

VPM Package

- [Package Manifest](#)

Package Manifest

VPM Package Manifest

Package manifest is a JSON file contained at the root of the package and is used to define the properties of the package, such as name, version, author, dependencies, etc.

UPM Properties

“ Unity UPM package manifest documentation:
<https://docs.unity3d.com/Manual/upm-manifestPkg.html>

Property	JSON Type	Description
<code>name</code> *	String	The officially registered package name. (e.g. <code>com.vrchat.udonsharp</code>)
<code>version</code> *	String	The package version number. Documentation
<code>displayName</code>	String	User-friendly name displayed in VRChat Creator Companion , Unity Package Manager and Project Window
<code>description</code>	String	Brief description of the package
<code>author</code>	Object	The author of the package. <code>name</code> * - Name of the author <code>email</code> - Email of the author <code>url</code> - URL to author's page (e.g. GitHub , GitLab , Booth)
<code>unity</code>	String	Lowest compatible Unity version for this package, e.g. <code>2019.4</code> . (In case of VPM packages, recommended to set to current version used by VRChat.)
<code>unityRelease</code>	String	Lowest compatible release of the Unity version defined in <code>unity</code> , e.g. <code>29f1</code>

Property	JSON Type	Description
<code>dependencies</code>	Object	A map of package dependencies. Keys are package names, and values are specific versions. They indicate other packages that this package depends on. Note: The Package Manager doesn't support range syntax, only SemVer versions.
<code>keywords</code>	String[]	An array of keywords used by the Package Manager search APIs. This helps users find relevant packages.
<code>samples</code>	Object[]	List of samples included in the package. Each sample contains a display name, a description, and the path to the sample folder starting at the Samples~ folder itself.
<code>license</code>	String	Identifier for an OSS license using the SPDX identifier format , or a string such as "See LICENSE.md file".
<code>licensesUrl</code>	String	Custom location for this package's license information specified as a URL.
<code>changelogUrl</code>	String	Custom location for this package's changelog specified as a URL.
<code>documentationUrl</code>	String	Custom location for this package's documentation specified as a URL.

* - Property required by UPM

Example package.json

```
{
  "name": "com.varneon.vpm-test-package",
  "displayName": "VPM Test Package",
  "version": "1.0.1",
  "unity": "2019.4",
  "unityRelease": "29f1",
  "description": "Test package for VPM.",
  "author": {
    "name": "Varneon",
```

```

    "email": "developer@varneon.com",
    "url": "https://github.com/Varneon"
  },
  "documentationUrl": "https://example.com/documentation.html",
  "changelogUrl": "https://example.com/changelog.html",
  "licensesUrl": "https://example.com/licensing.html",
  "license": "MIT",
  "keywords": [
    "vpm",
    "vrchat",
    "udon"
  ],
  "dependencies": {
    "com.vrchat.worlds": "3.1.9"
  },
  "samples": [
    {
      "displayName": "My Sample",
      "description": "Sample from the test package.",
      "path": "Samples~/MySample"
    }
  ]
}

```

VPM Additions

“ VPM manifest additions documentation:

<https://vcc.docs.vrchat.com/vpm/packages/#vpm-manifest-additions>

Property	JSON Type	Description
<code>url</code> *	String	Direct download link to a zip file for a version of the package defined in the manifest <i>(Typically appended by automated system, e.g. GitHub Action for repository listing)</i>

Property	JSON Type	Description
<code>vpmDependencies</code>	Object	VPM-based dependencies your package requires. (<i>Dependencies are required to be available in the repository listings in your Creator Companion</i>).
<code>legacyFolders</code>	Object	Path-GUID properties for defining folders in the Assets directory used by old .unitypackage-based versions of this package. Matching folders will be removed when importing the new VPM package. (<i>If no GUID is provided, only path will be matched</i>).
<code>legacyFiles</code>	Object	Same as <code>legacyFolders</code> above, but for files.

* - Property required by VPM

Example package.json

```
{
  "url" : "https://packages.vrchat.com/com.vrchat.avatars-3.1.0.zip",
  "vpmDependencies" : {
    "com.vrchat.avatars" : "3.1.0"
  },
  "legacyFolders" : {
    "Assets\\FolderName" : "vr031f928e5c709x9887f6513084aaa51"
  },
  "legacyFiles" : {
    "ProjectVersion.txt" : "jf988739jfdskljf098323jjhf"
  }
}
```