

# HTTP Notification Provider Contest

Contest for the development and implementation of the HTTP Notification Module for external applications and services. This module should have an ability to send notifications via HTTP protocol.

## Motivation

Free TON holders need a module that provides notifications transmission via the HTTP protocol for interactive applications, online stores, IOT and other consumers. At the same time, anonymity of the blockchain users must be ensured.

## Timing

Submission period: 15 September 2021 - 15 October 2021 23:59 UTC

Voting (assessing) period: 15 days

## General architecture

In order to ensure the anonymity of blockchain users, a separation has been made between the blockchain data and the addresses of the recipients of this data. For this, the following modules are introduced:

- **Queue Provider** - knows what to send (data itself). It doesn't have any information about the real world address of the recipient. It allows the user to configure an event source based on the following parameters: "Account address" and its message types: internal / external In / external Out

Queue provider forwards prepared and encrypted messages to *Notification provider*. Each message contains a key by which the *Notification provider* can match the corresponding recipient

- **Notification Providers** - knows where to send (recipient real world address like IP and port, e-mail, APN ID, FCM ID, etc). It doesn't have any information about the data. It receives and sends the data encrypted.

It could be possible to have several types of Notification Providers depending on the type of recipient and the transport (browser, http-server, smartphone devices, e-mail, etc.)

This contest is about *HTTP Notification Provider Module* or in short, HTTP Notification Module.

*HTTP Notification Module* sends http requests with blockchain events to the registered consumer.

The Http Notification module provides users with the ability to configure itself via REST API.

*HTTP Notification Module's* possible consumers are online stores, external web-services, telegram bots, vkontakte, and any service with Internet connection and external access from the Internet. This means the consumer requirements include the presence of an http-server to receive push-notifications.

## General requirements:

1. Availability of HTTP API methods.
  - 1.1. Add a unique identifier and notification parameters to the internal database
  - 1.2. Get the configuration - optional
    - 1.2.1. Module information (name, description, logo, surf address - to be able to reach out to service developers for support).
    - 1.2.2. Get the structural input parameters for the current module.
2. All HTTP API methods must return a 200 response if the requested operation is successful and corresponding HTTP error code otherwise.
3. http-server with some UI (telegram bot, web page, etc) should be provided to test the work of the module

## Requirements for the HTTP Notification Module:

1. Guaranteed delivery of notifications within N time (for example, 1-24 hours) and repeated delivery of notifications if the delivery address is unavailable.
2. Support for HTTPS protocol
3. When adding a new URL address, verification of the ability to manage a domain, website or a specific url address should be performed by the person, requesting to send notifications to this address
4. Logging of events of http notifications for the possibility of displaying them in charts
5. Availability of documentation with usage examples.
6. Compiling, building, deploying, running and testing instructions with prerequisites.

## Parameters for the HTTP module:

- URL (the line starts with https://)
- Method (GET, PUT, POST, ...) (optional parameter, by default it's POST)
- Query (a parameter line), optional parameter, by default it's "param"

## Queue Provider API










API of the Queue Provider which could be used to get blockchain events stream is described in the following document:


<https://tonlabs.notion.site/Notification-provider-onboarding-3dd961bce8954d0da80208b9a908c773>

## Evaluation criteria

1. Compliance with the technical requirements provided in this contest description.
2. The quality of the documentation description for the module.
3. Easy to set up and simulate.
4. Operates in accordance with the terms of reference and the declared functions.
5. Cross-platform.
6. Source code (open source, Free Software licence).
7. Apart from uploading a submission, a code should be submitted in accordance with <https://github.com/freeton-org/readme>.

## Reward & Vesting

- 1st place -  100'000 TONs
- 2nd place -  75'000 TONs
- 3rd place -  50'000 TONs
- 4rd place -  40'000 TONs
- 5rd place -  30'000 TONs
- 6rd place -  20'000 TONs
- 7rd place -  10'000 TONs
- 8rd place -  5'000 TONs
- 9rd place -  3'000 TONs

- 10rd place -  1'000 TONs

Rewards up to 10K will be paid at the end of the contest. Rewards above 10K will be paid as follows: half at the end of the competition and half in equal parts over 12 months (vesting). The conditions for obtaining the vesting are as follows:

- Github issues should be responded to within 24 hours.
- Critical module malfunctions should be fixed within 3 days.
- In the case of a Queue Provider API changes or other blockchain changes, the code must be updated no later than within 1 month after the change.
- All other adequate issues should be resolved within one month.

## Jury

- Jury members whose team (s) intend (s) to take part in this contest by submitting materials will lose their right to vote in this contest.
- Each member of the jury will vote on a scale of 1-10 for each submission, or he can reject it if it doesn't meet the requirements, or he can abstain from voting if they consider themselves unqualified judges.
- The jury will provide feedback on your works.
- To avoid inconsistency in the counting of votes, each juror must either vote for all entries or abstain.
- Duplicates, modifications of other works that don't meet the requirements, as well as incomplete or inappropriate works will be rejected.
- The voting period is 15 days.

## Jury reward

An amount equal to 10% of the total of all tokens actually awarded to the winners will be divided equally among all jury members who voted and provided feedback. Mandatory voting for all proposals and feedback are required to receive this reward.

## Procedural requirements

- only 1 application is accepted from each team.

- Submitted works must not be a modified version of another work.
- All works must be available for opening and viewing by the jury, so double check your work. If the work is not available or doesn't meet the described criteria, the work may be rejected by the jury.
- Participants of the contest must submit their work before the deadline for accepting applications. If the work is not submitted on time, it will not be counted.
- If the submitted work contains links to the work performed, the content of these links must contain the participant's contact information, preferably Telegram ID, so that the jury members can compare them and check who owns the work. Otherwise, your work may be rejected.
- Jury voting should follow [DevEx global proposal requirements](#)

## Defense of contest submissions

At the end of the submission acceptance period, AMA-session will be appointed for participants, jurors and everyone else. At this session, each contestant team has to present their work.

The presentation language is English or Russian. The presentation time should not exceed 10-20 minutes.

If a contestant cannot present the work on-line, they should make a video recording and publish it on YouTube but any questions that arise should be answered.

## Governance rewards

An amount equal to 2% of the prize fund will be allocated to members who participated in organizing the contest, to be distributed equally among them:

- @EkaterinaPantaz
- @webcounters
- @Temofab
- @anovi

The same percentage of the monthly reward will be received by the persons responsible for its distribution (vesting distribution overhead).

## Contest announcement and attracting new members rewards

An amount equal to 5% of the prize fund will be allocated to announcing partners who participates in announcing the contest in different media according with the following table: media list for technical contests announcements 6, to be distributed equally among them:

- @anovi
- @Alex770
- @lesnik13utsa
- @Kronchs

Each participant of the contest, when submitting an application, will be asked through which announcing partner he/she learned about the contest. After the end of the contest, for each participant who won a reward, an amount equal to 5% of his/her reward will additionally be distributed:

- To the announcing partner who attracted him, if the referral was given during work submission;
- Or equally to all aforementioned partners of the announcement program, if the referral was not specified.