

Final Report

Michigan Treasury Local Government Data Parser

Adham Abdalla ,
Hasan Alameh,
Tanis Daniels-Wanamaker ,
Ahmed Mawari

Summer 2021

Version number	Date	Author	Description of change
1.0	8/20/21	Adham Abdalla, Hasan Alameh , Tanis Daniels-Wanamaker, Ahmed Mawari	Document created

Acknowledgments

This document contains all the information on the development and beyond of the project “Michigan local government data parser” for Nicholas Brousseau and Stephanie Leiser. Their commitment and involvement in the project made it a pleasure to do.

Table of Contents

Acknowledgments	2
Introduction	5
1.1 Goals and Objectives	5
1.2 Statement of Scope	5
1.3 Software Context	5
1.4 Major Constraints	5
Requirements/Analysis Model	6
2.1 User Profiles	6
2.2 User Stories	6
2.3 Data Objects	6
2.4 Data Dictionary	7
2.5 Functional Model and Description	7
2.6 Behavioral Model and Description	11
Hardware/Software Design	13
3.1 Data Design	15
3.2 Architecture design	17
3.3 Component design	19
3.4 Software Interface Description	19
3.5 User Interface design	20
4.0 Implementation Details	22
5.0 Testing/SQA	22
5.1 Scope and intent of SQA activities	22
5.3 Task Overview	23
5.4 SQA Resources	23
5.5 Reviews and Audits	39

5.6 Problem Reporting and Corrective Action/Follow-up	40
5.7 Software Process Improvement Activities	41
5.8 Software Configuration Management Overview	41
5.9 SQA Tools, Techniques, Methods	42
6.0 Future Maintenance Suggestions	42
6.1 Maintaining the back-end parser	42
7.0 Delivery/Acceptance Statement	42
8.0 References and Bibliography	42
Appendix A – User Manual	42
Appendix B – Program Listing	44
Appendix C – Team Member Resumes	46
Appendix D – Project Plan and Log Book	50
Appendix E – Project Demo Notes	52
Appendix F – Final Presentation Slides	53

1.0 Introduction

1.1 Goals and Objectives

- Every year, local governments in Michigan upload their audited financial statements stating the different financial activities that occurred that year.
- Michigan Department of Treasury analysts are manually viewing these PDFs and finding specific information to fill out F65 forms.
- The financial statement PDFs have a wide range of varying formats with slightly different wording, numbering techniques, etc.
- Our program is designed to take a PDF as input, find the rows of interest from several tables, and append them into a comma-separated CSV file.

1.2 Statement of Scope

- Major Inputs
 - PDF file containing tables with data values
- Processing
 - PDF pages will be read starting at page 0
 - Every line that has no predefined keyword of interest will be immediately skipped
- Major Outputs
 - Lines that include a keyword of interest will be passed to a function that separates text from numbers and outputs the row to a CSV, with each value under its appropriate column

1.3 Software Context

- The client has specified which exact rows they are interested in extracting. Some of the rows may not exist in some formats, or they may be written under a different name. Our priority is to extract as much of the rows as possible.

1.4 Major Constraints

- One constraint is that of the type of PDF the program is dealing with. The program can not read text from a PDF that is scanned/contains images. This was mutually agreed on with the client.
- Another constraint is the variety of formats of PDFs. We have made the client aware that due to the number of different formats, the program may not extract every single row of interest on all PDFs. It is possible for the program to skip over some rows due to different naming and formatting conventions.

1.5 Tutorial

[Link to tutorial on YouTube](#)

2.0 Requirements/Analysis Model

2.1 User Profiles

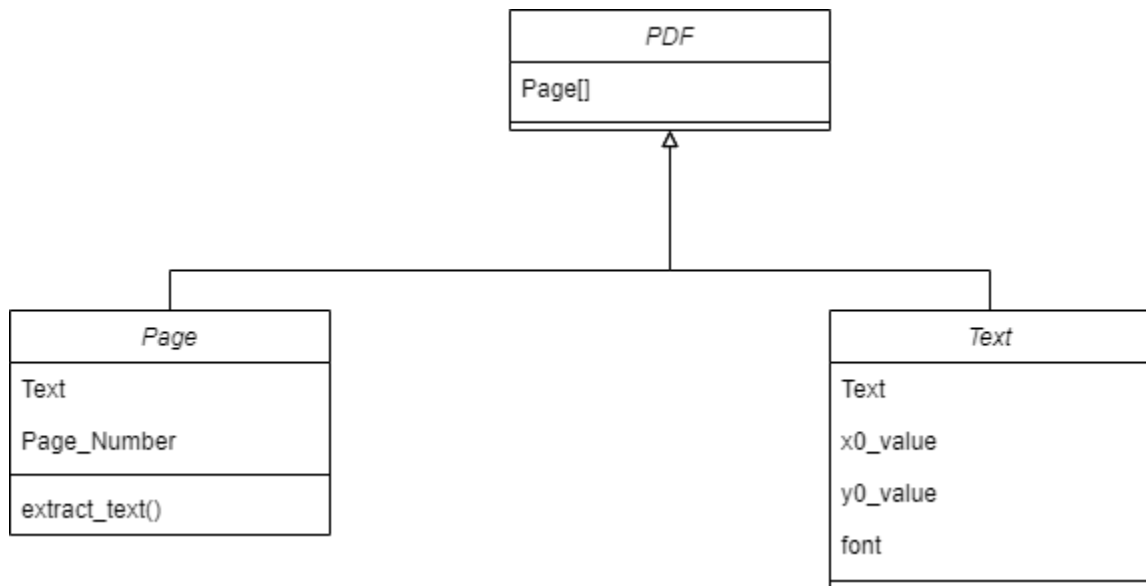
- Treasury Analyst: This user is interested in the program's output to the CSV file. They will download a PDF, input it into the program, and open the CSV file in Excel so they can build visualizations of the data provided.

2.2 User Stories

- Browse for PDF: The user clicks on the "Browse" button and selects a PDF file using the File Explorer.
- Submit for parsing: The user will click on the "Submit" button and wait for the parsing/extraction process to be completed.
- Open CSV: The user will either click on the "Open Master CSV" button or browse manually to the CSV file and open it to view the results.

2.3 Data Objects

- PDF file: Contains scannable, non-image pages
- CSV file: comma-separated list of rows and columns. Can be opened in Excel



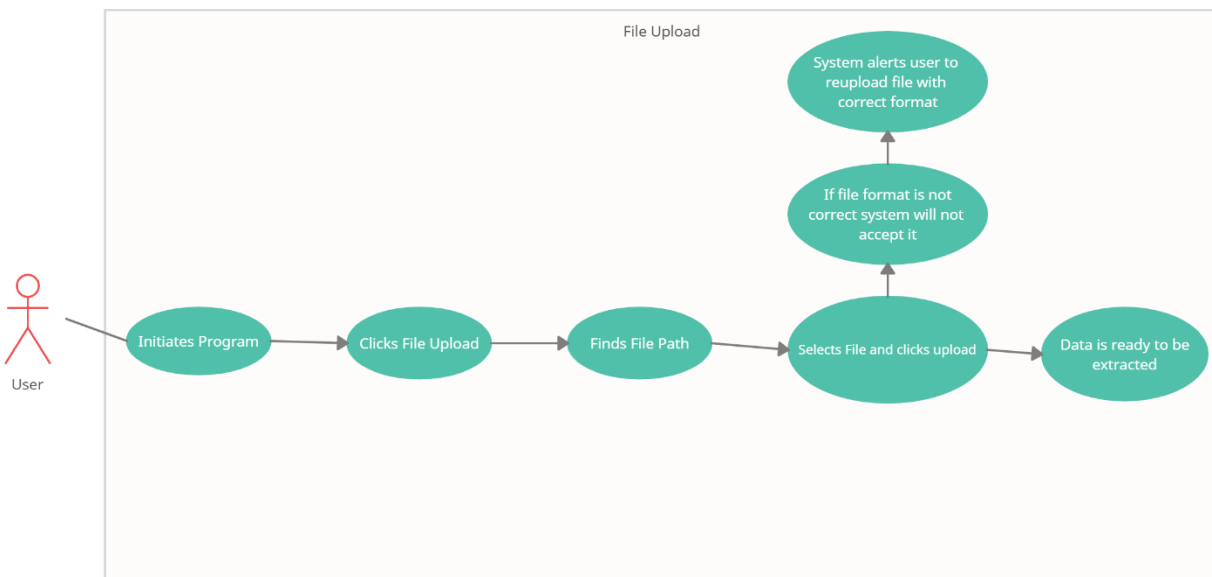
2.4 Data Dictionary

NAME	DESCRIPTION	USED BY (ACTORS)
1. File Upload	Explains how the user will upload a pdf to the program	User, System
2. Extract file data	Explains how the program will extract data from the file uploaded by the user	System
3. Generate Output File	Explains how the program will generate an output CSV file when user clicks download	User, System
4. Open Output File	Explains how the user can now open, view, and edit the output CSV file	User

2.5 Functional Model and Description

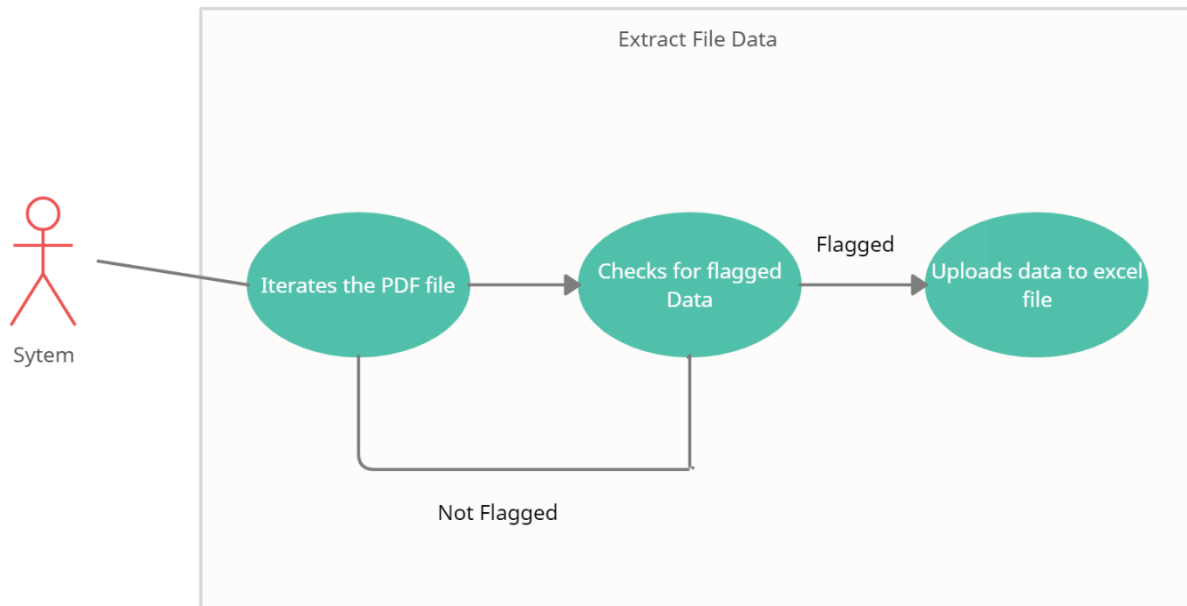
Function Point 1: File Upload

- Actors: User (Analyst)
- Preconditions: Scannable PDF, valid file path is specified
- Triggers: Submit button is clicked
- Scenario Description: User submits formatted PDF to be parsed
- Postconditions: File is ready to be parsed
- Exceptions: Scanned PDF (image), invalid file path



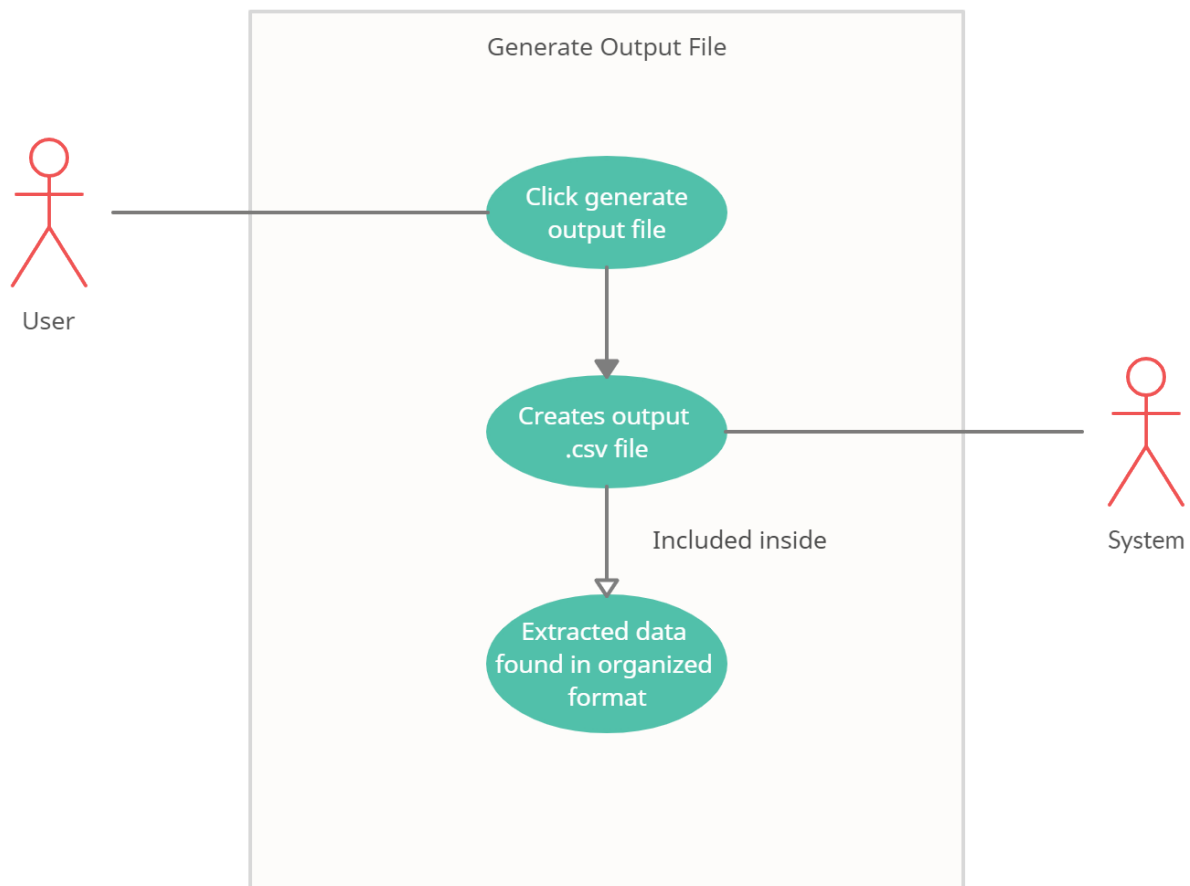
Function Point 2: Extract PDF Data

- Actors: System
- Preconditions: Valid PDF is submitted
- Triggers: Submit is clicked
- Scenario Description: Parser finds and saves rows containing keywords
- Postconditions: New rows (if available) are appended into output CSV
- Exceptions: Strings with no keywords



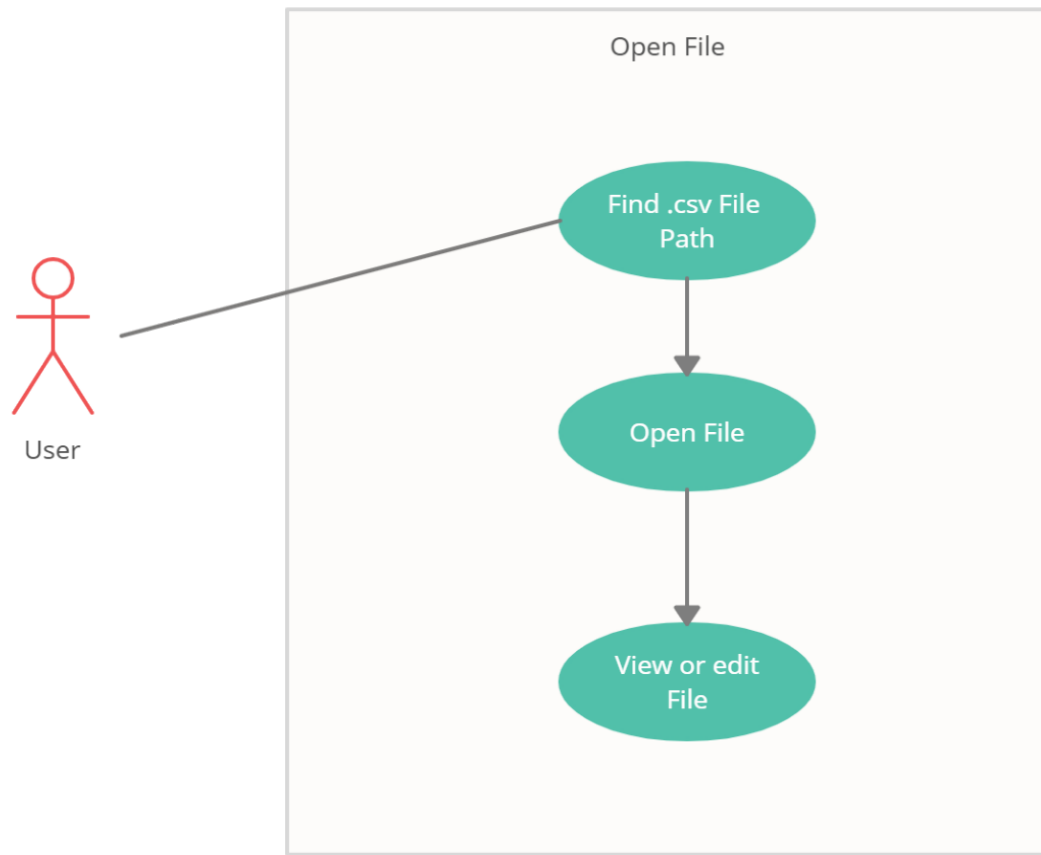
Function Point 3: Generate Output File

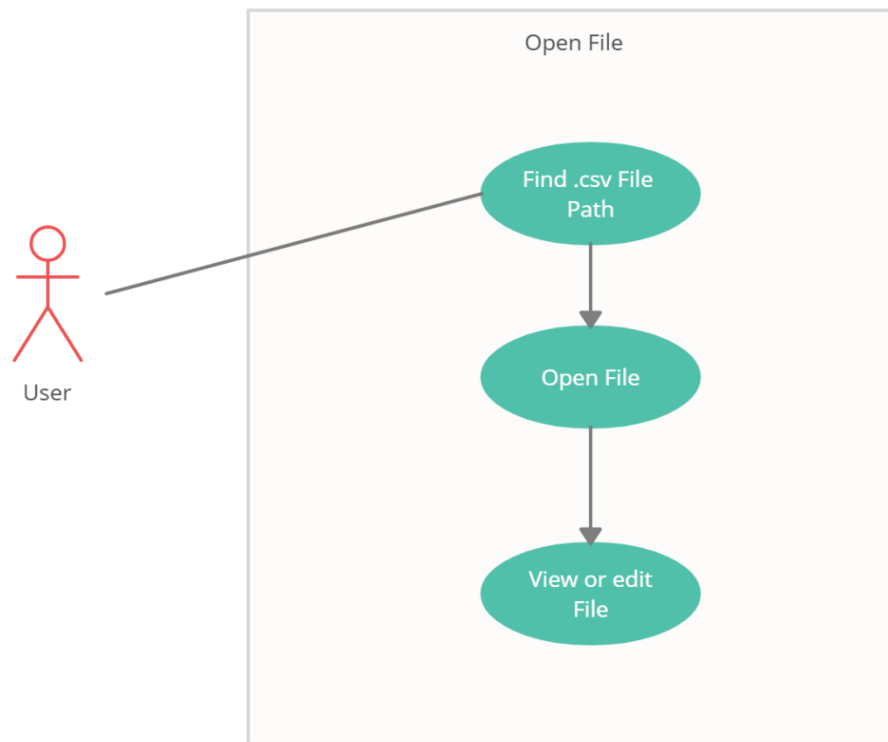
- Actors: User, System
- Preconditions: Data of interest is found in PDF and saved in data structure(s)
- Triggers: Data is to be outputted, but the output file does not exist
- Scenario Description: Program will create output.csv file and add necessary columns, correctly formatted
- Postconditions: Organized file is created
- Exceptions: CSV already exists, no data to output



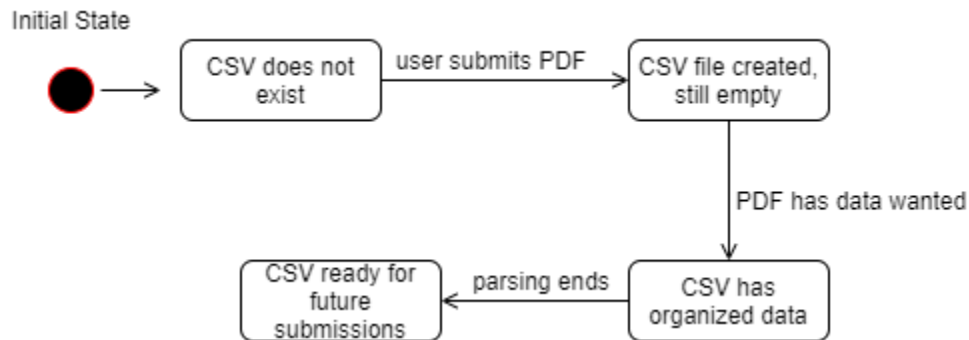
Function Point 4: Open Output File

- Actor: System, User
- Preconditions: Output file exists
- Triggers: User clicks “Open Output CSV”
- Scenario Description: CSV file will be opened using default CSV viewing program (usually Excel)
- Postconditions: CSV is opened and ready for viewing/editing
- Exceptions: CSV file does not exist

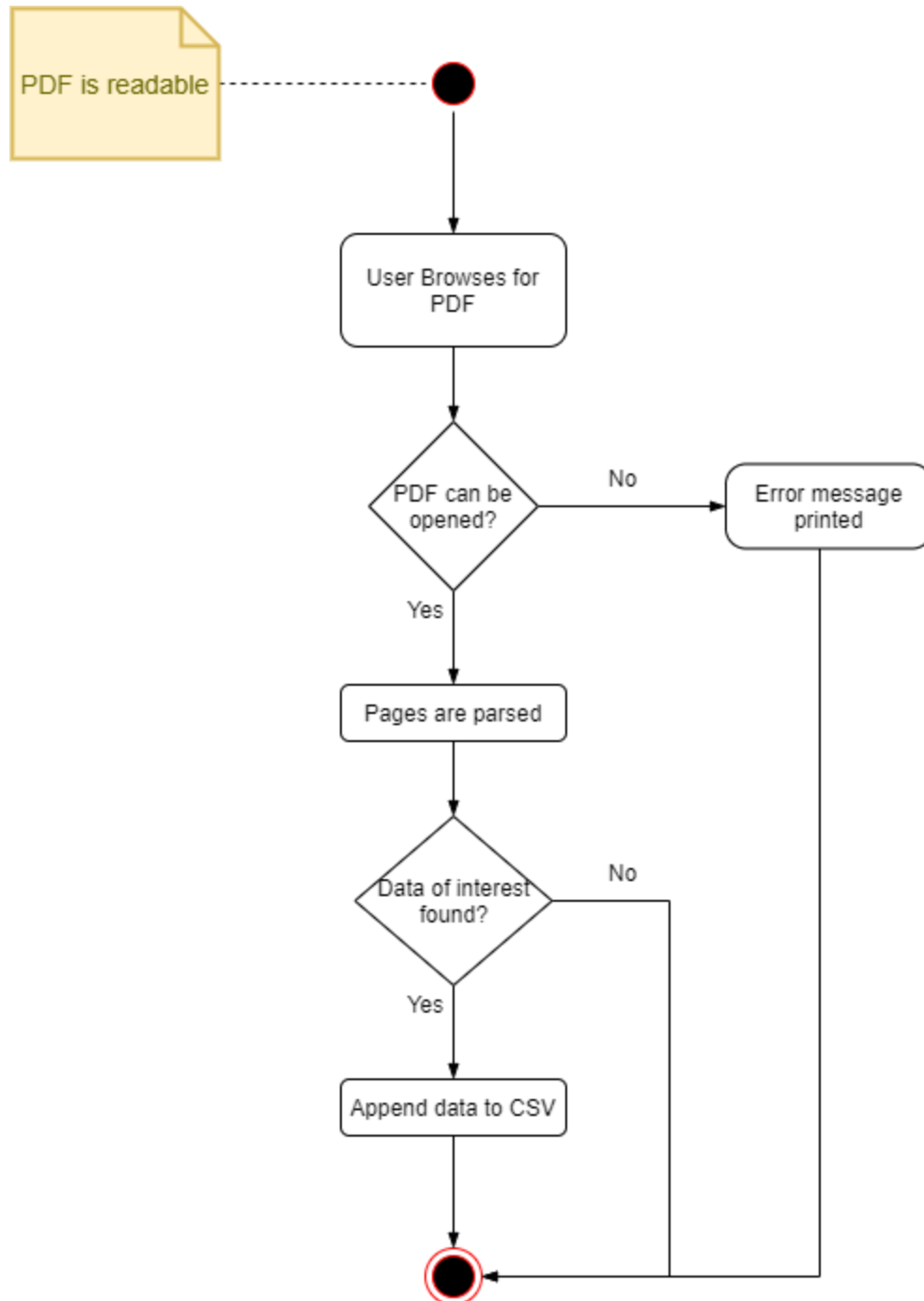




2.6 Behavioral Model and Description



State Diagram 1: Model showing the different states the CSV file goes through



Activity Diagram 1: Shows the flow of the program when a PDF is submitted

3.0 Hardware/Software Design

3.1 Data design

3.1.1 Internal software data structure

File Submitter:

File Submitter contains the following data objects:

- **FILE_NAME** - Holds the name of the file submitted.
- **FILE_PATH** - Holds the absolute path string of the file's location.
- **RESULT_PATH** - Holds the path to the resulting CSV file.
- **ROOT** - Holds the dimensions of the UI window to be opened.
- **TITLE** - Holds the title to be displayed for the UI window.
- **BUTTON** - Holds the command to be executed based on click.

File Parser:

File parser contains the following data objects:

- **HEADER_LIST** - Holds the specific category of the data to be extracted. These headers are written into the CSV.
- **CURR_PAGE** - Holds the reference to the current page the parser is iterating over.
- **CURR_TEXT** - Holds all textual data on the current page.
- **FILTERS** - Filters are used to keep track of various/multiple names the data appears as in the pdf.
 - **COUNTY_FILTER** - Keeps track of counties that may be formatted as simply county names or possibly the word 'county' appended to the county.
 - **STATEMENT_OF_ACTIVITIES_FILTER** - Keeps track of multiple formats which statement of activities may appear within the pdf.
 - **BUS_ACTIVITY_FILTER** - Keeps track of multiple formats which business activities may appear within the pdf.
 - **DATE_FILTER** - Keeps track of all date formats appearing in the pdf.
 - **GOV_ACTIVITY_FILTER** - Keeps track of multiple formats which gov activity filter may appear within the pdf.
 - **TOTAL_FILTER** - Keeps track of multiple formats which total may appear within the pdf.

- **COMPONENT_UNIT_FILTER** - Keeps track of multiple formats which component unit filter may appear within the pdf.
- **DF_LINE** - Holds tags of all columns of data to be extracted.
- **CURR_LINE_INDEX** - Holds the index of current line iteration on the page.
- **FULL_PAGE_TEXT** - Holds all textual data on this current page.
- **SPLITS** - Splits are used to hold data points which may appear as a total different name in the pdf.
 - **GRANTS_ROW_SPLIT** - Holds a reference to similar data points which may not be called GRANTS ROW.
 - **PROPERTY_TAX_SPLIT** - Holds a reference to similar data points which may not be called PROPERTY_TAX.
 - **TOTAL_GEN_REV_TRANSFER_SPLIT** - Holds a reference to similar data points which may not be called TOTAL GEN REV TRANSFER.
 - **CHANGE_IN_NET_POS_SPLIT** - Holds a reference to similar data points which may not be called CHANGE_IN_NET_POS.

3.1.2 Global data structure

- **FINAL_DATA** - This is a global component which appends all flagged parsed data from the PDF. It holds parsed data which then will be converted into a dataframe for permanent storage throughout execution.
- **Tkinter Root** - This global root controls all operations of the software. It handles events on click, on transitions, and on changes.

3.1.3 Temporary data structure

N/A, As of now we are not using any temporary data structure.

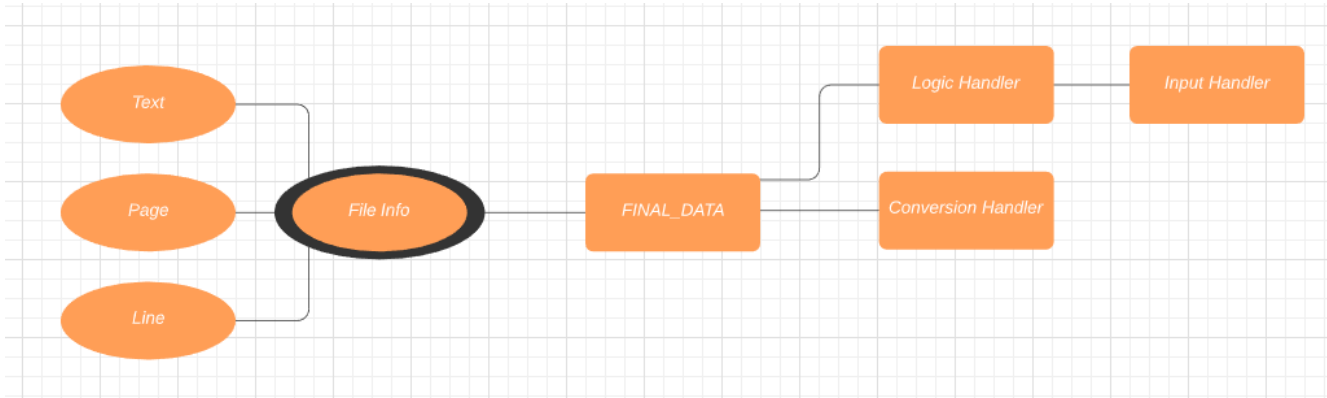
3.1.4 Database description

- **DATAFRAME** - We decided to use a dataframe rather than a database. It reduces steps in order to transfer data to a CSV.
 - **DATAFRAME_COLUMNS:** Columns hold the specific type of data as displayed on the column header.
 - **DATAFRAME_ROWS:** Rows hold a number of records of that data throughout the pdf.

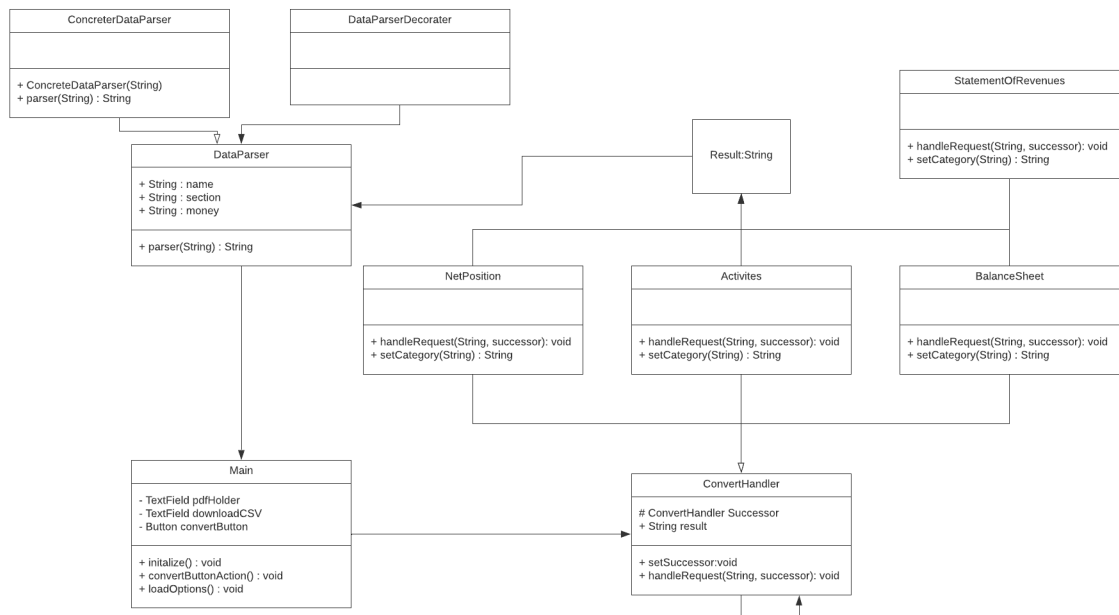
3.2 Architectural and component-level design

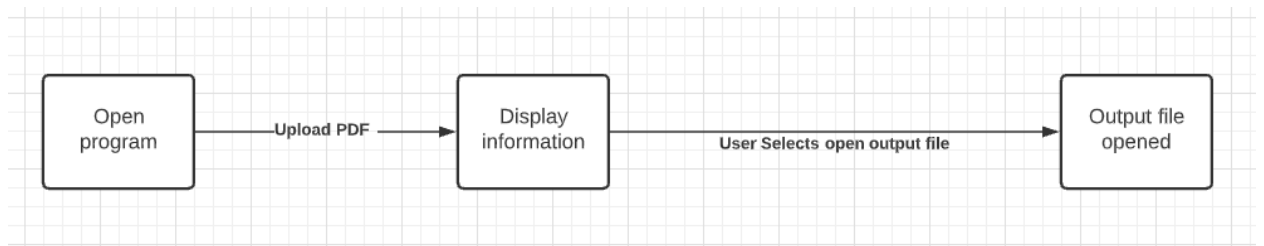
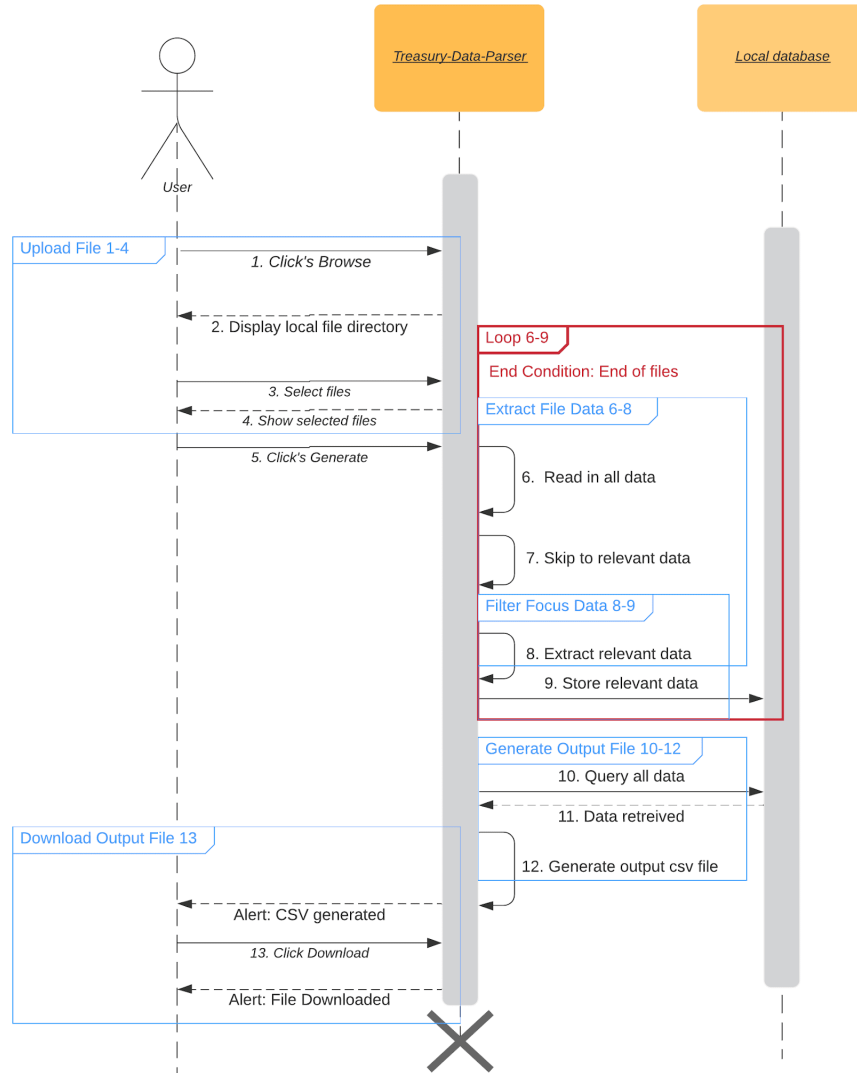
3.2.1 Program Structure

3.2.1.1 Architecture diagram



3.2.1.2 Alternatives





3.3 Component Design

3.3.1 Processing narrative (PSPEC)

File Info: A combination of all info gathered from the PDF

Text: The specific text at any given time being gathered

Page: The page that our program is currently scanning and gathering from

Line: The specific line that our program is reading

Logic Handler: Handles the logic of the program and makes sure that it keeps moving through the document

Conversion Handler: Makes sure that we are properly converting the PDF text into usable information

Input Handler: Handles the PDF that is being entered by the user

FINAL_DATA: The heart of it all, this handles everything

3.3.2 Component and interface description.

File Info: Outputting all the information given by the PDF to be used by the Conversion Handler

Text: Outputs the text gathered from the file

Page: Outputs to the program which page we are currently on

Line: Outputs the current line our program is reading to be broken up

Logic Handler: Takes in all the information gathered from the file

Conversion Handler: Takes in all the information from the logic handler and outputs it onto a CSV file

Input Handler: Takes in the PDF itself

FINAL_DATA: Takes in all the information of the program and outputs it back to the user

3.3.3 Sub-Component n.m processing detail

File Info: Gathers info from text, page, and line then sends it to FINAL_DATA

Text: Gathers text

Page: Gathers page number

Line: Gathers current line of PDF

Logic Handler: Takes input PDF and sends it off to FINAL DATA

Conversion Handler: takes FINAL DATA information and formats it, sends back CSV file to FINAL DATA

Input Handler: Takes in a PDF from user and sends it ot Logic Handler

FINAL_DATA: Gathers all things and sends all things.

3.3.3.1 Interface description.

File Info: No display

Text: No display

Page: No display

Line:No display

Logic Handler: No display

Conversion Handler: No display

Input Handler: No display

FINAL_DATA: Displays the CSV download to the user

3.3.3.2 Restrictions/limitations

File Info: PDF must be correctly formatted

Text: PDF must be correctly formatted

Page: PDF must be correctly formatted

Line: PDF must be correctly formatted

Logic Handler: PDF must be correctly formatted

Conversion Handler: PDF must be correctly formatted

Input Handler: PDF must be correctly formatted

FINAL_DATA: PDF must be correctly formatted

3.3.3.3 Performance issues

File Info: N/A

Text: N/A

Page: N/A

Line: N/A

Logic Handler: N/A

Conversion Handler: N/A

Input Handler: N/A

FINAL_DATA: N/A

3.3.3.4 Design constraints

File Info: Needs a properly formatted PDF

Text: Needs a properly formatted PDF

Page: Needs a properly formatted PDF

Line: Needs a properly formatted PDF

Logic Handler: Needs a properly formatted PDF

Conversion Handler: Needs a properly formatted PDF

Input Handler: Needs a properly formatted PDF

FINAL_DATA: Needs a properly formatted PDF

3.4 Software Interface Description

3.4.1 External machine interfaces

N/A This will only be held on one machine

3.4.2 External system interfaces

N/A This will be a self sufficient program

3.4.3 Human interface

2 Screens will be made for users to view and interact with.

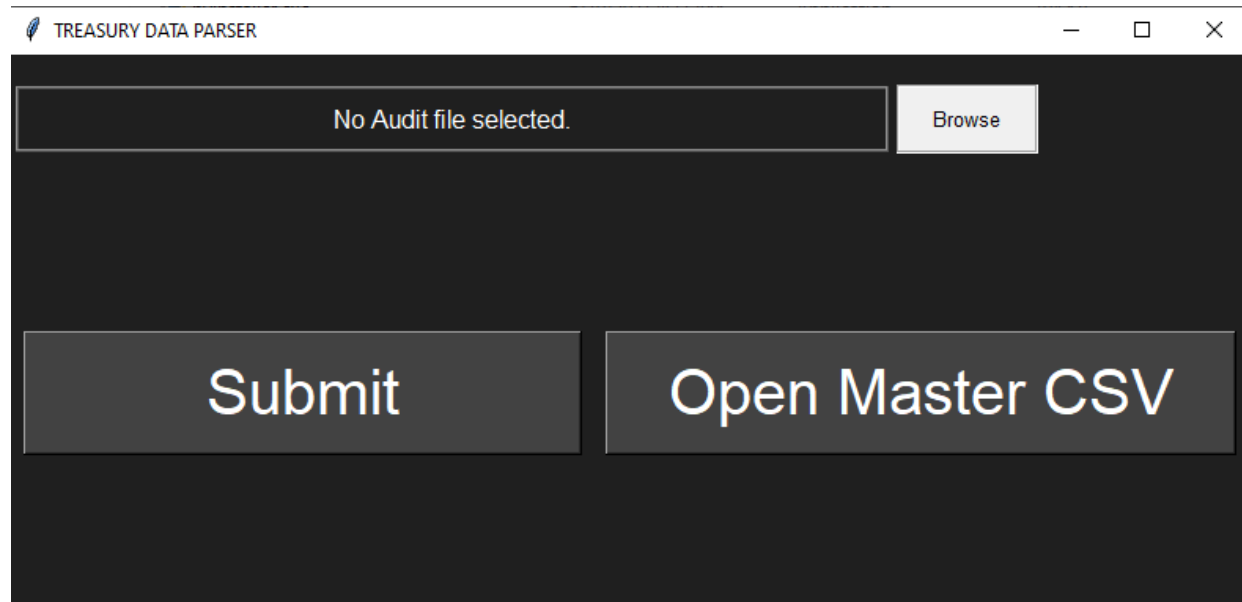
3.5 User interface design

3.5.1 Description of the user interface

We should have only 2 screens in our final program

3.5.1.1 Screen images

The image below is the user interface for selecting a pdf file through windows file explorer , submitting the file and opening the master output file .



The image below shows a console window of the program showing a trace of the data extraction as requested by the client.

```

C:\Windows\py.exe
=====USER INSTRUCTIONS=====
- CLOSE THE MASTER CSV FILE PRIOR TO RUNNING THE PROGRAM, program cannot write to an open CSV file.
- The program can ONLY handle readable PDF files (not image PDFs, textual only), therefore
please check before you feed in the files to the system.
- Browse Button - This is used to navigate to your PDF file & get the path.
- Submit Button - This is used to initiate the parser.
- Open Button - This is used to open the CSV file after the data has been parsed and transferred into it.
- NOTE: The CSV file is rewritten everytime the tool is ran,
which means the old data is wiped out and new data is written based on the PDF uploaded.
Therefore, it's best to save the CSV elsewhere before trying a second run.
=====

```

3.5.1.2 Objects and actions

- 1st Screen
 - Browse button: Will allow the user to search their computer directory for the PDF to enter.
 - Submit button: Will start processing the data of the imputed PDF.

- Open master output file: will allow the user to open the CSV output file.
- 2nd screen
 - This a console window of the program trace.

3.5.2 Interface design rules

- Python -> TKinter Module was used to create UI screens.
- Easy to Learn
- Readability
- Easy navigate between interfaces

3.5.3 Components available

- TextBox
 - Displays information entered at design time, entered by the user
- Label
 - Graphical area that a user won't be able to change directly
- Line
 - Displays a line of text
- CommandButton
 - A button used to invoke an action out of the program directly, such as finding a file in your computer when you press the browse button.

3.5.4 UI description

The user interface will be developed based on what our code needs, if our code needs a certain file then it will ask the user for it. It will be designed with simplicity and absolute no chance of messing it up with an average user in mind. Only the necessity of information will be displayed as users will not need anything else past the CSV from the PDF that they have entered.

4.0 Implementation Details

- **Resource Limitation**

- Not having appropriate libraries or modules is a limitation for us. During the early stages of prototype development, we were looking for modules which could be helpful for parsing a PDF, however there aren't many in the market. We are limited to one or two which are open & free to use. The third party resources we found offer limited functionalities since they are still under full development.

- **Team Skills**

- Not having much coding experience with this sort of tool among us as a team is a set back. Plus due to limited resources offered in the market, we are constrained to using one programming language, which most of us are not fully comfortable with.

- **Input File Constraint**

- We are constrained to use one PDF file provided by the client. Our software is pretty much relying on that one PDF to measure successful parsing. As mentioned by the client, the software must successfully function with the given PDF for approval. Ideally, the software should handle other PDFs. But we are limited to code based on the formatting of the given PDF file.

5.0 Testing/SQA

5.1 Reviews and Audits

5.2 Quality Tracking

The measures of our program's quality are the following:

- How many rows of interest were extracted from the PDF
- Whether the rows have the text properly separated from the values
- Whether the values are under the proper columns

These measures were what we were observing during the testing phase of our project. The clients also helped track the program's quality by giving their opinions on the output values.

5.3 Testing strategy

5.3.1 Unit testing

The main testing strategy to be used is brainstorming the possible main test cases with their expected outputs, then running the test through the program to see whether the results are as expected.

The program outputs a CSV file based on the PDF the user inputs, so the components tested will involve the PDF, program, and output file.

5.3.2 Integration testing

We will input a PDF, submit it, and have the CSV file open for viewing the results.

5.3.3 Validation testing

For validation, we will use unit testing and performance validation. This will help us evaluate the program's reliability.

5.3.4 High-order testing

High-order testing is the next step of module testing. Unit testing, integration testing, validation, and performance testing all ensure the fulfillment of client requirements and criteria.

5.4 Testing procedures

5.4.1 Test case log

BALDWIN TOWNSHIP

Test Desc	Expected Output	Actual Output
Extract township name	Baldwin Township	BALDWIN TOWNSHIP
Extract date	3/31/2020	3/31/2020
Extract Net Position:		
Govt. Activities		
Cash	480856	480856
Capital assets	21.841	21.841
Total assets	526.813	526.813
Net investment in capital assets	21841	21841
Unrestricted	180.202	180.202
Total net position	526813	526813
Extract Statement of Activities:		

Govt. Activities		
Property taxes	169708	169708
Change in net position	57682	57682
Extract Balance Sheet:		
General fund:		
Cash:	134960	134960
Fund Balances: Restricted	-	324452
Fund Balances: Committed	-	27319
Fund Balances: Unassigned	148.117	148.117
Total Fund Balances	148.117	148.117
Total government funds:		
Cash:	480856	480856
Fund Balances: Restricted	324770	324770
Fund Balances: Committed	27319	27319
Fund Balances: Unassigned	148.117	148.117
Total Fund Balances	500.206	324.452
TOTAL ENTERPRISE FUNDS:		
Total Fund Balances		27.637
???:		
Total Fund Balances		500.206

BROOKFIELD TOWNSHIP

Test Desc	Expected Output	Actual Output
Extract township name	BROOKFIELD TOWNSHIP	BROOKFIELD TOWNSHIP
Extract date	3/31/2020	3/31/2020
Extract Net Position:		
Govt. Activities		
CASH AND CASH EQUIVALENTS	107784	107784
TOTAL CURRENT ASSETS	132469	132469
TOTAL CURRENT ASSETS - NONCURRENT ASSETS - CAPITAL ASSETS, NET	1900	1900
TOTAL ASSETS	134369	134369
TOTAL CURRENT LIABILITIES	37565	37565
TOTAL CURRENT LIABILITIES - NONCURRENT PORTION OF LONG-TERM DEBT	-	-
TOTAL NONCURRENT LIABILITIES	-	-

TOTAL LIABILITIES	37565	37565
UNRESTRICTED	94904	94904
TOTAL NET POSITION	96804	96804
TOTAL NET POSITION - SEE ACCOMPANYING NOTES WHICH ARE AN INTEGRAL PART OF THE FINANCIAL STATEMENTS.	N/A	-
Business Activites		
CASH AND CASH EQUIVALENTS	48470	48470
TOTAL CURRENT ASSETS	110713	110713
TOTAL CURRENT ASSETS - NONCURRENT ASSETS - CAPITAL ASSETS, NET	2544751	2544751
TOTAL ASSETS	2655464	2655464
TOTAL CURRENT LIABILITIES	172098	172098
TOTAL CURRENT LIABILITIES - NONCURRENT PORTION OF LONG-TERM DEBT	820700	820700
TOTAL NONCURRENT LIABILITIES	820700	820700
TOTAL LIABILITIES	992798	992798
UNRESTRICTED	109328	109328
TOTAL NET POSITION	1662666	1662666
TOTAL NET POSITION - SEE ACCOMPANYING NOTES WHICH ARE AN INTEGRAL PART OF THE FINANCIAL STATEMENTS.	N/A	-
Total		
CASH AND CASH EQUIVALENTS	156254	156254
TOTAL CURRENT ASSETS	243182	243182
TOTAL CURRENT ASSETS - NONCURRENT ASSETS - CAPITAL ASSETS, NET	2546651	2546651
TOTAL ASSETS	2789833	2789833
TOTAL CURRENT LIABILITIES	209663	209663
TOTAL CURRENT LIABILITIES - NONCURRENT PORTION OF LONG-TERM DEBT	820700	820700
TOTAL NONCURRENT LIABILITIES	820700	820700
TOTAL LIABILITIES	1030363	1030363
UNRESTRICTED	204232	204232
TOTAL NET POSITION	1759470	1759470

TOTAL NET POSITION - SEE ACCOMPANYING NOTES WHICH ARE AN INTEGRAL PART OF THE FINANCIAL STATEMENTS.	N/A	9-
Extract Statement of Activities:		
Govt. Activities		
Total governmental activities	-237010	-
Property taxes	60499	60499
State-shared revenues	141921	141921
Total general revenues	203516	203516
Change in net position	-33494	-33494
Business Activities		
Total governmental activities	-	-237010
Property taxes	-	-
State-shared revenues	-	-
Total general revenues	1046	1046
Change in net position	108053	108053
Total		
Total governmental activities	237010	
Property taxes	60499	60499
State-shared revenues	141921	141921
Total general revenues	204562	204562
Change in net position	74559	74559

BEAR CREEK TOWNSHIP

Test Desc	Expected Output	Actual Output
Extract township name	BEAR CREEK TOWNSHIP	BEAR CREEK TOWNSHIP
Extract date	3/31/2020	3/31/2020
Extract Net Position:		
Govt. Activities		
CASH AND CASH EQUIVALENTS	2851061	2851061
INVESTMENTS	2253600	2253600
CAPITAL ASSETS - NET OF DEPRECIATION	5911234	5911234
TOTAL ASSETS	11148631	11148631
DUE TO OTHER GOVERNMENTAL UNITS - DUE WITHIN ONE YEAR	-	-
NONCURRENT LIABILITY - DUE TO THE CITY OF PETOSKEY - DUE IN MORE THAN ONE YEAR	-	-

TOTAL LIABILITIES	45955	45955
UNRESTRICTED	3705907	3705907
TOTAL NET POSITION	11102676	11102676
Business-Type Activities		
CASH AND CASH EQUIVALENTS	711418	711418
INVESTMENTS	2222634	2222634
CAPITAL ASSETS - NET OF DEPRECIATION	2249540	2249540
TOTAL ASSETS	7160554	7160554
DUE TO OTHER GOVERNMENTAL UNITS - DUE WITHIN ONE YEAR	-	120309
NONCURRENT LIABILITY - DUE TO THE CITY OF PETOSKEY - DUE IN MORE THAN ONE YEAR	1489340	1489340
TOTAL LIABILITIES	1680015	1680015
UNRESTRICTED	3230999	3230999
TOTAL NET POSITION	5480539	5480539
Total		
CASH AND CASH EQUIVALENTS	3562479	3562479
INVESTMENTS	4476234	4476234
CAPITAL ASSETS - NET OF DEPRECIATION	8160774	8160774
TOTAL ASSETS	18309185	18309185
DUE TO OTHER GOVERNMENTAL UNITS - DUE WITHIN ONE YEAR	-	120309
NONCURRENT LIABILITY - DUE TO THE CITY OF PETOSKEY - DUE IN MORE THAN ONE YEAR	1489340	1489340
TOTAL LIABILITIES	1725970	1725970
UNRESTRICTED	6936906	6936906
TOTAL NET POSITION	16583215	16583215
Component Units		
CASH AND CASH EQUIVALENTS	283049	283049
INVESTMENTS	N/A	
CAPITAL ASSETS - NET OF DEPRECIATION	20139	20139
TOTAL ASSETS	352545	352545
DUE TO OTHER GOVERNMENTAL UNITS - DUE WITHIN ONE YEAR	80000	-
NONCURRENT LIABILITY - DUE TO THE CITY OF PETOSKEY - DUE IN MORE THAN ONE YEAR	-	-

TOTAL LIABILITIES	99251	99251
UNRESTRICTED	-	-
TOTAL NET POSITION	253294	253294
Extract Statement of Activities:		
Govt. Activities		
TOTAL BUSINESS-TYPE ACTIVITIES	-	-35964
PROPERTY TAXES	1332726	1332726
PROPERTY TAX ADMINISTRATION FEE	177181	177181
STATE REVENUE SHARING	555673	555673
TOTAL GENERAL REVENUES	2174379	2174379
CHANGE IN NET POSITION	393772	393772
Business-Type Activities		
TOTAL BUSINESS-TYPE ACTIVITIES	-35964	-35964
PROPERTY TAXES	-	-
PROPERTY TAX ADMINISTRATION FEE	-	-
STATE REVENUE SHARING	-	-
TOTAL GENERAL REVENUES	43266	43266
CHANGE IN NET POSITION	7302	7302
Total		
TOTAL BUSINESS-TYPE ACTIVITIES	-35964	-
PROPERTY TAXES	1332726	1332726
PROPERTY TAX ADMINISTRATION FEE	177181	177181
STATE REVENUE SHARING	555673	555673
TOTAL GENERAL REVENUES	2217645	2217645
CHANGE IN NET POSITION	401074	401074
Component Units		
TOTAL BUSINESS-TYPE ACTIVITIES	-	
PROPERTY TAXES	-	-
PROPERTY TAX ADMINISTRATION FEE	-	-
STATE REVENUE SHARING	-	-
TOTAL GENERAL REVENUES	-	-
CHANGE IN NET POSITION	-68683	-68683

ARGENTINE TOWNSHIP

Test Desc	Expected Output	Actual Output
-----------	-----------------	---------------

Extract township name	Argentine Township	ARGENTINE TOWNSHIP
Extract date	6/30/2020	6/30/2020
Extract Net Position:		
Govt. Activities		
CASH AND CASH EQUIVALENTS	3955763	3955763
CAPITAL ASSETS NOT BEING DEPRECIATED	45000	45000
CAPITAL ASSETS, NET OF ACCUMULATED DEPRECIATION	597694	597694
TOTAL ASSETS	5651389	5651389
NONCURRENT LIABILITIES - DEBT DUE WITHIN ONE YEAR	73000	73000
NONCURRENT LIABILITIES - DEBT DUE IN MORE THAN ONE YEAR	560000	560000
TOTAL LIABILITIES	731745	731745
INVESTMENT IN CAPITAL ASSETS	642694	642694
UNRESTRICTED	1237833	1237833
TOTAL NET POSITION	4919644	4919644
UNRESTRICTED - TOTAL NET POSITION - SEE ACCOMPANYING NOTES TO THE FINANCIAL STATEMENTS	N/A	-
Business-Type Activities		
CASH AND CASH EQUIVALENTS	1145802	1145802
CAPITAL ASSETS NOT BEING DEPRECIATED	31525	31525
CAPITAL ASSETS, NET OF ACCUMULATED DEPRECIATION	766052	766052
TOTAL ASSETS	2429107	2429107
NONCURRENT LIABILITIES - DEBT DUE WITHIN ONE YEAR	-	-
NONCURRENT LIABILITIES - DEBT DUE IN MORE THAN ONE YEAR	-	-
TOTAL LIABILITIES	26970	26970
INVESTMENT IN CAPITAL ASSETS	797577	797577
UNRESTRICTED	1604560	1604560
TOTAL NET POSITION	2402137	2402137
UNRESTRICTED - TOTAL NET POSITION - SEE ACCOMPANYING NOTES TO THE FINANCIAL STATEMENTS	N/A	4-
Total		
CASH AND CASH EQUIVALENTS	5101565	5101565

CAPITAL ASSETS NOT BEING DEPRECIATED	76525	76525
CAPITAL ASSETS, NET OF ACCUMULATED DEPRECIATION	1363746	1363746
TOTAL ASSETS	8080496	8080496
NONCURRENT LIABILITIES - DEBT DUE WITHIN ONE YEAR	73000	73000
NONCURRENT LIABILITIES - DEBT DUE IN MORE THAN ONE YEAR	560000	560000
TOTAL LIABILITIES	758715	758715
INVESTMENT IN CAPITAL ASSETS	1440271	1440271
UNRESTRICTED	2842393	2842393
TOTAL NET POSITION	7321781	7321781
UNRESTRICTED - TOTAL NET POSITION - SEE ACCOMPANYING NOTES TO THE FINANCIAL STATEMENTS	N/A	1
Extract Statement of Activities:		
Govt. Activities		
TOTAL GOVERNMENTAL ACTIVITIES	-675083	-
TOTAL PRIMARY GOVERNMENT	-675083	53011
PROPERTY TAXES	244698	244698
STATE SHARED REVENUE	584081	584081
TOTAL GENERAL REVENUES AND TRANSFERS	1029401	1029401
CHANGE IN NET POSITION	354318	354318
Business-Type Activities		
TOTAL GOVERNMENTAL ACTIVITIES	-	-675083
TOTAL PRIMARY GOVERNMENT	53011	-622072
PROPERTY TAXES	-	-
STATE SHARED REVENUE	-	-
TOTAL GENERAL REVENUES AND TRANSFERS	-7284	-7284
CHANGE IN NET POSITION	45727	45727
Total		
TOTAL GOVERNMENTAL ACTIVITIES	-675083	
TOTAL PRIMARY GOVERNMENT	-622072	
PROPERTY TAXES	244698	244698
STATE SHARED REVENUE	584081	584081

TOTAL GENERAL REVENUES AND TRANSFERS	1022117	1022117
CHANGE IN NET POSITION	400045	400045
Expenses		
TOTAL GOVERNMENTAL ACTIVITIES	2956869	2956869
TOTAL PRIMARY GOVERNMENT	3764876	3764876
Charges for Services		
TOTAL GOVERNMENTAL ACTIVITIES	523336	523336
TOTAL PRIMARY GOVERNMENT	1384354	1384354
Operating Grants & Contributions		
TOTAL GOVERNMENTAL ACTIVITIES	1700185	1700185
TOTAL PRIMARY GOVERNMENT	1700185	1700185
Captial Grants & Contributions		
TOTAL GOVERNMENTAL ACTIVITIES	58265	58265
TOTAL PRIMARY GOVERNMENT	58265	58265

TOWNSHIP OF BEAVERTON

Test Desc	Expected Output	Actual Output
Extract township name	TOWNSHIP OF BEAVERTON	TOWNSHIP OF BEAVERTON
Extract date	3/31/2020	3/31/2020
Extract Net Position:		
Govt. Activities		
CASH AND CASH EQUIVALENTS	563416	563416
INVESTMENTS	144597	144597
TOTAL ASSETS	845439	845439
TOTAL LIABILITIES	12036	12036
NET INVESTMENT IN CAPITAL ASSETS	70412	70412
UNRESTRICTED	512356	512356
TOTAL NET POSITION	733278	733278
Extract Statement of Activities:		
Govt. Activities		
TOTAL GOVERNMENTAL ACTIVITIES	-355329	
PROPERTY TAXES LEVIED FOR GENERAL PURPOSES	193850	193850
STATE REVENUE SHARING	175053	175053
TOTAL GENERAL REVENUES	378157	378157

CHANGE IN NET POSITION	22828	22828
Extract Balance Sheet:		
General Fund		
CASH AND CASH EQUIVALENTS	338295	338295
INVESTMENTS	144597	144597
TOTAL ASSETS	515524	515524
TOTAL LIABILITIES	3168	3168
Fund Balances: UNASSIGNED	512356	512356
TOTAL FUND BALANCES	512356	512356
TOTAL LIABILITIES, DEFERRED INFLOWS AND FUND BALANCES	515524	515524
Total Governmental Funds		
CASH AND CASH EQUIVALENTS	563416	563416
INVESTMENTS	144597	144597
TOTAL ASSETS	775027	775027
TOTAL LIABILITIES	12036	12036
Fund Balances: UNASSIGNED	512356	512356
TOTAL FUND BALANCES	662866	662866
TOTAL LIABILITIES, DEFERRED INFLOWS AND FUND BALANCES	775027	775027
Extract Statement of Revenues, Expenditures, and Changes in Fund Balances		
General Fund		
PROPERTY TAXES	41322	41322
TOTAL REVENUE	292937	292937
CAPITAL OUTLAY	3500	3500
TOTAL EXPENDITURES	193853	193853
NET CHANGE IN FUND BALANCES (-62266	-62266
NET CHANGE IN FUND BALANCES FOR GOVERNMENTAL FUNDS	25732	25732
ADD: CAPITAL OUTLAY	3500	3500
Total Governmental Funds		
PROPERTY TAXES	193850	193850
TOTAL REVENUE	548149	548149
CAPITAL OUTLAY	3500	3500
TOTAL EXPENDITURES	522417	522417
NET CHANGE IN FUND BALANCES (25732	25732

BESSEMER TOWNSHIP

Test Desc	Expected Output	Actual Output
Extract township name	BESSEMER TOWNSHIP	BESSEMER TOWNSHIP
Extract date	3/31/2020	3/31/2020
Extract Net Position:		
Govt. Activities		
CASH AND CASH EQUIVALENTS	578845	578845
RESTRICTED CASH	-	-
TOTAL ASSETS	1890965	1890965
BONDS, DUE WITHIN ONE YEAR	12000	12000
BONDS, DUE IN MORE THAN ONE YEAR	156000	156000
TOTAL LIABILITIES	196378	196378
NET INVESTMENT IN CAPITAL ASSETS	1232774	1232774
UNRESTRICTED	458358	458358
TOTAL NET POSITION	1691132	1691132
TOTAL NET POSITION - THE ACCOMPANYING NOTES TO THE FINANCIAL STATEMENTS ARE AN INTEGRAL PART OF THIS STATEMENT.	N/A	-
Business-Type Activities		
CASH AND CASH EQUIVALENTS	430712	430712
RESTRICTED CASH	132507	132507
TOTAL ASSETS	3323343	3323343
BONDS, DUE WITHIN ONE YEAR	31500	31500
BONDS, DUE IN MORE THAN ONE YEAR	1335800	1335800
TOTAL LIABILITIES	1384303	1384303
NET INVESTMENT IN CAPITAL ASSETS	1061541	1061541
UNRESTRICTED	744992	744992
TOTAL NET POSITION	1939040	1939040
TOTAL NET POSITION - THE ACCOMPANYING NOTES TO THE FINANCIAL STATEMENTS ARE AN INTEGRAL PART OF THIS STATEMENT.	N/A	12
Total		
CASH AND CASH EQUIVALENTS	1009557	1009557

RESTRICTED CASH	132507	132507
TOTAL ASSETS	5214308	5214308
BONDS, DUE WITHIN ONE YEAR	43500	43500
BONDS, DUE IN MORE THAN ONE YEAR	1491800	1491800
TOTAL LIABILITIES	1580681	1580681
NET INVESTMENT IN CAPITAL ASSETS	2294315	2294315
UNRESTRICTED	1203350	1203350
TOTAL NET POSITION	3630172	3630172
TOTAL NET POSITION - THE ACCOMPANYING NOTES TO THE FINANCIAL STATEMENTS ARE AN INTEGRAL PART OF THIS STATEMENT.	N/A	
Extract Statement of Activities:		
Expenses		
TOTAL GOVERNMENTAL ACTIVITIES	628327	628327
TOTAL BUSINESS-TYPE ACTIVITIES	515471	515471
TOTAL PRIMARY GOVERNMENT	1143798	1143798
Charges for Services		
TOTAL GOVERNMENTAL ACTIVITIES	50106	50106
TOTAL BUSINESS-TYPE ACTIVITIES	441822	441822
TOTAL PRIMARY GOVERNMENT	491928	491928
Operating Grants & Contributions		
TOTAL GOVERNMENTAL ACTIVITIES	29863	29863
TOTAL BUSINESS-TYPE ACTIVITIES	-	-
TOTAL PRIMARY GOVERNMENT	29863	29863
Captial Grants & Contributions		
TOTAL GOVERNMENTAL ACTIVITIES	65332	65332
TOTAL BUSINESS-TYPE ACTIVITIES	-	-
TOTAL PRIMARY GOVERNMENT	65332	65332
Govt. Activities		
TOTAL GOVERNMENTAL ACTIVITIES	-483026	-
TOTAL BUSINESS-TYPE ACTIVITIES	-	-73649
TOTAL PRIMARY GOVERNMENT	-483026	-73649
TAXES, LEVIED FOR GENERAL OPERATIONS	360168	360168
STATE SHARED REVENUE	115495	115495

TOTAL GENERAL REVENUES, TRANSFERS, AND SPECIAL ITEMS	487492	487492
CHANGE IN NET POSITION	4466	4466
Business-Type Activities		
TOTAL GOVERNMENTAL ACTIVITIES	-	-483026
TOTAL BUSINESS-TYPE ACTIVITIES	-73649	-73649
TOTAL PRIMARY GOVERNMENT	-73649	-556675
TAXES, LEVIED FOR GENERAL OPERATIONS	-	-
STATE SHARED REVENUE	-	-
TOTAL GENERAL REVENUES, TRANSFERS, AND SPECIAL ITEMS	18635	18635
CHANGE IN NET POSITION	-55014	-55014
Total		
TOTAL GOVERNMENTAL ACTIVITIES	-483026	
TOTAL BUSINESS-TYPE ACTIVITIES	-73649	
TOTAL PRIMARY GOVERNMENT	-556675	
TAXES, LEVIED FOR GENERAL OPERATIONS	360168	360168
STATE SHARED REVENUE	115495	115495
TOTAL GENERAL REVENUES, TRANSFERS, AND SPECIAL ITEMS	506127	506127
CHANGE IN NET POSITION	-50548	-50548

ACME TOWNSHIP

Test Desc	Expected Output	Actual Output
Extract township name	ACME TOWNSHIP, GRAND TRAVERSE COUNTY	ACME TOWNSHIP, GRAND TRAVERSE COUNTY
Extract date	6/30/2020	6/30/2020
Extract Net Position:		
Govt. Activities		
CASH	3062861	3062861
TOTAL CURRENT ASSETS	3319146	3319146
TOTAL NON-CURRENT ASSETS	535220	535220
TOTAL ASSETS	20528645	20528645
TOTAL CURRENT LIABILITIES	162446	162446
TOTAL LIABILITIES	862446	862446
NET INVESTMENT IN CAPITAL ASSETS	15919279	15919279
UNRESTRICTED	2275707	2275707

TOTAL NET POSITION	19666199	19666199
Business-Type Activities		
CASH	2673591	2673591
TOTAL CURRENT ASSETS	2798977	2798977
TOTAL NON-CURRENT ASSETS	0	-
TOTAL ASSETS	9393461	9393461
TOTAL CURRENT LIABILITIES	138441	138441
TOTAL LIABILITIES	260461	260461
NET INVESTMENT IN CAPITAL ASSETS	6366225	6366225
UNRESTRICTED	2766775	2766775
TOTAL NET POSITION	9133000	9133000
Total		
CASH	5736452	5736452
TOTAL CURRENT ASSETS	6118123	6118123
TOTAL NON-CURRENT ASSETS	535220	535220
TOTAL ASSETS	29922106	29922106
TOTAL CURRENT LIABILITIES	300887	300887
TOTAL LIABILITIES	1122907	1122907
NET INVESTMENT IN CAPITAL ASSETS	22285504	22285504
UNRESTRICTED	5042482	5042482
TOTAL NET POSITION	28799199	28799199
Extract Statement of Activities:		
Expenses		
TOTAL GOVERNMENTAL ACTIVITIES	1768944	1768944
TOTAL BUSINESS-TYPE ACTIVITIES	723742	723742
Charges for Services		
TOTAL GOVERNMENTAL ACTIVITIES	149958	149958
TOTAL BUSINESS-TYPE ACTIVITIES	897056	897056
Operating Grants & Contributions		
TOTAL GOVERNMENTAL ACTIVITIES	25833	25833
TOTAL BUSINESS-TYPE ACTIVITIES	0	-
Captial Grants & Contributions		
TOTAL GOVERNMENTAL ACTIVITIES	343509	343509
TOTAL BUSINESS-TYPE ACTIVITIES	0	-
Govt. Activities		
TOTAL GOVERNMENTAL ACTIVITIES	-1249644	-
TOTAL BUSINESS-TYPE ACTIVITIES	0	173314

TAXES	1594095	1594095
STATE GRANTS	377947	377947
TOTAL GENERAL REVENUES	2030590	2030590
CHANGE IN NET POSITION	780946	780946
Business-Type Activities		
TOTAL GOVERNMENTAL ACTIVITIES	0	-1249644
TOTAL BUSINESS-TYPE ACTIVITIES	173314	173314
TAXES	0	-
STATE GRANTS	0	-
TOTAL GENERAL REVENUES	11845	11845
CHANGE IN NET POSITION	185159	185159
Total		
TOTAL GOVERNMENTAL ACTIVITIES	-1249644	
TOTAL BUSINESS-TYPE ACTIVITIES	173314	
TAXES	1594095	1594095
STATE GRANTS	377947	377947
TOTAL GENERAL REVENUES	2042435	2042435
CHANGE IN NET POSITION	966105	966105

ARCADA TOWNSHIP

Test Desc	Expected Output	Actual Output
Extract township name	Arcada Township	ARCADA TOWNSHIP
Extract date	3/31/2020	3/31/2020
Extract Net Position:		
Govt. Activities	<Data>	
Business-Type Activities	<Data>	
Total	<Data>	
Extract Statement of Activities:		
Expenses		
TOTAL GOVERNMENTAL ACTIVITIES	315487	315487
TOTAL PRIMARY GOVERNMENT	364336	364336
Charges for Services		
TOTAL GOVERNMENTAL ACTIVITIES	15969	15969
TOTAL PRIMARY GOVERNMENT	87259	87259
Operating Grants & Contributions		
TOTAL GOVERNMENTAL ACTIVITIES	888	888
TOTAL PRIMARY GOVERNMENT	888	888

Captial Grants & Contributions		
TOTAL GOVERNMENTAL ACTIVITIES	-	-
TOTAL PRIMARY GOVERNMENT	-	-
Govt. Activities		
TOTAL GOVERNMENTAL ACTIVITIES	-298630	-
TOTAL PRIMARY GOVERNMENT	-298630	22441
PROPERTY TAXES	158704	158704
STATE SHARED REVENUE	154732	154732
TOTAL GENERAL REVENUES	324729	324729
CHANGE IN NET POSITION	26099	26099
Business-Type Activities		
TOTAL GOVERNMENTAL ACTIVITIES	-	-298630
TOTAL PRIMARY GOVERNMENT	22441	-276189
PROPERTY TAXES	-	-
STATE SHARED REVENUE	-	-
TOTAL GENERAL REVENUES	821	821
CHANGE IN NET POSITION	23262	23262
Total		
TOTAL GOVERNMENTAL ACTIVITIES	-298630	
TOTAL PRIMARY GOVERNMENT	-276189	
PROPERTY TAXES	158704	158704
STATE SHARED REVENUE	154732	154732
TOTAL GENERAL REVENUES	325550	325550
CHANGE IN NET POSITION	49361	49361

TOWNSHIP OF ADAMS

Test Desc	Expected Output	Actual Output
Extract township name	TOWNSHIP OF ADAMS	TOWNSHIP OF ADAMS
Extract date	3/31/2020	3/31/2020
Extract Net Position:		
Govt. Activities		
CASH & INVESTMENTS	804497	804497
TOTAL ASSETS	1149319	1149319
TOTAL LIABILITIES	586	586
NET INVESTMENT IN CAPITAL ASSETS	293091	293091
UNRESTRICTED	846294	846294
TOTAL NET POSITION	1148733	1148733

Extract Statement of Activities:		
Net (Expense) Revenue	Net (Expense) Revenue	Govt. Activities
PROPERTY TAX, LEVIED FOR GENERAL PURPOSES	62214	62214
STATE SHARED REVENUE	179738	179738
TOTAL GENERAL REVENUES	265688	265688
CHANGE IN NET POSITION	100647	100647

5.5.1.1 Conducting a Review

We will mainly have two reviews for the project. First, we will conduct an internal review between our team members to make sure we have met our requirements and client expectations. The second review will be conducted with the client, where we will make sure we are all on the same page, and everything agreed on is present in the product.

5.5.1.2 Roles and Responsibilities

As mentioned in section 1.2 (SQA organizational role), the roles and responsibilities of team members may overlap. Still, the following roles are set by the team:

- Software Quality Assurance Lead: Ahmed Mawari
- Lead Developers: Hasan Alameh and Tanis Daniels-Wanamaker
- Testing Lead: Adham Abdalla

5.5.1.3 Review work products

A list of bugs/issues with the software will be created and managed through GitHub's "issue" functionality. The bugs will be ranked according to their impact on the user experience and how much they affect the data parsing. The issues will then be resolved in decreasing order of importance by different team members (more so by the Lead Developers) in a reasonable time window.

5.5.2 Formal Technical Reviews

5.5.2.n.1 Description and focus of the review

The System Specification is expanded more upon in the SRS document. We will review the general structure and delivery of the code as a whole, ensuring it meets the requirements we have set.

5.5.2.n.2 Timing of the review

The System Specification Review will be executed after the completion of the System Specification, estimated to be less than a month after the start of development. Its completion in a timely manner will prevent future development issues.

5.5.2.n.3 Work products produced

The SQA leader will produce an evaluation document pertaining to the System Specification review. It will note any adjustments to the software's internal system design, as well as including potential issues and their solutions.

5.5.3 Change control reviews and audits

The use of Github will make it easy for the team members to control code push/pull changes and revert to previously working commits if any errors occur.

5.5.3 SQA Audits

The most important part is making sure the clients are always informed about our recent progress towards the final products. Following that, internal team audits, comments, and suggestions are always made to make sure our quality testing is acceptable. The use of Github and instant chat between team members makes it easy to communicate and make changes in a timely manner.

5.6 Problem Reporting and Corrective Action/Follow-up

5.6.1 Reporting mechanisms

Problems will be reported in a document stored on the team's google drive and will be communicated to each team member in our Discord.

5.6.2 Responsibilities

Software quality assurance lead is responsible for all corrective actions and follow ups with team members, the rest of the team will assist but it is mainly the responsibility of the Software quality assurance lead.

5.6.3 Data collection and evaluation

A running list of errors/defects will be used to evaluate the current state of the software. Communication with the client and getting feedback will help the team with design and functionality uncertainties.

5.6.4 Statistical SQA

The team will rank defects from 1 (minor) to 4(critical) in a running error/defect list and this will help the focus efforts in fixing bugs in order of importance to the quality of the software.

5.7 Software Process Improvement Activities

5.7.1 Goal and objectives of SPI

The goal and objectives of SPI are to make sure all defects are dealt with accordingly and record how much time it takes in order to fix any defect that shows up. First the team must identify how critical the defect is and if it affects the software heavily. The goal is to use previously recorded defects and how long it took to fix them in order to plan for fewer mistakes in the upcoming future.

5.7.2 SPI tasks and responsibilities

The whole group is responsible for any tasks or issues to come up since we are a small team and will be working together in order to take responsibility and make the product to the best of our ability.

5.8 Software Configuration Management Overview

The team is always in reach of each other in case of any urgent communication needed but the team constantly meets in order to resolve any issues that come up. The developers will always update the document with any new defects or bugs that come up in order to review them and allow time to fix them. The client will receive every new milestone achieved in order to stay on board with the team and

see how far along the development of the project has come. The team will use google drive, github, and discord to keep track of the software and documents.

5.9 SQA Tools, Techniques, Methods

The tools used are programming platforms such as PyCharm or Visual Studio in order to make the source code for the project. Github and Google Drive will be used in order to keep everything in the drive shared with the whole team for editing and viewing. Techniques and methods used will be designing documents, researching about the project, splitting tasks between team members and communicating with each other and the clients. Another important technique to be implemented is time management in order to keep track of how far the project is coming along.

6. Future Maintenance Suggestions

As long as new PDF formats are being used by local governments every year, the program may need small updates to keep up with them.

7. Delivery/Acceptance Statement

The acceptance statement was sent to Professor Bruce Maxim (bmaxim@umich.edu). We do not currently have a copy of the letter.

8. References & Bibliography

<https://github.com/jsvine/pdfplumber>

9. Appendix A - User Manual

Introduction

This program takes in a financial statements PDF and outputs rows of interest that were agreed on with the client.

Purpose of product

You can use this product to gather all data from different local governments into a single CSV file.

Operating environment

This program was only tested on Windows 10, though it can run on Unix-based operating systems too.

General functionality

You will need to provide a PDF, click submit, and wait for the program to finish parsing.

Limitations

The program will not read scanned PDFs, pages that are rotated, or ones with special non-english fonts

Installation

No installation required. Run the UI .exe file to start the program

Physical requirements

A working PC

Tutorial

1. Make sure you have a PDF meeting the requirements.
2. Run the UI exe of the program
3. Browse to your PDF location and select it
4. Click on Submit and wait till it is done
5. Click 'Open Master CSV' to see the results

More info on running the program with visual help:

https://youtu.be/6aHW_yJO_eg

Appendix B - Sample Output

The sample output is two pages due to conversion from CSV file to pictures.

1	DATE	COUNTY	PAGE	DATA TITLE
2	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	CASH AND CASH EQUIVALENTS
3	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	TOTAL CURRENT ASSETS
4	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	CAPITAL ASSETS NOT BEING DEPRECIATED
5	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	CAPITAL ASSETS BEING DEPRECIATED, NET
6	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	TOTAL NONCURRENT ASSETS
7	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	TOTAL ASSETS
8	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	LEASES PAYABLE - DUE WITHIN ONE YEAR
9	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	TOTAL CURRENT LIABILITIES
10	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	LEASES PAYABLE - DUE IN MORE THAN ONE YEAR
11	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	TOTAL LIABILITIES
12	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	NET INVESTMENT IN CAPITAL ASSETS
13	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	UNRESTRICTED
14	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION	TOTAL NET POSITION
15	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF ACTIVITIES	TOTAL GOVERNMENTAL ACTIVITIES
16	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF ACTIVITIES	TOTAL GENERAL REVENUES
17	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF ACTIVITIES	CHANGE IN NET POSITION
18	31-Mar-20	BARODA TOWNSHIP	BALANCE SHEET - GOVERNMENTAL FUNDS	CASH AND CASH EQUIVALENTS
19	31-Mar-20	BARODA TOWNSHIP	BALANCE SHEET - GOVERNMENTAL FUNDS	TOTAL ASSETS
20	31-Mar-20	BARODA TOWNSHIP	BALANCE SHEET - GOVERNMENTAL FUNDS	TOTAL LIABILITIES
21	31-Mar-20	BARODA TOWNSHIP	BALANCE SHEET - GOVERNMENTAL FUNDS	NONSPENDABLE
22	31-Mar-20	BARODA TOWNSHIP	BALANCE SHEET - GOVERNMENTAL FUNDS	UNASSIGNED
23	31-Mar-20	BARODA TOWNSHIP	BALANCE SHEET - GOVERNMENTAL FUNDS	TOTAL FUND BALANCES
24	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF REVENUES, EXPENDITURES, AND CHANGES IN FUND BALANCES - GOV FUNDS	TOTAL REVENUES
25	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF REVENUES, EXPENDITURES, AND CHANGES IN FUND BALANCES - GOV FUNDS	DEBT SERVICE: - PRINCIPAL
26	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF REVENUES, EXPENDITURES, AND CHANGES IN FUND BALANCES - GOV FUNDS	DEBT SERVICE: - INTEREST
27	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF REVENUES, EXPENDITURES, AND CHANGES IN FUND BALANCES - GOV FUNDS	TOTAL EXPENDITURES
28	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF REVENUES, EXPENDITURES, AND CHANGES IN FUND BALANCES - GOV FUNDS	TOTAL OTHER FINANCING SOURCES
29	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF REVENUES, EXPENDITURES, AND CHANGES IN FUND BALANCES - GOV FUNDS	NET CHANGE IN FUND BALANCES
30	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF REVENUES, EXPENDITURES, AND CHANGES IN FUND BALANCES - GOV FUNDS	LIABILITIES OF THE CURRENT PERIOD. FOR THIS PURPOSE, THE TOWNSHIP CONSIDERS REVENUES TO BE AVAILABLE IF THEY ARE
31	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION - PROPRIETARY FUNDS	CASH AND CASH EQUIVALENTS
32	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION - PROPRIETARY FUNDS	TOTAL CURRENT ASSETS
33	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION - PROPRIETARY FUNDS	CAPITAL ASSETS, NET OF ACCUMULATED DEPRECIATION
34	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION - PROPRIETARY FUNDS	TOTAL ASSETS
35	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION - PROPRIETARY FUNDS	NET INVESTMENT IN CAPITAL ASSETS
36	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION - PROPRIETARY FUNDS	UNRESTRICTED
37	31-Mar-20	BARODA TOWNSHIP	STATEMENT OF NET POSITION - PROPRIETARY FUNDS	TOTAL NET POSITION
38	31-Mar-20	BARODA TOWNSHIP	Statement of Revenues, Expenses, and Changes in Fund Net Position - Proprietary Funds	TOTAL OPERATING EXPENSES
39	31-Mar-20	BARODA TOWNSHIP	Statement of Revenues, Expenses, and Changes in Fund Net Position - Proprietary Funds	TOTAL NONOPERATING REVENUE

1	EXPENSES	CHARGES	OPERATIN CAPITAL G Net (EXPE	GOVERNMENT BUSINESS	TOTAL	COMPONENT UNITS	GENERAL FUND	TOTAL GOVERNMENTAL FUNDS	TOTAL ENTERPRISE FUNDS
2				997462	128979	1126441			
3				1060707	146942	1207649			
4				29861	-	29861			
5				1497351	28892	1526243			
6				1527212	28892	1556104			
7				2587919	175834	2763753			
8				88145	-	88145			
9				113573	-	113573			
10				86554	-	86554			
11				200127	-	200127			
12				1352513	28892	1381405			
13				298809	146942	445751			
14				2387792	175834	2563626			
15	1013685	280510	5000						
16				195800	-	195800			
17				7182	144	7326			
18							288740	997462	
19							376743	1174321	
20							78525	139042	
21							-	591	
22							286112	286112	
23							286112	1023173	
24							352787	1120621	
25							-	82627	
26							-	9056	
27							265860	1550831	
28							-	261326	
29							86927	-168884	
30							60		
31									128979
32									146942
33									28892
34									175834
35									28892
36									146942
37									175834
38									16901
39									144

Appendix C - Team Member Resumes

Ahmed Mawari

5060 Williamson St. | Dearborn, MI 48126
afinawari@gmail.com | (313) - 428 - 1309

OBJECTIVE To obtain a role as a software engineer in a collaborative work environment where I am able to utilize my skills in programming languages, scripting languages, databases, and emulation software to produce innovative products.

EDUCATION **University of Michigan – Dearborn** Dearborn, MI
Bachelor of Science in Computer Information Science
Major: Software Engineering Expected Graduation: Aug 2021

COMPUTER SKILLS **Languages:** C, C++, C#, JAVA, Kotlin, Visual Basic, Python, MySQL, Microsoft sql | **Software:** Eclipse IDE, BayesiaLab, Android studio, IntelliJ IDE, Microsoft Visual Studio IDE, Microsoft Office, Microsoft Visio, Minitab Statistical Software, Enterprise architect, Git/SCM, docker.

COURSE WORK C/C++ Programming, Calculus I – II, Mobile Application Design and Implementation, Computer Networking, Discrete Structures, Computer Organization, Data Structures and Algorithm Analysis, Operating Systems, Probability and Statistics, Matrix/Linear Algebra, Software Engineering, Theory of Computation, Physics, Analytical Chemistry, Technical Writing for Engineers, Advanced Essay Writing.

PROJECTS UNIVERSITY OF MICHIGAN-DEARBORN

Mobile Application Design and Implementation - Course Winter Semester 2021

- Created a weather app for android
- Utilized REST API
- Designed and created UI using XML.
- Utilized Jetpack standards

Software Engineering II - Course Fall Semester 2020

- Created UML models and design documentation for a dealership inventory system
- Collaborated with a team of four using GitHub and slack.
- Coded and tested use cases to ensure completeness and quality.
- Semester long project with numerous presentations

Operating Systems- Course Winter Semester 2020

- Designed and developed a traffic control simulation
- Used docker to create a platform independent runtime environment
- Used threading and semaphores/mutex locks to demonstrate concurrency.

Probability and Statistics for Engineering - Course Winter Semester 2019

- Compiled statistics and deduced relations and factors for car accidents for Michigan across a five-year period.
- Created data visualization using BayesiaLab and Minitab Statistical software.
- Worked in a team of four to present the findings in an oral presentation.

HASAN ALAMEH

7330 Middlepointe St.

(313) 815 - 0575

Dearborn, MI 48126

halameh8@gmail.com

Objective Statement

Seeking a position as a Software Engineer Intern utilizing acquired engineering skills, abilities, and experiences gained through previous job experiences, relevant education, and independently learned skills to contribute to the ongoing success of the company.

Education

- A.A.S in Computer Information Systems (August 2017 to December 2018)
Henry Ford College (GPA: 3.75)
- Bachelor's Degree in Software Engineering (Jan 2019 to December 2021)
University of Michigan-Dearborn (GPA: 3.14)

Relevant Courses:

- Software Engineering
- Data Structures & Algorithm Analysis
- Assembly Language & Microprocessors

Relevant Team Projects:

- CPU simulation in C++
 - Role:**
 - Creation of a time unit on which tasks depend on
 - Enqueuing random tasks with different priority levels
 - Outputting a log, events, and statistics files for tasks
- Virtual Online Voting System
 - Role:**
 - Designing and coding GUI using C#
 - Setting up Azure SQL database and relevant tables
 - Connecting C# program to database and executing SQL queries

Link: <https://github.com/cyhamUMICH/CIS-376-VOTE>

Work Experience

Student Co-op (Part-Time) (June 2020 to Present)

DTE Gas - Allen Road Center, Melvindale, MI

- Created and maintained PowerBI reports with data pulled from SharePoint tables.
- Utilized Office 365 Suite to create PowerApps used by employees.

Technology Summary

• C++	• Unix/Linux	• Git	• Azure
• Python	• Java	• SQL	• Agile Development
• Command-Line	• C#	• Visual Studio	• Relational Databases

Tanis Daniels-Wanamaker
Westland, MI 48186
734-258-5391, tdanielswanamaker@gmail.com

EDUCATION

University of Michigan-Dearborn Dearborn, MI
Bachelor of Science in Computer Science

SKILLS

- Proficient with C++ programming using Microsoft Visual Studio
- Adept at Microsoft Excel, Word, and PowerPoint as well as HTML/CSS web design
- Working knowledge of the MiniTab and OpenMarkov statistical and analytical softwares
- Versed in UML design and creation using Sparx Systems Enterprise Architect
- Familiar with designing technical reports for both technical and non-technical audiences

EXPERIENCE

U.S. Army DEVCOM Ground Vehicle Systems Center Warren, MI
Student Trainee (Engineering Technician) May 2021 - August 2021

- Preparation of systems requirements for robotics in IBM Rational DOORS
- Production of various project templates for use with the IBM Rational Publishing Engine
- Exporting of performance specification documents in both department and program office standardization formats

Fandom, Inc. Remote
Volunteer Web Content Editor August 2015 - Present

- Creation and editing of articles for informative, free public knowledge on various game software, including technical game mechanics and decision trees
- Design, documentation, and maintenance of template systems in the MediaWiki engine
- Discussion of sub-site improvements for better user interaction and article SEO

PROJECTS

- Statistical Analysis of Steam Games (2020), a group project involving substantive data analysis of game software based on attributes such as genre, engine, and accessibility. During the process, we sourced, created, and merged various data sets in programs such as Microsoft Excel.

Adham Abdalla

810-922-4648 • Farmington Hills, MI • adhama2025@gmail.com
Software Engineer/Management Student

EXPERIENCE

Senior Software Engineer/Management Intern
Robert Bosch | Plymouth, MI

Feb 2021 – Present

Worked with a cross-functional team under the head scrum master to contribute and learn about processes and tools used to ensure client requirements are met and satisfied.

- Updated and assigned tasks in different project dashboards to stakeholders throughout sprints
- Created and maintained dashboard reports, tables, and statuses for projects via tools ex. Excel/PowerBi, JazBuilder, Scripts
- Participated in fulfilling customer requests using process and tools to meet their demands
- Ensured clear communication between stakeholders by contributing to daily and weekly meetings

Junior Software Engineer/Management Intern
Robert Bosch | Plymouth, MI

Nov. 2019 – Aug 2020

Worked with a cross-functional team under the global engineering manager to contribute and learn about processes and tools used to ensure client requirements are met and satisfied.

- Contributed to creating a website about how to use and apply APSICE and the V-Model in producing a product
- Created and managed a social media channel for the engineering team district
- Tested functionalities of cars and test drove multiple cars
- Updated and assigned tasks in different project dashboards to stakeholders throughout sprints
- Ensured clear communication between stakeholders by contributing to daily and weekly meetings

Vendor Manager
Timeless Creations | Dearborn, MI

Oct 2018 - Sep 2019

- Managed interactions with customers to set up events, weddings, funerals, etc.
- Managed crew of 10+ people to ensure the event is setup properly
- Organized money transactions with customers through excel
- Managed social media of company and grew its popularity and profit by double
- Interviewed customers after events to see customer satisfaction trend

Restaurant Manager
Wayback Burgers | Dearborn Heights, MI

Jan 2017 – Aug 2018

Responsible for day-to-day management of the restaurant

- Managed weekly schedule of all employees and data entry of inventory to place new shipment every week
- Managed day to day interactions between employees and customers to ensure the place ran smoothly
- Conducted customer interviews understand the customer satisfaction trend and how to improve it
- Achieved employee of the month 7 times

EDUCATION

University of Michigan | Bachelor of Science in Software Engineering
Scrum Alliance | Certified Scrum Master (CSM)
Scrum Alliance | Certified Scrum Product Owner (CSPO)
Udemy | Become a Product Manager

Expected Aug. 2021
July 2021
Aug 2021
In Progress

SKILLS

Scrum, Agile, Communication, Cross-functional Team-experience, Management, People Skills, Soft Skills, Processes, Tools, Stakeholder Interviews, Customer Satisfaction, C++, C, Java, JavaScript(some), Python(some), Data analysis, Data Entry, Operating Systems, Windows, Microsoft Office, Github, Visual Studio, Jenkins, Confluence, RTC, JazBuilder, Android Studio(some), Kotlin(some) and more

Bilingual – English and Arabic

Appendix D - Project Plan & Logbook

Project planning documents with corrected estimates and schedule.

Logbook summarizing the design changes made following technical reviews might also include your commit logs.

Task	Duration (estimate) (days)	Predecessor Tasks	Successor Tasks	% Complete
Complete SRS	14	None	Presentation	100
Presentation	1	SPMP, ERD	Requirement Details	100
Requirement Details	7	Presentation	Designing User Stories/ Use Case for Sprints	100
Designing User Stories/ Use Case for Sprints	14	Requirements Details	Implementing Design	100
Making RMMM	5	SRS	Implementing Design	100
Software Project Plan	5	SRS	Implementing Design	100
Presentation 2	1	Software Project Plan, RMMM	Implementing Design	100
Accepting a PDF file (Sprint 1)	3	Designing User Stories/ Use Case for Sprints	Testing/Debugging	100

Parser - Whole Back-end (Sprint 1)	35	Designing User Stories/ Use Case for Sprints	Integration of UI & Back-end	100
User Interface Feature - Whole Front End (Sprint 1)	20	Designing User Stories/ Use Case for Sprints	Integration of UI & Back-end	100
Integration of UI & Back-end	4	Complete Back-End & UI	Testing/Debugging	100
Testing/Debugging	7	Integration of UI & Back-end	Deployment	100
Deployment	1	Testing/Debugging	Client Demo	100
Client Demo	1	Testing/Debugging	Final Report	100
Final Report	4	Client Demo	Presentation	100
Presentation 3	1	Testing and Debugging of Features		100

Appendix E - Project Demo Notes

For the demo, we will first give a brief description of the program & client requirements. Then, we will explain the PDF format differences, show an example of a PDF, then input a PDF into the program. We will then explain briefly the different functions the PDF goes through, then open to show the results on the CSV output.

Appendix F – Final Presentation Slides





Michigan Treasury Financial Data Parser

Final Project Demo


CIS 4962 - Professor Maxim & Professor Akingbehin

Ahmed Mawari
Tanis Daniels-Wanamaker
Adham Abdalla
Hasan Alameh



Project Overview

- Every year, local governments in Michigan (cities, townships, villages, etc.) upload an audited financial statement in a PDF
- These financial statements include different tables with rows that our client is interested in extracting into a CSV file
- Our program takes the PDF as an input, and outputs the rows of interest in a comma-separated and properly organized CSV file



```
graph LR; PDF[PDF] -- Input --> EXE[EXE]; EXE -- Output --> CSV[CSV]
```

Project Overview

- Audit formats are non-standardized
 - Location of certain sections may differ between audits
 - Tables may have different names for different fields
- Data parser reads the different formats and outputs a CSV file with select, more-standardized values (as specified by the client)
 - Outputted Excel sheet can then be used by the client for analytical purposes
 - Sheet data can also be compared to the standardized, manually-transcribed F-65 audit form to detect clerical errors



Program Demo

