

Application Types

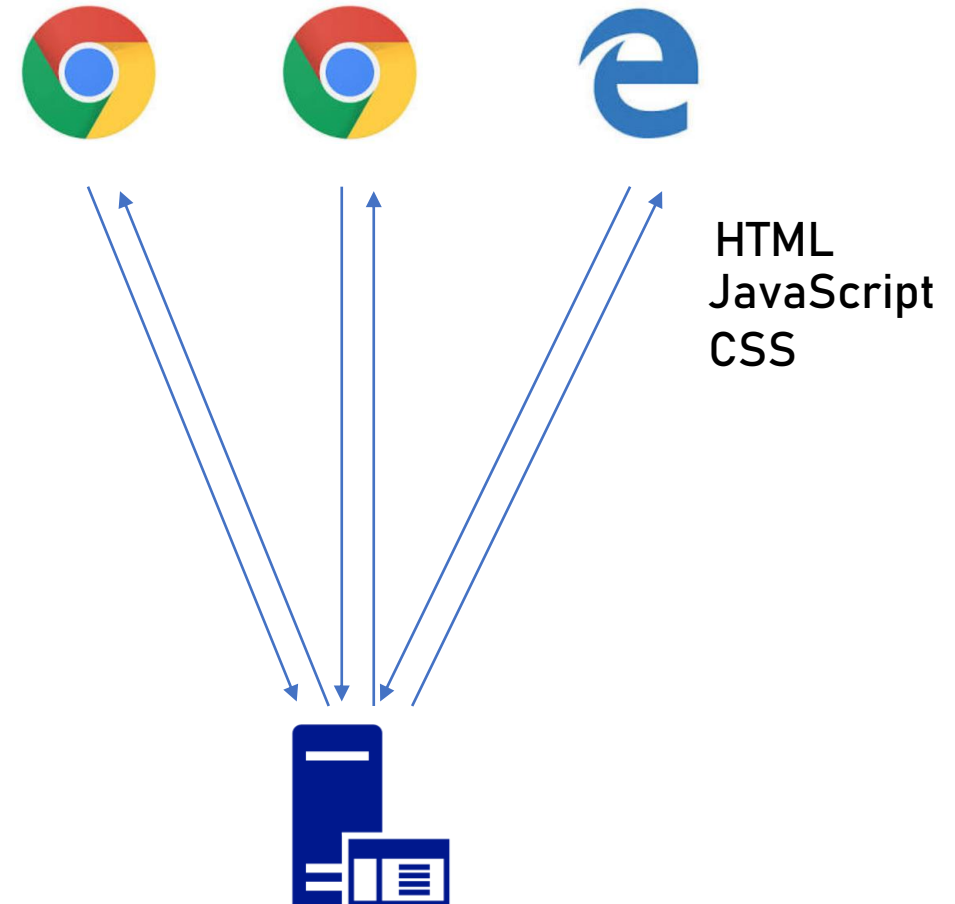
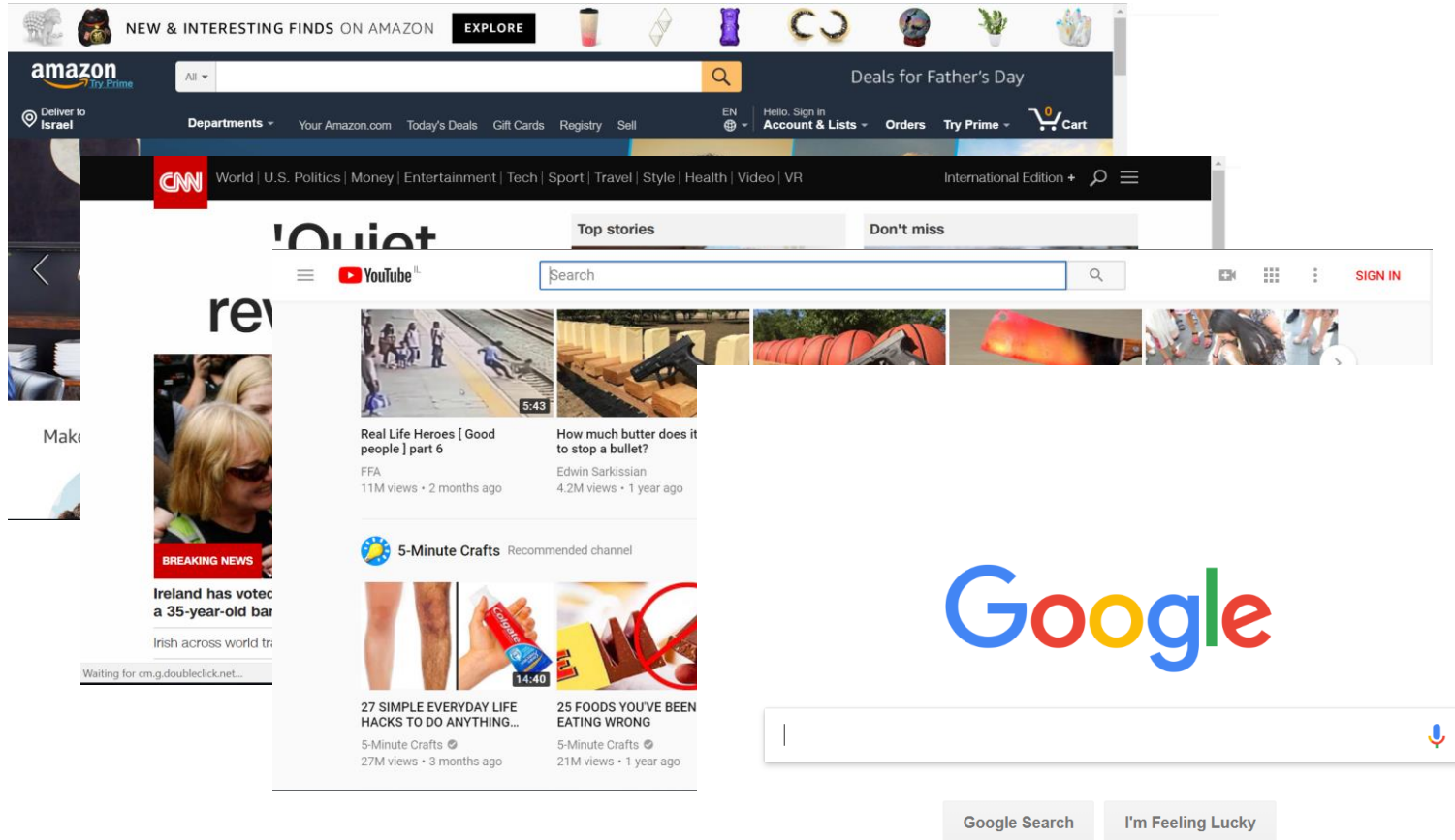
Memilavi
www.memilavi.com



Application Types

- Decided after requirements are set
- Common Types:
 - Web Apps
 - Web API
 - Mobile
 - Console
 - Service
 - Desktop

Web Apps



Web Apps

- Best suited for systems that require:
 - User Interface
 - User Initiated Actions
 - Large Scale
 - Short, Focused Actions
- Request-Response based

Web API

DeveloperUse casesProductsDocsMore

Apply

Sign In

facebook for developers

DocsToolsSupport

Search developers.facebook.com

Bas

Acc

Two

Post, Twee

Get T

Curat

Optin

Search

Filter

Graph AP

Overview

Using th

Graph E

Comm

Handlin

Debugg

Using C

FAQ

Referenc

Commor

Other AP

Webhool

Advance

Changek

Server-S

Google

Search for services, methods, and recent requests...

Loading...

Microsoft

Developer Network

Dev Center

Explore

Get Started

Code Samples

Resources

Documentation

Table of contents

Introduction

Authenticating apps

Accessing and syncing data

Integrating non-Microsoft file types

Making your app easy to use and find

OneNote

Office 365 REST API reference

Office 365 APIs

Use the Outlook REST API

Batch Outlook REST requests

Outlook Mail

Last Updated: 5/25/2018

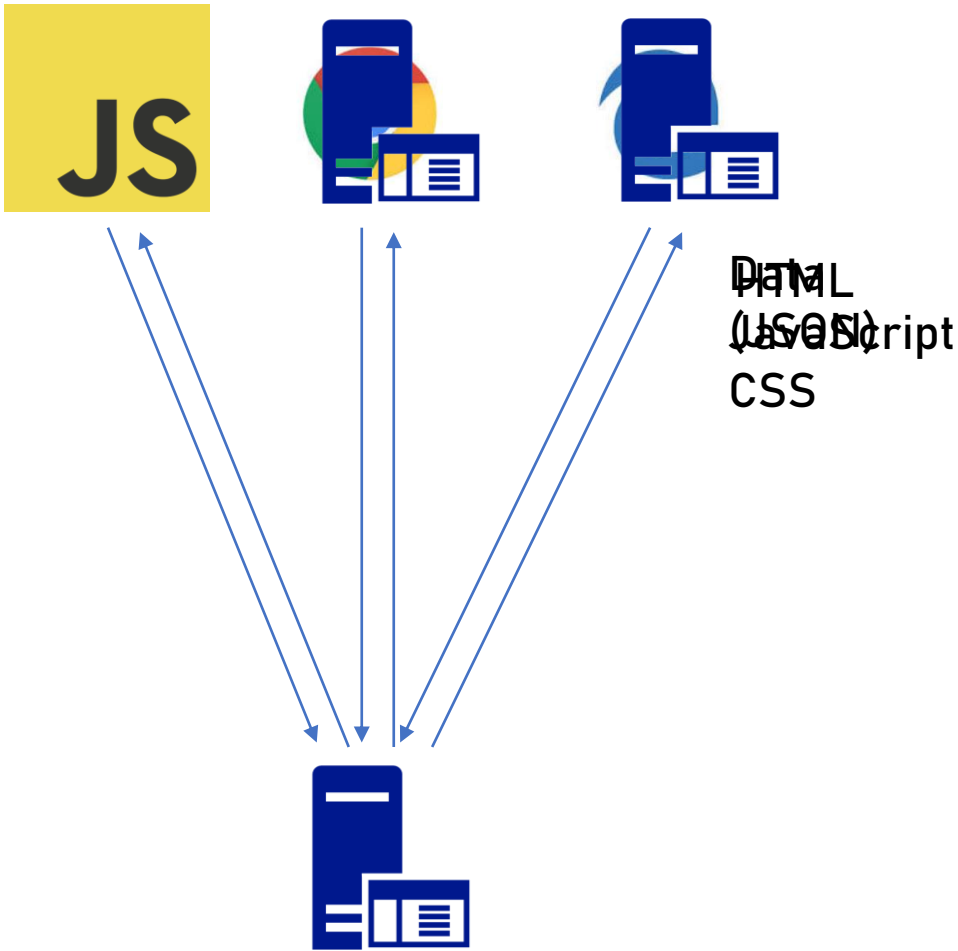
Office 365 API reference

Applies to: Exchange Online | Office 365 | OneDrive for Business

Office 365 API functionality is also available through the Microsoft Graph, a unified API that includes APIs from other Microsoft services such as Outlook, OneDrive, OneNote, Planner, and Office Graph, accessible through a single endpoint and with a single access token. We recommend using the Microsoft Graph in your apps when possible.

The Office 365 APIs enable you to provide access to your customer's Office 365 data, including the things they care about most--their mail, calendars, contacts, users and groups, files, and folders--all right from within your app itself.

You can access the Office 365 APIs from solutions across all mobile, web, and desktop platforms. No matter your development platform or tools. So whether you're building web applications using .NET, PHP, Java, Python, or Ruby on Rails, or creating apps for Windows Universal Apps, iOS, Android, or on another device platform, it's your choice.



Web API

- Usually REST based
- Combination of:
 - URL (`https://www.mysite.com/api/orders`)
 - Parameters (`date=10/10/2017`)
 - HTTP Verb (`GET`)

Web API

- Examples:

GET <https://www.mysite.com/api/users/17>

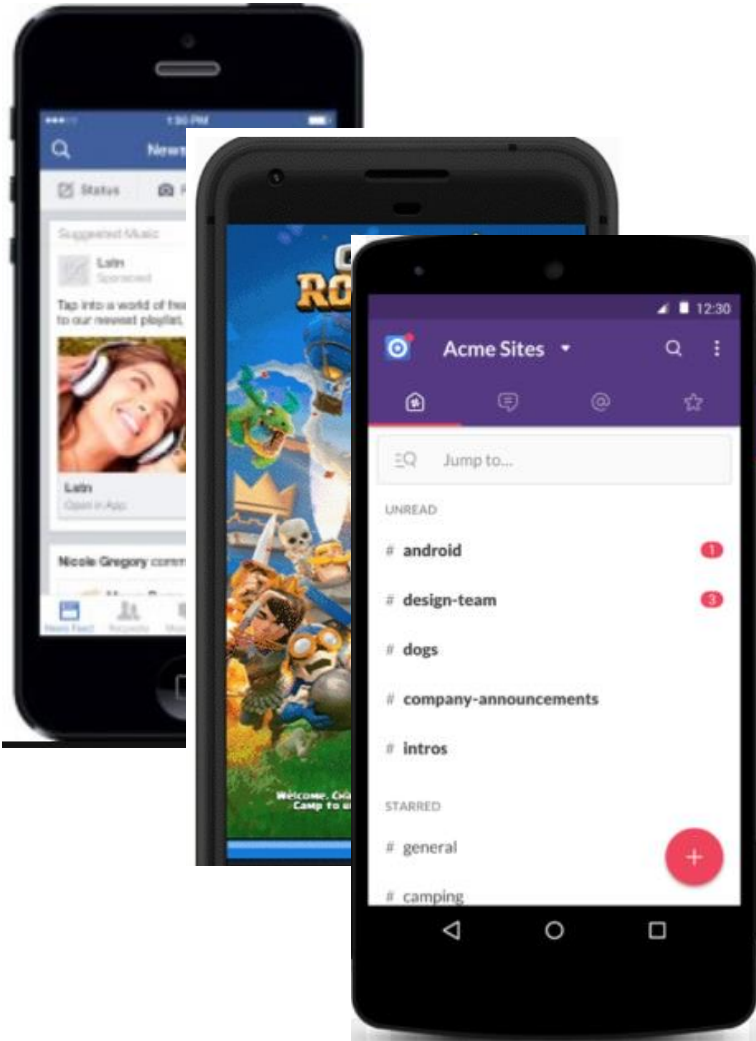
DELETE <https://www.mysite.com/api/orders/156>

- Returns Data, not HTML
- Very Accessible

Web API

- Best suited for systems that require:
 - Data Retrieval and Store
 - Client Initiated Actions
 - Large Scale
 - Short, Focused Actions
- Request-Response based

Mobile Apps



- Run on, well, mobile
- Usually works with Web API

Mobile Apps

- Best suited for systems that require:
 - User Interaction (games, social apps)
 - Front End for Web API
 - Location

Console Applications

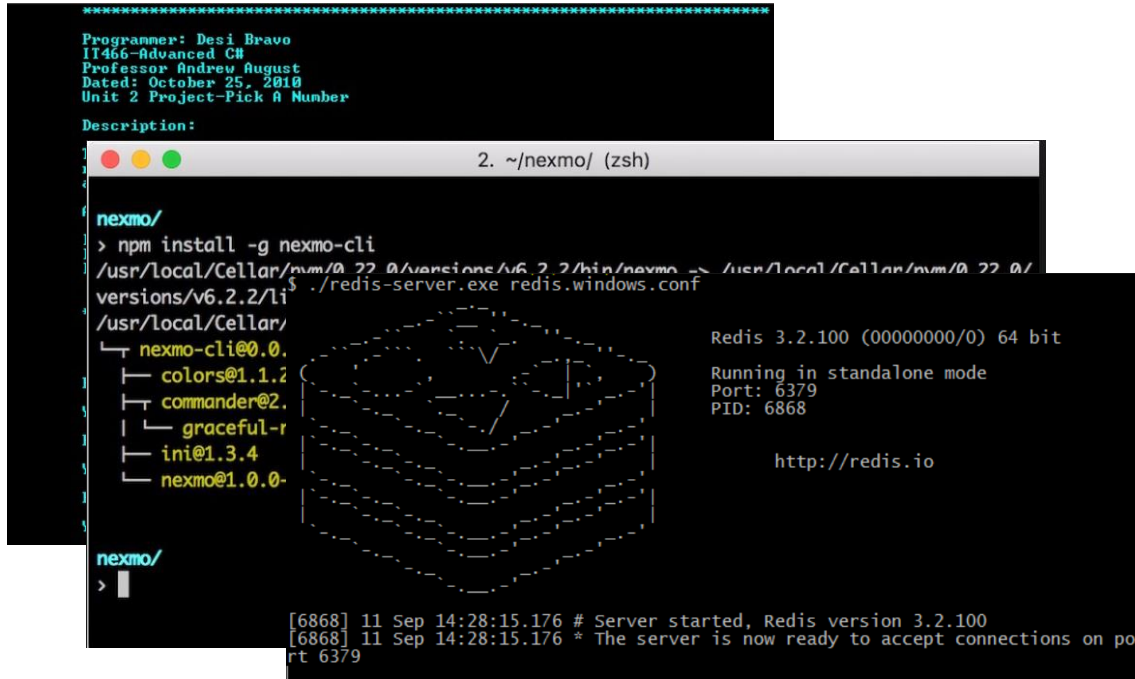
```
*****
Programmer: Desi Bravo
IT466-Advanced C#
Professor Andrew August
Dated: October 25, 2018
Unit 2 Project-Pick A Number
Description:

2. ~/nexmo/ (zsh)

nexmo/
> npm install -g nexmo-cli
/usr/local/Cellar/nvm/0.22.0/versions/v6.2.2/bin/nexmo -> /usr/local/Cellar/nvm/0.22.0/
versions/v6.2.2/lib
/usr/local/Cellar/
├─ nexmo-cli@0.0.0
├─ colors@1.1.2
├─ commander@2.
├─ graceful-r
├─ ini@1.3.4
├─ nexmo@1.0.0-
└─

nexmo/
>

[6868] 11 Sep 14:28:15.176 # Server started, Redis version 3.2.100
[6868] 11 Sep 14:28:15.176 * The server is now ready to accept connections on po
rt 6379
```



- No Fancy UI
- Require Technical Knowledge
- Limited Interaction
- Long- or Short-running Processes

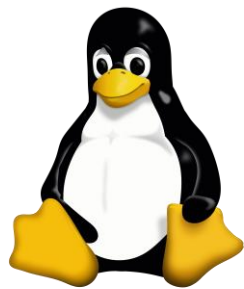
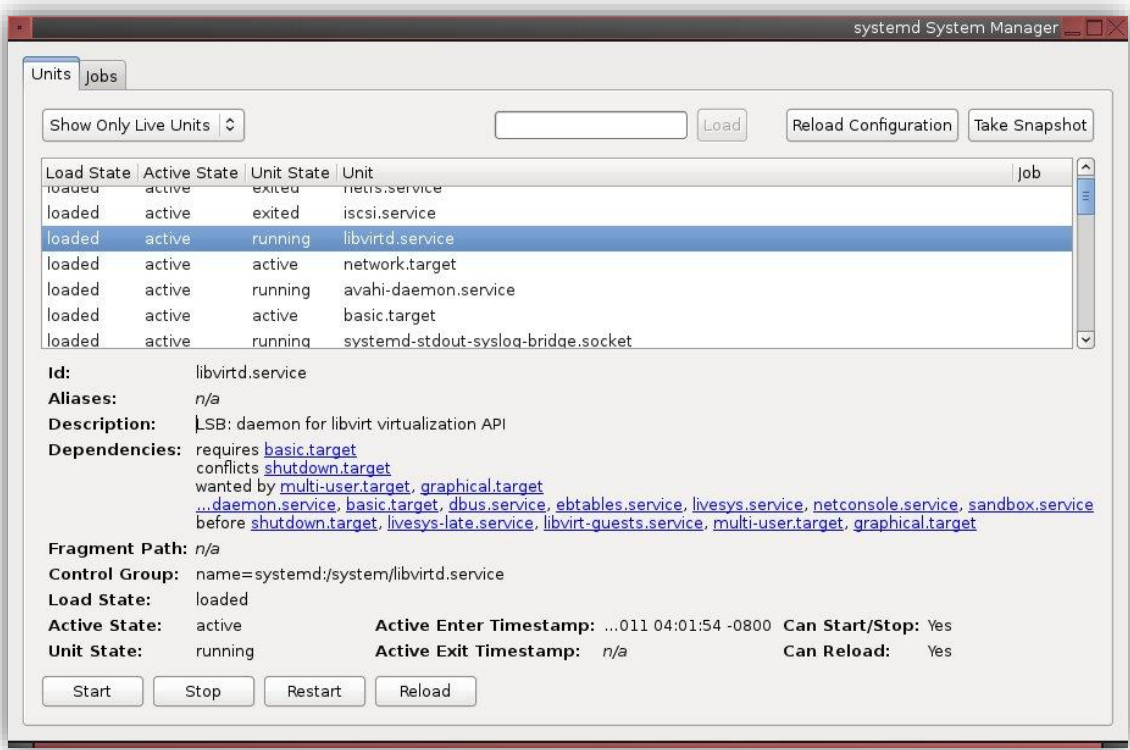
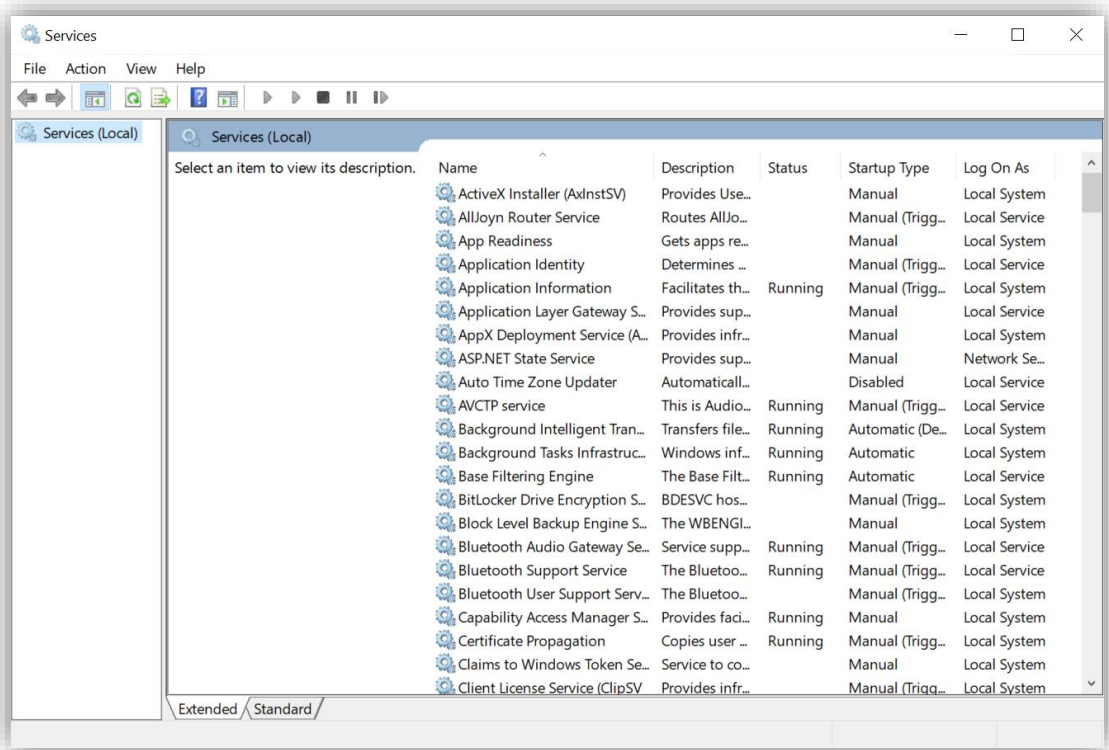
Console Applications

- Best suited for systems that require:
 - Long-running processes
 - Short actions by trained power-users

Services

- Like Console, but:
 - No UI at all
 - Managed by Service Manager

Service Managers



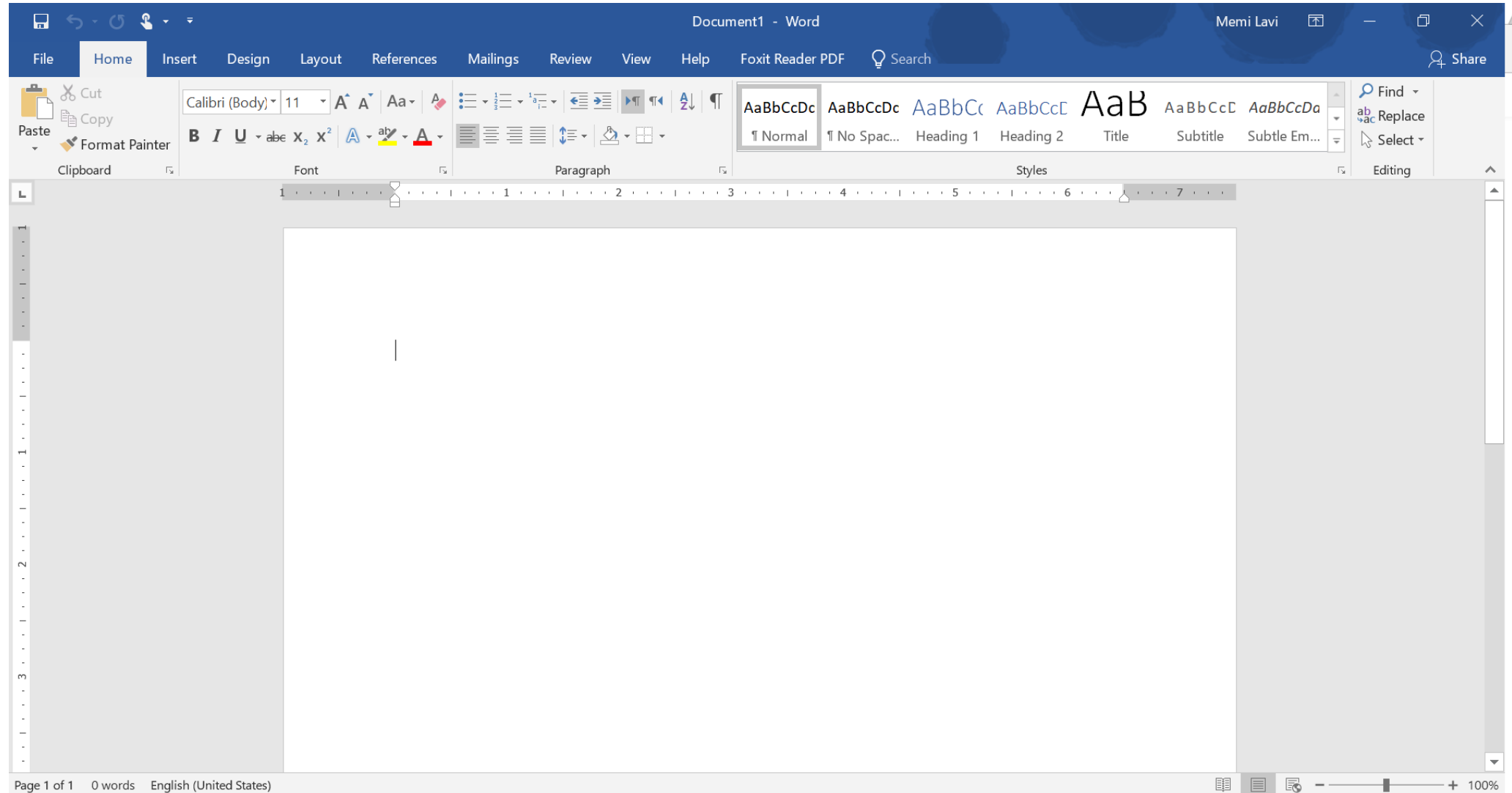
Services

- Best suited for systems that require:
 - Long-running processes (ie. Monitoring folders and processing files)

Desktop Applications

- Has all its resources on the PC
- Might connect to the web
- Great UI

Desktop Applications



Desktop Apps

- Best suited for:
 - User Centric Actions (ie. Word Processing)
 - Gaming

Application Types - Summary

- More Application Types:



Application Types - Summary

- Application Type should be set early
- Can be more than one
 - Example: Web App + Service