Logging in / Viewing/updating user-specific information

Upon opening the app, a UserController object is created. It loads a UserManager object that will check if there are any Users stored on the device, and it will create a UserManagerViewer and UserManagerUpdater (one for viewing User-related information and one for updating). The user enters a username, which the UserController calls onto UserManagerViewer to verify. If an account with the specified username is present, the user is logged in. Otherwise, the user can register a new account. If so, they will need to store some personal details (the UserController will instruct the UserManagerUpdater to create and store a new User object). When 'logged in', the current username is stored in UserController. In the main menu, the user can click for topics (2nd paragraph), user profile and user history. If profile is chosen, it displays the user's profile as well as the user's reported scores on specific topics; (UserController asks UserManagerViewer for a map of details and a map of scores for User for the UI to display). If user history is chosen, it shows all past records of the user generating a worksheet with the most recent appearing first (For each worksheet generated, the details of the generated worksheet (i.e. time, number of questions, difficulty) will be stored in the User's history). The user has the options to add a score for each record or to delete a record. If recording a score, the UserController will instruct the UserManagerUpdater to update the User's score record.

Generating worksheets

The user is currently on the topic select screen. Through a series of screens the user selects to generate 10 "addition", "standard", "medium" types of questions with font size of 16, the question format as "Horizontal" (ex. 10 + 68 = ?), and the title is "Test addition worksheet". Upon filling in these parameters, the "Generate worksheet" button becomes pressable. This button results in 3 major actions. First, it calls on WorksheetGenerator to generate multiple equations by repeatedly calling EquationGenerator. In response, EquationGenerator generates BedmasEquations (subclass of Equation) that adheres to the specified parameters (addition, standard, medium) and returns a string representation of these BedmasEquations (as {question, answer}) to WorksheetGenerator. WorksheetGenerator then packages these string representations of BedmasEquations into a list and stores them in a Worksheet instance. Secondly, it calls on PDFPresenter to create 2 PDFs (with and without answers) by applying the font size and equation format to the worksheet equations and stores them back in the Worksheet. Finally, it redirects you to a new page with a view of the PDF and stores . The user can then press "Download PDF". If the user presses "Download PDF", it calls on PDFPresenter to download the PDF onto the phone.