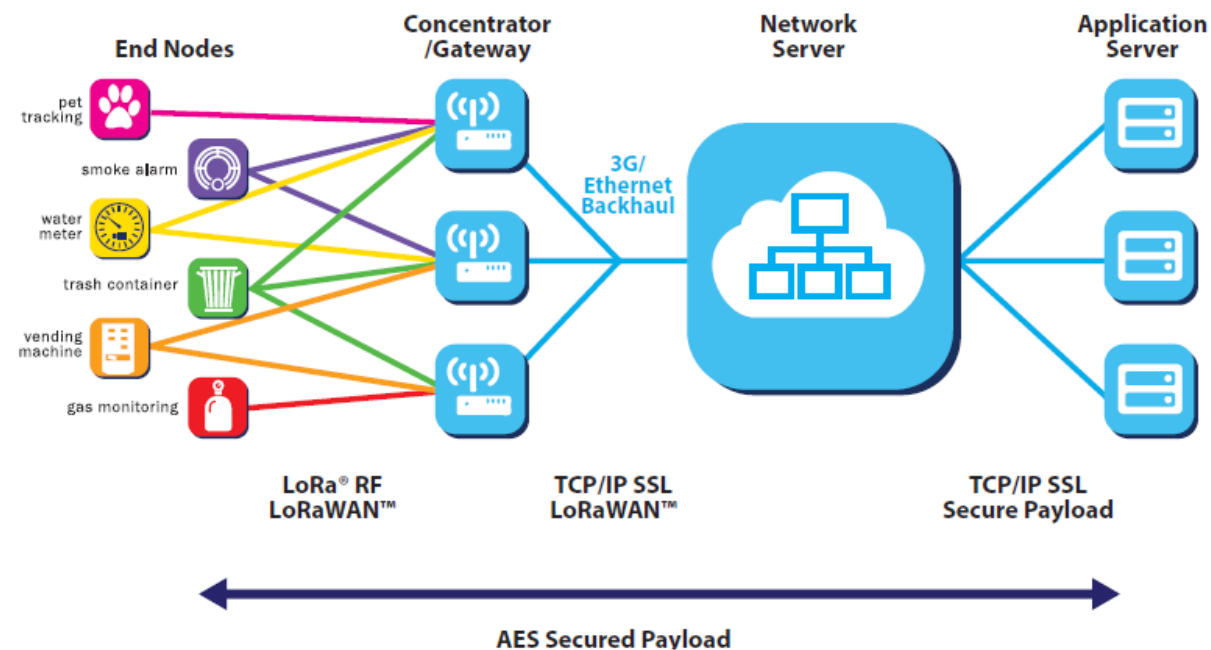


LoRaWAN



LoRaWAN

- LoRaWAN spada u klasu LPWAN komunikacionih tehnologija
- Najvažniji elementi sistema:
 - **Krajnji uređaj, Senzorski uređaj, senzorski čvor (eng. End Device, Node, Mote)** - uređaj sa niskom potrošnjom energije opremljen komunikacionim modulom
 - **Koncentrator (eng. Gateway)** - uređaj koji komunicira sa senzorskim čvorovima i prosleđuje poruke od i ka mrežnom serveru
 - **Mrežni server (eng. Network Server)** - računar koji služi kao sprega između koncentratora i aplikacionih servera
 - **Aplikacioni server (eng. Application Server)** - server na kojem se nalazi korisnička aplikacija
 - **Uplink poruka** - poruka koja se prosleđuje od senzorskog čvora ka aplikaciji
 - **Downlink poruka** - poruka koja se prosleđuje od aplikacije ka senzorskom čvoru



LoRaWAN parametri*

- LoRaWAN koristi više različitih brojeva za identifikaciju čvorova, koncentratora i aplikacija:
 - **DevEUI**: 64-bitni identifikator senzorskog čvora (jedinствен)
 - **DevAddr**: 32-bitna adresa uređaja (ne mora biti jedinstvena)
 - **AppEUI**: 64-bitni identifikator aplikacije
 - **GatewayEUI**: 64-bitni identifikator koncentratora (jedinствен)
- Prilikom bežične komunikacije, koristi se dinamička 32-bitna adresa uređaja (DevAddr), koja ima fiksni 7-bitni prefiks vezan za mrežu (0x26 ili 0x27 u slučaju TTN mreže)
- Preostalih 25 bita dodeljuju se uređaju prilikom procesa aktivacije
- Postoje dva načina aktivacije:
 - **ABP** (eng. Activation By Personalization)
 - **OTAA** (eng. Over-The-Air Activation)

**Ovi parametri su aktuelni za verziju protokola 1.0.3, koja je implementirana u okviru biblioteke LMIC*

Aktivacija LoRaWAN uređaja

ABP (Activation By Personalization)

- U firmware uređaja se unapred upisuju parametri, koji su ujedno poznati i mreži, odnosno aplikaciji:
 - Adresa uređaja (**DevAddr**)
 - Ključ za enkripciju u bežičnoj mreži (**NwkSKey**)
 - Ključ za enkripciju podataka između čvora i aplikacije (**AppSKey**)
- Nema potrebe za razmenom parametara putem "rukovanja"
- Uređaj je spreman za komunikaciju bez ikakve dodatne procedure

OTAA (Over-The-Air Activation)

- Zasniva se na globalno jedinstvenom identifikatoru uređaja
- Obavlja se bežično "rukovanje" (eng. Handshaking) između senzorskog čvora i mreže
- Poruke koje se razmenjuju tokom "rukovanja" u bežičnoj mreži, zaštićene su aplikacionim ključem (**AppKey**)
- Čvor emituje identifikator uređaja (**DevEUI**) i aplikacije (**AppEUI**) u okviru tzv. zahteva za pridruživanjem (eng. Join Request)
- Koncentrator šalje odgovor (eng. Join Accept) koji sadrži:
 - Dinamički dodeljenu adresu uređaja (**DevAddr**)
 - Informaciju pomoću koje senzorski uređaj izvodi ključ za enkripciju u bežičnoj mreži (**NwkSKey**) i ključ za enkripciju podataka između čvora i aplikacije (**AppSKey**)

The Things Network (TTN)

- **The Things Network (TTN)** je najveća i najpopularnija svetska open-source LoRaWAN mreža
- Korišćenje mreže je besplatno u varijanti *The Things Stack – Community Edition*, uz registrovanje korisničkog naloga:
<https://account.thethingsnetwork.org/>
- Korisnici imaju mogućnost da sami doprinesu izgradnji mrežne infrastrukture registrovanjem gejtujeja na mrežu
- Tehnička dokumentacija i tutorijali:
<https://www.thethingsnetwork.org/docs/>



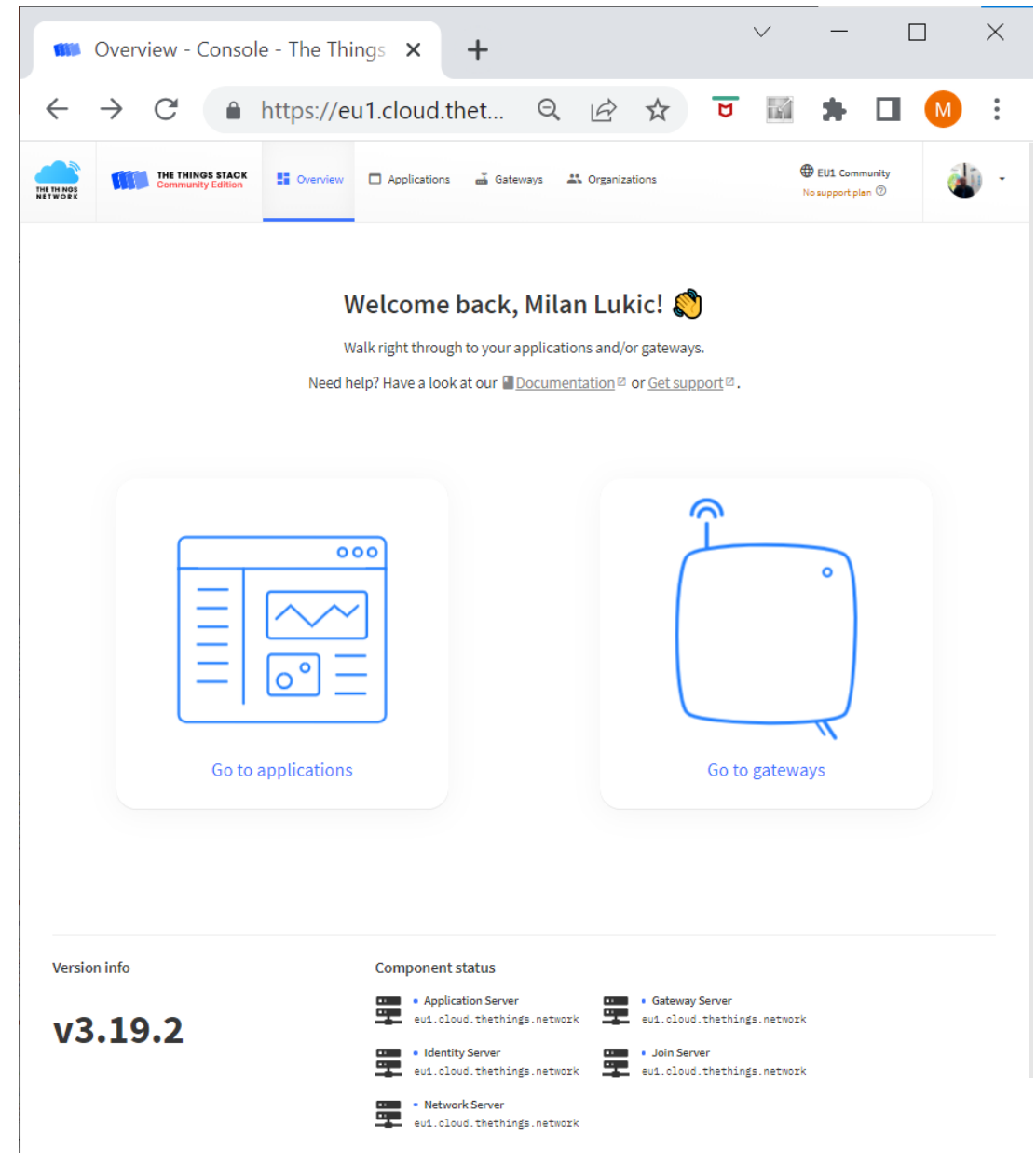
Gejtveji i gde ih naći?

- Neophodan preduslov za korišćenje LoRaWAN mreže je da se u dometu ređaja nalazi bar jedan aktivan gejtvej
- U slučaju gejtveja koji je za unutrašnju upotrebu (npr. u stanu korisnika), u urbanim uslovima domet emitovanja dostiže do nekoliko stotina metara
- Mapa sa trenutnim rasporedom gejtveja može se videti na stranici <https://www.thethingsnetwork.org/map>



Kreiranje TTN aplikacije

- Nakon kreiranja korisničkog naloga i logovanja na sistem, potrebno je ući u konzolu (<https://console.cloud.thethings.network/>), izabrati klaster *Europe 1*, a zatim opciju *Go to applications -> Add application*
- Aplikacija u okviru LoRaWAN mreže predstavlja logički entitet za grupisanje uređaja
- Po ulasku u novokreiranu aplikaciju, opcija *Add end device* koristi se za dodavanje novog uređaja. Na raspolaganju su dodavanje nekog od predefinisanih tipova uređaja (*From the LoRaWAN Device Repository*) ili opcija ručnog dodavanja (*Manually*) koja će biti ovde demonstrirana



Definisanje parametara novog uređaja

- **Frequency plan:** Europe 863-870 MHz (SF9 for RX2 – recommended)
- **LoRaWAN version:** 1.0.3
- **DevEUI:** moguće je ručno uneti proizvoljnu vrednost ili dozvoliti mreži da ga automatski generiše, pri čemu je broj pokušaja ograničen na 50
- **AppEUI (tj. JoinEUI u novijim verzijama protokola):** moguće je ručno uneti, ili popuniti nulama (Fill with zeros)
- **AppKey:** Preporučljivo je automatski generisati vrednost
- **End device ID:** Ova vrednost nije deo protokola, ali je preporučljivo izabrati ime koje će korisniku biti lako razumljivo i prepoznatljivo

Register end device - Umrezeni E x +

https://eu1.cloud.thet...

THE THINGS STACK Community Edition

Overview Applications Gateways Organizations EU1 Community No support plan

Umrezeni Embedded Sistemi

Overview End devices Live data Payload formatters Integrations Collaborators API keys General settings

Applications > Umrezeni Embedded Sistemi > End devices > Register manually

Register end device

From The LoRaWAN Device Repository [Manually](#)

Frequency plan ⓘ *

Europe 863-870 MHz (SF9 for RX2 - recommended)

LoRaWAN version ⓘ *

LoRaWAN Specification 1.0.3

Regional Parameters version ⓘ *

RP001 Regional Parameters 1.0.3 revision A

[Show advanced activation, LoRaWAN class and cluster settings](#)

DevEUI ⓘ *

01 23 45 67 89 AB CD EF [Generate](#) 7/50 used

AppEUI ⓘ *

B1 6B 00 B5 60 0D B0 0B [Fill with zeros](#)

AppKey ⓘ *

57 9B 82 27 CF C6 36 D2 C5 5C E6 0C E6 9B 02 30 [Generate](#)

End device ID ⓘ *

ues-01

This value is automatically prefilled using the DevEUI

After registration

☒ View registered end device

☐ Register another end device of this type

[Register end device](#)

< Hide sidebar

© 2022 The Things Stack by The Things Network and The Things Industries EN v3.19.2 Documentation [Get support](#) [Status page](#)

ESP32 aplikacija

- Unutar Arduino IDE neophodno je instalirati biblioteku ***MCCI LoRaWAN LMIC library***
- Nakon instalacije, potrebno je pronaći konfiguracioni fajl unutar foldera biblioteke na sledećoj putanji:
“*.../libraries/MCCI_LoRaWAN_LMIC_library/project_config/lmic_project_config.h*”.
- U tom fajlu treba izmeniti definicije regiona tako da odgovaraju evropskim frekvencijama:

```
#define CFG_eu868 1  
//#define CFG_us915 1  
#define CFG_sx1276_radio 1
```

- Aplikacija se bazira na primeru *Examples -> MCCI LoRaWAN LMIC library -> ttn-otaa*
- ***ttn-otaa.ino***

ESP32 aplikacija

- U programskom kodu treba izmeniti nizove APPEUI i DEVEUI, tako što se unesu vrednosti u inverznom poretku (little endian) u odnosu na vrednost koja se čita iz TTN konzole

- **PRIMER:**

DevEUI

0xEF, 0xCD, 0xAB, 0x89, 0...

lsb



<>



```
static const u1_t PROGMEM DEVEUI[8]={ 0xEF, 0xCD, 0xAB, 0x89, 0x67, 0x45, 0x23, 0x01 };
```

- Za razliku od APPEUI i DEVEUI, APPKEY je jedini parametar koji je potrebno uneti ne u inverznom, nego u direktnom poretku (big endian)

Aktivacija uređaja i slanje uplink poruke

```
COM16
284159: engineUpdate, opmode=0xc
284196: TXMODE, freq=868100000, len=23, SF=7, BW=125, CR=4/5, IH=0
600658: RXMODE_SINGLE, freq=868100000, SF=7, BW=125, CR=4/5, IH=0
605126: EV_JOINED
605129: engineUpdate, opmode=0x808
605171: TXMODE, freq=867100000, len=26, SF=7, BW=125, CR=4/5, IH=0
921632: RXMODE_SINGLE, freq=867100000, SF=7, BW=125, CR=4/5, IH=0
924167: Received downlink, window=RX1, port=-1, ack=1
924171: EV_TXCOMPLETE (includes waiting for RX windows)
Received ack
```

☒ Autoscroll ☐ Show timestamp Both NL & CR 115200 baud Clear output

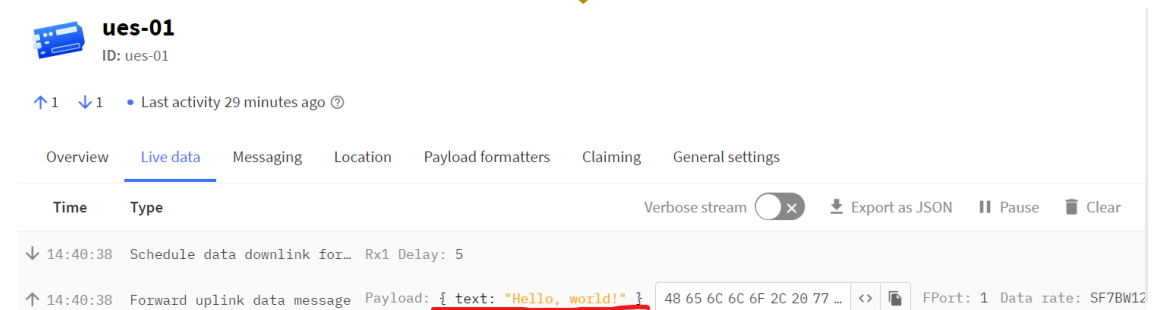
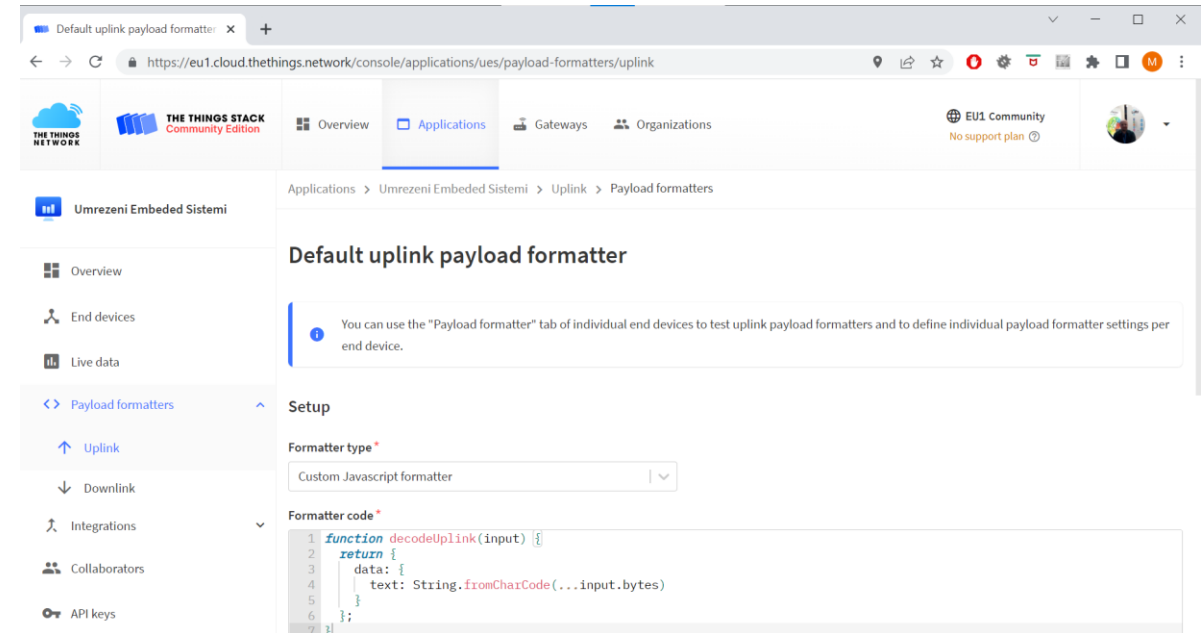
- Nakon pokretanja programa na ESP32, u konzoli aplikacije (*End devices -> End device ID -> Live data*), vidljive su sledeće poruke:
 - Zahtev za aktivacijom i potvrda aktivacije
 - Uplink poruka („Hello, world!“), vidljiva u formi hex stringa
 - Downlink poruka (ACK)

The screenshot shows the 'Live data' page for device 'ues-01' in the The Things Network console. The left sidebar contains navigation links: Overview, End devices, Live data, Payload formatters, Uplink, Downlink, Integrations, Collaborators, and API keys. The main content area shows the device 'ues-01' with its ID 'ues-01' and last activity '18 minutes ago'. Below this, there are tabs for Overview, Live data (selected), Messaging, Location, Payload formatters, Claiming, and General settings. The 'Live data' tab displays a table with columns 'Time' and 'Type'. The table contains several entries: a downlink message at 14:11:25 to schedule a data downlink, an uplink message at 14:11:25 for a forward uplink data message with a MAC payload of 48 65 6C 6C 6F 2C 20 77, an uplink message at 14:11:25 for a successfully processed data message, and an uplink message at 14:11:21 for a forward join-accept message. Below the table, there are two console messages: '14:11:20 Accept join-request' and '14:11:15 Console: Events cleared The events list has been cleared'. The top of the console shows the URL 'https://eu1.cloud.thethings.network/console/applications/ues/devices/ues-01/data' and the 'THE THINGS STACK Community Edition' logo.

Dekodovanje uplink poruke na strani TTN aplikacije

- Poruka koju šalje senzorski čvor može biti u različitim formatima (tekstualni, binarni, JSON itd.)
- U okviru aplikacije, opcija *Payload formatters* -> *Uplink* -> *Formatter type* -> *Custom Javascript formatter* pruža mogućnost dekodovanja primljene poruke pomoću JS funkcije

```
function decodeUplink(input) {  
  return {  
    data: {  
      text: String.fromCharCode(...input.bytes)  
    }  
  };  
}
```



Slanje i obrada downlink poruka

- Downlink poruke mogu se generisati na 2 načina:
 - Iz TTN konzole
 - Eksterno, korišćenjem API poziva
- Usled uplink-centrične prirode LoRaWAN protokola, downlink poruka se ne šalje u trenutku generisanja, nego se smešta u FIFO red koji održava mrežni server. Slanje downlink poruke uvek neposredno prati prijem sledeće uplink poruke
- Kako bi senzorski uređaj prikazivao sadržaj primljenih downlink poruka, potrebno je modifikovati kod unutar funkcije **onEvent**, kao što je prikazano u listingu desno
- Pored toga, dodat je kod kojim se logika periodičnog slanja uplink poruka zamenjuje čekanjem na pritisak tastera PRG na ploči

```
case EV_TXCOMPLETE:
    Serial.println(F("EV_TXCOMPLETE (includes waiting for RX
windows)"));
    if (LMIC.txrxFlags & TXRX_ACK)
        Serial.println(F("Received ack"));
    if (LMIC.dataLen)
    {
        Serial.print(F("Received "));
        Serial.print(LMIC.dataLen);
        Serial.print(F(" bytes of payload: "));

        for(int i = 0; i < LMIC.dataLen; i++)
            Serial.printf("%02X ", LMIC.frame[LMIC.dataBeg + i]);
        Serial.println();
    }
    // Trigger next transmission
    Serial.println("Press PRG button to transmit...");
    while (digitalRead(0));
    os_setTimedCallback(&sendjob, os_getTime()+sec2osticks(1), do_send);
    break;
```

Kreiranje downlink poruke iz TTN konzole

- Downlink poruka se kreira pomoću opcije *Messaging -> Downlink*
- Opcija *Insert Mode -> Replace downlink queue* obezbeđuje da novokreirana poruka bude jedina u redu za slanje
- *Fport* predstavlja port na koji se upućuje poruka; dozvoljena je proizvoljna vrednost u opsegu od 1 do 223
- Opcijom *Payload type* bira se jedan od dva raspoloživa tipa poruke, u ovom primeru u pitanju je binarni string koji se zadaje u formi niza heksadecimalnih okteta
- **Važna napomena:** izbegavati korišćenje opcije *Confirmed downlink*, pošto FW na ESP32 uređaju nije u stanju da izađe na kraj sa njom 😊

Overview Live data **Messaging** Location Payload formatters Claiming General settings

Uplink **Downlink**

Schedule downlink

Insert Mode

☒ Replace downlink queue

☐ Push to downlink queue (append)

FPort *

42

Payload type

☒ Bytes ☐ JSON

Payload

BE EF F0 0D

The desired payload bytes of the downlink message

☐ Confirmed downlink

Schedule downlink

**ues-01**

ID: ues-01

↑ 3 ↓ 3 • Last activity 17 seconds ago ?

[Overview](#) [Live data](#) [Messaging](#) [Location](#) [Payload formatters](#) [Claiming](#) [General settings](#)

Time	Type	Data preview	Verbose stream	Export as JSON	Pause	Clear
↓ 19:49:04	Schedule data downlink for...	FPort: 42 MAC payload: 15 A7 04 BB <> Rx1 Delay: 5	☐	Export as JSON	Pause	Clear
↑ 19:49:04	Forward uplink data message	Payload: { text: "Hello, world!" } 48 65 6C 6C 6F 2C 20 77 ... <> FPort: 1 Data rate: SF7BW125 SNR: 7.5 RSSI: -7	☐	Export as JSON	Pause	Clear
↑ 19:49:04	Successfully processed dat...					
↓ 19:48:52	Receive downlink data mess...	FPort: 42 Payload: BE EF F0 0D <>	☐	Export as JSON	Pause	Clear



TTN konzola prilikom slanja downlink poruke

Serijski terminal prilikom slanja downlink poruke



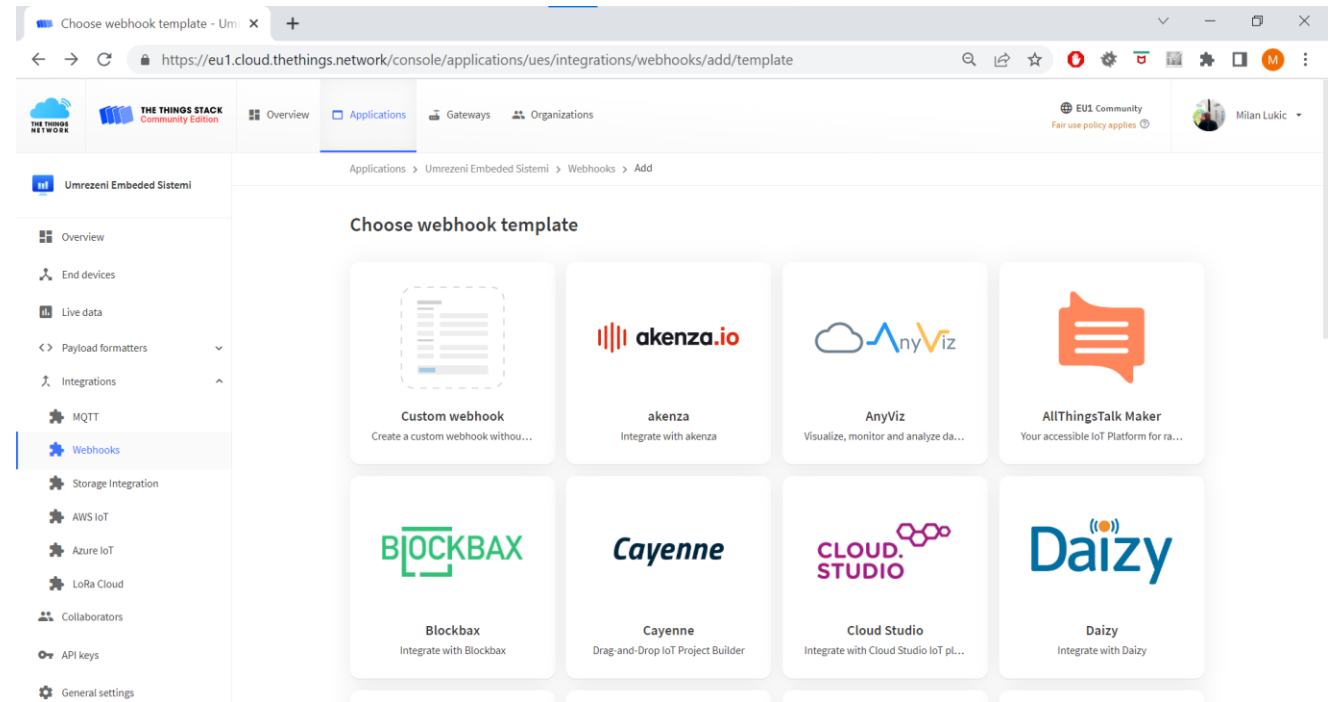
COM16

5074289: engineUpdate, opmode=0x800
Send, txCntrl: 3
5136790: engineUpdate, opmode=0x808
5136831: TXMODE, freq=868100000, len=26, SF=7, BW=125, CR=4/5, IH=0
Packet queued
5453293: RXMODE_SINGLE, freq=868100000, SF=7, BW=125, CR=4/5, IH=0
5456150: Received downlink, window=RX1, port=42, ack=1
5456155: EV_TXCOMPLETE (includes waiting for RX windows)
Received ack
Received 4 bytes of payload: BE EF F0 0D
Press PRG button to transmit...

☒ Autoscroll ☐ Show timestamp Both NL & CR 115200 baud Clear output

Integracija TTN aplikacije sa spoljnim serverom

- Pomoću opcije *Integrations* -> *Webhooks* -> *Add webhook* moguće je povezati TTN aplikaciju sa spoljnim serverom ili nekom od IoT platformi (npr. *ThingSpeak* platforma korišćena u vežbi 5)
- Za povezivanje sa spoljnim serverom potrebno je izabrati opciju *Custom webhook*



Parametri integracije

Applications > Umrezeni Embedded Sistemi > Webhooks > Edit

Edit webhook

The Webhooks feature allows The Things Stack to send application related messages to specific HTTP(S) endpoints. You can also use webhooks to schedule downlinks to an end device. Learn more in our [Webhooks guide](#).

General settings

Webhook ID *

ues-server

Webhook format *

JSON

Base URL *

http://199.247.17.15/test.e01

Downlink API key

The API key will be provided to the endpoint using the "X-Downlink-APIkey" header

Request authentication ⓘ

☐ Use basic access authentication (basic auth)

Additional headers

+ Add header entry

Enabled event types

For each enabled event type an optional path can be defined which will be appended to the base URL

☒ Uplink message

/ttn.php

An uplink message is received by the application

☐ Join accept

An end device successfully joins the network and starts a session

☐ Downlink ack

A confirmed downlink is acknowledged by an end device

☐ Downlink nack

A sent confirmed downlink fails confirmation by the end device

☐ Downlink sent

A downlink is sent to an end device or multicast group

☐ Downlink failed

A downlink cannot be sent

☐ Downlink queued

A downlink is added to the downlink queue

☐ Downlink queue invalidated ⓘ

The downlink queue is reset due to frame counter mismatch

☐ Location solved

An integration succeeded locating the end device

☐ Service data

An integration emits an event

Save changes

Delete Webhook

Kreiranje tabele u bazi i HTTP endpointa

- U MySql bazi podataka na serveru neophodno je kreirati tabelu **ttn_log** u koju će biti smeštan sadržaj poruka koje stižu sa TTN servera:

```
CREATE TABLE ttn_log (  
  id int NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  ts timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  payload text NOT NULL  
)
```

Showing rows 0 - 4 (5 total, Query took 0.0004 seconds.)

SELECT * FROM `ttn\_log`

☐ Profiling [Edit inline] [Edit] [Explain SQL]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

		id	ts	payload
☐	Edit Copy Delete	1	2022-06-02 21:13:11	{"end_device_ids":{"device_id":"ues-01","applicati...
☐	Edit Copy Delete	2	2022-06-02 21:13:19	{"end_device_ids":{"device_id":"ues-01","applicati...
☐	Edit Copy Delete	3	2022-06-02 21:13:35	{"end_device_ids":{"device_id":"ues-01","applicati...
☐	Edit Copy Delete	4	2022-06-02 21:13:53	{"end_device_ids":{"device_id":"ues-01","applicati...
☐	Edit Copy Delete	5	2022-06-02 21:14:01	{"end_device_ids":{"device_id":"ues-01","applicati...

- U folder `/var/www/html/prezime.exx` na serveru dodati datoteku **ttn.php**, sledećeg sadržaja:

```
<?php  
$_POST = file_get_contents("php://input");  
  
$server = "localhost";  
$user = "prezime.exx";  
$pass = "...";  
$db = "db_prezime_exx";  
  
$conn = new mysqli($server, $user, $pass, $db);  
if ($conn->connect_error)  
    die("Connection failed: " . $conn->connect_error);  
  
$sql = "INSERT INTO ttn_log (payload) VALUES ('" . $_POST . "')";  
if ($conn->query($sql) === TRUE)  
    echo "Unos uspesno izvršen!" . PHP_EOL;  
else  
    echo "Greska: " . $sql . $conn->error . PHP_EOL;  
  
$conn->close();
```

?>



Zadaci za vežbu

1. Dodati u TTN aplikaciju integraciju sa *ThingSpeak* aplikacijom i prikazati podatke u okviru odgovarajućeg widgeta.
2. Dopuniti PHP endpoint tako da iz sadržaja koji stiže sa TTN servera izdvaja korisnu informaciju (očitavanje senzora). Zatim prikazati grafički u front-endu rezultate poslednjih 10 očitavanja.
3. U okviru web stranice realizovane u prethodnom zadatku, dodati formu koja omogućava korisniku unos periode očitavanja senzora, a zatim po unosu posredstvom TTN API-ja šalje informaciju o periodi na uređaj kao downlink poruku.