**What is QuickCoin?**

QuickCoin is an experimental digital currency that enables instant payments to anyone, anywhere in the world.QuickCoin uses peer-to-peer technology to operate with no central authority: managing transactions and issuing money are carried out collectively by the network. QuickCoin Core is the name of open source software which enables the use of this currency.

**License**

QuickCoin Core is released under the terms of the MIT license. See COPYING for more information or see <https://opensource.org/licenses/MIT>.

**Development Process**

The master branch is regularly built and tested, but is not guaranteed to be completely stable. Tags are created regularly to indicate new official, stable release versions of QuickCoin Core.

The contribution workflow is described in CONTRIBUTING.md.

The developer mailing list should be used to discuss complicated or controversial changes before working on a patch set.

Developer IRC can be found on Freenode at #QuickCoin-core-dev.

**Testing**

Testing and code review is the bottleneck for development; we get more pull requests than we can review and test on short notice. Please be patient and help out by testing other people's pull requests, and remember this is a security-critical project where any mistake might cost people lots of money.

**Automated Testing**

Developers are strongly encouraged to write unit tests for new code, and to submit new unit tests for old code. Unit tests can be compiled and run (assuming they weren't disabled in configure) with: make check. Further details on running and extending unit tests can be found in /src/test/README.md.

There are also regression and integration tests of the RPC interface, written in Python, that are run automatically on the build server. These tests can be run (if the test dependencies are installed) with: qa/pull-tester/rpc-tests.py

The Travis CI system makes sure that every pull request is built for Windows, Linux, and OS X, and that unit/sanity tests are run automatically.

**Manual Quality Assurance (QA) Testing**

Changes should be tested by somebody other than the developer who wrote the code. This is especially important for large or high-risk changes. It is useful to add a test plan to the pull request description if testing the changes is not straightforward.

**Translations**

Changes to translations as well as new translations can be submitted to QuickCoin Core's Transifex page.

Translations are periodically pulled from Transifex and merged into the git repository. See the translation process for details on how this works.

**Important：**

We do not accept translation changes as GitHub pull requests because the next pull from Transifex would automatically overwrite them again.

Translators should also subscribe to the mailing list.

**technical adviser**

technical adviser WeChat:503771144