

Terminal Application Presentation

Irene (Binh Bach) Nguyen

Overview



Tasks-tracker is a CLI application to manage and track multiple tasks built with:

- **Typer**: A library for building Command Line Interface (CLI) applications based on Python's type hints.
- **Rich**: A library for rich text and beautiful formatting in the terminal.
- **Sqlite3**: A lightweight disk-based database that does not require a separate server process and allows accessing the database using a nonstandard variant of the SQL query language.

Main features



- Add a new task with a title and optional values like description and priority.
- List all tasks with filter options like priority, status, start date and end date.
- Update a task with a given unique ID.
- Delete a task with a given unique ID.
- Delete all tasks.

Coding structure



This application uses the standard Python package code structure.

- **The Actual Module:** The code of interest
- **Setup.py:** Package and distribution management.
- **Requirements File:** Development dependencies.
- **Test Suite:** Package integration and unit tests.
- **Styling conventions configs:** Configs for styling convention tools.
- **Documentation:** Package reference documentation.

Styling convention



This application uses the style conventions in **PEP8**, a document that provides guidelines and best practices for writing Python code.

Package tools:

- **FLake8**: a tool for style guide enforcement that wraps [PyFlakes](#), [pycodestyle](#) and [Ned Batchelder's McCabe script](#).
- **Black**: an uncompromising Python code formatter to cede control over minutiae of hand-formatting.
- **Isort**: a Python utility/library to sort imports alphabetically and automatically separated into sections and by type.

Feature 1: Add command

A function allows users to **add** a new task with a **title** as a required argument and other options to provide more information for the task.

```
> tasks-tracker add --help

Usage: tasks-tracker add [OPTIONS] TITLE

Arguments
*  title      TEXT [default: None] [required]

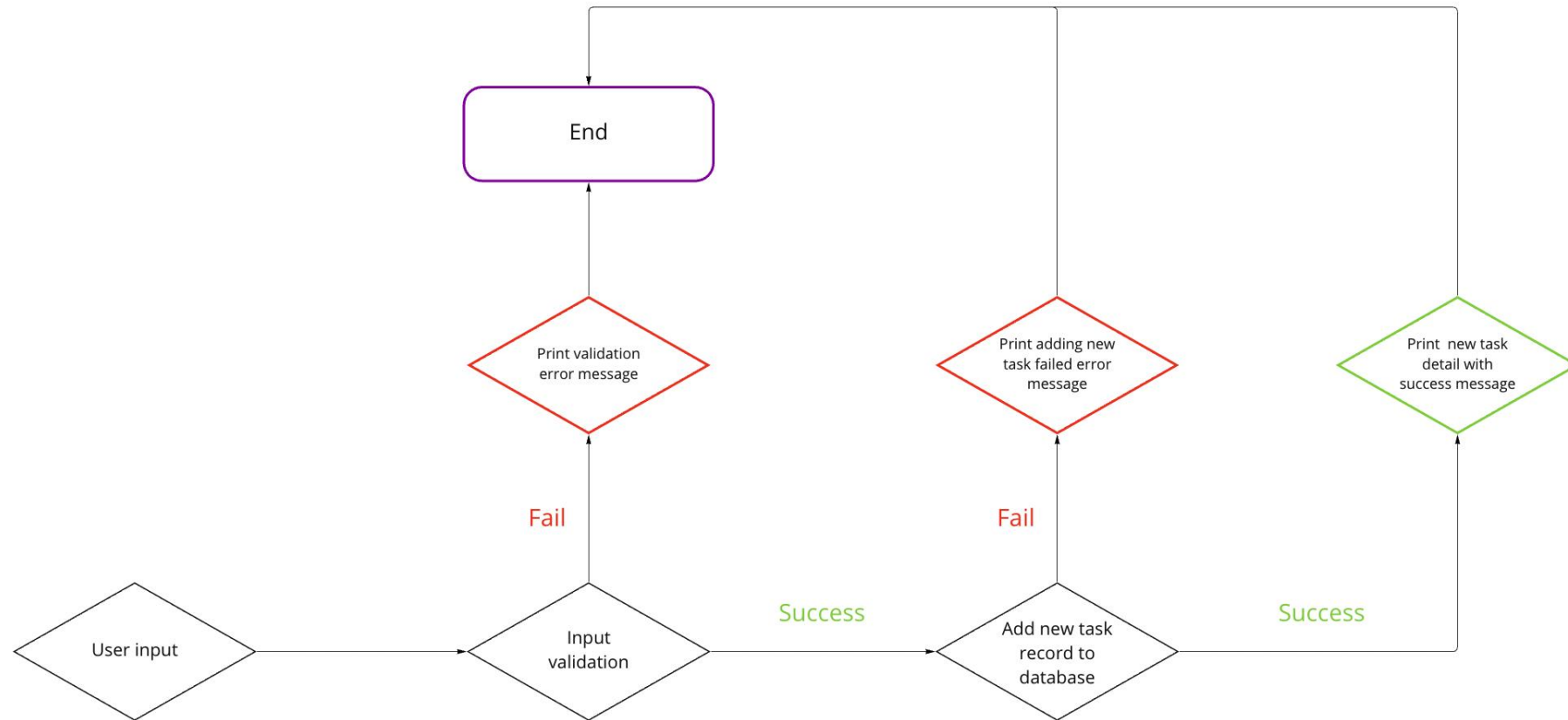
Options
--priority -p [high|medium|low] Set task's priority.
--status   -s [not_started|in_progress|on_hold|done] Set task's status.
--description -d TEXT Set task's description.
--start-date -sd [%d/%m/%Y] Set the start date. E.g 22/02/2022
--end-date  -ed [%d/%m/%Y] Set the end date. E.g 22/02/2022
--help

> tasks-tracker add "Javascript fundamental course" -d "A course to learn basic Javascript" --priority high --start-date 26/09/2022 --end-date 30/10/2022

New task added successfully.
```

ID	d583e3e901
Title	Javascript fundamental course
Description	A course to learn basic Javascript
Priority	HIGH
Status	NOT STARTED
Start date	26/09/2022
End date	30/10/2022

Feature 1: Add command - Flowchart



Feature 2: List command

A function allows users to **list** all tasks with filter options.

```
> tasks-tracker list --help
```

Usage: tasks-tracker list [OPTIONS]

Options

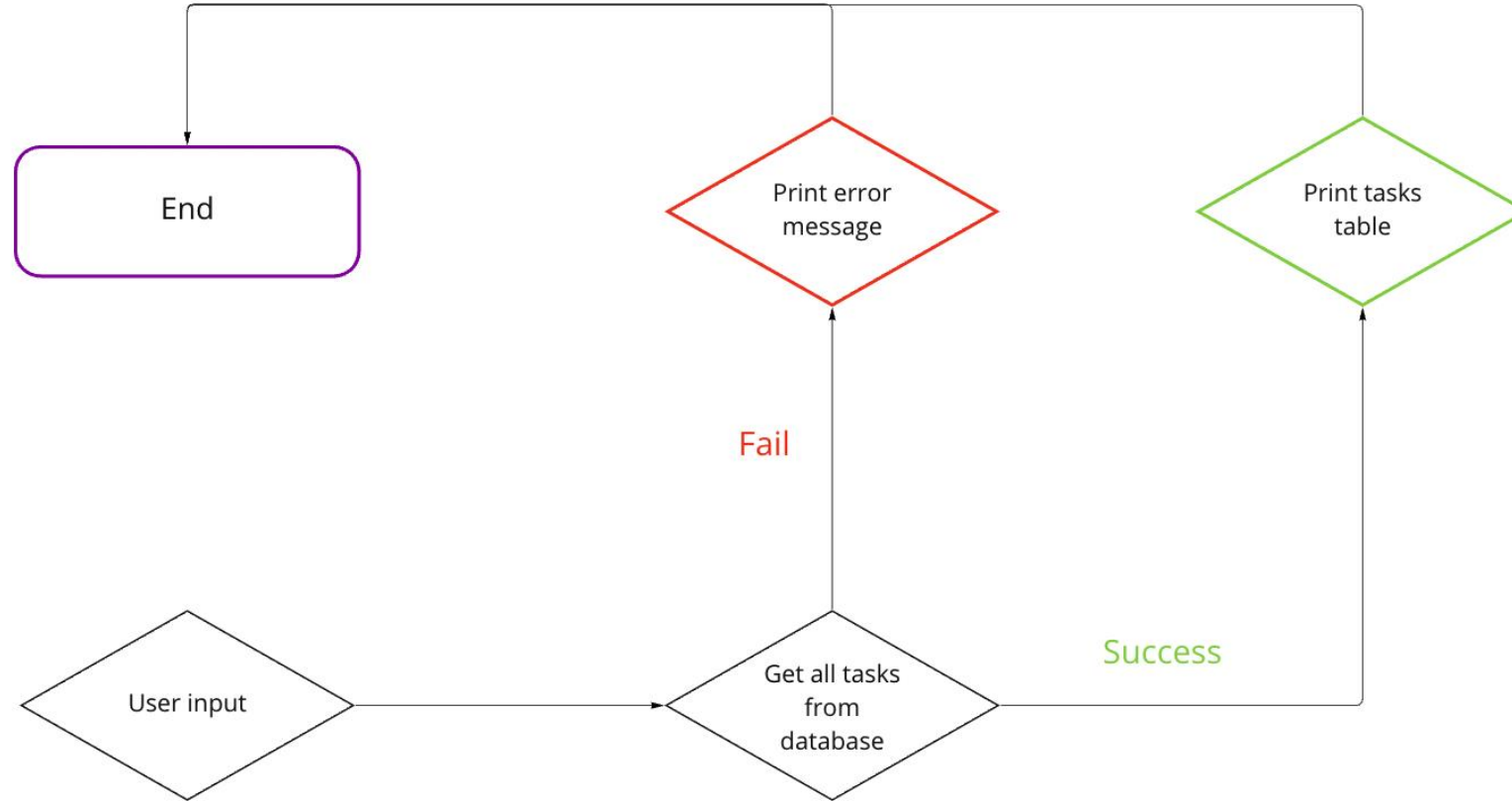
--status	-s	[not_started in_progress on_hold done]	Filter by status.
--priority	-p	[high medium low]	Filter by priority.
--start-date	-sd	[%d/%m/%Y]	Filter by the date from the start date. E.g 22/02/2022
--end-date	-ed	[%d/%m/%Y]	Filter by the date before the end date. E.g 22/02/2022
--help			Show this message and exit.

```
> tasks-tracker list
```

TASKS LIST

Id	Title	Description	Priority	Status	Start date	End date
f0726f3738	Go to bookshop	Buy the favourite book before it is sold out	HIGH	NOT STARTED	20/09/2022	22/09/2022 Expired
e86eb37988	Go to gym	Not provided	MEDIUM	DONE	24/09/2022	Not setting
05fe4cf7cb	Clean house	Clean house before going to shopping	HIGH	NOT STARTED	24/09/2022	25/09/2022 Expired soon

Feature 2: List command - Flowchart



Feature 3: Update command

This feature allows users to **update** an existing task with a unique ID as a required argument and other options to update more information for the task.

```
> tasks-tracker update --help

Usage: tasks-tracker update [OPTIONS] ID

Arguments
*   id      TEXT [default: None] [required]

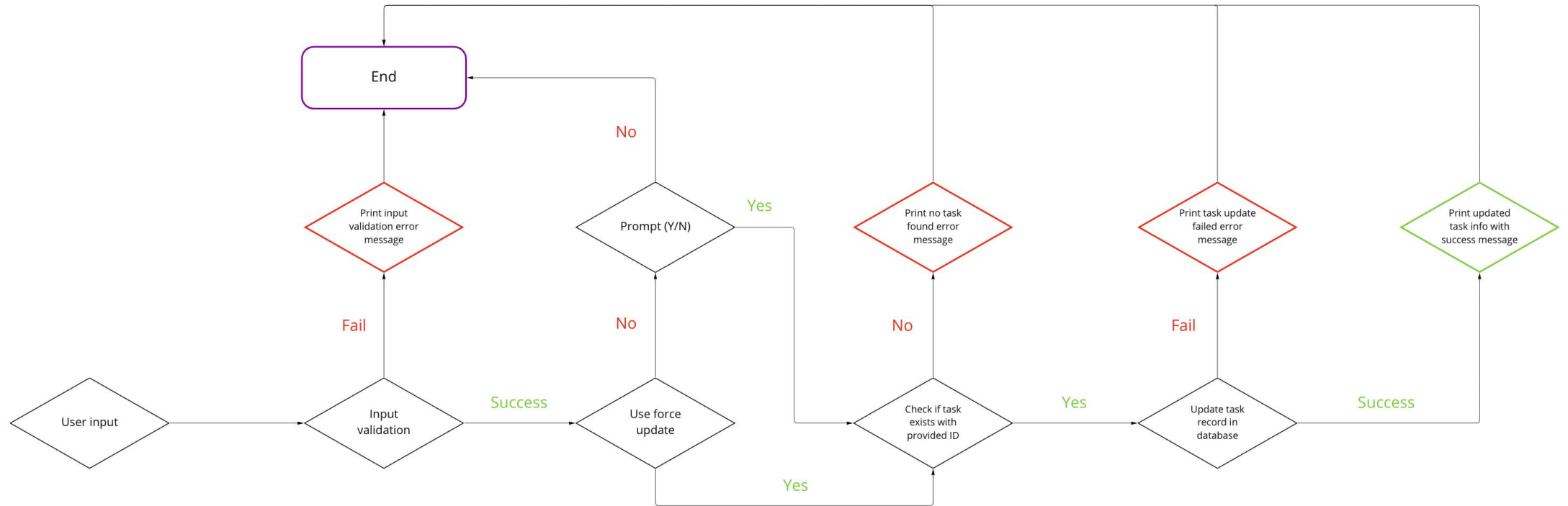
Options
--force      -f          Force update task.
--title      -t          TEXT      Update task's title.
--priority   -p          [high|medium|low] Update task's priority.
--status     -s          [not_started|in_progress|on_hold|done] Update task's status.
--description -d          TEXT      Update task's description.
--start-date -sd         [%d/%m/%Y] Update the start date. E.g 22/02/2022.
--end-date   -ed         [%d/%m/%Y] Update the end date. E.g 22/02/2022.
--help                               Show this message and exit.

> tasks-tracker update "e86eb37988" --description "Go to gym in every weekday."
Update this task with provided data? [y/N]: Y

Task updated successfully.
```

ID	e86eb37988
Title	Go to gym
Description	Go to gym in every weekday.
Priority	MEDIUM
Status	DONE
Start date	24/09/2022
End date	Not setting

Feature 3: Update command - Flowchart



Feature 4: Delete command

A function allows users to **delete** an existing task with a unique ID as a required argument.

```
> tasks-tracker delete --help

Usage: tasks-tracker delete [OPTIONS] ID

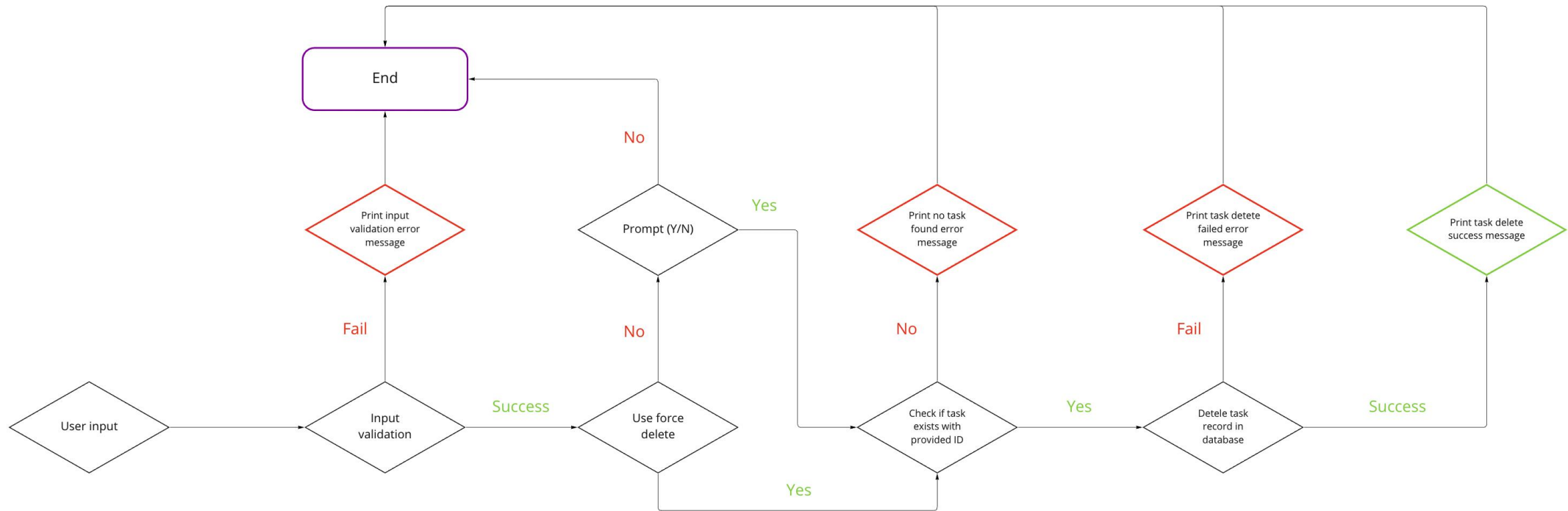
Arguments
  * id      TEXT  [default: None] [required]

Options
  --force -f      Force deleting task.
  --help          Show this message and exit.

> tasks-tracker delete "e86eb37988"
Surely you want to delete this task? [y/N]: y

Task deleted successfully.
```

Feature 4: Delete command - Flowchart



Feature 5: Delete all command

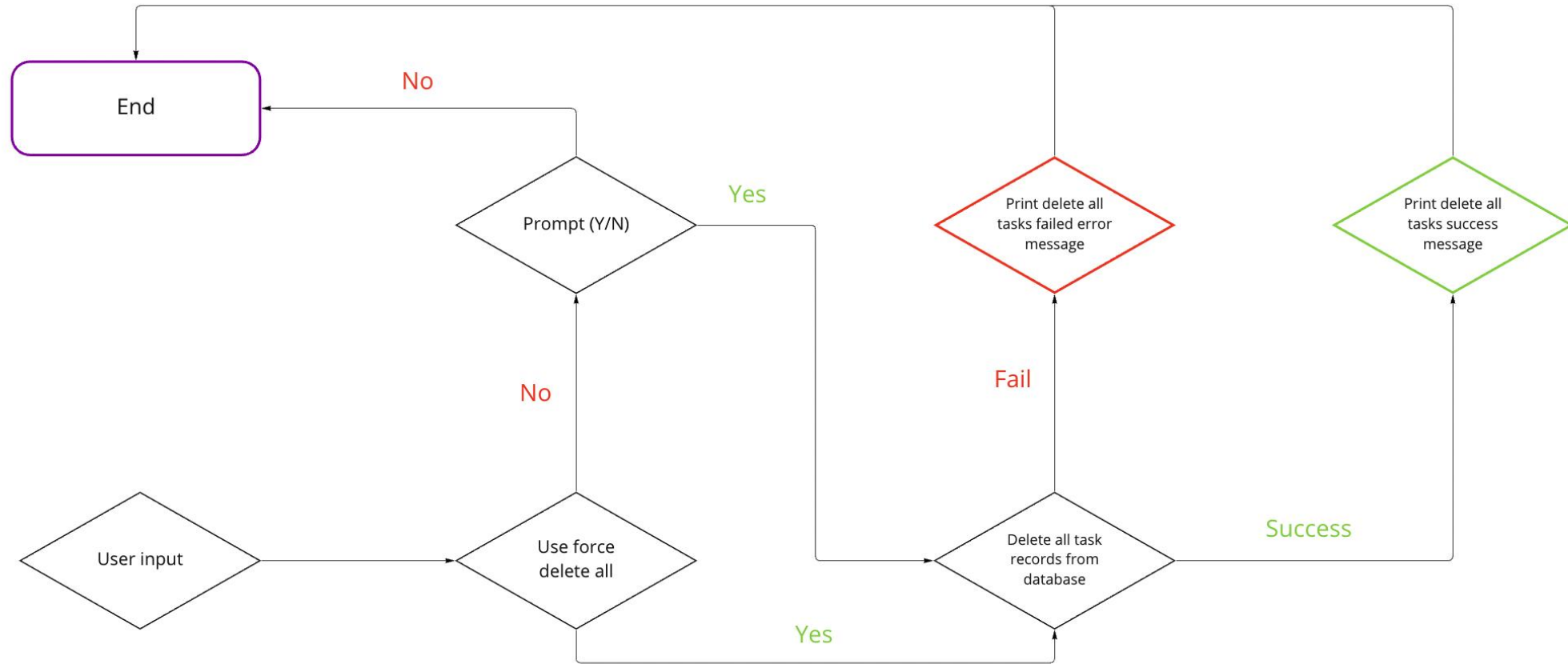
A function allows users to delete all tasks.

```
> tasks-tracker delete-all --help

Usage: tasks-tracker delete-all [OPTIONS]

Options
  --force -f      Force deleting all tasks.
  --help          Show this message and exit.
```

Feature 5: Delete all command - Flowchart



Thank you for listening.