

Article

[Henrique Dias](#) · Aug 26, 2020 3m read

[Open Exchange](#)

Updating Patient Resource using fhir.js

Hi Community,

I shared my experience working with FHIR for the first time in [this article](#).

In that article, I wrote how I explored the FHIR Resources and talked about the information that I found useful in FHIR documentation.

The first version of my app only shows the information that FHIR Resource provides.

I wanted to make the user able to update the patient details, so I to search for an example of how to do it.

The [Github page of fhir.js](#) has an example of how to update a resource.

Update Resource

To update a resource, call `fhir.update({type: resourceType, id: identifier, resource: resourceObject})`. In case of success, the callback function will be invoked.

Example:

```
this.fhirClient.update({
  type: "Patient",
  id: 1,
  resource: {
    name: 'New Name'
  }
}).catch(function(e){
  console.log('An error happened while updating patient: \n' + JSON.stringify(e));
  throw e;
}).then(function(bundle){
  console.log('Updating patient succeeded');
  return bundle;
});
```

Looking at this example made me think that it was effortless.

To update the resource, you only need to inform the properties that you want to change. Wow



I thought it would be something like:

```
this.fhirClient.update({
  type: "Patient",
  id: 1,
  resource: {
    name: 'Henrique',
    birthDate: '1984-02-10'
  }
});
```

```

    }
  }).catch(function(e){
    console.log('An error happened while updating patient: \n' + JSON.stringify(e));
    throw e;
  }).then(function(bundle){
    console.log('Updating patient succeeded');
    return bundle;
  });

```

But didn't work exactly that way.

To make it work, I needed to make a few adjustments in my code.

I am defining a variable with the patient object that returns when I fill the form.

```

// Perform a search to retrieve patient details for a specific patient
window.loadForm = function (patientId) {
  client.search({
    type: 'Patient',
    query: {
      _id: patientId
    }
  }).then((res) => {
    const bundle = res.data;
    bundle.entry.forEach((patient) => {
      objPatient = patient;
    });
  });
}

```

Here the variable objPatient will be the object that I'll send in the FHIR Update.

The button Update, when clicked, triggers a function updatePatient.

And the function updatePatient is responsible for updating the objPatient data and then send the object to the FHIR Server.

```

window.updatePatient = function (patientId) {
  objPatient.resource.id = $("#fhirId").val();
  objPatient.resource.identifier[2].value = $("#SSN").val();
  objPatient.resource.name[0].given[0] = $("#firstName").val();
  objPatient.resource.name[0].family = $("#lastName").val();
  objPatient.resource.birthDate = $("#dateofbirth").val();
  objPatient.resource.gender = $("#gender").val();
  objPatient.resource.address[0].line[0] = $("#address").val();
  objPatient.resource.address[0].city = $("#city").val();
  objPatient.resource.address[0].state = $("#state").val();
  objPatient.resource.address[0].country = $("#country").val();
  client.update({
    type: "Patient",
    id: parseInt(patientId),
    resource: objPatient.resource
  }).catch(function (e) {
    showToast(1);
    $("#updateData").prop('disabled', false);
    throw e;
  }).then(function (bundle) {
    showToast(0);
    $("#updateData").prop('disabled', false);
  });
}

```

```
        return bundle;
    });
};
```

Doing this way, the FHIR Server is updated with the information that you need.

I hope that these pieces of code could be useful to someone, someday. 😊

You can try this feature here!

<http://iris-fhir-portal.eastus.cloudapp.azure.com:32783/csp/user/fhirUI/patientlist.html>

If you liked the app or the article, and think I deserve your vote, please vote for iris-fhir-portal! 😊

<https://openexchange.intersystems.com/contest/current>

[#FHIR #Tips & Tricks #InterSystems IRIS for Health](#)

[Check the related application on InterSystems Open Exchange](#)

Source URL: <https://community.intersystems.com/post/updating-patient-resource-using-fhirjs>