

TECHNOLOGY CONSULTING

IN THE GLOBAL COMMUNITY

Final Consulting Report

National Development
Bank of Palau

Halanna Yuh
August 2022

Carnegie Mellon University





National Development Bank of Palau (NDBP) Executive Summary

Student Consultant, Halanna Yuh
Community Partner, Claire Harvey

I. About the Organization

The mission:

To initiate and promote sustainable economic development within the Republic of Palau.

The national development bank is a 100% government owned bank. It does not provide commercial daily activities and doesn't keep any money. Its main operations is to provide commercial and housing loans to the people of Palau.

II. Reduce manual labor by automating the completion of loan product documents

The bank has various loan products (i.e. solar, housing, fishing) and each of them has their own set of documents that the bank staff has to fill out. Filling out many documents for each individual customer is a time-consuming and repetitive task.

A VBA application was created so that bank employees can submit all the information about the customer that is needed to go into loan documents, and once they hit a submit button, the filled-in documents will be saved to a folder for that customer. Videos were created to train employees on how to use and configure the application.

This objective was partially met. The application was able to reduce the time spent completing loan documents from 30 minutes to less than 1 minute. However, the application had not been released to all NDBP employees yet by the end of the consulting engagement.

NDBP was presented with the following recommendations:

1. Invest time in adding more loan products to the application.
2. Set standards on maintaining the application.

3. If the VBA application is too difficult to maintain, consider using Formstack's Documents product at <https://www.formstack.com/products/documents>. Compared to the 0 costs of using the VBA application, Formstack's cheapest tier starts at \$92/month and there are limits on how many documents can be autofilled, the number of users, and number of loan products that can be added per pricing tier.

III. Improve allocation of employee workload by simplifying the generation of monthly lending reports

Every month, Karla generates a lending report that is then given to Claire and the NDBP Board of Directors during board meetings to guide business decisions. Karla spends up to 4 days compiling this report. The time taken to make this report should be reduced so that Karla can spend more time doing more important analysis work.

Appendix E analyzes 2 potential solutions a future student consultant can consider. Both solutions would require a script to automatically output the KPI graphs needed for the report; the only difference is one solution suggests continuing to use Excel spreadsheets to store client data while the other suggests migrating data to a MySQL database.

NDBP was presented with the following recommendations:

1. Ensure the next student consultant can access this NDBP Final Report, particularly Appendix E.
2. Have Karla document the steps to generate the lending report.
3. Present Appendix F to Insite.
4. Collaborate with the Financial Institutions Commission (FIC).

IV. Additional Recommendations

It is recommended that NDBP update Microsoft Office versions for all employee laptops to Office 365 and reach out to Halanna Yuh at halanna@halanna.com for brief assistance after the consulting engagement if needed.

Consulting Partner

Claire Harvey
charvey@ndbp.com

National Development Bank of Palau
<http://www.ndbp.com>

About the Consultant

Halanna Yuh
halanna@halanna.com

Halanna Yuh graduated from Carnegie Mellon University with a Bachelor of Science in Electrical and Computer Engineering and a minor in Business Administration. She will be starting her career as a product manager at Capital One.



National Development Bank of Palau (NDBP) Final Consulting Report

Student Consultant, Halanna Yuh
Community Partner, Claire Harvey

I. About the Organization

Organization

The mission:

To initiate and promote sustainable economic development within the Republic of Palau.

The national development bank is a 100% government owned bank. It does not provide commercial daily activities and doesn't keep any money. Its main operations is to provide commercial and housing loans to the people of Palau.

Facilities

The bank operates in one building built on a land size of approx. 600sq mt. The building is secured with CCTV cameras capturing the entrance and the parking areas. A security personnel guards the building from 6pm – 6am. Each employee is provided with a personal laptop by the bank. There are 3 desktops total. 5 of the bank employees own an individual printer. The employees that own an individual printer include Claire, Karla, Prescott, and Shannin. Claire predominantly uses the shared printers because it is quicker, while the individual printer is mainly used when dealing with confidential information. The bank is working on upgrading the printers and adding 2 more printers. All employees have access to a printer. There are 2 colored printers. All the printers are 2 in 1 scanners as well. There is one scanner machine separately provided to the front desk. The bank recently purchased 2 new all-in-one fax machines, photocopiers, and printers.

Programs

The bank provides two types of loan programs, housing and business.

The different types of housing loans provided are as follows:

- Housing loans
- First time homeowner

- Micro-finance loan – housing
- Pre-development loan housing

The different types of business loans are as follows:

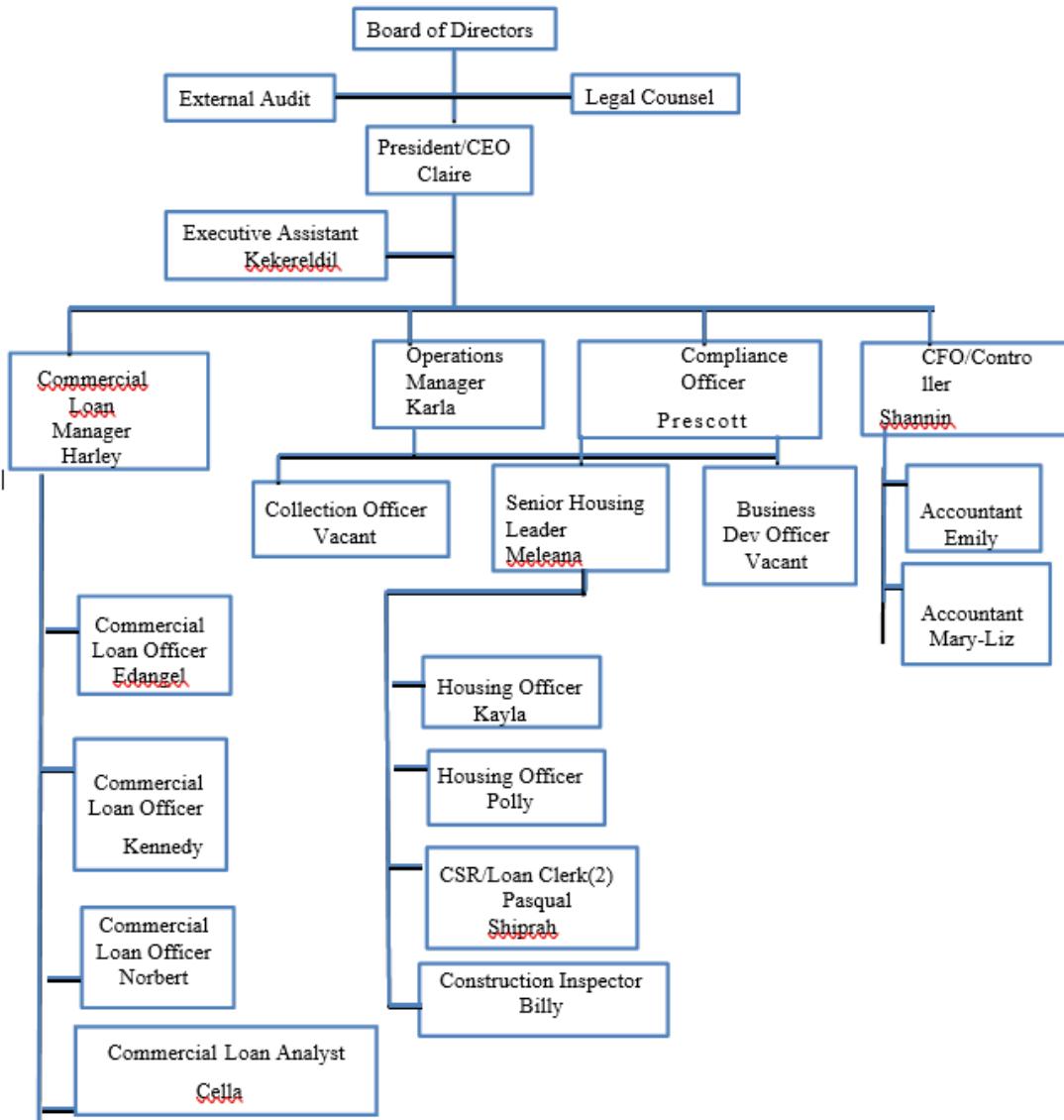
- Agriculture / Aquaculture loan
- Microfinance
- Pre-development loan
- Fishing loan
- Tourism loan
- Business loan / Commercial loan
- Lines of credit - customer has certain level they can withdraw funds up to, supposed to be paid off every year, generally used by businesses as working capital

The bank also offers the following 3 programs:

- Women, youth, medium, and small enterprises – This is a program that was funded through Taiwan. The program is aimed at supporting women, youth, medium and small enterprises. This is a new program that has been going on for 2 years.
- COVID program – This program provides low interest rate loans to help businesses survive COVID.
- Energy Efficiency Subsidy Program – NDBP offers a subsidy to new home owners who install energy efficiency measures.

Staff

NDBP currently has a staff of 17 people. Leadership consists of Claire Harvey, President and CEO of NDBP, and 3 managers: Harley, Karla, and Shannin. Since the bank offers housing and commercial loans, they have 2 separate departments that oversee the respective loan programs.



Technology Infrastructure

All employees have access to the software needed to perform the required activities. Applications used by them regularly are mentioned below. Some have been trained in the software, some did not get any training and learned through use only. This is applicable to the software which has been used for the last 15 years. Whenever a software update is performed, they are trained for it.

Operating Systems	Everyone on windows 10
Server	Windows Server 2012

Banking system for loans management	ASI Insite v 8.0.56
For Accounting	Sage ERP MAS90 v 4.50.5.0
For commercial loans(loan analysis)	Optimist version 7
For backup	Acronis
Firewall	Untangle
Antivirus	Kaspersky endpoint security 10

Internet connection: There is one Wi-Fi in the CEO office connected through the LAN. There is also a public Wi-Fi option for customers called PPUC Palau Public Utilities Wi-Fi option, though it is rarely used.

Technology Management

Technology is managed by an external agency called the [Computers Plus](#). For every small or big technical problem, an IT technician is called to resolve the problem. The consulting firm charges them a fixed monthly fee, and everything over that is charged on an hourly basis.

All tasks related to the backing up of critical data, installing and updating the software and virus detection is handled by the external IT consultant only.

Currently, the backup of all the information is done in 2 ways. Firstly, a daily backup is done on an external hard disk (size 6TB). This disk is then replaced by another hard disk every Friday by the Computers Plus technician and this backed up hard disk is kept in their office. In this way, a remote backup of the data, up to the last week, is always present in the Computers Plus office.

Secondly, a daily backup is done in another hard disk (size 3TB). This is removed from the server room every day while leaving the office and kept in personal possession by the CFO. This secondary backup is done as in the past when the organization was hit with malware, the bank had lost that week's data, as the remote backup at the Computers Plus was for the previous week.

All the databases related to software, except for Optimist, are backed up through the server.

The bank can always get backup for banking information from Insite.

Both the commercial as well as housing loan departments use laptops to store all the client related as well as loan related information on a shared drive.

Technology Planning

All the decisions about technology planning is done by the President and the board members. There is no technology planning committee. If there is any need for a new software update or new hardware requirement for any of the employees, the managers, or the external IT consultant brings the concerns to the President's attention and then the decision is made accordingly.

Communication

Each employee has access to the shared folders created for them. The information internally is passed through the shared folders. The shared folders are backed up in the server. Other information is shared through email. NDBP has its own official email shared with Gmail.

The bank has its own website (<https://www.ndbp.com>), but it has not been updated since a long time.

Information Management

Information critical to the bank are the documents related to loans and accounting. The 2 departments, loans and accounting, have their own ways of information management.

Loans

The management of loan information is mainly through electronic systems. ASI Insite is the banking system NDBP uses for loan management. Once a loan is closed, the loan information is added to Insite. Prospective clients' information cannot be added to Insite, so their information is tracked in the loan log Excel spreadsheets. Insite is also updated by the loan clerks Pasqual and Shiprah when a customer makes a payment. Insite can be edited by the managers and the loan clerks, while the loan officers are only able to view Insite and cannot make edits.

The commercial loan officers use Optimist for loan analysis to help them make decisions on how to maintain profitable commercial loan portfolios.

In terms of loan reporting, NDBP's Operations Manager, Karla, is responsible for compiling a final lending report to be sent to the President and the Board of Directors to evaluate the performance of the bank. All the loan officers fill out a disbursement report and the housing and commercial loan departments fill out the housing loan logs and commercial loan logs respectively. These reports are kept in Excel spreadsheets. On the 5th of every month, these reports are sent to Karla to be compiled into a final lending report. Karla also uses Report Writer from Insite to extract a raw Excel sheet containing information about each loan. This information is used to create an accounting provision report in Excel for the bank's portfolios, and the analysis from this report is included in the final lending report.

Accounting

Shannin, CFO and head of NDBP's accounting department, uses a program called SAGE MAS90 and information stored in various Excel spreadsheets to generate the Finance Presidents Report. This report is given to Claire and the Board of Directors to review.

Past attempts at automation include automation of the loan reporting process that Karla is responsible for. The previous student consultant who worked at NDBP in 2019 created Excel macros using VBA to reduce the time it takes to generate the final lending report. A training document was shared with the employees to guide them to install macros in the future. However, NDBP was unable to sustain

this solution due to a short training period and a lack of experience in VBA to relink the connections between documents when changes were made to the documents.

Business Systems

All accounting related processes are managed in SAGE MAS90.

II. Reduce manual labor by automating the completion of loan product documents

Motivation

NDBP has various loan products (i.e. they may give out solar, housing, fishing loans) and each loan product has their own set of documents that NDBP's loan officers have to fill out manually for clients applying to those loans. Manually filling out many documents for each individual client is a time-consuming and repetitive task that can take between 30 minutes to up to 2 hours depending on the officer. To provide a perspective on the impact of these issues, NDBP has a total of 8 loan officers between its commercial and housing loan departments. Loan officers fill out loan documents for clients on a nearly daily basis.

Outcomes

Activities and Resulting Outputs

The following activities were performed to achieve the goal of automating the completion of loan product documents.

Understanding the problem and analyzing solutions

The first 3 weeks were spent researching the context and impact of the problem of completing loan documents manually. I was first introduced to the problem when Ms. Claire Harvey presented me with the loan documents that loan officers have to fill out for clients in their solar loan program. The documents for the solar loan products, as well as for all of NDBP's other loan products, were stored as Word documents. Upon preliminary research on solutions on how to automatically fill in certain fields on Word documents, programming Excel macros with VBA was a common solution due to ease of customization. I spent the first week learning VBA and built a simple VBA application as a proof of concept that can autofill several fields on a single document. During the second and third weeks, I researched possible automation tools that could achieve the goal and compared these solutions with building a custom VBA application. The output of this research is a list of functional requirements to achieve the goal and a solutions analysis matrix of how well each solution fulfilled the functional requirements (Appendix A). The decision to build a custom VBA application came down to ease of customization, no extra costs for the bank, and the loan officers' familiarity with using Office tools. Once it has been decided that I will build a custom VBA application, I drafted a workflow of the custom VBA application that was sent for review by Ms. Claire and Karla (Appendix B). The workflow has been iterated upon multiple times throughout the duration of the consulting engagement, so the final application workflow has some slight variations. Refer to the "How To Use Autofill Application" training video in Appendix C to get a complete idea of how the autofill application works.

Implementing the application

Weeks 4-8 were used for implementing the loan document autofill application. The output of this activity is a VBA application that allows the bank's loan officers to fill out a mastersheet with the client's information, and then seamlessly auto populate Word documents for the loan product of their choice using the mastersheet with the client's information. Although the application is intended to support all NDBP loan products, it currently only supports the solar loan product since it is a time-consuming task to identify all the fillable fields for any given loan product. The intention of the consulting engagement is to sufficiently train NDBP staff on how to use VBA and the autofill application so that they have the skills to configure the application to support new loan products on their own time.

Ensuring solution sustainability

The last 8-10 weeks were focused on ensuring the sustainability of the autofill application solution. 2 NDBP employees underwent training on how to use the application and how to edit the application to support additional loan products. The employees were Karla, NDBP's operations manager, and Norbert, a commercial loan officer. Karla underwent the training because she holds a leadership position at NDBP, and Norbert was selected to participate in training because he is considered to be the most comfortable with using technology compared to the rest of NDBP's employees. 3 training videos and a supporting document for one of the videos were produced (Appendix C).

Outcomes and Indicators of Success

The success in achieving the goal of reducing the overall manual labor of NDBP employees can be measured by the following objectives:

1. The time it takes to fill out loan documents is reduced.

This objective has been achieved. The time it takes to complete loan documents can range from about 30 minutes for the most technologically savvy loan officer to up to 2 hours. The VBA application was able to complete loan documents for the solar loan product in under a minute.

The amount of time the VBA application takes to complete loan documents can vary depending on the loan product; however, it will still be drastically faster than having to fill the documents manually.

2. All NDBP loan officers frequently use the application whenever they need to fill out loan documents.

This objective has not been achieved. At the conclusion of the consulting engagement, the application has not been released to the rest of the bank's employees aside from the ones who underwent the training sessions. Karla and Norbert wished to finalize the loan documents and add them to the application before the rest of the loan officers can use it.

Risks to Sustainability

The largest risk to sustainability is that the application has not been deployed to all NDBP loan officers

at the conclusion of the consulting engagement. The concern is that if technical issues arise when the application is deployed to the entire bank and they do not know how to resolve them, NDBP may abandon efforts to use the application altogether. Information on how NDBP can receive support is mentioned in Section IV Additional Recommendations.

Recommendations

Invest time to add new loan products

Configuring the application to support new loan products can be a time consuming and tedious process. I estimate that for 1 person, adding a new loan product can take 1-2 work days, and there may be additional time spent learning how to configure the application. Even then, the time it takes for 1 or 2 employees to add the products is still significantly less than the time and effort it takes for all loan officers to manually fill out documents on a frequent basis (as mentioned in the Motivation section above, this has been found to range between 30 minutes to 2 hours). I encourage NDBP to consider the long term benefits and invest the initial time to configure the application so that all loan officers can save time in the long term.

All training materials on how to configure the application is available in Appendix C.

Maintaining the application

Here are my recommendations on how NDBP can maintain the autofill application so that everyone can benefit from it in the long term:

1. Assign no more than 3 people to be responsible for adding new loan products to maintain consistency. If someone has an idea for a change to be made to a document, contact these people. Once the changes are made, contact everyone about the changes and upload the document to a shared drive accessible to everyone.
2. To the employees responsible for configuring the application, establish consistency when designing document templates. For example, consider whether to include special characters (\$, %) in the document template or would the loan officers have to include these characters themselves when they input information into Excel. The “How To Use Autofill Application” document and video in Appendix C discusses how special characters such as \$ and % impact formatting on the final documents.
3. To the employees responsible for configuring the application, create a separate document describing what each field name means and how the input for that field should be formatted. For example, if one of the fields requires the loan officer to fill in a loan amount, perhaps it would be helpful to specify that they must input “=DOLLAR(<loan amount>)” in the Excel mastersheet (the reason for this is specified in the “How To Use Autofill Application” document and video in Appendix C). This lets the employees know what they need to do to ensure loan documents are created properly.
4. Establish a system for resolving issues. If an employee faces an error with the application, who do they contact? How do they contact the people to fix issues? I recommend that the NDBP leadership establish this system by answering these questions.

Consider using Formstack

If the VBA application is too difficult to maintain, consider using Formstack's Documents product at <https://www.formstack.com/products/documents>. However, Formstack's cheapest pricing tier starts at \$92/month and there are limits on how many documents can be autofilled, the number of users, and number of loan products that can be added per pricing tier. Refer to Formstack's pricing page (<https://www.formstack.com/pricing/documents>) for more information.

III. Improve allocation of employee workload by simplifying the generation of monthly lending reports

Motivation

During meetings between NDBP's president and board of directors, the leadership team relies on monthly lending reports to inform their business decisions. These reports are generated manually by Karla and NDBP employees store the information needed for the reports in Excel spreadsheets. This manual process is causing problems primarily with efficiency since the reports need to be sorted and rearranged after they are pulled from Insite's Report Writer, NDBP's loan management system. It can take Karla up to 4 days to pull information from the individual reports the loan officers make and the raw Excel data from Insite's Report Writer to compile the final lending report. Another issue with maintaining multiple Excel sheets is that there is redundant information stored in multiple places that do not get updated properly, leading to inaccurate information.

The reason why it is important to make the lending report generation process more efficient is because the manual work takes time away from the managers to complete the more important analysis work. Claire has a philosophy to work output known as \$10/hr work, \$100/hr work, and \$1000/hr work. The loan officers are responsible for the \$10/hr work of gathering and putting information together, the managers are responsible for the \$100/hr work of analyzing the information, and as President, Claire does the \$1000/hr work of driving the bank's business strategy with that analysis. Managers like Karla spend too much time on the \$10/hr work of copying and pasting information between Excel sheets as opposed to the \$100/hr work of providing analysis. The goal of simplifying the lending report generation process can give Karla more time to analyze whether the lending figures met the bank's target metrics or how loans in certain business sectors are performing, which would better support NDBP's mission of promoting economic development in Palau.

Outcomes

Activities and Resulting Outputs

Understanding how lending reports are generated

The first 2 weeks were spent interviewing NDBP staff to learn more about how the monthly lending reports are generated. Based on the interviews, I created a flowchart that illustrates and explains how Karla takes the information she needs to generate a lending report (Appendix D).

Analyzing solutions to simplify generation of lending reports

The next 3 weeks were spent on analyzing possible solutions for simplifying or automating the generation of lending reports. To guide my solutions analysis, I created a specification document for the system that will allow automation of lending report generation (Appendix E). It is important to note that NDBP sought help for this same problem from their student consultant in 2019. However, the

solution from the 2019 consulting engagement, which was a set of Excel VBA macros to calculate required fields for the reports, did not sustain. According to Karla, the reason for this is because she was not able to follow the training documents provided when something went wrong with the macros without the consultant's assistance.

Although my original idea was to recreate VBA macros but spend more time on training (I was unable to reuse the macros because NDBP had lost track of where they were stored), I realized that it may be better to completely convert NDBP's current database system of using Excel spreadsheets into a more flexible database system. This realization was made after I contacted the Financial Institutions Commissions (FIC), who had the same issue of maintaining too many Excel spreadsheets and wanted a faster way to complete ad hoc analysis of company data. FIC's student consultant resolved these issues by moving FIC's data to a MySQL database. To help NDBP convert out of an Excel database system, I created a solutions analysis of database systems, how well they fulfill the specification document for lending report generation, and why MySQL is my final recommendation for a database system (Appendix E).

Understanding how to better use Insite

Insite is NDBP's loan management system. Karla extracts a raw data report from Insite's Report Writer which she uses to generate an accounting provision report for NDBP. Information from the accounting provision report is then included in the final lending report. If the account provisioning report can automatically be pulled from Insite, it would save Karla some time from having to make the accounting provision report herself.

I met with Andrew Tetschner and Greg Suelter from Insite to discuss how Insite can be customized to automatically provide an accounting provision report. From the conversation, it seems that the Insite team was open to customizing Insite features to provide the report. I created a document to specify what exact actions Karla takes to convert the raw report from Insite's Report Writer to the provisioning report (Appendix F). Although scheduling a follow-up meeting with Insite to present Appendix F was a planned activity, it was never achieved during the consulting engagement. It is recommended that NDBP either present Appendix F to Insite on their own time or during the next consulting engagement.

Outcomes and Indicators of Success

The goal of simplifying lending report generation so that Karla can spend more time on analysis was not achieved during the consulting engagement. This was mainly because Goal II of automating loan document completion took greater priority due to the number of employees impacted, frequency that the solution will be used (almost daily compared to the lending reports being generated monthly), and likelihood to be achieved by the end of the consulting engagement. However, what was achieved was documentation to best support future consulting efforts to achieve this goal. This documentation includes Appendices D-G.

Recommendations

Ensure the next student consultant can access this report (NDBP Final Report)

This report, especially Appendix E, will be extremely helpful for a future student consultant to implement a solution to automate the lending reports. The 2 solutions presented in Appendix E are to either keep using Excel spreadsheets to store client data or to move to another system for data storage.

Consider how often Karla changes the formatting of the monthly lending report as this will have an impact on what solution to move forward with.

Document the steps to generate the lending report

Have Karla document the steps she takes to generate the lending report before the student consultant arrives. Although sessions were arranged to discuss this process with me, this process was complex enough that time constraints prevented me from understanding this process with enough specificity to decide on a solution. Having this process documented beforehand would give the student consultant more time to decide on and implement a solution.

Present Appendix F to Insite

Appendix F explains in detail how Karla creates the provisioning report from the raw report pulled from Insite's Report Writer. Present Appendix F to Insite partners so that they can help NDBP on configuring Insite to reduce time spent on generating the provisioning report.

Collaborate with Financial Institutions Commission (FIC)

Based on the reports written by TCinGC consultants who worked with FIC, FIC had similar issues with NDBP in terms of moving away from Excel spreadsheets as a means of storing data and automating reports. I recommend connecting with them to ask how they solved these issues, how well current solutions are working for them, and encouraging future consultants to connect with them to figure out viable solutions for NDBP.

IV. Additional Recommendations (if any)

Office 365 Update

It is recommended that NDBP update Microsoft Office versions for all employee laptops to Office 365. When deploying the application to autofill loan product documents, the difference in Office versions between what was used to build the application and the version that employees are currently using caused some errors that we needed to take extra steps to circumvent. Updating to Office 365 can prevent technical issues with future consulting projects. However, how updating to 365 will affect other NDBP software should be considered as a precaution.

Getting Support

It is recommended that NDBP reach out to the student consultant or the TCinGC program advisors for support with any of the solutions. I can be reached through email at halanna@halanna.com and I am willing to schedule a time to provide brief support should NDBP need it.

About the Consultant

Halanna Yuh graduated from Carnegie Mellon University with a Bachelor of Science in Electrical and Computer Engineering and a minor in Business Administration. She will be starting her career as a product manager at Capital One.

Appendices

Appendix A: Document Autocompletion Solutions Analysis

The following are possible solutions I explored to automate the completion of loan product documents:

1. Custom VBA application
2. Jotform: an online form builder that has a “Prefill forms” feature
3. Formstack: a workplace productivity platform. Its “Documents” product will be most useful for document autocompletion.

Solutions Analysis

The solutions above are analyzed according to the following functional requirements:

Document Autocompletion

The first set of functional requirements involves document autocompletion. The system must allow employees to:

1. Select what set of loan product documents they want to autofill.
2. Easily fill in all information needed for a client for the selected loan product in one place. This information will be used to autofill the loan documents.
3. Store the autofilled documents in a manner that is easy for employees to associate those documents with the appropriate client.

The following table summarizes how well each solution met the functional requirements.

	Custom VBA App	Jotform	Formstack
Loan product selection	YES	YES	YES
Prefill information	YES	YES	YES
Document storage	YES	NO ¹	YES

The following are reasons for a NO:

1. Jotform: After autofilling fields, there was no built-in way to export the form with the filled fields and store them in a registered account.

Customizability

The second set of functional requirements involves customizability. The system must allow employees to:

1. Add a new set of loan product documents and configure the system to autofill those documents.
2. Add or edit fillable fields to any existing document supported by the system.
3. Change the contents in autofilled documents.
4. Specify what fields are needed to complete a set of loan documents. For example, solar loan documents may require client name, client address, etc.

5. Change what fields are needed to complete a set of loan documents. For example, solar loan documents may once require a client address, but client address is no longer needed for future solar loan documents.

	Custom VBA App	Jotform	Formstack
Loan product addition	YES	YES	YES
Field addition to document	YES	YES	YES
Change document content	YES	NO ¹	YES
Specify fields	YES	YES	YES
Change fields	YES	YES	YES

The following are reasons for a NO:

1. Jotform: The form is not able to autofill unless the name of the field is on the form (i.e. client address). Ideally, the name of the field should not be on the final document, only the content. There is also little flexibility to format the form text.

Tracking Usage

The third set of functional requirements involves tracking loan products and clients supported by the system. The system must allow employees to:

1. View what loan products are being supported by the system so far.
2. View what documents have been filled out for which clients and access those documents.

	Custom VBA App	Jotform	Formstack
View loan products	YES	YES	YES
View completed documents	YES	NO ¹	YES

The following are reasons for a NO:

1. Jotform: After autofilling fields, there was no built-in way to export the form with the filled fields. The only option to save filled content was clicking the “Submit” button, which would store the filled content in a spreadsheet, but not in document form.

Solutions Discussion

The following is relevant research on the functionalities of each solution.

Custom VBA application

Building a custom VBA application can meet all functional requirements. The biggest drawback is that it may be difficult to set up and maintain.

Jotform

I reached out to Jotform support with the following message to find out how Jotform features adhere to the functional requirements:

“I am wondering which Jotform product or service can accomplish the task of autofilling forms. For my company, I need to fill out multiple form templates for a client with that client's specific information. The set of forms I need to fill out differ depending on what product the client buys.

I have an example of how I autofill forms in the attached screenshot. Let's say I have a client, Jane Doe, who wants to buy Product A. I need to fill out Form 1 and Form 2 for Jane Doe. Form 1 and Form 2 requires the following fields: today's date, client name, and client address (highlighted in gray). Thus, I have an Excel mastersheet for Product A that I fill out with Jane Doe's information. I use VBA to autofill Forms 1 and 2 with the information with Excel, and the outcome is in the bottom 2 images where Form 1 and Form 2 are completed with the necessary information. Product B may require different information (i.e. client date of birth, client citizenship) and thus I would have a separate mastersheet and a different set of forms for Jane Doe if she buys Product B.

Basically, I am looking for the following functionality:

- Create different mastersheets for each of my products. In each mastersheet, I can specify what information I need for the product the sheet corresponds to.
- Create a set of form templates for each product.
- Link the fields in the mastersheet to where they show up on the forms. A field can show up multiple times in a single form or through multiple forms, which is why I am currently automating this process with VBA.

I would like to stop using VBA and am wondering what Jotform products can accomplish what I am currently doing with VBA.

Jane Doe wants to buy product A. For all the forms I need to fill out for Jane Doe if she buys product A, I would need today's date, client name, and client address. The Excel sheet with client Jane Doe's information

A screenshot of Microsoft Excel showing a spreadsheet titled "Client Mastersheet.xlsx". The sheet contains four rows of data:

1	Today's Date	8/19/2022
2	Client Name	Jane Doe
3	Client Address	123 Street
4		
5		

I need to fill out Form 1 and Form 2 for Product A. The gray areas need to be ~~autofilled~~ with Jane Doe's information.

A screenshot of Microsoft Word showing two separate documents side-by-side. Both documents have their ribbon tabs set to "Developer".

Form 1 Template.docx:

Insert today's date
Dear insert client's name:
We would like to inform you your package has been approved.

Form 2 Template.docx:

This package will be delivered to:
insert client name
insert client address
USA

Forms ~~autofilled~~ with client Jane Doe's information

A screenshot of Microsoft Word showing the same two documents as above, but with their content filled in.

Jane Doe Form 1.docx:

08/19/2022
Dear Jane Doe:
We would like to inform you your package has been approved.

Jane Doe Form 2.docx:

This package will be delivered to:
Jane Doe
123 Street
USA

Jotform support replied with the following:

"Yes, that can be achieved in Jotform. However, the work can be simpler. First, you can have all the Products in just one form and have each product being shown depending on the selection of the user. Please take a look at this guide:

- [How to Show or Hide Fields Based on User's Answer](https://www.jotform.com/help/316-how-to-show-or-hide-fields-base-on-users-answer/)
(<https://www.jotform.com/help/316-how-to-show-or-hide-fields-base-on-users-answer/>)

For today's date, we have a Date Picker element that you can set with today's date. Please take a look at this guide:

- [Quick Overview of Form Elements](https://www.jotform.com/help/46-quick-overview-of-form-fields/)
(<https://www.jotform.com/help/46-quick-overview-of-form-fields/>)

Regarding the excel file, we do have a [Spreadsheet to Form widget](https://www.jotform.com/widgets/spreadsheet-to-form)
(<https://www.jotform.com/widgets/spreadsheet-to-form>) that can execute that perfectly. Please take a look at this guide:

- [How to Use the Spreadsheet to Form Widget](https://www.jotform.com/help/442-how-to-use-the-spreadsheet-to-form-widget/)
(<https://www.jotform.com/help/442-how-to-use-the-spreadsheet-to-form-widget/>)

Additionally, here is a [Test Form](https://form.jotform.com/222309215558051) (<https://form.jotform.com/222309215558051>), the code to have Jane Doe's information pop up is 1001. You can [clone](#) it in your Jotform account."

After exploring Jotform, I concluded that it does not meet all functional requirements.

Formstack

Information about Formstack's "Documents" product: <https://www.formstack.com/products/documents>

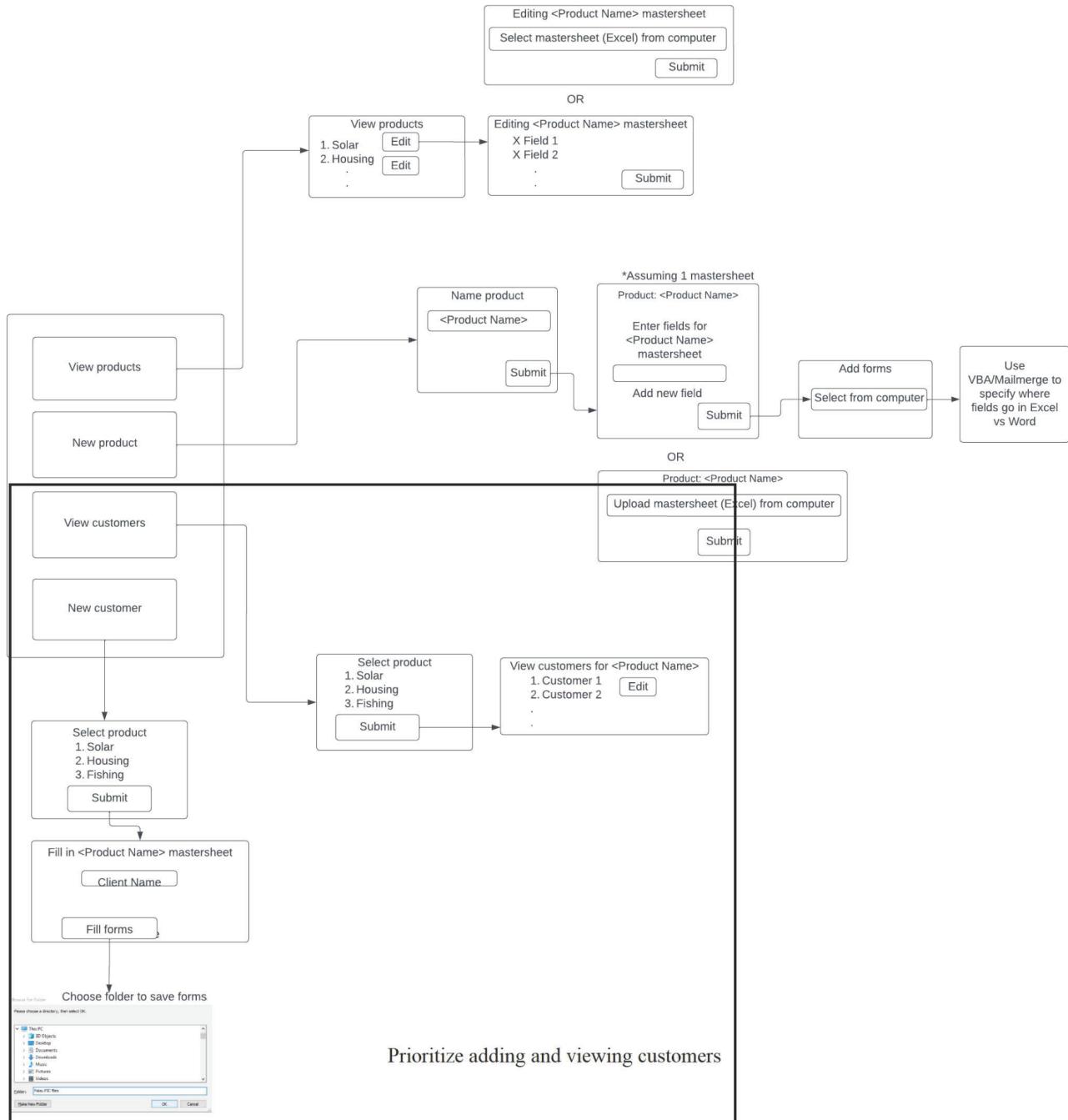
Information about Formstack's pricing: <https://www.formstack.com/pricing/documents>

Formstack seems to be able to accomplish all the functional requirements for document autocompletion. The biggest drawback with Formstack is that the cheapest pricing tier starts at \$92/month and there are limits on how many documents can be autofilled, the number of users, and number of loan products that can be added per pricing tier.

Final Recommendation

Building a custom VBA solution and Formstack are solutions that meet all functional requirements. Because of Formstack's pricing structure and various usage limitations, it has been decided to build a custom VBA solution, which is free and has no usage limitations. However, because a custom application can be difficult to set up and maintain, if NDBP refrains from using the application in the long term due to these difficulties, I recommend NDBP to consider using Formstack.

Appendix B: Loan Product Documents Auto-Completion Workflow



The bank has various loan products (i.e. solar, housing, fishing) and each of them has their own set of documents that the bank staff has to fill out. If a custom application could be built to auto-complete those documents for any loan product, this figure represents the workflow that would be most intuitive to bank employees.

Appendix C: Training Materials

How to use the autofill application:

https://drive.google.com/file/d/1w42URZdpnoIjDACCByEZPooF8Qz_G6Sm/view?usp=sharing

How to use the autofill application written manual:

https://docs.google.com/document/d/1Owrw8YC89dGi5bBEPzmSSFkKi_djaOkO/edit?usp=sharing&oid=117855649691426213179&rtpof=true&sd=true

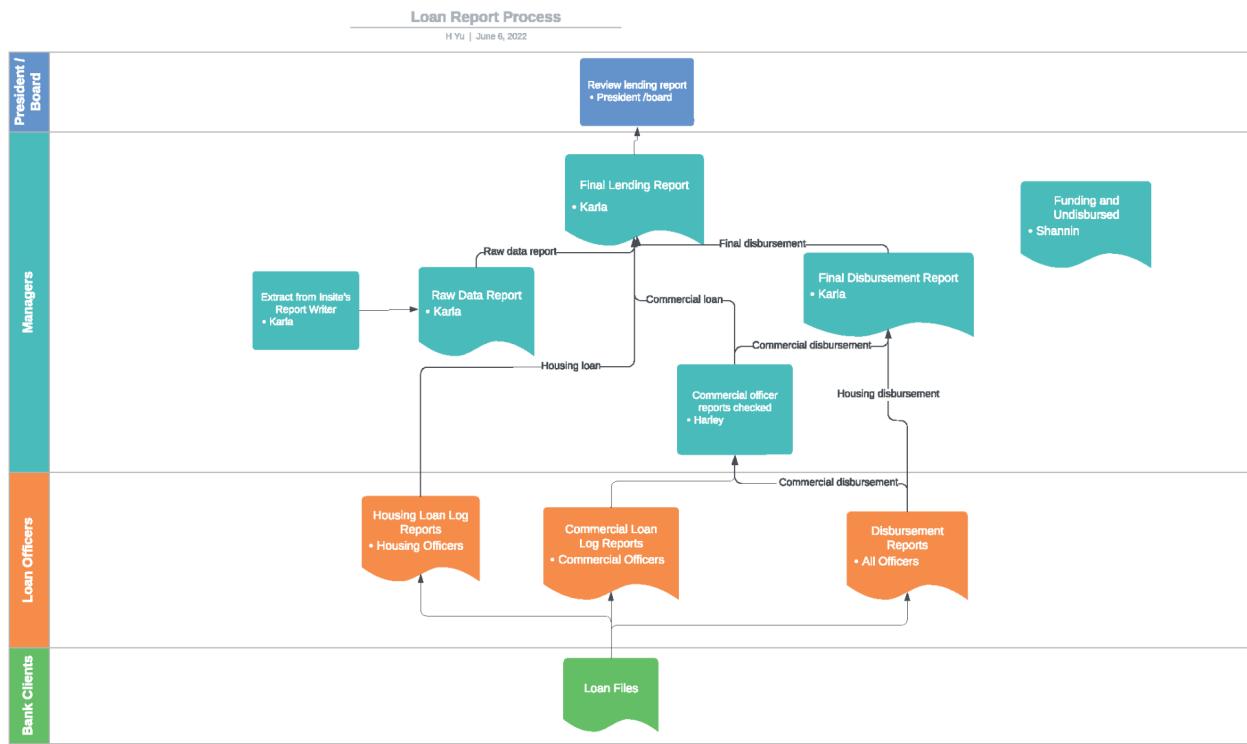
How to add new fields to a Word document:

<https://drive.google.com/file/d/1kuuVZg03c8EuFIPWL1zTIqf6lXv1iSoM/view?usp=sharing>

How to add new loan products to the application:

<https://drive.google.com/file/d/1UHO9ZypZL8e3Y04ZpYSN3HJ1uI8jRwE0/view?usp=sharing>

Appendix D: Lending Report Flowchart



Four hierarchies of people are shown: the bank clients, loan officers, managers, and the president/board of directors. The diagram represents the documents each hierarchy fills out in which the information is used to make the final lending report. The bullet points indicate specifically which individual or group within that hierarchy completes that document.

Appendix E: Lending Report Database System Solutions Analysis

I have identified 2 possible solutions to automating the generation of monthly lending reports:

1. Excel database with R script

Excel spreadsheets would still be used to store client data. A script developed in a data analysis language (i.e. R), would automatically generate the components needed for the final lending report, such as graphs and tables.

2. MySQL database with R script

Migrate NDBP's database from Excel spreadsheets to MySQL. Same with the first solution, a script would generate the lending report components.

Similar solutions were also implemented by TCinGC consultants Mehak Sikka¹ and Leknath Gunathilake² when they worked with FIC in summer 2018 and 2021 respectively. Because FIC had similar goals to automate reports, it is worthwhile to connect with FIC to evaluate whether this is a solution NDBP should implement. Although FIC went with a SQLite database due to implementation issues with the MySQL database, SQLite is not a fit for NDBP because SQLite does not support multiple users. This is described further in the first functional requirements table below.

Solutions Analysis

The solutions above are analyzed according to the following functional requirements:

The first set of functional requirements involve data entry and management.

1. The system must allow loan officers to make an entry for different types of reports (i.e. the housing loan log reports, commercial loan log reports, disbursement reports, and funding and undisbursed shown in Appendix D).
2. The system must allow managers to specify what type of information the loan officers would need to provide for an entry into the system. For example, if a loan officer needs to add an entry to a loan log report, they would need to provide customer name, type code, approved \$, etc.
3. The system must allow loan officers to view, edit, or delete previously entered entries.
4. The system must allow different levels of organizing report entries. For example, entries in a loan log report may be further separated into categories of approved/declined/closed loans, pipeline, accounts to be reviewed and prospects.
5. The system must allow entries to be moved to other categories. For example, an entry under the pipeline category in a loan log report could be moved to approved/declined/closed loans.
6. The system must allow loan officers to view entries submitted by other loan officers. For example, the commercial loan logs are checked by the commercial loans manager before being passed to Karla (Appendix D).

	Excel with R script	MySQL Database with R script
Data entry	YES	YES
Specify entry fields	YES	YES
Edit previous entries	YES	DO NOT KNOW

¹ Mehak Sikka's report: [Status Report 1 \(wixstatic.com\)](https://wixstatic.com)

² Leknath Gunathilake's report: [Final Report_PalauFIC_Leknath.docx \(cmu.edu\)](https://cmu.edu)

Entry organization	YES	YES ³
Recategorize entries	YES	DO NOT KNOW
Multiple people accessing entries	YES	YES ⁴

The second set of functional requirements involve data usage. Loan officers may use the data stored in the system to conduct analysis or generate reports.

1. The system must allow loan officers to filter entries within reports (i.e. by type code, by month, etc).
2. The system must allow Karla to generate KPI graphs (pie charts, line charts, bar graphs) and tables from entries in the system.
3. The system must allow Karla to get the report from a single (or few) action(s), like pressing a “Generate Report” button.
4. The system must allow Karla to easily customize what she includes in the lending report.

	Excel with R script	MySQL Database with R script
Entry filtering	YES	YES
Graph and table generation	YES	YES
Minimal actions to get report	YES	DO NOT KNOW
Report customization	NO ¹	DO NOT KNOW

The following are reasons for a NO:

1. If an R script were used to automate the reports, this would make it difficult for loan officers to customize their reports themselves because that would mean they would need to have some knowledge on R.

Solutions Discussion

The following is a discussion on the pros and cons of each solution:

Excel database with R script

Pros

³ Page 1 of Mehak Sikka's 2018 FIC report stated that MySQL was used to store data in different tables based on type of balances: [Status Report 1 \(wixstatic.com\)](https://wixstatic.com)

⁴ This comparison between MySQL and SQLite stated that MySQL can support multiple users, while SQLite does not: [MySQL vs SQLite | Top 14 Differences You Should Learn \(educba.com\)](https://www.educba.com/mysql-vs-sqlite/)

1. Satisfies all but one functional requirement.
2. Loan officers are already familiar with using Excel spreadsheets to store data.

Cons

1. No intuitive way for Karla to customize the lending report herself without R knowledge.
2. No standardization of data entry. If the loan officers were to reformat their Excel sheets in any way, it can cause a script to not work because of hard coded connections between Excel and the script.

MySQL database with R script

Pros

1. Possibly a standardized way of data entry, making it easy for Karla to customize the lending reports without R knowledge.
2. Do not have to maintain so many Excel spreadsheets.

Cons

1. Possible challenges in implementation. Lakanth's FIC report stated that the reason the FIC did not use the MySQL database developed by Mehak was because of implementation issues. However, the nature of these issues are not specified in his report.
2. Getting loan officers to use a system they are not familiar with.

Final Recommendations

To prepare for the next student consultant, I recommend Karla document in detail the steps she takes to generate the lending report at the same time she is doing the report.

The next student consultant should determine how often Karla changes the format of her lending report or the data she uses for the reports. The largest risk of keeping Excel as a database and why it is not a clear advantageous solution despite meeting most functional requirements is because of the lack of ability to customize reports without having programming knowledge.

The consultant should also find answers to the “Do not know” sections in the above tables regarding the MySQL solution.

If the MySQL solution fails to meet any of the data entry and management functional requirements, then this solution should be discarded because being able to edit and recategorize entries is important for NDBP to track where their clients are in the loan approval pipeline. If the MySQL solution meets those requirements, then I recommend further investigation on whether MySQL presents a clear advantage in terms of minimizing action from Karla to make the report and ability to customize reports. If there is a clear advantage, I recommend moving NDBP’s database from Excel to MySQL. If not, I recommend keeping the Excel database and building an automation script. Efforts must also be made to standardize data entry and prevent small changes in Excel formatting to break the script.

Appendix F: Creating a Provisioning Report

This document details the steps Karla takes to create the monthly provisioning report from Insite’s Report Writer:

- Shift the first row left so that columns have the appropriate name.

A	B	C	D
+	JOAN	NAME	unsecured
1663	DAN	NAME	unsecure TERM

- 'PROJECT Type' are in one cell. Instead, 'PROJECT' should be in 1 cell and 'Type' should be in another cell because they represent 2 different columns.

Y	Z
PROJECT Type	

- Insert a column called "Month end" between "NEXTDUE" and "REVEIW". Populate that column for all rows with the last day of the current month.

NEXTDUE	Month end	REVEIW
7/23/2022	6/30/2022	0/00/0000
6/20/2022	6/30/2022	5/08/2008
6/18/2022	6/30/2022	0/00/0000
11/24/2022	6/30/2022	0/00/0000
4/10/2022	6/30/2022	0/00/0000
7/30/2022	6/30/2022	0/00/0000
9/30/2022	6/30/2022	0/00/0000
5/30/2022	6/30/2022	0/00/0000
3/30/2022	6/30/2022	0/00/0000

- Add a new column called "Past Due" that takes the number of days between the "NEXTDUE" and "Month end" column. Because the result shows up as a date, go to "Home", "Number" tab, and click the apostrophe symbol to change the format to "Accounting" so that the "Past Due" column shows a numerical value.

P	Q	R	S
NEXTDUE	Month end	Past Due	REVEIW
7/23/2022	6/30/2022	23.00	0/00/0000
6/20/2022	6/30/2022	(14.00)	5/08/2008
6/18/2022	6/30/2022	(12.00)	0/00/0000
11/24/2022	6/30/2022	147.00	0/00/0000
4/10/2022	6/30/2022	(81.00)	0/00/0000
7/30/2022	6/30/2022	30.00	0/00/0000
9/30/2022	6/30/2022	92.00	0/00/0000
5/30/2022	6/30/2022	(31.00)	0/00/0000
3/30/2022	6/30/2022	(92.00)	0/00/0000

The screenshot also shows the Excel ribbon at the top and the Home tab selected. In the Number group of the ribbon, the Accounting button is highlighted with a red box. The formula bar shows the formula =DAYS(P17,Q17).

5. Go back to Insite and figure out why some values in “NEXTDUE” are 0. This may be caused by the loan start agreement and non-matching type codes. Ideally, every entry in the NEXTDUE column should have no zero values.

P	Q	R
NEXTDUE	Month end	Past Due
7/23/2022	6/30/2022	23.00
6/20/2022	6/30/2022	(10.00)
6/18/2022	6/30/2022	(12.00)
11/24/2022	6/30/2022	147.00
4/10/2022	6/30/2022	(81.00)
7/30/2022	6/30/2022	30.00
9/30/2022	6/30/2022	92.00
5/30/2022	6/30/2022	(31.00)
3/30/2022	6/30/2022	(92.00)
9/30/2022	6/30/2022	92.00
7/01/2022	6/30/2022	1.00
9/30/2022	6/30/2022	92.00
1/30/2022	6/30/2022	(151.00)
6/30/2022	6/30/2022	-
7/05/2022	6/30/2022	5.00
0/0/0000	6/30/2022	#VALUE!

6. Add 2 columns “Prov %” (provisioning percentage) and “Prov \$” (provisioning amount) in between “Past Due” and “REVIEW” columns. The provisioning percentage is calculated based on the “Past Due” days (i.e. if Past Due is greater or equal than a certain number, then provisioning percent is 100%, if less than a certain number, it is 75%, etc.). There should be a feature that allows NDBP to submit what number of days correspond to what provisioning percentage, and have the percentages automatically calculated for them based on “Past Due.”

Past Due	Prov %	Prov \$	REVIEW
(396.00)	100%	0/0/0000	
(396.00)	100%	0/0/0000	
(396.00)	100%	0/0/0000	
(365.00)	100%	0/0/0000	
(350.00)	100%	0/0/0000	
(319.00)	100%	0/0/0000	
(314.00)	100%	0/0/0000	
(309.00)	100%	10/26/2010	
(304.00)	100%	0/0/0000	
(304.00)	100%	0/0/0000	
(302.00)	100%	3/29/2009	
(273.00)	75%	0/0/0000	
(273.00)	5%	0/0/0000	
(243.00)	5%	0/0/0000	
(212.00)	5%	0/0/0000	
(212.00)	5%	0/0/0000	
(182.00)	5%	0/0/0000	
(182.00)	5%	0/0/0000	
(181.00)	5%	0/0/0000	
(169.00)	5%	0/0/0000	
(166.00)	5%	0/0/0000	
(151.00)	5%	0/0/0000	
(151.00)	5%	0/0/0000	
(151.00)	5%	0/0/0000	
(151.00)	5%	1/18/2018	

7. The “Prov \$” is calculated by taking the product of the “PRINCIPAL” and the “Prov %”.

	M	N	O	P	Q	R	S	T	U	V	W
1	PRINCIPAL	INTEREST	Interest Amount	P&I	Pay-off Amount	NEXTDUE	Date Formatted	Month end	Days Deliq	Prov	Prov \$
2	1	0								100%	-
3	0	18114.01								100%	18,114.01

8. The sum of “Prov \$” is then calculated. NDBP may also want the sum of other columns. If there is an Insite feature that allows NDBP to specify what columns they want a sum for and automatically pull a report that contains those sums, it would be helpful.

Prov \$
251.15
703.32
733.39
739.81
488.61
747.96
521.67
560.78
6,060.03
445.79
111.85
576.07
509.80
750.00
-
716.56
750.00
750.00
-
7,475,082.26