README

TheraScript - Physical Therapy Prescription Generator Application for iPhone

Written by Brad Davis in fulfillment of the Udacity iOS nanodegree capstone

**Overview**

TheraScript is an app that automates the process of generating a physical therapy (PT) prescription. The app is designed for use by medical providers to create a PT prescription, and forward it to the therapist of choice electronically. The app captures all of the relevant data required by the therapist to properly care for the patient:

1. Patient demographics
2. Patient diagnosis list
3. Prescribed therapy regimen

When the provider generates the prescription, the result is currently forwarded to both the therapist and the patient via SMS/MMS id the device is SMS enabled.

This application benefits the user by eliminating the mountain of physical therapy prescription pads that tend to accumulate within a medical practice. It also streamlines the prescription process as well as creating a favorite therapy practice list for each user.

The app obtains diagnostic codes and descriptions via the Aqua.io healthcare API (<https://aqua.io/>).

User data is stored using NSKeyedArchiver. PT provider data is stores using Core Data.

**Application User Flow**

Initialization

Upon initial execution, the app directs the user to the *User Information* screen. This screen collects demographic data from the user, and allows the selection of an optional graphic icon representing the user’s practice. This data is used to customize the app itself, as well as communicate the provider’s data to the receiving physical therapist.

Provider Information

This screen allows the user to input personal and practice demographic data. This data populates the *New Prescription* view, and is added to the final therapy prescription. Once complete, the user selects the **Next button**to save the data.

Icon Image Selection

This screen allows the user to select a graphic icon that accompanies the user’s demographic data. The user may use the device’s camera to capture an image (if available), select an existing photo, or use a default image supplied by the app. The user completes the process by selecting the **Accept** button in the upper right corner. The user may use the **Back** button to navigate between screens, however the app requires that the user enter demographic data before operation.

New Prescription View

Following initialization, the user is then directed to the *New Prescription* screen. This screen is the default view for the remainder of the app’s life span. This view collects all the data entered by the provider, and brings it together as a complete prescription. This screen displays:

1. Patient information
2. Patient diagnosis List
3. Selected PT practice
4. Patient prescription

This view also allows the provider to alter his/her demographic data via the selection of the **Gear** navigation bar button in the upper right corner of the view. Selection of this button returns the user to the User Information screen. The user may clear the current prescription by selecting the **Clear** button in the lower left corner, and may generate the final prescription for transmission by selecting the **Generate** button in the right lower corner.

Patient Information

The patient information section of the view allows the user to input demographics for the patient to whom this prescription will belong. This information will be used to identify the patient to the therapist, as well as to locate nearby therapy practices. Finally, the patient’s phone number will be used to send the prescription via SMS/MMS.

Diagnosis List View

This view allows the user to specify up to three diagnoses from a specified list. The user can search either the ICD9 or ICD10 diagnostic code database. The user may search using a text string (e.g. ‘lumbar’) or by ICD code (e.g. 722 [ICD9], M65[ICD10]). The search returns a list of matching diagnoses listed by ICD code designator and plain text description.

The user may select up to three diagnoses. Selection is denoted by a checkmark to the right of the diagnosis. Attempting to select more than three diagnoses results in a dialog message alerting the user to the 3-term limitation.

The user completes the selection process by selecting the **Accept** button in the upper right corner of the view. The user may also return to the New Prescription view by selecting the back button.

Once completed, the user is returned to the *New Prescription* view and the diagnosis list is displayed in this section.

Therapy Practice View

The therapy practice screens are a set of views connected through a tab bar controller. The three connected views are:

1. Manual Practice Entry

The manual practice entry screen allows the user to enter physical therapy practice information by hand. The user completes this process by selecting the **Accept** button in the upper right corner.

1. Map Location

The map location screen allows the user to locate physical therapy practices local to the street address and zip code of the patient, using the MKLocalSearch functionality built in to the Apple MapKit. The search creates a pin on the map for each location that self describes as a physical therapy practice. The user may then select the pin representing the practice he/she wishes to utilize.

1. Saved Favorites

The saved favorites list view is a list of therapy practices saved to disk by the user, bypassing the manual entry and map location screens. The user completes this process by selecting the **Accept** button in the upper right corner. The user may also remove entries from the favorites list by selecting the **Remove** button in the upper right corner. This will allow selection of the desired entry, and confirm removal via the **DELETE** button in the selected row.

The *Map Location* screen performs the search for local PT practices using the address and Zip code data for the patient. If this information is missing, an error dialog will display and the user will be redirected back to the *New Prescription* view to enter patient data.

Once the user has selected a therapy practice, the app will check the saved favorites list. If the selected practice is not currently on the list, a dialog allows the user to save the practice is desired. The chosen practice data is loaded, and the user is returned to the *New Prescription* view and the practice data is populated in this section.

The user may return to the *New Prescription* view without selecting a diagnosis by using the **Back button** in the upper left corner.

Therapy Prescription View

This view allows the user to create a set of instructions for the physical therapist. The current version of the app is directed at orthopedic spine providers, and so focuses on spine related therapy. A future version may allow customization of this view by specialty, and may also allow users to create their own therapy templates.

Aside from the visit selector, which is variable using the segmented control, all treatments are a binary toggle on selection. A checkmark to the right of the treatment denotes selection.

The *Evaluate and Treat* section allows the user to select the number of physical therapy visits authorized, and designates whether a report from the therapist is required. The visit value defaults to 4, but is variable, with a range between 1 and 20 visits selectable by the user.

The remaining selection sections allow the user to select specific treatments that should be rendered to the patient. The user may select as many treatments as necessary.

The user completes the selection process by selecting the **Accept** button in the upper right corner of the view. The user may also return to the New Prescription view by selecting the back button.

Once completed, the user is returned to the *New Prescription* view and completion is indicated in this section.

**Therapy Prescription Generation**

Prescription Generation View

After the user has completed input into the *New Prescription* view to his/her satisfaction, selection of the **Generate** button displays the *Generate Prescription* view. The view is pre-populated with the data the user input via the *New Prescription* view in an efficient format suitable for transmission.

The app currently supports transmission to the patient and the therapist via SMS/MMS (if enabled on the device). Selecting the **Action** button in the upper right corner initiates the transmission. A dialog notifies the user of the transmission result. Currently, SMS/MMS transmission is the sole supported method. A future version will support email, printing, and transmission of a PDF file. A future version will also allow selection of recipients.

The user may return to the *New Prescription* view by selecting the **Back** button in the upper left corner.

**Author’s Notes**

I have several observations I want to make known to the reviewer, and to the Udacity staff in general. If I am chosen for the exit interview process I will expand on some of these observations, but placing them here seems the right thing, so here goes.

1. I undertook the Udacity iOS nanodegree program expressly to learn how to write this app. I am a full time orthopedic P.A. who specializes in spine care. This app was conceived out of my frustration with the mountains of paper prescriptions I must complete daily, as well as the mess the prescription pads make of my desk. I’ve known from day one that this would be my capstone project. I expected it to be a big project, but it turned out to be even bigger than I imagined.
2. I am not new to programming. I have been involved in computer software and hardware design since the late 1980s (I am currently 58). I have successfully launched software projects for the Atari ST, x86 PC, Amiga, and Mac platforms through the years. Despite this experience, writing this app has been a major accomplishment. I’ve been trying to get this app (or something like it) made for at least 3 years. I have made 4 separate attempts to write the app myself, and contracted it out once. I see my success at creating this app in a little less than six months as an amazing testament to the power of Udacity’s power to teach complex programming concepts. Despite the app’s complexity, I am amazed that I was able to use the Apple Developer docs and a little ingenuity to create this app on my own. This course helped me to think about programming problems in a new way, and synthesize know information to solve new problems efficiently. For the first time in my life, I actually *feel* like a competent programmer. I am very grateful to all of you folks at Udacity for making that possible!
3. I consider this app to be a pre-alpha version. I am planning on distributing it to a group of early adopters following completion of the capstone. Although the app contains basic functionality, I am aware that it needs to be extended to be a useful product. My rationale in submitting it now is that I believe it meets the specification and rubric, and I estimate that it will take another 3-6 months of work to add the features I would like to see in a near final version.