

1. Parameter `{cors}` adalah link proxy untuk NodeJS dari heroku. Link ini akan memvalidasi request kita untuk menentukan apakah kita dapat mengakses webservice yang dideploy di heroku.
2. Potongan kode tersebut melakukan normalisasi pada data menjadi bentuk json yang sesuai.
3. Menambahkan staf farmasi :
 - a. Membuat method `getAllFarmasi` di `Appointment.js` yang berisi pemanggilan webservice `getAllStaffFarmasi`.
 - b. Membuat `DaftarFarmasiRow.jsx` di folder `components` yang berisi row dari data farmasi yang sudah diisi
 - c. Membuat `DaftarFarmasi.jsx` di `screens` untuk menampilkan table.
 - d. Menambahkan link `/all-farmasi` di `Route` dan menambahkan navbar `Daftar Staff Farmasi` di `App.js`

Menambahkan hasil lab :

- a. Membuat `FormHasilLab` di `containers` (serupa dengan form update)
- b. Membuat `CreateHasilLab.jsx` di `screens` berisi `render`, `constructor` dan `form handler`.
- c. Membuat method `addLabResult` di `Appointment.js` yang melakukan `POST` pada API `addLabResult`
- d. Menambahkan tombol `Add Hasil` di `DaftarPasien`
- e. Menambahkan link `/tambah-lab/:id` di `App.js`