Azure Pipeline for Laravel 11

**Database: SQLite**

1. Create a App Service > Stack: PHP 8.2
   1. Download and Install [Composer](https://getcomposer.org/Composer-Setup.exe)
   2. **Local Machine: composer create-project laravel/laravel application-name**
2. Go to Environment Variables > Add > PORT 8000
3. Add Start up Command: **php artisan serve --host 0.0.0.0**
4. Uploaded to Azure Repos.
5. YAML File:

trigger:

  - main

pool:

  vmImage: 'ubuntu-latest'

variables:

  phpVersion: 8.2

steps:

  - script: |

      sudo add-apt-repository ppa:ondrej/php -y

      sudo apt-get update

      sudo apt-get install php8.2 -y

      php -v

    displayName: 'Install PHP 8.2 from PPA'

  - script: |

      sudo apt-get update

      sudo apt-get install -y \

        php8.2-xml \

        php8.2-mbstring \

        php8.2-dom \

        php8.2-zip \

        php8.2-xmlwriter \

        php8.2-curl \

        php8.2-gd

      php -m  # List enabled extensions

    displayName: 'Install PHP Extensions'

  # Download secure environment file

  - task: DownloadSecureFile@1

    inputs:

      secureFile: '.env'

    displayName: 'Download .env Secure File'

  # Move .env file to the working directory

  - script: |

      mv $(Agent.TempDirectory)/.env $(System.DefaultWorkingDirectory)

    displayName: 'Move .env to working directory'

  - script: |

      sudo update-alternatives --set php /usr/bin/php$(phpVersion)

      sudo update-alternatives --set phar /usr/bin/phar$(phpVersion)

      sudo update-alternatives --set phpdbg /usr/bin/phpdbg$(phpVersion)

      sudo update-alternatives --set php-cgi /usr/bin/php-cgi$(phpVersion)

      sudo update-alternatives --set phar.phar /usr/bin/phar.phar$(phpVersion)

      php -version

    displayName: 'Use PHP version $(phpVersion)'

  - script: |

      composer install --no-interaction --prefer-dist

    displayName: 'Install Composer Dependencies'

  # Package application files, including the .env file

  - script: |

      mkdir -p $(Build.ArtifactStagingDirectory)

      cp .env $(Build.ArtifactStagingDirectory) # Include .env file

      zip -r $(Build.ArtifactStagingDirectory)/app.zip .

    displayName: 'Package Application'

  - task: PublishBuildArtifacts@1

    inputs:

      PathtoPublish: '$(Build.ArtifactStagingDirectory)'

      ArtifactName: 'drop'

      publishLocation: 'Container'

    displayName: 'Publish Build Artifacts'

  - task: AzureRmWebAppDeployment@4

    inputs:

      ConnectionType: 'AzureRM'

      azureSubscription: 'TechOkay (ab1b1ec7-da7b-480d-97e9-5d6daf8271d5)'

      appType: 'webAppLinux'

      WebAppName: 'techokaylaravel'

      packageForLinux: '$(Build.ArtifactStagingDirectory)/\*\*/\*.zip'

    displayName: 'Deploy to Azure Web App'

1. Upload the .env file to Pipelines > Library > Secure Files > Upload

# Post-Deployment Steps:

* Open SSH of App Service:
  + touch /home/site/wwwroot/database/database.sqlite
  + php artisan config:clear
  + php artisan cache:clear
  + php artisan migrate --force

# For MySQL –

* Update the .env file and turn off “Require\_Secure\_Transport”