

Node.js: validare un indirizzo email universitario

In questo articolo vedremo come validare un indirizzo e-mail universitario in ExpressJS.

Due sono i requisiti di un indirizzo e-mail universitario valido:

1. Deve essere un indirizzo e-mail sintatticamente valido.
2. Deve appartenere ad un dominio universitario valido.

Nel nostro esempio useremo i domini universitari italiani. È bene ricordare tuttavia che se un'università non fa riferimento al suo FQDN principale ma utilizza un dominio separato, la validazione non avrà successo.

Utilizzeremo le API pubbliche raggiungibili all'indirizzo `http://universities.hipolabs.com/search` a cui passeremo il parametro `country` per avere un array JSON con le informazioni su ciascun ateneo ed i suoi domini.

Definiamo due funzioni di utility. La prima effettua la richiesta GET remota alle API, la seconda verifica che il dominio ricavato dall'indirizzo e-mail sia presente.

```
'use strict';

const superagent = require('superagent');

module.exports = {
  apiRequest( country = 'italy' ) {
    const url =
`http://universities.hipolabs.com/search?
country=${country}`;
```

```

        return superagent.get(url);
    },
    isUniDomain(data, domain) {
        const result = data.find(element => { return
element.domains.includes(domain); });
        return result ? true : false;
    }
};

```

La nostra route validerà l'indirizzo e-mail ed estrarrà il dominio di base da tale indirizzo per sottoporlo alla seconda validazione.

```

router.post('/validate', async (req, res, next) => {
    const { email } = req.body;
    let valid = true;

    if(!validator.isEmail(email)) {
        valid = false;
        res.json({ valid });
        return;
    }

    const parts = email.split('@');
    const domainParts = parts[1].split('.');
    const domain = domainParts[domainParts.length -
2] + '.' + domainParts[domainParts.length - 1];

    try {
        const response = await apiRequest();
        const data = JSON.parse(response.text);
        valid = isUniDomain(data, domain);
        res.json({ valid });
    } catch(err) {
        res.json({ err });
    }
}

```

```
    }  
  });  
  
  module.exports = router;
```

Spesso gli account e-mail degli studenti vengono creati su un dominio di terzo livello, quindi nel nostro caso abbiamo dovuto estrarre solo il dominio base e validarlo.

Codice sorgente

GitHub