Concussion and Traumatic Brain Injury in Pediatric Patients

User Manual

The FHIRStarters
Jan Marie Andersen
Darryl Buswell
Spencer Knight
Lamar Phillips
Brian Wells

Table of Contents

Concussion and Traumatic Brain Injury in Pediatric Patients: A CDS App

This web-based app provides clinical decision support for the area of Mild Traumatic Brain Injury (mTBI) in children and adolescents. It provides an extensible CDS framework which can be expanded with new rules as well as a customizable Angular-based user interface.

The app was started in the Fall of 2017 as a semester project for Georgia Tech's Introduction to Health Informatics course and was greatly informed by the input of our mentors (listed below).

Development Team

Semester team name: FHIRStarters

- Jan Marie Andersen (QA, Developer)
- Darryl Buswell (Project Manager)
- Spencer Knight (Developer)
- Lamar Phillips (QA)
- Brian Wells (Developer)

Project Mentors

- Juliet Haarbauer-Krupa (CDC)
- Kelly Sarmiento (CDC)
- Melvin Crum (CDC)
- Jon Bidwell (Georgia Tech)

Source Code Repository

Source code for the project may be found on GitHub at the following link:

https://github.com/CDCgov/GaTech-Fall2017-Krupa-Concussion-FHIRStarters

Thank You

Thank you to Juliet, Kelly, and all others who provided guidance during development. It would not have been possible without them.

Installation

The following instructions will install the app and all dependencies by utilizing Docker. The app currently uses sample data on a standalone FHIR server, but with future work the app will be launchable from within an EHR system using SMART on FHIR. SMART on FHIR is already included in the app code.

1. If you don't already have docker (https://www.docker.com) installed, do so.
2. Clone this repo.
3. In a command window, navigate to the directory GaTech-Fall2017-Krupa-Concussion-FHIRStarters
4. Run `docker-compose up`
5. Wait for everything to build and startup.
6. In a browser, navigate to http://localhost:8083
7. Observe that sample data is automatically pulled from the FHIR server using smart-on-fhir.

Development Environment Setup

1. Download Visual Studio 2017 Installer
2. In the installer, under **Workloads>Web & Cloud**, select **ASP.NET and web development** and **Data storage and processing**. Under **Individual Components**, make sure **.NET Framework 4.6.2** (SDK and targeting) is selected. Install the selected components.
3. Download and install MySQL (scroll down for the installer, no need to login, just say "No thanks"). During installation, choose a custom install and select the **MySQL Server** as well as the **Connector/NET** (and a client if you want one).
4. Download and install MySQL for Visual Studio (choose **Typical** install, again no need to login).
5. Open the MySQL command line client or if you installed a client, open it. Run the follwing commands:
   - `CREATE DATABASE mtbicds .`
   - `CREATE USER 'fhirstarter'@'localhost' IDENTIFIED BY '[insert some password]';`
   - `GRANT ALL PRIVILEGES ON mtbicds.* TO 'fhirstarter'@'localhost' WITH GRANT OPTION;`
   - `CREATE USER 'fhirstarter'@'%' IDENTIFIED BY '[insert some password]';`
   - `GRANT ALL PRIVILEGES ON mtbicds.* TO 'fhirstarter'@'%' WITH GRANT OPTION;`
6. Update the **DefaultConnection** string in the **appSettings.Development.json** file in the fhirStartersApp directory to reflect the username and password you use.
7. Start Visual Studio 2017 and open the **fhirStartersApp.sln** file and wait for dependencies to install.
8. Follow these instructions for connecting to a MySQL database.
9. Attempt to run the server. If you experience an error such as *The command "node node_modules/webpack/bin/webpack.js" exited with code 1*, you may need to install a newer version of NodeJS (e.g. 6.11).

Project Overview

The Problem

Concussion is a well-known type of traumatic brain injury and is also known as mild traumatic brain injury (mTBI). Research suggests that patients with mTBI can present with different symptoms, which makes care and management of children with mTBI challenging. In addition, healthcare providers don't necessarily have the time or training to systematically assess and manage patients with suspected mTBI. The CDC reports that rates of emergency department visits of individuals aged 19 or younger presented with TBI doubled between 2001 and 2012. Therefore, the problem is: how to improve *evidence-based diagnosis* and management of mTBI in these patients?

In general, as described by our mentors, there are two components to improving the diagnosis and management of mTBI:

- Improve the ability of healthcare providers to diagnose mTBI at the time of injury by using evidence-based guidelines, and
- Improve communication between clinicians, families, and schools in order to improve post-injury management

The Solution

Our mentors identified the need for intelligent clinical decision support software to help facilitate these components. This project therefore aims to address the first component by developing a provider interface which provides clinical decision support based on input from the web interface and automatically-obtained information from integration with the provider's EHR via FHIR.

Notable Features

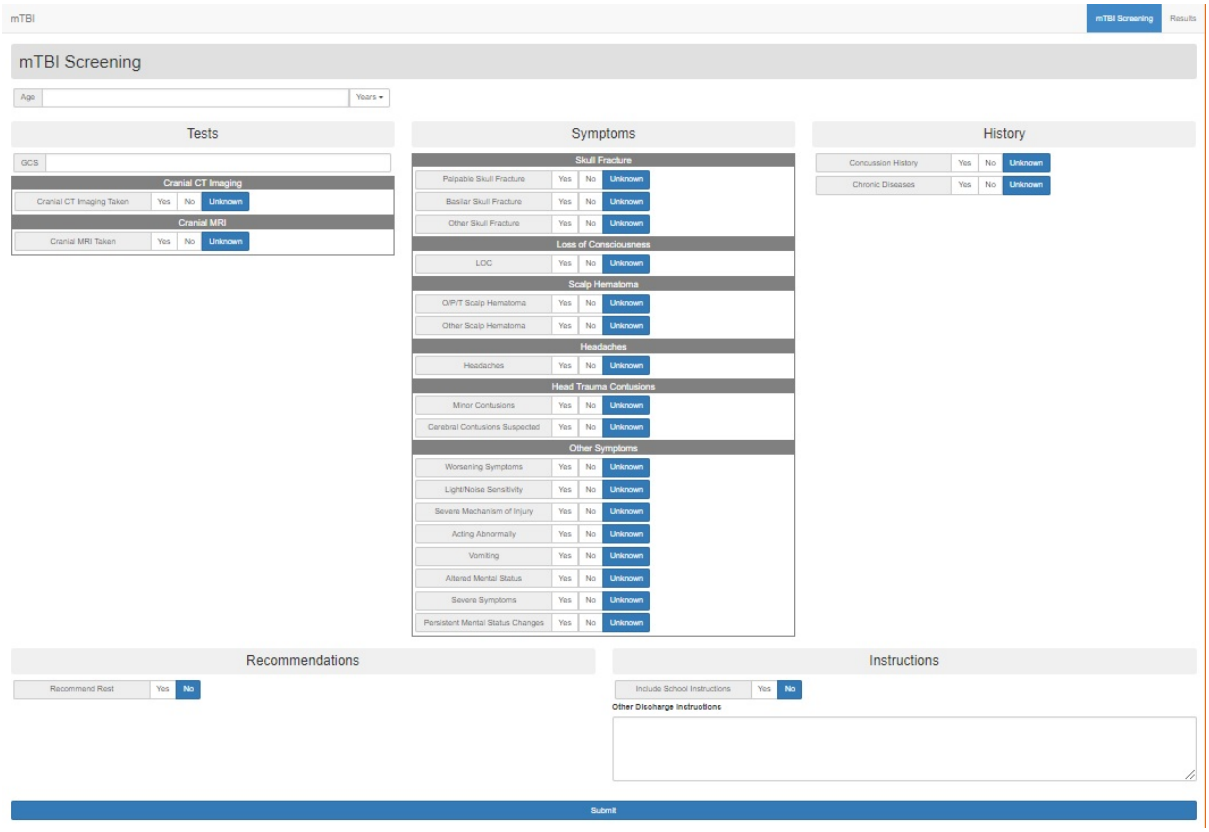
- Extensible CDS rule structure
  - Several CDS rules have been implemented
  - Rule framework makes implementing new rules plug-and-play
- Responsive front-end design
  - Data retrieved from the EHR using SMART on FHIR are highlighted for the physician
- Compiles to Docker image for easy deployment
  - docker-compose file creates containers for the web application, a MySQL database for persisting application state, and FHIR server to simulate an EHR system

User Guide

Launch

If you haven't already done so, please install the app using the instructions in the Installation section. Once this is done, navigate to http://localhost:8083 in a web browser to launch the app.

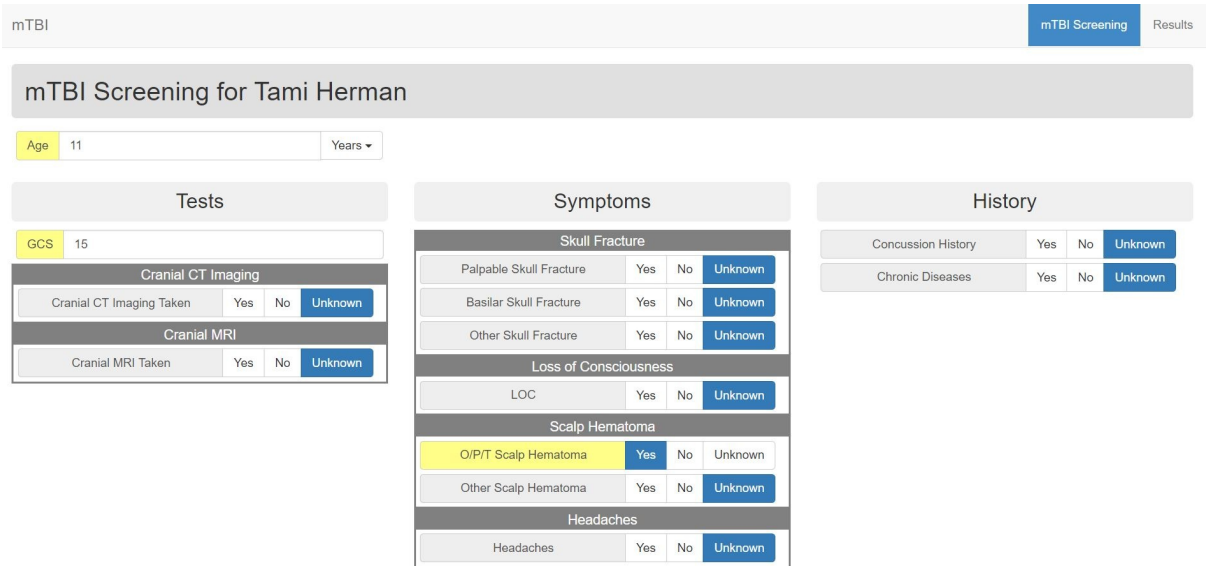
Data Collection



On launch, the app displays the data collection view to receive input for the CDS system. Data fields consist of number inputs (for example, the Age and GCS fields), tri-state inputs (allowing for Yes, No, and Unknown responses), and a free-text field for custom discharge instructions.

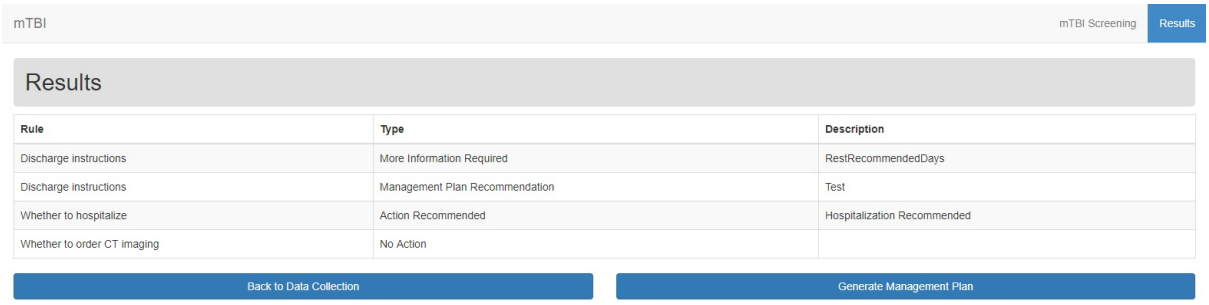
Some additional inputs only display once other relevant inputs have been filled. For example, the LOC Time field only displays once the user has indicated that LOC has occurred by choosing Yes on the LOC field.

Future work will enable some of this data to be retrieved via FHIR from the EHR which launches the app. These values will be highlighted in yellow to indicate they were obtained automatically and should be checked, as shown in the following screenshot. All values in the data collection view can be changed prior to submission regardless of their source.



After data has been entered, press the Submit button. When results are returned from the server's CDS logic, you are automatically redirected to the results view.

Results



This view displays the output of the CDS logic rules. Each row in the table represents one rule. The title of the rule is displayed in the first column. The second column shows one of the following for each rule:

- **No Action**: The rule has not recommended any actions.
- **Error**: An error occurred in the CDS rule.
- **More Information Required**: The rule requests more information which can be entered on the Data Collection view. Requested data fields are marked with an exclamation point.
- **Action Recommended**: The rule recommends an action be taken, such as a diagnostic image or hospitalization.
- **Management Plan Recommendation**: The rule has added recommendations to the management plan.

The content of the third column is dependent on the type of rule output. For an action recommendation, the column shows the recommended action. When more information is required, the column shows which fields are needed for the rule to complete. If an error occurs, the column shows the error message.

Next Steps

Depending on the rule output, you can use the Back to Data Collection button to enter more data or click the Generate Management Plan button to download the auto-generated management plan as a Microsoft Word document. This document contains special instructions based on entered data as well as the custom discharge instructions entered in the Data Collection view.