Software Engineering
2023; 11(1): 6-14
http://www.sciencepublishinggroup.com/j/se
doi: 10.11648/j.se.20231001.12
ISSN: 2376-8029 (Print); ISSN: 2376-8037 (Online)

Revolutionizing Field Data Collection and Management Using PowerApps and Robotics Process Automation (RPA)

Sai Madhur Potturu

Robotics Center of Excellence (CoE) Zoetis Inc, Parsippany, United States

Email address:
saimadhurpotturu@gmail.com

To cite this article:
Sai Madhur Potturu. Revolutionizing Field Data Collection and Management Using PowerApps and Robotics Process Automation (RPA). *Software Engineering*. Vol. 11, No. 1, 2023, pp. 6-14. doi: 10.11648/j.se.20231001.12

Received: June 7, 2023; **Accepted**: June 27, 2023; **Published**: July 6, 2023

Abstract: This paper introduces a digital solution that leverages Microsoft PowerApps and Robotic Process Automation (RPA) to transform field data collection and management in rural farms with limited internet connectivity. The purpose of this solution is to address the challenges associated with offline data capture, synchronize the data with a central database, and effectively utilize it for various purposes within the organization. The research methods employed involve the utilization of Microsoft PowerApps and RPA. The PowerApps application enables field force team members to capture and store service order data on mobile/tablet devices offline and synchronize the data with the database when they are online. Additionally, an RPA solution retrieves the data from the database and automatically creates service orders in various systems. The paper discusses the design process of the digital solution, focusing on enabling offline capabilities using PowerApps for both Model-Driven Apps and Canvas Apps. It also explains the synchronization process, where data is saved locally on the device and later synchronized with the database. Furthermore, the integration of PowerApps and RPA is explored, highlighting the advantages of seamless data synchronization and automation capabilities. Simulations are conducted to evaluate the performance of the digital solution, including offline capability, data synchronization, response time, and user experience. The results demonstrate the solution's effectiveness in maintaining data integrity, seamless synchronization, improved response time, and enhanced user experience. In conclusion, the paper summarizes the benefits of the digital solution, including improved service quality, enhanced customer experience, and reduced manual effort. It highlights the game-changing offline capability of the app and its applications in various industries.

Keywords: Rural Farms, Offline Data Capture, PowerApps, Robotic Process Automation (RPA), Increased Efficiency, Service Quality Improvement, Customer Experience Enhancement, Manual Effort Reduction

1. Introduction

In many rural farms, internet connectivity is a challenge [1]. Field force team members face challenges when capturing service order data due to unreliable internet connectivity. As a result, they must write down the order details on paper, resulting in errors and omissions in the data, suboptimal service quality, and customer experience. To overcome these challenges, a PowerApps application was created that enables the field force team to capture and store service order data on their mobile/tablet devices, both online and offline [2, 3, 12]. This application automatically saves data locally when there is no internet connection and synchronizes the data with the database when the device is reconnected to the internet. Additionally, an RPA solution is utilized to retrieve the data from the database and automatically create service orders in various systems [4]. This paper explores the benefits of this digital solution in improving service quality, enhancing the customer experience, and reducing manual effort.

The previous research on offline-capable solutions for collecting data in remote areas had limited information to address various challenges, including data synchronization issues, complex integration with existing systems, a complicated user interface and user experience, limited scalability, and performance, as well as limitations in data security and privacy [16]. However, a digital solution has been developed using Powerapps, and RPA systems offer several advantages. Powerapps, a low-code application development platform, offers seamless data synchronization with multiple databases, easy integration with over 400 connectors [11], the

ability to develop simple and rich user interfaces, and high data security [2, 3]. The integration of Powerapps and RPA solutions also contributes to improved scalability and performance.

2. Solution

The solution discusses different approaches to enable offline capabilities in Microsoft PowerApps. It focuses on two scenarios: offline capabilities for Model-Driven Apps with Dataverse and offline capabilities for Canvas Apps with other databases (such as SharePoint, SQL, Excel, etc.). This plays a crucial role in the solution that enables field force employees to collect data offline on their mobile and tablet devices, thereby improving service quality and customer experience.

The solution also discusses various automation options to utilize the collected data for various automation activities and synchronize it with different systems within the organization. This facilitates seamless operations, reduces manual effort, and improves efficiency.

Offline Capable Microsoft PowerApps [6-8, 12]:

There are three types of Microsoft Power Apps: Canvas Apps, Model-Driven Apps, and Power Portals.

Canvas Apps are designed for internal organizational use and offer flexibility in creating custom app layouts and designs. They can be integrated with both standard and custom connectors (such as Outlook, SharePoint, SAP, etc.).

Model-Driven Apps, also built for internal organizational use, provide predefined layouts and templates based on the type of data added to the app. They can only be integrated with Dataverse [5].

Power Portals are external-facing websites that grant access to the organization's data for external users. These apps can only be integrated with Dataverse.

Offline Capable Model-Driven Microsoft PowerApps with Dataverse: [6] Microsoft has recently introduced a Mobile Offline-First experience with the Model-Driven Apps made with PowerApps where users can now use Model-driven apps seamlessly without the need to be concerned about intermittent or unreliable internet connectivity. However, an internet connection is required to synchronize the data between the mobile device and Microsoft Dataverse.

The Model-Driven Power Apps mobile app always runs offline first, either with or without an internet connection. This optimization significantly enhances offline performance and guarantees a consistent user experience when changing various locations [6].

Online: When offline mode is not configured, the model-driven app works as a regular app that works on a PC or web with an internet connection but doesn't work without the internet. [6]

Offline-first without an internet connection: Data is downloaded and saved in your mobile device as a cache. The changes are automatically saved to Dataverse when the internet connection is restored. [6]

Offline Capable Canvas Microsoft PowerApps with other databases (SharePoint, SQL, Excel, etc.): [2, 7] The canvas apps are more flexible in design and integration with other systems when compared to model-driven apps which can only be integrated with Dataverse and require premium licenses to configure and use data tables on Dataverse.

The canvas apps with 400+ standard connectors [11] (SharePoint, Outlook, Excel, etc..) are free to use with Office E3 and E5 licenses and can empower users to go above and beyond in meeting their specific needs by designing, developing, sharing, and scaling personalized Web/Windows/Mobile applications.

Canvas apps built on Microsoft Dataverse now have offline capability. However, canvas apps created using standard connectors do not have out-of-the-box offline capability [7]. However, we can create offline-capable Canvas PowerApps integrated with a data source/database (SharePoint) with the combination of different functionalities in the Canvas app (Save Data and Load Data).

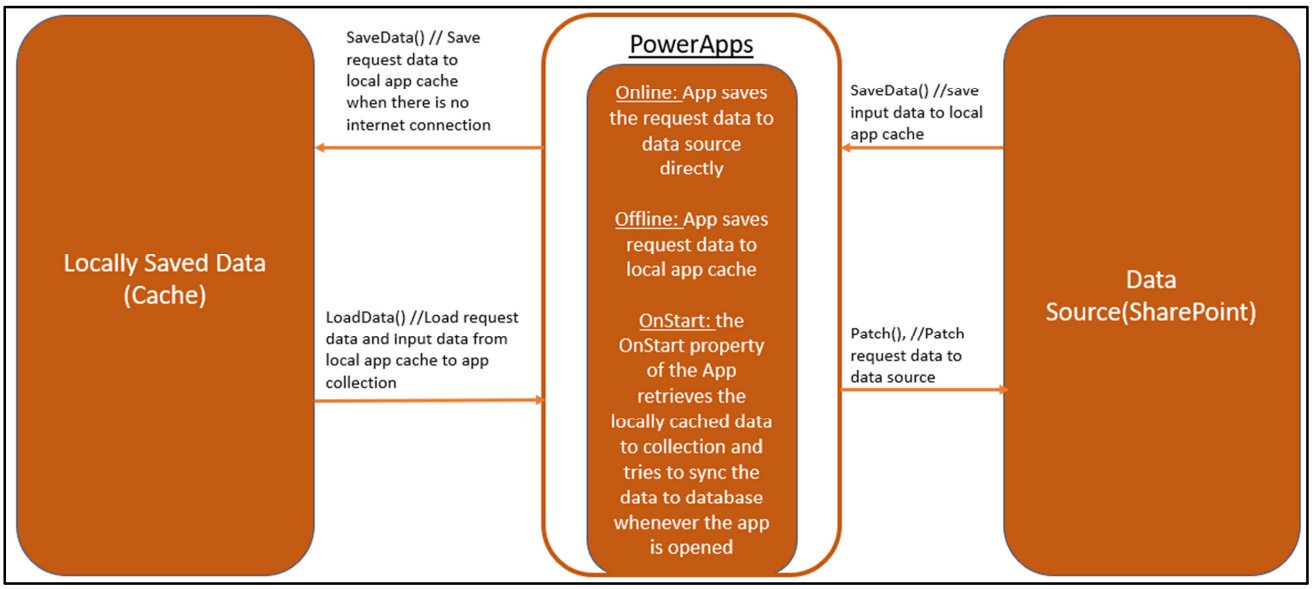

Figure 1. *A schematic representation of the Offline capable PowerApps Canvas App [8].*

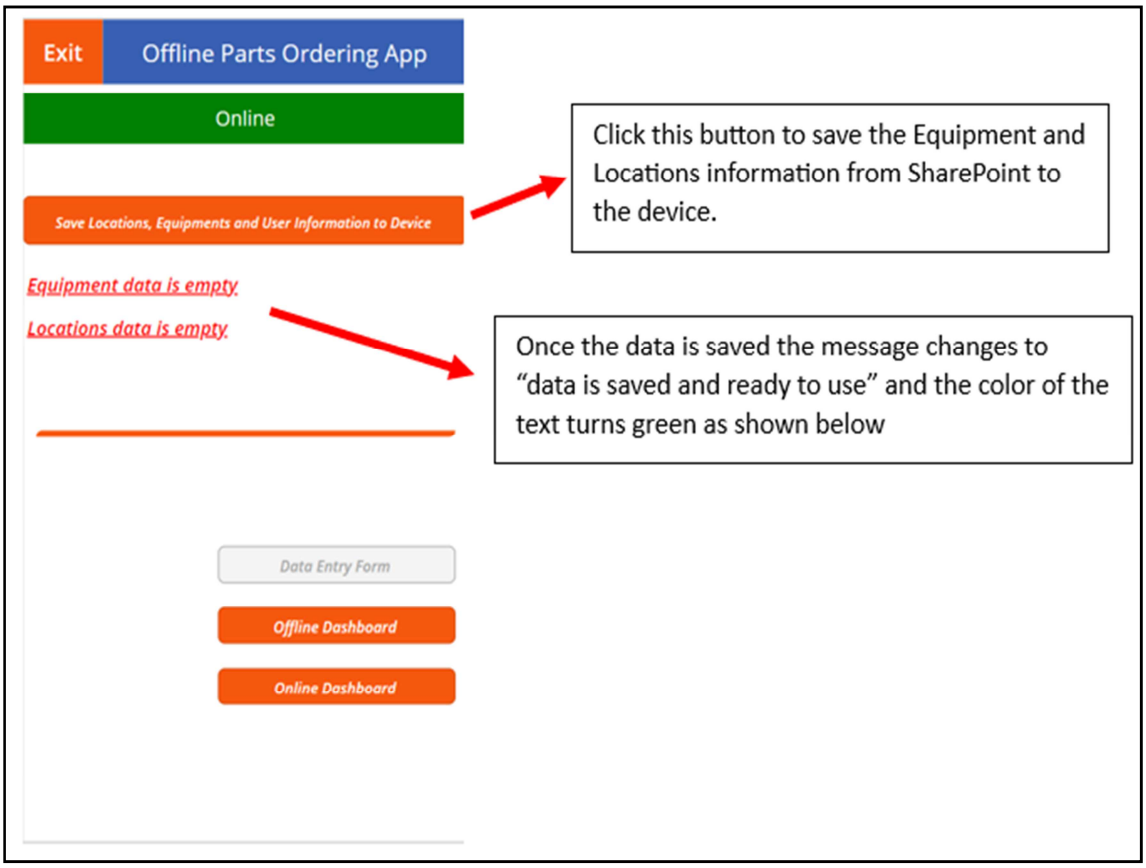

Figure 2. *Save data from SharePoint to the device (App Cache).*

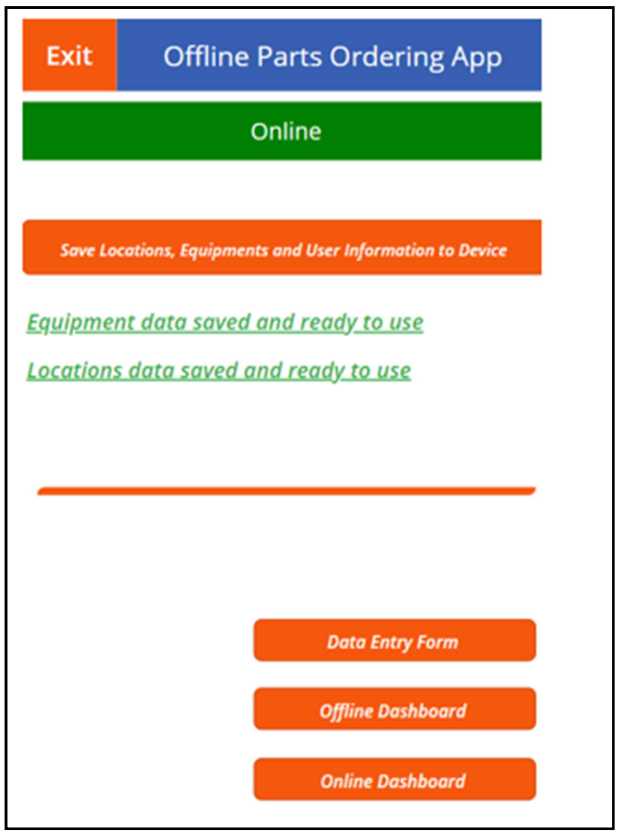

Figure 3. *The text color changes to green and the text message is updated.*

The process of creating an offline-capable canvas app Power App is illustrated using an example of a Hatchery data collection app, specifically designed for collecting data when working in remote locations with limited internet connectivity.

Save Data: [7, 8, 12]

To ensure that the data on the device (such as Mobile, iPad, Tablet, etc.) is constantly up to date and accessible even without an internet connection, the field force employee using the app should download the data from the online platform once daily.

The download button runs the below PowerFX code:

```
Concurrent (
Clear(vColEquipment);SaveData(vColEquipment,"Equipments");
ClearCollect(vColEquipment,HatcheryEquipmentAndParts);
SaveData(vColEquipment,"Equipments"),
Clear(vColLocations);SaveData(vColLocations,"Locations");
ClearCollect(vColLocations,HatcheryLocations);SaveData(vColLocations,"Locations")
);
```

1. The Equipment and Locations data is saved to the app cache "Equipments" and "Locations" in the above code.
2. Concurrent Function: Concurrent function executes the functions/formulas written in this function concurrently.
3. Clear(vColEquipment): The collection has a table structure that holds tabular data. The clear function will clear the collection table.

4. SaveData(vColEquipment,"Equipments"): Saving the empty collection table to the "Equipments" cache to clear the cache. (Clearing out the app cache "Equipments" in the device)
5. ClearCollect(vColEquipment, HatcheryEquipmentAndParts): Save the SharePoint list "HatcheryEquipmentAndParts" to the collection. This will save the updated SharePoint list data to the collection.
6. SaveData(vColEquipment,"Equipments"): Saves the updated collection data to the App cache "Equipments" to the device.

Click the "Data Entry Form" button to start entering the data.

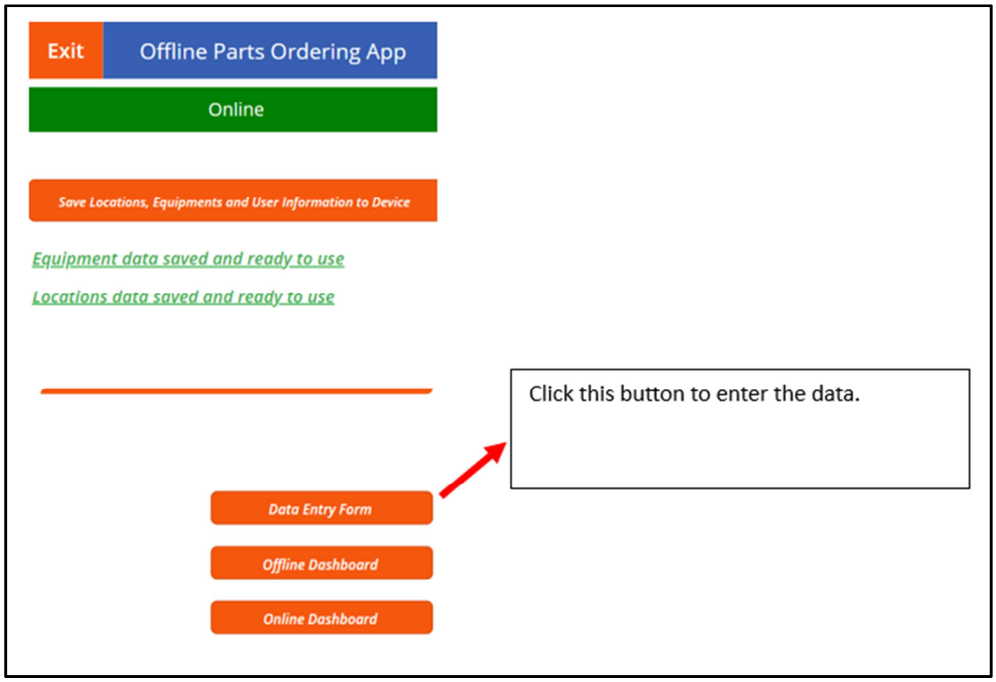

Figure 4. *Data entry form button.*

The data saved earlier can be accessed in the Site Name, Site Location, Equipment, and Part dropdowns.

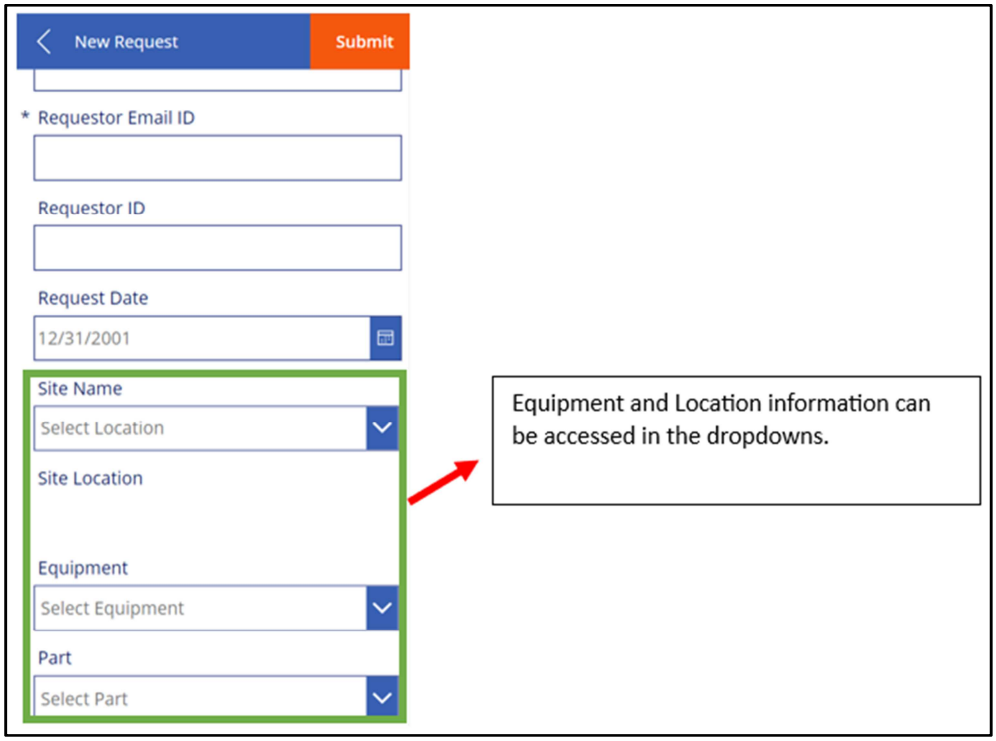

Figure 5. *Data entry form.*

Submit Request and Save Data: [7, 8, 12]

Upon entering the request information, click the Submit button. If an internet connection is available, the app will store the data in the designated data source (SharePoint). However, if there is no internet connection, the app will save the data to the device's app cache under the name "Orders".

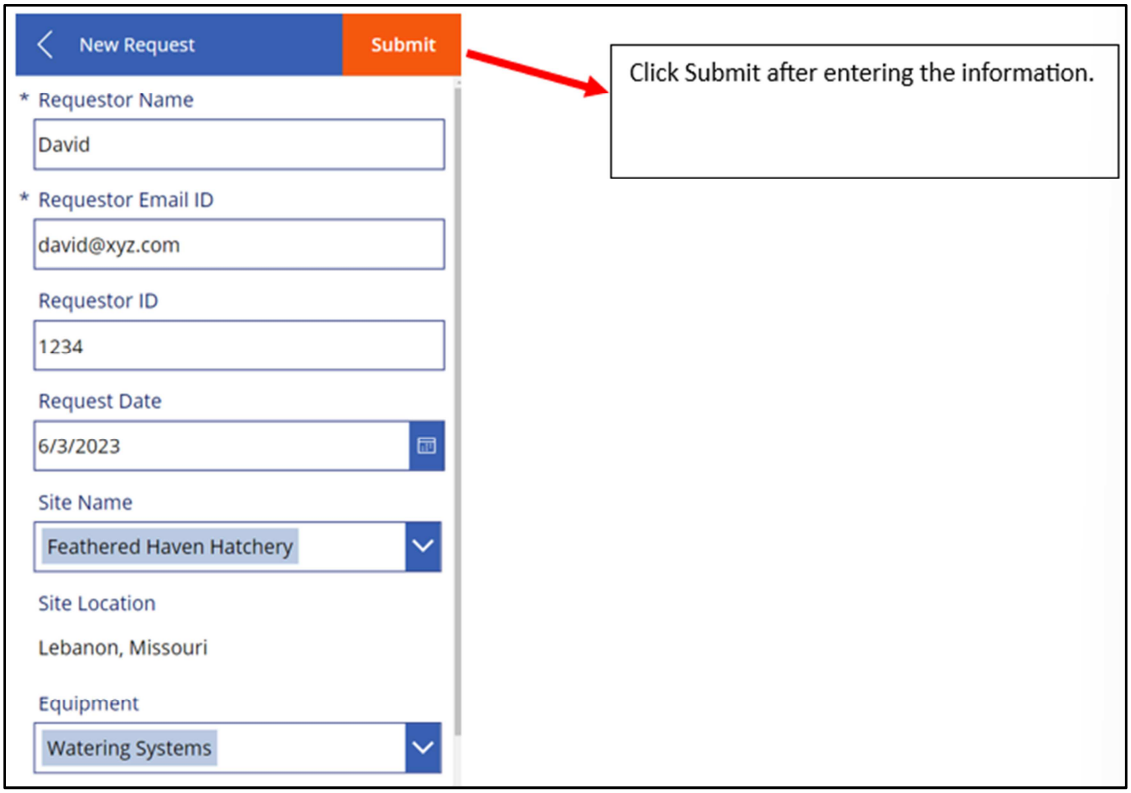

Figure 6. *Enter data and submit.*

Submit button runs the below PowerFX code:

//Clear the collection and load the cache data from the device to the collection. If there is any previously saved data in the device, the Load Data function will retrieve the cache data to the collection.

```
Clear(vColOrders);
    LoadData(
        vColOrders,
        "Orders",
        true
    );
```

//Collect the current Order details and append it to the collection

```
Collect(vColOrders,{Title:DataCardValue11.Text,RequestorEmailID:DataCardValue12.Text,RequestorID:DataCardValue13.Text,RequestDate:DataCardValue14.SelectedDate,SiteLocation:DataCardValue15.Text,SiteName:ComboBox_SiteName.Selected.Value,Equipment:ComboBox_Equipment.Selected.Value,Part:ComboBox_Part.Selected.Part,PostingStatus:"Request Submitted"});
```

```
// Check whether the internet connection is active
If(Connection.Connected,
```

// If true (Connected), Save collection to the data source(SharePoint), clear collection, and clear cache.

```
Collect(OfflineCapablePowerApp,vColOrders);Clear(vColOrders);
SaveData(vColOrders,"Orders"),
```

// If False (Not Connected), Save collection to the device (Cache), clear collection.

```
SaveData(vColOrders,"Orders")
);
```

//Reset the form to delete the previously selected values and show the popup for dashboard navigation.

```
ResetForm(Form2);UpdateContext({ShowDashabordPopup: true})
```

When the device is connected to the Internet, this code checks whether there is any request information saved to the device, retrieves the data to a collection, appends it with the current request information and saves the entire data to the data source, and clears the cache.

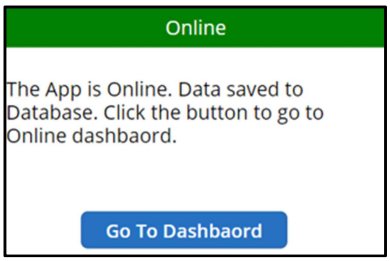

Figure 7. *Online Dashboard popup.*

When the device is connected to the internet, the Online Dashboard window appears. This dashboard retrieves data from the designated data source (SharePoint) and displays it on the dashboard interface.

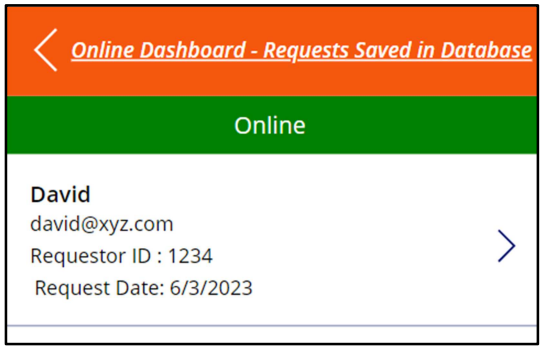

Figure 8. *Online Dashboard.*

In the absence of an internet connection, this code verifies if there is any saved request information on the device. It retrieves the data and adds it to a collection along with the current request information. Finally, it saves the complete dataset to the device's cache.

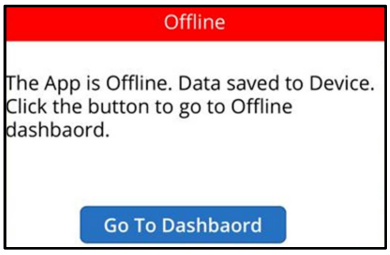

Figure 9. *Offline Dashboard popup.*

When the device is not connected to the internet, an Offline Dashboard window appears. This dashboard retrieves data from the app cache and displays it on the dashboard interface.

Figure 10. *Offline Dashboard.*

Synchronize the data to the data source (SharePoint) [7, 8, 12].

To synchronize the data with the data source, the field force employee should launch the app while connected to the internet. The code written in the "OnStart" property of the app will subsequently verify the presence of an active internet connection and initiate the synchronization process.

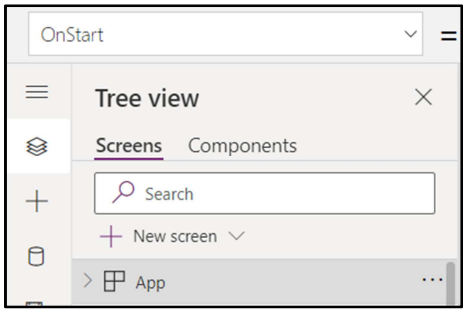

Figure 11. *On Start property of the App.*

OnStart property of the app runs the below PowerFX code:

```
// Clear the collection and save the previously cached request data to the collection.

Clear(vColOrders);
    LoadData(
        vColOrders,
        "Orders",
        true
    );

//Check whether the device is connected to the internet.
If(
    CountRows(vColOrders)>0                    &&
Connection.Connected,

//If True(Connected), the apps saved the retrieved cached data to the data source (SharePoint)
    ForAll(
        vColOrders,
        Patch(
            OfflineCapablePowerApp,
            Defaults(OfflineCapablePowerApp),
            {
                Title: Title,
                RequestorEmailID: RequestorEmailID,
                RequestorID: RequestorID,
                RequestDate: RequestDate,
                SiteName: SiteName,
                SiteLocation: SiteLocation,
                Equipment: Equipment,
                Part: Part,
                PostingStatus: PostingStatus
            }
        )
    );
// Clears the collection and cleared the cache
    Clear(vColOrders);
```

```
        SaveData(
            vColOrders,
            "Orders"
        ),

    // No Action is taken when the device is not connected to the internet. The cached "Order" data will remain saved in the device.
    );

    //Concurrent function executes the functions/formulas written in it concurrently.

    Concurrent(

    // Retrieves the Equipments data from the cache to collection to use it in the app.

        Clear(vColEquipment);
        LoadData(
            vColEquipment,
            "Equipments",
            true
        ),
    // Retrieves the Locations data from the cache to collection to use it in the app.
        Clear(vColLocations);
        LoadData(
            vColLocations,
            "Locations",
            true
        )
    );
```

Integrating Powerapps and RPA (Robotics Process Automation)

Robotic process automation (RPA) is a software technology that makes it easy to build, deploy, and manage software robots that emulate human actions interacting with digital systems and software. Just like people, software robots can do things like understand what's on a screen, complete the right keystrokes, navigate systems, identify, and extract data, and perform a wide range of defined actions. But software robots can do it faster and more consistently than people, without the need to get up and stretch or take a coffee break [4, 10, 13, 14].

The solution developed using PowerApps allows field force team members to capture and store service order data on their mobile/tablet devices, even in areas with poor internet connectivity. Once the team captures the data, it is saved in a database. An RPA (Robotic Process Automation) solution is then used to retrieve the data from the database and create service orders in different systems automatically.

RPA Leaders:

Some of the industry leaders that have established a strong presence in the RPA field are UiPath, Automation Anywhere, Microsoft Power Automate, Blue Prism, etc. [10, 13, 14]

Different system integration with RPA software:

RPA bots can perform actions like data entry, interact with applications' user interface, mimic mouse clicks, and keystrokes, extract data, and validation tasks [9].

1. Web Applications: RPA bots can interact with different web-based applications, such as online forms, e-commerce websites, customer portals, and web-based enterprise systems.
2. Windows Applications: RPA bots can interact with different Windows-based applications, such as desktop software, client applications, and legacy systems.
3. ERP Systems: RPA can interact with Enterprise Resource Planning (ERP) systems, such as SAP, Oracle, Microsoft Dynamics, or any other industry-specific ERP software.
4. Database Systems: RPA can interact with various database systems, such as SQL databases, NoSQL databases, and cloud-based database platforms.
5. Legacy Systems: RPA can interact with legacy systems that may lack modern APIs or direct integration capabilities.
6. CRM Systems: RPA can interact with Customer Relationship Management (CRM) systems, such as Salesforce, Microsoft Dynamics 365 CRM, or other CRM platforms.

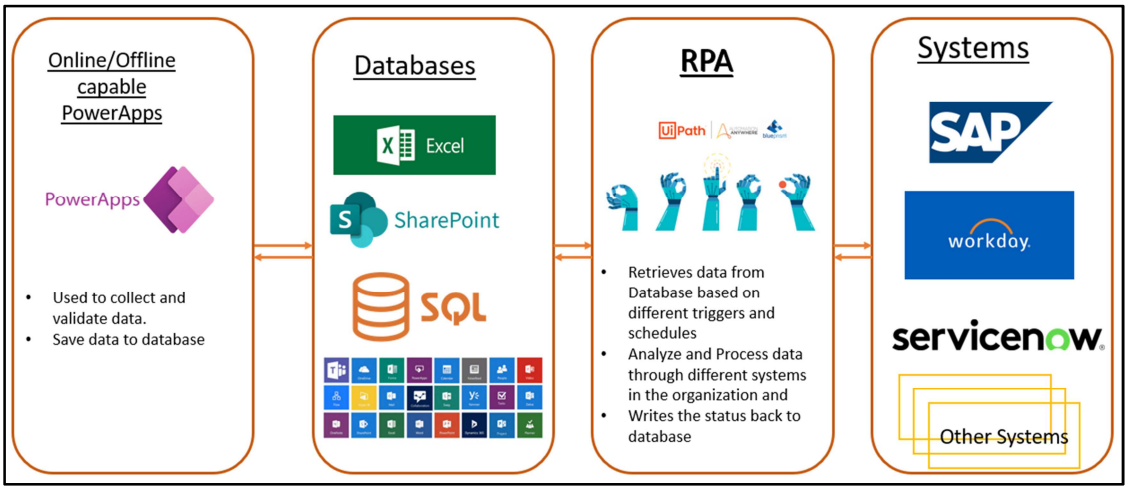

Figure 12. *Architecture diagram.*

3. Simulations to Evaluate the Solution's Performance

Multiple experiments are conducted to test the performance of the digital solution in various aspects such as offline capability, data synchronization, response time, and overall user experience. The observations are as follows.

Offline Capability Simulation:

The Powerapps solution is tested with varying levels of network connectivity. The app can detect the internet connection status, allowing seamless synchronization of data with the database when connected, regardless of the strength of the connection. In cases where the app is not connected to the internet, it seamlessly saves the data to the device cache, effectively preventing any potential data loss.

Data Synchronization Experiment:

The app is intentionally switched between online and offline modes by turning airplane mode on and off on the mobile device.

Airplane mode turned on: After entering the initial request data in the app, airplane mode was enabled on the device, followed by clicking the "Submit Request" button. The code associated with the button detected the absence of internet connectivity and promptly saved the data to the local device cache.

Airplane Mode turned off: The second request data is entered in the app while the airplane mode is still turned on. Following this, airplane mode was disabled, and the "Submit Request" button was clicked. The code associated with the button promptly detected an active internet connection, retrieved the first request data from the device cache, and seamlessly synchronized both the first and second request data to the database.

Response Time Evaluation:

The end-to-end digital solution consisting of Powerapps and RPA was tested rigorously and observed that the solution has significantly reduced the delays in capturing the request information and creating service orders in various systems. The solution has improved the overall response time and efficiency of the process by 30-40%.

User Experience Assessment:

The implementation of the digital solution has resulted in substantial reductions in order processing delays, improved delivery times, prevented data loss, enhanced service quality, and overall, enhanced the user experience. [10]

4. Benefits of the Solution

Improved Service Quality: [10, 15]

The digital solution improves the quality of service by ensuring that all necessary information is captured accurately and in real time. With the digital solution, field force team members can capture data on the spot, reducing the chances of errors and omissions. The RPA solution ensures that the data is retrieved promptly and accurately, enabling the creation of service orders without delay.

Enhanced Customer Experience: [10, 15]

The digital solution improves the overall customer experience by providing faster, more accurate, and more efficient service. With the ability to capture data on mobile devices and work offline, the field force team can deliver a more reliable service, even in areas with poor internet connectivity. The RPA solution ensures that the service orders are created promptly and accurately, reducing delays, and improving the customer experience.

Reduced Manual Effort: [10, 15]

The digital solution significantly reduces manual effort in creating service orders. With the ability to capture data on mobile devices and work offline, field force team members no longer need to manually record data on paper and then re-record it in different systems. The RPA solution retrieves the data automatically, eliminating the need for manual input.

Offline App as a Game Changer:

The offline capability of the app is a game-changer as it allows field force team members to capture data even in areas with poor internet connectivity, providing a more reliable service. The digital solution enables field force teams to deliver a more efficient service, even in remote locations, leading to improved service quality and enhanced customer experience.

Potential Use in Other Industries:

The combination of PowerApps online/offline mobile app with RPA has significant potential for use in other industries beyond animal healthcare. For example, in the agriculture industry, this solution could be used for field data collection and management. Farmers and field workers can use the mobile app to capture data on crops, livestock, and machinery, even in areas with poor internet connectivity, and the RPA solution can automatically retrieve and analyze this data to optimize farming practices and increase efficiency.

In the manufacturing industry, this solution can be used to streamline supply chain management, with field workers using the mobile app to capture data on inventory, production, and shipping. The RPA solution can then retrieve and analyze this data to ensure timely and accurate delivery of products to customers, reducing delays and improving the customer experience.

5. Conclusion

The combination of PowerApps' online/offline mobile app with RPA has tremendous potential for various industries, including pharma, agriculture, manufacturing, construction, and more. The offline-capable PowerApps enable the collection and synchronization of data to the database, while an RPA solution analyzes and processes the data for various purposes within the organization. This solution streamlines field data collection and management, improving service quality, enhancing the customer experience, and reducing manual effort.

References

[1] Strover, S. (2021). Rural internet connectivity. *Telecommunications Policy*, 331-347. https://doi.org/10.1016/S0308-5961(01)00008-8

[2] Leung, T. (2021). Introducing Power Apps. In: Beginning Power Apps. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-6683-0_1

[3] Malvik, S. (2022). Power Apps. In: Mastering Azure API Management. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-8011-9_12

[4] Anagnoste, Sorin. "Robotic Automation Process - The next major revolution in terms of back office operations improvement" Proceedings of the International Conference on Business Excellence, vol. 11, no. 1, 2017, pp. 676-686. https://doi.org/10.1515/picbe-2017-0072

[5] "Microsoft (n. d.). What is Microsoft Dataverse? Microsoft Learn" https://learn.microsoft.com/en-us/power-apps/maker/data-platform/data-platform-introhttps://learn.microsoft.com/en-us/power-apps/mobile/mobile-offline-overview

[6] Microsoft (n. d.). Mobile offline overview (preview). Microsoft Learn. https://learn.microsoft.com/en-us/power-apps/mobile/mobile-offline-overview

[7] Microsoft (n. d.). Develop offline-capable canvas apps. Microsoft Learn. https://learn.microsoft.com/en-us/power-apps/maker/canvas-apps/offline-apps

[8] Wörmer, L. (n. d.). Power Apps: Building Offline Capability for Canvas Apps. Lennartwoermer.de. https://lennartwoermer.de/2020/07/power-apps-offline-capability-for-canvas-apps/

[9] Automation Anywhere (n. d.). Example of entering data into a web form from a worksheet. Docs. Automationanywhere.com. https://docs.automationanywhere.com/bundle/enterprise-v2019/page/enterprise-cloud/topics/aae-client/bot-creator/commands/enter-data-into-webform-from-file.html

[10] Microsoft (n. d.). What you should know about the benefits of RPA. Microsoft Learn. https://powerautomate.microsoft.com/en-au/benefits-of-rpa-robotic-process-automation/

[11] Microsoft (n. d.). List of all Standard tier connectors. Microsoft Learn. https://learn.microsoft.com/en-us/connectors/connector-reference/connector-reference-standard-connectors

[12] Leung, T. (2021). Working Offline. In: Beginning Power Apps. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-6683-0_23

[13] Moreira, S., Mamede, H. S., & Santos, A. (2023). Process automation using RPA – a literature review. *Procedia Computer Science*, *219*, 244-254. https://doi.org/10.1016/j.procs.2023.01.287

[14] UiPath (n.d.). What is robotic process automation? Www.Uipath.com. https://www.uipath.com/rpa/robotic-process-automation

[15] Prakash Pradhan, S. (2022). Introduction: Microsoft Power Apps. In: Power Platform and Dynamics 365 CE for Absolute Beginners. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-8600-5_1

[16] R. K. Lomotey, Y. Chai, S. Jamal and R. Deters, "MobiCrop: Supporting Crop Farmers with a Cloud-Enabled Mobile App," 2013 IEEE 6th International Conference on Service-Oriented Computing and Applications, Koloa, HI, USA, 2013, pp. 182-189, doi: 10.1109/SOCA.2013.19.