. A transaction like SQ *COFFEE-SPOT 123 should go into Food or Coffee. Many companies pay for outside help to do this. It's slow (because of network calls), costly, and the company can't control the categories.

We want to build 'FinSort,' a simple Python tool that solves this problem internally. Our system will take messy transaction info, clean it up, and then use a simple machine learning model to assign a category and a confidence score. It'll all happen on your computer, so it's fast, free after setup, and secure.

Here's How It Will Work:

Cleaning Up the Data: First, we'll clean up the messy transaction info. We'll use rules to clean the text, removing extra numbers, codes (like SQ *), and punctuation to get to the key part: the store's name (like COFFEE-SPOT).

Training a Smart Model: We'll use transaction data to train a machine learning model (maybe Logistic Regression or Random Forest since they're easy to understand). The model will pick up on patterns, like how coffee or cafe points to Food & Dining. We'll tune this model to aim for a 0.90 macro F1-score, a high accuracy target.

Predicting Locally: The trained model will be saved as a file. Our program will load the file, take a new, clean transaction, and quickly guess the category and its confidence level.

What Makes Us Different: Our innovation is how we handle the Customisable & Transparent need. It's our best feature.

The Config File: Instead of having the model predict Food & Dining directly, we'll train it to predict simple tags like coffee_shop, airline, or grocery. Then, we'll add a simple configuration file in JSON or YAML format. This file will be a map, like: { coffee_shop: Food & Dining, airline: Travel } If someone wants to change Travel to Business Travel, they just edit that file. They don't need to touch our code or retrain the model. This meets the need for admin-controlled changes without code edits and makes our tool flexible.

Bonus Features:

Explainable Results: We want to add a feature to show why the model made a certain choice. For a Starbucks transaction, we could show that starbucks was the biggest clue for our coffee_shop tag. This builds trust.

Feedback: For predictions with low confidence, our tool will flag them. A user can review and correct the category. These fixes will be saved, and we can use them to retrain our model later, making it smarter.

Fair AI: We'll check our model for biases, making sure it doesn't unfairly link high amounts to fraud or specific categories, and we'll note this in our report.

The Outcome: 'FinSort' will let any developer add fast, accurate transaction sorting to their app with no API costs and full control over their categories. We're excited because it's a practical, complete solution to a real problem.