

# Enterprise Business Rules

Equation (Interface)	
	BedmasEquation LcmGcfEquation
<ul style="list-style-type: none"><li>• A solvable equation with a question and an answer.</li><li>• solve(), uses question to find answer</li><li>• getEquation()/toList() -&gt; returns Equation as a list (question, answer)</li></ul>	

BedmasEquation	
	Equation
<ul style="list-style-type: none"><li>• Question with bedmas operators and numbers (either fractions or integers) and answers.</li><li>• Ex. <math>(5+4) + 7 = 16</math></li><li>• BedmasEquation(expTree, answer)</li><li>• getEquation() → returns [BinaryExpTree, answer]</li></ul>	BinaryExpTree

LcmGcfEquation	
	Equation
<ul style="list-style-type: none"><li>• LCM (lowest common multiple) or GCF (greatest common factor) questions with answers.</li><li>• Ex. <math>LCM(10, 5, 15) = 5</math></li><li>• LcmGcfEquation(List[int], answer)</li></ul>	

BinaryExpTree <Generic>	
<ul style="list-style-type: none"><li>• Stores the question part of BedmasEquations</li><li>• Generic → can be a BinaryExpTree of Integer or Fraction</li></ul>	Fraction

Fraction	
<ul style="list-style-type: none"><li>• Stores a numerator, denominator, and whole number. Can be used in a BedmasEquation</li></ul>	

User	
<ul style="list-style-type: none"><li>• Store private attributes (username, name, age, role). Upon initialization, stores empty records for worksheet to recorded score, and an empty history.</li><li>• getUsername(); returns username for user</li><li>• getDetails(); returns mapping to user details</li><li>• getScores(); return mapping of worksheet (key includes difficulty) to recorded score, if available.</li><li>• getHistory(); return list of details of worksheets user generated</li><li>• removeFromHistory(int index); removes worksheet details at index i from history</li><li>• addToHistory(&lt;T&gt; item); appends latest worksheet details to list of user history</li><li>• setWorksheetScore(String worksheetKey, &lt;T&gt; score)<ul style="list-style-type: none"><li>◦ Records score for worksheet (key includes difficulty)</li></ul></li></ul>	

# Application Business Rules

## EquationGenerator

- Creates multiple Equations (questions and answers) and returns them as an array.
- createEquation (typeOfEquation, difficulty)
  - Returns equations for entire worksheet as an array representation [[question, answer], [question, answer], ...]

Equation

## UserManagerViewer

- Access information from User for UserController.
- verifyUsername(String username)
  - Verifies that a username is in list of Users
- getUserScores(String username)
  - Calls User.getScores()
- getUserDetails(String username)
  - Calls User.getDetails() on specified user to return user details
- getUserHistory(String username)
  - Calls User.getHistory()

User

## UserManager

- Has a list of all Users.
- Generates UserManagerViewer UserManagerUpdater.
- findUsers()
  - Access all the users from a text file, stores them as a list
- createViewerAndAdapters()
  - Creates the UserManagerViewer and UserManagerUpdater

UserManagerViewer  
UserManagerUpdater  
User

## UserManagerUpdater

- Update information in User for UserController.
- createUser(String username, \*\*args)
  - Creates and stores User object and their user details
- storeUserAction(String username, \*\*args)
  - Calls User.addToHistory(\*\*args); to store details of latest action of User
- removeUserAction(String username, int index, \*\*args)
  - Calls User.removeFromHistory(); to remove action
  - Calls User.setWorksheetScore(); to update corresponding worksheet value to last value or null
- storeUserScore(String username, String worksheetKey, int score)
  - Calls User.setWorksheetScore(string worksheetKey, int score); to store worksheetKey - score pair

User

# Interface Adapters

## Worksheet

- Holds the worksheet and PDF representation.
- `getWorksheetEquations()`
- `setWorksheetEquations(array)`
- `setWorksheetPDF(PDF)`
- `getWorksheetPDF()`

## WorksheetGenerator

- Generates the equations for a worksheet and stores it in the Worksheet class.
- `generateWorksheet(equationType, numEquations, difficulty)` → calls `EquationGenerator.createEquations` assigns it to `Worksheet.setWorksheetEquations()`

Worksheet  
EquationGenerator

## PDFPresenter

- Creates visual PDF output using Worksheet and formatting parameters.
- `createWorksheetPDF(fontSize, equationFormat, title)` → creates a PDF of the worksheet using `Worksheet.getWorksheet`, `fontSize`, `equationFormat` and `title`. Calls `setWorksheetPDF`, storing the PDF in worksheet
- `getPDF()` → accesses the PDF stored in Worksheet using `getWorksheetPDF`
- `downloadPDF(path)` → accesses the PDF, downloads it into the path of choice.

Worksheet

## UserController

- Stores UserManager, current username and (temporarily) user details.
- `verifyUsername(String username)`
  - `UserManagerViewer.verifyUsername()` to verify username for login. If successful, save username to current username and call `UserManager` to get user information
  - Return boolean
- `getUserDetails()`
  - Calls `UserManagerViewer.getUserDetails()`; returns hashmap of profile info
- `getUserScores()`
  - Calls `UserManagerViewer.getUserScores()` returns hashmap of scores
- `getUserHistory()`
  - Calls `UserManagerViewer.getUserHistory()` to get all past actions of user
- `createUser(String username, **args)`
  - Calls `UserManagerUpdater.createUser(username, **args)` to create new user
- `storeUserScore(String worksheetKey, int score)`
  - Calls `UserManagerUpdater.storeUserScore()`; to store score for a worksheet
- `storeUserAction(**args)`
  - Calls `UserManagerUpdater.storeUserAction()`; to store type of worksheet generated, time, num questions, etc. in User history
- `removeUserAction(int index)`
  - Calls `UserManager.removeUserAction(index)`; to removes user action at index from history

UserManagerViewer  
UserManagerUpdater  
UserManager

# Frameworks & Drivers

UserInterface	
<ul style="list-style-type: none"><li>• Provides interactive visual display to user (move between different panels).</li><li>• Stores the equationType (ex. standard addition), numEquations, fontSize, equationFormat (ex. horizontal) as they are pressed.</li></ul> <p><u>Worksheet Generating Methods</u></p> <ul style="list-style-type: none"><li>• Based on buttons pressed, can call on ...<ul style="list-style-type: none"><li>◦ WorksheetGenerator to generate the equations for a Worksheet (ex. equationType, numEquations, difficulty)</li><li>◦ PDFPresenter to display a PDF of a worksheet based on formatting details (ex. fontsize, questionFormat, title)</li><li>◦ PDFPresenter to download the PDF to a specified path</li></ul></li></ul> <p><u>User Methods</u></p> <ul style="list-style-type: none"><li>• Based on button pressed, if user-related command, sends various method calls to UserController to either store or retrieve information about Users.</li></ul>	WorksheetGenerator PDFPresenter UserPresenter