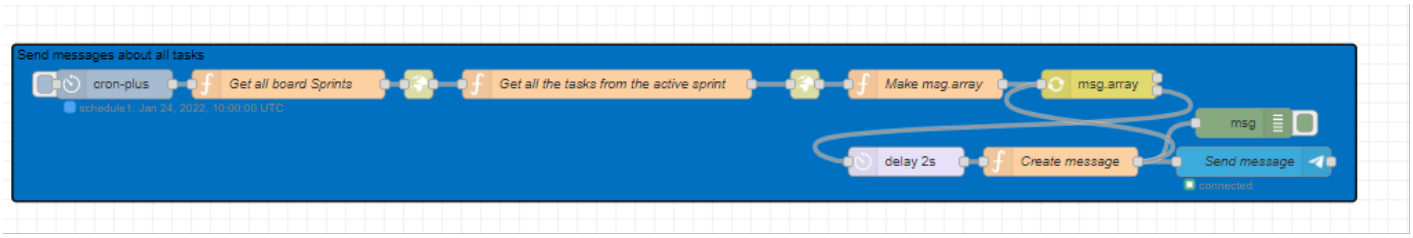
Jira Telegram Notification

Даний кейс складається з двох частин: 1) Отримання кожного дня у визначений час повідомлення про статус кожного завдання з активного спринту; 2) Отримання сповіщення при закріпленні працівника за завданням. Даний кейс несе інформативний характер, який сприятиме покращенню менеджменту проекту.

- Відправлення повідомленя про статус завдань активного спринту
- Отримання сповіщення при закріпленні працівника за завданням

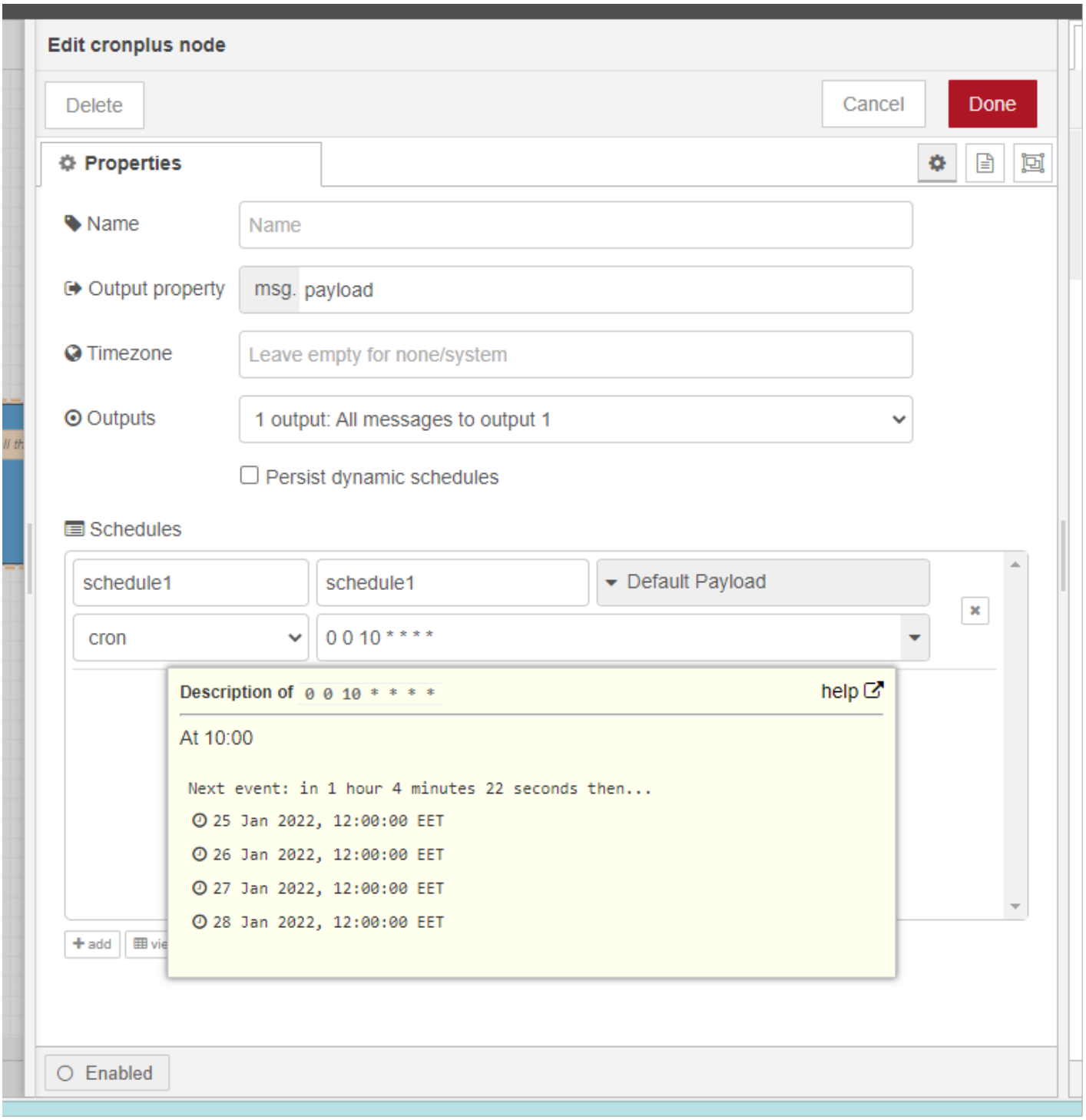
Відправлення повідомленя про статус завдань активного спринту

Для відправлення повідомлення про статус завдань з активного спринта ми створюємо потік в середовищі Node Red, в якому здійнюється два Http Request: 1) для отримання всіх спринтів з певної дошки; 2) Для отримання всіх завданнь з активного спринта. Далі ми збираємо всю потрібно нам інформацію і відправляємо повідомлення в телеграм за допомогою телеграм бота. Даний потік складається з наступних вузлів:



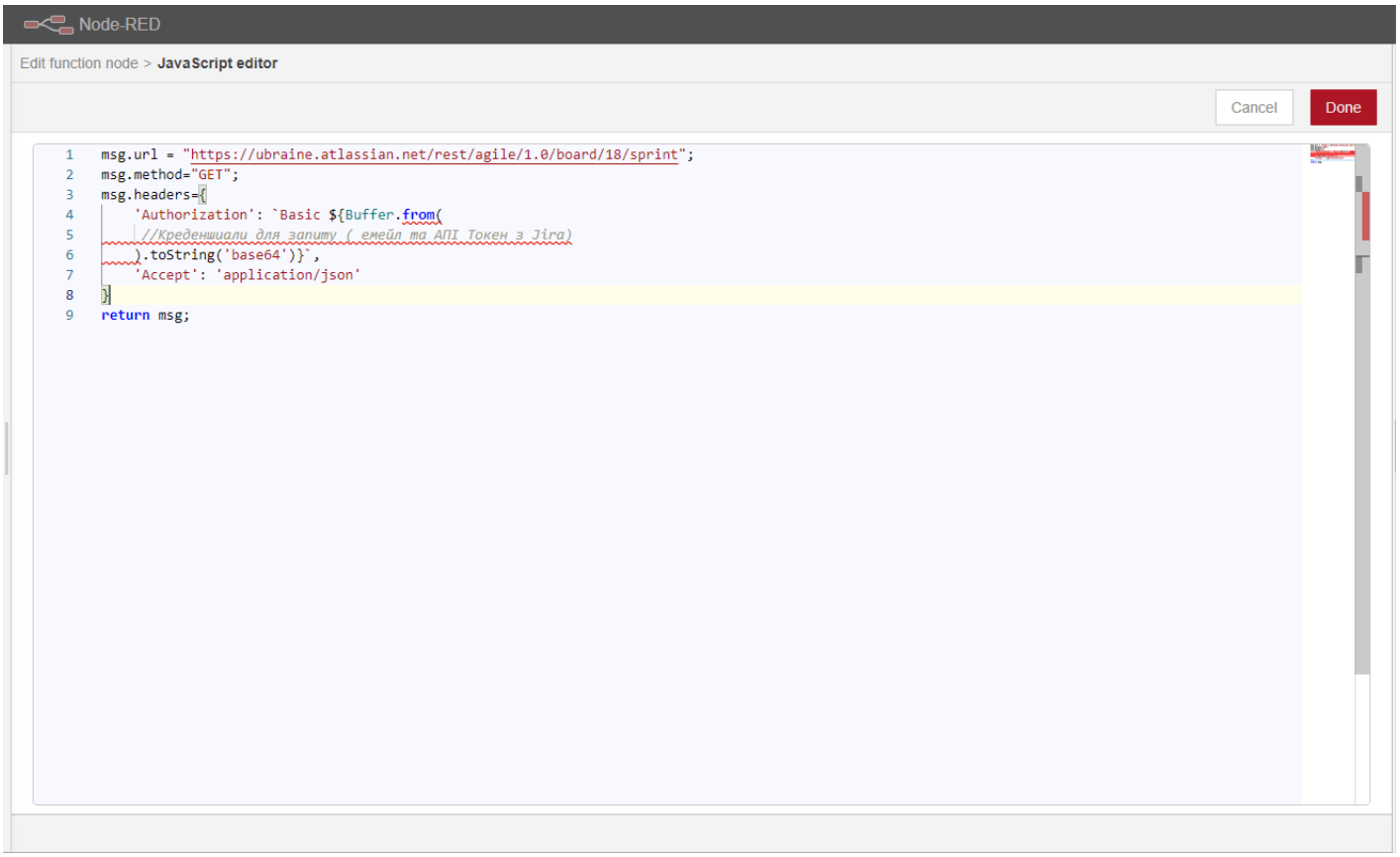
- Cron plus;
- 4 Function;
- 2 Http request;
- array-loop;
- delay;
- telegram sender;
- debug

Вузол `Cron Plus:`



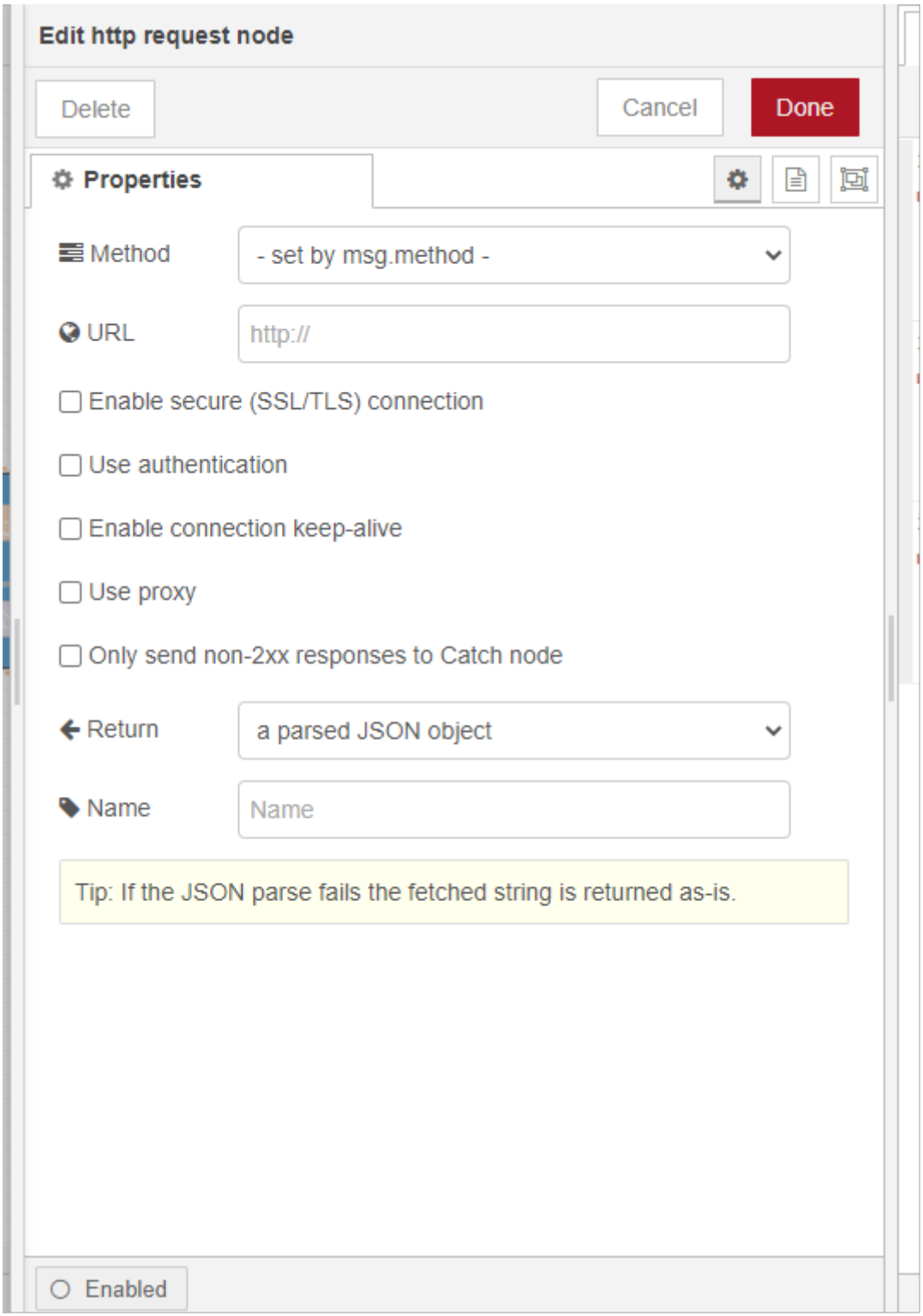
В налаштуваннях цього вузла ми обрали щоб він запускав потік кожного дня об 12й годині ранку.

Вузол `function` **(Get all board Sprints):**



Тобто це дані, які ми відправлємо через вузол `Http Request` і у відповідь отримуємо всі спринти з вказаного **board (18)**

Вузол `http request:`



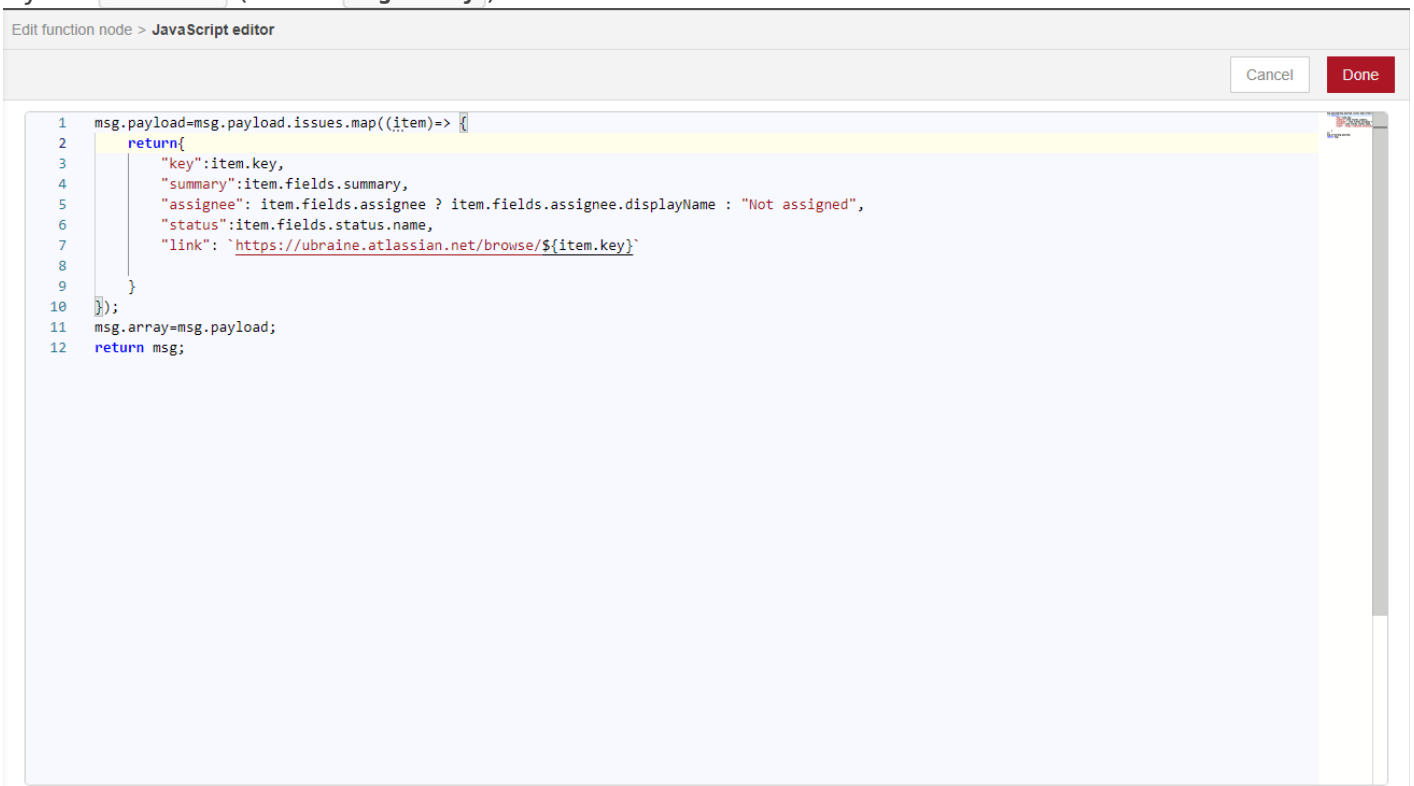
Вузол `function` (**Get all the tasks from the active sprint**):



Спочатку ми фільтруємо всі отримані спринти, знаходимо активний, та формує новий запит, який отримає всі завдання з даного спринта.

`msg.payload[0].self` - це посилання на даний спринт.

Вузол `function` (**Make** `msg.array`):



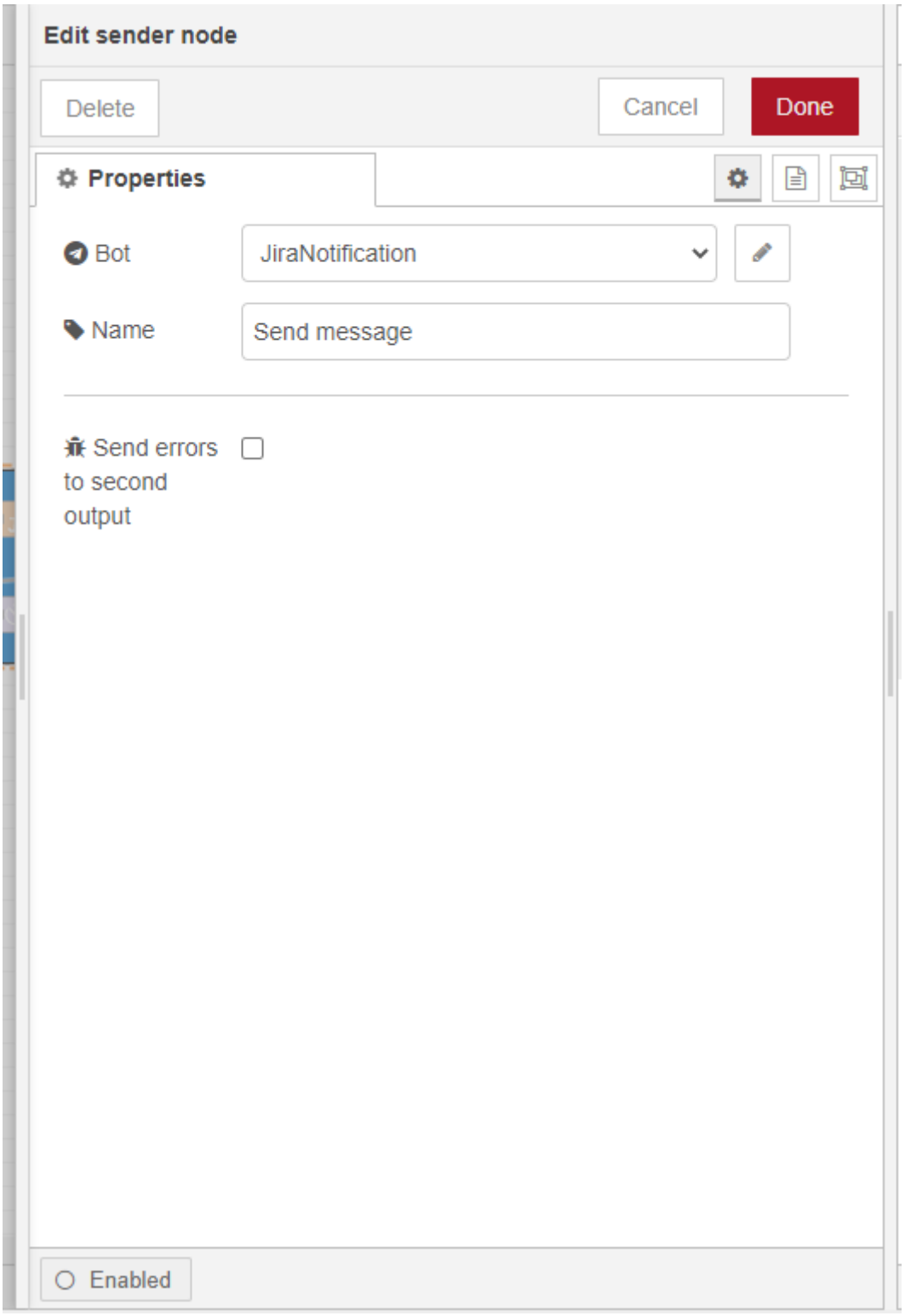
Тобто ми збираємо з отриманого масиву завдань потрібну нам інформацію, і записуємо новий масив в змінну msg.array.

За допомогою вузла `array-loop` (**msg.array**) здійснюється цикл, який відправляє в наступні вузли по одному елементу з масиву msg.arry, після повторного виклику цього вузла

відправляється наступний елемент.
Вузол `delay` - створює затримку між відправлення даних в наступний вузол (в даному випадку це буде 2 секунди).
Вузол `function` (**Create message**) - формує повідомлення яке відправлятиметься у телеграм.



Вузол `telegram sender` (**Send message**) - здійснює відправлення повідомлення у телеграм. В налаштуваннях цього вузла потрібно лише добавити телеграм бота, вказавши йому ім'я та ввівши токен бота.

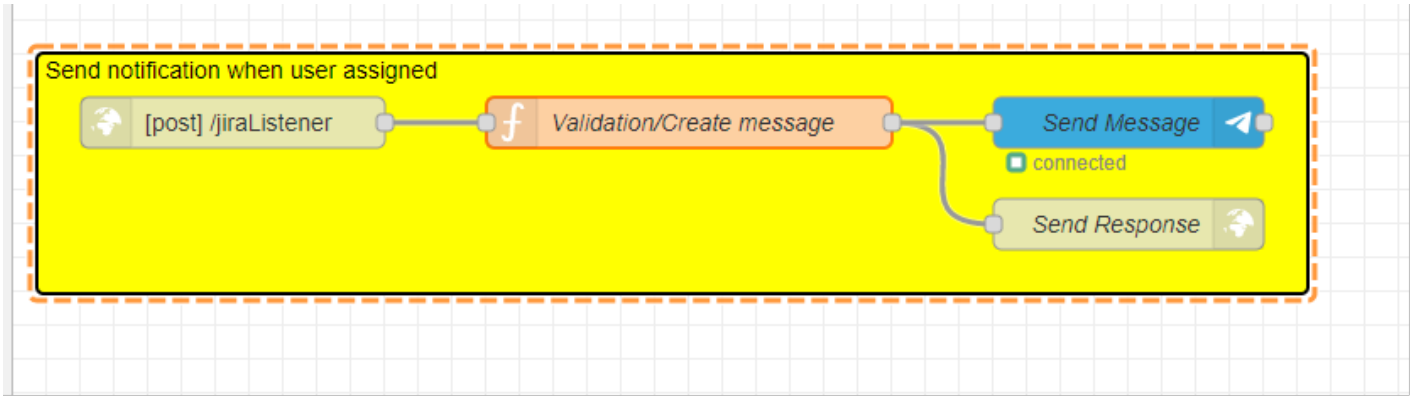


Отже, даний потік здійснює два Http запити, за допомогою яких отримує всі завдання з активного спринта, і по черзі по кожному завданню відправляє повідомлення в телеграм, в казуючи саммарі завдання, посилання та статус.

Отримання сповіщення при закріпленні працівника за завданням

Для отримання сповіщення при закріпленні працівника за завданням нам потрібно створити потік в середовищі Node Red, та створити **веб хук** в **Jira**, в якому вказати посилання на вузол `http in` нашого потоку.

Потік в середовищі Node Red:



Даний потік складається з наступних вузлів:

- http in;
- function;
- telegram sender;
- http response.

Вузо `function` (**Validation/Create message**):



Веб хук з **Jira** відправляє дані в наш потік кожного разу коли оновлюється інформація про завдання, в тому числі і закріплення працівника за завданням. В вузлі **function** ми спочатку перевіряємо чи це потрібний нам тип оновлення інформації про завдання, а після цього формує повідомлення для відправки.

Вузол `telegram sender` - здійснює відправлення повідомлення.

Вузол `http response` - відсилає відповідь в Jira.