

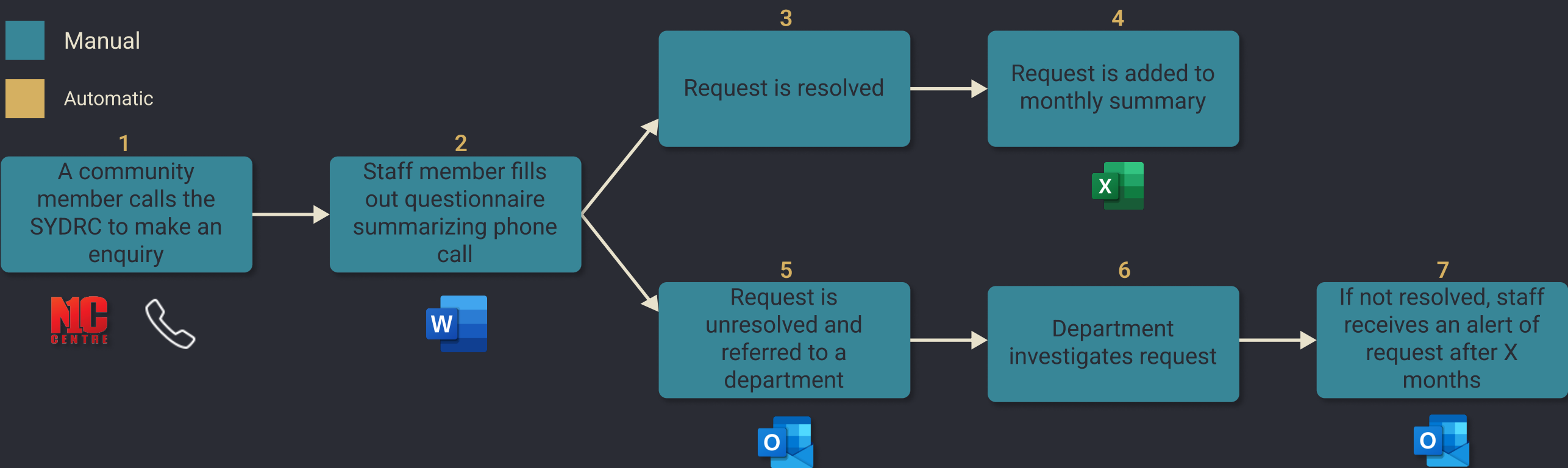
# Case Work

The case work department works with members of the community that request help from the SYDRC. Community members call the SYDRC with questions and requests for help pertaining to housing, education, crime, and more. The service is available to anyone in the community, and each week the SYDRC receives about 9-10 casework enquiries.

We interviewed the department head about the process of creating, editing, and completing a case. First, a community member calls the SYDRC asking for help. The staff member that takes the call writes down all the pertinent information in a Word document. Some requests can be resolved immediately (for example, the staff might provide a government hotline or forward a form where the person can apply for assistance); in this case the enquiry is complete, and no further action is needed. For each case, the department needs to track who made the initial contact, the nature of the request, the staff member(s) involved with the request, and any subsequent follow up actions. Oftentimes an enquiry cannot be immediately resolved. In this scenario, the enquiry becomes a case work, which can go unresolved for several weeks or even months as it may need to be referred to a different staff member or organization that can better help the community member. The referred staff member must create and maintain a Word document to report case information. Information saved in the document includes the client information, previous interactions with them, the category of the enquiry, and how the client might be able to help themselves in the future. The case document may be revisited in the future if another case of a similar nature comes up.

Figure 11 outlines the workflow we learned about prior to our recommendations. Through our analysis of *Views* as well as these interviews and observations of the staff using *Views*, we identified two main areas of improvement: the staff were splitting tasks between different applications and doing many tasks manually. As seen in step 2 of Figure 11, information from the initial phone enquiry was entered in a Microsoft Word document. Although Microsoft Word is easy to work with, it offers no statistical capabilities for summarizing enquiries. If the caseworker was unable to resolve the request over the phone, they would need to manually send a referral email to a colleague (step 5), who would make a personal note to remind themselves of cases they need to work on. If the information needed to be transferred into *Views*, the information had to be copied over from the Word document, taking time away from other tasks. With no established place to store the information efficiently inside *Views*, this task was decentralized as can be seen in step 4 of Figure 11. In addition, since cases may remain open for long periods, it is important to have reminders when a case has been open and untouched for extended periods, but staff had no way to automate reminders or alerts (step 7), leaving room for human error.

**Figure 11**  
Case work’s previous workflow



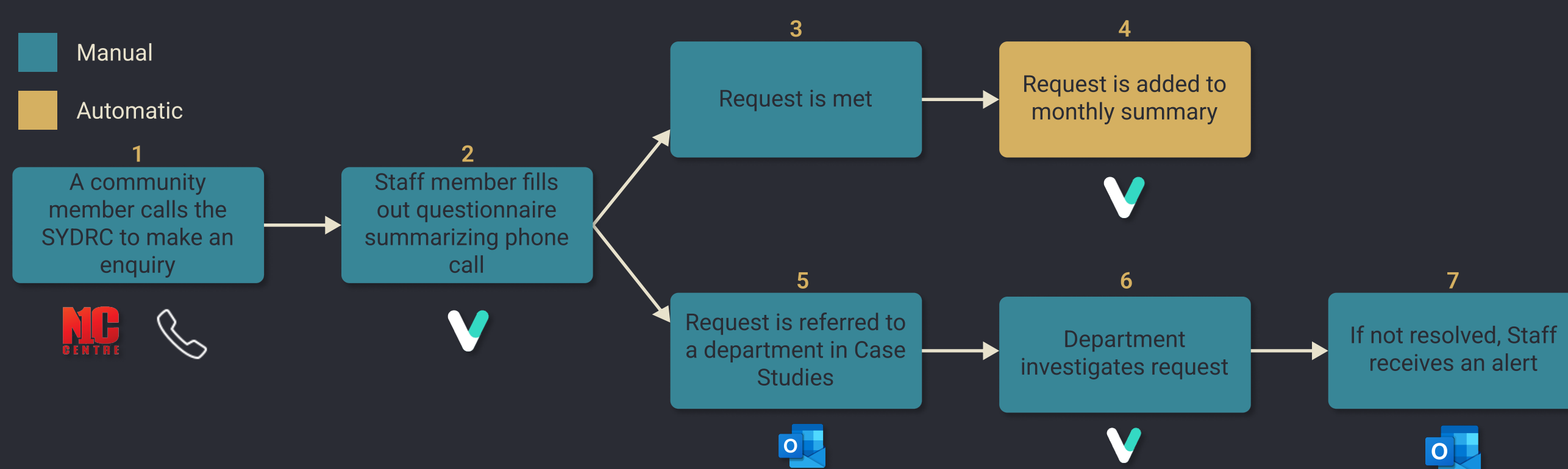
After mapping out this workflow and its shortcomings, we identified two key goals: to centralize all data entry into *Substance Views* and to provide callers with the tools to solve such problems by themselves in the future. Based on these goals, we determined the following design requirements:

- 1 Use *Views* instead of Word to collect, store, and retrieve client information and descriptions of the enquiry. Centralizing the data will improve speed and decrease error in the handling of cases.
- 2 Sort cases by status (open or closed) to help the staff easily identify unresolved cases.
- 3 Alert staff about unresolved cases after a certain period to help the staff prioritize their work and ensure that cases are not overlooked.

After implementing changes in *Views* to address these problems, we held ad hoc interview and observation sessions with the case work staff to show them the changes and gather their feedback. Using this feedback, we made further adjustments and repeated the interview process. Part of this process involved our role-playing as a community member seeking help from the SYDRC to better understand the workflow of the case study department. Half of our team observed the workflow of the department while one member was dedicated to role-playing as a community member. From this we were able to create a new workflow as seen in Figure 12.

**Figure 12**

New workflow for case work department



We moved enquiry questionnaires and data summaries to be entirely in *Views*, reducing redundancy, centralizing the information, and saving the staff the time it takes to handle operating with different applications like Microsoft Word and Outlook at the same time. Instead of using Word to take down the enquiry information, staff members will now use an enquiry questionnaire within *Views* (step 2). Since Word offers no statistical capabilities, the staff used to manually summarize the qualitative data. However, questionnaire data can be analyzed and reported through the statistics function in *Views*, allowing the department to easily track their progress. As mentioned before, if the enquiry cannot be immediately resolved, it turns into a case work. The SYDRC will use questionnaires to keep track of all their case works (step 6). In this workflow, the staff member filling out the enquiry form enters the name of the staff being referred to in the referral question. Then, they have to send an email to the person the case is referred to (step 5).

This person may be inside or outside the organization. The staff member can then log into *Views* and look at the case. If the enquiry is referred to someone outside of the SYDRC, the case work will be downloaded as a pdf manually sent to the organization and the case will be considered closed. This system will allow the SYDRC to better track their long-term progress on cases worked on by multiple parties. Although not shown in Figure 12, client feedback data will be gathered using automated mailed-out Questionnaires in *Views*. The client will be able to fill out the feedback form on their own and the SYDRC will receive the data straight into *Views*. Statistical overviews can easily be created from this Questionnaire data so the department can assess their ability to not only help clients, but also teach themselves what they need to improve on in the future. Questionnaires will centralize the data for the case work department and make measuring their impact easier to quantify.



# MPower

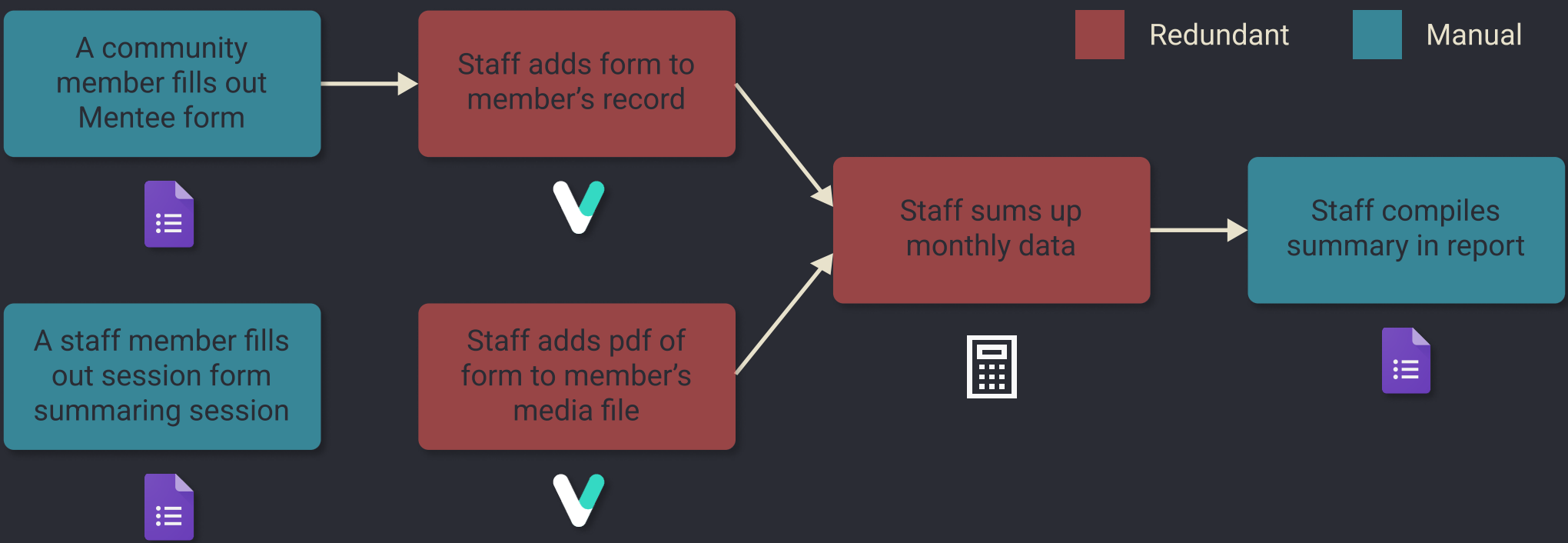
The MPower department, a branch of the youth services department, works with youth to provide them mentoring resources to help them improve academically. The SYDRC offers weekly youth mentoring sessions (individual and group) at schools throughout Camden. The sessions are organized and delivered by SYDRC staff.

The department uses a mentee survey and MPower session report form to track mentee progress and evaluate the success of sessions. Student participants fill out the mentee survey, which asks them to rate (on a scale from 1 (Never) to 5 (Always)) their well-being in a number of areas. For example, one question asks a mentee if at school, is there an adult who really cares about them while another asks them about their general outlook on their lives. The SYDRC prioritizes evaluating the responses of a mentee over time as it allows them to track their impact on mentees. For example, if a mentee answers 1 in after their first mentoring session, but then responds 3 and 4 after a couple weeks of sessions, the SYDRC has data showing their progress. Contrary to the mentee survey, the session report form summarizes the session, with specifics on individual mentees where applicable. Combined, these forms allow the SYDRC to monitor

their lessons and get a summary of how the session went for the whole group or individual mentee.

Figure 13 outlines the original workflow for staff members creating monthly session reports based on session report summaries and mentee feedback. Both the session report summaries and mentee feedback forms were created using Google Forms. Since the data inside Forms is external to *Views*, staff would ordinarily download form data as PDF documents, and upload them to the *Views* session and participant notes sections (respectively). By using PDF documents in this fashion, the data cannot be viewed within *Views*. To see the data, the user must download the PDF document and open it in an external program. Additionally, notes cannot be analyzed statistically, so staff were forced to compile monthly reports by hand into Google Docs, further decentralizing the data.

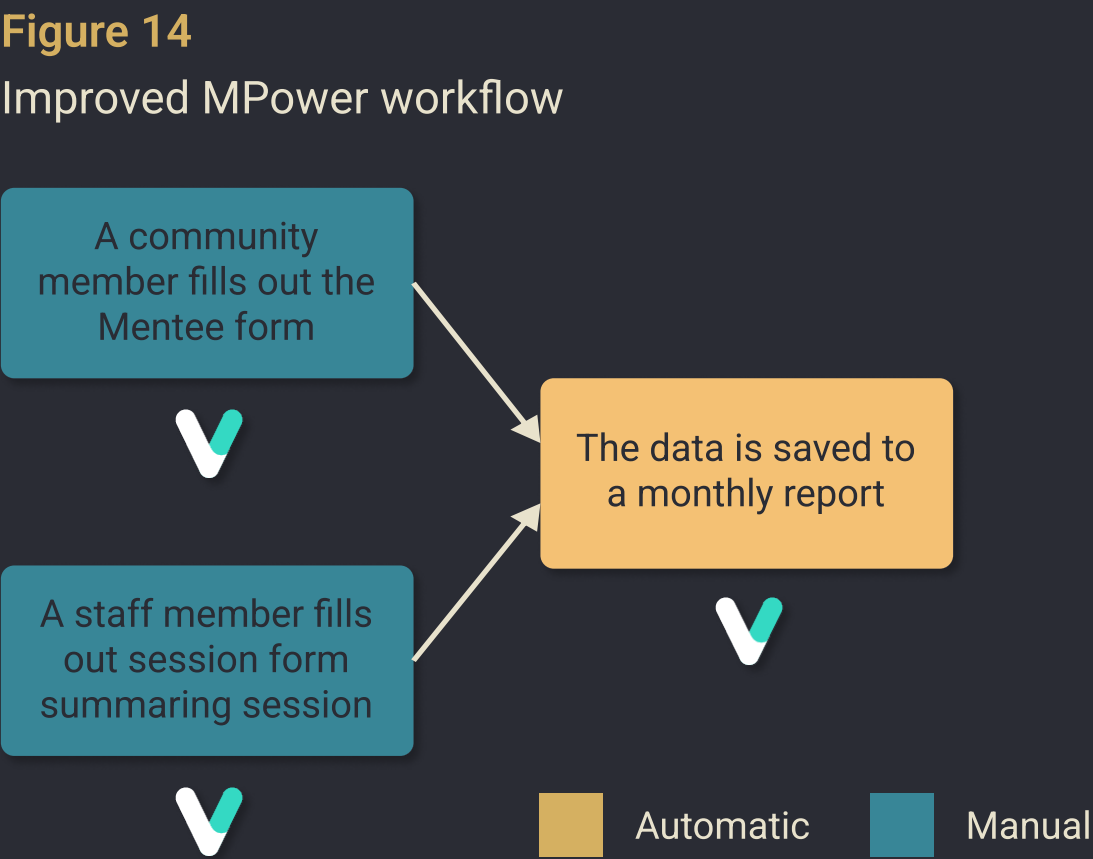
**Figure 13**  
Original MPower workflow



Through our process of working with MPower, we determined that they could better make use of *Views* in multiple ways. By storing the mentee survey as well as the session reports in questionnaires in *Views*, the data would be centrally accessible to all team members. Additionally, this would allow staff to generate summary statistics to better track progress of the youth members. Thus, we established two concrete design requirements:

- 1 Convert mentee final survey and session report to *Views* questionnaires so that the data is centrally stored and may be statistically analyzed more easily within *Views*.
- 2 Generate sample reports to highlight *Views*' statistical capability, to assist the staff members in creating reports and track progress of individuals and groups, faster.

As in the case work department, we implemented these changes using an iterative process. Figure 14 details the final workflow following our changes.



In sum, Google Forms were moved into *Views* questionnaires. The mentee questionnaire can be linked to participants and session reports can be linked to sessions. This allows staff to create summaries based on all responses for a session or participant, which was not possible when they uploaded their PDF reports to the notes section. Since all questionnaire data is stored in *Views*, staff can automatically create monthly summaries of questionnaire responses. To facilitate creation of these statistics, similarly to how we taught the case work department, we gave the department members homework to create statistics on their own. By doing this, the staff members both learned how to better use the software, and they gave feedback on the documentation which allowed us to iteratively improve it.



# Youth Services

The youth services department runs more general activities for members of the community such as youth football matches. The outcomes of these activities are important to the department because it can show progress of their clients. They report the progress of their clients to organizations that provide funding as proof that they are making an impact.

We interviewed the department head on the processes he completes to record all the work done in the department. This department encompasses many participants, activities, and sessions. The department must track what youth are registered for each event, who attends the event, the description of the event, as well as both the youth and staff feedback on the event. All this data is also aggregated every quarter into a report into Excel. The SYDRC keeps track of outcomes from clients, like improved self-esteem or improved cooking skills. They have a defined list of dozens of possible outcomes. These outcomes can occur after any number of sessions and can occur multiple times. The lead staff determine when participants have achieved a given outcome. This data is also kept track of in Excel. The SYDRC uses outcome data to demonstrate program impact on the client.

After interviewing staff in the department, we came to the conclusion that moving their current workflow to *Views* would be inefficient and cause more work for staff. *Views* does not offer an automatic way to summarize outcomes, therefore shifting their data collection to *Views* does not offer many advantages. Although their main workflow was not shifted to the *Views* framework, there were three areas where we improved their ability to collect and visualize data in *Views*:

- 1** Track activity attendance for each session.
- 2** Use questionnaires to gather feedback information from youth about sessions
- 3** Create statistics of the aforementioned data.

The youth services department will now use the session attendance feature in *Views* to consistently track attendance data. This is superior to Excel because *Views* can effectively track attendances across different sessions, while Excel only shows the total number of attendances per participant. Staff can also create the attendance statistic internal to *Views* where all the information is held. This information can automatically update to show attendance over the rolling periods of time. Furthermore, the data can be exported to Excel to display data to other organizations. Also, using a questionnaire to track feedback from participants allows more data to be centralized into *Views* and easy creation of statistics and summaries within the software. These two functionalities allow the department to then create statistics of this data in *Views* collected from the two processes.

# Administration

The administration department manages and oversees all SYDRC operations. They create monthly reports summarizing programs offered and measuring the impacts. They also manage all of the SYDRCs venue bookings.

To create the monthly reports, staff compile statistics and graphs created using data submitted from each of the other departments. The administration department emails out a Microsoft Form that the staff fills out outlining what they did in the past month, such as the number of sessions they all held and the number of participants at each session. However, since some departments were using Microsoft Word and other applications to track this, analyzing their data to fill out the form could prove tedious and time consuming as they lacked a central place to store and query their data. Although *Views* was a central place for all data to be stored, it was not being used effectively by each of the departments.

After learning what the administration department does, our team identified three goals for administration workflow: Creating statistics within *Views* that can be used in monthly reports, displaying statistics that need frequent viewing on a dashboard, and creating a new and improved booking system. This led us to create the following design requirements:

- 1 Develop a guide for creating a statistic so that the administration staff are comfortable creating statistics on their own
- 2 Implement the created statistics in requirement 1 into an administration dashboard to allow the staff to obtain an overview of the progress the SYDRC has made.
- 3 Update the booking system with an emphasis on creating the most robust and automated system as possible.

We added statistics that will be useful to the organization and taught members of the administration department how to create statistics. Now that all organizational information will be put into *Views* instead of external documents, the statistics will automatically update each month. Furthermore, the administration department will need to ask the other departments less often for information because more information will already be in *Views*; they are able to access it themselves, saving a lot of time for the other departments. Unfortunately, *Views* is not powerful enough to analyze written responses and understand data trends, but it can display numerical, quantitative data effectively, like number of case works completed and participation data.

The statistics (new) functionality of *Views* is currently being rolled out but with the caveat that the feature is missing a lot of functionalities. One of the missing features is the ability to add statistics (new) to a dashboard. Although we cannot add the new statistics to the dashboard at the time of this project, the functionality is on the runway, and we created documentation to walk them through adding them to the dashboard. Once they can add statistics to the dashboard, they will be able to follow our documentation to do it.

## Booking System

The N1C Centre, the building that the SYDRC occupies, contains several rooms that can be booked by community members for a fee. In 2019 prior to the COVID-19 pandemic, the SYDRC earned close to £30,000 from bookings. Anybody in the community may book a room as long as the SYDRC approves it, which is a decision they make on a case-by-case basis. For example, if users damage the room or fail to clean up adequately, their next booking may be denied. Additionally, there are some clients who have recurring bookings that are pre-approved by the SYDRC.

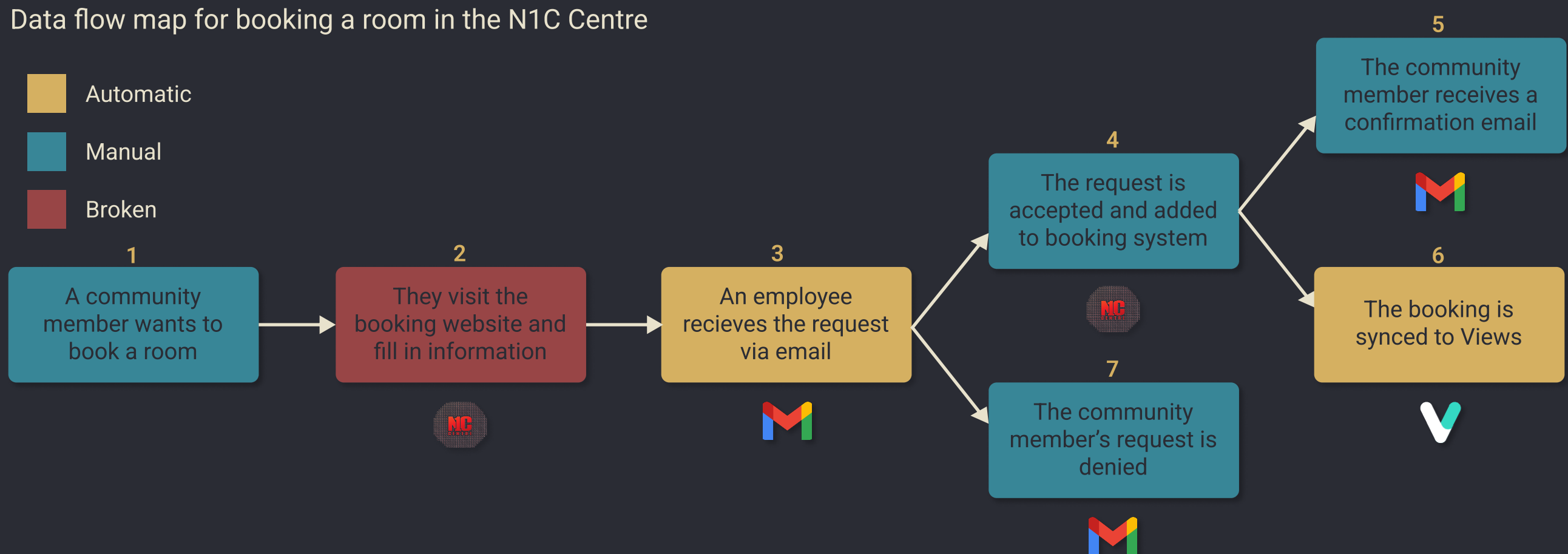
Originally, booking requests were made through phone calls or emails, which was slow and error prone. To improve the situation, a 2020 WPI student group created a booking application from scratch using the NodeJS framework, hosted on Amazon Web Services (AWS).



The application workflow as shown in Figure 15 allowed community members to request rooms online (step 1), and the application then automatically forwarded an email request to the SYDRC (step 3). Staff members then had to sign into the booking system to approve or deny that booking request, as seen in steps 4 and 7, which is reflected in the *Views* calendar. This tight integration with *Views* meant that the SYDRC did not need to handle each booking through direct email or phone conversation with the client, and the reservation was automatically populated into *Views* where it may be analyzed (Aponte et al., 2020). Unfortunately, at the start of this project, the booking system was no longer working, shown in red in step 2. Even if it were, it still required a lot of manual input from staff members as shown in steps 4 and 7. For example, to confirm a booking, a staff member had to manually copy the booking request information back into the booking system from the request email and send a confirmation email to the community member. This step could be simplified to a yes or no button inside the request email which automatically creates the booking in *Views* and sends a confirmation. There may also be other modifications that could improve the efficiency and effectiveness of the booking system.

**Figure 15**

Data flow map for booking a room in the N1C Centre



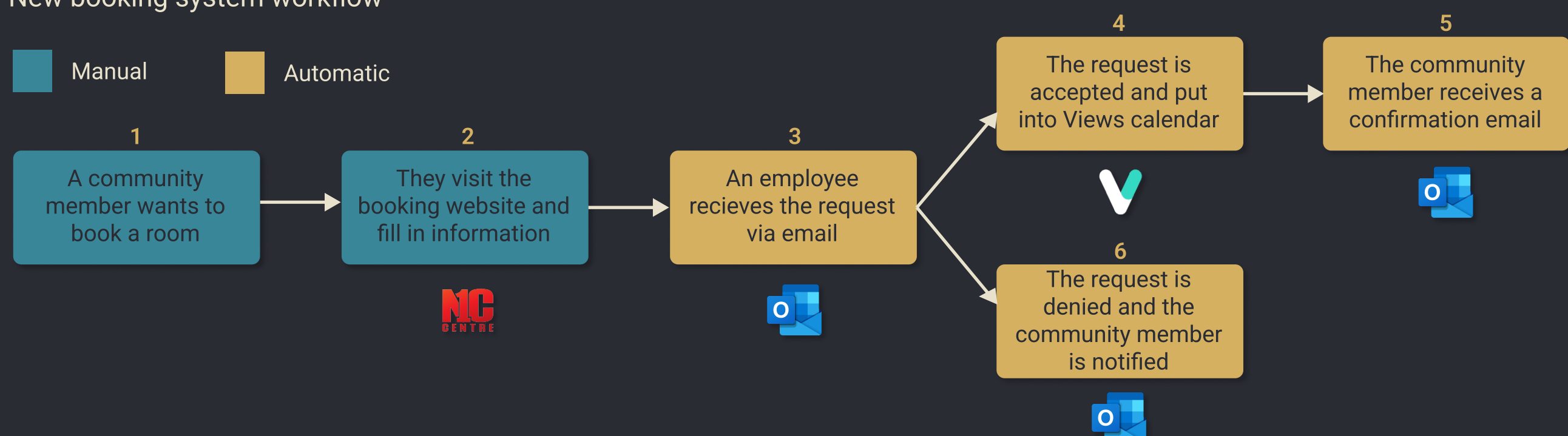
Through background research and conversation with the administration department, we determined that there were two general categories for improvement with regards to the booking system: Robustness and Automation. With these goals in mind, we established two concrete design requirements for our iteration of the booking system:

- 1 Rework the backend to be more reliable so that the system does not start working for unpredictable reasons.
- 2 Automate acceptance and denial of the requests to save time and create more consistent interactions.

We implemented this updated workflow, and then presented it to the administration department, gathering feedback and iteratively updating it to better fit the needs of the department. The updated workflow for the booking system can be seen in Figure 16.

**Figure 16**

New booking system workflow






The changes to the booking system resulted in a robust system with a simple interface for both the staff and clients. The application was implemented in the TypeScript programming language using Cloudflare Workers on the backend and React/Cloudflare Pages on the frontend. The client view to request a new booking can be seen in Figure 17. When submitting the form, the SYDRC will be notified that there is a new booking request (step 3), and the client will receive a confirmation email to confirm that the request was received.

**Figure 17**

Booking request form

### Book a room



Main Hall

Booking date \*

Start \*

End \*

April 25, 2022

-- : --

-- : --

Recurring \*

None

Full Name \*

Organization

Phone Number \*

+44 (000)-000-0000

Email \*

your@email.com

Description \*

Describe what you will be using the room for

Birthday party for 10 year old

Required Supplies \*

e.g. tables, chairs, projector, flipchart board, room layout

When a booking is confirmed, you are liable to the [Terms and Conditions](#)

Submit

Cloudflare Workers is the source of reliability in the system. Workers is an implementation of serverless computing, which comes with significant benefits for reliability.


Because the app is only being executed while a request is in flight, a fatal error will only stop that single process and when the next request arrives a new instance will be started. As a result, bugs within the code do not cause catastrophic failure. The last group's system failed due to the *Views* account credentials being changed. As a result, the system failed to contact *Views* and shut down without a simple way for the SYDRC to restart it. In this case, the new system would continue to serve requests, but simply not input the requests into *Views* with a relevant error message. To fix the issue, the SYDRC would simply need to input the new credentials into the Cloudflare dashboard. While robustness was the primary focus of the changes in the booking system, automation has been addressed too.

In contrast to the old system where a staff member was required to manually correspond with the client, the new booking system handles sending confirmation and denial emails automatically (steps 4 and 6). Additionally, instead of the staff member copying the booking information into the system directly, they simply click a link within the request email and then press accept or deny on the booking system website (shown in Figure 18). When denying a request, the staff member may provide a reason which is forwarded to the prospective client. One other benefit of the new booking system is that the SYDRC is able to make small edits to the content of the emails being sent through a web editor. Overall, these changes have created a system which is easier to use for the SYDRC while also increasing the robustness of the system overall.

**Figure 18**

Booking confirmation within the booking system

### Confirmation



ON 2022-04-20 FROM 09:00 TO 11:00

BY Matt Worzala (WPI)

CONTACT mworzala@wpi.edu OR +1000112222

DESCRIPTION I will use the room to demo the booking system capabilities.

Decline

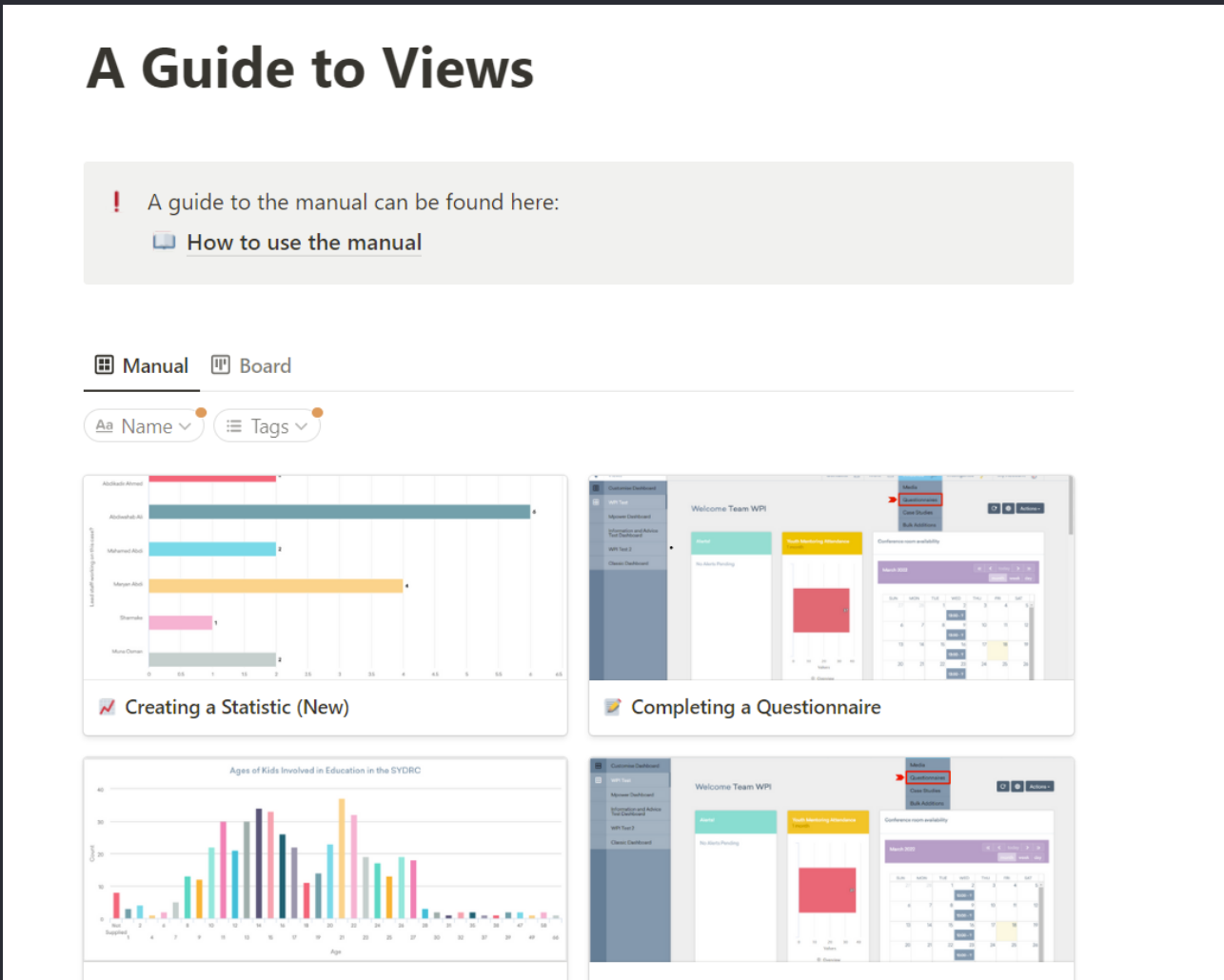
Accept



# Deliverables

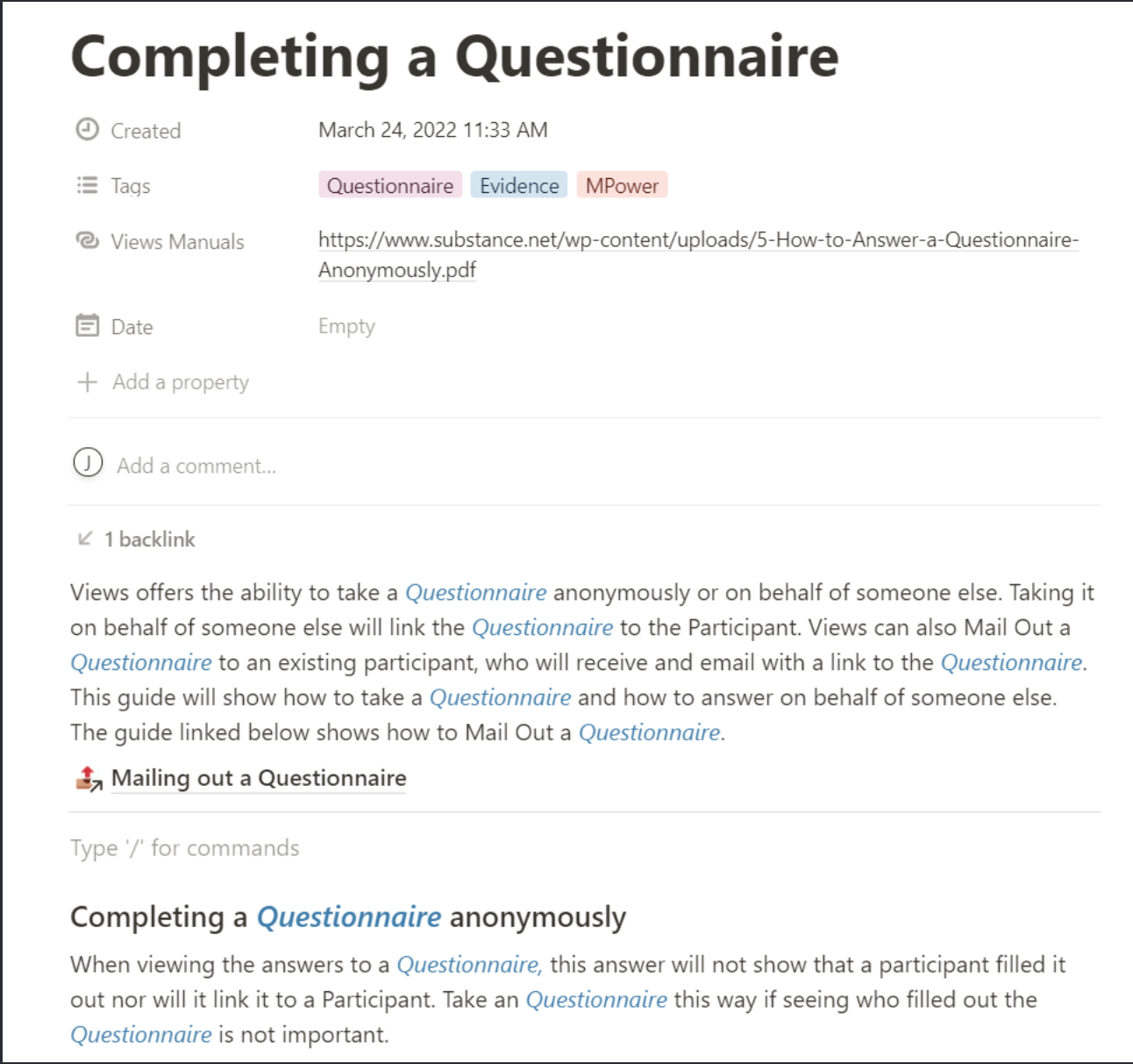
We created two deliverables: (1) a searchable user manual and (2) a new room booking system. The user manual outlines how to do processes important to the SYDRC in *Views*. We created the manual in a software called Notion. Notion is highly customizable software used for project management, databases, and wikis. The first page is an overview of the various topics/procedures the user may want to learn. We chose a gallery template to display a sneak-peek into each page, seen in Figure 19, where each page describes a distinct process within *Views*. Users can click on a page to enter and view the guide for the step by step process. We also included a page at the top of the manual on how to use the user manual called “How to use the manual”. It explains how to navigate through the manual to find the desired information.

**Figure 19**  
The front page of the user manual



Every page follows a similar format; there is a short paragraph at the top that outlines the process, when a user should use the process, and what the result of the process looks like. Some pages have an embedded video that ‘walks’ the user through the steps in the page. Underneath, the page describes each step the user must take in order to complete a task. Every step is outlined with a visual with the aid of arrows and boxes to highlight where to click. Figure 20 displays one step in the user manual.

**Figure 20**  
Example page of the user manual



We designed the user manual iteratively throughout the term. First, we created a page on each topic or procedure, then we gave the staff homework and told them to use our guide to complete the relevant procedure outlined in the manual. We then asked them if they were successful and whether they got stuck on any particular steps we outlined. We then made improvements based on their feedback, if necessary, and improved our explanations in the pages. We did this for a handful of key processes that many staff will complete, including staff with less *Views* experience, so it was very important that our explanations were concise.

The SYDRC staff can access the Notion page in a web browser by simply clicking a link provided by us. We recommend the staff bookmark the guide to their browser, so they do not need to keep track of the link. The QR code below will bring the reader to the manual.



# Authorship

Our team had a wide variety of skills that allowed our project to run smoothly. The entire team worked together to write and edit all sections of the report; there was no primary writer or editor. Matthew Dzwil researched background information of the SYDRC, worked on the methods and results section of the report, as well as created user guides and aided with the creation of the booking system. James conducted general research on CRMs, researched the functionality of *Views* by looking into the documentation provided by *Views* along with experimenting inside of the *Views*’ software, and worked on creating guides for the user manual and recording videos for staff. Bradley researched CRM implementation in non-profits, worked inside of *Views* to understand its functionalities, and created guides and videos in the user manual for staff. Matt Worzala researched why the original booking system failed and created the new booking system along with its user guide. He also was the primary designer for the booklet.



Matthew Dzwil  
Data Science



James Hammel  
Mechanical Engineering



Matthew Worzala  
Computer Science



Bradley Markiewicz  
Mechanical Engineering



# References

- About us – N1C Centre – Community Centre. N1C Centre Community Centre. (2022). Retrieved January 18, 2022, from <https://www.n1ccentre.org/about-us/>
- Aponte, G., Gonzalez Villalobos, D., Mara, P., Bonaventura, A., & Reiser, T. (2020). Developing a Digital Management System for the SYDRC. : Worcester Polytechnic Institute.
- Bull, C. (2003). Strategic issues in customer relationship management (CRM) implementation. *Business Process Management Journal*, 9(5), 592–602. <https://doi.org/10.1108/14637150310496703>
- CRM: Best Practices to get the most ROI out of your CRM. (n.d.). Salesforce.com. Retrieved February 14, 2022, from <https://www.salesforce.com/crm/best-practices/>
- Department for Communities and Local Government. (2009). The Somali Muslim community in England understanding Muslim ethnic communities. Retrieved January 18, 2022, from <http://karin-ha.org.uk/wp-content/uploads/2013/02/1210847.pdf>
- Grattan, K. E. (2012). CRM implementation in nonprofits: An analysis of the success factors (3504476). (1011657248). <http://ezproxy.wpi.edu/login?url=https://www.proquest.com/dissertations-theses/crm-implementation-nonprofits-analysis-success/docview/1011657248/se-2?accountid=29120>
- History – Somali Youth Development and Resource Centre- Somali community in Camden, Somali Youth in Camden, Somali in London. Somali Youth Development and Resource Centre Somali Community in Camden Somali Youth in Camden Somali in london. (n.d.). Retrieved January 17, 2022, from <https://www.sydr.org/history/>
- Janzen, J. H.A. (2021, March 10). Somalia. Encyclopedia Britannica. <https://www.britannica.com/place/Somalia>
- Kristoffersen, L., & Singh, S. (2015). Successful application of a customer relationship management program in a nonprofit organization. *Journal of Marketing Theory and Practice*, 12(2), 28–42. <https://doi.org/10.1080/10696679.2004.11658517>
- Nonprofit Impact Measurement Practices that Actually Work. (2020, June 16). Prosper Strategies. <https://prosper-strategies.com/nonprofit-impact-measurement/>
- Open Society Foundations. (2014). Somalis in London. Open Society Foundations. <https://www.opensocietyfoundations.org/uploads/cfbed57a-9d85-454f-9e8d-d65b3aeaf1b9/somalis-london-20141010.pdf>
- Payne, A., & Frow, P. (2005). A Strategic Framework for Customer Relationship Management. *Journal of Marketing*, 69(4), 167–176.
- Plan International Norway. Plan International Norge. (2021, January 27). Retrieved February 25, 2022, from <https://www.plan-norge.no/english>
- Schroeder. (2020). Use Microlearning to Train Staff. *The Major Gifts Report*, 22(8), 4–4. <https://doi.org/10.1002/mgr.31529>
- Somali Youth Development Resource Centre - Charity 1092646. Register of Charities. (n.d.). Retrieved January 17, 2022, from <https://register-of-charities.charitycommission.gov.uk/charity-search/-/charity-details/3981672/financial-history>
- Somali Youth Development Resource Centre. (2021, March 9). SYDRC Trustees' Report. Retrieved January 24, 2022, from [https://register-of-charities.charitycommission.gov.uk/charity-search?p\\_p\\_id=uk\\_gov\\_ccew\\_onereg\\_charitydetails\\_web\\_portlet\\_CharityDetailsPortlet&p\\_p\\_lifecycle=2&p\\_p\\_resource\\_id=%2Faccounts-resource&p\\_p\\_cacheability=cacheLevelPage&\\_uk\\_gov\\_ccew\\_onereg\\_charitydetails\\_web\\_portlet\\_CharityDetailsPortlet\\_objectiveId=A10854488&\\_uk\\_gov\\_ccew\\_onereg\\_charitydetails\\_web\\_portlet\\_CharityDetailsPortlet\\_priv\\_r\\_p\\_organisationNumber=3981672](https://register-of-charities.charitycommission.gov.uk/charity-search?p_p_id=uk_gov_ccew_onereg_charitydetails_web_portlet_CharityDetailsPortlet&p_p_lifecycle=2&p_p_resource_id=%2Faccounts-resource&p_p_cacheability=cacheLevelPage&_uk_gov_ccew_onereg_charitydetails_web_portlet_CharityDetailsPortlet_objectiveId=A10854488&_uk_gov_ccew_onereg_charitydetails_web_portlet_CharityDetailsPortlet_priv_r_p_organisationNumber=3981672)

Somali Youth Development Resource Centre. (2021, March 9). SYDRC Trustees' Report.

Retrieved January 24, 2022, from <https://register-of-charities.charitycommission.gov.uk/charity-search?>

[p\\_p\\_id=uk\\_gov\\_ccew\\_onereg\\_charitydetails\\_web\\_portlet\\_CharityDetailsPortlet&p\\_p\\_lifecycle=2&p\\_p\\_resource\\_id=%2Faccounts-resource&p\\_p\\_cacheability=cacheLevelPage&\\_uk\\_gov\\_ccew\\_onereg\\_charitydetails\\_web\\_portlet\\_CharityDetailsPortlet\\_objectiveId=A10854488&&\\_uk\\_gov\\_ccew\\_onereg\\_charitydetails\\_web\\_portlet\\_CharityDetailsPortlet\\_priv\\_r\\_p\\_organisationNumber=3981672](https://register-of-charities.charitycommission.gov.uk/charity-search?p_p_id=uk_gov_ccew_onereg_charitydetails_web_portlet_CharityDetailsPortlet&p_p_lifecycle=2&p_p_resource_id=%2Faccounts-resource&p_p_cacheability=cacheLevelPage&_uk_gov_ccew_onereg_charitydetails_web_portlet_CharityDetailsPortlet_objectiveId=A10854488&&_uk_gov_ccew_onereg_charitydetails_web_portlet_CharityDetailsPortlet_priv_r_p_organisationNumber=3981672)

Substance. (2022). Views. Retrieved February 1, 2022, from <https://www.viewsapp.net/>

Szymczuk, M. (December 12, 2019). 5 Types of CRM Software: What's Right For Your Business?

Retrieved February 14, 2022, from <https://softwarehut.com/blog/business/5-types-of-crm-software>