NIX FORGE

BECOME A NIX SMITH OVER THE WEEKEND

FOSDEM, Brussels, Belgium, 2026

Ivan Mincik, @imincik

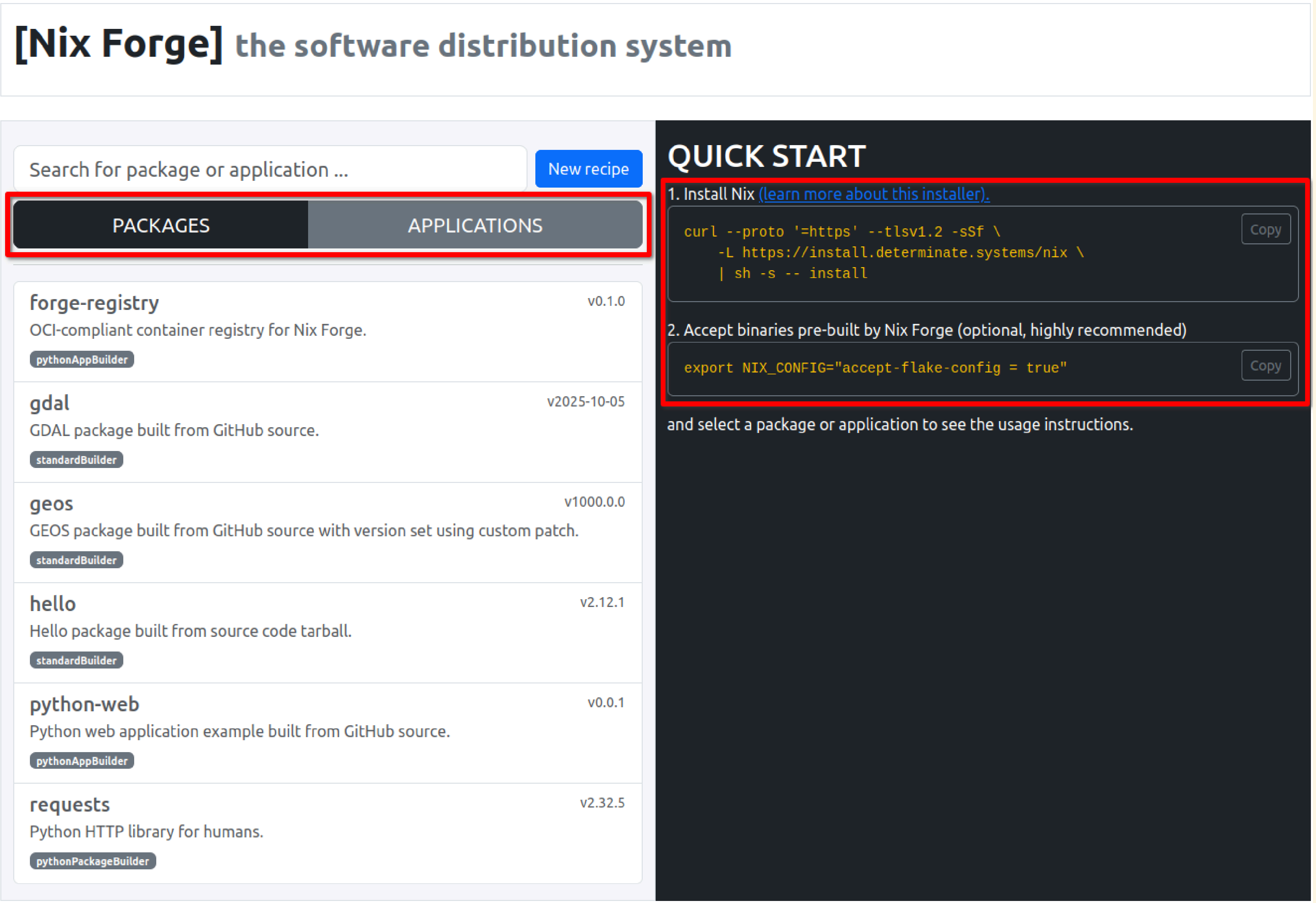
IVAN MINCIK, @IMINCIK

- Nix Geospatial Team
- Nix@NGI Team
- Nix/NixOS consulting services

NIX FORGE

Simple software **packaging**, **distribution** and **deployment** with Nix

USER INTERFACE



[Nix Forge] the software distribution system

forge-registry v0.1.0
OCI-compliant container registry for Nix Forge.
pythonAppBuilder

gdal v2025-10-05
GDAL package built from GitHub source.
standardBuilder

geos v1000.0.0
GEOS package built from GitHub source with version set using custom patch.
standardBuilder

hello v2.12.1
Hello package built from source code tarball.
standardBuilder

python-web v0.0.1
Python web application example built from GitHub source.
pythonAppBuilder

requests v2.32.5
Python HTTP library for humans.
pythonPackageBuilder

QUICK START

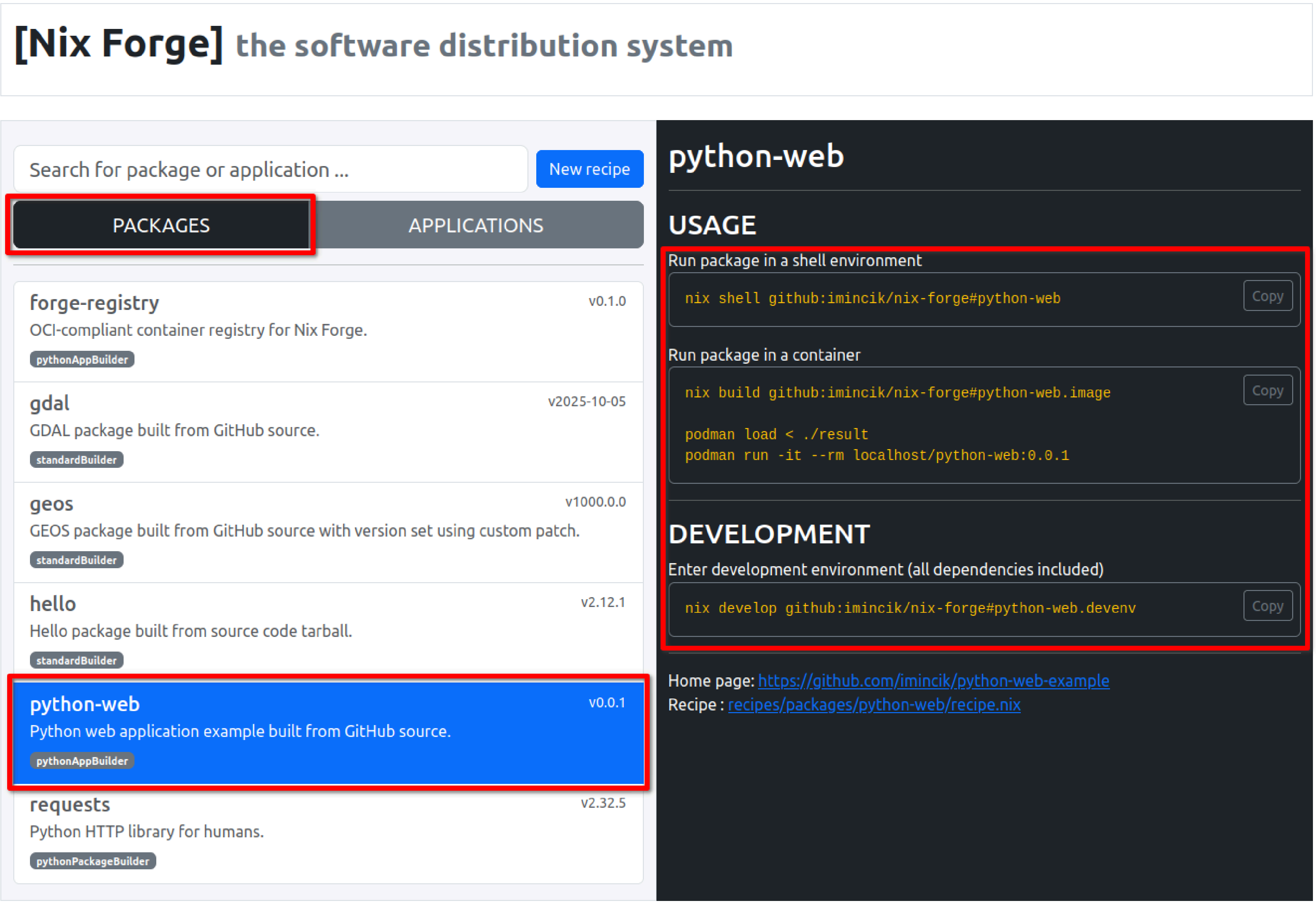
1. Install Nix (learn more about this installer).

```
curl --proto '=https' --tlsv1.2 -sSf \
    -L https://install.determinate.systems/nix \
    | sh -s -- install
```

2. Accept binaries pre-built by Nix Forge (optional, highly recommended)

```
export NIX_CONFIG="accept-flake-config = true"
```

and select a package or application to see the usage instructions.



[Nix Forge] the software distribution system

forge-registry v0.1.0
OCI-compliant container registry for Nix Forge.
pythonAppBuilder

gdal v2025-10-05
GDAL package built from GitHub source.
standardBuilder

geos v1000.0.0
GEOS package built from GitHub source with version set using custom patch.
standardBuilder

hello v2.12.1
Hello package built from source code tarball.
standardBuilder

python-web v0.0.1
Python web application example built from GitHub source.
pythonAppBuilder

requests v2.32.5
Python HTTP library for humans.
pythonPackageBuilder

python-web

USAGE

Run package in a shell environment

```
nix shell github:imincik/nix-forge#python-web
```

Run package in a container

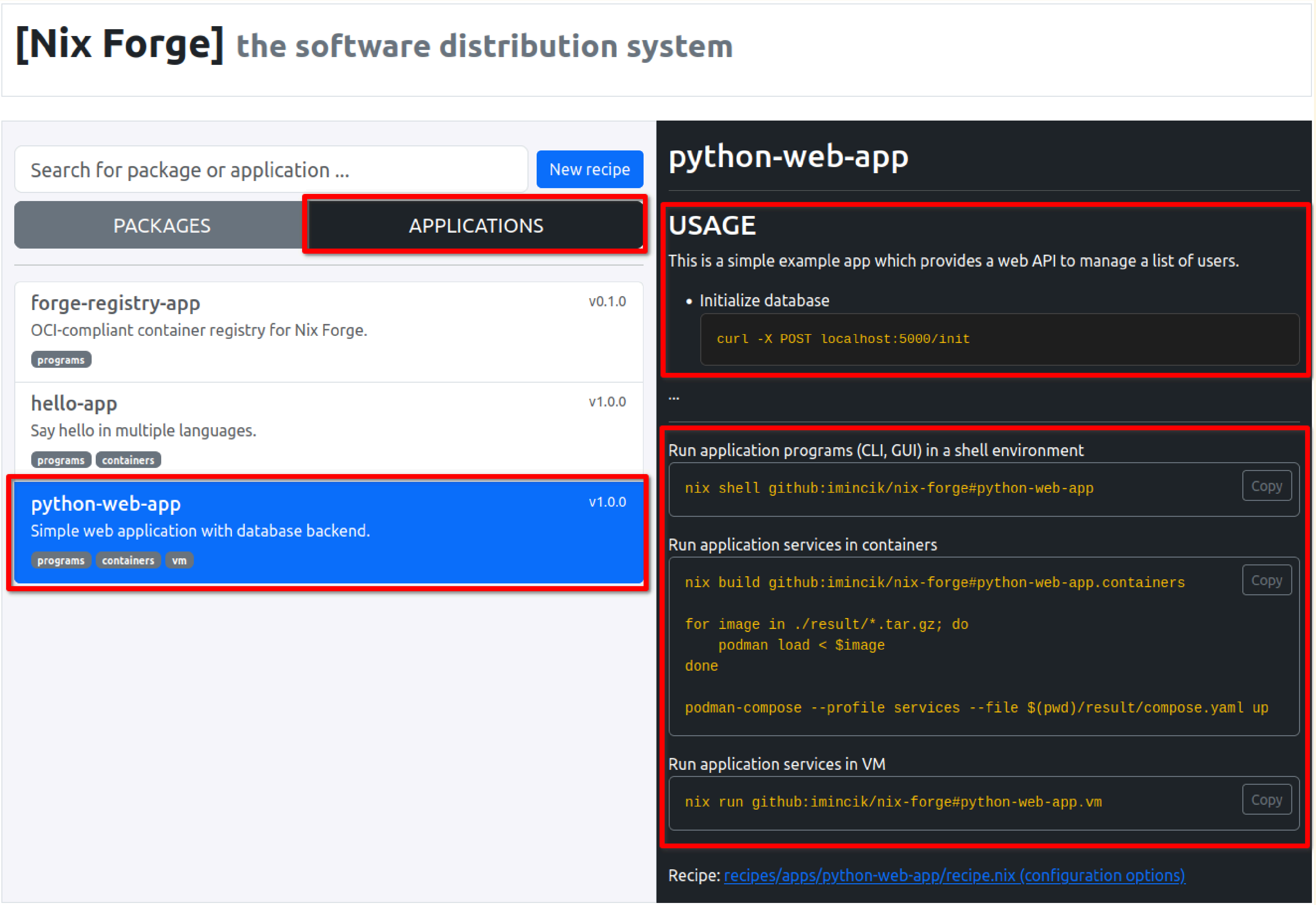
```
nix build github:imincik/nix-forge#python-web.image

podman load < ./result
podman run -it --rm localhost/python-web:0.0.1
```

DEVELOPMENT

Enter development environment (all dependencies included)

```
nix develop github:imincik/nix-forge#python-web.devenv
```

Home page: https://github.com/imincik/python-web-example
Recipe : recipes/packages/python-web/recipe.nix



[Nix Forge] the software distribution system

forge-registry-app v0.1.0

OCI-compliant container registry for Nix Forge.

programs

hello-app v1.0.0

Say hello in multiple languages.

programs containers

python-web-app v1.0.0

Simple web application with database backend.

programs containers vm

python-web-app

USAGE

This is a simple example app which provides a web API to manage a list of users.

- Initialize database

```
curl -X POST localhost:5000/init
```

...

Run application programs (CLI, GUI) in a shell environment

```
nix shell github:imincik/nix-forge#python-web-app
```

Run application services in containers

```
nix build github:imincik/nix-forge#python-web-app.containers

for image in ./result/*.tar.gz; do
    podman load < $image
done

podman-compose --profile services --file $(pwd)/result/compose.yaml up
```

Run application services in VM

```
nix run github:imincik/nix-forge#python-web-app.vm
```

Recipe: recipes/apps/python-web-app/recipe.nix (configuration options)

PACKAGING

NAME, VERSION, SOURCE

```
name = "python-web";
version = "0.0.1";
description = "Python web application example built from GitHub source.";
homePage = "https://github.com/imincik/python-web-example";
mainProgram = "python-web";

source = {
  git = "github:imincik/python-web-example/v0.0.1";
  hash = "sha256-nSW5746+criXHPrxmJ+0zhJCMwl78eer03qQAvDIo5U=";
};
```

CHOOSE BUILDER

- **standardBuilder** - autotools, Makefile, CMake
- **pythonPackageBuilder** - Python modules
- **pythonAppBuilder** - Python apps

BUILD RECIPE

```
build.pythonAppBuilder = {
  enable = true;

  requirements.build-system = [
    pkgs.python3Packages.setuptools
  ];

  requirements.dependencies = [
    pkgs.python3Packages.flask
    pkgs.python3Packages.psycopg2
  ];
};

# Advanced
build.extraDrvAttrs = { };
```

MULTI-COMPONENT APPLICATIONS

NAME, VERSION, DESCRIPTION, USAGE

```
name = "python-web-app";
version = "1.0.0";
description = "Simple web application with database backend.";
usage = ''
  Instructions how to use your app in markdown format.
'';
```

1. PROGRAMS (SHELL ENVIRONMENT) - CLI OR GUI

```
programs = {
  enable = true;

  requirements = [
    pkgs.gdal
    pkgs.qgis
  ];
};
```

2. CONTAINERS - SERVICES

```
containers = {
  enable = true;

  images = [
    {
      name = "api";
      requirements = [ pkgs.mypkgs.python-web ];
      config.CMD = [ "python-web" ];
    }
    # ...
  ];

  composeFile = ./compose.yaml;
};
```

3. NIXOS VM - SERVICES

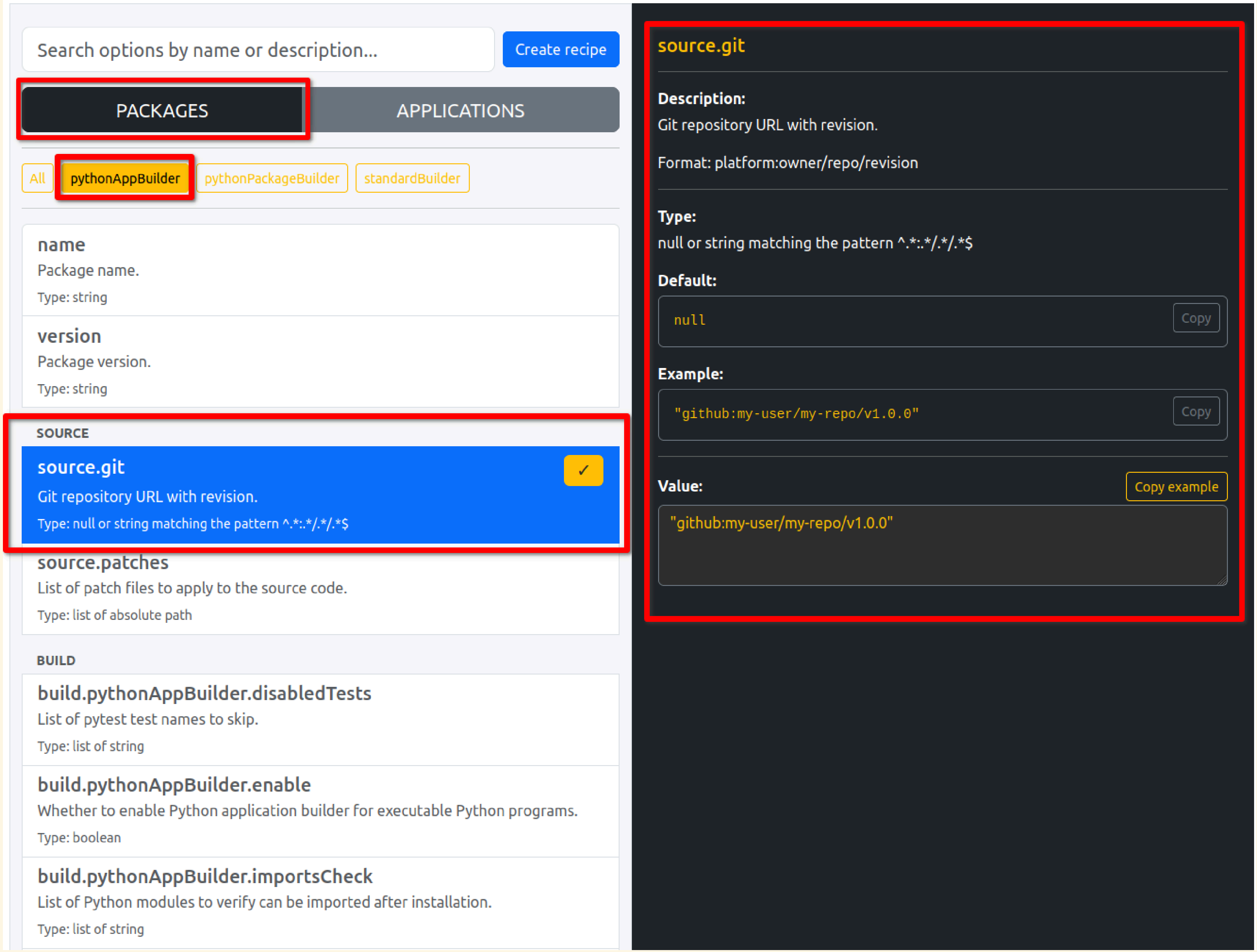
```
vm = {
  enable = true;

  name = "database";

  config.system = {
    services.postgresql.enable = true;
    systemd.services.api.script = "${pkgs.mypkgs.python-web}/bin/python-web";
    systemd.services.api.wantedBy = [ "multi-user.target" ];
  };

  config.ports = [ "5000:5000" ];
};
```

CREATING RECIPES

USING LLM (CLAUDE)

```
Based on instructions in AGENTS.md file, analyze the source code located in
<SOURCE-CODE-LOCATION> and create a Nix Forge package and application recipes.
```

USING WEB UI



name
Package name.
Type: string

version
Package version.
Type: string

SOURCE

source.git
Git repository URL with revision.
Type: null or string matching the pattern ^.*:.*/.*/.*$

source.patches
List of patch files to apply to the source code.
Type: list of absolute path

BUILD

build.pythonAppBuilder.disabledTests
List of pytest test names to skip.
Type: list of string

build.pythonAppBuilder.enable
Whether to enable Python application builder for executable Python programs.
Type: boolean

build.pythonAppBuilder.importsCheck
List of Python modules to verify can be imported after installation.
Type: list of string

source.git

Description:
Git repository URL with revision.

Format: platform:owner/repo/revision

Type:
null or string matching the pattern ^.*:.*/.*/.*$

Default:

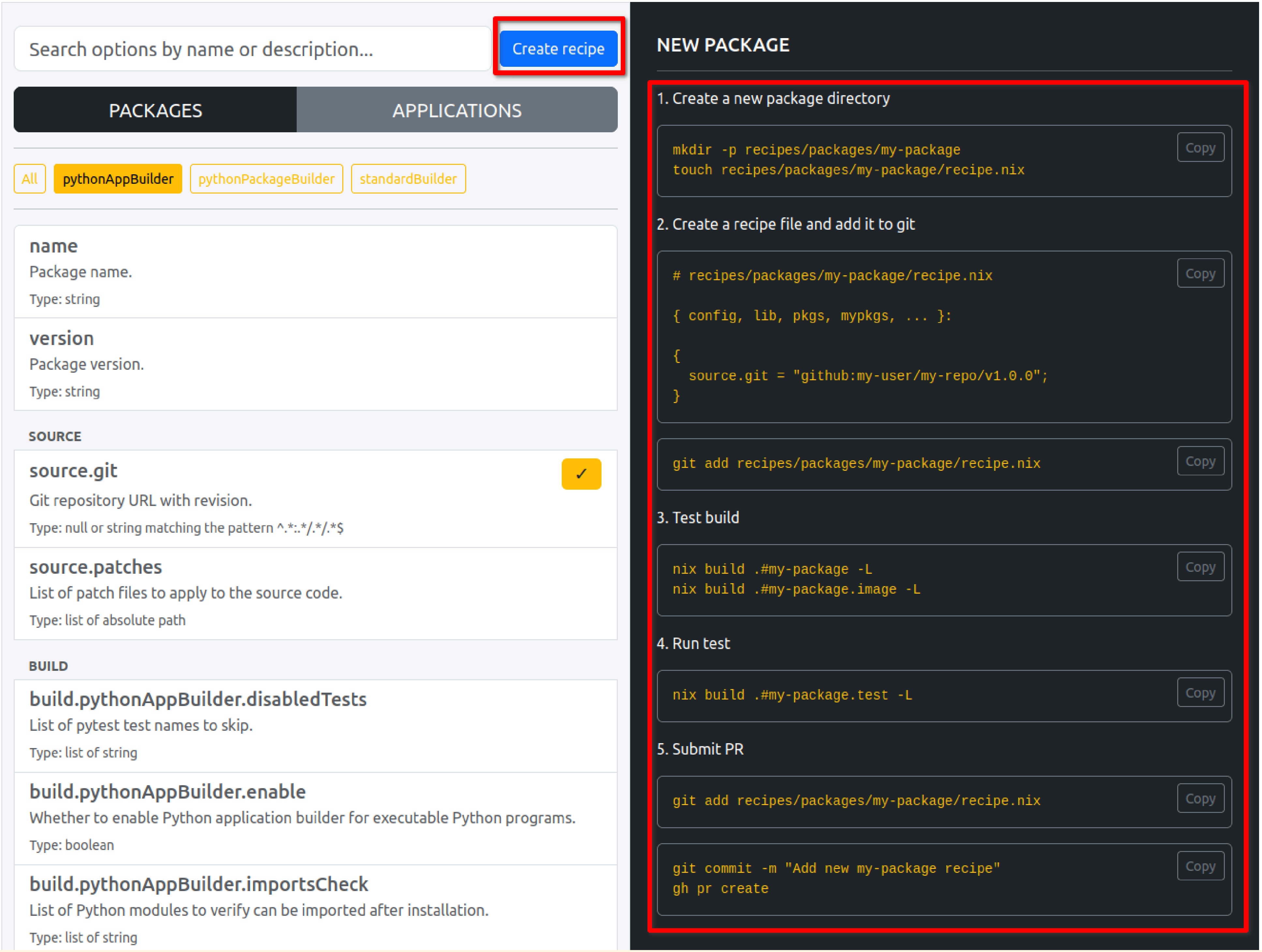
```
null
```

Example:

```
"github:my-user/my-repo/v1.0.0"
```

Value:

```
"github:my-user/my-repo/v1.0.0"
```



name
Package name.
Type: string

version
Package version.
Type: string

SOURCE

source.git
Git repository URL with revision.
Type: null or string matching the pattern ^.*:.*/.*/.*$

source.patches
List of patch files to apply to the source code.
Type: list of absolute path

BUILD

build.pythonAppBuilder.disabledTests
List of pytest test names to skip.
Type: list of string

build.pythonAppBuilder.enable
Whether to enable Python application builder for executable Python programs.
Type: boolean

build.pythonAppBuilder.importsCheck
List of Python modules to verify can be imported after installation.
Type: list of string

NEW PACKAGE

1. Create a new package directory

```
mkdir -p recipes/packages/my-package
touch recipes/packages/my-package/recipe.nix
```

2. Create a recipe file and add it to git

```
# recipes/packages/my-package/recipe.nix

{ config, lib, pkgs, mypkgs, ... }:

{
  source.git = "github:my-user/my-repo/v1.0.0";
}
```

```
git add recipes/packages/my-package/recipe.nix
```

3. Test build

```
nix build .#my-package -L
nix build .#my-package.image -L
```

4. Run test

```
nix build .#my-package.test -L
```

5. Submit PR

```
git add recipes/packages/my-package/recipe.nix
```

```
git commit -m "Add new my-package recipe"
gh pr create
```

DEPLOYMENT

NIX FORGE CONTAINER REGISTRY

Podman

```
podman run registry.imincik.app:6443/applications/hello-app/hello-english:latest
```

Kubernetes

```
kubectl run python-web --image=registry.imincik.app:6443/applications/hello-app/hello-english:latest
```

WARNING: this software is in a prototype state

SELF HOSTING

```
nix flake init --template "github:imincik/nix-forge#example"
```

- https://forge.imincik.app
- https://www.imincik.com