# Contributing to Authlogic

## Issues

### Security Issues

\*\*Do not disclose security issues in public.\*\* Instead, please email:

```

Ben Johnson <bjohnson@binarylogic.com>,

Tieg Zaharia <tieg.zaharia@gmail.com>,

Jared Beck <jared@jaredbeck.com>

```

We will review security issues promptly.

### Non-Security Issues

Please use github issues only for bug reports and feature suggestions.

### Usage Questions

Please ask usage questions on

[stackoverflow](http://stackoverflow.com/questions/tagged/authlogic).

## Development

Most local development should be done using the oldest supported version of

ruby. See `required\_ruby\_version` in the gemspec.

### Testing

Tests can be run against different versions of Rails:

```

BUNDLE\_GEMFILE=gemfiles/rails\_6.0.rb bundle install

BUNDLE\_GEMFILE=gemfiles/rails\_6.0.rb bundle exec rake

```

To run a single test:

```

BUNDLE\_GEMFILE=gemfiles/rails\_6.0.rb \

bundle exec ruby -I test path/to/test.rb

```

Bundler can be omitted, and the latest installed version of a gem dependency

will be used. This is only suitable for certain unit tests.

```

ruby –I test path/to/test.rb

```

### Test MySQL

```

mysql -e 'drop database authlogic; create database authlogic;' && \

DB=mysql BUNDLE\_GEMFILE=gemfiles/rails\_5.2.rb bundle exec rake

```

### Test PostgreSQL

```

psql -c 'create database authlogic;' -U postgres

DB=postgres BUNDLE\_GEMFILE=gemfiles/rails\_6.0.rb bundle exec rake

```

### Linting

Running `rake` also runs a linter, rubocop. Contributions must pass both

the linter and the tests. The linter can be run on its own.

```

BUNDLE\_GEMFILE=gemfiles/rails\_6.0.rb bundle exec rubocop

```

To run the tests without linting, use `rake test`.

```

BUNDLE\_GEMFILE=gemfiles/rails\_6.0.rb bundle exec rake test

```

### Version Control Branches

We've been trying to follow the rails way, stable branches, but have been

inconsistent. We should have one branch for each minor version, named like

`4-3-stable`. Releases should be done on those branches, not in master. So,

the "stable" branches should be the only branches with release tags.

### A normal release (no backport)

1. git checkout 4-3-stable # the latest "stable" branch (see above)

1. git merge master

1. Update version number in lib/authlogic/version.rb

1. In the changelog,

- Add release date to entry

- Add a new "Unreleased" section at top

1. In the readme,

- Update version number in the docs table at the top

- For non-patch versions, update the compatibility table

1. Commit with message like "Release 4.3.0"

1. git push origin 4-3-stable

1. CI should pass

1. gem build authlogic.gemspec

1. gem push authlogic-4.3.0.gem

1. git tag -a -m "v4.3.0" "v4.3.0"

1. git push --tags origin 4-3-stable

1. update the docs in the master branch, because that's what people look at

- git checkout master

- git cherry-pick abcd1234 # the SHA from the stable branch

- commit and push with `[ci skip]`