**Wire Frames Requirements for Key Outcomes Tracker Tool**

**Frame #0**

1. This represents the icon to open the application.

**Frame #1A**

1. If touch “Enter and View Classes and Study Sessions”, then frame #3 will open.
2. If touch “Enter and View Instructors and TA’s”, then frame #5 will open.
3. If touch “Enter and View Class Activities”, then frame #7 will open.
4. If touch “Track and View Key Outcomes”, then frame #9 will open.
5. If touch the “information icon”, then frame #2 will open.
6. If touch the “SP” (Student Profile Icon), then frame #1B will open. This data in frame #1B is not required.
7. A Requirement: By touching “Succeeding In College: Do’s and Don’ts from Fellow Students”, this PDF will open. (The PDF is embedded in the application.)

**Frame #1B**

1. This frame represents the student (user) who has downloaded this application to his/her device. “First Name”, “Last Name”, “Student ID”, “School Name”, “Other”, and “Description” should be in light font (hint). When the user touches a field, the wording will no longer appear and the user can then enter the appropriate data through a key board screen.

**Frame #2**

1. When the user touches “contact us”, it should display the user’s email icons. Based on the email option selected by the user, the following email address should be populated in the “To” line. [info@BestPracticesToGraduate.org](mailto:info@BestPracticesToGraduate.org)
2. Located at the bottom of this frame is a hyperlink to <http://bestpracticestograduate.org/>

**Frame #3**

1. This frame will open with a view of all classes that were previously entered.
2. All fields are read only.
3. By touching the “add button” at the top of the frame, it will take the user to frame #4 where the user can add a class and corresponding number of targeted study sessions.
4. By touching the “edit” button to the right of the “class name”, the user will be taken to a pre-populated frame #4 in edit mode for that class. Any changes made in frame #4 can be saved by touching the “save and add” button or the “save and finish” button. The “save and add” button will first save the data and then clear the screen so the user can add another class. The “save and finished” button will first save the data and then take the user back to frame #3.
5. If the user deletes a class through the “trash can”, then the application should ask whether the user is sure he/she wants to make this deletion. It should also state that this action cannot be undone.

**Frame #4**

1. “Class name” should be in a light (hint) font. When the user touches “class name”, then “class name” will no longer appear and the user can then enter the data through a key board screen. “Course number” should be in a light (hint) font. When the user touches “course number”, then “course number” will no longer appear and the user can then enter the data through a key board screen.

1. Date of First Class” and “Date of Last Class” should be in a light (hint) font. When the user touches “Date of First Class” or “Date of Last Class”, a calendar should appear. The user should be able to scroll the calendar to the date they are interested in.
2. After touching the days that the class is scheduled, then those days should be highlighted. I.e. highlighted in a grey tint.
3. After touching the days that the class is scheduled, then those days will appear along with corresponding options to enter start and end times. The user should be able to touch the time(s) and then scroll to the correct time. Note – After selecting the times for the first scheduled day, then those same times should populate all other days scheduled. The user should be able to touch any time and change that time.
4. With “Enter # of targeted weekly study session”, the “help button” provides the following guidance: 1) One study session equates to 50 minutes of studying. 2) Suggested Guideline: Under a traditional 15 week semester, you should plan on six 50 minute study sessions per class, per week. (This guideline is seen in “frame A”.)
5. “Class Location” and “Description” should be in light (hint) font. When the user touches either of these fields, the field names will no longer appear and the user can then enter the appropriate information.
6. All fields above the “class location” field are required fields. If any of these fields above “class location” are left blank, the application should not allow the user to save a class. If the user attempts a save with any of these fields not entered, then the application should clearly indicate which required fields need to be entered.
7. Under “Instructor’s Name”, the user should be able to select a drop down list that displays Instructor Names that have already been entered. The user can then select an Instructor’s name from this drop down list. If the user does not see the Instructor’s name, then the user will have the option to touch the “add” button to the right of “Instructor’s name” and add a name through Frame #6. When the user then touches “save and finish” in frame #6, it should first save the data and then take the user back to frame #4. After adding an Instructor, the user should now be able to select that Instructor’s name from the drop down list in frame #4.
8. After the user selects an Instructor’s name, the user could then touch the “edit” button next to this name and be taken to a pre-populated frame #6 in edit mode. After edits have been completed, the user can then touch “save and finish” in frame #6 and return to frame #4.
9. Under “TA’s Name” the user should be able to select a drop down list that shows TA’s that have already been entered. The user can then select a TA’s name from this drop down list. If the user does not see the TA’s name, then the user will have the option to touch the “add” button to the right of “TA’s Name” and add a name through Frame #6. If the user then touches “save and finish” in frame #6, it should first save the data and then take the user back to frame #4. After adding a TA, the user should now be able to select that TA’s name from the drop down list in frame #4.
10. After the user selects a TA’s name, the user could then touch the “edit” button next to this name and be taken to a pre-populated frame #6 in edit mode. After edits have been completed, the user can then touch “save and finish” in frame #6 and return to frame #4.
11. If in “edit mode” at the top of the frame, the user should be able to edit all fields. This includes changing a “class name” and “course number” or correcting a misspelled “class name” or correcting a “course number”.
12. At the bottom of the frame, touching “save and add” will first save the data and then clear the screen so the user can add another class. The “save and finished” button will first save the data and then take the user back to frame #3.
13. If the user deletes a class through the “trash can” icon, then the application should ask whether the user is sure he/she wants to delete this class. It should also state that this action cannot be undone.

**Frame “A”**

”One study session equates to 50 minutes of studying.”

“Suggested Guideline: Under a traditional 15 week semester, you should plan on six 50 minute study sessions per class, per week.”

**Frame “B”**

The purpose of this frame is to list “Types of Activities” that are applicable to “type” referenced in frame # 8.

**Frame #5**

1. At the top of the frame, if the user touches “Instructors”, then Instructors will be listed in alphabetical order by last name; whereas, if the user touches “TA’s”, then TA’s will be listed in alphabetical order by last name. (Note – When this frame first opens, by default the “Instructors” will be displayed.)
2. All fields are read only.
3. By touching the “add button” at the top of the frame, it will take the user to frame #6 where the user can add an Instructor or TA.
4. By touching the “edit” button to the right of the name, the user will be taken to a pre-populated frame #6 in edit mode. Any changes made in frame #6 should be saved and integrated throughout the application.
5. If the user deletes an Instructor or TA by touching the “trash can” icon, then the application should ask whether the user is sure he/she wants to make this deletion. It should also state that this action cannot be undone.

**Frame #6**

1. To indicate whether this is an Instructor or TA, the user will touch the appropriate radio button.
2. “Name”, “Office Location”, “eMail Address”, and “Phone Number” should be in light (hint) font. When the user touches these fields, the field names will no longer appear and the user can then enter the appropriate information. The application should guide the user in entering proper email and phone number formats.
3. After touching the days that the Instructor or TA has office hours, then those days should be highlighted. I.e. highlighted in a grey tint. These days will then appear along with corresponding options to enter start and end times for office hours. The user should be able to touch the time(s) and then scroll to the correct time. Note – After selecting the times for the first scheduled day, then those same times should populate all other days scheduled. The user should be able to touch any time and change that time. In addition, there should be a drop down arrow so the user can enter more than one time segment for a given day.
4. The only fields that are mandatory on this frame are the radio button and the “Name”. If the user attempts a save with any of these fields not entered, then the application should clearly indicate which required fields need to be entered.
5. At the bottom of the frame, touching “save and add” will first save the data and then clear the screen so the user can add another Instructor or TA; whereas, after touching the “save and finished” button, the data will be first saved and the user will be taken back to Frame #5.
6. When in “edit mode” at the top of the frame, a user should be able to edit all fields. This includes correcting misspelled Instructor and TA names.
7. If the user deletes an instructor or TA by touching the “trash can” icon, then the application should ask whether the user is sure he/she wants to make this deletion. It should also state that this action cannot be undone.

**Frame #7**

1. Depending on what the user touches, there are two views for class activities: “this week” and “by class”. (Note – When this frame first opens, by default “this week” will be displayed.)
2. This view will show class activities that were previously entered in frame #8. The order of each view is based on the chronological order of due dates with the earliest due date showing first.
3. By touching the “add button” at the top of the frame, it will take the user to frame #8 where the user can add a class activity.
4. By touching the “edit” button to the right of “class name”, the user will be taken to a pre-populated frame #8 in edit mode.
5. The user can mark (touch) whether he/she completed the class activity. This ‘mark’ should first be auto saved to the database and reflected in frames #8 and #12.
6. After the user enters a grade (a free text field to accommodate a letter, number or a percent grade), the user must touch ‘submit’ for the entered grade to be saved to the database. This grade should also be reflected in frames #8 and #12.
7. If the user deletes a class activity through the “trash can”, then the application should ask whether the user is sure he/she wants to make this deletion. It should also state that this action cannot be undone.

**Frame #8**

1. “Class name” should be in a light font (hint). A drop down list should allow the user to select from classes that have already been entered. After selecting a “class name”, the corresponding “course number” should be auto populated.
2. After selecting a “class name” from the drop down list, if the user sees that the “class name” is misspelled, he/she can touch the “edit button” to the right of the “class name” and be taken to frame #4 in edit mode. After completing his/her edits, the user can touch “save and finish” in frame #4 and be taken back to frame #8.
3. If the “class name” had not been entered, then the user can select the “add button” to the right of “class name” and he/she will be taken to frame #4 where he/she can enter a new “class name”. Then, upon touching “save and finish” in frame #4, the user should be taken back to frame #8.
4. “Type”, should be in a light font (hint). A drop down list should allow the user to select from a list of types as seen in frame “B”.
5. The word “name” should be in a light font (hint). When touched, the word “name” should no longer be seen and the user can then enter a “name”.
6. When entering the “due date”, a calendar should appear. This should allow the user to scroll and select a date.
7. “Class Name”, “Course Number”, “Type”, “Name”, and “Due Date” are required fields. If the user attempts a save (seen at the bottom of this frame) with any of these fields not entered, then the application should clearly indicate which fields are required to be entered.
8. The word “description” should be in a light font (hint). When touched, the word “description” should no longer be seen and the user can then enter a description. The user should be provided three lines to type any data.
9. “Maximum Grade” should be in a light font (hint). When touched, the word “Maximum Grade” should no longer be seen and the user can then enter the “Maximum Grade”. (This is a free text field that will accommodate a letter, number or a percent.)
10. The user can ‘mark’ whether he/she completed the class activity.
11. The user can enter a grade (a free text field to accommodate a letter, number or a percent grade).
12. At the bottom of the frame, touching “save and add” will first save the data and then clear the screen so the user can add another activity; whereas, after touching the “save and finished” button, the data will be first saved and the user will be taken back to Frame #7. (This saved data should also be reflected in frames #7 and #12.)
13. If “edit mode” at the top of the frame, the user should be able to edit all fields. This includes changing a “class name” through a drop down list, correcting a misspelled “class name” through the edit button next to “class name”, or adding a new “class name” through the add button next to “class name”. (In adding a new “class name” or editing a misspelled “class name”, the user will be taken to frame #4; when the user touches “save and finish” in frame #4, the user should be taken back to frame #8.)
14. If the user deletes a class activity through the “trash can”, then the application should ask whether the user is sure he/she wants to make this deletion. It should also state that this action cannot be undone.

**Frame #9**

1. The calendar should scroll so the user can touch the date he/she is interested in tracking a key outcome. The day selected should be outlined in bold.
2. After touching the date of interest, the user can then touch one of three key outcomes in the body of the frame. Depending on the outcome touched, either frames 10, 11, or 12 will open.

**Frame #10**

1. In the calendar view, the days where classes are scheduled should be tinted. This calendar view should stay in place.
2. The day selected should be outlined in bold.
3. In the body of the frame, the user can then scroll and view his/her list of classes scheduled for that day selected.
4. Only classes scheduled for the day selected will show in this frame.
5. If the user touches “edit”, frame #4 will open in edit mode. After making edits, when the user touches “save and finish” in frame #4, he/she should be taken back to frame #10. Any changes the user makes must be saved and integrated throughout the application.
6. The user can ‘mark’ (touch) whether he or she attended the class or whether the class was canceled. I.e. due to a Holiday. This ‘mark’ will be auto saved. All week to date and term to date attendance rates in this frame and frame #3 should be immediately updated.
7. All other fields in this frame #10 are displayed in read only mode.

**Frame #11**

1. The day selected should be outlined in bold and the calendar view should stay in place.
2. For any given class – If the day selected in the above calendar is between and including the ‘date of first class’ and ‘date of last class’, then that class will be seen in this frame.
3. If the user touches “edit”, frame #4 will open in edit mode. After making edits, when the user touches “save and finish” in frame #4, he/she should be taken back to frame #11. Any changes the user makes must be saved and integrated throughout the application.
4. The “# of targeted weekly study sessions” is pre-populated from frame #4.
5. The user can scroll and enter the number of study sessions he/she completed for each class. These completed study sessions are logged under the day selected in the above calendar. After the user enters the number of study sessions, the user must touch ‘submit’ for this number to be saved to the database. This number should also be reflected in frame #3.
6. Based on the user entering the number of study sessions and then touching ‘submit’, all week to date and term to date completed study session rates should be immediately updated in this frame and in frame #3.
7. All other fields in this frame are displayed in read only mode.

**Frame #12**

1. In the calendar view, the days where there are class activities due should be tinted. This calendar view should stay in place.
2. The day selected should be outlined in bold.
3. Only class activities with a ‘due date’ for the day selected will show in this frame.
4. To the right of “class name”, if the user touches “edit” then frame #8 will open in edit mode. When the user touches “save and finish” in frame #8, the user should be taken back to frame #12. Any changes the user makes must be saved and integrated throughout the application.
5. The user can mark (touch) whether he/she completed the class activity. This ‘mark’ should first be auto saved to the database and reflected in frames #7 and #8.
6. After the user enters a grade (a free text field to accommodate a letter, number or a percent grade), the user must touch ‘submit’ for the entered grade to be saved to the database. This grade should also be reflected in frames #7 and #8.

**Other Requirements: (Subject to developer team time constraints)**

1. The tool should allow users to exit the application at any place in the application and then easily enter back into the application at that same place where the user exited.
2. Due dates for classes and due dates for class activities should be integrated with the mobile device’s native calendar.
3. Users should be notified two days before a class activity is due.
4. On the first day of every week, users should receive warning alerts for classes where his/her ‘term to date’ attendance rates is less than 75% and congratulation notices for those classes where the attendance rate is over 75%.
5. On the first day of every week, users should receive warning alerts for classes where his/her ‘term to date’ completion rates for study sessions is less than 80% and congratulation notices for those over 80%.
6. From frame #5, users should be able to easily email or call their professors and TA’s.