**Receipt Free Android Application**

**CSCI 6838 – Capstone**

**December 13, 2018**

Mr. Ravi K. Ganta

Mind Agile

Dr. Gary D. Boetticher

University of Houston – Clear Lake

Kuruvilla, Lovin

KuruvillaL2945@uhcl.edu.

Mandava, Maneesh

MandavaM8038@uhcl.edu.

Parikh, Nidhi Sunit

ParikhN7442@uhcl.edu.

Jalodara, Kena

JalodaraK8037@uhcl.edu.

**Acknowledgements**

Our sincere thanks to Mr. Ravi Ganta, our mentor for believing in us and supporting us in all means.

We are grateful to our capstone professor and mentor Dr. Gary D. Boetticher for supporting us throughout the project and giving us valuable feedback from time to time.

Finally, we thank the University of Houston-Clear lake for giving us this opportunity in gaining knowledge and building successful career.

**Table of Contents**

**Topic Page #**

**Introduction ……………………………………………………….……… 01**

**Market Analysis .…….…………………………………………………… 02**

**Requirements ……………………………………………………………… 03**

**System Architecture ……..………………………………………………… 04**

**System Design ……………………………………………………………… 06**

**Implementation ……………..……………………………………………… 10**

**Milestones …………………………………………………………………... 13**

**Conclusion …………………………………………………………………. 14**

**Future Work …………….………………………………………………… 15**

**References ……………….………………………………………………… 16**

**Appendix A - Code ……………….………………….……………………. 17**

Table of Figures

|  |  |  |
| --- | --- | --- |
| **Number** | **Name of Figure** | **Page no** |
| Figure 1 | Architectural Diagram | 4 |
| Figure 2 | Login Screen | 6 |
| Figure 3 | OCR Screen | 7 |
| Figure 4 | Categories Screen | 8 |
| Figure 5 | Result Screen | 9 |

**INTRODUCTION**

Personal financial management is a daunting and continuous task that can cause even the most economically savvy individual to become confused or short-sighted. “Indeed, in a world where assets and investments move quickly, and we link our bank accounts to innumerable services and make purchases with the touch of a button, financial management is a trickier concept than ever before” [10]. To make the most of the money at your disposal requires constant awareness and strategic thinking. Accountability might be considered as one of the potential solution for effective Financial Management. Traditionally, people store all kinds of receipts till the end of the month to account for their expenses. This process would be very much hectic and annoying for the people who have thousands of receipts to account for in a month. Having said that, to avoid all the troubles and difficulty in storing the receipts the company “Mind Agile” has taken an initiative to build an interest in developing a receipt processing app named “Receipt Free”. Receipt Free is an user oriented android application which solves the problem of maintaining accountability through traditional storage methods by replacing them with Google Vision OCR and Google storage methods which have necessary functionalities to automate the receipt processing and storing all the necessary information. It can be trusted and relied on as it was built on following powerful API integrations Google Cloud Vision OCR for text extraction through receipts, Firebase for secure authentication, Google Drive for optimal storage, Natural Language Processing for accurate classification of extracted text, Google Spreadsheets for storing all the receipt information needed for generating reports. Unlike the existing and traditional methods, handling of receipt free is pretty much easy as follows, just upload the receipt image from your gallery if it already exists or just scan it using camera and you are done. Further, the app performs internal processing utilizing all those integrated API’s briefed above (detailed insight into the app processing is provided in the Implementation section below) and generates a pie chart with detailed information in diversified categories through which user can visualize and acknowledge all his expenses in a selected time frame. Thus, Receipt Free can be considered as a powerful android application which serves both as a Financial cloud and worthy advisor in terms of maintaining accountability leading to successful financial management.

**MARKET ANALYSIS**

There are few apps like Trip it, Evernote, Square, Mint which help in tracking business expenses and tax filings and many mobile apps of banks which provide transaction details of the user. All these apps are limited to storing information regarding expenses and transactions with precise details like name of the firm, amount, date and time. In accounting through these apps, there is no guaranteed success in maintaining financial management by the user due to their drawback of limited information which fall short in accounting transactions where multiple people are involved in the same transaction, transactions where expenses are diversified as some into personal expenses and some into business expenses etc. Thus, Receipt Free considers these potential and real time problems to encounter the drawbacks of these apps by providing unique features and detailed information that could account for optimal Financial Management.

Receipt Free: This app poses the functionalities and features that are very much effective in solving the real time problems in terms of maintaining accountability. Few real time problems are as follows:

1) Losing the receipts through which business expenses can be claimed.

2) Lacking detailed knowledge on expenses made in different categories.

3) Tendency to forget shared expenses and accounting for it etc.

These problems have significant impact in effective Financial management as they account to a huge amount in broader time scale. Thus, considering all these scenarios receipt free helps in maintaining accountability leading to successful Financial management by providing following unique features.

**UNIQUE FEATURES OF RECEIPT FREE:**

1) Providing detailed information of the expenses made,

2) Secure authentication through firebase,

3) Google cloud Vision OCR for accurate text extraction,

4) Live camera access,

5) Editable text fields,

6) Storage of receipts of future reference,

7) Speech to Text functionality,

8) Dashboard depicting a pie chart and time-time expenses with detailed information,

9) Confidential.

**REQUIREMENTS**

This section provides all the requirements in designing Receipt Free Android Application. The requirements involve both Functional and Non-Functional types with their respective priorities.

Table 1: Requirements Table

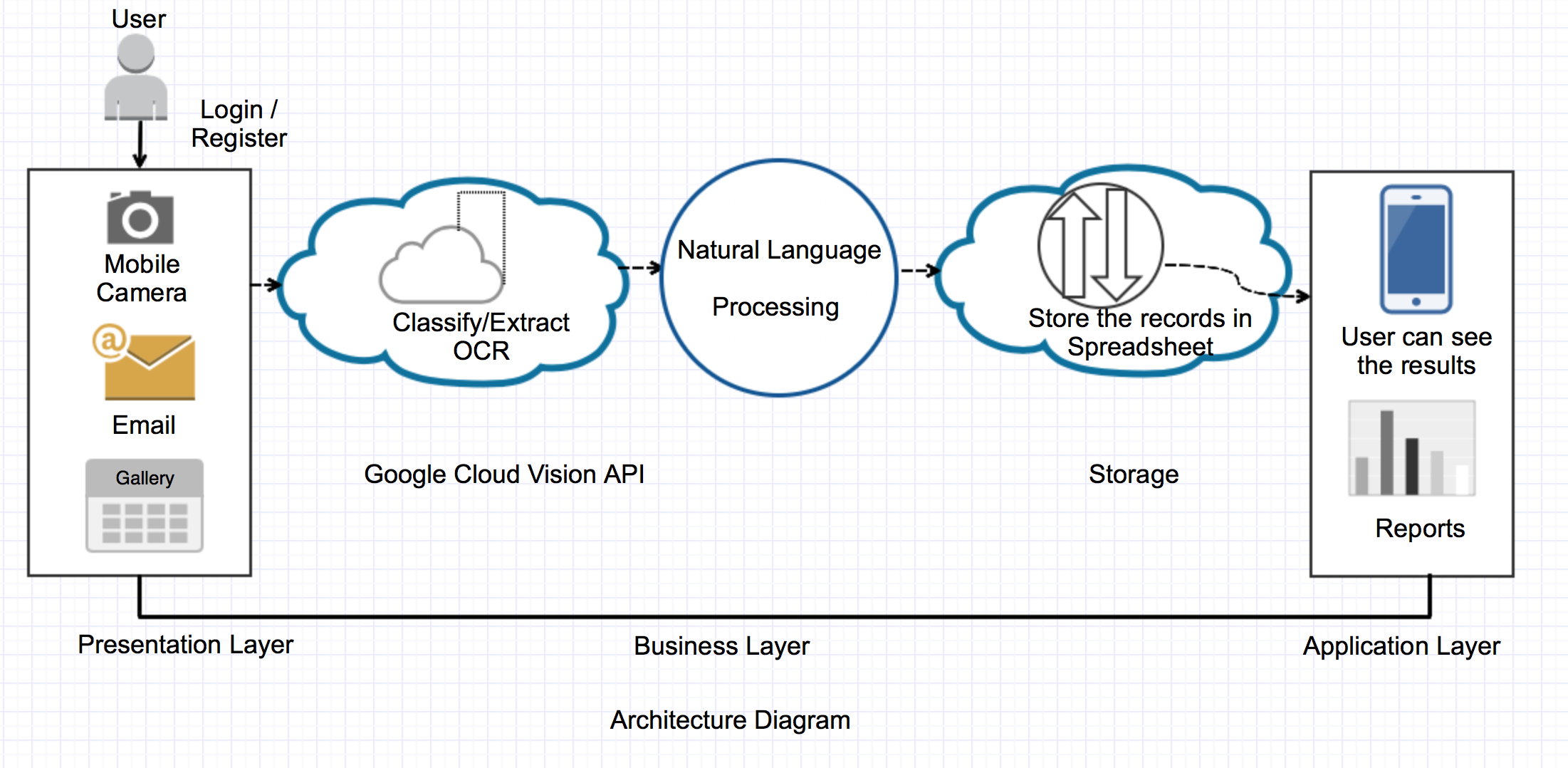
|  |  |  |  |
| --- | --- | --- | --- |
| Id | Requirements | Type | Priority |
| 1 | Layout App design using Wire frames and Architectural Diagram | Functional | High |
| 2 | Secure Login access integrating Fire base API into the app. | Functional | High |
| 3 | Integrating Google Cloud Vision OCR API into the app for extracting text through receipts. | Functional | High |
| 4 | Developing camera functionality to capture the receipts and store in the gallery to perform OCR. | Functional | Medium |
| 5 | Developing live camera algorithm to perform OCR through live screen. | Functional | High |
| 6 | Developing a functionality to reduce the size of the frames captured from the receipt and send it to perform OCR. | Functional | High |
| 7 | Developing an algorithm using regular expressions to categorize the extracted text through OCR. | Functional | High |
| 8 | Improving accuracy of categorization integrating Natural Language Processing API. | Functional | High |
| 9 | Developing “Speech to Text” Functionality to spot edit categorized text if necessary. | Functional | High |
| 10 | Integrating “Google Drive API” for secure storage of receipts | Functional | High |
| 11 | Developing a Dashboard Functionality for ease in Data Visualization. | Functional | High |
| 12 | User needs to scan the receipt using live camera, upload the image of saved receipt or extract it from google drive to perform OCR and storing receipts. | Non -Functional | High |
| 13 | User needs to verify the extracted text and save it if everything is as desired or modify it using speech to text, editable Text fields. | Non-Functional | High |
| 14 | User needs to verify their account before signing into the app for the first time. | Non-Functional | medium |
| 15 | User must provide a time frame to extract the stored receipt information and to provide data visualization using dashboard. | Non-Functional | medium |
| 16 | User must be able to access the receipt through link provided in dashboard. | Functional | High |
| 17 | Implementing Text Field Validation for optimal Functioning of the app. | Functional | High |

**SYSTEM ARCHITECTURE**

In this deliverable, the architecture diagram of the Receipt Free Scanner Application is presented. In addition, a detail process to determine the architecture that takes into account a set of differing inputs:

1. Functional requirements derived from end user requirements.
2. The original concept of the project and the project’s initial architectural principles and
3. The initial demonstration prototypes.

The aim of this effort was to ensure that the requirements and objectives of the project is fully addressed, and the architecture could deliver the required functionality.

A detailed process diagram is shown in Figure 1.  Figure 1: Architecture Diagram

The Architecture Diagram is divided into three layers: Presentation, Business and Application layer.

1. **Presentation Layer.**
   1. **User:** The one that will interact with the Android mobile phone by performing login/Register through application.

* **Registration:** This is the section that will be used by the app users to register into the app by registering via their email ID.
* **Log-in:** Once the user has registered into the app, they can then log-in to the app by using their login credentials (username & password) and also by using google account credentials.
  1. **The Camera /Gallery:** The user can take a picture of a text image or choose one from the mobile's directory.
* **Camera:** must have a certain resolution and must contain text.
* **Gallery (selected from the phone):** must contain text.
  1. **Email**: User can import receipts pictures via email.

1. **Business Layer**

Once the text of the image has been identified, it is then sent to OCR:

1. **Optical Character Recognition (OCR)**: By Using 'Deposit Receipt & Live Camera' feature, user can scan/capture receipt using phone camera. This feature allows users to convert any format document into text. This is quite a useful feature as when a user scans a document, the OCR will be able to detect it in real-time and will perform the actions according to the user activity.
2. **Natural Language Processing (NLP)**: Google Cloud Natural Language reveals the meaning of text by using Entity detection. Entity detection identifies entities and label by types such as person, organization, location.
3. **Storage & Spreadsheets**: Storage of receipts is done on external storage device as well as on SD card. Storage of receipts ultimately helps to organize the expense record efficiently. Final results are transformed into CSV files to itemized receipts.
4. **Application Layer**

In this layer user can view all reports and pie charts of all expense based on selected time frame.

**SYSTEM DESIGN**

Prior to building any application from scratch, it is important to plan the design flow of the application. The user interface design is essential to know how the application is intended to work. Planning an application involves some specific strategies.

Many strategic questions come to the mind while designing and planning an application like what’s the need of such an application in the market is, what features make your app unique, what will make the people use your app and not the ones designed by others, how much time the application will take to develop, what user requirements either small or big should be considered, what category should the app fall into, etc.

Wireframing is simply designing the flow of the application, describing how the navigation of the views will work. “Just in Mind” [1] is a very flexible tool for wire-framing an application. It becomes very essential to use such kind of tool to help design the app.

The design flow of the app is below: -

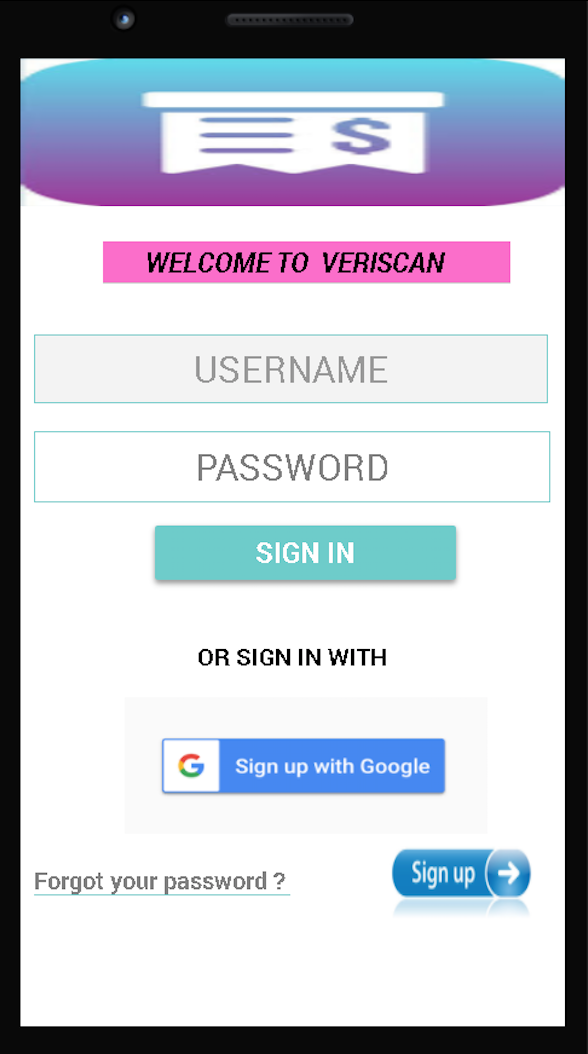


Figure 2: Login Screen

Step 1. Initially, user can sign-in to our application with their Google account and can register to our application with email id.



Figure 3: OCR Screen

Step 2. After successful login, a receipt can be selected from gallery if it is already stored or can be scanned using live camera. Optical Character Recognition (OCR) is performed on the receipt, then extracted text is categorized followed by receipt information classified into Name, Address, Time, Date and Total.

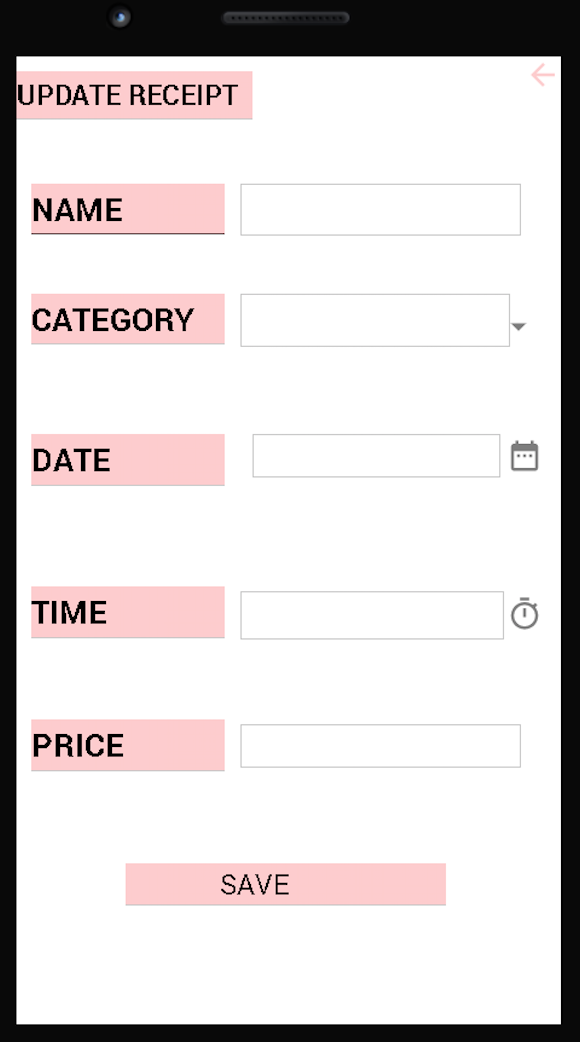


Figure 4: Categories Screen

Step 3. Now, user can check all the fields classified, edit if there are any changes and proceed to save and store details.

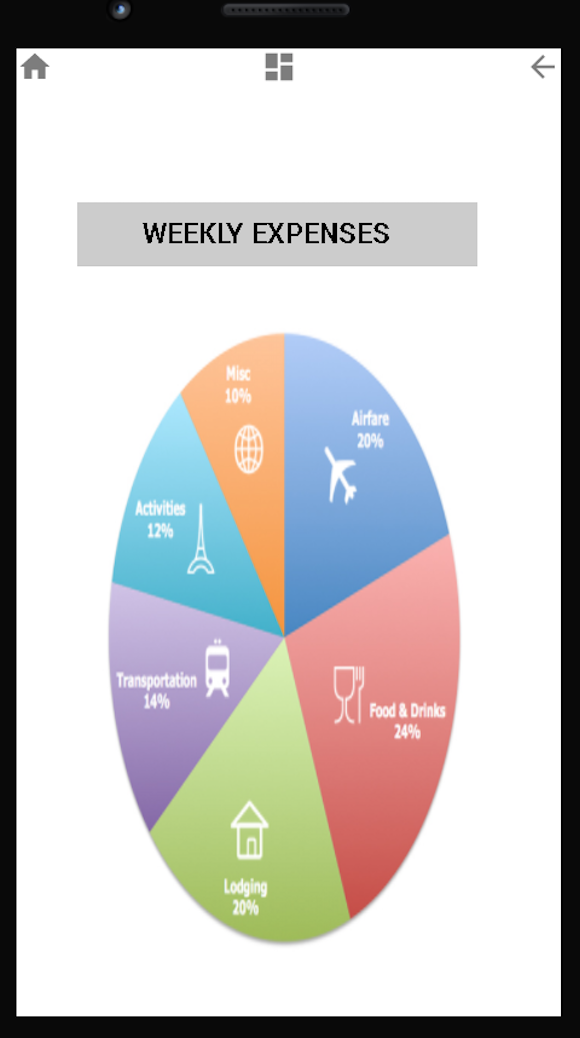


Figure 5: Result Screen

Step 4. Once the receipt information is stored into the spreadsheets, users can view all their reports and pie chart of all the expenses based on selected time frame if necessary.

**IMPLEMENTATION**

The whole process of building Receipt Free application with all the functionalities and the additional features integrated are explained in detail below:

**Firebase:**

Firebase is backend-as-a-service known as BAAS, app development platform built on google cloud platform. Firebase by default provides a great user experience developer can modify design accordingly. Firebase act as server, API and datastore. Firebase has two main features 1. Firebase Authentication 2. Firebase Database. For Authentication Purpose Firebase allows to login with Gmail, email, Facebook, Twitter and GitHub. This Application uses two options Gmail and Email. If user signs up with email he must provide first name, last name, mail id, password than verification link will be sent to their email id and using that user can successfully login. If user chooses to sign up with Gmail he must give all the permissions for login. Firebase stores the information into the cloud and functions only in real time environment thus, internet connection is mandatory.

**Optical Character Recognition (OCR) with Google Cloud Vision API:**

Optical Character Recognition (OCR) is a software technology that converts different types of documents, such as scanned paper documents, PDF files or images captured by a digital camera. It scans the documents by detecting and extracting the text from it.

The advantages of OCR are numerous, but namely it increases the efficiency and effectiveness of office work. The ability to instantly search through content is immensely useful, especially in an office setting that must deal with high volume scanning or high document inflow.

In this project, The Google Cloud Vision API [2] detect and extract text from images. The Text Detection annotation features supports OCR for extracting image from text. Google Cloud Vision API translate text and stores the results in Cloud Storage.

**Implementation Steps: -**

1. Initially, Application requires to creates the new project from Google Cloud Platform.
2. Application should enable the billing for the project selected.
3. It then Enables the Cloud Vision API on the Google Cloud Platform.
4. Application authenticates Cloud Vision API Key by using service account.
5. It makes a Vision API request to perform data analysis of image send in the request.
6. It Detects the Optical Character Recognition (OCR) on an image.
7. Application detects label and return identifying labels from an image.
8. It then sends the image data as base64-encoded text.

**Algorithm Steps: -**

Algorithm is designed using Regular expression for extracting Name, Address, Time, Total, Date from receipt.

1. To extract Name from the receipt image, Regular Expression “(^\\d{5}(?:[-\\s]\\d{4})?$)” looks for the pattern matching in the text and compares with the string each time.
2. To find Date various Regular Expression is used that process and converts each date in standard format “YYYY-MM-DD”. It checks for various format such as "(\\d{1,2}/\\d{1,2}/\\d{4})" ,\\w{3,}\\s\\d{1,2},?\\s\\d{4} matches dates and returns the result in one format.
3. For extracting Address Natural Language Processing (NLP) is integrated in the algorithm. Google Natural Language API provides natural language that help in analyzing text. Initially Natural Language Processing API Key is enabled in the project. Entity analysis [3] figures out entities name and type such as landmarks like restaurants on a text string and sends directly to the Natural Language API. With this analysis it gives precise location and details.
4. To get Time in form of AM and PM, time Matcher function finds the pattern in string. The algorithm process time independently and displays results.
5. For finding Total, Regular Expression checks for $ symbol at the end of each number. Algorithm finds out the highest number among all the numbers. With this it identifies the highest total value from the receipt.

**Live Camera Functionality:**

To perform OCR on the receipts the application needs camera access to scan the receipt or capture the receipt. Initially, during the testing phase a normal camera functionality is integrated into the app which can be utilized to capture the image of the receipt and store it in the gallery. Later, to perform OCR the image needs to be uploaded into the app through the gallery. This process was not as efficient as expected as it did not survive idealistic conditions of low light, skewed image etc. Further, live camera functionality has been integrated as an effective replacement for normal camera functionality where it can be utilized to perform OCR on the receipt through live screen. Though it was quite effective there was a slight issue in size of the frames in the image (8 MB initially). Thus, an algorithm which utilizes bit-map functionality was replaced into the live camera functionality. This algorithm just takes the best three frames avoiding first two frames (for user convenience) and sends it to Google Cloud Vision OCR for processing.

**Google Natural Language Processing:**

At times, OCR Algorithm doesn’t identify the Title on the receipt accurately. To overcome that issue this application uses Google Natural Language Processing. NLP is used for text analysis and annotation. There are several levels of analysis and each provides some valuable information. Three are various methods for text analysis such as Sentiment Analysis, Entity Analysis, Syntactic Analysis, Sentiment Entity Analysis, Content Analysis. For this application, Entity Analysis method was used which analyze the given text for known nouns and return its information.

After scanning the receipt, the extracted text is sent into the NLP method to analyze entities. This method looks for the entities in the data sources (Wikipedia, Google Knowledge Graph) if it finds noun in the data sources, it returns entities with other information such as type of entities (location, person, landmark), metadata (source information), salience score (how relevant the entity to the given text) and mentions(location in the text).

**Local Storage and Google Drive:**

As of now, Title, Total, Date, Time, Address, Category on the receipt have been captured using OCR and NLP. To create a report from this data the application creates the csv file. To create csv file this application uses Open Csv library which is third party library as java does not provide any support for handling csv file. OpenCsv is easy to use parser library from java. OpenCsv has classes such as CSV Parser, CSV Reader, CSV Writer, CSV TO Bean etc. CSV Writer servers the purpose of writing data on csv file. The Application allows user to save CSV file either on local storage or google drive. At last pdf of receipt will be stored on selected folder.

**Speech to Text:**

Android has in built Speech to Text Feature. Through this feature speech input is provided to this application. In background the speech input streams to a server and server converts the voice of the user for that time frame into text and sends to the application. OCR Algorithm and NLP uses that text to identify total, title, location, date, time and category.

**Dashboard and UI:**

The Receipt Free Android Application display end results of all reports and transaction in a selected time frame. The receipt data is visualized in the form of pie charts and table. It allows user to pick start and end date. Expense pie chart will show the categorization of receipt data. It provides more user accessibility for date query. Table provides all receipt information that helps user to look at all receipt data so far. Both pie chart and table are dynamic. If there is no receipt information for chosen date. It gives a message that no receipt data found for date range. This gives the user detailed information of all transactions made corresponding to all categories which in turns helps in optimization of accountability leading to successful financial management.

**MILESTONES**

**Key code:**

K = Kena Jalodara

L = Lovin Kuruvilla

M = Maneesh Mandava

N = Nidhi Parikh

**Table 1: Project Milestones**

|  |  |  |  |
| --- | --- | --- | --- |
| **Due Date** | **Activity** | **Percent Contributions** | **Status** |
| Sep 14 | Design | K (50), M (25), N (25) | Done: Sep 14 |
| Sep 17 | Project Website | L (50), M (25), N (25) | Done: Sep 17 |
| Sep 20 | Firebase | N (60), M (40) | Done: Sep 20 |
| Oct 5 | OCR | K (50), L (50) | Done: Oct 5 |
| Nov 2 | Live camera | K (50), M (50) | Done: Nov 2 |
| Nov 9 | Text Extraction Algorithm | L (100) | Done: Nov 9 |
| Nov 9 | NLP | K (100) | Done: Nov 9 |
| Nov 22 | Storage | N (50), M (25), K (25) | Done: Nov 22 |
| Nov 22 | Google Drive | M (100) | Done: Nov 22 |
| Nov 22 | Speech to text | L (100) | Done: Nov 22 |
| Nov 30 | Data Visualization | L (50), M (25), N (25) | Done: Nov 30 |
| Dec 3 | UI/UX | N (100) | Done: Dec 3 |
| Dec 5 | Presentations | M (50), N (30), K (20) | Done: Dec 5 |
| Dec 6 | Testing | L (50), N (30), K (20) | Done: Dec 6 |

**CONCLUSION**

***RECEIPT FREE***is an android application that helps users can receipt and store information in their local phone and in their Google Drive account. The application provides a secure session by authenticating the user and the data stored is confidential. The application can scan a receipt, identify key fields from the receipt, processes it and then store them. The implementation of data visualization tools such as color categorized pie chart and date querying can help the user identify receipts by its category and date. The application in future will also be implemented with a trained machine learning model to improve the accuracy of the receipt and its field identification. By using this application, the user will no longer have to keep track of their physical receipts, be more aware of their expenses and help form a better financial plan.

**FUTURE WORK**

The application was intended and targeted for android platform. One of the possible future advancement would be to transfer the application to an IOS platform or to a Windows platform. Doing this ensures that more users have access to the application and its functionalities. Key parts of the application code can be reworked to improve execution time and the accuracy with respect to correctly identifying the receipt fields. Both of these can be improved by modifying the current algorithm implemented. The algorithm can also be reconfigured so that it works with different regions and not just receipts printed in U.S.  For example, it would be beneficial to account for different date and currency formats used in other regions of the world. The camera is another aspect along with the implementation of a trained machine learning model that could help scan receipts that are handwritten, wrinkled or have other imperfections that result in an O.C.R. failure. Finally, querying by category can also be added to aid with data visualization for the user in addition to querying by date already present in the application.

**REFERENCES**

[1] https://www.justinmind.com

[2] https://cloud.google.com/functions/docs/tutorials/ocr

[3] https://cloud.google.com/natural-language/docs/analyzing-entities

[4] https://cloud.google.com/functions/docs/tutorials/ocr

[5] https://cloud.google.com/vision/docs/ocr

[6] https://cloud.google.com/natural-language/docs/basics

[7] https://www.google.com/amp/s/www.androidhive.info/2014/07/android-speech-to-text-

tutorial/amp/

[8] http://opencsv.sourceforge.net/

[9] https://www.google.com/amp/s/howtodoinjava.com/apache-commons/parse-read-

write-csv-opencsv/amp/

[10] https://www.skillsyouneed.com/rhubarb/personal-financial-management.html

**Appendix A – Code**

For some security reasons and the instructions given by our mentor the code is not shared here. A google drive link for the whole project apk file has been provided below strictly for academic purpose.

https://drive.google.com/open?id=1PgVRl\_O-fe5UuCBNVwKR\_f1Sq19fJ\_-4