Office 365 Web Add-Ins

Office 365 Add Ins Quality You Can See.

Cognitive Convergence

Http://www.Cognitiveconvergence.Com

Voice: +1 4242530744

Shahzad@cognitiveconvergence.com

ABOUT US

Cognitive Convergence is Subject Matter Expert in Office 365, Dynamics 365, SharePoint, Project Server, SAAS, Power Platform: Power Apps-Power BI-Power Automate-Power Virtual Agents

Our Core Office 365 services includes the following

- ✓ Office 365 – Cloud /On-premises
- ✓Customization in Office 365 Add-Ins Solutions
- ✓Business Consultancy
- ✓Office 365 integration
- ✓Office 365 support
- ✓Office 365 Add-Ins development
- ✓Rest API Development
- ✓Developing components, controls and plugins
- ✓Business Operations respective to the organizations
- ✓Office 365 managed services
- ✓Cloud license and account management

Current Location: Lahore, Pakistan

Planned Front-end Office: California/Washington States- USA

OFFICE 365 WEB ADD-IN CONSULTING SERVICE

- ✓ Solutions that extend Office applications and interact with content in Office documents
- ✓ Run in Office across multiple platforms, including Windows, Mac, iPad, and in a browser

Use the Office Add-ins platform to

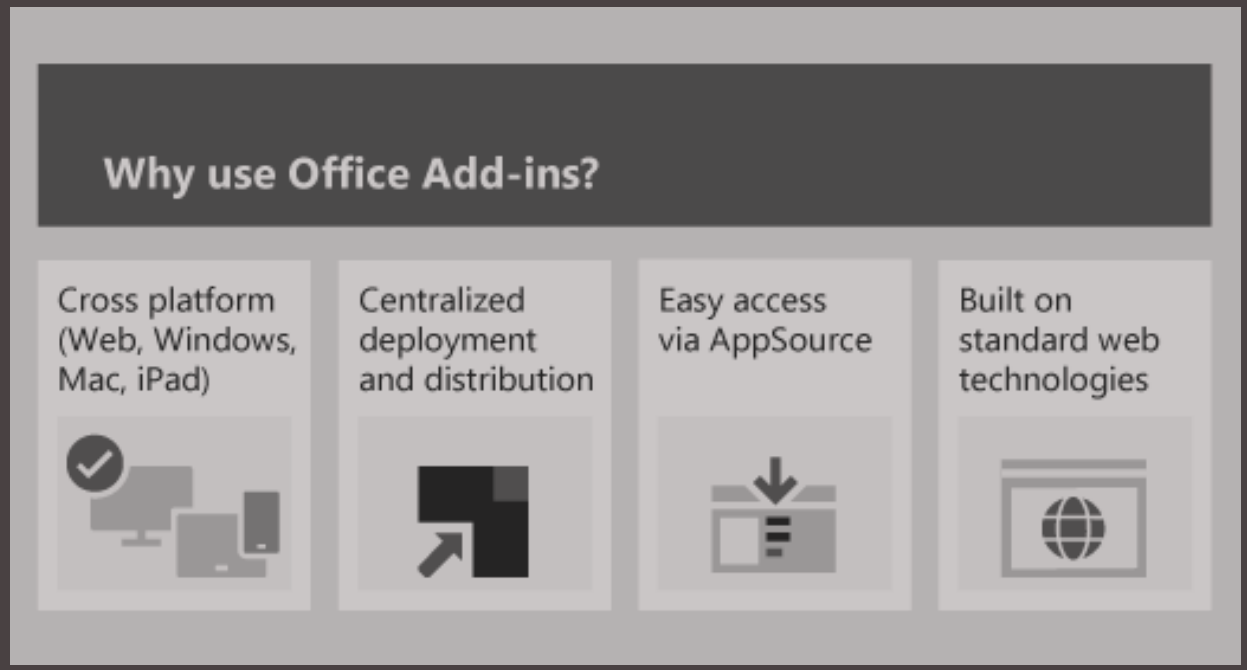
- ✓ Add new functionality to Office clients
- ✓ Create new rich, interactive objects that can be embedded in Office documents

OFFICE ADD-INS VS COM AND VSTO ADD-INS

COM or VSTO add-ins are earlier Office integration solutions that run only in Office on Windows. Office Add-ins provide the following advantages over add-ins built using VBA, COM, or VSTO:

- ✓ Cross-platform support. Office Add-ins run in Office on the web, Windows, Mac, and iPad
- ✓ Centralized deployment and distribution. Admins can deploy Office Add-ins centrally across an organization
- ✓ Easy access via AppSource. Solution can be made available to a broad audience by submitting it to AppSource
- ✓ Based on standard web technology. Any library of user's liking can be used to build Office Add-ins

VSTO Add-ins



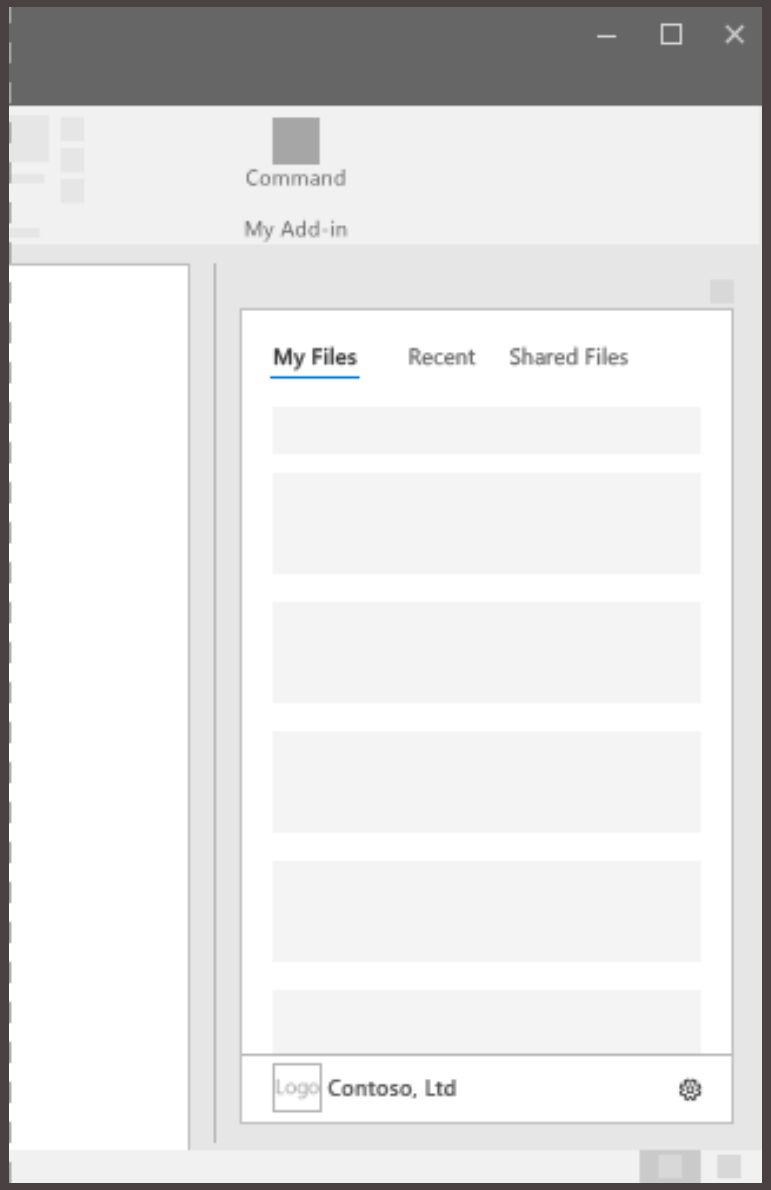
CAPABILITIES OF OFFICE 365 WEB ADD-IN

Office add-ins can

- ✓ add custom ribbon buttons or menu commands
- ✓ insert task panes
- ✓ add custom functions
- ✓ open dialog boxes
- ✓ embed rich, web-based objects such as charts or interactive visualizations

Add-in Commands

- ✓ UI elements that extend the Office UI and start actions in the add-in
- ✓ Used to add a button on the ribbon or an item to a context menu
- ✓ Help users find and use the add-in

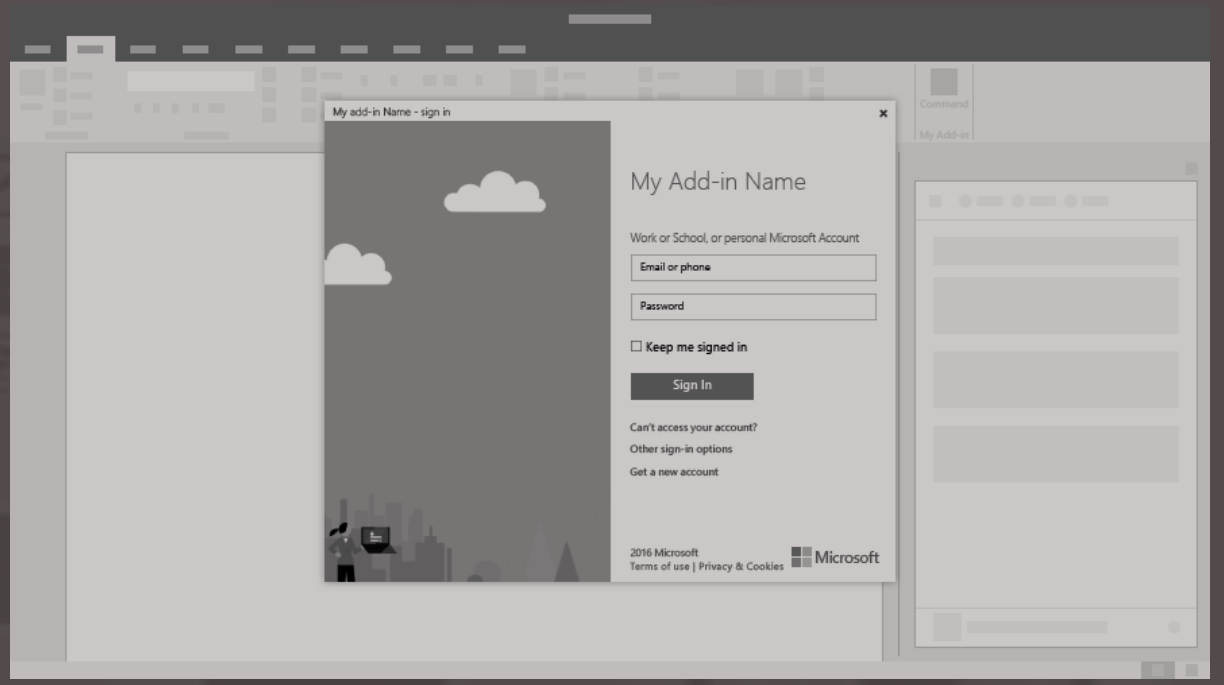


Custom Functions

- ✓ Enable developers to add new functions to Office application by defining those functions in JavaScript as part of an add-in
- ✓ Users can access custom functions just as they would any native function in Office applications

Dialog boxes

- ✓ Surfaces that float above the active Office application window
- ✓ Used to provide additional screen space for
- ✓ tasks such as sign-in pages

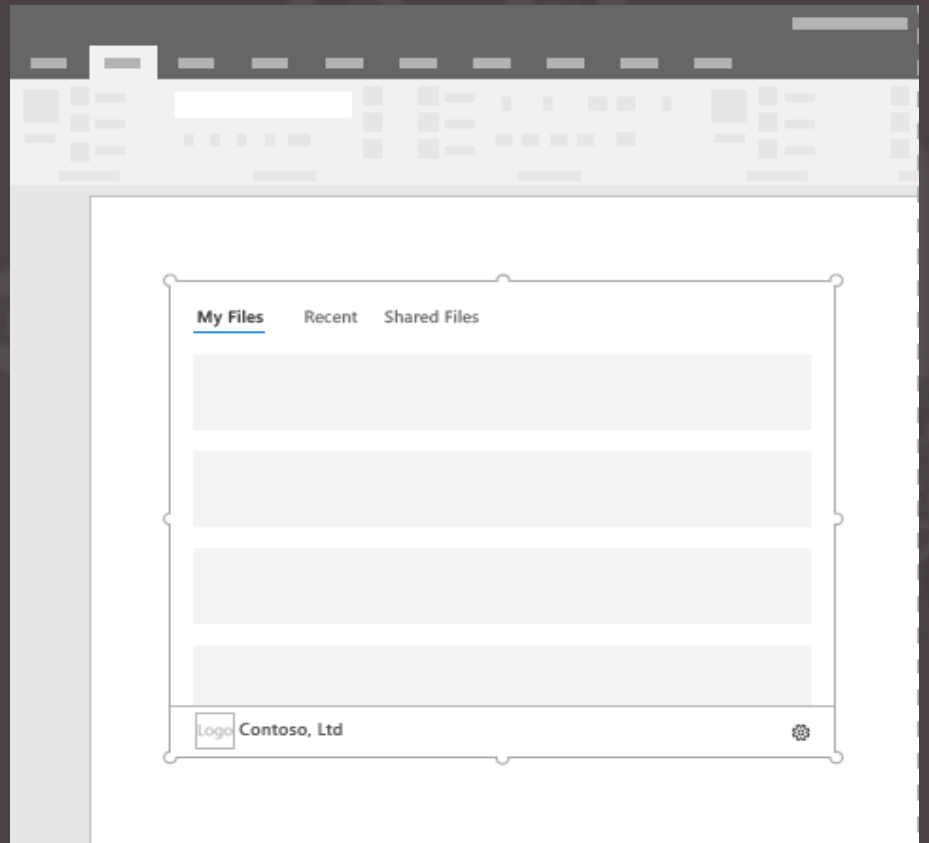


Content add-ins

- ✓ Surfaces that can be embedded directly into Excel or PowerPoint documents
- ✓ Give users access to interface controls that run code to modify documents or display data from a data source
- ✓ Used when user want to embed functionality directly into the document

Task panes

- ✓ Interface surfaces that typically appear on the right side of the window within Word, PowerPoint, Excel, and Outlook
- ✓ Give users access to interface controls that run code to modify documents or emails, or display data from a data source



Office 365 web Add-in Architecture

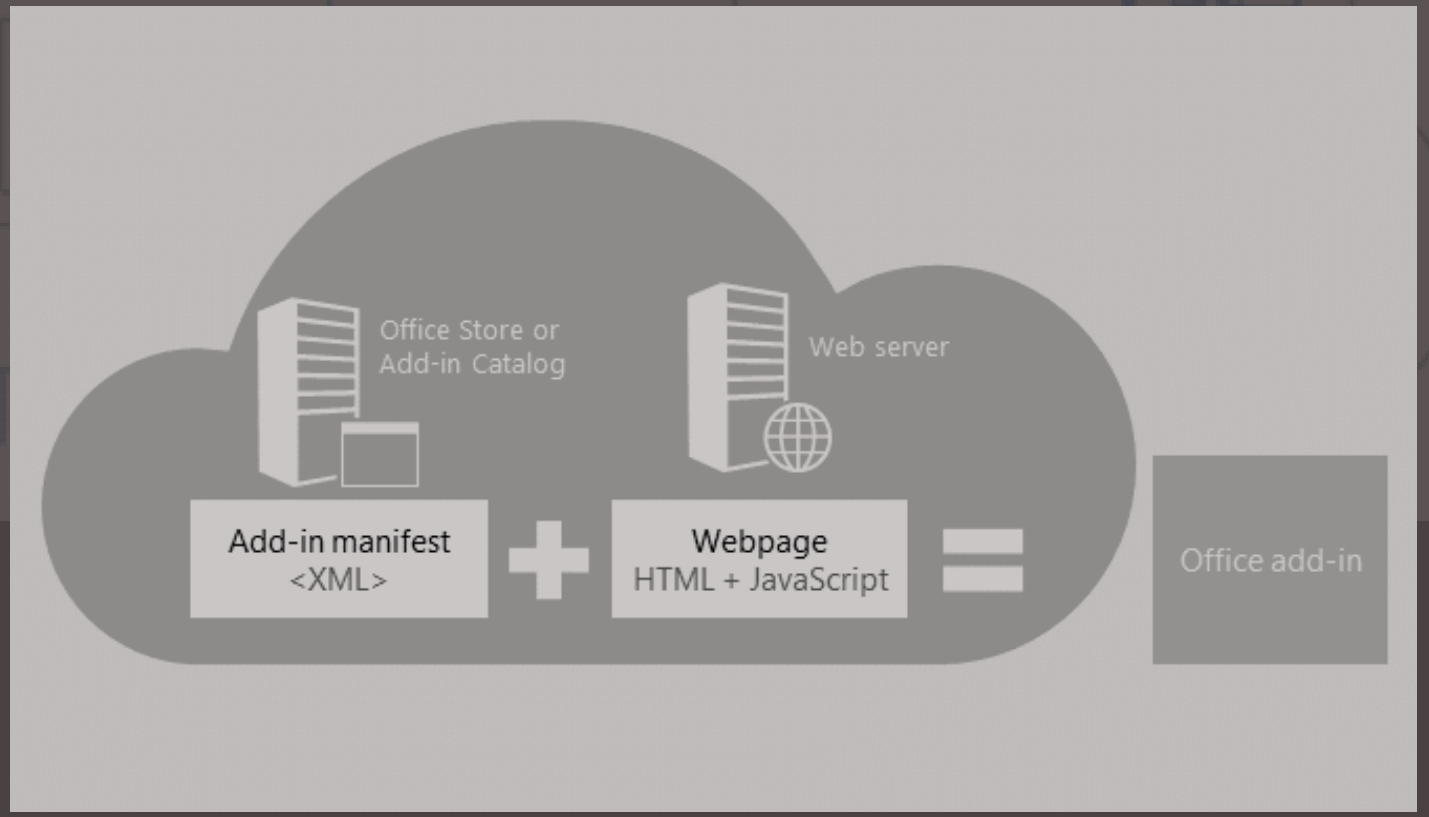
An add-in solution has two basic parts in its building architecture which are described below:

Add-in project

- ✓ Contains only an XML manifest file, which contains all the settings that describe the add-in
- ✓ Help the Office host determine when the add-in should be activated and where the add-in should appear
- ✓ User can change these settings any time by modifying the XML file

Web application project

- ✓ Contains the content pages of add-in, including all the HTML and JavaScript pages
- ✓ While developing the add-in, Visual Studio hosts the web application on the user's local IIS server



Excel Add-in

Extends Excel application functionality across multiple platforms including Windows, Mac, iPad, and in a browser

Use Excel add-ins within a workbook to:

- ✓ Interact with Excel objects, read and write Excel data.
- ✓ Extend functionality using web-based task pane or content pane
- ✓ Add custom ribbon buttons or contextual menu items
- ✓ Add custom functions
- ✓ Provide richer interaction using dialog window

Excel JavaScript API

An Excel add-in interacts with objects in Excel by using the Office JavaScript API, which includes two JavaScript object models:

Excel JavaScript API

- ✓ Application-specific APIs for Excel
- ✓ Provides strongly-typed objects that can be used to access worksheets, ranges, tables, charts, and more

Common APIs

- ✓ Used to access features such as UI, dialogs, and client settings that are common across multiple types of Office applications

PowerPoint Add-in

Used to build engaging solutions for users' presentations across platforms including Windows, iPad, Mac, and in a browser

There are two types of PowerPoint add-ins:

- ✓ **Content add-ins**

Add dynamic HTML5 content to presentations

- ✓ **Task pane add-ins**

To bring in reference information or insert data into the presentation via a service

PowerPoint JavaScript APIAdd custom functions

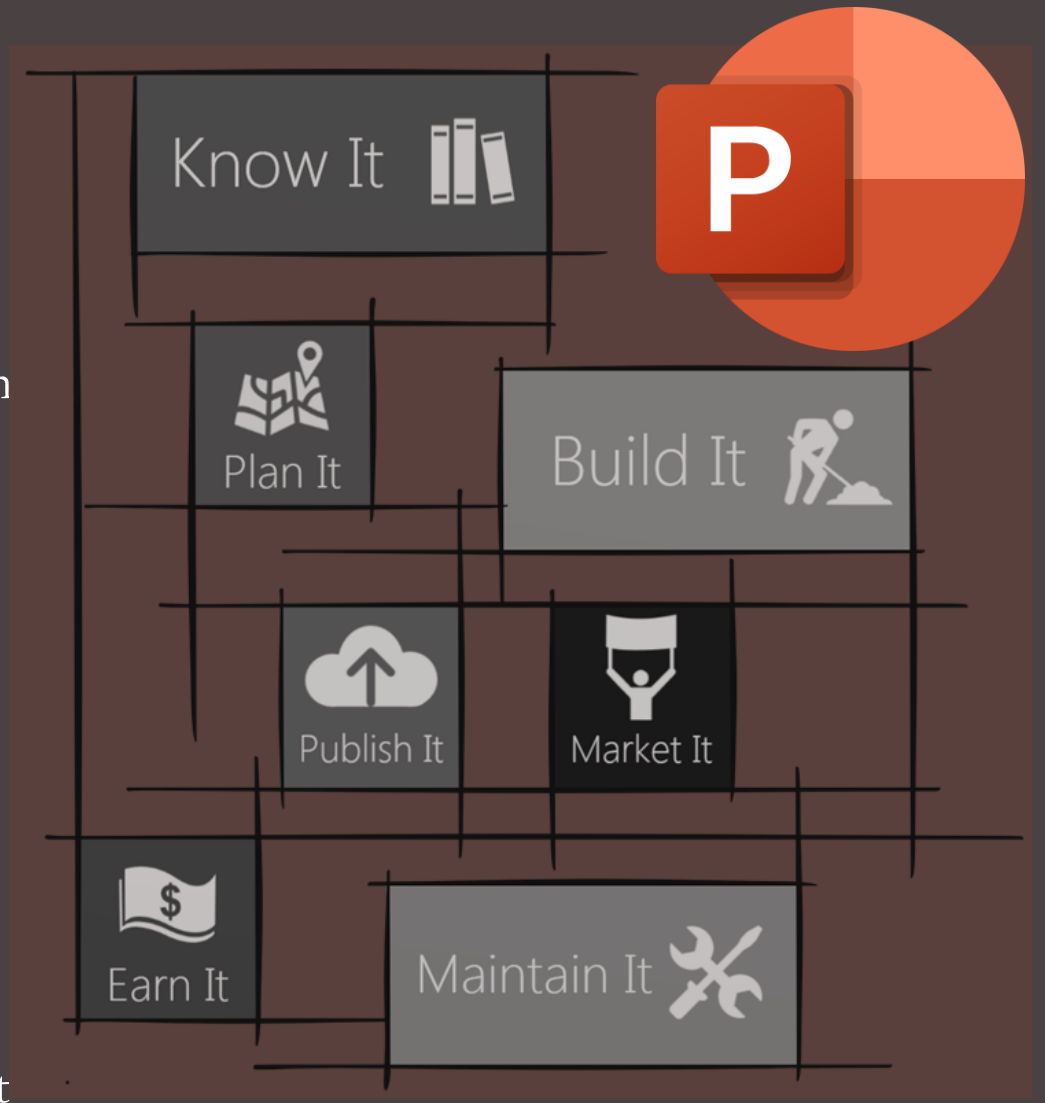
A PowerPoint add-in interacts with objects in PowerPoint by using the Office JavaScript API, which includes two JavaScript object models

PowerPoint JavaScript API

- ✓ Application-specific APIs for PowerPoint
- ✓ Provides strongly-typed objects that can be used to access objects in PowerPoint

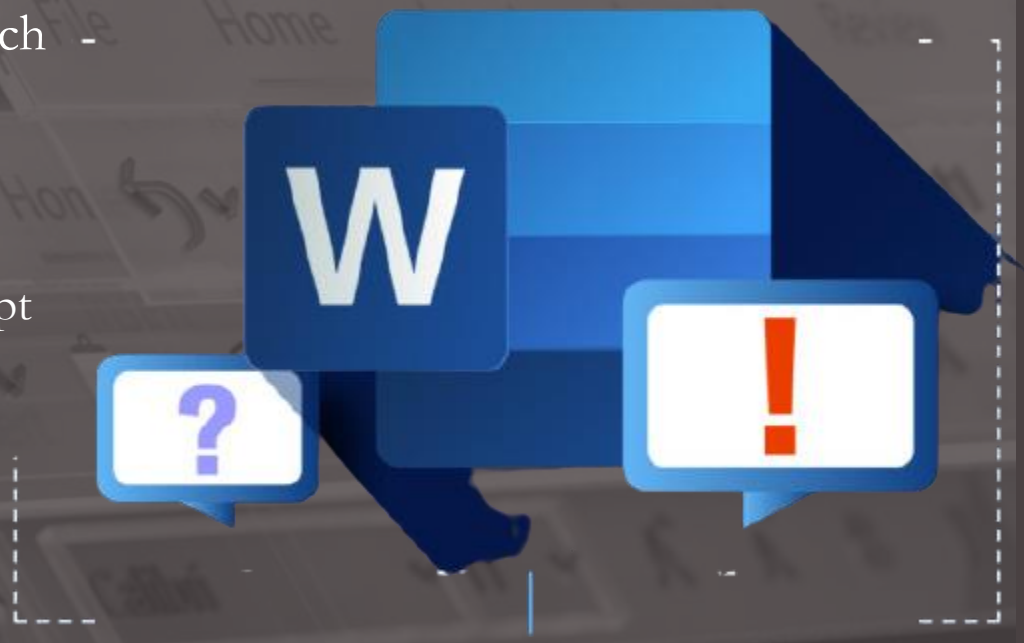
Common APIs

- ✓ Used to access features such as UI, dialogs, and client settings that are common across multiple types of Office applications



Word Add-in

- ✓ Includes the Word JavaScript API and the Office JavaScript API to extend Word clients running on a Windows desktop, on a Mac, or in the cloud
- ✓ Use JavaScript to access the paragraph object and update, delete, or move the paragraph
- ✓ Add-in commands can be used to extend the Word UI and launch task panes



Word JavaScript API

A Word add-in interacts with objects in Word by using the Office JavaScript API, which includes two JavaScript object models:

Word JavaScript API

- ✓ Application-specific APIs for Word
- ✓ Provides strongly-typed objects that can be used to access objects and metadata in a Word document

Common APIs

- ✓ Used to access features such as UI, dialogs, and client settings that are common across multiple types of Office application

Outlook Add-in

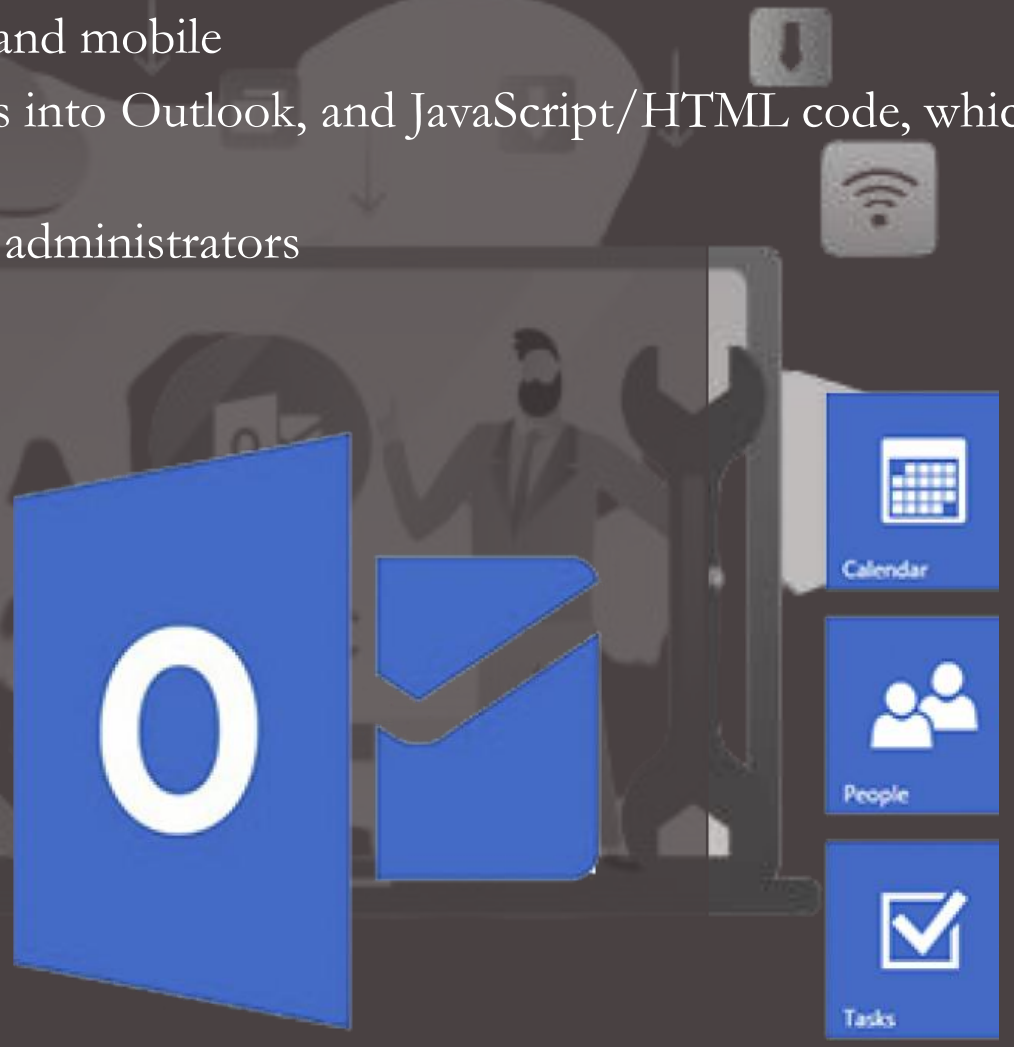
- ✓ Integrations built by third parties into Outlook by using web-based platform

Outlook add-ins have three key aspects:

- ✓ The same add-in and business logic works across desktop, web, and mobile
- ✓ Consist of a manifest, which describes how the add-in integrates into Outlook, and JavaScript/HTML code, which makes up the UI and business logic of the add-in
- ✓ Can be acquired from AppSource or sideloaded by end-users or administrators

The Outlook items that support add-ins include:

- ✓ email messages
- ✓ meeting requests
- ✓ responses and cancellations
- ✓ appointments



Extension Points

- ✓ Extension points are the ways that add-ins integrate with Outlook

The following are the ways this can be done:

- ✓ An add-in with command buttons on the ribbon
- ✓ A contextual add-in for a highlighted entity (an address)

Outlook add-in APIs

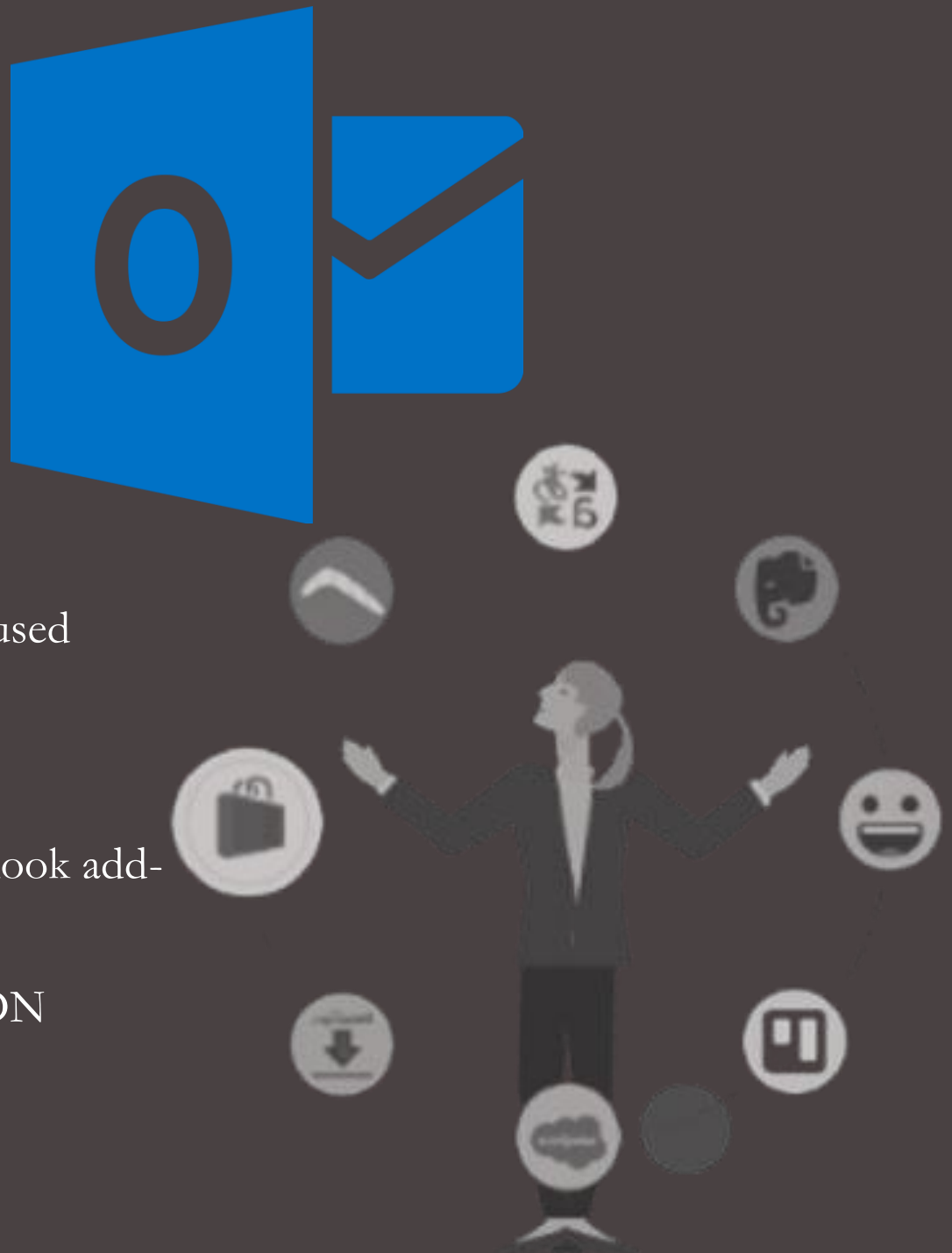
To use APIs in Outlook add-in, followings must be specified:

- ✓ Location of the Office.js library
- ✓ Requirement set
- ✓ Schema
- ✓ Permissions

Office JavaScript APIs exposed through the Mailbox object can be used primarily.

Office.js library

- ✓ JavaScript APIs in Office.js can be used to interact with the Outlook add-in API
- ✓ Add-ins submitted to AppSource must reference Office.js by CDN



OFFICE 365 WEB ADD-IN DEPLOYMENT

There are multiple methods of deploying an add-in solution. Any one from listed below methods can be used to deploy the Office Add-in for testing or distribution to users.

DEPLOY AND PUBLISH OFFICE ADD-INS

Method	Use...
Sideloading	To test add-in running on Windows, iPad, Mac, or in a browser. (Not for production add-ins.)
Network share	To test add-in running on Windows after add-in is published to a server other than localhost. (Not for production add-ins or for testing on iPad, Mac, or the web.)
Centralized Deployment	In a cloud deployment, to distribute add-in to users in the organization by using the Microsoft 365 admin centre.
SharePoint catalog	In an on-premises environment, to distribute the add-in to users in the organization.
AppSource	To distribute the add-in publicly to users.
Exchange server	In an on-premises or online environment, to distribute Outlook add-ins to users.

OFFICE 365 WEB ADD-IN DEPLOYMENT

Office 365 development, implementation, customization, and consultation service of **Cognitive Convergence** offers strategic opportunities to clients, investors, and partners that is

- ✓ Unique and industry defining
- ✓ Mutual interest centric business approach
- ✓ Significantly enhance company's footprint
- ✓ Turn grow revenues by entering into new and exciting **Technology Domains, App development ideas, Solution Development,** and **Joint venture projects**
- ✓ 1st mover advantage with
  - ✓ Talent: 100%
  - ✓ Timing:100%
  - ✓ Technology: 100%
  - ✓ Technique: 100%

THANK YOU

Profile
Shahzad Sarwar

Website
www.cognitiveconvergence.com

Email
shahzad@cognitiveconvergence.com

Phone
+1 4242530744

For questions or queries, contact us, we will be sure to get back to you as soon as possible.