

# Developing for Microsoft Teams



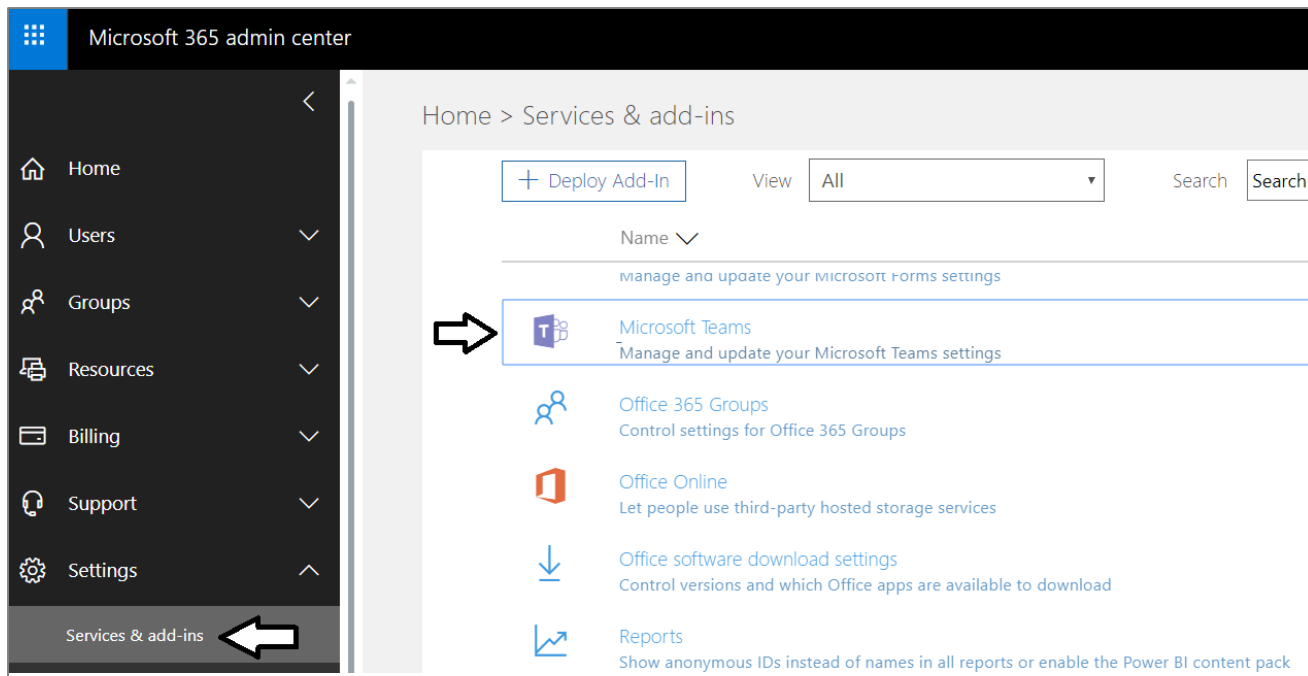
# Agenda

- Understanding the Teams Service, Teams and Channels
- Developing Team Apps with Tabs, Bots and Connectors
- Developing a Teams App using App Studio
- Developing a Teams App using Visual Studio and C#
- Developing a Teams App using Node.js
- Packaging and Publishing a Microsoft Teams App

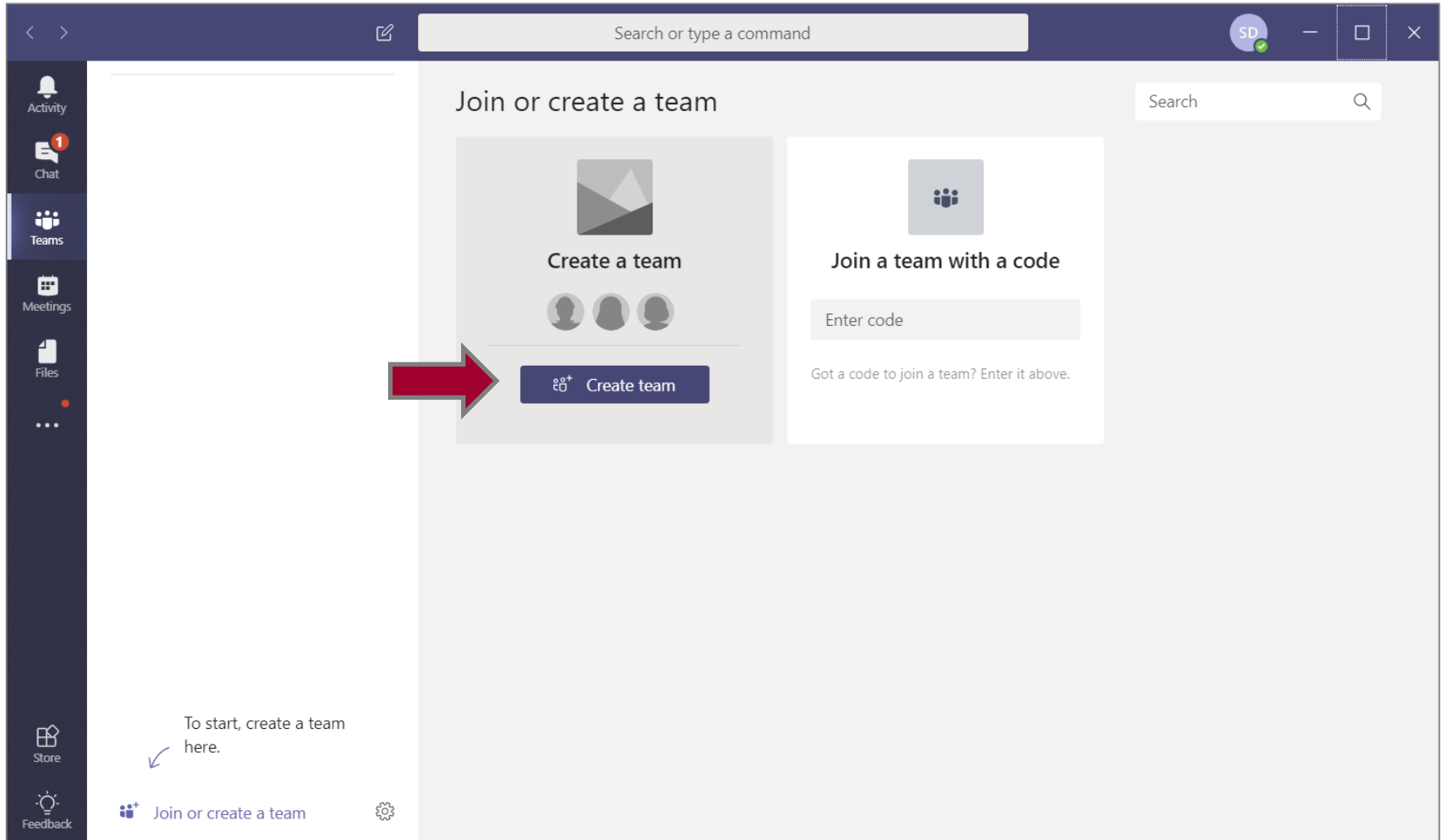


# Introduction to Microsoft Teams

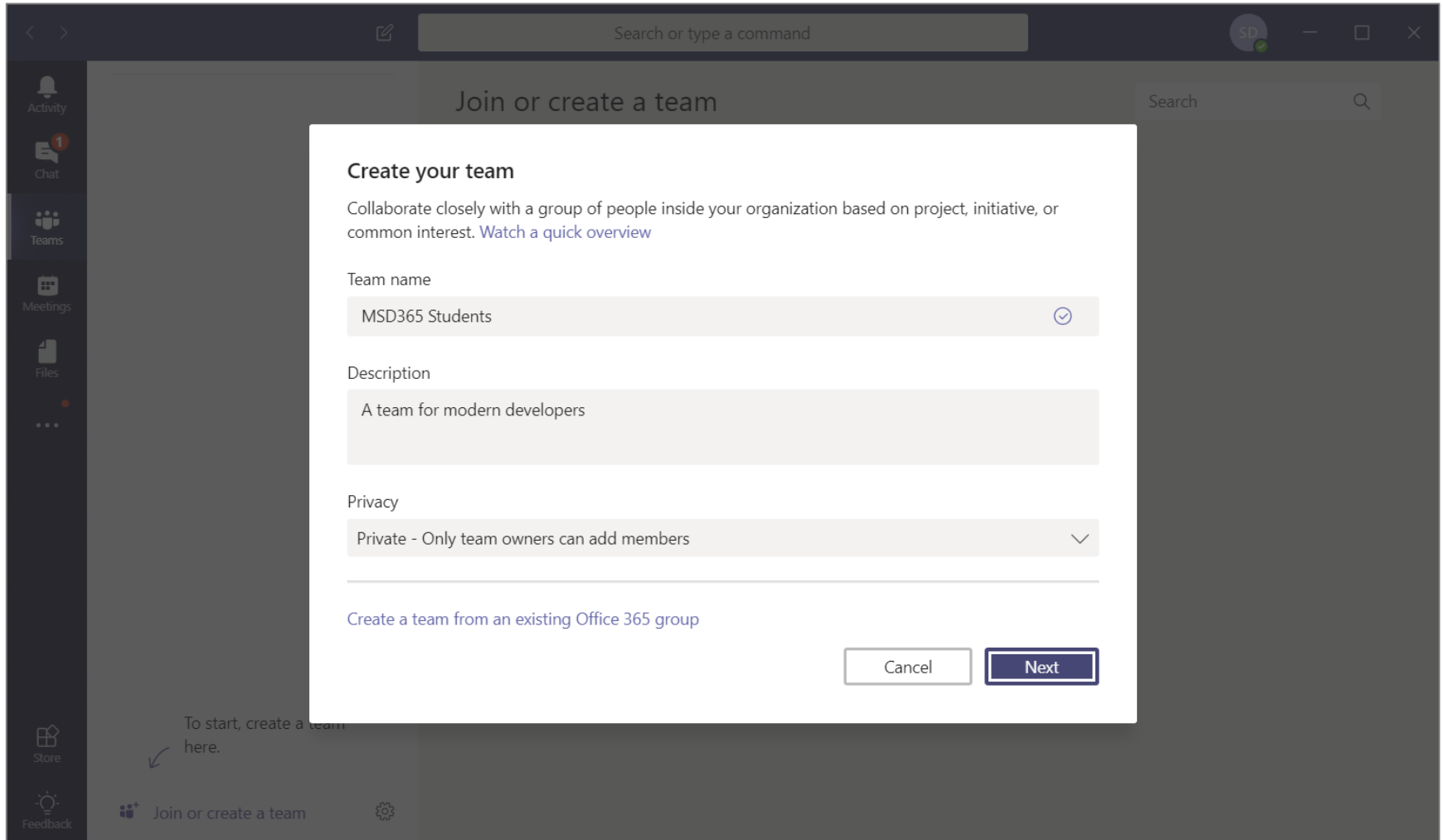
- Microsoft Teams is a Chat-based Workspace Service
  - Based on an evolved version of Skype for Business
  - Created by Microsoft to compete against Slack
  - Provides deep integration with Office 365
  - Offers native apps for Windows 10, Mac, iOS, Android and Windows Phone



# Creating a New Team



# Assign Team Name and Privacy Level



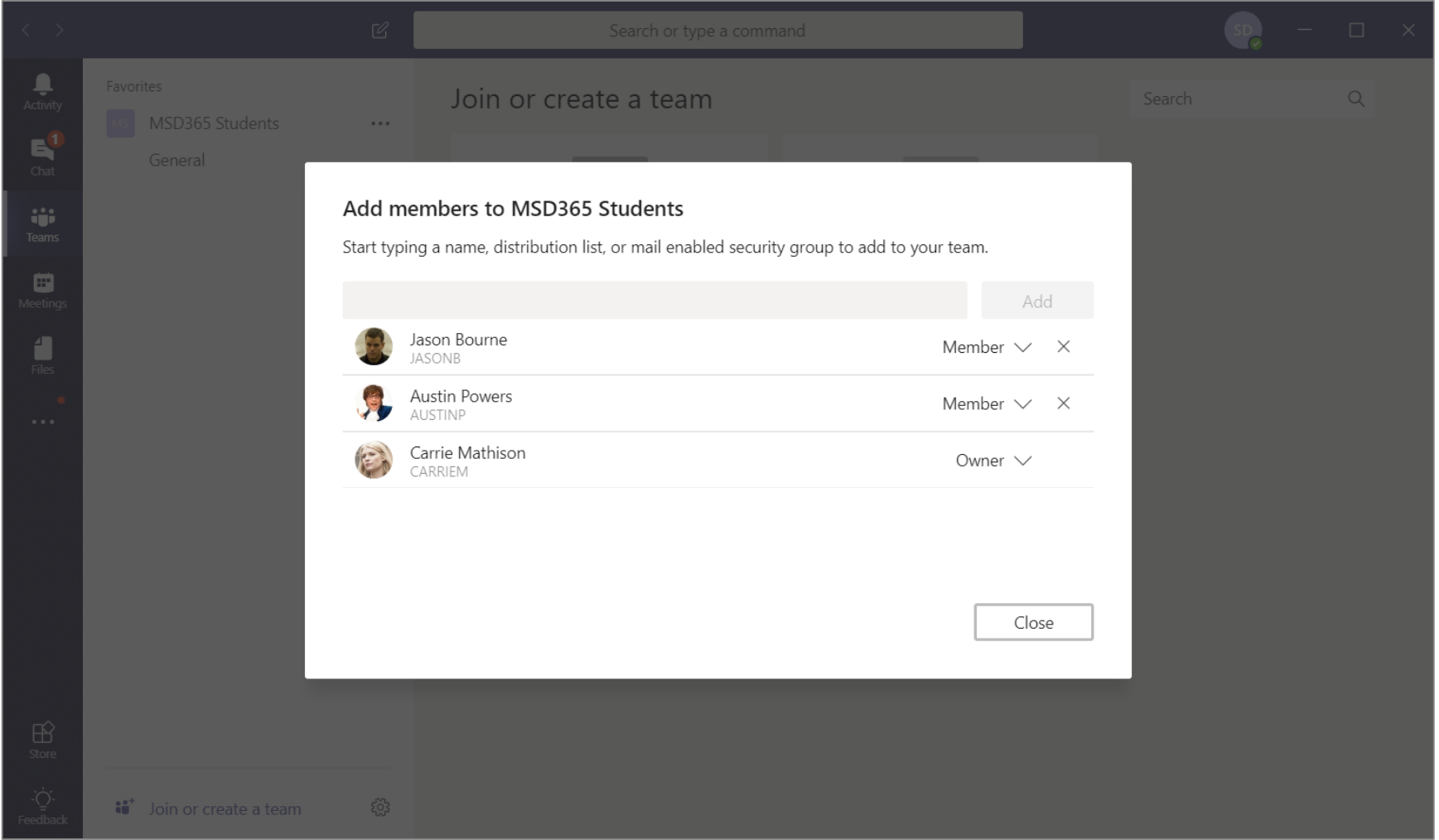
The screenshot shows the Microsoft Teams interface with a modal dialog box titled "Join or create a team". The dialog box contains the following fields and options:

- Team name:** A text input field containing "MSD365 Students" with a checkmark icon on the right.
- Description:** A text input field containing "A team for modern developers".
- Privacy:** A dropdown menu showing "Private - Only team owners can add members" with a downward arrow.
- Buttons:** "Cancel" and "Next" buttons at the bottom right.
- Link:** A link "Create a team from an existing Office 365 group" below the privacy dropdown.

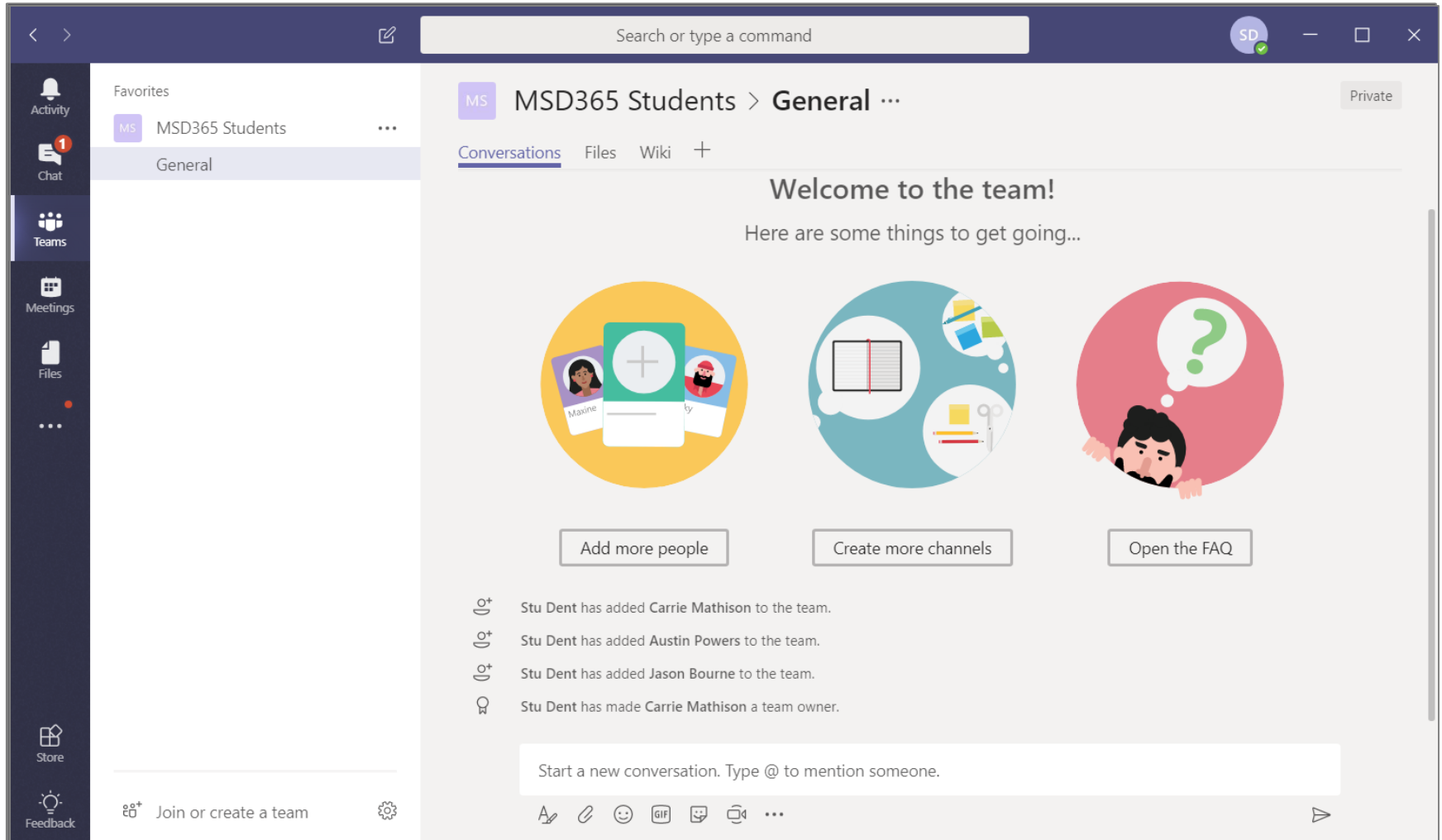
The background shows the Teams sidebar with icons for Activity, Chat, Teams, Meetings, Files, and Store, and a search bar at the top.



# Add Team Members

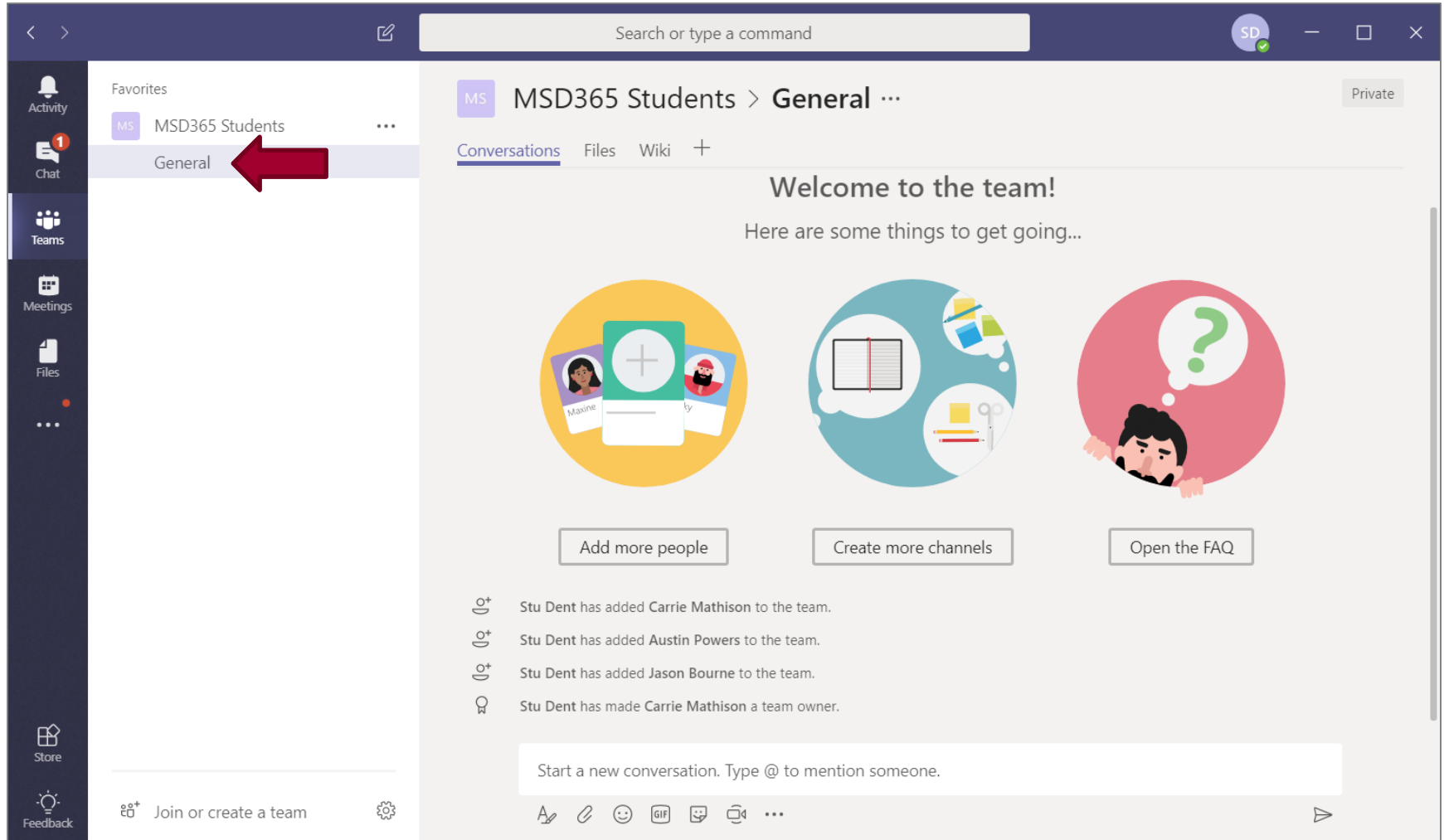


# A Team is Born



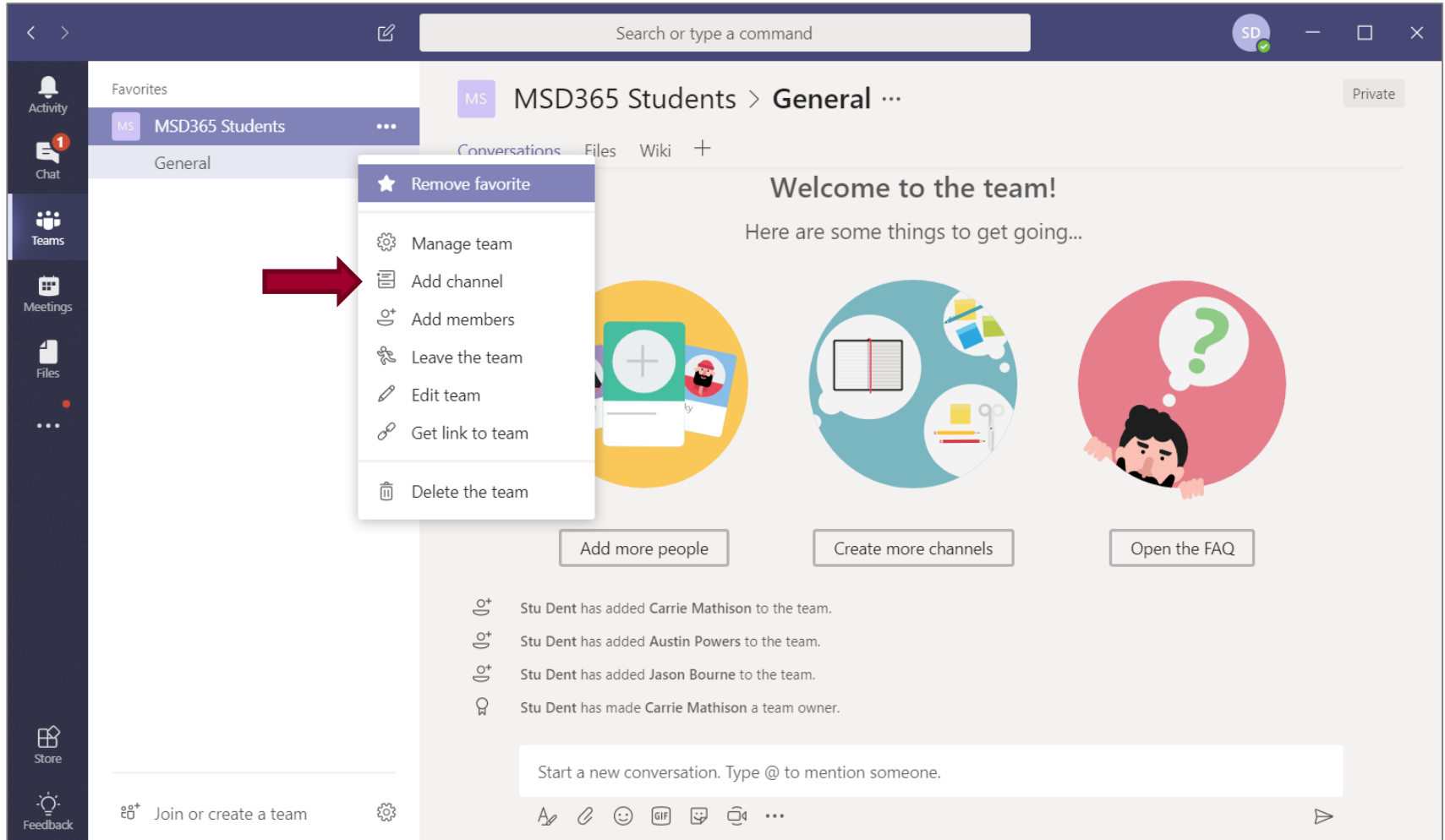


# New Team has General Channel by Default

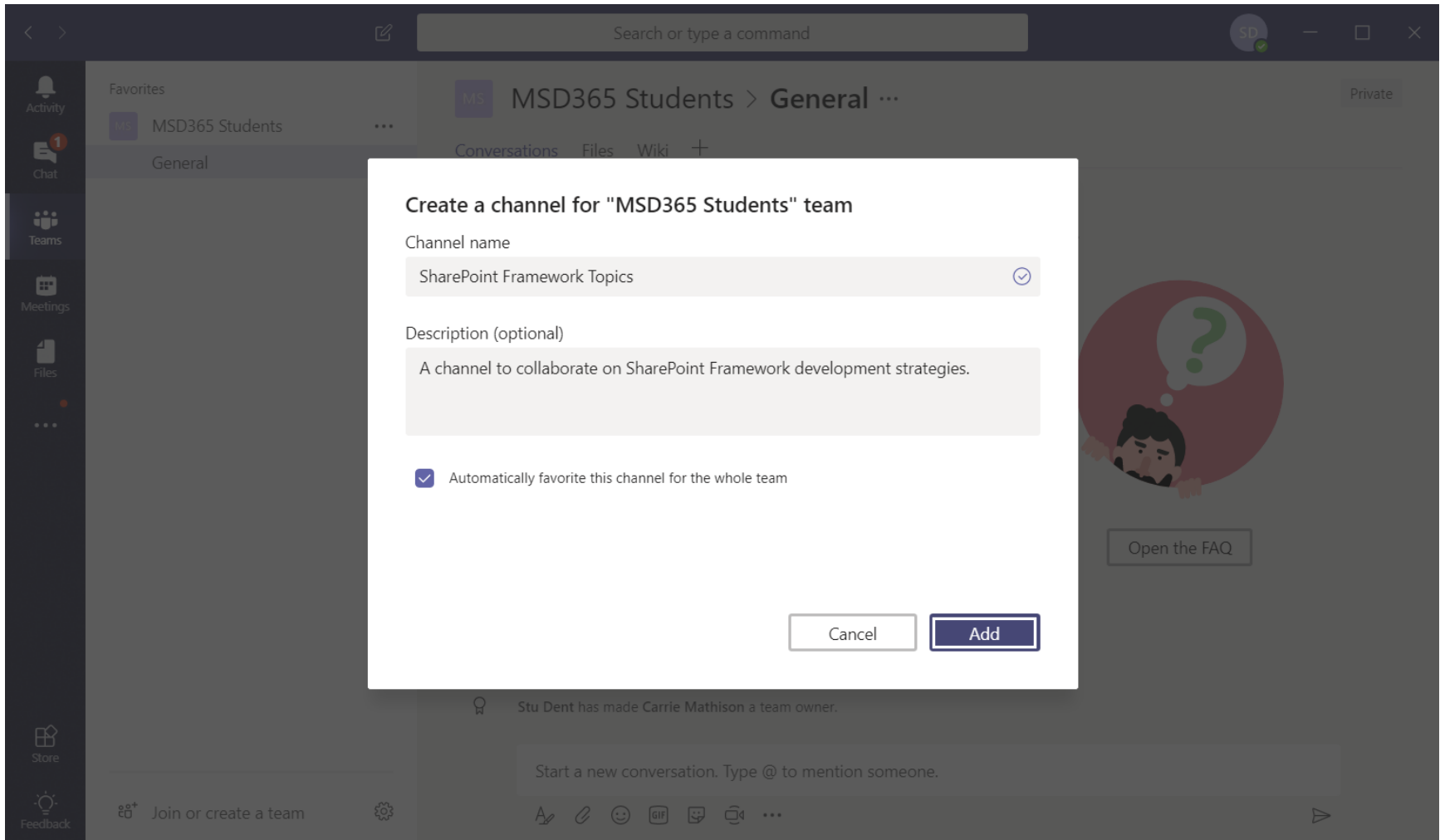




# Add a New Channel

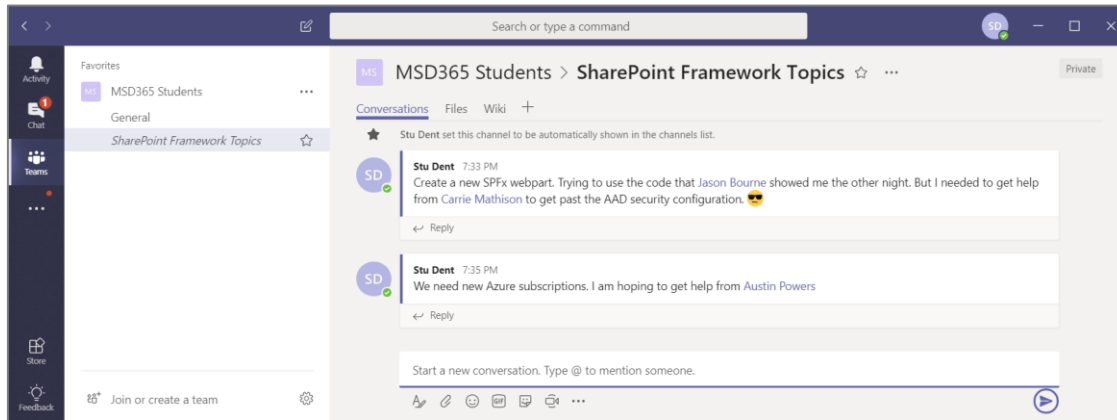


# Creating a New Channel

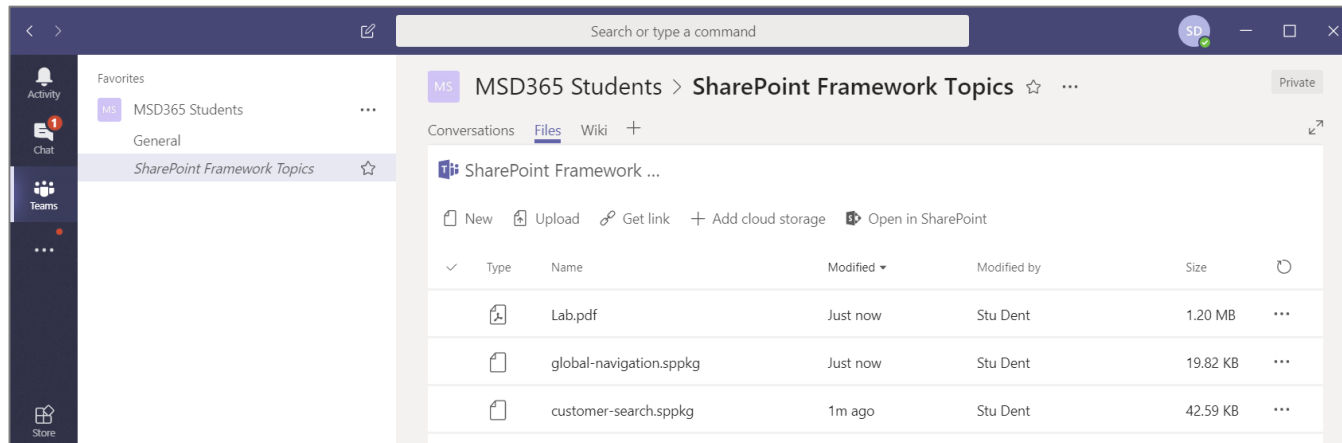


# What's Inside a Channel?

- Threaded conversations



- Files collection (i.e. document library)



# Adding Tabs

The screenshot shows the Microsoft Teams interface for the 'Northwind Traders' team, specifically the 'Development' channel. The left sidebar contains navigation options: Activity, Chat, Teams, Meetings, Files, and Apps. The 'Teams' section is expanded, showing the 'Northwind Traders' team with channels: General, Customer Accounts, Development (selected), Marketing, Project Management, Website, and 32 more channels. Below this, there are sections for 'Contoso', 'Engineering', 'Fun stuff', and 'Marketing'. The main content area is titled 'Northwind Traders > Development' and includes tabs for Conversations, Files, Notes, Feature Backlog (selected), Deployments, and Pull Requests. The 'Feature Backlog' tab is active, displaying a list of tasks organized into three columns: 'Future Features', 'Backlog', and 'Required for Open Beta'. Each task card includes a progress bar, a title, a description, a checklist, a user avatar, and a status indicator (e.g., 'LC', 'CD', 'DF').

**Future Features**

- Add Customizer config for logo's background colour (1/3)
- Sticker face tracking issue with Video Calls (1/3)
- Add power by Edify Hub to login page when using someone else's branding (2)
- Sermons page and widgets - sync with SermonAudio (1/3)

**Backlog**

- Add Customizer config for logo's background colour (4)
- Zendesk SSO (2/3)
- Admin notifications when info changes - with option to update info block (1)
- Set up archive of logs. (1/3)
- Upgrade OpenSSL (after Jul. 9) (1/3)
- Add ability to force password reset (4)
- Update login page to use new branding and styling (4)

**Required for Open Beta**

- Info.edifyhub.com, default to http instead of https. (2)
- Support "mark inactive" (2/3)
- Add keyboard shortcuts (1)
- Test Accounts Server (1)
- Want tips, usage examples, or API info? (2)
- CMS-2345: Categorize more of the unknown locations (2)

**Add a list**

# Microsoft Teams Development Platform

- Development features available today

 **Tabs** - surface rich content within Teams

 **Bots** - help users get tasks done in conversations

 **Connectors** - post rich updates to channels

- Development features in preview

 **Actionable Messages** - add rich interaction to connector cards

 **Activity Feed** - engage users via feed notifications

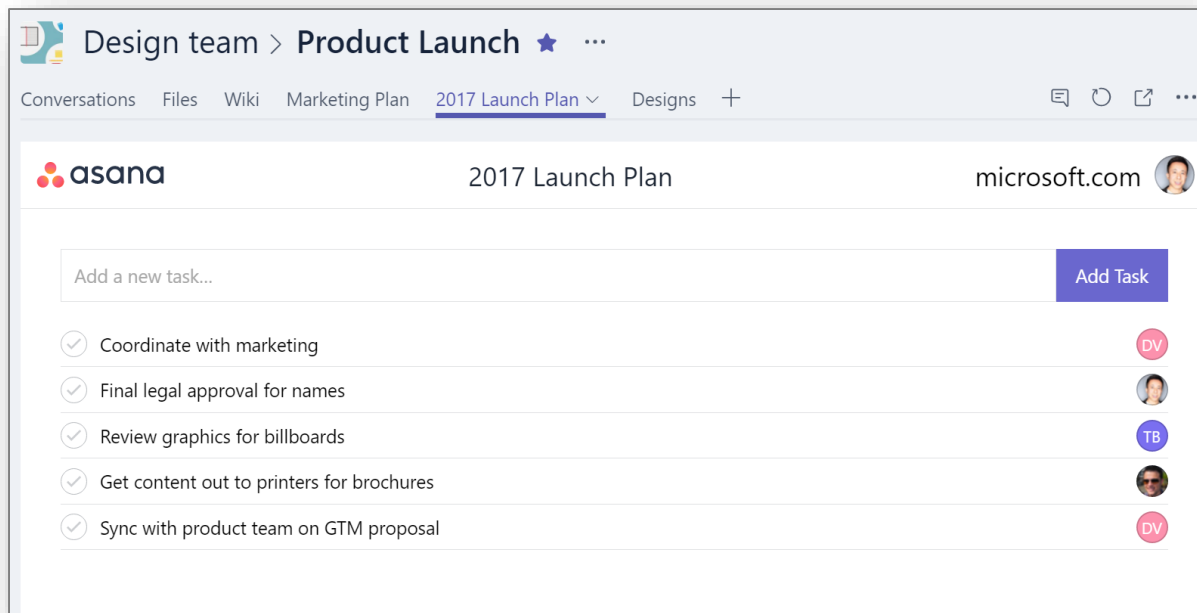
 **Compose Extensions** - users query & share cards in conversations

 **Office Store** - drive engagement by submitting app to Office Store



# Developing Tabs

- Rich surface area for your app
- As simple as hosting your existing web app
- Team productivity: within channels
- Personal productivity: app flyout
- Your services, experience, & users



# Tabs

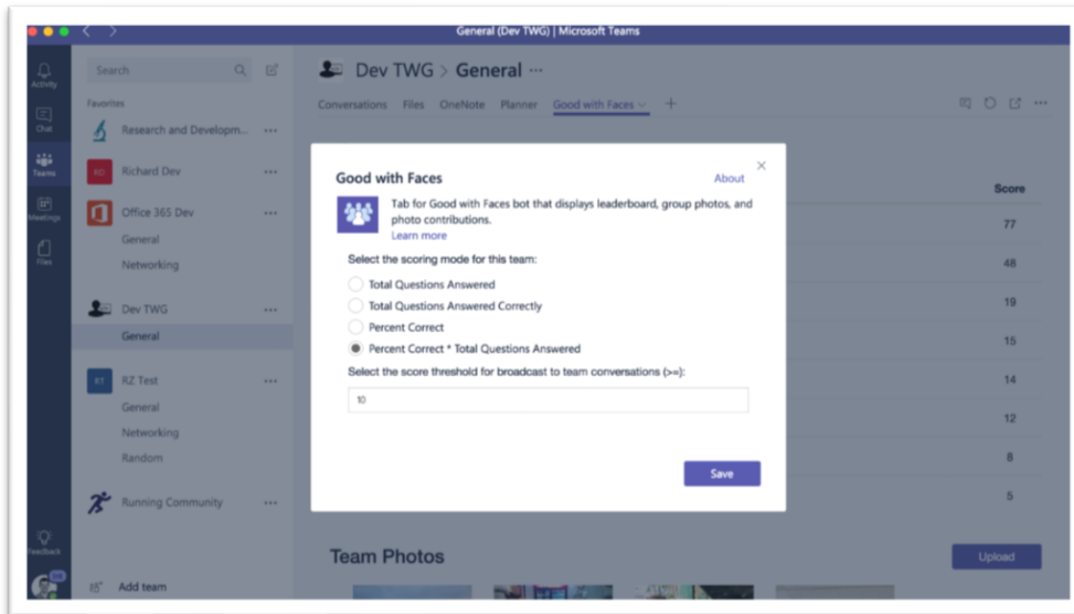
- Embed-ready web UX on web/desktop, deep link to native apps on mobile
- Leverages Microsoft Teams Tab Library javascript for context and interactivity
- Team or Personal Scope





# Tabs Pages

- Configuration Page
  - used to add/update tab and set content page
- Content Page
  - the primary page displayed in the tab
- Remove Page
  - optional page displayed when a tab is removed



# Tabs - Config Page

- Used to Add or Edit a Tab
  - Must reference Microsoft Teams Tab Library
  - Sets the **contentUrl** for the tab
  - Calls **setSettings** to save tab configuration
  - Must call **setValidityState** to enable the save button
  - Call **notifySuccess** when saved

```
microsoftTeams.initialize();
microsoftTeams.settings.registerOnSaveHandler(function(saveEvent){
    microsoftTeams.settings.setSettings({
        entityId: "someid",
        contentUrl: "https://somedomain/tab.html",
        suggestedDisplayName: "Tab Title",
        websiteUrl: "https://somedomain/info.html",
        removeUrl: "https://somedomain/tabremove.html"
    });
    saveEvent.notifySuccess();
});

function onSaveClick() {
    microsoftTeams.settings.setValidityState(true);
}
```



# Tabs - Context information

- getContext returns team/channel context:
  - tid – id of tenant
  - groupId – id of the O365 group
  - teamId – id of the team
  - channelId – id of the channel
  - entityId – id of the entity
  - theme – theme of the teams client
  - locale – the locale of the client
  - upn – UPN of the user

```
microsoftTeams.getContext(function (context) {  
    // context contains all context info  
});
```



# Tabs - Deep linking to Entities

- Share a Link from your Tab
  - Trigger creation of deep link to specific Entity for user to copy and paste into a conversation
  - Resultant link may be clicked on by any user to launch the tab and navigate directly to that Entity
  - Programmatically create your own link, for use via bot or other code flows

```
microsoftTeams.shareDeepLink({  
  subEntityId: <subEntityId>,  
  subEntityLabel: <subEntityLabel>,  
  subEntityWebUrl: <subEntityWebUrl>  
})
```

```
https://teams.microsoft.com/l/entity/<appId>/<entityId>?webUrl=<entityWebUrl>&  
label=<entityLabel>&  
context=<context>
```



6

Activity

Chat

Teams

Meetings

Files

Apps

Feedback

Microsoft Teams

Search

Favorites

Northwind Traders

...

General

Customer Accounts

Development

Marketing

Project Management

Website

32 more channels

Contoso

...

Engineering

General

Fun stuff

Marketing

Weekly Specials

4 more channels

More

+

Add team

Northwind Traders > Development

★ ...

Conversations

Files

Notes

Feature Backlog

Deployments

Pull Requests

+

Find

CONTOSO

Future Features

Backlog

Required for Open Beta

Add a list

Test bug

in list Project Tasks

Labels

mobile

ios

+

Edit the description...

Write a comment

@

send

Activity

Will Little added this card to Project Tasks Jan 31 at 1:08 pm

Hide Details

Add

Members

Labels

Checklist

Due Date

Attachment

Power-Ups

Github

Actions

Move

Copy

Subscribe

Archive

Add a card...

Add a card...

Add a card...



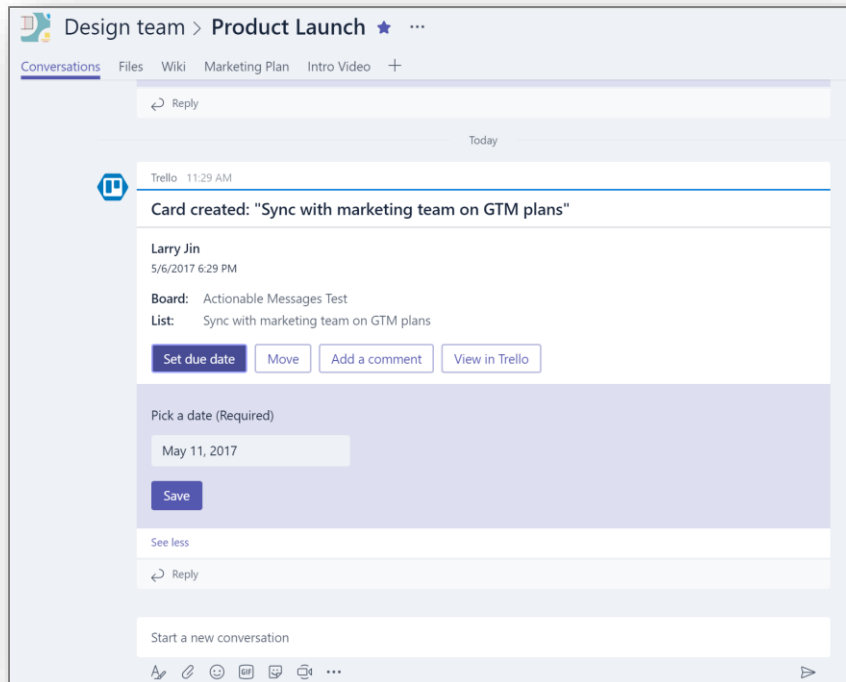
# Building a Great Tabs experience

- Configurable Tabs
  - In channel, tabs allow user to configure view to relevant content
  - Content should be the same for all users – leverage collaboration
  - Content should be locked – don't allow users to browse away
- Static Tabs
  - In personal scope, this content should be the relevant for either the all-up app (e.g. Help, FAQ, forums) or for a personal view of the experience



# Connectors with Actionable Messages

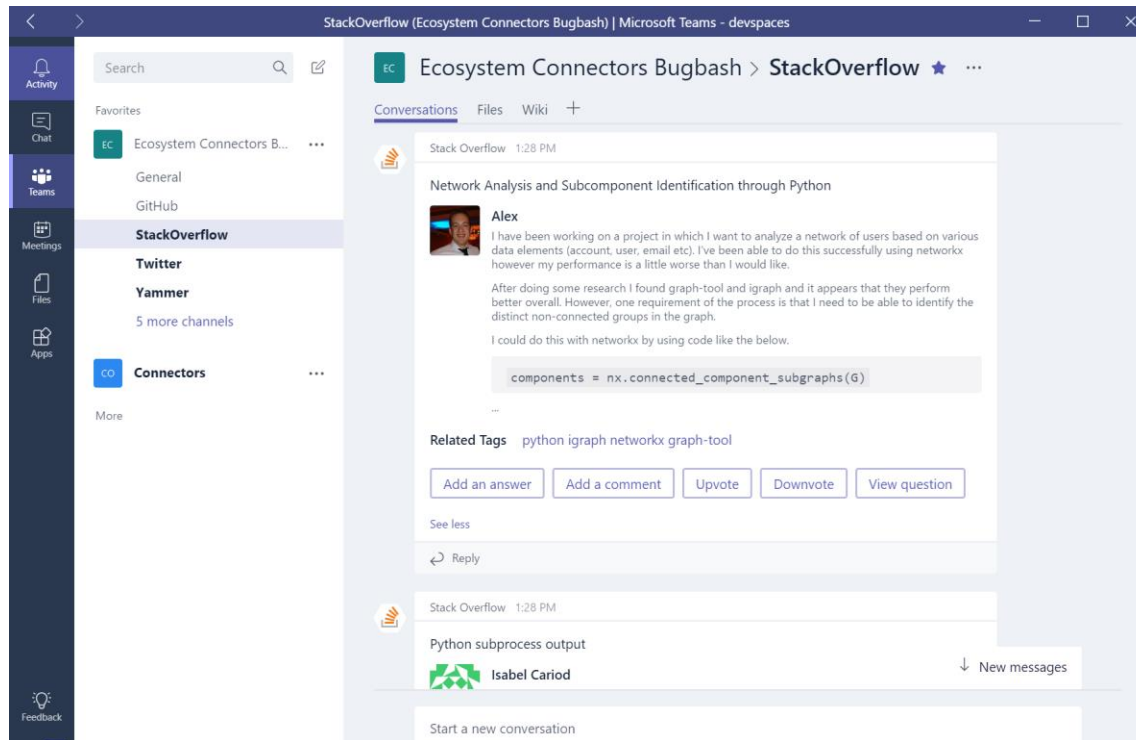
- Push rich interactive cards into channels
- Fully supported in both Teams and Outlook
- Users can take quick actions (e.g. comment or set a date)
- Uses incoming webhook API





# Office 365 Connectors

- Leverage webhooks to send activity into Teams
  - Build connector, registering on the Connector Developer Portal
  - Simply add Connector ID to manifest.json
  - Leverage Actionable Messages to allow inline work commands





6

Activity

Chat

Teams

Meetings

Files

Apps

Feedback

Microsoft Teams

Search

Feed

DF

Daichi Fukuda

Posted a new reply on your file.

NewDesign-Proposal.pptx

1m ago

MB

Marie Beaudouin

Replied to you in Website

Yes, but not all the concerns have been addressed.

25m ago

Contoso

Cassandra Dunn assigned you to a task

CMS-2346: Categorize more of the unknown locations

30m ago

Babek Shammass

Mentioned you in Marketing

I can get on board with that. Thanks for taking a look! Would you mind...

2h ago

Contoso

Daichi Fukuda sent you a pull request.

Your response is required

3h ago

Will Little

Invited you to attend

11:30am 1 hour

Social Calendar Client Review

Accept

Decline

4h ago

IA

Ivan Averyanov

Reacted to your reply in Website

Northwind Traders > Development

Conversations Files Notes Feature Backlog Deployments Pull Requests

Find

Future Features

Backlog

Required for Open Beta

Add a list

CMS 2346: Categorize more of the unknown locations

in list Project Tasks

Labels

mobile ios

Edit the description...

Write a comment

send

Activity

Cassandra Dunn assigned you to the task 2/24 at 1:00 pm

Babek Shammass created the task 2/3 at 10:32 am

Hide Details

Add

Members

Labels

Checklist

Due Date

Attachment

Power-Ups

Github

Actions

Move

Copy

Subscribe

Archive

Contoso

Activity 6

Chat

Teams

Meetings

Files

Apps

Feedback

Search

Feed

Daichi Fukuda  
Posted a new reply on your file.  
NewDesign-Proposal.pptx  
1m ago

Marie Beaudouin  
Replied to you in Website  
Yes, but not all the concerns have been addressed.  
25m ago

Contoso  
Cassandra Dunn assigned you to a task  
CMS-2346: Categorize more of the unknown locations  
30m ago

Babek Shammass  
Mentioned you in Marketing  
I can get on board with that. Thanks for taking a look! Would you mind...  
2h ago

Contoso  
Daichi Fukuda sent you a pull request.  
Your response is required  
3h ago

Will Little  
Invited you to attend  
11:30am 1 hour  
Social Calendar Client Review  
Accept Decline  
4h ago

Ivan Averyanov  
Reacted to your reply in Website

Northwind Traders > Development

Conversations Files Notes Feature Backlog Deployments Pull Requests

App v 5.6 / Northwind Traders

Pull requests

Filter by: Open Author Target branch I'm reviewing

Summary Reviewers Builds

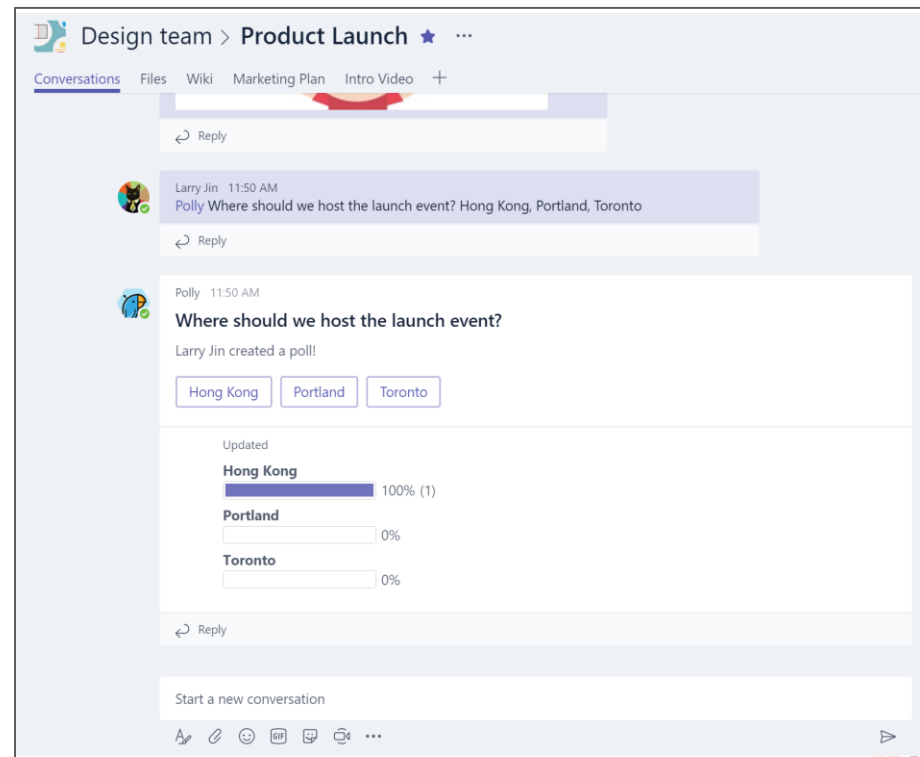
Bugfix/CMS-8876B cart is crashing → master  
Daichi Fukuda, last updated February 23 2017

Bugfix/CMS-5468C refresh issues → master  
Daichi Fukuda, last updated February 22 2017

Bugfix/CMS-4453 Color picker for widget selector changes → master  
Daichi Fukuda, last updated February 20 2017

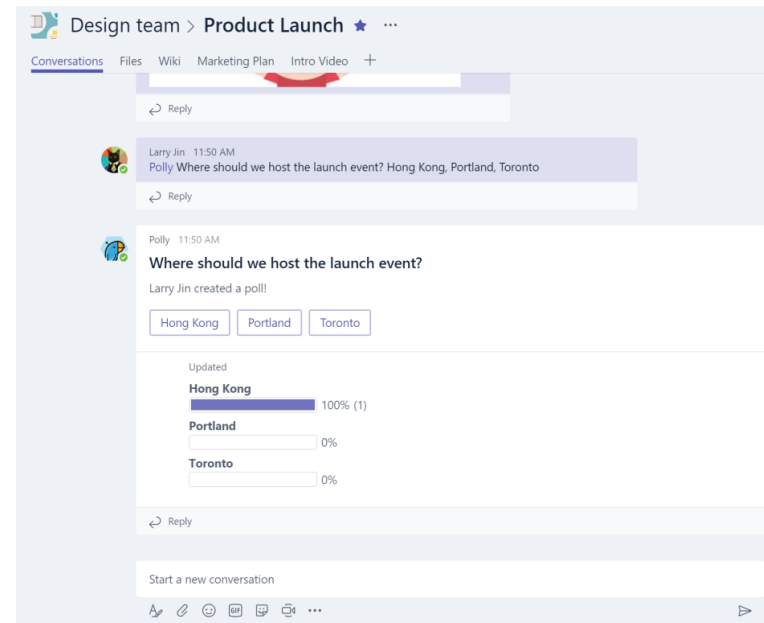
# Bots

- Bot is app with automated interaction in Team with conversations or 1:1
- Built using Microsoft Bot Framework
- Complete tasks via basic commands, menu or natural language
- Rich Microsoft Teams features, e.g.
  - Input menus
  - Dynamic message updates
  - Integrate with tabs



# Bots

- Based on Web Service you create and host
  - Use whatever web technology you prefer
  - Extra support for using Node.js or .NET on Azure
- Built using Microsoft Bot Framework
  - Call REST APIs directly or use libraries for Node.js and .NET
  - New Microsoft Teams Extension libraries (Node.js and .NET) provide channel-specific functionality



# Bots – Getting up and running

- Set up basic development environment
  - Install Bot Builder
  - Install Microsoft Teams extension SDKs for chosen language
- Leverage existing samples
  - Bot Framework sample
  - Teams Getting Started sample
- Run locally to test in the Bot Emulator





# Bots - Getting Teams-ready

- Register your bot in the Bot Framework Dev Portal
  - Set a unique bot name and handle
  - Set your bot Endpoint – your hosted domain, or your debug instance (we recommend Ngrok)
  - Create an AppID (aka BotID) and Password
  - Add Microsoft Teams as a Channel
  - Note: for 1:1 bot, you may reference your bot directly via its AppID
  - Enter your AppID and Password into your Development environment
  - Build and publish or run locally with Ngrok



# Bots - Accessing in Teams

- Create a Teams App Manifest
  - Microsoft provides SimpleBot package
  - SimpleBot has manifest with sample icons
  - Add your BotID in the Bots object
  - Sideload into a Team

```
{
  "manifestVersion": "1.0",
  "version": "1.0.0",
  "id": "%APP_GUID%",
  "packageName": "com.samples.samplebot",
  "developer": {
    "name": "Microsoft",
    "websiteUrl": "https://www.microsoft.com",
    "privacyUrl": "https://www.microsoft.com/privacy",
    "termsOfUseUrl": "https://www.microsoft.com/termsofuse"
  },
  "name": {
    "short": "Sample Bot App"
  },
  "description": {
    "short": "This is a sample bot manifest",
    "full": "This sample bot manifest has minimum info."
  },
  "icons": {
    "outline": "contoso20x20.png",
    "color": "contoso96x96.png"
  },
  "accentColor": "#60A18E",
  "bots": [
    {
      "botId": "%BOT_FRAMEWORK_ID%",
      "scopes": [
        "team",
        "personal"
      ]
    }
  ],
  "permissions": [
    "identity",
    "messageTeamMembers"
  ]
}
```



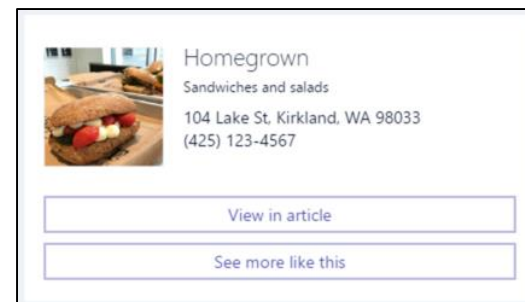
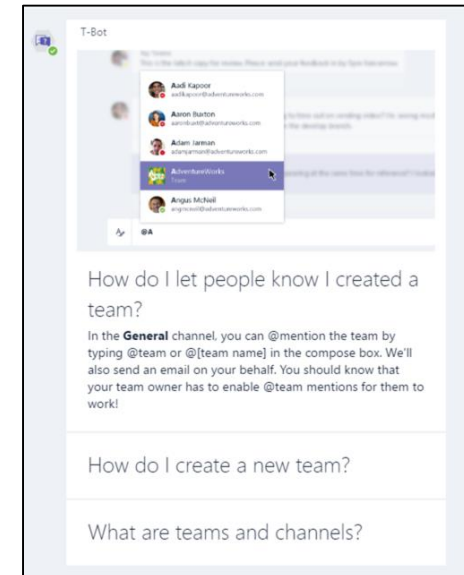
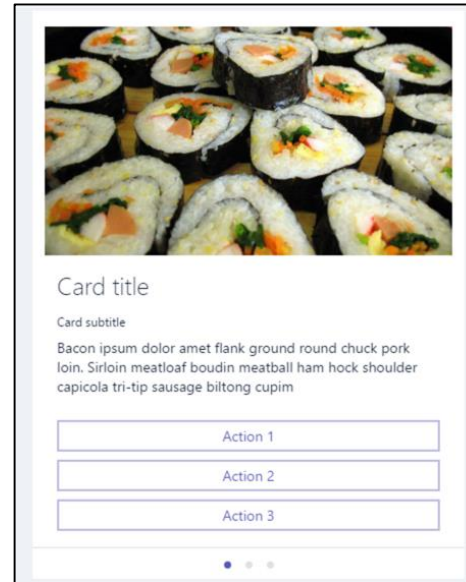
# Bot - Messaging

- Responsive messages (via 1:1 or @mentioning in channel)
  - Proactively create messages
  - Create direct (1:1) message – must know user ID
  - Create new channel reply – bot must be added to team and know channel information
- Update messages sent by Bot
  - One-way only – e.g. push notifications



# Bot - Content

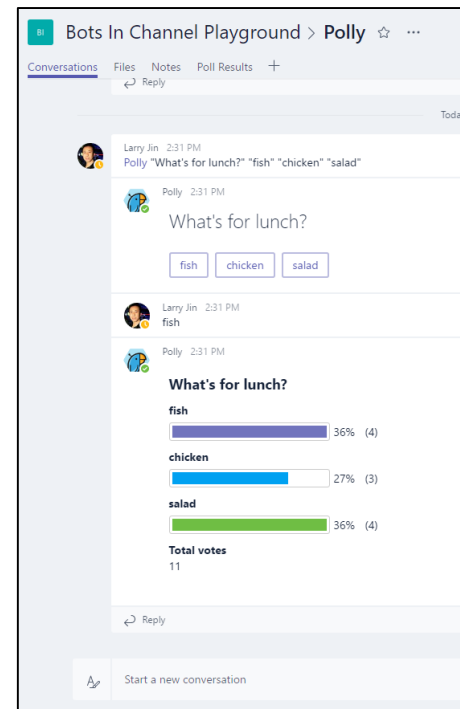
- User to bot
  - Plain or rich text
  - Inline images
  - Note: files not currently supported
- Bot to user
  - Plain text
  - Markdown or XML
  - Cards
  - Hero Cards
  - Thumbnails
  - Sign-in
  - Carousel or List layouts
  - Buttons
  - openUrl – open in browser
  - imBack – trigger chat response
  - invoke – silent JSON payload



# Bots in Team Channels - Basics

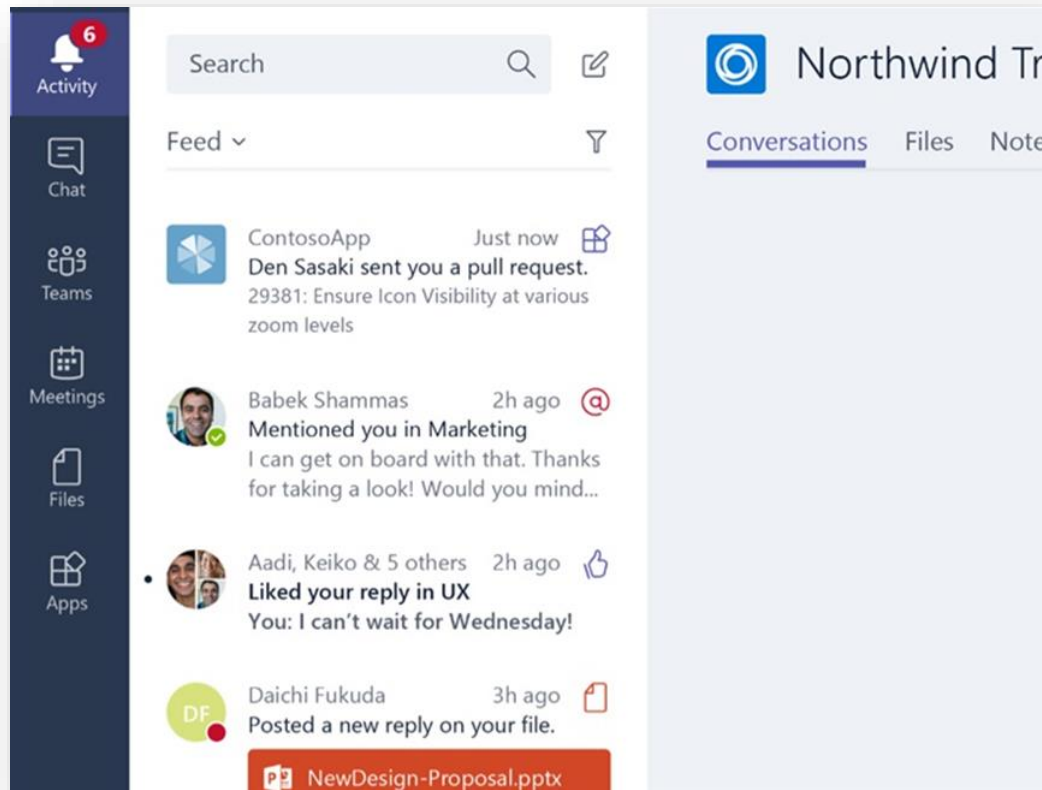
- Bot only listens to @mentions
  - Replies go to the conversation thread
  - Inbound payload in channels as flagged as isGroup and contains Teams channel data:

```
"channelData": {  
  "channel": {  
    "id": "19:293ecdb923ac4458a5c23661b505fc84@thread.skype",  
  },  
  "team": {  
    "id": "19:293ecdb923ac4458a5c23661b505fc84@thread.skype"  
  },  
  "tenant": {  
    "id": "72f988bf-86f1-41af-91ab-2d7cd011db47"  
  }  
}
```



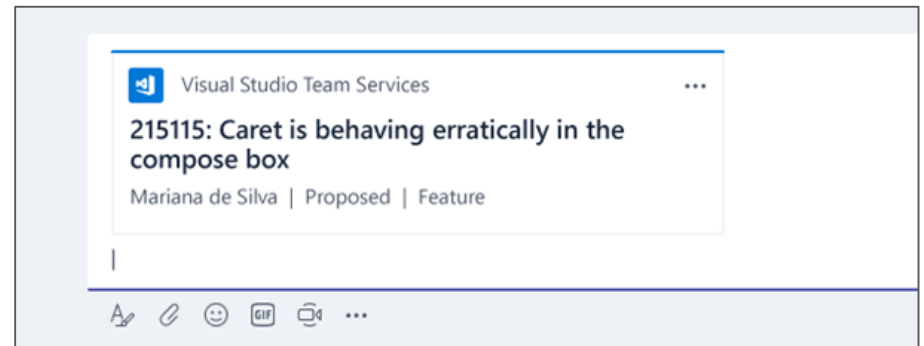
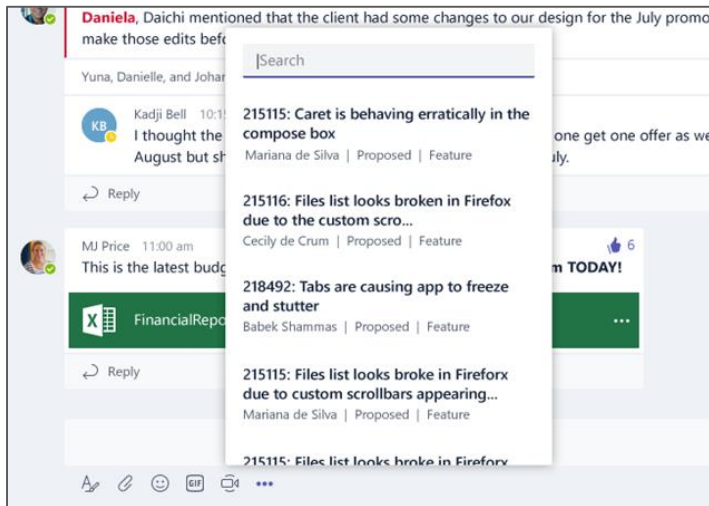
# Activity Feed

- Notify users personally via the activity feed
- Same API as sending bot messages
- Deep link straight into tabs



# Compose Extensions

- Personalize Teams Compose box with your App & Services content
- Users can query and insert your app content into conversations
- Example: Bug details from VSTS
- Example: Sales opportunities from CRM
- You can reuse services you built for Bots or Tabs





# Microsoft Teams App

- Microsoft Teams Apps
  - Deliver custom solution using solution package (.zip)
  - Single App definition – manifest.json
  - Single package used for Sideloading or Office Store



# Building a Great Microsoft Teams app

- Teams Scope - your experience exists in the team context
  - Team owners may add your experience to a team:
  - Bots and Compose Extensions – available on all channels
  - Teams Tabs (“Configurable”) – added and customized on a per-channel basis
- Personal Scope – your experience exists in individual user context
  - Individuals add the experiences they want for personal use
  - Bots and Compose Extensions – available in 1:1 chat
  - Personal Tabs (“Static”) – available to end users via the App bar or alongside 1:1 bots



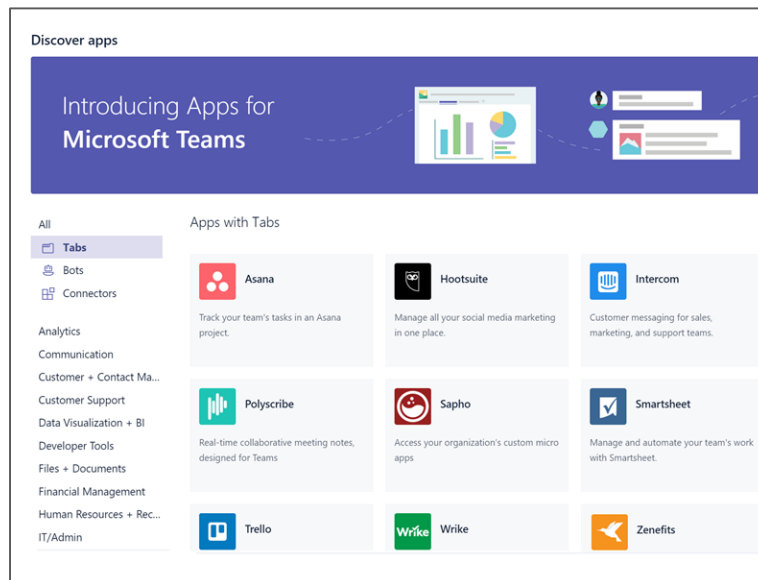
# Getting Started

- Tabs - Bring your own Service
  - JavaScript APIs to tie your experience to Teams context
- Connectors – Leverage the Connectors Developer Portal
  - Use new Actionable Cards layouts that work in Outlook and Teams
- Bots – Build using the Bot Framework
  - Use the .NET and Node.js SDKs, or call the REST endpoints
  - Teams Extension SDKs provide Teams-specific functionality
  - Flag important messages for inclusion in Activity Feed
- Compose Extensions – Reuse your bot
  - Use the .NET and Node.js SDKs, or call the REST endpoints
  - Teams Extension SDKs provide Teams-specific functionality



# Deploying

- ISVs – Distribute through the Office Store
  - Create or use your existing DevCenter/Seller Dashboard account
  - Upload your package and enter your product page metadata
  - Office Store Validation ensure compliance with Office Store Policy
  - Validated apps published to Office Store appear in Microsoft Teams Discovery view
- Line-of-Business apps – Sideload your package
  - Leverage the same testing methodology to deploy within your org



# Compose Extensions

- Service that listens to user requests and returns cards or rich media
  - Leverage the Bot you've already built, or build another
  - Define commands and parameters in the manifest.json
  - Handle via Bot Framework activity, leveraging the Microsoft Bot Extension SDKs

```
"composeExtensions": [  
  {  
    "botId": "57a3c29f-1fc5-4d97-a142-35bb662b7b23",  
    "scopes": [  
      "personal"  
    ],  
    "commands": [{  
      "id": "searchCmd",  
      "description": "Search Bing for information on the web",  
      "title": "Search",  
      "initialRun": "true",  
      "parameters": [{  
        "name": "searchKeyword",  
        "description": "Enter your search keywords",  
        "title": "Keywords"  
      }]  
    }  
  ]  
},  
],
```



# Building a Compose Extension Experience

- Compose Extensions in Microsoft Teams
  - Share content that makes sense in conversations.
  - Add on as part of core Bot offering to facilitate easier management.
  - Take advantage of the rich card types provided as part of the platform.
  - Optimize search results to keep compose extensions snappy and responsive.



[Conversations](#)
[Files](#)
[Notes](#)
[Feature Backlog](#)
[Deployments](#)
[Pull Requests](#)
[+](#)

IT Service Desk | 764CVC

Type: **Fault** Components: **Extranet** Status: **In Progress**  
Priority: **Major** Labels: **None**

Relates to: [762CHJ](#) Scroll performance

[Show full summary](#)

Contoso 12:30 pm  
The following item was marked as closed. **Shiproom**

## IT Service Desk | 884CVC

**Details**

Type:	<b>Fault</b>	Components:	<b>Extranet</b>	Status:	<b>Closed</b>
Priority:	<b>Major</b>	Labels:	<b>None</b>		

Relates to: **884CVB** Users can't locate files

[Show full summary](#)

[Start a new conversation](#)

# Activity Feed

- Use the Activity Feed to send important information to your users
- Leverage existing Bot to tag specific messages as notification alerts
- Encode deep link to entity (tab) for one-click navigation to content

