CMSC 25-316

Financial Receipt Capture and Analysis System

Final Design Report

Prepared for

Mehul Garnara

Capital One

By

Will Southerland, Josh Tarongoy, Joshua Whitchurch, Lindsey Marandina

Under the supervision of

Thang Dinh

5/2/2025

Executive Summary

Project Overview

This project aims to develop an application that streamlines the process of capturing and managing hardcopy credit card transaction receipts through capture, storage, and analysis of physical receipts. The system will allow users to:

- Search receipts by date, vendor, category, and other criteria.
- Analyze spending across different periods and categories.
- Identify spending trends to gain valuable insights.
- Utilize a categorization system to enhance financial tracking and analysis.

Design Specifications

- Utilize Amazon Web Services tools for key functions like data storage, database management, OCR, file storage, and cloud hosting
- Key information extraction: Identifying critical data points from receipts (e.g., date, vendor,
- amount).
- Categorization: Providing both automatic and/or manual categorization options.
- Search capabilities: Enabling efficient searching based on multiple criteria.
- Analysis tools: Offering tools for analyzing spending trends and patterns.

Spring Semester Updates

During the Spring Semester, the team made numerous updates to both the front end and back end. In order to get more accurate OCR results, we used Textract instead of Tesseract. Tesseract required intensive training that would have gone a bit beyond the scope of our resources, and Textract has a comprehensive API library with functions specifically geared towards analyzing financial documents. Upon switching to AWS Textract, we found that we were able to get the accuracy we needed for this project. We implemented a way for users to edit receipts both upon uploading and after uploading as well, in order to handle situations where OCRs results were not fully accurate.

We created an EC2 instance that allowed us to preserve our local computers resources, and host the project live on AWS's cloud computing platform. There were also numerous updates to the user interface, including redesigning the whole front end, and adding features to highlight newly updated receipts. A mobile UI version viewable through a mobile web browser was also implemented.

Table of Contents

Section A. Problem Statement

Many people face challenges keeping track of spending and are often unaware of how much they are actually spending versus how much they think they are spending. With many often spending more than what they can afford leading to credit card debt. The manual process of storing, searching, and analyzing physical receipts is time consuming, inefficient, and prone to errors. Financial receipts are often lost, discarded, or difficult to analyze, making it challenging for individuals to track their spending and make informed financial decisions. This project will allow individuals to keep track of their spending by digitally uploading receipts so they are able to make informed financial decisions.

Capital One is an American financial services company known for their innovative approach to banking and credit card services. Capital One offers many different credit cards and services to fit the needs of their customers. The company follows a data-driven approach with a "reverence for insights derived from data" (Bean, 2024) and "adopted a "You Build, Your Data" approach to give our data users greater ownership and accountability over their data" (Lebonitte, 2024).

The problem has currently been addressed through applications that allow users to track their expenses through connecting their accounts to track spending and income. One well known application is Rocket Money. Rocket Money "is a personal finance app…that helps you manage your money, lower your bills, and even cancel unwanted subscriptions" (Hamilton, 2022). Other banking companies, like Bank of America, have similar expense and spending insights. Bank of America has "Category-specific interactive charts and longer-term spending trends help you identify where you are frequently over or under budget" (Use the spending & budgeting tool to get a clear view of your finances). One study found that "financial literacy helps people make informed and responsible financial decisions, boosting financial stability and reducing financial worries" (Bai, 2023)

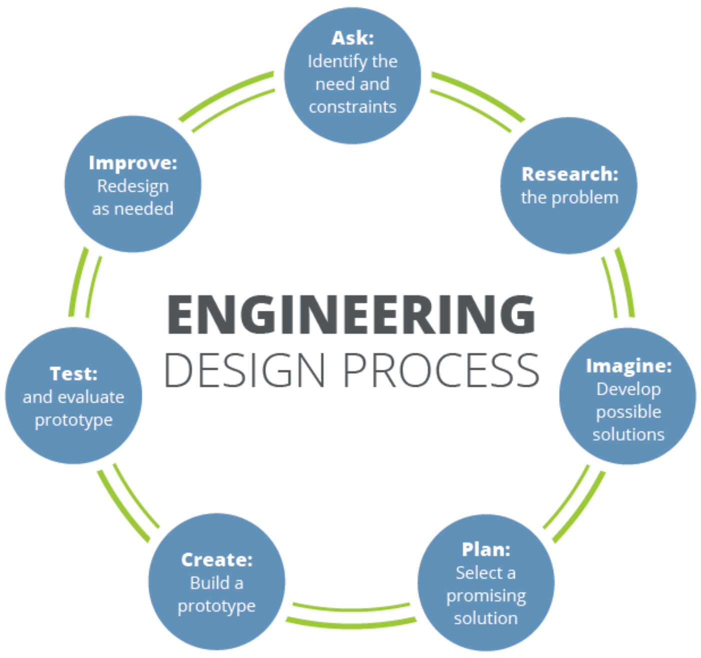

Ask: Identify the need and constraints
Research: the problem
Imagine: Develop possible solutions
Plan: Select a promising solution
Create: Build a prototype
Test: and evaluate prototype
Improve: Redesign as needed
ENGINEERING
DESIGN PROCESS

Figure 1. The iterative nature of the engineering design process [2].

Section B. Engineering Design Requirements

B.1 Project Goals (i.e. Client Needs)

The goals and design objectives presented in this section were constructed with the help of mentors from Capital in order to meet their expectations.

- Digitally store receipts (physical and digital)
- Ability to search through receipts efficiently
- Ability to query receipts at the item level
- Insights and trending for the consumer spending

B.2 Design Objectives

- The design will utilize AWS and other appropriate technologies to store receipt and receipt metadata
- The design will allow for the capture of both digital and physical receipts
- The design will allow for the categorization of receipts by receipt type (eg Gas, Groceries, Candles)
- The design will include functionality for searching
- The design will provide insights and trending for consumers

B.3 Design Specifications and Constraints

- Design must have the ability to receive images (jpg, png, etc.) of receipts for extraction. (Functional constraint)
- Design must have the ability to process receipts of varying layouts and formats. (Functional constraint)
- Design must have accurate OCR results when extracting text from receipts (Functional constraint)
- Design must have the ability to extract key information (Functional constraint) – examples include date, vendor, total, etc.
- Design must be able to automatically and/or manually categorize receipts for organization (Functional constraint) – Categorical examples include grocery, shopping, entertainment, medical, utilities, dining out, transportation, travel, etc.
- Design must have the ability to search for receipts (Functional constraint) – examples include search by date, vendor, cost, category, etc.
- Design must analyze spending based on time periods and categories. (Functional constraint)

Design must show trend analysis to identify spending patterns. (Functional constraint)

B.4 Codes and Standards

Standards

- **W3C HTML5 and CSS3 Standards**
  Our frontend adheres to W3C standards for HTML and CSS to ensure that the system functions consistently across all modern browsers and devices. This promotes cross-platform accessibility and maintainability.

- **IEEE 830-1998 – Software Requirements Specification**
  Our project documentation and feature planning have followed the structure and recommendations of IEEE 830 to ensure clarity, traceability, and alignment between user needs and technical design.
- **AWS Well-Architected Framework (WAF)**

  While not a formal standard, AWS WAF guidelines helped guide decisions around reliability, cost optimization, and operational excellence in our EC2- and DynamoDB-based infrastructure.

Section C. Scope of Work

C.1 Deliverables

Project Deliverables

- Functional web-based receipt capture system with-
    - Front end UI built using React, including-
        - Receipt upload form
        - Dashboard for viewing stored receipts
        - Search/Filter functionality
    - Back-end API built using Flask with endpoints for-
        - Uploading and processing receipt images
        - Parsing Receipt data using Textract
        - Storing and retrieving data from DynamoDB
- Data Schema and database model for DynamoDB
- Deployment of the system to a live server using EC2 and S3
- Demo video for presentation
- Capstone Deliverables-
    - Team contract
    - Project proposal
    - Preliminary design report
    - Fall semester poster
    - Final Design report
    - Spring design poster

Risk Mitigation and Objectives

Campus-Dependent Deliverables

None of the technical deliverables require physical campus access. All development work is done on personal laptops and cloud platforms. Posters for presentations may be printed on campus, but digital copies are sufficient for backup.

Remote Work & Resources

All development work can be done remotely using:

- Github
- AWS
- Visual Studio Code
- Zoom/Discord

Third-Party Dependencies & Supply Chain
No physical components or hardware are required.
Reliance on AWS services (Textract, S3, EC2, DynamoDB) introduces dependency risks:

- AWS service outages could delay receipt processing and testing.
- Cost management is important to prevent resource throttling.
- Mitigation: monitor usage with budget alerts; rely on local fallback testing where possible.

No external vendors are needed for system functionality.

Foreseeable Obstacles

- **OCR inconsistency**: Poor receipt images may lead to extraction errors.
  Mitigation: Implement multiple preprocessing pipelines and fallback logic.
- **Integration issues**: Merging front-end and back-end functionality may present challenges.
  Mitigation: Conduct weekly integration tests and code reviews.

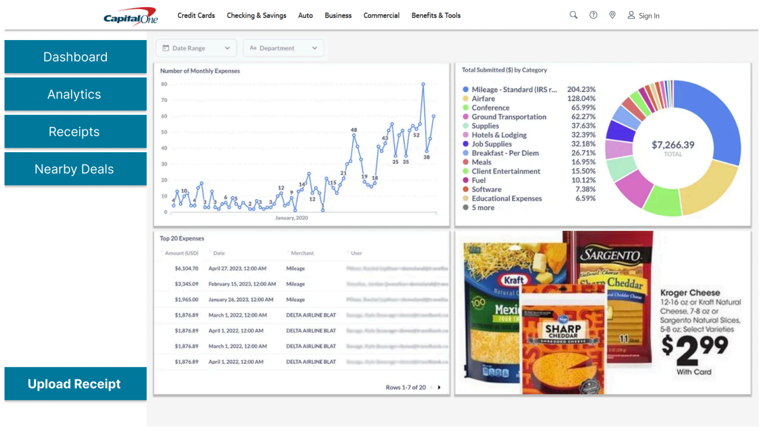
C.2 Milestones

Semester 1: MVP	● Extract metadata from receipt images, create data model, wiremock UX flow ● Search functionality
Semester 2: Refinement	● Insight features including spending trends based on categories ● Integrate features on web interface for upload, search, and insight
Final: Demo	● Present demo

C.3 Resources

- ExpressExpense Receipt Database: Five-thousand US receipts of various types for training and testing purposes
- AWS Lambda: API calls to access the database.
- AWS DynamoDB: NoSQL database to store receipt information for the users
- AWS S3: Storing images is not possible within Dynamo, so an additional storage system is necessary

Section D. Concept Generation



Total Submitted ($) by Category

Mileage - Standard (IRS r...	204.23%
Airfare	128.04%
Conference	65.99%
Ground Transportation	62.27%
Supplies	37.63%
Hotels & Lodging	32.39%
Job Supplies	32.18%
Breakfast - Per Diem	26.71%
Meals	16.95%
Client Entertainment	15.50%
Fuel	10.12%
Software	7.38%
Educational Expenses	6.59%
5 more	

$7,266.39 TOTAL

Top 20 Expenses

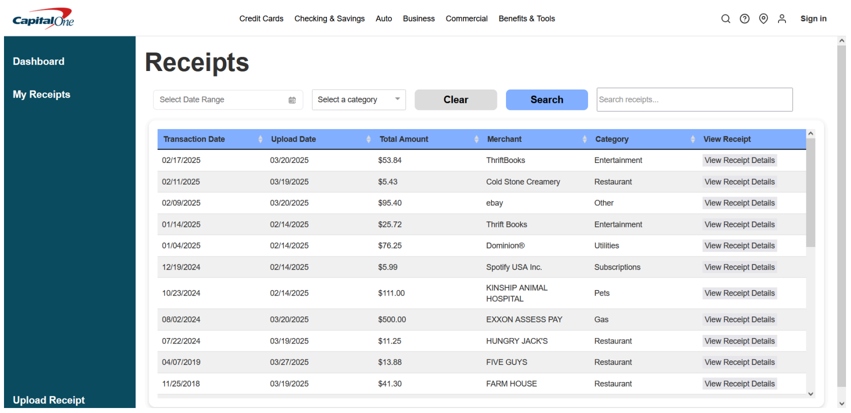
Amount (USD)	Date	Merchant	User
$6,104.70	April 27, 2023, 12:00 AM	Mileage	[redacted]
$3,345.09	February 15, 2023, 12:00 AM	Mileage	[redacted]
$1,965.00	January 26, 2023, 12:00 AM	Mileage	[redacted]
$1,876.89	March 1, 2022, 12:00 AM	DELTA AIRLINE BLAT	[redacted]
$1,876.89	April 1, 2022, 12:00 AM	DELTA AIRLINE BLAT	[redacted]
$1,876.89	March 1, 2022, 12:00 AM	DELTA AIRLINE BLAT	[redacted]
$1,876.89	April 1, 2022, 12:00 AM	DELTA AIRLINE BLAT	[redacted]

Rows 1-7 of 20

Figure 1: Mockup design of the dashboard made using Figma.

The webpage for the Financial Capture and Analysis System needed to give users ways of monitoring changes in their spending habits. For most of our system, we looked at other financial analysis systems from other companies for inspiration on how we should design ours.

- A line chart is used to show the spending habits over the course of time.
- The donut chart shows how much of a user's expenses are going to which category (gas, food, shopping, etc.)
- At the top are filtering options that allow users to focus on a range of data.



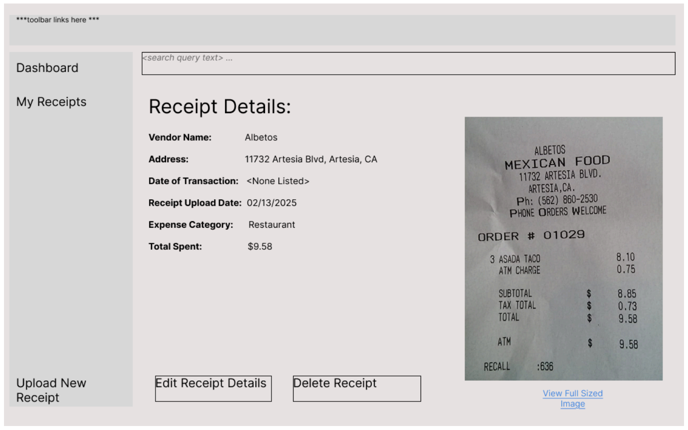
Receipts

Transaction Date	Upload Date	Total Amount	Merchant	Category	View Receipt
02/17/2025	03/20/2025	$53.84	ThriftBooks	Entertainment	View Receipt Details
02/11/2025	03/19/2025	$5.43	Cold Stone Creamery	Restaurant	View Receipt Details
02/09/2025	03/20/2025	$95.40	ebay	Other	View Receipt Details
01/14/2025	02/14/2025	$25.72	Thrift Books	Entertainment	View Receipt Details
01/04/2025	02/14/2025	$76.25	Dominion®	Utilities	View Receipt Details
12/19/2024	02/14/2025	$5.99	Spotify USA Inc.	Subscriptions	View Receipt Details
10/23/2024	02/14/2025	$111.00	KINSHIP ANIMAL HOSPITAL	Pets	View Receipt Details
08/02/2024	03/20/2025	$500.00	EXXON ASSESS PAY	Gas	View Receipt Details
07/22/2024	03/19/2025	$11.25	HUNGRY JACK'S	Restaurant	View Receipt Details
04/07/2019	03/27/2025	$13.88	FIVE GUYS	Restaurant	View Receipt Details
11/25/2018	03/19/2025	$41.30	FARM HOUSE	Restaurant	View Receipt Details

Figure 2: Image of the current design of the Receipts page.

Users need to be able to manually change information about receipts, and sort by any kind of category (such as the most expensive) to search for specific receipts.

- A sortable table seemed the best idea as it allows users to sort the listing results.
- Options for viewing or deleting a receipt are also available for each record.
- Sorting options are also available just like in the dashboard.



***toolbar links here ***

<search query text> ...

Dashboard

My Receipts

Upload New Receipt

Receipt Details:

Vendor Name:	Albetos
Address:	11732 Artesia Blvd, Artesia, CA
Date of Transaction:	<None Listed>
Receipt Upload Date:	02/13/2025
Expense Category:	Restaurant
Total Spent:	$9.58

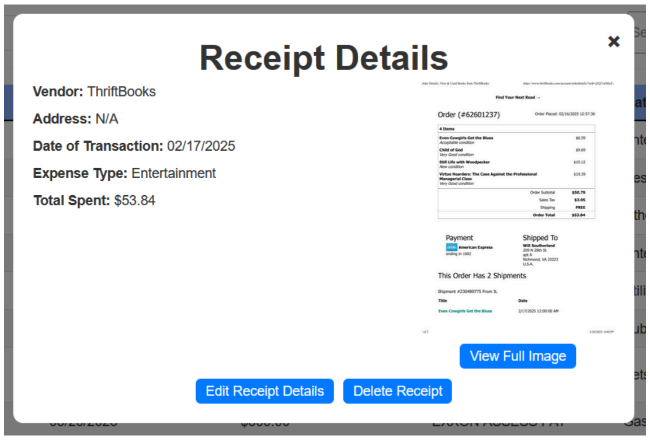
ALBETOS
MEXICAN FOOD
11732 ARTESIA BLVD.
ARTESIA,CA.
Ph: (562) 860-2530
PHONE ORDERS WELCOME

ORDER # 01029

3 ASADA TACO		8.10
ATM CHARGE		0.75
SUBTOTAL	$	8.85
TAX TOTAL	$	0.73
TOTAL	$	9.58
ATM	$	9.58

RECALL :636

Figure 3: Initial design of the receipt details page



Receipt Details

Vendor: ThriftBooks

Address: N/A

Date of Transaction: 02/17/2025

Expense Type: Entertainment

Total Spent: $53.84

Figure 4: Final design of the receipt details modal

When the user wants to view a receipt's details, it should clearly provide details as well as options for editing or deleting.

- Clicking on 'view receipt' will open a modal instead of loading information into a new page. This allows users to quickly view receipts without having to wait for the page to load.
- Each receipt's key details are clearly listed on the left-hand side of the modal.
- A link to the receipt's image is also stored in the record, so that the user can view the digital copy of their receipt for the actual details.
- Users can also edit information that is stored per record (such as changing the category of the expense, changing the vendor's name, address, etc.)

Section E. Concept Evaluation and Selection

When it came down to choosing a design of how the webpage would look, the initial design (see Section D) was the only one that was made.

The design, as explained by the sponsors, should have an importance towards giving the user granular control over the tool to understand their expense patterns and history. As such, our initial design worked well and accomplished many of the requirements:

- Users should be able to upload a photo of the receipt to the system.
- Show the user the data from the collected receipts.
- Allow the user to manage the scope of the data being analyzed through filtering tools.
- Allow the user to view and edit the receipts they had uploaded to the application.

Section F. Design Methodology

F.1 Computational Methods (e.g. FEA or CFD Modeling, example sub-section)

Testing OCR was one of the main focuses of testing, since the whole backbone of the project required accurate OCR extraction. This was simply done by ensuring that the extracted data matched the original image.

F.2 Experimental Methods (example subsection)

Testing OCR accuracy involved using what we considered to be ‘good’ images, and ‘bad images’.   A good image is one that has plain font text with no logos, a clear image with little to no noise, and obvious fields like total spent, vendor name, date of transaction, and address. A bad image is one that would predictably give bad OCR results- any logos, crumpled receipts, and poor lighting would yield bad results. The idea behind this was to see how well our OCR tool could handle a diverse range of inputs.

F.3 Architecture/High-level Design (example subsection)

The system uses a client-server architecture:

- Front-end: React.js interface for uploading receipts and visualizing parsed data
- Back-end: Flask API handling uploads, invoking OCR engines, parsing, and storing results
- Database: AWS DynamoDB stores structured receipt data, including user and vendor records
- Storage: AWS S3 bucket stores the original receipt image

AWS EC2 hosts the application, enabling scalability and reliable access. Error handling and fallback logic ensure the system remains operational under variable OCR performance.

F.5 Validation Procedure

Validation will ensure that the system meets the client’s functional requirements, including:

- Uploading receipts with a variety of formats and image qualities
  Accurately extracting and storing key data fields
- Displaying results in an accessible and meaningful way for end users

Validation occurred in mid April to ensure that all project requirements were met. We presented a working version of the project and held a feedback session. Feedback from our sponsor group was captured by taking notes and updating last minute changes as required.

Section G. Results and Design Details

One of the big focuses of last semester's implementation was to use cheaper, open source tools for image processing. The first approach was to use Tesseract in conjunction with a few different image processing tools in order to normalize images, making them easier for Tesseract to read. Image normalization involved taking in the original image, and using tools to change the image to optimize it for OCR processing. These optimizations included

- Black and white conversion
  - If a pixels brightness value is above a certain threshold, convert it to white
  - If the pixels brightness value is below that threshold, convert it to black
- Bold black pixels slightly
- Make the image larger
- Gaussian blur
  - An image smoothing technique that reduces noise and detail by averaging pixel values
  - Each pixels new value is computed as a weighted average of its neighbors, where closer pixels have more influence
  - σ controls the amount of blur

$$G(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{x^2+y^2}{2a^2}}$$

This approach proved to be insufficient for the wide variety of ways a receipt can be formatted. For an image where this produced sufficient results, there would be another where it did not. Ultimately, the wide variety of images this project needed to process made it impossible to find an image processing pipeline that could give universal results.

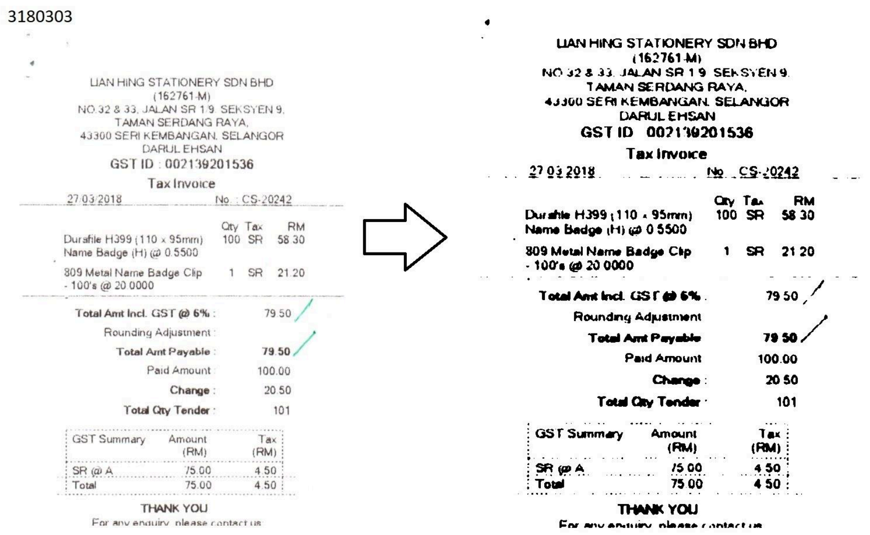
The next attempt was to train Tesseract using machine learning. To train Tesseract requires three things-

- An image (in our case, a receipt)
- A ground truth text- a text file that is a line for line transcription of the image
- A box file- a file that defines the coordinates of each corner surrounding each character. The box file tells the ML model where on an image the character within a ground truth text is located

We found that creating a sufficient training data set was too far out of the time constraints of this project to be a reasonable approach. Ultimately, we decided to implement AWS's OCR tool, Textract. Textract includes an API library specifically for financial documents, so out of the box it was able to instantly give us the level of accuracy that was required for this project.

In terms of the rest of the project, we were successful in having the boiler plate for the front end completed last semester. Aside from a visual update, and going back and updating things as needed, the structure of the program itself was completed in the Fall.

G.1 Modeling Results (example subsection)



3180303

LIAN HING STATIONERY SDN BHD
(162761-M)
NO.32 & 33, JALAN SR 1/9, SEKSYEN 9,
TAMAN SERDANG RAYA,
43300 SERI KEMBANGAN, SELANGOR
DARUL EHSAN
GST ID : 002139201536
Tax Invoice
27/03/2018 No. : CS-20242

	Qty	Tax	RM
Durafile H399 (110 x 95mm) Name Badge (H) @ 0.5500	100	SR	58.30
809 Metal Name Badge Clip - 100's @ 20.0000	1	SR	21.20

Total Amt Incl. GST @ 6% :	79.50
Rounding Adjustment :	
Total Amt Payable :	79.50
Paid Amount :	100.00
Change :	20.50
Total Qty Tender :	101

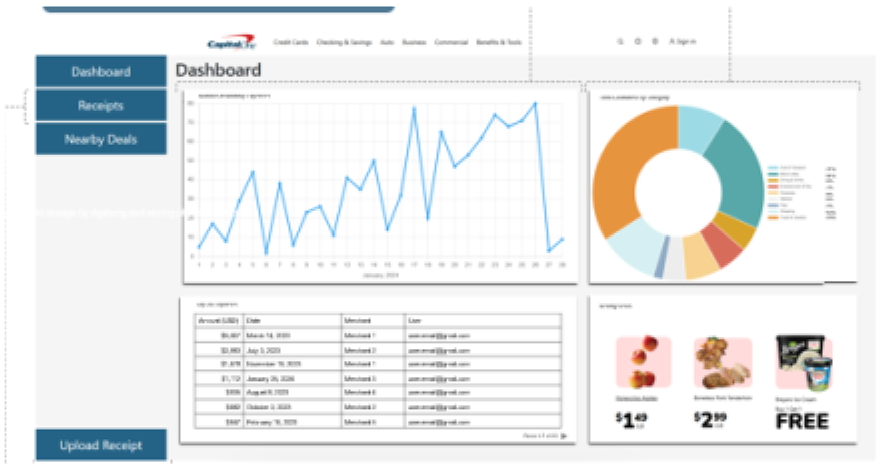
GST Summary	Amount (RM)	Tax (RM)
SR @ A	75.00	4.50
Total	75.00	4.50

THANK YOU
For any enquiry, please contact us

LIAN HING STATIONERY SDN BHD
(162761-M)
NO.32 & 33, JALAN SR 1/9, SEKSYEN 9,
TAMAN SERDANG RAYA,
43300 SERI KEMBANGAN, SELANGOR
DARUL EHSAN
GST ID 002139201536
Tax Invoice
27/03/2018 No. CS-20242
Qty Tax RM
Durafile H399 (110 x 95mm) Name Badge (H) @ 0.5500 100 SR 58.30
809 Metal Name Badge Clip - 100's @ 20.0000 1 SR 21.20
Total Amt Incl. GST @ 6% 79.50
Rounding Adjustment
Total Amt Payable 79.50
Paid Amount 100.00
Change 20.50
Total Qty Tender 101
GST Summary Amount (RM) Tax (RM)
SR @ A 75.00 4.50
Total 75.00 4.50
THANK YOU
For any enquiry, please contact us

Example of one of many ways in which image normalization/Tesseract optimization was attempted

Spring UI Mock

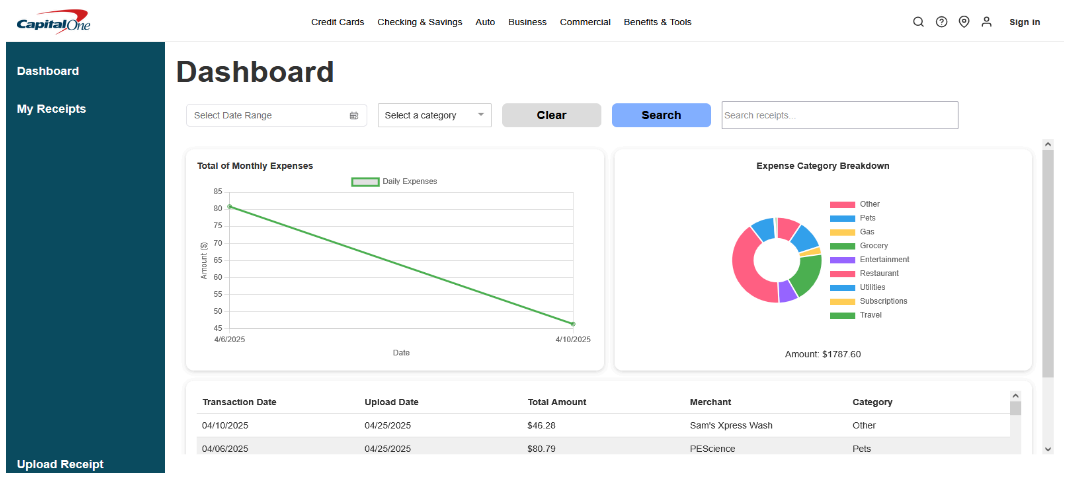


G.2 Validation

All features, code, and design were validated during weekly meetings with the Capital One team. Every week, we would show our own personal progress, discuss implementation, struggles, concepts, and how we see work the next week going. The sponsor team offered feedback, guidance, and suggestions on new features to implement. As well as our sponsor meetings, we met internally each week to discuss progress in a similar manner. Any work, ideas, or questions would be discussed in our Discord channel.

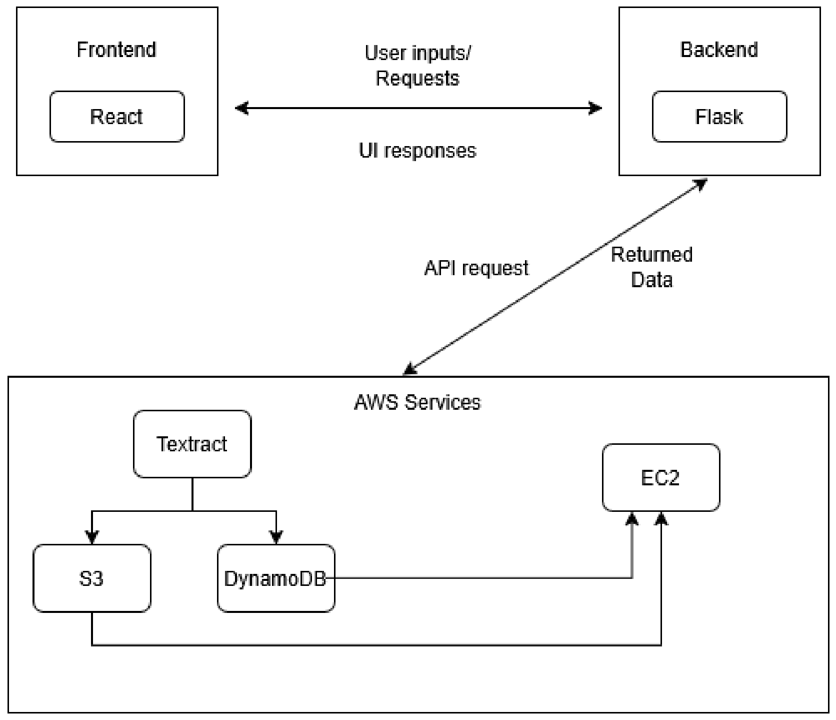
G.3. Final Design Details/Specifications

Final UI



Transaction Date	Upload Date	Total Amount	Merchant	Category
04/10/2025	04/25/2025	$46.28	Sam's Xpress Wash	Other
04/06/2025	04/25/2025	$80.79	PEScience	Pets

Data Flow Diagram



Section H. Societal Impacts of Design

With this application, users are better able to keep track of their spending habits. As such, the Financial Capture and Analysis System affects people in positive ways, as after every purchase they print a receipt or have it emailed to them. Then they can upload it to the system and have it stored and used in analysis tools. Over the course of time, users can get a better idea of what they spend most of their money on and how frequently they spend.

H.1 Personal Economics

The application allows users to keep track of their spending habits by uploading their receipts whenever they make a purchase. If a user notices they spend a lot of money on groceries, it can lead them to try and limit how much they buy, or to look for deals that can save them some money. In the long run, the application should help them by being financially responsible.

H.2 Energy and Upkeep Costs

A negative impact of this project is the amount of energy and money required to keep the server running, as well as storing digital data on servers. When a user uploads a receipt, that image is stored on the server and can be accessed at any time. For as long as that user has an account to use this application, that receipt needs to be kept. Over time, this can increase as the user makes daily purchases over the course of years, and that’s not considering multiple users.

H.3 Environmental Impacts

Some unintended consequences that can impact other areas are the waste of paper used for physical receipts. Assuming everyone uses this application, with every purchase a receipt is printed, uploaded digitally, then the receipt gets thrown away. This would negatively impact the ecological environment with plenty of waste that could be avoided by the average person.

H.4. Economic Impacts

Another impact could be economically, as people would be encouraged to spend less, or to only buy things that are necessary. Certain brands or companies could stop producing certain products since no one or too few people are buying them to justify the cost of making those products.

Section I. Cost Analysis

Over the course of development between the two semesters, we incurred only one expense for a dataset of receipt images in order to use for testing purposes, which was $250.

Section J. Conclusions and Recommendations

We made sure to have the front end design squared away in the Fall semester, so we could spend more time in the Spring semester hashing out technical details. The front end evolved as new features were added, but it stayed consistent throughout the duration of this project. While we faced hurdles in terms of technical implementation, there were great lessons learned in how to best approach a project of this nature. One of the initial goals of this project was to use open source tools where we can. As discussed previously, this idea was primarily centered around using an open source OCR tool. Since this project required a strong level of accuracy on an extremely diverse data set, the work it took to achieve this was far out of the scope of our capabilities in terms of time constraints. Implementing AWS tools saved us a lot of time and resources, and still ended up aligning with our goals of keeping this project as low cost as possible.

Our design was centered around intuitive use that required little to no instruction. The only section where user instructions were provided was on how to best photograph the receipt images being uploaded to the system, which is a common practice in similar tools. We wanted to design the database to provide only relevant information to the user, while keeping relational/identifying information tucked away in the table. The dashboards charts were also successful in supplying useful information about users' spending habits, with filtering tools that any regular user would ideally understand how to use with no instruction.

Overall, we met a majority of the initial requirements laid out for us. As with any project, requirements evolve as time goes on. In the future, there are some features that we believe would greatly improve the functionality of this project. One being a system to allow for multiple users to create accounts. This could be achieved numerous ways, but one way we wanted to attempt was by using AWS's Cognito service. It would require updating the database with a user ID field, and updating our S3 bucket to append the user ID to each image for security purposes. Another interesting feature for the back end would be more advanced expense category detection. In our final implementation, expense categories were manually selected by users. In the future, a hash table could be implemented to associate specific vendors with their expense category. For example, if a user uploads a Kroger receipt and selects the 'Grocery' category, future receipts from Kroger would automatically be detected as 'Grocery'. As with each receipt, it would require user validation prior to uploading.

Appendix 1: Project Timeline

Milestone 1	Convert Receipts into metadata, create data model, wiremock UX Flow								
Milestone 2	Search functionality using various search criteria								
Milestone 3						Insight features including trends based on categories			
Milestone 4						Integrate features on web interface for upload, search, and insight features			
Project Deliverables	Team Contract	Project Proposal	Fall Poster	Prelimi-nary Report	Team Contract update		Expo Poster & Abstract	Expo	Final Design Report
Due dates	September	October	November	December	January	February	March	April	May

Appendix 2: Team Contract (i.e. Team Organization)

Basic Information

Team Member Name	Strengths each member bring to the group	Other Info	Contact Info
Will Southerland	Organized, excellent communicator, punctual	I'm open to new perspectives and ideas, and work best with organization and a deadline.	southerlandw@vcu.edu
Josh Tarongoy	Early starter, worker, I do things.	Communicates with others to ensure everyone's on the same page.	tarongoyj@vcu.edu
Joshua Whitchurch	Hard-worker, punctual.	I love to find creative solutions to problems and value communication with teams.	whitchurchjm@vcu.edu
Lindsey Marandina	Organized, Communicative	Open to any ideas	marandinalc@vcu.edu

Other Stakeholders	Notes	Contact Info
Thang Dinh (Faculty Sponsor)		tndinh@vcu.edu
Mehul Garnara (Capital One Sponsor)		mehul.garnara@capitalone.com

Team Culture

Culture Goals	Actions	Warning Signs
Make sure every team member is aware of progress on the project	- Communicate when work is being done, either via our Discord server or email -Upload progress to our github repo	-Progress is not communicated -Repo is not updated clearly

Alert others about upcoming delays.	- A simple message telling the others about their schedule is enough. (I.e. Busy due to a club meeting, exam, etc.)	- Other members will question if you have done your part of the work.
Foster an environment where all ideas are valued	-Make sure team members who want to approach a problem their own way have the space to do so	-If you disagree with another team members approach, communicate it with respect and an open mind

Time Commitments, Meeting Structure, and Communication

Meeting Participants	Frequency Dates and Times / Locations	Meeting Goals Responsible Party
Students Only	Saturdays at 10 am, voice call on discord	Update group on day-to-day challenges and accomplishments, progress on the project, scheduling future meetings, etc.
Students + Faculty advisor	Fridays at 11 am on Slack and as needed	Update faculty advisor and get answers to our questions
Project Sponsor	Fridays at 11 am on Slack	Update our sponsor on progress, receive feedback, hear expectations

Individual Roles and Responsibilities

Team Member	Role(s)	Responsibilities
Will Southerland	Project Manager	● Facilitate communication between the group, advisors, and sponsor ● Schedule meetings ● Keep group members on task and make sure work is being accomplished on time

Lindsey Marandina	Logistics Manager	● Documents meetings and goals ● Keeps track of upcoming meetings and tasks ● Follow up with task progress
Joshua Whitchurch	Test Engineer	● Facilitate planning for sprints as they occur ● Research necessary software, data mining algorithms, etc as needed and provide information to other team members
Josh Tarongoy	Systems Engineer	● Analyzes client initial design specification and comes up with product specifications for the team. ● Manages internal subsystems in developing prototype ● Develops system architecture and manages product interfaces

Signature

Team Member: Will Southerland	Signature: Will Southerland__
Team Member: Josh Tarongoy	Signature: ___Josh Tarongoy_____
Team Member: Joshua Whitchurch	Signature: Joshua Whitchurch
Team Member: Lindsey Marandina	Signature: Lindsey Marandina

References

Provide a numbered list of all references in order of appearance using APA citation format. The reference page should begin on a new page as shown here.

[1] VCU Writing Center. (2021, September 8). *APA Citation: A guide to formatting in APA style.* Retrieved September 2, 2024. https://writing.vcu.edu/student-resources/apa-citations/

[2] Teach Engineering. *Engineering Design Process*. TeachEngineering.org. Retreived September 2, 2024. https://www.teachengineering.org/populartopics/designprocess