

## HTML pseudocode

I created a skeleton of this page - **create\_order.html** - you can use it to have the idea.

### Form

#### Client Information\_\_\_\_\_

- Name - string
  - Picked on clients tab
- Last Name - string
  - Picked on clients tab
- Gender - string
  - Picked on clients tab
- Diagnostic - string
  - Entered by user
- Doctor - string
  - Picked on Doctor tab
- Age - integer
  - Picked and calculated by client tab

#### Service Information\_\_\_\_\_

- Store - String
  - Picked by doctor tab
- Service - String
  - Picked by Services tab
- Payment Method - string
  - <select> picked between
    - Cash
    - Credit
    - Debit
    - Check
- Price - float, 2 decimal
- Status - string
  - <select> picked between
    - Paid
    - Incomplete\*
    - Not paid
- \* If status incomplete add dynamically "Amount Paid" field - float, 2 decimal
- Discount
  - Entered by the user
  - Number
  - The amount need to be subtracted from the original price

SUBMIT BUTTON

## Tabs

Clients Tab

---

- Create client Button
  - Modal Form - the modal form already exist on **patients.html**
    - Name - string
    - Last Name - string
    - Date of Birth - string dd/mm/yyyy
    - Gender - string, Male female
    - Street - string
    - City - string
    - State - string
    - Country - string
    - Zipcode - string
    - Email - string
    - Phone number - string
    - Submit and Cancel Button

DATA TABLE, populate data table with REST API

Doctor Tab

---

- Create Doctor Button
  - Modal Form - the modal form already exist on **doctors.html**
    - Name - string
    - Last Name - string
    - Speciality- string
    - Licence - string
    - Street - string
    - City - string
    - State - string
    - Country - string
    - Zipcode - string
    - Email - string
    - Phone number - string
    - Submit and Cancel Button

DATA TABLE , populate with REST API

Stores Tab

---

DATA TABLE , populate with REST API

Services Tab

---

DATA TABLE , populate with REST API

## FUNCTIONALITY (JavaScript)

We will start by populate the Tabs:

- Clients
  - GET
  - [https://bwlf2cpbd4.execute-api.us-east-1.amazonaws.com/dev\\_0/](https://bwlf2cpbd4.execute-api.us-east-1.amazonaws.com/dev_0/)
  - JSON
  - Elements attributes:
    - "client\_id": Not visible on data table
    - "name": Visible on data table
    - "family\_name": Visible on data table
    - "date": Visible on data table
    - "gender": Not visible on data table
    - "street": Not visible on data table
    - "city": Not visible on data table
    - "state": Not visible on data table
    - "country": Not visible on data table
    - "zipcode": Not visible on data table
    - "phone": Visible on data table
    - "email": Visible on data table
- Doctors - REST API- GET
  - GET
  - [https://ug4lc0jiwg.execute-api.us-east-1.amazonaws.com/dev\\_0/](https://ug4lc0jiwg.execute-api.us-east-1.amazonaws.com/dev_0/)
  - JSON
  - Elements attributes:
    - "doctor\_id": Not visible on data table
    - "Name": Visible on data table
    - "family\_name": Visible on data table
    - "medic": Visible on data table
    - "Licence": Not visible on data table
    - "Street": Not visible on data table
    - "city": Not visible on data table
    - "state": Not visible on data table
    - "country": Not visible on data table
    - "zipcode": Not visible on data table
    - "phone": Visible on data table
    - "email": Visible on data table

- Stores
  - GET
  - [https://iermfbynl5.execute-api.us-east-1.amazonaws.com/dev\\_0/](https://iermfbynl5.execute-api.us-east-1.amazonaws.com/dev_0/)
  - JSON
  - Elements attributes:
    - "store\_id": Not visible on data table
    - "Name": Visible on data table
    - "Street": Visible on data table
    - "city": Visible on data table
    - "state": Not visible on data table
    - "country": Not visible on data table
    - "zipcode": Not visible on data table
    - "gps": Not visible on data table
    - "phone": Visible on data table
    - "email": Visible on data table
- Services
  - This data table will only show the specific store Services
  - If no store is selected do not show services, else show that specific store services. You can do a filter option in the data table.
  - GET
  - [https://rufyedk586.execute-api.us-east-1.amazonaws.com/dev\\_0/](https://rufyedk586.execute-api.us-east-1.amazonaws.com/dev_0/)
  - This REST API will give you all the existing services in all stores.
  - JSON
  - Elements attributes:
    - "service\_id": Not visible on data table
    - "name": Visible on data table
    - "products\_id": Not visible on data table
    - "products": Not visible on data table
    - "quantity": Not visible on data table
    - "unit": Not visible on data table
    - "price": Visible on data table
    - "store\_name": Not visible on data table

The user must pick from the tabs the information needed to complete the form. In case the Doctor or Client does not exist, they are allowed to create a new one.

New Client button

- Clients
  - POST
  - [https://bwlf2cpbd4.execute-api.us-east-1.amazonaws.com/dev\\_0/](https://bwlf2cpbd4.execute-api.us-east-1.amazonaws.com/dev_0/)

- JSON
- Modal example is on **patients.html**
- For client\_id use the code at the end of the document to create a uuid
- Elements attributes:
  - "client\_id": string
  - "name": string
  - "family\_name": string
  - "date": string
  - "gender": string
  - "street": string
  - "city": string
  - "state": string
  - "country": string
  - "zipcode": string
  - "phone": string
  - "email": string

New doctor button

- Doctors
  - POST
  - [https://ug4lc0jiwg.execute-api.us-east-1.amazonaws.com/dev\\_0](https://ug4lc0jiwg.execute-api.us-east-1.amazonaws.com/dev_0)
  - JSON
  - Modal example is on **doctors.html**
  - For doctor\_id use the code at the end of the document to create a uuid
  - Elements attributes:
    - "doctor\_id": string
    - "name": string
    - "family\_name": string
    - "medic": string
    - "Licence": string
    - "Street": string
    - "city": string
    - "state": string
    - "country": string
    - "zipcode": string
    - "phone": string
    - "email": string

Step one - select client

- From the data table the user can search in the data table and select an existing client
- Once selected we need to calculate the age by subtracting the date of birth and current date
- Finally add the values to the main form.

#### Step two - select doctor

- From the data table the user can search in the data table and select an existing doctor
- Finally add the values to the main form.

#### Step three - select store

- From the data table the user can search in the data table and select an existing Store
- Finally add the values to the main form.

#### Step four - select study

- From the data table the user can search in the data table and select an existing Service
- Finally add the values to the main form.

#### Step five - complete form

- User must complete diagnostic and status inputs

#### Step six - submit form

- Submit form in JSON format

#### Submit Button

- POST
- [https://arvixe3616.execute-api.us-east-1.amazonaws.com/dev\\_0/](https://arvixe3616.execute-api.us-east-1.amazonaws.com/dev_0/)
- JSON
- All fields must be completed in order to create an order.
- Create uuid with the code at the last of this document.
- Elements attributes:
  - "order\_id": string
  - "name": string
  - "family\_name": string
  - "doctor\_name": string
  - "doctor\_id": string
  - "service\_name": string
  - "service\_id": string
  - "age": string
  - "diagnostic": string
  - "due\_payment": string
  - "gender": string
  - "payment\_method": string
  - "doctor\_email": string
  - "client\_email": string
  - "status": string
  - "store": string
  - "price": string

- “Discount”: string
- “date”: string (current local date, format mm/dd/yyyy)
- “time”: string (current local time, format hh:mm:ss)

## REST APIS

- Clients:
  - [https://bwlf2cpbd4.execute-api.us-east-1.amazonaws.com/dev\\_0/](https://bwlf2cpbd4.execute-api.us-east-1.amazonaws.com/dev_0/)
- Doctors:
  - [https://ug4lc0jiwg.execute-api.us-east-1.amazonaws.com/dev\\_0/](https://ug4lc0jiwg.execute-api.us-east-1.amazonaws.com/dev_0/)
- Stores:
  - [https://iermfbynl5.execute-api.us-east-1.amazonaws.com/dev\\_0/](https://iermfbynl5.execute-api.us-east-1.amazonaws.com/dev_0/)
- Services:
  - [https://rufyedk586.execute-api.us-east-1.amazonaws.com/dev\\_0/](https://rufyedk586.execute-api.us-east-1.amazonaws.com/dev_0/)
- Orders :
  - [https://arvixe3616.execute-api.us-east-1.amazonaws.com/dev\\_0/](https://arvixe3616.execute-api.us-east-1.amazonaws.com/dev_0/)

## Additional Code

### UUID

```
var dt = new Date().getTime();
var uuid = 'xxxxxxxx-xxxx-4xxx-yxxx-xxxxxxxxxxxx'.replace(/[xy]/g, function (c) {
    var r = (dt + Math.random() * 16) % 16 | 0;
    dt = Math.floor(dt / 16);
    return (c == 'x' ? r : (r & 0x3 | 0x8)).toString(16);
});
```