Solución Desafío - Citas médicas

Requerimiento 1

```
const http = require('http')
const axios = require('axios')
let users = []
http
 .createServer((req, res) => {
   if (req.url.startsWith('/usuarios')) {
       axios.get('https://randomuser.me/api').then((u) => {
       const { first, last } = u.data.results[0].name
       users.push({ first, last})
       })
   }
 })
 .listen(3001, function () {})
```

Requerimiento 2

```
const http = require('http')
const axios = require('axios')
const { v4: uuidv4 } = require('uuid')
let users = []
http
 .createServer((req, res) => {
   if (req.url.startsWith('/usuarios')) {
     axios.get('https://randomuser.me/api').then((u) => {
       const { first, last } = u.data.results[0].name
       users.push({ first, last, id: uuidv4().slice(30)})
      })
    }
 })
 .listen(3001, function () {})
```

Requerimiento 3

```
const http = require('http')
const axios = require('axios')
const { v4: uuidv4 } = require('uuid')
let users = []
const moment = require('moment')
http
 .createServer((req, res) => {
   if (req.url.startsWith('/usuarios')) {
     res.writeHead(200, { 'Content-Type': 'text/html' })
     axios.get('https://randomuser.me/api').then((u) => {
       const { first, last } = u.data.results[0].name
       const timestamp = moment().format('MMMM Do YYYY, h:mm:ss a')
       users.push({ first, last, id: uuidv4().slice(30), timestamp })
     })
   }
 })
 .listen(3001, function () {})
```

Requerimiento 4

```
const http = require('http')
const axios = require('axios')
const { v4: uuidv4 } = require('uuid')
const _ = require('lodash')
let users = []
const moment = require('moment')
http
 .createServer((req, res) => {
   if (req.url.startsWith('/usuarios')) {
     res.writeHead(200, { 'Content-Type': 'text/html' })
     axios.get('https://randomuser.me/api').then((u) => {
       const { first, last } = u.data.results[0].name
       const timestamp = moment().format('MMMM Do YYYY, h:mm:ss a')
       users.push({ first, last, id: uuidv4().slice(30), timestamp })
       _.forEach(users, (u) =>
         console.log(`Nombre: ${u.first} - Apellido: ${u.last} - ID:
${u.id} - Timestamp: ${u.timestamp}`)

       )
       res.write('<ol>')
       _.forEach(users, (u) =>
         res.write(`<li>Nombre: ${u.first} - Apellido: ${u.last} - ID:
${u.id} - Timestamp: ${u.timestamp} </li>`)
       )
       res.write('</ol>')

       res.end()
     })
   }
 })
 .listen(3001, function () {})
```

Requerimiento 5

```
const http = require('http')
const axios = require('axios')
const chalk = require('chalk')
const { v4: uuidv4 } = require('uuid')
const _ = require('lodash')
let users = []
const moment = require('moment')
http
 .createServer((req, res) => {
   if (req.url.startsWith('/usuarios')) {
     res.writeHead(200, { 'Content-Type': 'text/html' })
     axios.get('https://randomuser.me/api').then((u) => {
       const { first, last } = u.data.results[0].name
       const timestamp = moment().format('MMMM Do YYYY, h:mm:ss a')
       users.push({ first, last, id: uuidv4().slice(30), timestamp })
       _.forEach(users, (u) =>
         console.log(
           chalk.blue.bgWhite(`Nombre: ${u.first} - Apellido: ${u.last}
- ID: ${u.id} - Timestamp: ${u.timestamp}`)
         )
       )

       res.write('<ol>')
       _.forEach(users, (u) =>
         res.write(`<li>Nombre: ${u.first} - Apellido: ${u.last} - ID:
${u.id} - Timestamp: ${u.timestamp} </li>`)
       )
       res.write('</ol>')

       res.end()
     })
   }
 })
 .listen(3001, function () {})
```

Requerimiento 6

```
nodemon index.js
```