Lab: Packaging 101 with UPack

Length: 20 minutes

Scenario

Your team is currently in the middle of a project to ensure that no resources required for day to day operations exist in non-versioned environment.

Currently, all the utilities used by your team only exist on a UNC share.

Your task today is to take the resources for the `FileCounter` utility, and package it to be served on the internal ProGet server.

> Note: For this lab, you will not have access to a physical ProGet server to do the testing with. Instead, you'll walk through test installing the package via a file path.

Resources

The files for the `FileCounter` utility are available at https://github.com/swiftkickin/FileCounter/releases/.

Step 1: Installing UPack

> Note: If you already have UPack installed on your machine, you can skip this step.

There are two methods for installing UPack on your machine. Choose the method that works best for your environment.

Chocolatey (Windows Only)

Our preferred method for installing UPack is with Chocolatey. To install UPack, go to your console and type:

```
choco install upack
```

Direct Install

Direct installation requires you download the UPack executable from GitHub. Head over to https://github.com/Inedo/upack/releases and download the latest version of upack.exe.

Save upack.exe wherever you would like.

> Note: upack.exe is more useful if it is globally accessible. We recommend that wherever you save upack.exe is accessible via your PATH environment variable. If you're unsure how to do this, we recommend How To Geek Tutorial.

Step 2

Download the current working copy of `FileCounter` at the link in the Resources section. Save the copy to a local working directory.

> Note: The lab will reference c:\filecounter as the location of the files. Please feel free to substitute this with the location of your choice.

Step 3

In the `c:\filecounter` directory, create a new file called `upack.json`. Open `upack.json` in a file editor of your choice.

In the file editor, write the following JSON:

```
{
    "name": "FileCounter",
    "version": "1.0.0",
    "title": "File Counter Utility",
    "description":"This command line utility will count the number of files in the current directory."
}
```

This list of properties in the `upack.json` file are self-explanatory, but let's review anyway.

`name` is universal identifier for your package. When people are looking for your package to install, this is the identifier they will use.

`version` is the current version of the package, and follows the rules of semantic versioning.

`title` is the friendly name of the package. If you are viewing the package online or in a list, the `title` will help your users better understand what they're looking at.

`description` is a long-form description of what your package is, or what the function of your package is.

Step 4

In your console (inside the `c:\filecounter` directory) , type the following command:

> upack pack . --manifest=upack.json

The `upack pack` command tells UPack you want to generate a new package. It will use the current directory (depicted by the `.`), and uses the `upack.json` file to depict the metadata for the package.

It's not necessary to use `upack.json` for the metadata. An alternative would be to use the command line arguments for `upack pack` to provide the same required information.

If this is done successfully, you will see a new file in the `c:\filecounter` directory called `FileCounter-1.0.0.upack`.

> If you have access to a ZIP explorer utility, such as 7zip, take a moment to open your `FileCounter-1.0.0.upack` file. While it doesn't use the `.zip` extension, it is still a ZIP file.

Step 5

Now that we have a packaged version of our `FileCounter` utility, we want to take a moment to show you how this should be installed.

Assume the current package exists at `c:\filecounter\FileCounter-1.0.0.upack`. Your location might be different.

Go to a different directory, for example `c:\tools\` in your console. Type the following command:

```
upack install FileCounter -source="c:\filecounter\FileCounter-1.0.0.upack"
```

If everything works correctly, FileCounter should be installed in the current directory.

Congratulations! You've finished the lab. Way to go.