



iThink 365
Microsoft 365 Application Development
Service

Published: 29/03/2021

Prepared by
Simon Doy

Microsoft 365 Application Development Service

Overview

The Microsoft 365 Development Service provides our customers with a way to deliver productivity solutions on the Microsoft 365 platform.

Some examples of the types of solutions that can be built include:

- Automating approval processes for key content/documents in your business.
- Replace Excel spreadsheets with application(s) that allow you to connect your data and use it to work smarter and more effectively.
- Remind you something needs to happen, for example your fire safety certificate needs to be renewed before it expires.
- Deliver Management information reports to improve the way that the business makes decisions or provide insight into how the business is performing.
- Automating document processes to generate PDFs or Word documents from other data sources.
- Build an application to manage the jobs and tasks within the business.
- Provide a hub to help users find the information that they need.
- To provide a support tool where on-site engineers can request help with a problem and have access to knowledge base videos and information.
- Use Artificial Intelligence (Machine Learning) to automatically process invoices and extract information such as supplier name, invoice total, invoice date etc from a scanned invoice.

The opportunities for the types of solutions that can be delivered are endless.

Using the Microsoft 365 platform as the foundation to build solutions on provides the following benefits:

- Quicker delivery cycle
- Lower cost to deliver.
- Easier to change.
- Achieve things at a reasonable cost which are hard and/or expensive to do.

Why iThink 365?

We approach these projects with a value-based approach so we are looking at what will the solution do for and what value to your business will it create.

Our team has the following qualities:

- Skilled and certified engineers with experience in building solutions on Microsoft 365.
- Projects are backed by a design authority with 15+ years of experience building IT solutions on the Microsoft platform.

- Reuse, we bring to the table our experience and will reuse approaches which have worked before and highlight approaches which have not worked.
- We have a rounded approach thinking about the full implication of delivering the solution.

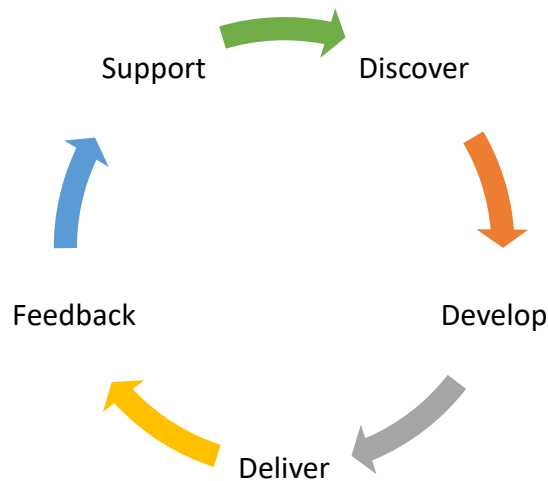
When we are designing and building the solution we are thinking about the following:

- What is the best for you?
- How to ensure your system is secure.
- What are the licensing implications and costs?
- What is the approximate cost of running your solution?
- How can the application be designed which works for your users (user centric design)?

Our approach

We deliver solutions using the following approach.

- Discover – understand and build a vision of your application. Gather the requirements (user stories) and document the solution design.
- Develop – Build, develop and demo the application in small development cycles (sprints)
- Deliver – Perform end-user training, admin training and handover the application.
- Feedback – solicit feedback, understand what worked, what did not work and if there are any immediate issues that need to be addressed.
- Support – providing you with application support.



Discovery Phase

Project Kick off

This will be a virtual session with key personnel at your organisation to mark the start of the project.

The session is usually a two-hour session which:

- Define project goals and objectives to measure success.
- Establishes the roles/responsibilities.
- Defines expectations from the project team.
- Establishes ways of working.

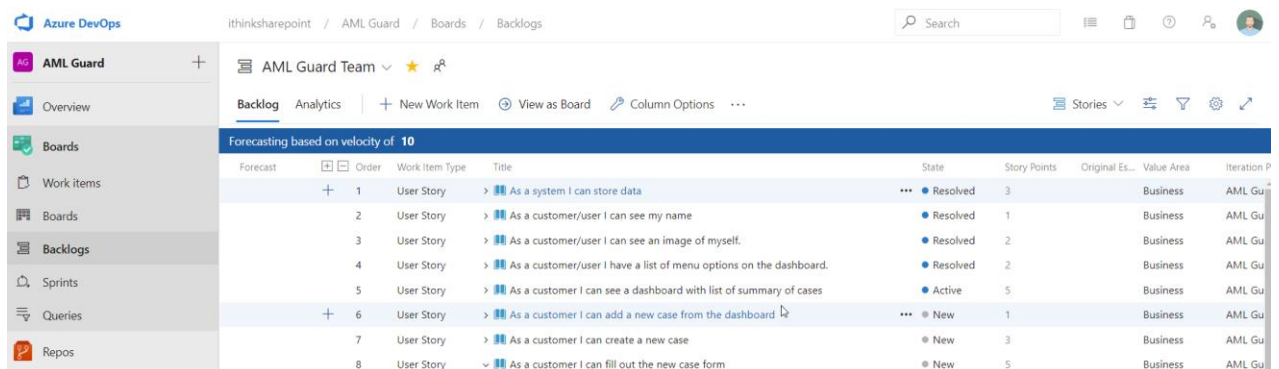
Requirement workshop

The work starts with a requirement gathering workshop. This will delve into the challenges that the solution is to resolve. The workshop gets into the detail so that the team highlight the requirement in full.

This is essential so the solution can be designed, estimated, developed, and delivered with confidence.

The output of the requirements workshop is captured into our development platform, Azure Dev Ops. This defines a product backlog for the solution. An example of a set of requirements in Azure Dev Ops is shown in **Figure 1**. The requirements are written up as user stories which provide three key pieces of information:

- Who is the requirement aimed at?
- A brief description of the requirement.
- A reason why the requirement is needed.



The screenshot shows the Azure DevOps interface for the 'AML Guard' team. The 'Backlog' view is active, displaying a list of user stories. The table below represents the data shown in the screenshot.

Forecast	Order	Work Item Type	Title	State	Story Points	Original Es...	Value Area	Iteration P
+	1	User Story	> As a system I can store data	Resolved	3		Business	AML Gu
	2	User Story	> As a customer/user I can see my name	Resolved	1		Business	AML Gu
	3	User Story	> As a customer/user I can see an image of myself.	Resolved	2		Business	AML Gu
	4	User Story	> As a customer/user I have a list of menu options on the dashboard.	Resolved	2		Business	AML Gu
	5	User Story	> As a customer I can see a dashboard with list of summary of cases	Active	5		Business	AML Gu
+	6	User Story	> As a customer I can add a new case from the dashboard	New	1		Business	AML Gu
	7	User Story	> As a customer I can create a new case	New	3		Business	AML Gu
	8	User Story	> As a customer I can fill out the new case form	New	5		Business	AML Gu

Figure 1 - An example of Azure Dev Ops Product Backlog

The product backlog is used to build the detailed design document and estimate the required effort.

Detailed design document

A design document will be created from the list of requirements. This design document will provide the following:

- Overview of the entire solution and its processes.

- Detailed design description to explain what is to happen at each step.
- The system's architecture diagram.
- Breakdown of sub processes.
- Supporting documentation for each component in the diagram, including role and responsibility.
- Data Model and Information Architecture.

The design documentation will be shared with the project team. A session will be arranged to discuss the design, agree changes, and get sign off.

Note:

The design phase once complete may adjust the effort/costs which are currently provided in this proposal. This will be agreed with all parties before the development phase can commence.

Development Phase

Build solution.

This will deliver the application as described by the design documentation created in the discovery phase.

Our approach to development uses an agile development approach where we develop in sprints. There are daily stand-ups with the development team which you/product owner can join. We do ask that we meet at least once a week to discuss progress with the customer. This provides a mechanism to highlight issues, ask questions and keep the project team involved in the process.

At the end of each sprint the development team will invite the rest of the project team to view a demo of the solution as it stands at that point in time.

This allows you to give feedback, suggest improvements or changes as often its only when you can see the solution do you appreciate more of what you want.

During the development process there are quality assurance steps. These integrate and test the solution to ensure it works as intended. This is essential for delivering a quality solution.

Our approach to development uses automated Dev Ops processes. This allows the solution to be delivered automatically with continuous integration and continuous deployment practices.

Once the development phase has been completed then the customers UAT environment is provisioned using our automated Dev Ops toolchain. This enables constancy, reducing the risk of error and configuration drift as the solution is deployed in each environment.

Let us discuss an example solution to help give you an understanding of what services are involved in a solution.

An example solution

The following section provides an example of a common solution approach. This example will make use of the following services in Microsoft 365:

- Microsoft SharePoint Online.
- Microsoft Teams
- Microsoft Power Apps.
- Microsoft Power Automate.
- Power BI.
- Microsoft Azure Functions.
- Microsoft API Management.
- Microsoft Exchange Online

An overview of the architecture is provided in Figure 2.

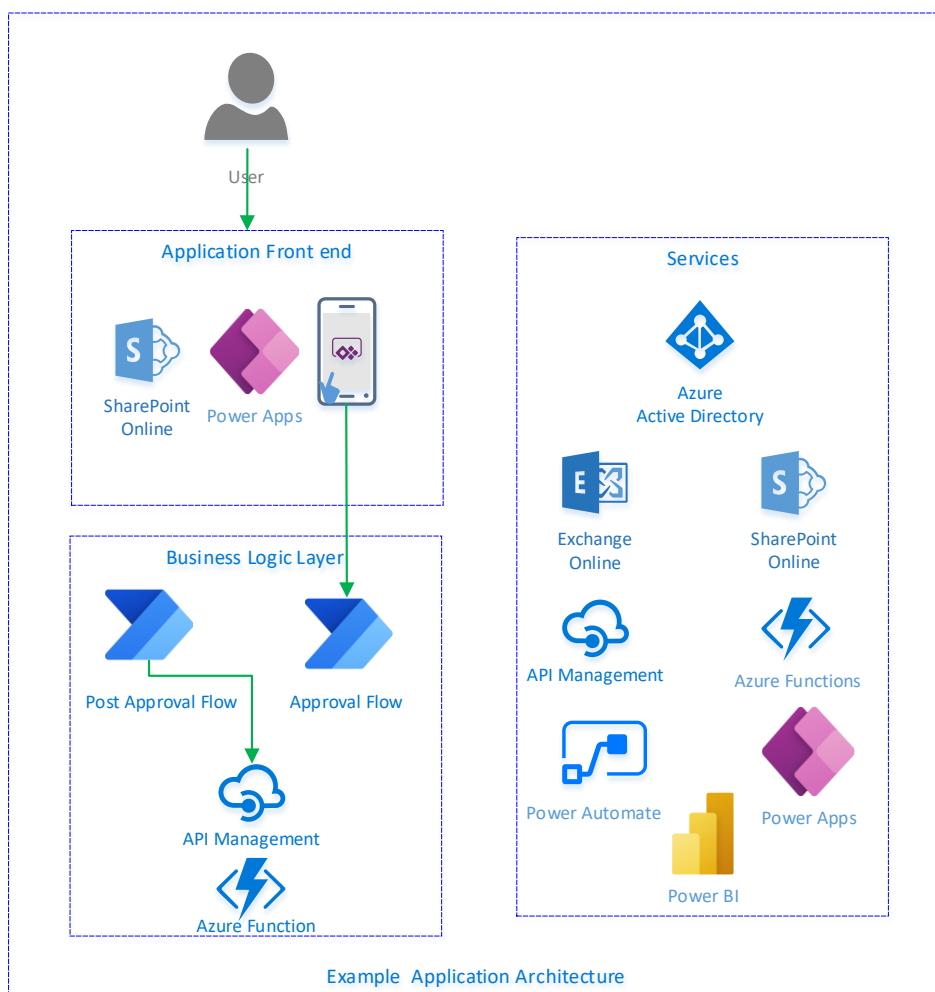


Figure 2 - Solution Architecture Overview

In our example a set of Microsoft 365 services were mentioned, here is what each of them provide.

Service	Description
SharePoint Online	This provides the mechanism to store documents and information related to the document and application. SharePoint Online provides an easy-to-use window into the application. Including links to application support information and user guides so that users can self-help.
Microsoft Teams	Microsoft Teams provides the entry point to the application. It provides a single pane of glass where we can integrate the various components so that the user does not have to hunt to find the application.
Power Apps	Power Apps provides a user interface for the application can be hosted in Microsoft Teams. It is a rapid application development environment for building forms to capture data.
Power Automate	Power Automate is the workflow platform within Microsoft 365. It provides the glue to an application and implements the process steps for the application. It can integrate with other systems using connectors.

Power BI	Power BI provides management information dashboards and reporting for the application.
Microsoft Azure Functions	Azure Functions provide a cheap way to host and execute custom code. Sometimes Power Automate connectors cannot provide the functionality that you need, here we use Microsoft Azure Functions to meet that functionality need.
Microsoft API Management	This is used to secure and provide access to the Microsoft Azure Functions.
Exchange Online	This provides email and messaging services.

User Guide and Admin Guide

Documentation is important as without it, key information is forgotten or misunderstood. This is a common problem that you will have as people change jobs and roles within the organisation or move on.

The User guide is created with the following aims:

- Ensure that users understand how the process works.
- Ensure that users understand how to use the solution.
- Provide a set of Frequently Asked Questions to resolve commonly asked questions or problems.

The user guide is provided in Microsoft Word format and is written with the end user in mind so its not overly technical in detail.

The Admin guide provides information to the administrators and/or 1st / 2nd line support teams.

The admin guide will be created with the following aims:

- Provide a detailed operating manual.
- Explain how to administer the system.
- Explain how the system works.
- Explain application settings and where to find them.
- Explain security configuration such as permissions setup.
- How to find out more information when something goes wrong.
- Provide troubleshooting steps.

Delivery Phase

This phase is when you get your hands on the solution and we see if the solution works with your users. We will be honest; this is often the time when missing requirements are discovered so be prepared for that.

User Acceptance Testing

Once the user acceptance testing (UAT) environment and user guides are in place then user acceptance testing can take place.

We will install the solution into a user acceptance testing environment which is either a separate Microsoft 365 tenant or in separate site. It is important to have a UAT environment so that change to the application can be managed and introduced to the users. This is as important for the first release and for subsequent releases (presuming there are enhancements to the solution).

This stage requires you to provide a team of users who will perform testing of the solution to ensure that it meets the needs of the business. Ideally, these users are from the department areas that will end up using the system. Also, they should have a range of capabilities so that a cross section of technical literacy is tested.

This stage will identify any issues and missing requirements that need to be resolved for a successful solution.

We will support the user acceptance testing team through the process. For example, providing support to answer any questions that the team has.

Note:

Changes to requirements or missing requirements will go through change control and a cost will be provided to implement the change. This will require sign off before the change can be made.

Production Installation

Once the user acceptance testing phase has been completed and you are happy with the solution. The release is signed off and we will work with your administrators to install the solution into production. This will involve a discussion with the Microsoft 365 administration team to explain the installation process, any pre-requisites that are needed.

The installation will be performed with the Microsoft 365 administration team.

End User Training and Administrator Training

Our approach to training is to provide train the trainer training. This is where key people within the areas of the business are identified and training is provided to them. They can then in turn provide training and support to the end users. The approach provides a good balance between cost and adoption as users like having someone they know and local to them to support them.

However, if you require end-user training, we can provide also via training sessions with up to 6 people at a time.

Administrator training is always provided to the team that are going to look after the solution. These are provided as dedicated sessions which are arranged separately from the end-user training.

Feedback Phase

The feedback phase is where you can tell us how the project went. We are interested in hearing about:

- What went well?
- What went badly?
- Are there any outstanding issues that need to be resolved?
- Would you use again?
- What is next on the roadmap?

This allows you to get a better service from us and gives an opportunity to correct things before they get to uncomfortable.

Support Phase

On-going maintenance

To provide you with the experience that we feel you deserve we provide all our solutions with 12 months maintenance.

Software does develop issues and over time it will not behave as we would all want it to. So, we provide to you third line support as part of standard deliverable.

Support can be requested 24/7 from our support desk, however our SLA is to respond and resolve with office hours Monday – Friday (9:00 – 17:00) with an 8-hour response.

How much does the service cost?

To provide a cost for delivering a solution is difficult because the solution has so many variables. However, commonly we see projects which are 20-30 days in length and have a cost of £18000-£25000 with support for 12 months.

The other area that needs to be considered are the following:

- Licensing costs
- Cloud costs

An overview of the licensing costs with an estimate for cloud costs will be discussed and provided as part of the design documentation during the discovery phase.

Please note prices are exclusive of VAT.

How long does it take?

As mentioned above in terms of project days, it is 20 – 30 days but the entire elapsed time depends on the organisation and normally is between 2 and 3 months from the start of the project.