# Contribution Guidelines

Any change to the code needs to be proven by passing the Unit or Systems Integration tests. Developers should add accompanying tests (of type either Unit or Systems Integration) to the set of changes in a PR. If a PR lacks the required tests, it should wait till all the tests are written before being merged into `develop` branch.

AppVeyor CI builds PRs and runs Unit tests. This step must be passed for PR before its merge.

Systems Integration tests(on Travi-CI) do not run on PRs and most likely a PR will be retargetted at another branch so maintainers can test the code before merging it into `develop` branch.

## PR Checklist

Before creating a pull request, make sure that your PR

- Has the HEAD commit from `develop` branch

- Has a clear message saying why this PR

- References/Explains any related issues

- Has updated changelog in `CHANGELOG.md` file

## Tests

Unit Tests and Systems Integration Tests are meant to be examples for our users of how to interact with the Bot API. It is necessary for test methods to be highly readable, clear on their intents, and show expected behaviour of both systems.

If commits in PR contain any changes to tests, ensure:

- Types are explicitly declared (no use of `var` keyword).

- If possible, method calls to `ITelegramBotClient` have argument names explicitly mentioned

## Code Style

### Bot API Requests

All requests to Telegram Bot API are represented by classes derived from `RequestBase<TResult>`. Required properties of a request must be get-only with value assigned in the constructor.

If a request class (and its accompanying method on `ITelegramBotClient`) accepts a collection, the type must be `IEnumerable<T>`. Also, return types of JSON array responses will be `TResult[]`.

For example, here is a request with required `allowedUpdates` and optional `offset` parameters that returns a JSON array as result:

```c#

Task<Update[]> GetUpdatesAsync(

IEnumerable<string> allowedUpdates,

int offset = default

);

```

```c#

class GetUpdatesRequest : RequestBase<Update[]> {

IEnumerable<string> AllowedUpdates { get; }

int Offset { get; set; }

public GetUpdatesRequest(IEnumerable<string> allowedUpdates)

: base("getUpdates")

{ AllowedUpdates = allowedUpdates; }

}

```

## Related Documents

- [Change Logs](./CHANGELOG.md)

- [Systems Integration Tests - How To](./docs/wikis/tests/sys-integ-tests.md)