EUROPEAN COLLABORATION
SUMMIT 2024

ARCHITECTING AND BUILDING END-TO-END MICROSOFT 365 SOLUTIONS

PAOLO PIALORSI
@PaoloPia

Solution Architect @ PiaSys.com
MVP, MCSM

EUROPEAN COLLABORATION SUMMIT

Microsoft run.events

AURUM AvePoint CoreView dox42

EasyLife 365 resco veeam

---

adesso business. people. technology. Allied Global ASCENT BCC DE CIX devoteam

empowerID FPT Software glueck kanja Jabra GN kaspersky LightningTools

nintex Rencore ShareGate: Spot by NetApp SysCloud Syskit WEBCON LOW-CODE, BUT BETTER.

Use ECS Coins for Swag!

Top 3 win an Atari 2600+

1. Get the app
2. Visit sessions and sponsors, rate sessions
3. Earn ECS Coins
4. Spend ECS Coins

csmmt.eu/app

run.events

LET ME INTRODUCE MYSELF

- Solution Architect, Consultant, Trainer
  - PiaSys.com based in the USA and in Italy
- More than 50 Microsoft certification exams passed
  - MCSM – Charter SharePoint
  - MVP M365 Development
  - Microsoft 365 & Power Platform Community Steering Group Member
- Focused on SharePoint, Teams, Viva, Copilot, and Microsoft 365
- Author of many books about XML, SOAP, .NET, LINQ, SharePoint, and Microsoft 365
- Speaker at main IT conferences worldwide

- Follow me on X: @PaoloPia
- Subscribe to: https://youtube.com/@PiaSysTechBites

Microsoft SharePoint 2013 Developer Reference
Full Contact XML Web Services
Build Windows 8 Applications with Microsoft Visual C# and Visual Basic Step by Step
Microsoft SharePoint 2010 Developer Reference
XML
Programmare con XML
Introducing Microsoft LINQ
Building Windows 8 Apps with Microsoft Visual C++ Step by Step
Programming Microsoft LINQ in .NET 4
Programming Microsoft LINQ
Programming Microsoft Office 365

Microsoft CERTIFIED Master

MVP Microsoft® Most Valuable Professional

/PnP

MAIN TOPICS COVERED

- Why building Microsoft 365 Solutions?
- Common architectural needs and challenges
- What is the role of Microsoft Azure + Microsoft 365?
- Architecting real solutions
- Wrap up

- Disclaimer: I’m here to share my own experience on the fields (25 years, so far …)
  - We might be on the same page
  - We might have different opinions/angles
  - Sharing is caring …

WHY BUILDING MICROSOFT 365 SOLUTIONS?

WHY BUILDING MICROSOFT 365 SOLUTIONS?

- Microsoft 365 is an open and extensible platform
- We can extend
  - Microsoft SharePoint Online
  - Microsoft Teams
  - Microsoft Viva Connections
  - Microsoft Copilot
  - Etc.
- Microsoft 365 is a really good host for third party solutions, too
- Rather than reinventing the wheel, you should simply rely on Microsoft 365 + Microsoft Azure to create business-level solutions

MOST COMMON SCENARIOS

- Content provisioning
- Content Management Systems
- Order/Contract approval requests management
- Vacation/Time-off requests approval
- Document approvals and management
- Quality assurance management
- Suppliers pipeline management
- 3rd Party Systems integrations
- Etc.

COMMON ARCHITECTURAL NEEDS AND CHALLENGES

COMMON NEEDS/CHALLENGES

- Single-tenant vs Multi-tenant
- Managing data and data isolation in multi-tenant solutions
- Data privacy and security
- Building UI components/layers
- Multiple environments (DEV, TEST, PROD)
- Running background processes
- High Availability
- Asynchronous Processing
- Security: Authentication and Authorization
- Managing Settings
- Reusable Components
- Extensibility
- Governance
    - Logging
    - Monitoring
    - Backup
    - Etc.

MICROSOFT AZURE + MICROSOFT 365

SINGLE-TENANT VS MULTI-TENANT

- Single-tenant
  - You’re building a solution for your company
  - Or a custom, tailor-made solution for a single customer
  - It will not be “reused” on any other businesses (tenant)
  - One deployment/hosting infrastructure
    - No need for data isolation
    - Need for data security, though
- Multi-tenant
  - You plan to reuse the same app across multiple customers/tenants
  - You plan to sell a service via SaaS model
  - One deployment/hosting infrastructure
    - Data isolation across multiple customers
    - Need for data security

HOW TO ISOLATE CONTENT FOR MULTI-TENANCY?

- What is content?
  - Files
  - Items
  - Settings
- Rely on Microsoft 365 for storing “content”
  - Isolation will be “implicit” and “free”
- Consider using
  - Microsoft SharePoint Online for documents
  - Microsoft Lists for lists of items
  - Microsoft OneDrive for Business for personal data
- Microsoft Graph to access data in a secure and per-user partitioned manner

HOW TO ISOLATE DATA FOR MULTI-TENANCY?

- Relational data
  - One DB for each customer
    - Consider using Elastic Pools for better cost management
  - Unique DB with partitioned data
    - *TenantId* everywhere and queries filtered by *TenantId*
      - Mind data privacy and encryption at rest
- No-SQL data
  - One Cosmos DB for each customer
    - Mind the budget … eventually consider the server-less option
  - Unique Cosmos DB
    - One container for each tenant
    - Mind the service limits
- You will need a “Configuration” repository with settings about all the tenants
- If you need to support Search capabilities
  - Consider using Microsoft Azure AI Search
    - But to index SPO data you will need one search instance for each tenant …

DATA PRIVACY AND SECURITY

- Encrypt data at rest
  - Use symmetric keys to encrypt data (Data Encryption Key – DEK)
  - Use asymmetric keys to protect data keys (Key Encryption Key – KEK)
- Use different key for different customers/tenants when in multi-tenant scenario
- Rely on a cloud service like Azure Key Vault to store, manage and rotate keys
- Rely on Azure Entra ID and possibly on Azure Managed Identities to securely access keys
- Services like SQL Azure Database already have Transparent Date Encryption (TDE) by default
  - You can use “Database level customer-managed key (CMK)” instead of server level keys

ALWAYS THINK ABOUT REST APIs (Part 1)

- Implement a back-end of REST APIs
  - Reusable in multiple scenario
- Provide governance APIs
  - Not only functional APIs
- Support OpenAPI with your APIs
  - You can consider using Kiota to generate fluent client libraries …
  - You can easily generate a Postman collection for testing and documentation purposes

ALWAYS THINK ABOUT REST APIs (Part 2)

- Build the front-end and the back-end jobs/services on top of those APIs
- Create the APIs as Azure Functions
  - Highly scalable
  - Easy security
  - Out of the box monitoring
  - Ready to go instrumentation
- Rely on Power Platform Connectors to make the APIs securely available in the Power Platform
  - Power Automate
  - Power Apps
  - Copilot Studio
  - Etc.

BUILDING UI COMPONENTS/LAYERS

- SharePoint Online
  - Web Parts
  - Extensions
- Microsoft Teams
  - Tabs (Personal/Configurable)
  - BOT
  - Messaging Extensions
- Microsoft Viva Connections
  - Adaptive Card Extensions (ACEs)
- Microsoft Power Apps
  - To easily build low-code UI apps/forms
- Microsoft Copilot
  - Plugins/Graph Connectors
  - Custom Copilots with Microsoft Copilot Studio
- Rely on reusable UI components
  - Adaptive Cards JSON
  - Microsoft Graph Toolkit
  - PnP Controls

MICROSOFT 365 EXTENSIBILITY POINTS OF ATTENTION

- Teams Message Extensions
  - To extend the Search experience
  - To improve the message composition experience
  - To easily extend Microsoft 365 Chat (Microsoft Copilot for Microsoft 365) with plug-ins
- Teams Personal Apps
  - Microsoft Teams
  - Microsoft Outlook
  - Office Portal
- One solution, one set of back-end REST APIs, one shared back-end logic
  - Multiple usage patterns and experiences

REUSABLE UI COMPONENTS

- Don’t reinvent the wheel
- Rely on consolidated community standards
- Rely on common UI/UX patterns
- Personally, we build the Web UI with React
- Use the de facto standard UI components and design patterns
  - SharePoint Web UI kit
  - Microsoft Graph Toolkit
  - PnP React Reusable Controls

SUPPORTING MULTIPLE ENVIRONMENTS

- Microsoft Azure deployment slots for web apps/jobs/functions/etc.
- Dedicated Azure infrastructure for every single environment
- Multiple Entra ID application registrations (DEV, TEST, PROD)
- Maintainable list of allowed tenants for DEV, TEST, PROD
- Shared secure storage (Azure Key Vault) with security keys/certificates/credentials
- BICEP deployment for all the environments
  - See what Microsoft Teams Toolkit does for example with BICEP + .env files per environment

RUNNING BACKGROUND PROCESSES

- For quick background processes
  - Azure Function Apps
  - Containers
- For long running processes
  - Azure Logic Apps

HIGH AVAILABILITY

- Deploy multiple-instances of services
  - Configure scaling accordingly to your needs
- Keep into account peaks of load
  - Auto-scale
  - Asynchronous processing
    - You can’t always process any request in real-time

ASYNCHRONOUS PROCESSING

- Define back-end APIs and services
  - Azure Functions in Azure Function Apps are one of the best options
    - Can easily work on schedule
    - Can be executed upon triggers
    - Can be easily monitored and instrumented
  - Consider creating Azure Logic Apps
    - If you rather want a more low-code oriented approach
    - Or if you need to run long-running processes
      - Eventually with human-interaction
- Rely on asynchronous communication channels
  - Azure Blob Queue
  - Azure Service Bus
    - Scales better …
  - Both can trigger an Azure Function or a Logic App

AUTHENTICATION AND AUTHORIZATION

- Register one Entra ID application for every environment
  - Keep into account the consent flow, especially in multi-tenant apps
  - Mind permission updates during solution lifecycle (additional consents required)
  - Consider that you can expose APIs, too … not only consume other APIs
    - Secure communication with your Functions (consider EasyAuth …)
- Rely on OAuth 2.0 for authorization
  - Think carefully about Delegated vs Application permissions
    - Mind SPO requirements (like X.509 certificate for app-only)
  - Use MSAL or Azure.Identity for token retrieval
- Use the On-Behalf-Of flow to securely access back-end APIs/Services on behalf of a user
- Need to support external identity providers?
  - Consider using AAD B2C
    - Mind Microsoft Graph and Microsoft 365 APIs permissions limits for external users

SECURITY ACROSS SERVICES AND ARCHITECTURAL LAYERS

- Single-tenant solution
  - Rely on Azure Managed Identities for cross-services security
  - Best option for managing access to resources secured with Entra ID
    - Can be native Azure resources
    - Can be your own custom applications
  - Credentials of Managed Identities are managed, rotated, and protected by Azure
    - Internally relies on Managed Identity Resource Provider (MSRP) and on an internally issued certificate
  - No one can access the credentials
    - The Global Admin neither
    - You don’t need any longer to rely on passwords, secrets, certificates, etc.
- Multi-tenant solution
  - Rely on OAuth 2.0 for cross-services security

MANAGING SETTINGS

- Application Settings
  - Single-tenant: you can simply consider using Azure settings (App Service, Function Apps, Logic Apps, etc.)
    - You can secure settings with Azure Key Vault
  - Multi-tenant: rely on a configuration repository (SQL or no-SQL)
    - Security while accessing these settings is a key requirement!
  - In case you have multiple services/components/nodes consider Azure App Configuration service
    - Supports encryption at rest and complements Azure Key Vault
  - If the solution is based on SharePoint Online, you can also consider Tenant Settings
- Users' Settings
  - Consider using a per-tenant configuration repository (SQL or no-SQL)
  - Or rely on Microsoft OneDrive for Business and on the Application's Personal Folder
    - You can store any file, including JSON settings or other stuff
    - Use Microsoft Graph to read/write/manage users' settings

REUSABLE COMPONENTS

- Stop reinventing the wheel!
- Rely on reusable components/frameworks
    - Microsoft Graph SDK
    - PnP Framework
    - PnP Core SDK
    - PnP Provisioning Engine
    - PnP Transformation Framework
    - Etc.

EXTENSIBILITY

- Not all the customers are the same!
  - But most of the basic requirements are the same …
- Think about extensibility points/hooks for custom logic
  - You can eventually plug-in Power Automate flows or Logic Apps flows instead of pro-code extensions
- Provide webhooks-like extensibility points
  - To keep an open extensibility model
  - Keep into account security when providing webhooks

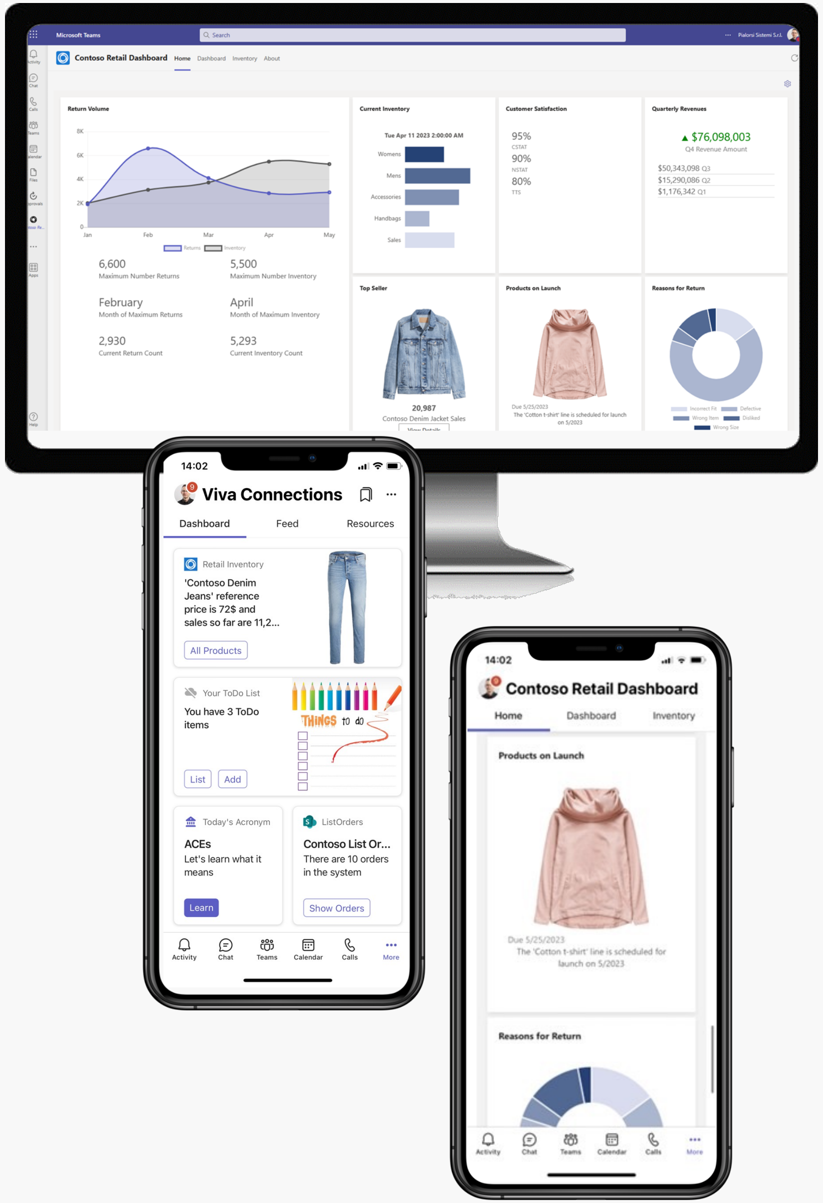
GOVERNANCE: LOGGING, MONITORING, BACKUP, ETC.

- At least use Application Insights for monitoring and logging
- Consider using Azure Log Analytics
  - For querying and analyzing log data
  - For creating charts on log data
- Consider using Azure Monitor
  - Alerts
  - Auto scale
  - Dashboards
  - Power BI integration
- Provide PowerShell/CLI command line management tools/scripts
- Provide reporting for monitoring and measuring usage
- Rely on cloud-based services for backup/restore policies
  - You pay for PaaS … enjoy PaaS …

SAMPLE ARCHITECTURES

CONTOSO RETAIL

- Contoso Retail
  - Fashion retail company
- Functional requirements
  - Dashboard to monitor sells, revenues, and returns
  - Access products information from mobile devices
  - Share information about products in Teams
  - Work on products within Microsoft Copilot
- Rely on Microsoft 365 ecosystem
  - Microsoft Teams, Microsoft Viva, Outlook.com, Microsoft Copilot, and Microsoft 365 in general



▲ $76,098,003
Q4 Revenue Amount

$50,343,098 Q3
$15,290,086 Q2
$1,176,342 Q1

6,600
Maximum Number Returns

5,500
Maximum Number Inventory

February
Month of Maximum Returns

April
Month of Maximum Inventory

2,930
Current Return Count

5,293
Current Inventory Count

20,987
Contoso Denim Jacket Sales

Due 5/25/2023
The 'Cotton t-shirt' line is scheduled for launch on 5/2023

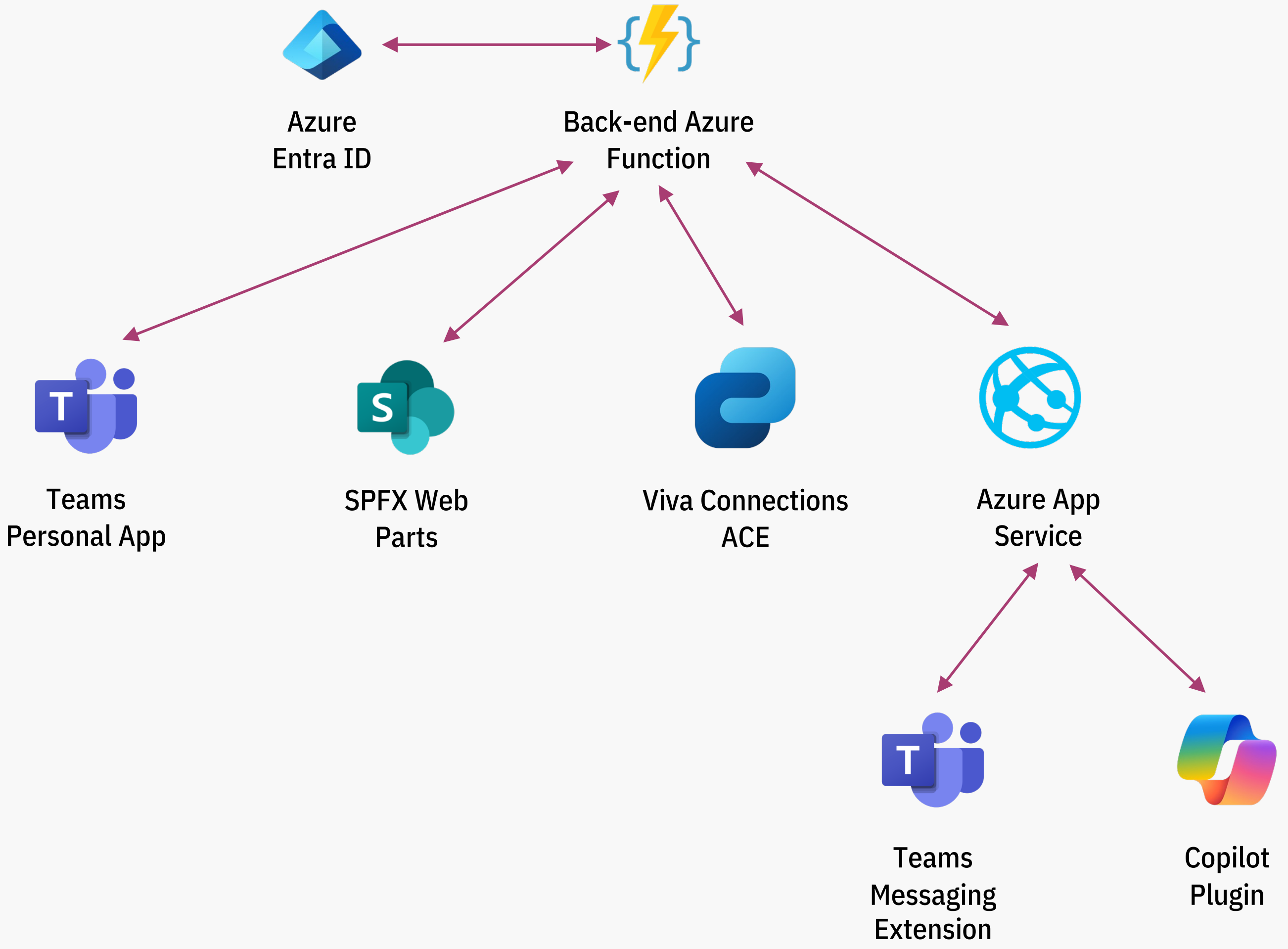
'Contoso Denim Jeans' reference price is 72$ and sales so far are 11,2...

You have 3 ToDo items

ACEs
Let's learn what it means

Contoso List Or...
There are 10 orders in the system

CONTOSO RETAIL



DEMO: CONTOSO RETAIL IN ACTION

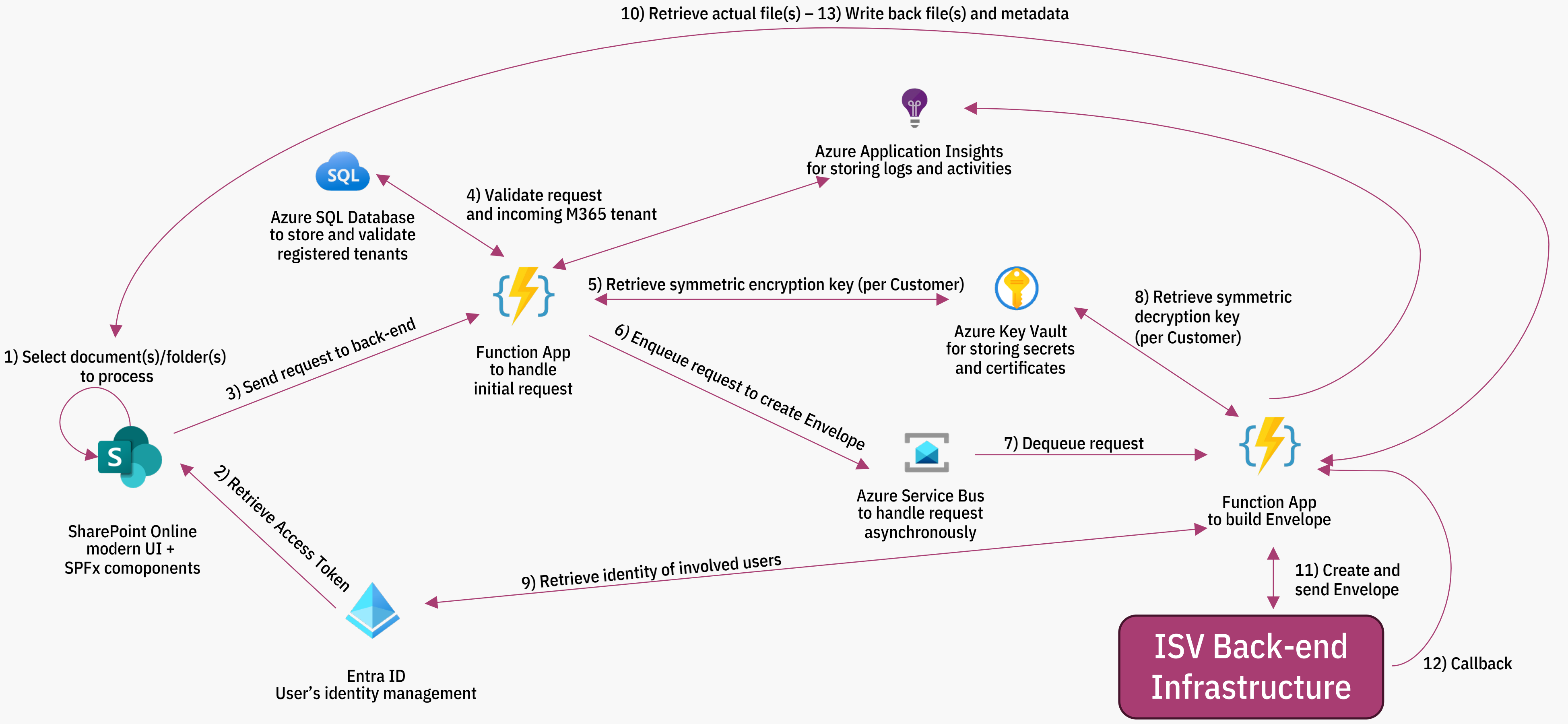
https://adoption.microsoft.com/en-us/sample-solution-gallery/sample/pnp-spfx-reference-scenarios-samples-react-retail-dashboard/

ISV SOLUTION TO “EXTEND” AND “PROCESS” DOCUMENTS STORED IN SHAREPOINT ONLINE

- Multi-tenant solution architecture for an ISV
- Users can select one or more documents
    - Or folders, with documents and subfolders included
    - Any folder hierarchy depth should be supported
- All the documents will go through processing
    - Using a 3rd party platform
    - Existing documents can get metadata augmentation or content changes
    - New documents can be created with custom metadata
    - A long running (weeks/months) process can be run in the back-end
- The main UI should be in SharePoint Online

EUROPEAN COLLABORATION SUMMIT

ISV SOLUTION TO “EXTEND” AND “PROCESS” DOCUMENTS STORED IN SHAREPOINT ONLINE



WRAP UP

WRAP UP

- Microsoft 365 is an open and extensible platform
- Don’t reinvent the wheel
- Reuse what you have to save code, time, tests, etc.
- Use Microsoft Azure PaaS as much as you can for hosting
- Use Azure Entra ID as much as you can to secure the whole communication across layers
- Use an Asynchronous processing model to scale the right way
- Don’t forget to monitor and log as much as you can, you will not regret it!

- And … Copilot (one more time 😉) just for the sake of it! 😂

THANK YOU,
YOU ARE AWESOME ❤️

PLEASE RATE THIS SESSION IN THE MOBILE APP.

X: @PaoloPia
LinkedIn: https://www.linkedin.com/in/paolopialorsi/
YouTube: http://youtube.com/@PiaSysTechBites