Final Project Report on

Reporting Visualization & Automation

by

HARDIK SHARMA

N045

70471017047

Under the Guidance of

TITHI MUKHERJEE

&

PROF. ALANKNANDA LONARE

In partial fulfillment for the award of the degree of

5 Years Integrated MBA (Tech) Program

IN COMPUTER ENGINEERING

At

SVKM'S
NMIMS
Deemed-to-be UNIVERSITY

MUKESH PATEL SCHOOL OF TECHNOLOGY MANAGEMENT & ENGINEERING (MPSTME), MUMBAI

JULY, 2021

CERTIFICATE

MANAGEMENT INTERNSHIP REPORT Semester IX MBA (Tech)

Submitted in Partial Fulfillment of the Requirements for Management Project/Training for Semester IX MBA (Tech)

Name of the Student: - Hardik Sharma

Roll No & Batch: - N045 & 2022

Academic Year: - 2017 - 2022

Name of the Discipline: - Computers / Finance

Name and Address of the Company: - Morgan Stanley / Nirlon Knowledge Park, St Yadav Road Block B2 (Level 3, 4 & 8), Block B4/5 (Level 7 - 9) and Block B7 (Level 1)

Training Period: From 3rd May, 2021 to 9th July, 2021

THIS IS TO CERTIFY THAT

Mr. Hardik Sharma has Satisfactorily Completed his Training/Project Work, submitted the training report and appeared for the Presentation & VIVA as required.

External Examiner **Internal Examiner** **Faculty Mentor**

Date:

Place:

Seal of University:

CERTIFICATE

DECLARATION

I, Hardik Sharma, Roll No. N045 MBA Tech (Computer Engineering), IX semester understand that plagiarism is defined as anyone or combination of the following:

1. Un-credited verbatim copying of individual sentences, paragraphs or illustration (such as graphs, diagrams, etc.) from any source, published or unpublished, including the internet.
2. Un-credited improper paraphrasing of pages paragraphs (changing a few words phrases, or rearranging the original sentence order)
3. Credited verbatim copying of a major portion of a paper (or thesis chapter) without clear delineation of who did wrote what. (Source: IEEE, The institute, Dec. 2004)
4. I have made sure that all the ideas, expressions, graphs, diagrams, etc., that are not a result of my work, are properly credited. Long phrases or sentences that had to be used verbatim from published literature have been clearly identified using quotation marks.
5. I affirm that no portion of my work can be considered as plagiarism and I take full responsibility if such a complaint occurs. I understand fully well that the guide of the seminar/ project report may not be in a position to check for the possibility of such incidences of plagiarism in this body of work.

Signature of the Student:

Name: Hardik Sanjay Sharma

Roll No.: N045

Place: Mumbai

Date: 7th July, 2021

ACKNOWLEDGEMENTS

The management internship at **Morgan Stanley** has been a professionally satisfying experience. This opportunity helped me expand my knowledge base and learn about several new reporting tools as well as their implementation. It enabled me to enhance my technical skill set by learning to work with Power BI, VBA Macros and other related tools. I got the opportunity to create corporate dashboards, build reports and present my ideas to upper-level management regarding newer customized reporting tools.

I would like to begin by expressing my sincerest gratitude to my reporting manager and Industry Mentor, Mrs. Tithi Mukherjee**,** *Vice President – Risk COO,* for being an excellent mentor to me during this internship. As a part of her team, I picked up on several managerial skills like delegation, effective communication, and leadership. I would also like to thank my buddy, Mr. Preetam Mishra, for helping me through the course of my internship, consistently supporting me in my projects and taking time out of his schedule to resolve my queries.

The internship has been a steppingstone for me towards working with highly skilled professionals and executives. I am indebted to the HR team at Morgan Stanley for successfully managing this internship program. Even within the Work from Home dynamic, the HR team ensured that I felt like a part of the organization. The internal training modules and study resources provided have truly enhanced my knowledge on several key topics of the industry.

I want to thank all my other team members from FRM who made me feel welcome and provided me with several opportunities to understand the workings within the department, giving me a real-life exposure to organization wide risks and their management.

I would like to acknowledge the backing and guidance of all the professors and staff members of NMIMS' MPSTME for including this MIP in the curriculum so that students could get a professional experience of working at an organization and gain industry experience.

I would like to thank my *faculty mentor*- Prof**.** Alaknanda Lonare for her continued support at all stages and for providing me the necessary guidance. Once again, my hearty thanks to everyone at Morgan Stanley for having made the internship program a fruitful one. Last but not the least, I would like to express my gratitude towards my parents for acting as the constant source of inspiration and for motivating me.

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

In today's business world of constant growth and development, data crunching and analysis has taken a key position in the decision-making process. However, effective reporting is equally important for beneficial decision making. Data analysis can prove to be futile if the reports don't convey the information correctly.

This need for proper reporting has given rise to several visualisation tools like Power BI and Tableaux amongst many others. During this internship, I understood the workings and intricacies of Power BI and how we can use this tool for creating better reports, dashboards and different charts. To improve the reporting processes in my organisation, I helped develop a custom visual in Power BI and also tried to find ways to link the capabilities of Power BI with a spreadsheet tool like Excel.

Decisions Making is a dynamic and time driven process. Often, we notice that the right decisions at the wrong time can prove to be harmful for organisations. Thus, it's important for any risk reporting team to be quick and agile on their feet. To achieve such efficiency, the team must strive towards automating and saving time on repetitive processes while ensuring the overall quality of their work. I was assigned the task of automating one such process for the company by using Visual Basics Applications (Macros) in MS Excel.

As is the case with all systems and processes around the world, we must always try to keep ourselves updated with the best improvements and advances in the industry. It is important to explore newer options for our internal processes in order to make work easier for the personnel. As part of my internship, I had the opportunity to explore different applications that could serve as improvements or alternates to the current tools used within the organisation.

I feel that this internship has benefited me in a profound manner and this report provides a brief summary of the projects that I have worked on.

INTRODUCTION

1.1 About the Company

Morgan Stanley is an American multinational investment bank and financial services company headquartered at 1585 Broadway in New York City. With offices in more than 42 countries and more than 60,000 employees, the firm's clients include corporations, governments, institutions, and individuals. The main areas of business for the firm today are institutional securities, wealth management and investment management.

Morgan Stanley has been operating in India for over 20 years from its national headquarters in Mumbai, providing a range of services to domestic and international clients. It provides a premier institutional securities platform, with a full range of investment banking, capital markets, research and equity and fixed income sales & trading services, in addition to infrastructure, real estate and private equity investment businesses. The Firm set up its first GIC in Mumbai over ten years ago, providing IT, Operations, Finance, Compliance Fund Services, and other support to its other business units across geographies.

1.2 Project Objective

- Provide “Value-added” and “Valuable” reporting through innovative solutions, automations and optimizations
- Actively participate in Power BI exploratory projects.
- Develop business and technical skills, to ensure effective contribution and high-quality delivery.

1.3 Problems at Hand

As an intern in the firm risk management and reporting department, I had the opportunity to understand the various functions performed by the team. Due to certain data security objectives of the department, I could not be given access to the datasets from live ongoing projects. However, my primary goal was to assist the team on exploration of new ideas and help resolve any current problems being faced by the members using newer methods. Listed below are some of the difficulties that my projects would help to overcome:

- We are limited by the functionalities offered by the Default Visuals available in Power BI. Thus, there was a need to expand those functionalities and create custom made visuals tailored to our needs.

- To combine the analytical power of Excel with the data visualization capabilities of Power BI and allow users to visualize their datasets side by side enabling them to perform certain analysis without having to toggle between applications.
- Processes that are repetitive in nature consume a lot of time, when done manually. Such processes can also be prone to human errors. Automating such processes can help us to save a lot of time and ensure quality of work.
- Since the team has only recently shifted to Power BI, the tool is relatively new for many team members. This required the creation of a tutorial that could serve as a one stop source of information for new joinees and users unfamiliar with the reporting tool.
- As a reporting team, we keep trying to find avenues for self-improvement. With the fast-paced developments in the technology world, there are several third-party tools which could be helpful for better, efficient, and effective reporting. It was therefore necessary to first explore these tools and create a detailed report showcasing their features.

1.4 Project Overview

During the internship, I worked on numerous projects which could potentially solve the problems described above. A brief overview of my projects is given below:

❖ *Power BI*

- Learning and understanding the process to build reports and dashboards in Power BI by performing exercise projects on dummy datasets.
- Exploring the various methods to build a custom visual in Power BI. Developing a ‘Dual Axis Gauge’ custom visual using Node.js approach.
- Learning Power BI and DAX queries for quicker and effective reporting in Power BI.
- Preparing a Power BI tutorial for new beginners comprising details on ways to load the data, data transformation processes and a brief on the default charts available in the visualization pane.

❖ *MS Excel – VBA Macros*

- Achieving Excel-Power BI integration using web scraping. An automated bot was used to navigate the cloud-based Power BI workspace, find the required visual and fetch its image into an Excel worksheet to enable the user to work on data and see the graphical representation of data alongside in the same window.
- Developing a program to perform automated tabulation of data using VBA Macros to preserve time and reduce manual intervention.

- *Other Exploratory Projects*
  - Creating various documents containing the overview, features and benefits of the third-party tools which could potentially benefit the team.

1.5 Limitations

Due to data security and confidentiality concerns, I was not provided with access privileges to any live projects that the team was working on. Owing to the same privacy policies of the company, this report has been prepared after redacting certain key elements of my workings on the projects, which must remain confidential to the company. Since most of the projects were exploratory in nature, the reporting on such projects is primarily only research oriented. The practical implementation of these projects may be conducted by the organisation in the future as per their requirements.

PROJECT 1: CUSTOM VISUAL – DUAL AXIS GAUGE

2.1 Purpose

The Power BI Visualisation Pane offers several default visuals and chart types to be used for creating reports and dashboards. However, with the increasing requirements in the reporting framework, we may need to develop some new visuals that are tailored to serve our specific needs.

Thus, the primary purpose of this project was to unlock the pathway to developing custom visuals within Power BI. With the ability to develop our own visuals, we will have no limitations in the options available and can display data in a much better and effective manner.

2.2 Methods to develop visuals

There are various methods to create a custom visual. Some of them are:

- R and Py scripts in Power BI
- Development tools like Charticulator
- Using the D3.js Visual
- Node.js and Powerbi-Visual-Tools Package Approach

2.2.1 R & Py Scripts Method

R Visuals

R packages are collections of R functions, data, and compiled code that are combined in a well-defined format. When R is installed, it comes with a standard set of packages, and other packages are available for download and installation. Once installed, an R package must be loaded into the session to be used. The primary source of free R packages is CRAN, the Comprehensive R Archive Network. Power BI Desktop can use any type of R packages without limitation. We can install R packages for use in Power BI Desktop on our own.

R visuals in the Power BI service have a few limitations:

- Data size limitations – data used by the R visual for plotting is limited to 150,000 rows. If more than 150,000 rows are selected, only the top 150,000 rows are used and a message is displayed on the image. Additionally, the input data has a limit of 250 MB.
- Resolution - all R visuals are displayed at 72 DPI.

- Plotting device - only plotting to the default device is supported.
- Calculation time limitation – if an R visual calculation exceeds 60 seconds the script times out, resulting in an error.
- R visuals are refreshed upon data updates, filtering, and highlighting. However, the image itself is not interactive and does not support tool tips.
- R visuals respond to highlighting other visuals, but you cannot click on elements in the R visual to cross filter other elements.

Python Visuals

The support for Python in Power BI enables us to use Python scripts to prepare our dataset, apply sophisticated analytics or machine learning in the Power BI Desktop, and then plot the results in our Power BI reports using any of the hundreds of open-source Python visualization packages.

Python visuals in Power BI Desktop have a few limitations:

- Calculation time limitation - If a Python visual calculation exceeds five minutes the execution times out which results in an error.
- Relationships - As with other Power BI Desktop visuals, if data fields from different tables with no defined relationship between them are selected, an error occurs.
- Python visuals are refreshed upon data updates, filtering, and highlighting. However, the image itself isn't interactive and can't be the source of cross-filtering.
- Python visuals respond to highlighting other visuals, but you can't click on elements in the Python visual to cross filter other elements.
- Only plots that are plotted to the Python default display device are displayed correctly on the canvas. Avoid explicitly using a different Python display device.
- Python visuals do not support renaming input columns. Columns will be referred to by their original name during script execution.

Py and R Visual

The R and Python script visuals can be used together if we need to work with statistics and data analysis. R and Python script visuals can be developed using the Default R and Py Visual in Power BI Visualization Pane. It can also be used if we need to work with many rows.

Figure 1: R and PY Visuals in Power BI

We can use most of the libraries available in R and Python to perform various operations on the Dataset. It must be noted that we cannot create a chart with a Custom Design by using this method. We can only display the charts that are available in python and R library such as the mathplotlib and so on.

2.2.2 Charticulator Method

Charticulator (charticulator.com) is the no-code way to create custom and reusable chart designs. Users can create a custom chart right within Power BI using the Microsoft Charticulator Visual, either from scratch or using a template.

Figure 2: Charticulator Development Tool

Charticulator is a website where users can build custom visuals using the drag and drop components available in the tools section of the Website. We do not know how to code in the language required to make the Visual. It provides us with a GUI to drag and drop the components required to make the visual.

This method, however, had certain restrictions. It provides limited options in terms of chart design and its functionalities. Also, as a third part software, it cannot be directly implemented within the IT infrastructure of the organisation due to strict security policies.

2.2.3 D3.js Visual

Figure 3: D3.js Custom Visual

D3.js visual helps to bring data to life by using HTML, SVG, CSS via the D3.js framework. Either the visual can be created from scratch or an existing D3.js visual can be used. We can create a custom visual using this visual without having to perform any tedious installation processes. To use this visual, we must download it from the Microsoft AppSource.

The D3.js visual is a custom visual itself, within which we can write the JavaScript code using the D3.js library. This visual has three main components.

- The first component is the JavaScript Editor where we can type our D3.js code to render the charts. We can also write the logic using JavaScript to fetch the data and interact with the rendered chart.
- Second is the Css Editor where we can define the Css styling for various chart elements. This can be considered similar to Css file in HTML. Adding CSS is optional as all the necessary styling options can also be provided using JavaScript.
- The third and final component contains various important objects like the dimension object, data object and the colour object. These objects are used to get important attributes like the dimension of the visual, the colours that the user has selected and the data that has been given to the visual.

We can create a variety of attractive visuals using these features provided by the D3.js custom visual, hence, making it is a very powerful development tool.

Despite these commendable features, it does have the following two limitations:

- The process of data input from the user has certain shortcomings. The user must add all the required fields under a single heading of dataset. For the code to understand the input data, either the user needs to enter the fields in a specific order or the field's name needs to match with the labels mentioned in the code. This process is not responsive and hence puts a restriction on the user. If the user wants to enter the fields in a random order along with a name that is different than the one in the code, then the user will have to modify the code before using it.

- Another limitation of the visual is that it does not allow a user to modify the basic formatting options provided by default visuals. There is no way to edit the text size in the visual other than manually changing it in the code. Plus, one does not even have any option to update the tooltip format along with the data input structure. Even though there is an option to select various colours in this visual for different elements of the chart, the limitation here is that these colour options do not have a label to specify which element the colour is applied to. There are a total of 8 colour selectors, but they are labelled as Colour 1, Colour 2 and so on. So, the user might need to identify the linked element using the trial-and-error method.

2.2.4 Node.js Method

Node.js and PowerBi Visual Tools package can be used to create a custom visual in the proper way. Using this method, we can create any kind of visual with all the custom formatting options and data options available. We can also get our custom visuals certified once the visuals are created using this method.

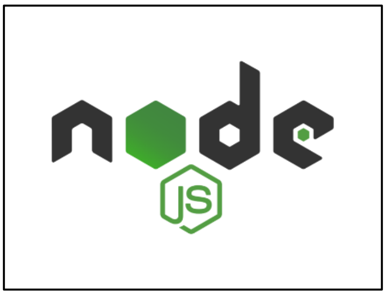


Figure 4: Node.js Approach to develop visuals

All the visuals, i.e., the default visuals available in the visualization pane and the visuals available on the AppSource have been created using the Node.js Approach. To create a visual this way, we need to install Node.js and Power BI Visual Tools package on our system. After setting-up the environment, we can create a new project. There are mainly 4 files that we might need to edit in order to create a custom visual. They are the Visual.ts file, the Settings.ts file, the capabilities file and the pbiviz file.

The Visual.ts file is the one in which we write the code and the logic on the basis of which the chart will be rendered. This is written using TypeScript language. Here we can define how the Visual will update itself when the input data changes, for example interaction with slicers or changing input fields, etc.

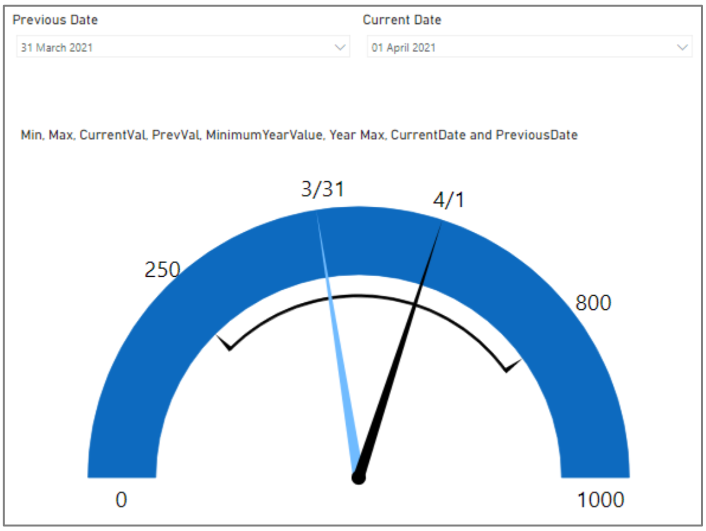
Next, the Settings.ts file helps us to link the capabilities to the Visual.ts file. This code fetches all the user inputs like the formatting options and the data fields from the capabilities and load them into the Visual. This file acts as a connector between the User Inputs and the Back-end of the Visual.

In the Capabilities file, we can define all the input fields as various Data Roles. Using these we can provide the user the ability to enter the field under the proper label. We can define whether the input field will be for a measure or a column or both. We also have the option to work with objects. These include all the various formatting options like the colour selector, toggle buttons, text input, numeric inputs, drop down list and so on. These can be modified and linked to various chart elements thereby providing several additional functionalities to the custom visual.

Lastly, we can also define a Data Mapping view. The data mapping view is the way the input fields will interact with each other. There are multiple options like single, categorical, tabular and matrix. Single can be used while creating a visual like the card visual. Similarly categorical can be used while creating a visual like bar charts and so on. We can also use this to limit the number of fields entered under a label.

Once all these three files are completed and the Visual is ready to be used, we can define the Visual name, its version, developer's name, etc in the pbiviz json file. We can then simply compile and package the file into a '.pbiviz' extension file.

2.3 Dual Axis Gauge Custom Visual



Previous Date: 31 March 2021

Current Date: 01 April 2021

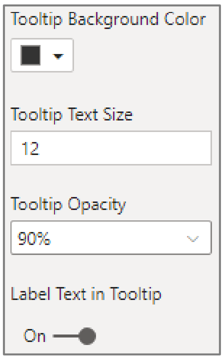
Min, Max, CurrentVal, PrevVal, MinimumYearValue, Year Max, CurrentDate and PreviousDate

Figure 5: Dual Axis Gauge

One of the main projects that I worked on is developing a Dual Axis Custom Gauge. The default gauge visual in Power BI only has two pointers to display the previous value and the current or target value.

The need for creating this visual was to display additional variables on the gauge and provide a better insight using the data. Such a visual was available within the organisation in an Excel environment. However, they required a similar visual in Power BI for further empowerment of their visualisation abilities.

This visual is designed to display the minimum and maximum value within a specific time range and compare the value between any two dates selected using the current and previous date slicers.



Tooltip Background Color

■ ▾

Tooltip Text Size

12

Tooltip Opacity

90%

Label Text in Tooltip

On

Figure 6: Customizable Formatting Options

This Visual provides the user with various formatting options like the colour of the pointers and the arc, the customizable tooltip and the text size and its colour. This is created using the Node.js approach and hence can be easily modified to implement the latest features that might be released by Microsoft for Power BI. I believe, this visual can have wide applicability in any area where the user is interested in knowing the extremes seen over a particular time period and make a best judgement based on recent comparable data trends.

2.4 Practical Applications

The visual can have several applications from a practical standpoint. It can be considered as an upgrade on the traditional single axis gauge available in Power BI as it helps us display some additional data points within the same format. Some examples of possible applications are:

- Stock market price fluctuation over a period
- Credit risk via loans over months or years
- To show the target value for a task like sales vs the accomplished value

PROJECT 2: VBA AUTOMATION PROJECT

3.1 Purpose

Automation of processes can help users resolve several issues with respect to human errors, time investment vs reward conflicts and can also help increase the efficiency of the final output. It also ensures consistency and helps to make reporting easier. This project aims to improve upon certain manual processes carried on within the reporting team.

3.2 VBA Marcos

Visual Basic for Applications (VBA) is part of Microsoft Corporation's (NASDAQ: MSFT) legacy software, Visual Basic, which Microsoft built to help write programs for the Windows operating system. Visual Basic for Applications runs as an internal programming language in Microsoft Office (MS Office, Office) applications such as Access, Excel, PowerPoint, Publisher, Word, and Visio.

VBA allows users to customize beyond what is normally available with MS Office host applications—VBA is not a stand-alone program—by manipulating graphical-user-interface (GUI) features such as toolbars and menus, dialogue boxes, and forms. You may use VBA to create user-defined functions (UDFs), access Windows application programming interfaces (APIs), and automate specific computer processes and calculations.

Within MS Office applications, Visual Basic for Applications allows users to perform myriad functions that go beyond simple word processing and spreadsheet operations. For the typical user, VBA helps to make frequent everyday tasks less repetitive via macros.

Macros can automate just about any task—like generating customized charts and reports, and performing word- and data-processing functions. For example, you can write a macro that, with a single click, will make Excel create an entire balance sheet from a series of accounting entries in a spreadsheet.

3.3 Steps in VBA Automation

This project has various steps to it. The first step is to automate the task of loading the data into excel sheets from the system generated files. The user needs to select the required file which will enable the code to fetch the data from the various columns. Here, we check for each column and in case the code is not able to locate a specific column in the source file, it allows the user to manually intervene and select the respective column in order to continue with the data loading task. This results in a smooth and responsive data loading process, even when there are any changes in the source file.

After this step, various other columns are added to the existing database and necessary formulas are applied to complete the dataset using the loaded data. This task is done in order to perform additional operations on the data.

Next step is to filter and sort the data on the basis of the predefined custom lists. This sorted and filtered data is pasted into formatted tables on the basis of various categories. This can be performed in two ways. One, by creating Pivot tables. Second, by creating a new table and using the auto filter functions available in VBA. These tables' cells are then formatted for the purpose of uniformity. The operations performed here are format of headers, alternate row colours, number format, trailing zeros, etc.

In the end, these tables are then further sub-categorized into smaller tables. For instance, these tables can be divided based on region and tier, to help the user to view the required section's data with ease. Even these tables are formatted with a specific table format and are also given specific table headings to make it easier for the user to understand what data the table contains.

This process can be repeated for several data sources at different points in time. Data from different sources are compiled in the same excel workbook and hence various tables are created for easy reporting.

The same process when done manually can be a very tedious one. This automation helps to save a lot of time, approximately 10 hrs monthly.

3.4 Project Applications & Limitations

This project was developed on dummy data in line with a live project of the company and was supposed to immediately be deployed within the system. Therefore, the reporting is restricted only to a brief explanation of the process without any specific details on the code snippets and the dummy data sets used for the purpose of development.

PROJECT 3: POWER BI – EXCEL INTEGRATION PROJECT

4.1 Purpose

There is no support for dynamic interaction between Excel and Power BI Service. The use of VBA Macros and APIs for importing images of visuals required the Report’s URL and resulted in the process not being dynamic. To resolve this issue, it was necessary to find an innovative solution for importing the visual without needing manual intervention in terms of finding a specific report URL.

4.2 About Selenium VBA

Selenium VBA is a Windows COM library that uses the popular Selenium web testing tool.

It makes it possible to automate web browsing using Excel and VBA code or using a simple VBS file executed by double-clicking on it. This application works with real browsers, which makes it possible to interact with dynamic contents created by JavaScript.

In this project, this library is used to control the Chrome Driver, log in to the Power BI Service and find various Web Elements on the Webpage. This library is used to navigate through the Power BI website and perform certain operations like click on required web elements to reach the necessary visual and copy the same to the clipboard.

4.3 About Web Scraping

Web scraping is the process of using bots to extract content and data from a website.

Unlike screen scraping, which only copies pixels displayed onscreen, web scraping extracts the underlying HTML code, and with it, data stored in a database. The scraper can then replicate the entire website content elsewhere.

Web scraping is used in a variety of digital businesses that rely on data harvesting. Legitimate use cases include:

- Search engine bots crawling a site, analysing its content and then ranking it.
- Price comparison sites deploying bots to auto-fetch prices and product descriptions for allied seller websites.
- Market research companies using scrapers to pull data from forums and social media (e.g., for sentiment analysis).

4.4 Tools Required for the Project

- MS Excel – VBA
- Selenium Basic (SeleniumBasic-2.0.9.0.exe)
- Chrome Driver (Download according to the version of the Chrome browser)
- Other supported Browsers include Firefox, Edge, Internet Explorer, Opera and Safari

4.5 Process

The entire process can be divided into three parts:

- User Inputs
- Back-End Processes
- Output

User Inputs

- On running the Macros, the user will be prompted to enter the Report Name, Page Name and Visual Title Name.

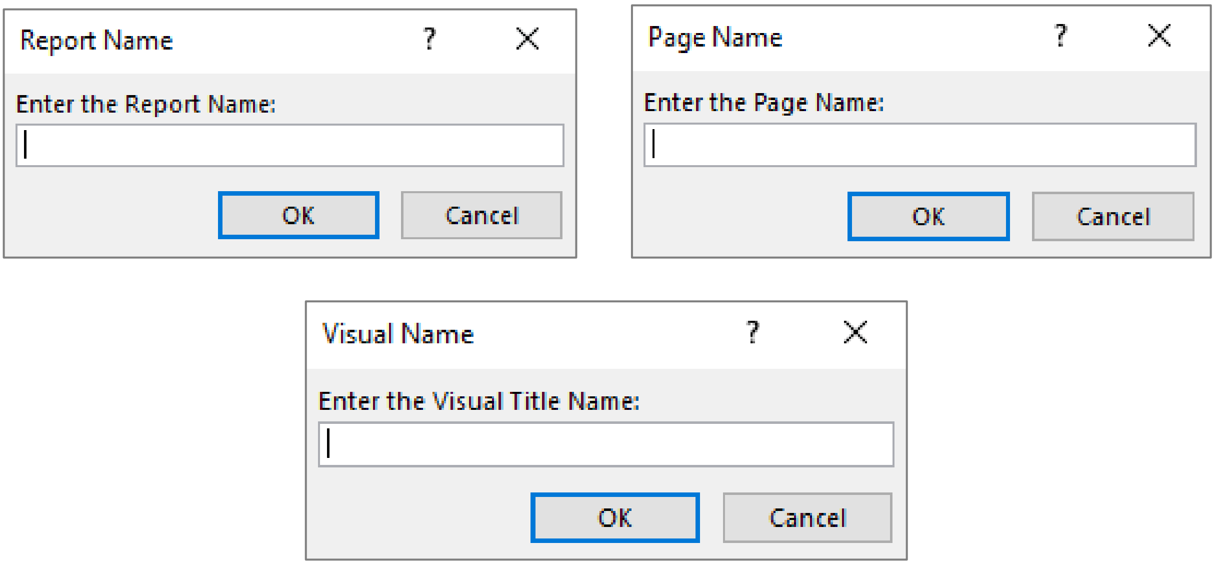


Figure 7: User Input Boxes for Power BI – Excel Integration

- The *Report Name* is required to help the code identify the required report in the *My Workspace* section of the Power BI Service.
- The *Page Name* is required to identify the required page from the *Pages Navigation* section in the Report.

- The *Visual Title* is required to select the Visual and perform the copy operation.
- Note:
    - All the Names entered in the Input Boxes should exactly match the Report, Page and Title Name. The search is *case-sensitive.*
    - The visual cannot be copied if it does not have a title as the code will not be able to uniquely identify the visual to be copied.
    - In this project the Sign in process has been setup of one specific user and the code searches for the report in the My Workspace section.

The code can be modified to accept the user's credentials to login to his/her Power BI Account and also ask the user to enter the Workspace name where the report is located.

Back-End Processes

- Initialize the Chrome Driver and accept the user inputs.
- The Selenium library can work in two modes:
    - User-Interface: This is the default mode and it displays the user interface of the browser as the code is executed. The user can see the actions performed by the bot on the websites.
    - Headless: In this mode, the bot performs all its actions without the browser's user interface being displayed to the users. All the processes occur in the background.
- The Chrome Driver goes on to the cloud service and logs into the User's Power BI Service account.
- Next, the bot navigates to the *Content section* in the *My Workspace* where it can search for the required report using the *Report Name.*
- The bot then searches for the *Page Name* inside the opened Report.
- Bot then tries to find the required visual using the *Visual Title*. When found, it performs the copy operation using the *copy visual as image* icon option provided by Power BI.
- The bot returns to the excel workbook and pastes the copied image in the worksheet.

- The bot can also be programmed to fetch the report’s URL from the Chrome Driver alongside the image of the visual.

Output

- We can get the Power BI visual in the worksheet along with the hyperlink which can direct us back to the Report.
- We can also get the URL of the page of the report from where we have fetched the visual.

4.6 Future Scope & Development

- We can create user forms to let the user enter all the required details at the same time.
- We can customize the code to fetch multiple visuals from the report at the same time.
- We can further extend the functionalities by using python to implement the same procedure.

4.7 Reporting Limitations

For reporting purposes, the process has only been explained in the form of a brief overview. Addition of code snippets and execution visuals was restricted due to system limitations on file transfers outside the remote desktop.

OTHER EXPLORATORY PROJECTS

5.1 Power BI Tutorial

5.1.1 Purpose

Power BI was relatively new for many of the team members as the company has recently transitioned to it as a Reporting Tool. Thus, I was asked to make a tutorial on it. The main objective of this project was to help users who are not familiar with the application, to have a one stop source of necessary information required to get started with Power BI.

5.1.2 About Power BI

Power BI is a collection of software services, apps, and connectors that work together to turn unrelated sources of data into coherent, visually immersive, and interactive insights. The data may be an Excel spreadsheet, or a collection of cloud-based and on-premises hybrid data warehouses. Power BI allows us to easily connect to various data sources, visualize and discover what's important, and share that with anyone or everyone.

Power BI consists of several elements that all work together, starting with these three basics:

- A Windows desktop application called Power BI Desktop,
- An online SaaS (Software as a Service) service called the Power BI service. Power BI mobile apps for Windows, iOS, and Android devices, and
- Power BI Desktop

5.1.3 Features

- In this document, I have tried to cover various key points like how to load data from different sources, different operations that can be performed on the dataset, how to create reports and dashboards in Power BI and so on.
- Covered the Query editor tool and explained its user interface, all the different data types available in Power BI, pre-processing functions, append and merge queries functionalities to transform the data.
- I have also tried to cover some important concepts like data modelling and data schemas that can be useful to build efficient reports.
- Explained the types of cardinalities available, functions of a disconnected tables, cross filtering directions and so on.
- Focused on how to sort and filter the data in Power BI data view.

- I have explained various types of visuals available in Power BI along with their uses. Also included the most commonly used DAX functions along with its explanation and examples.
- Displayed various types and features of slicers along with various drill modes available.
- I have explained various Power BI tools available like Power BI desktop, service and app. I have covered their individual features and also made a comparison on these individual tools.
- Had a look at different components in Power BI service like Workspaces, Datasets and Reports.
- Covered a detailed process description on building dashboards along with its advantages.
- Apart from this, I have added a Best Practices section which can be helpful for a new joinee to understand the standard practices used by the team to build Power BI reports.

5.2 Zebra BI

5.2.1 About Zebra BI

- Zebra BI comprises the most intuitive visualization tools to take Power BI and Excel reports to the next level and deliver insight from the data in record time.
- Zebra BI has two products:
  - Zebra BI for Excel
  - Zebra BI for Power BI
- It provides various reporting tools which are fully customizable, prepared according to the latest best practices, and easy to use.
- Zebra BI not only provides an Excel Plugin and Power BI visuals, but also provides a variety of templates, webinars and expert help if needed. (Depends on the plan selected)
- Zebra BI Visuals can be easily exported to PDF, PowerPoint, etc.

5.2.2 Zebra BI for Power BI

- Zebra BI provides two custom visuals which are Microsoft Certified and International Business Communication Standards Certified (IBCS). Each visual is actionable and has multiple reporting and formatting options.
- Zebra BI provides one visual to render charts and it is named Zebra BI Charts.
- The other visual is used to render tables in a graphical format to convey additional information and is called Zebra BI Tables.

5.2.2.1 Features in Zebra BI Visual

- *Chart Slider:* By hovering over the visual, we can see two arrows on the left and right side of the visual. This can be used to change the chart type of the visual. The Zebra BI Chart has 5 different views and Zebra BI Tables has 2 different views.
- *Responsive Layout:* These Visuals render the charts according to the space availability. If more space if available, instead of increasing the current chart size it will help to visualize data using additional charts and utilize the available space effectively.
- *Small Multiples:* The Visuals are provided with an additional dimension called Grouping. This can be used to categorize the data and help the user to visualize various categories using multiple small charts. All the charts will be automatically rendered on to the same scale (automatically synchronized Y-axis). It has three different layout formats to choose from.
- *Fully Responsive Hierarchy Tables:* Advanced hierarchical tables will automatically calculate absolute and relative variances and adjust the display of all the elements to available space. We can add an unlimited number of levels and show exactly the ones that we want. They are ideal for presenting a year-over-year analysis and to evaluate trends in the dataset. These hierarchical tables are ideal for visualizing Profit & Loss reports (income statements).

5.2.2.2 Zebra BI Charts

- Zebra BI Charts features charts with a horizontal axis, such as waterfall, variance, column, area, line, dot, lollipop and other types of charts.
- We can create Single or Combination (combo) charts with Variances or Forecasts.
- This can also be used to create small multiples of any selected chart type.
- This visual is mainly used to display time-related data. It is useful to compare the data with previous year, or can also be used to compare the current year's data with a plan.
- This visual can automatically calculate previous year's data, plan or forecast variances, and year-over-year growth rates.
- It is great to report the variances on a time-related data.

5.2.2.3 Zebra BI Tables

- Zebra BI Tables is a powerful table/matrix visual in Power BI.
- We can build extremely flexible tables with embedded charts.
- Zebra BI Tables can provide various features like expand/collapse rows, reorder columns, do a P&L calculation, top N analysis and much more.
- Using this visual, we can add unlimited number of levels to create a hierarchy table.

- We can easily manage the column structure by:
    - Presenting column grand totals and subtotals Expand/collapse groups of columns
    - Renaming any column in a table (including subtotals and grand totals)
    - Sorting by any column or total in 1 click
    - Displaying any column (including the totals) as a table or a chart

5.2.3 Zebra BI for Excel

- Zebra BI provides a Plugin for excel which contains all the charts present in the Zebra BI for Power BI visuals.
- It has variety of chart options which are not available in Excel by default.
- This can be used to display time-related data, variances, contribution and many more.
- The charts can be easily embedded into tables. It provides with a feature to resize the chart and merge it with a table to create a table-chart report.
- User can easily highlight the required data and can also add comments to the charts using the formatting options provided in the visuals.
- Zebra BI charts also provide a feature to synchronize the chart's axis to help the user to compare charts easily.
- Zebra BI for Excel also provides small multiple charts feature to enable the user to view graphs for various categories effectively.
- Reports made with Zebra BI can be exported to PDF, PowerPoint and Static Excel sheets. Only the user with Zebra BI Plugin installed can edit the charts.
- Some of the Chart types available in Zebra BI for Excel are Variance charts, "Waterfalls", "Hills & Valleys", "Lollipops" and much more.

5.3 Camunda

5.3.1 About Camunda

- Camunda Platform is an open-source workflow and decision automation platform.
- Camunda Platform ships with tools for creating workflow and decision models, operating deployed models in production, and allowing users to execute workflow tasks assigned to them.
- It is a lightweight, Java-based framework.
- It can be used as a standalone process engine server or embedded inside custom Java applications.

- It offers non-Java developers a REST API and dedicated client libraries to build applications connecting to a remote workflow engine.
- It provides a Business Process Model and Notation (BPMN) standard compliant workflow engine and a Decision Model and Notation (DMN) standard compliant decision engine, which can be embedded in Java applications and with other languages via REST.

5.3.2 Features provided by Camunda

- The Camunda Modeller desktop application allows developers to create and edit BPMN process diagrams and DMN decision tables.
- Created files are deployed in the Camunda Engines, which use a BPMN parser to transform BPMN 2.0 XML files, and DMN XML files, into Java Objects, and implements BPMN 2.0 constructs with a set of BPMN Behaviour implementations.
- Typical use cases for the Camunda BPMN Workflow Engine can be micro services orchestration and human task management.
- The Camunda DMN Decision Engine executes business-driven decision tables.
- It is pre-integrated with the Workflow Engine but can be used as a stand-alone application via REST or inside Java applications.
- Camunda's additional web applications provide the following tools for developers and business users:
    - *Cockpit:* A tool for technical process operations enabling users to monitor workflows and decisions in production, to analyse and solve technical problems.
    - *Tasklist:* Allows end users to work on assigned tasks and provides additional visibility when using the Camunda Workflow Engine for human task management.
    - *Optimize:* An analytics and reporting tool to identify errors and bottlenecks in workflow processes.
    - *Cawemo:* A collaborative modelling tool allowing multiple users to create, edit and specify BPMN process diagrams. The tool was originally a standalone web app created by Camunda developers and was brought into the main product stack in May 2019.
    - *Admin:* Allows users to manage Camunda web applications or REST API users. Existing user management can be integrated via LDAP.

5.4 SingleStore

5.4.1 About SingleStore

- SingleStore DB is a distributed, relational database that handles both transactions and real time analytics at scale.
- Querying is done through standard SQL drivers and syntax, leveraging a broad ecosystem of drivers and applications.
- SingleStore is a scalable SQL database that ingests data continuously to perform operational analytics for the front lines of your business.
- Ingest millions of events per second with ACID transactions while simultaneously analysing billions of rows of data in relational SQL, JSON, geospatial, and full-text search formats.

5.4.2 Features

1. Both a transactional database and analytical data warehouse:
    - SingleStore is designed to act as both a transactional (row store) database, and an analytical (column store) data warehouse.
    - This unique technology eliminates the need for an organization to spin up a transactional database for row-level writes, and a separate analytical data warehouse for historical analysis.
    - Organizations can concurrently write and read from the same distributed system using ANSI SQL.
2. Real-time analytics:
    - The ability to stream operational data into relational format while concurrently reading data means SingleStore can discover insights on the latest events or compare those with historical data.
    - This approach enables real-time financial reporting or insights on live customer interactions with a single unified database.
    - The integrated architecture can replace more complex systems that require copying of data from a transactional database or data lake into a dedicated data warehouse for analysis.
    - Most relational database management systems (RDBMSS) cannot write and read data at the same time.

- This unique architecture makes SingleStore an attractive technology for organizations that want to analyse their business continuously while providing a historical reference store for predicting events or for making strategic decisions.

3. Scalability:
    - SingleStore scales similar to a Hadoop architecture in that SingleStore doesn't require exotic hardware; it scales linearly on commodity hardware.
    - This allows SingleStore to run on-premises, in the cloud, or as a managed service.
    - It's also possible to scale SingleStore instances without taking the cluster offline, which means SingleStore can be easily scaled "just-in-time" when new performance or storage is necessary.

LEARNINGS AND CONCLUSION

This internship has been an amazing avenue for learning not just on the professional but also on a personal front. It provided me with a variety of opportunities to develop my technical as well as interpersonal skills. I believe that all my projects presented a new challenge for me to overcome. A few takeaways for me through this internship as listed below:

- All my projects throughout this internship required me to ideate on newer methods and find innovative solutions to each problem. This helped me get in touch with my creative side and also allowed me to learn about several new areas in the fields of risk, reporting, programming languages and customized tools.
- As someone who had never used Power BI before, I learnt how to create reports and dashboards for effective reporting. I could understand the working of Power BI, the different chart types available and the process of developing custom visuals.
- Programming has always been one of my strengths and I was glad that I got the opportunity to expand into different languages during my internship. Even though I had worked on Basic VBA codes before, the automation project introduced me to some advanced operations like pivot tables, sorting, filtering and so on.
- The research on the third-party applications proved fruitful in understanding various advanced and customized tools which can help in improving the work routines of my team at Morgan Stanley.
- I was introduced to the concept of Web Scraping which allowed me to explore the various dimensions that it can be used in. I found this concept very interesting and look forward to implementing it in some real-life projects in the future.
- On a personal level, over the course of these 10 weeks, I have been able to develop some important qualities like time management, task prioritization and working within a team. The corporate exposure I gained really helped me improve on my communication skills.
- The additional learning resources and training sessions held regularly by the HR team introduced several new concepts of finance, risk management and provided an opportunity to upskill myself.

In conclusion, this internship was a wholesome experience for me, and I am grateful for all the learning opportunities that came my way.

REFERENCES

- https://docs.microsoft.com/en-us/power-bi/visuals/service-r-visuals
- https://docs.microsoft.com/en-us/power-bi/connect-data/desktop-python-visuals
- https://www.investopedia.com/terms/v/visual-basic-for-applications-vba.asp
- https://powerbi.microsoft.com/en-us/developers/custom-visualization/
- https://charticulator.com/
- https://docs.singlestore.com/
- https://camunda.com/
- https://zebrabi.com/