NUNO BISPO

PIP INSTALL YOURS

A PRACTICAL GUIDE TO PACKAGING AND PUBLISHING PYTHON PROJECTS ON PYPI

YOUR CODE DESERVES TO BE ON PYPI.
PIP INSTALL YOURS IS A STEP-BY-STEP GUIDE TO PACKAGING AND PUBLISHING PYTHON PROJECTS — FROM PROJECT STRUCTURE TO YOUR FIRST LIVE RELEASE ON PYPI.

pip install yours

A Practical Guide to Packaging and Publishing Python Projects on PyPI

Nuno Bispo

This book is available at https://leanpub.com/pip-install-yours

This version was published on 2026-03-05

Tweet This Book!

Please help Nuno Bispo by spreading the word about this book on Twitter!

The suggested tweet for this book is:

Just grabbed ”pip install yours” by @NunoBispo – a practical guide to packaging and publishing Python projects on PyPI. If you’ve ever wanted your code on pip, this is the book! 🐍📦 #Python #PyPI #pipinstallyours

The suggested hashtag for this book is #pipinstallyours.

Find out what other people are saying about the book by clicking on this link to search for this hashtag on Twitter:

#pipinstallyours

Contents

Chapter 0: Welcome

This booklet is a short, practical guide to **packaging Python projects for PyPI**, the Python Package Index. If you've written a script or a small library and want others to install it with `pip install yourpackage`, you're in the right place.

* * *

Purpose

The goal is simple: take you from zero to a published package in one sitting. You'll learn how to:

- Structure a project with a modern `pyproject.toml`
- Build a wheel and a source distribution
- Upload to Test PyPI (for practice) and then to the real PyPI
- Add tests, a README, and optional automation (e.g. GitHub Actions)

No long theory, just the steps and choices that matter. By the end you'll have a package anyone can install with a single `pip` command.

* * *

What you need

You'll need **Python** (3.8 or newer is a safe baseline), **pip**, and a free account on PyPI (and optionally on Test PyPI for practice). Everything in this guide uses a single config file, `pyproject.toml`, no `setup.py` required.

* * *

PyPI, Test PyPI, and what you'll build

PyPI is the official Python Package Index: when someone runs `pip install something`, pip fetches from PyPI by default.

Test PyPI is a separate, sandbox index where you can upload trial packages without affecting the real one; we use it to rehearse the release flow.

You'll produce two kinds of artifacts: a **wheel** (`.whl`), the preferred binary format for installs, and a **source distribution** (sdist), a version of your project that pip can build from source. Both are created with the same command and uploaded together.

* * *

About the author

Nuno Bispo is a Senior Software Engineer with more than 20 years of experience in software development.

He has worked in various industries - insurance, banking, and airlines - building software on low-code platforms.

In recent years he has deepened his skills in Python and Django, working as a freelance consultant on international projects and writing several innovative articles on his blog.

Nuno currently works as a Solutions Architect for a major multinational corporation.

He holds a degree in Computer Engineering.

Chapter 1: Introduction

You've written Python code that works. Maybe it's a script, a small library, or a helper module. The next step is to share it so others can install it with:

```
pip install yourthing
```

That's what packaging is for.

You turn your project into a *distribution* that pip can download from the Python Package Index (PyPI). Once it's there, anyone can install your package with a single command.

What you'll do in this booklet: build one minimal package from scratch, upload it to Test PyPI (a practice index), then publish it on the real PyPI. By the end you'll have a package that anyone can install with `pip`.

What you need:

- **Python 3.8+** (or whatever recent version you already use)
- **pip** (comes with Python)
- **A PyPI account**, free at pypi.org (create one before you publish to the real PyPI in Chapter 4)
- **Optional:** a Test PyPI account for practice uploads in Chapter 2
- **Optional:** build and twine, we'll install these when we need them

No prior packaging experience is required. If you can run commands in a terminal and edit a config file, you're set.

* * *

PyPI and pyproject.toml in a nutshell

PyPI (pypi.org) is the default place pip looks for packages. When someone runs `pip install requests`, pip fetches the package from PyPI. You'll upload your built files there so the same command works for your package.

Test PyPI (test.pypi.org) is a separate, sandbox index for trying things out, same workflow, but installs use `pip install --index-url https://test.pypi.org/simple/ ...` so nothing goes to the real index until you're ready.

When you "build" a package, you produce **artifacts** that pip can install. A **wheel** (`.whl`) is a ready-to-install archive; pip prefers it because installation is fast and doesn't require a build step. A **source distribution** (sdist, `.tar.gz`) contains your source code; pip can install from it when no wheel is available or when someone explicitly asks for it. For most pure-Python packages, building produces both; we'll do that in the next chapter.

All of your package's metadata and build settings live in **`pyproject.toml`** at the project root. This file is the standard (PEP 517, PEP 518) and replaces the old `setup.py` / `setup.cfg` approach. You don't need a `setup.py` for a new package, just `pyproject.toml` and your code.

Next: Chapter 2 creates the minimal project layout and `pyproject.toml`, then builds the package and uploads it to Test PyPI.

Chapter 2: Your First Package

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Step 1: Create the project layout

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Step 2: Add `pyproject.toml`

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Step 3: Build the package

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Step 4: Install locally (editable)

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Step 5: Upload to Test PyPI

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Step 6: Install from Test PyPI

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Chapter 3: Structure, Metadata, and Building

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Project layout: the `src/` layout

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Metadata in `pyproject.toml`

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Version: one place, semantic versioning

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Dependencies and optional extras

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Build backend and what `build` produces

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Chapter 4: Tests, README, and Publishing to PyPI

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Running tests

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

README as the PyPI long description

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

PyPI account and API token

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Upload with twine

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Test on Test PyPI first, then production

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Release flow: version, tag, build, upload

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

One simple GitHub Action

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Chapter 5: Next Steps and Reference

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Next steps

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Minimal `pyproject.toml` template

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Key commands

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.

Pre-publish checklist

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/pip-install-yours.