# Contributing to Angular Ngrx Material Starter

We would love for you to contribute to starter project and help make it even better than it is

today! As a contributor, here are the guidelines we would like you to follow:

## Getting started

Please check out one of the getting started guides about GitHub fork / pull requests workflow:

\* [Forking project](https://guides.github.com/activities/forking/)

\* [Fork a repo](https://help.github.com/articles/fork-a-repo/)

\* [Forking](https://gist.github.com/Chaser324/ce0505fbed06b947d962)

## How to sync your fork

Your fork of the repo can fall behind as more work is done in the original repository.

It is always good idea to update your work before starting to work on new issue.

The fork can be updated by navigating to your fork directory and running the following command...

`git checkout master --force && git fetch upstream && git merge upstream/master && git push`

This command assumes you're using unix or unix like environment (macOS, cygwin, WSL, ...).

If not you might need to execute commands one by one instead of chaining them with `&&`.

## Commit Message Guidelines

We have very precise rules over how our git commit messages can be formatted. This leads to \*\*more

readable messages\*\* that are easy to follow when looking through the \*\*project history\*\*. But also,

we use the git commit messages to \*\*generate the [change log](https://github.com/tomastrajan/angular-ngrx-material-starter/blob/master/CHANGELOG.md)\*\*.

### Format

Each commit message consists of a \*\*header\*\*, a \*\*body\*\* and a \*\*footer\*\*. The header has a special

format that includes a \*\*type\*\*, a \*\*scope\*\* and a \*\*subject\*\*.

The \*\*header\*\* is mandatory and the \*\*scope\*\* of the header is optional.

```

<type>(<scope>): <subject>

<BLANK LINE>

<body>

<BLANK LINE>

<footer>

```

### Type

Commit types that will appear in changelog:

\* \*\*feat\*\* - adding new feature

\* \*\*fix\*\* - fixing bug (please use `fixes #<issue-number>` at the end of commit message)

\* \*\*perf\*\* - changes to improve performance

\* If there is \*\*BREAKING CHANGE\*\* text anywhere in the commit message, the commit will always appear in the changelog

Other types that will not appear in changelog:

\* \*\*docs\*\* - changes in documentation

\* \*\*chore\*\* - changes in build or other application unrelated changes

\* \*\*refactor\*\* - changes to implementation without changes to functionality

\* \*\*test\*\* - adding tests

\* \*\*style\*\* - changes to codestyle (should be unnecessary since we use prettier)

Any line of the commit message should not be longer 100 characters! This allows the message to be easier

to read on GitHub as well as in various git tools.

### Scope

The commit scope is specified inside of the parentheses. Scope can be omitted when changed

relates to multiple modules or even whole project. If possible, scope should refer to a module where change was performed.

Approximate steps to specify scope:

1. change belongs to a well defined module like `core`, `shared`, `app`, ...

2. change belongs to a "separated" bigger chung of a module (eg: `todos` from the `examples` module)

3. change belongs to a unique cross-cutting concern like `animations`, `logging` or `build`

4. change belongs to a unique entity like `readme` or `contributing`

### Examples

```

fix(polyfills): add missing hammerjs, fixes #247

```

```

refactor(settings): reorder imports

```

```

test(todos): add dispatch filter action test, adjust existing tests

```

```

fix(app): rework main layout to prevent scrollbar issues, closes #221, closes #240

```

```

docs(readme): add logo, update meta assets

```

```

feat(settings): add runtime animations toggles

```

```

fix(animations): fix dynamic animations in prod build, fixes #71

```

```

chore(release): 6.0.1

```

Even more [examples](https://github.com/tomastrajan/angular-ngrx-material-starter/commits/master) of commit messages from commit history of this project.