

THEO II

Product Changes

05022024 – Changes Requested by WellTrackONE Staff

Background

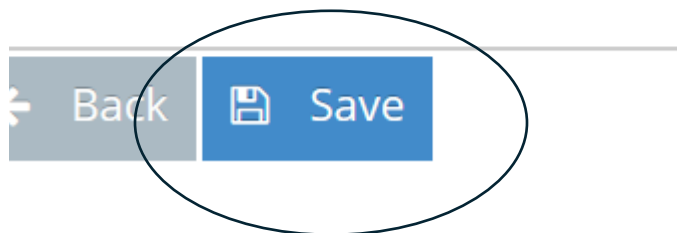
The old THEO I (iOS) version had some backend processing that would merge the PostGres data into the SQL data. These steps added certain descriptors to the data that were/are important to our staff for post-processing.

I have devised a solution that is relatively easy so we accomplish the same thing now that we no longer have that backend processing taking place.

Trigger

Your new code will be invoked when the user presses the SAVE button at the bottom of the template:

Medical Decision Making



Coding

Here are the steps the new code will go through. Note that this assumes you have already created the row for Appointments and EncounterHistory and have those Primary Keys available. You also know the Patient_ID of the PatientProfile table and you know the Org_ID of the OrgProfile table.

---Establish your database

USE Wellness_eCastEMR_Data

---Initialize variables

```
SELECT      @Appointments_ID = <get your primary key of the record you are writing to>
SELECT      @Encounter_ID =    <get your primary key of the record you are writing to>
SELECT      @Patient_ID =      <get your primary key of the record you are writing to>
SELECT      @Provider_ID =     <get your primary key of the record you are writing to>
SELECT      @Org_ID            <get your primary key of the record you are writing to>
SELECT      @TOA                = 'AWV Visit (THEO II)'
SELECT      @Notes              = 'Annual Wellness Visit'
SELECT      @PMSReason          = 'AWV Visit (THEO II)'
SELECT      @EncounterDescription_ID = 0
```

---Get the EncounterDescription_ID

```
SELECT      @EncounterDescription_ID = EncounterDescription_ID
FROM        EncounterDescriptionList
WHERE       Org_ID = @Org_ID AND Provider_ID = @Provider_ID
```

---Now update the Appointment and the Encounter tables with your data

UPDATE Appointments

```
SET TOA = @TOA, Notes = @Notes,
PMSReason = @PMSReason,
EncounterDescription_ID = @EncounterDescription_ID
WHERE
Encounter_ID = @Encounter_ID
```

UPDATE EncounterHistory

```
SET EncounterDescription = @PMSReason,
ChiefComplaint = @PMSReason,
EncounterNotes = @Notes,
PMSReason = @PMSReason,
EncounterDescription_ID = @EncounterDescription_ID
WHERE
Encounter_ID = @Encounter_ID
```

End of Document