**C868 – Software Capstone Project Summary**

**Task 2 – Section C**

|  |  |
| --- | --- |
| **Capstone Proposal Project Name:** | http://www.idevnews.com/views/images/uploads/general/wgu_logo.png  StudyTrackr |
| **Student Name:** | James Travis Mundy |

**Table of Contents**

[Table of Contents 2](#_Toc63972409)

[Application Design and Testing 5](#_Toc63972410)

[Design Document 5](#_Toc63972411)

[Class Design 5](#_Toc63972412)

[UI Design 6](#_Toc63972413)

[Unit Test Plan 7](#_Toc63972414)

[Introduction 7](#_Toc63972415)

[Test Plan 8](#_Toc63972416)

[Source Code 16](#_Toc63972417)

[Link to Live Version 16](#_Toc63972418)

[User Guide 17](#_Toc63972419)

[Introduction 17](#_Toc63972420)

[Installation and Using the Application 17](#_Toc63972421)

[Guide for Maintaining Application 17](#_Toc63972422)

[Prerequisites 17](#_Toc63972423)

[Installation 18](#_Toc63972424)

[Guide for Students 19](#_Toc63972425)

[Managing Student Accounts 19](#_Toc63972426)

[Create Student Account 20](#_Toc63972427)

[Edit Student Account 22](#_Toc63972428)

[Deleting Student Account 26](#_Toc63972429)

[Logging In and Using StudyTrackr 29](#_Toc63972430)

[Logging into StudyTrackr 29](#_Toc63972431)

[Managing Terms 30](#_Toc63972432)

[Creating a Term 30](#_Toc63972433)

[Editing a Term 32](#_Toc63972434)

[Dropping a Term 34](#_Toc63972435)

[Courses 36](#_Toc63972436)

[Adding a Course to a Term 36](#_Toc63972437)

[Editing a Course 39](#_Toc63972438)

[Dropping a Course 42](#_Toc63972439)

[Course Notes 44](#_Toc63972440)

[Adding a Course Note 44](#_Toc63972441)

[Editing a Course Note 48](#_Toc63972442)

[Sharing a Course Note 53](#_Toc63972443)

[Deleting a Course Note 58](#_Toc63972444)

[Course Assessments 62](#_Toc63972445)

[Adding a Course Assessment 62](#_Toc63972446)

[Editing an Assessment 66](#_Toc63972447)

[Deleting an Assessment 71](#_Toc63972448)

[Personal Notes / Course Notes 75](#_Toc63972449)

[Create a Personal Note / Course Note 75](#_Toc63972450)

[Editing Personal / Course Note 79](#_Toc63972451)

[Sharing a Personal Note / Course Note 83](#_Toc63972452)

[Deleting a Personal / Course Note 87](#_Toc63972453)

[Term Report 90](#_Toc63972454)

[View Term Report 90](#_Toc63972455)

# Application Design and Testing

## Design Document

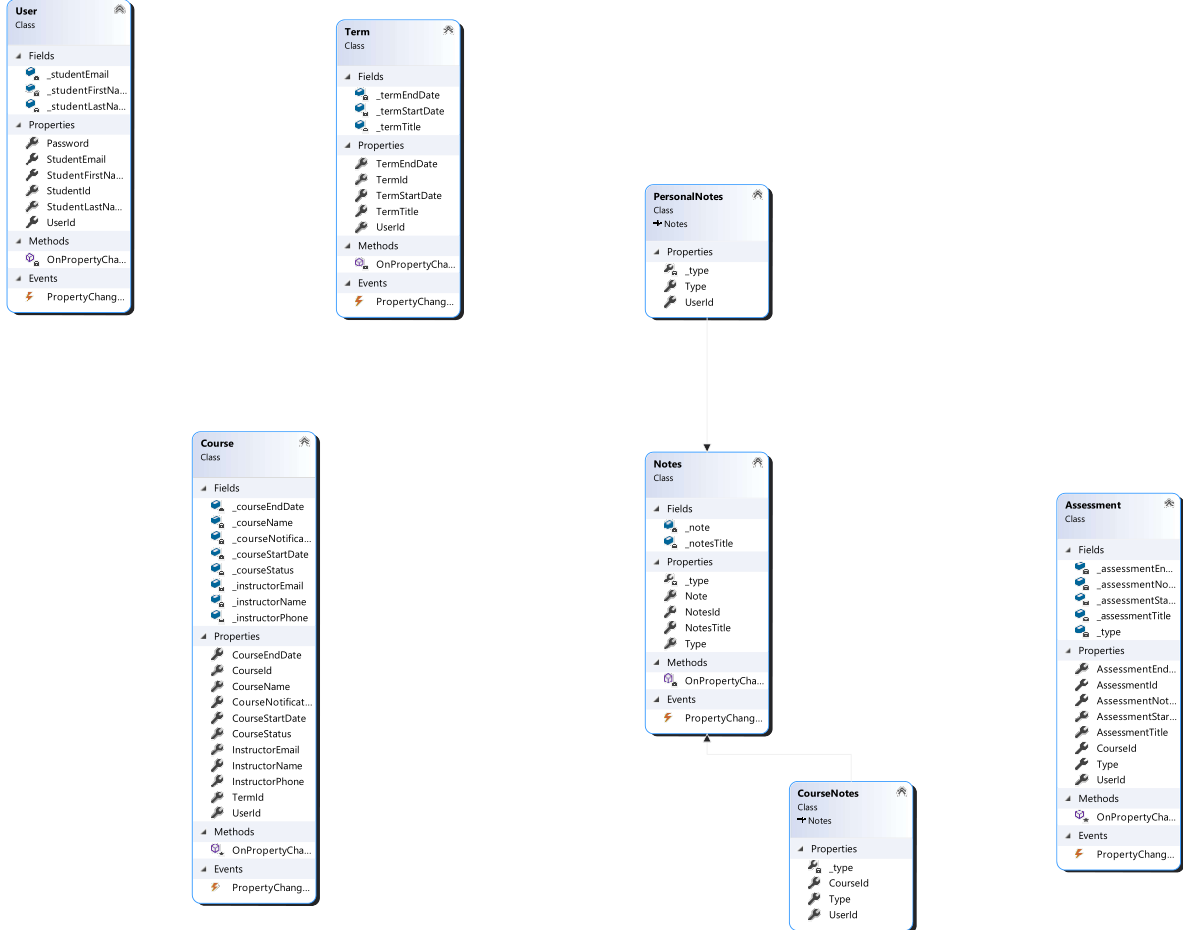
### Class Design

The figures below show the class design for the StudyTrackr mobile application. Applications using the Xamarin.Forms framework are made with the MVVM (Model, View, ViewModel) design pattern in mind. I have chosen to show the model classes that make up the underlying foundation of the application.

The main classes are the user, term, course, notes, and assessments classes. There is a one-to-many relationship between user, term, course, notes, and assessments. A one-to-many relationship between term, course, notes, and assessments. A many-to-many relationship between course, notes, and assessments. A user can have many terms, courses, notes, and assessments. A term can have one user, but have many courses, notes, and assessments. A course can have one user, one term, but have many notes and assessments. There is also a polymorphic relationship between notes, course notes, and personal notes. This is indicated by the type property, which was implemented because a student may need to create notes that do not pertain to any courses.

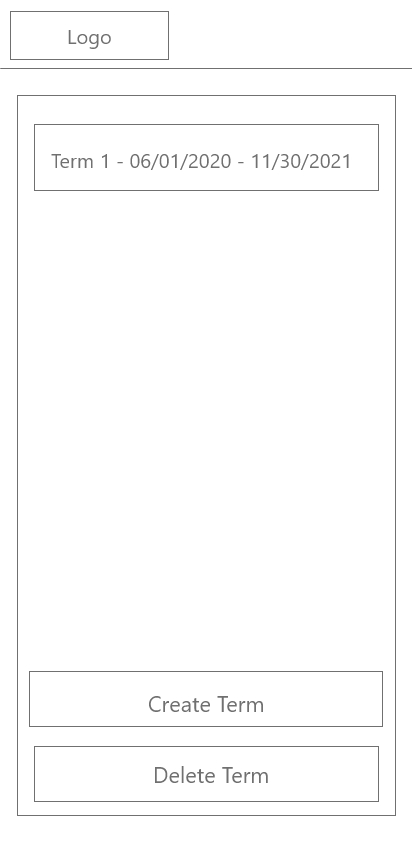
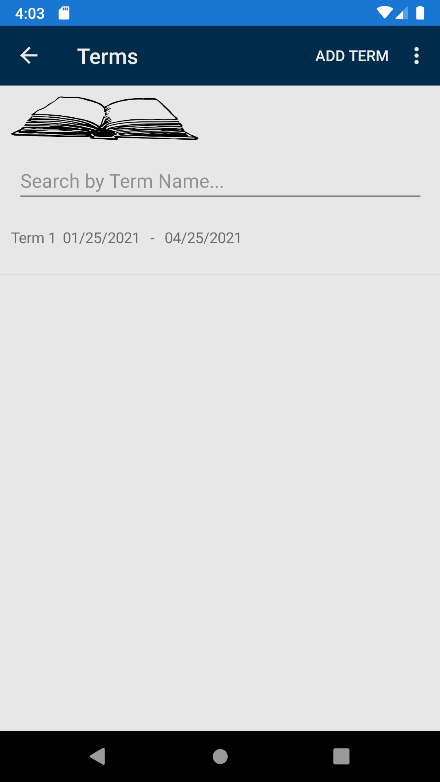
The OnPropertyChanged method and PropertyChanged events that are common between the classes are used by the INotifyPropertyChanged interface that each base class implements. This is necessary for value changes to reflect in the mobile application’s UI.

The database schema for SQLite is generated directly from these classes. Each table name would be the class name and each field would be a database column.



### UI Design

The images below are the low and high-fidelity examples from the terms page. The university logo should be at the top left, with a screen in the middle that lists the term and the term beginning and end dates. Based on the requirements document, a “Create Term” and “Delete” term button were added. However, in the high-fidelity mockup the design was changed slightly; the “Create Term” button was moved up to the tool bar. To delete a term the student must first select it by tapping it. It was found that this design and flow choice made a lot more sense during initial testing. Note the secondary menu on the toolbar indicated by the 3 dots; tapping the 3 dots brings up another menu to navigate to the notes screen to add a personal or course note.

Low Fidelity Wire-Frame High-Fidelity Design Prototype

## Unit Test Plan

### Introduction

#### Purpose

To test the validation class against the values of other classes, I created unit tests for the validation class using the NUnit testing framework. Since the validation class is central to sanitizing the data before it is saved to the database, I wanted to test it against various cases where the wrong data or no data is input for the various classes.

#### Overview

The application, at its core, is a data tracker. The data stored in the application is very important to a student’s success or failure and should be validated and sanity checked before being saved to the database. The unit tests are written utilizing the C# programming language with the NUnit testing framework. During tests, Visual Studio displays whether the test passed or failed in a separate console.

### Test Plan

#### Items

To test whether the validation class and its methods are up to the task, I created objects for each test that contain data that needs to be checked for sanity; namely, the user and course classes contain email address fields. Additionally, the course class contains a phone number field for an instructor. Any object could be used to test whether a particular field is empty or null.

#### Features

Each test creates an object and then uses the validation class and one of its methods against the objects properties to test if the validation class and the method are working as intended.

For instance, the validation class contains three methods:

* IsPhoneNumberValid() - Acts as a sanity check for the number entered, in that it is a reasonable amount and combination of numbers that could be classified as a “phone number”
* IsEmailValid() - Acts as a sanity check that the value provided can reasonably be an email – it contains an @ symbol, ends in a domain, and follows guidelines for an email address
* IsFieldNull() - Acts as validation that the field is not null or empty.

Before being able to test these methods, each class needs to be instantiated using the “new” keyword and assigning values to the fields.

The validation methods are invoked against fields in the newly instantiated classes. For instance, a user class contains an email field; Validation.IsEmailValid() is invoked against it. Similarly, IsPhoneValid() is invoked against the instructor phone field in the course class. The IsFieldNull() method is invoked against all fields in a class before being saved to the database. Any class can be used to test using the IsFieldNull() method.

#### Deliverables

When these tests are run, Visual Studio opens a new “Test Explorer” window that lists each test, test type, the variables, and whether it passed (Green Check) or failed (Red X).

##### Tasks

1. Create the code to be tested.
2. Install the NUnit framework.
3. Create test code utilizing NUnit methods.
4. Run the test using Visual Studio.
5. Examine the output to determine what test passed or failed.

##### Needs

Visual Studio will need to be running and then the tests run.

Requirements:

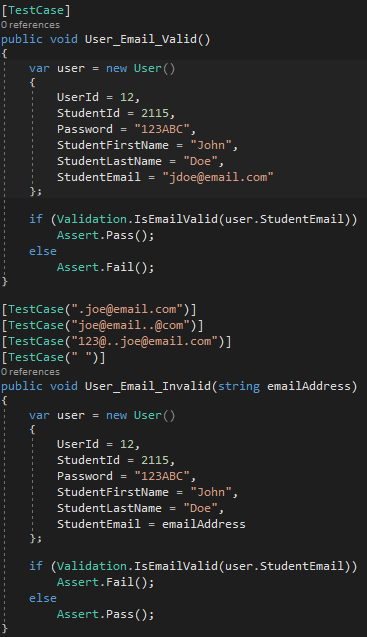
* NUnit 3.13.1 (NUnit test framework)
* Visual Studio 2019 (IDE that is used for development and testing)

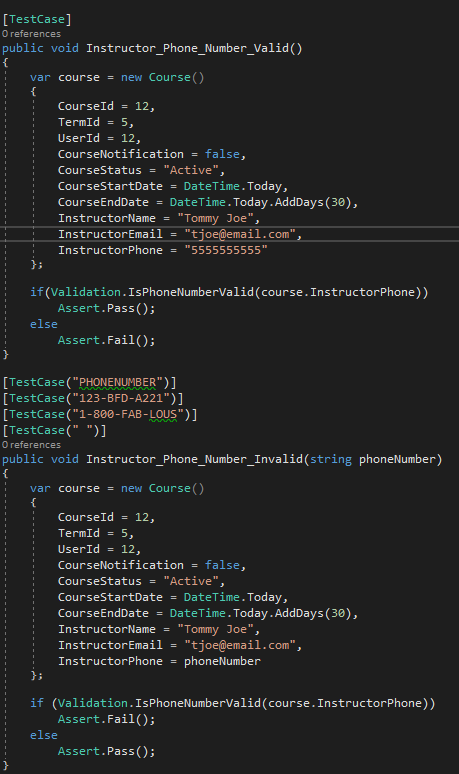
#### Pass/Fail Criteria

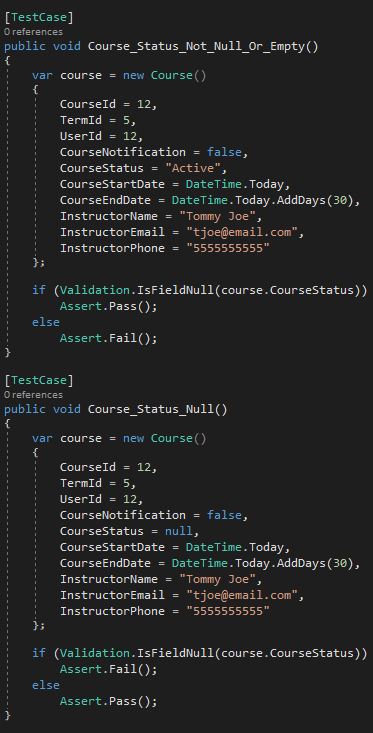
The criteria for a “pass” is a green check mark in the output from Test Explorer, which indicates that the test completed successfully and had the expected outcome. A failing test would be a red x, which indicates that the test was either not able to be completed or that it did not have the expected outcome.

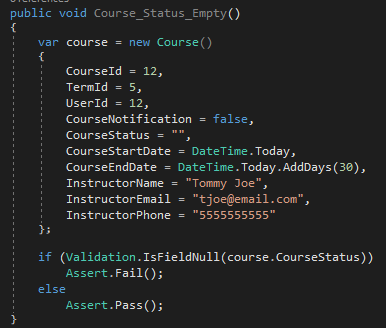
#### Specifications

Provided are screenshots of the test code mentioned above. These test methods are a part of the overall test class.









#### Procedures

The test method User\_Email\_Valid() instantiates a user object, uses the IsEmailValid() method from the validation class to do a sanity check on the student email field. If the criteria are met for it to potentially be a valid email address, the test passes.

User\_Email\_Invalid() has a string parameter, emailAddress, where each test case above is treated as a parameter and a separate test. Each test case provides a string parameter that is used in the creation of a user object as the email address. The field, emailAddress, is tested against the validation class’s IsEmailValid() method. If the criteria are met for it to be an invalid email address, the test passes.

Instructor\_Phone\_Number\_Valid() instantiates a course object and then uses the IsPhoneNumberValid() method from the validation class to do a sanity check on the instructorPhone field. If the criteria are met that it could be a valid phone number, the test passes.

Instructor\_Phone\_Number\_Invalid() has a string parameter, phoneNumber, where each test case above is treated as a parameter and a separate test. Each test case provides a string parameter (phoneNumber) that is used in the creation of the course object as the instructor phone number, instructorPhone. If the criteria are met that for it to be an invalid phone number, the test passes.

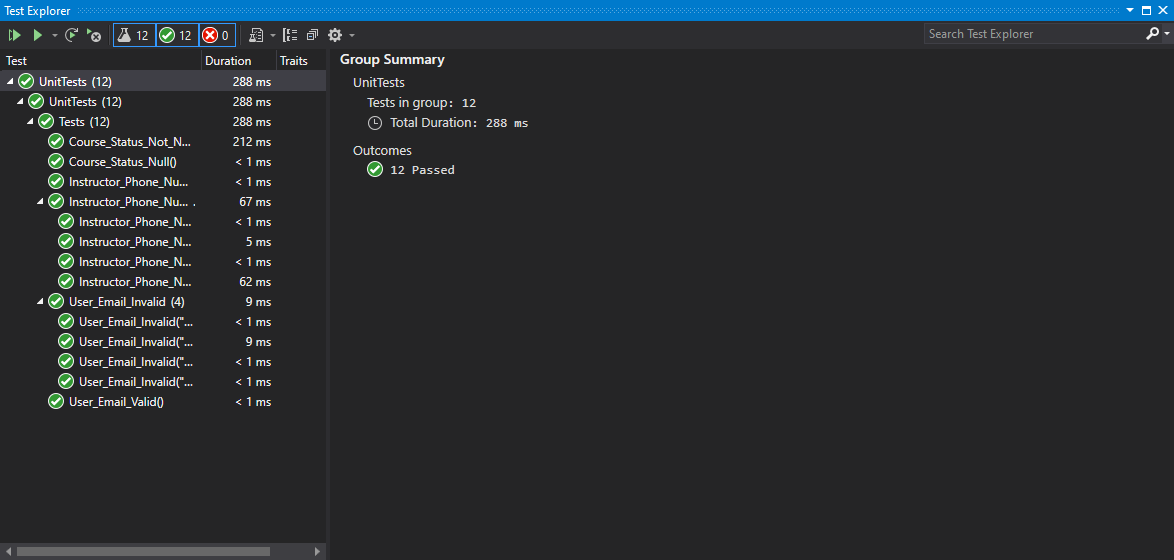
Course\_Status\_Not\_Null\_Or\_Empty() instantiates a course object and then uses the IsFieldNull() method from the validation class to check whether the field is empty or null. If the criteria are met that indicates the field is not empty or null, the test passes.

Course\_Status\_Null() instantiates a course object and then uses the IsFieldNull() method from the validation class to check whether the field is empty or null. If the criteria are met that the field is null, the test passes.

Course\_Status\_Empty() instantiates a course object and then uses the IsFieldNull() method from the validation class to check whether the field is empty. If the criteria are met that the field is empty, the test passes.

#### Results

The following screenshot from Test Explorer is generated when running the above tests in Visual Studio. The green check marks indicate that each test ran successfully, there were no errors in the test, and no unexpected outcomes.



# Source Code

See additional file also submitted, StudyTrackr.zip

# Link to Live Version

Not Available.

# User Guide

## Introduction

Included is a guide for maintaining the application using a Windows 10 workstation with Visual Studio 2019 installed. Additionally, there is a guide for students who will use this application day-to-day.

## Installation and Using the Application

This mobile application will need to be installed on an android mobile device using whatever app store AU ultimately decides on. Alternatively, the application can be installed manually – also referred to as “sideloading” if the apk is made available by AU.

## Guide for Maintaining Application

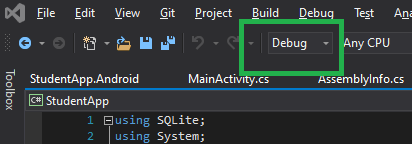
Several steps and technologies are required to setup and run this mobile application locally on a Windows 10 desktop for the purposes of debugging, making changes, or adding additional features.

### Prerequisites

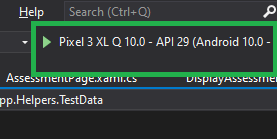
* Visual Studio 2019
* Android SDK
* Xamarin.Forms
* Android Virtual Machine or Android Device that meets minimum requirements

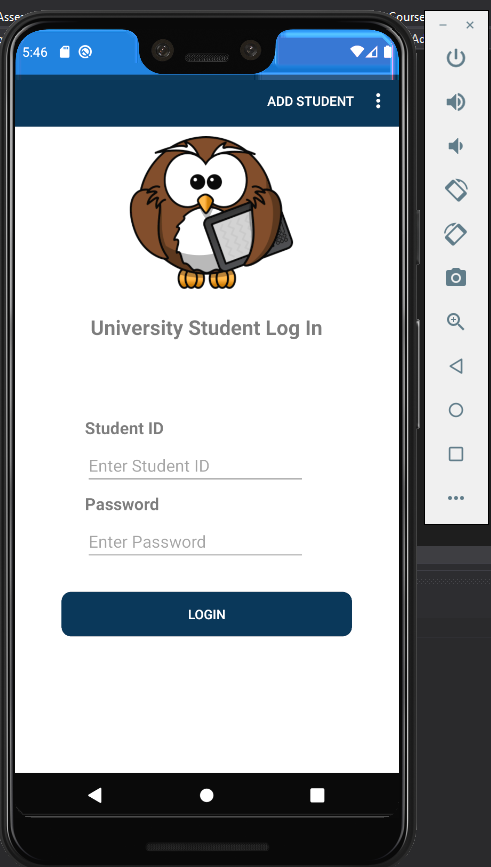
### Installation

1. Unzip the archive provided, StudyTrackr.zip.
2. Navigate to the StudyTrackr directory and double click the solution file, StudyTrackr.sln to open it in Visual Studio.
3. Set up an Android VM or Connect an Android Smartphone that meets the minimum requirements. Please see <https://docs.microsoft.com/en-us/windows/android/emulator> for additional information.
4. For debugging make sure the Solution is configured for “Debug” mode.



1. Click on the Green “Play” button. If everything was installed correctly, the application will build, and the Android VM (or Android Smartphone) will open with the app running.





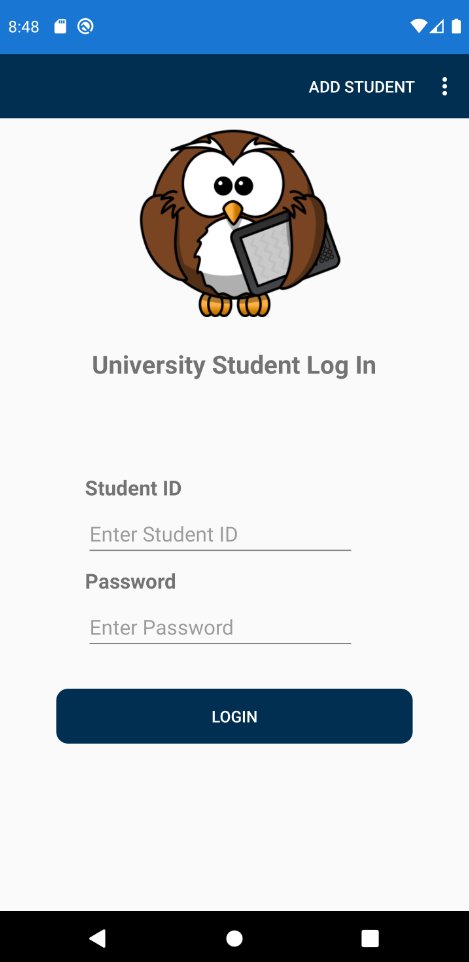
# Guide for Students

To log in, the student will need to create a student account. Please see “Create Student Account” below and then “Logging into StudyTrackr”.

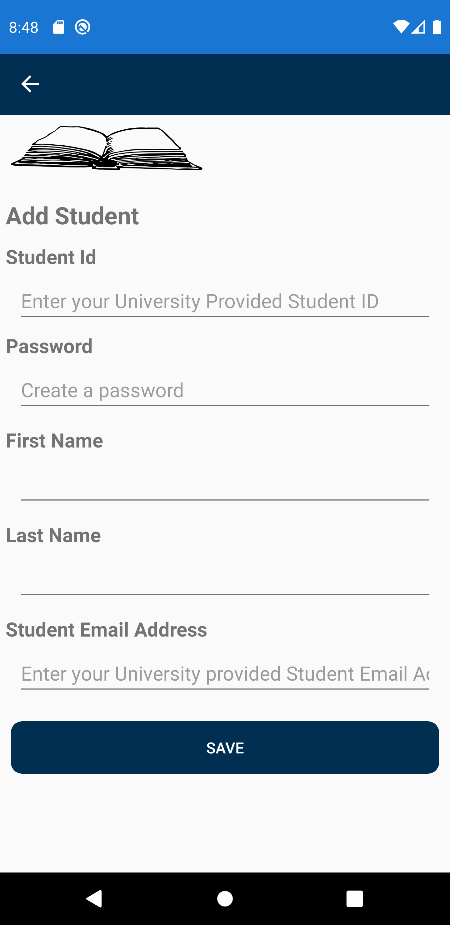
## Managing Student Accounts

Student accounts are created using AU provided information and are used to hold education data that pertains directly to the student.

### Create Student Account

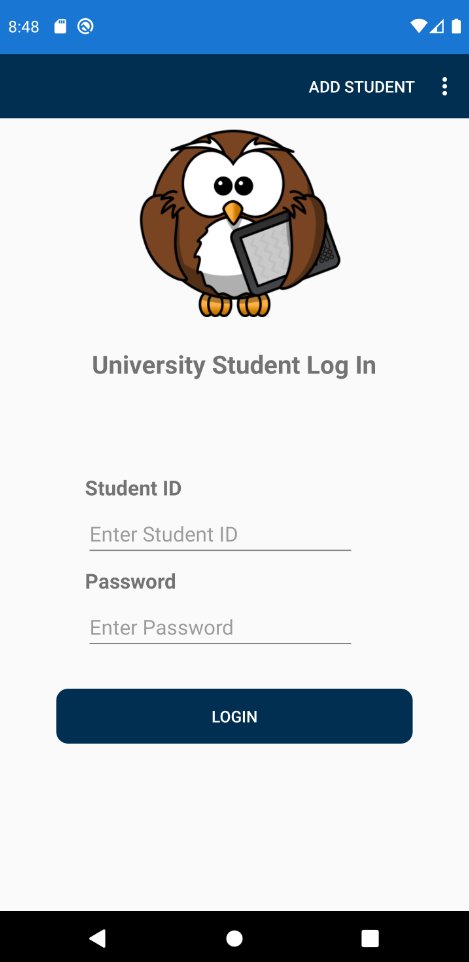


1. Tap the “Add Student” tool bar item in the top right. 

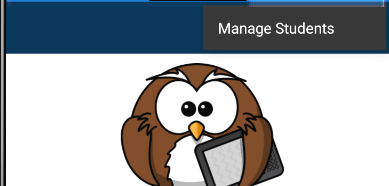


1. Enter the Archimedes University provided information, along with your first name, last name, a password of your choosing, and tap Save. The student account is now created!

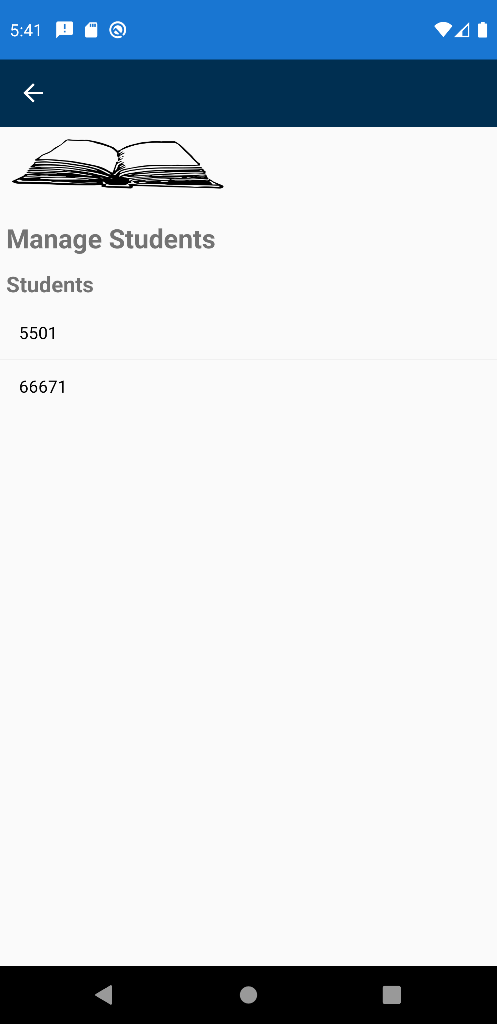
### Edit Student Account



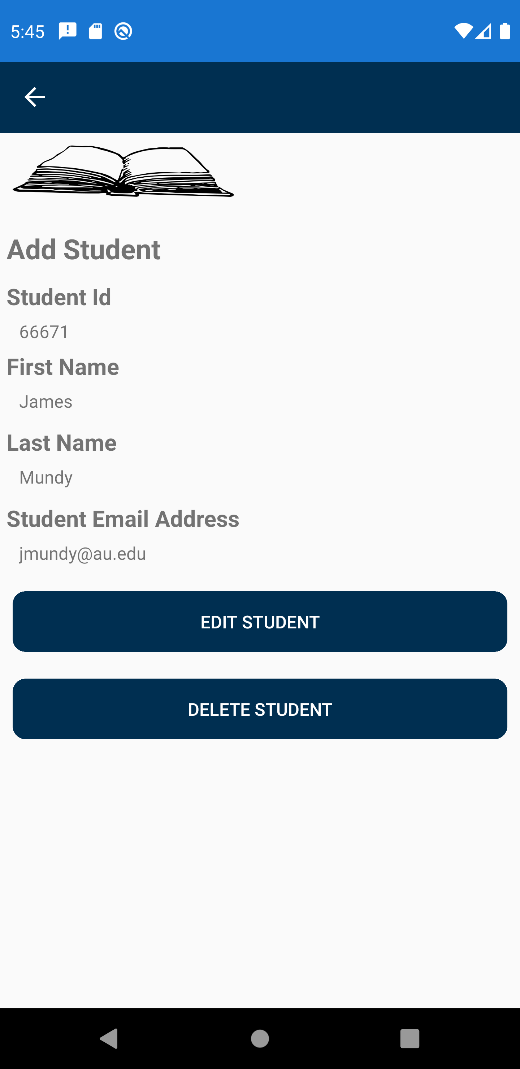
1. Tap on the 3 dots tool bar item. 



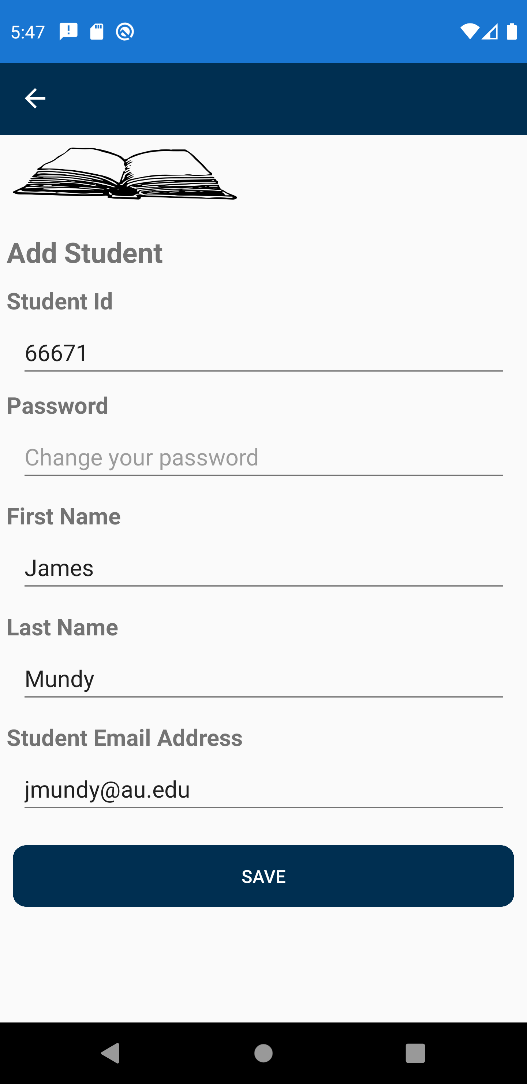
1. On the secondary tool bar, tap “Manage Students”.



1. Tap on a Student from the student list.



1. Tap “Edit Student”.

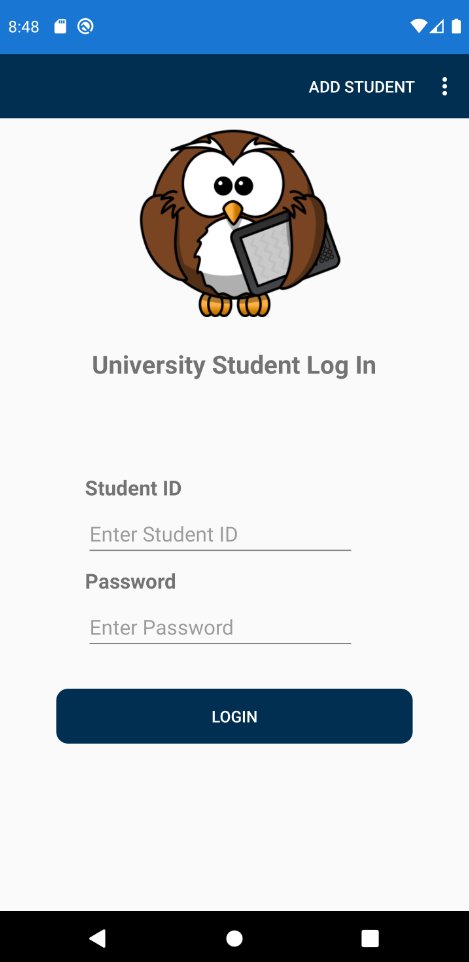


1. Make any desired changes.

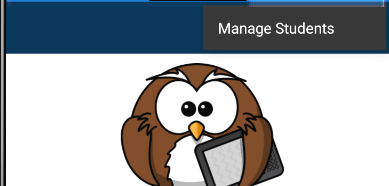
*Note: The Student Id cannot be changed. If this type of change needs to be done, delete this student account and recreate. Also, this is where you would change your password if desired.*

1. Tap “Save”.

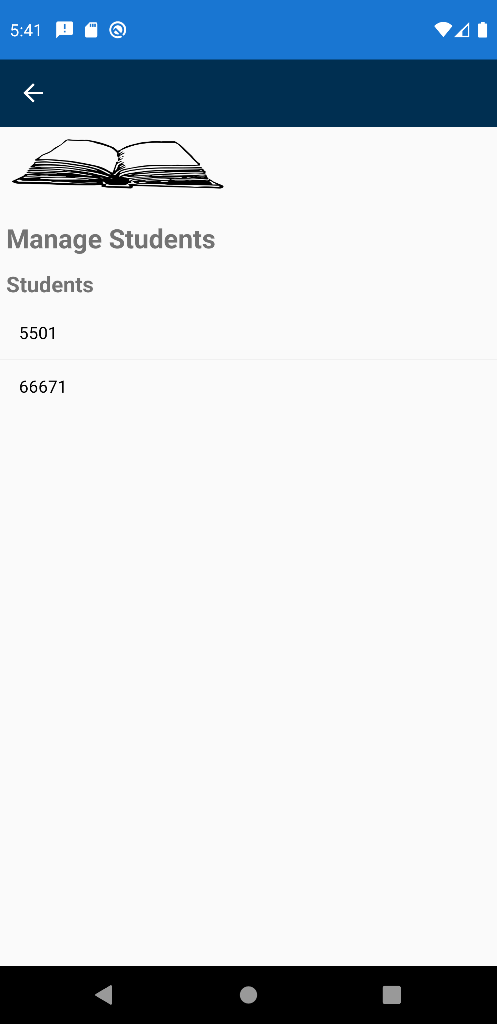
### Deleting Student Account



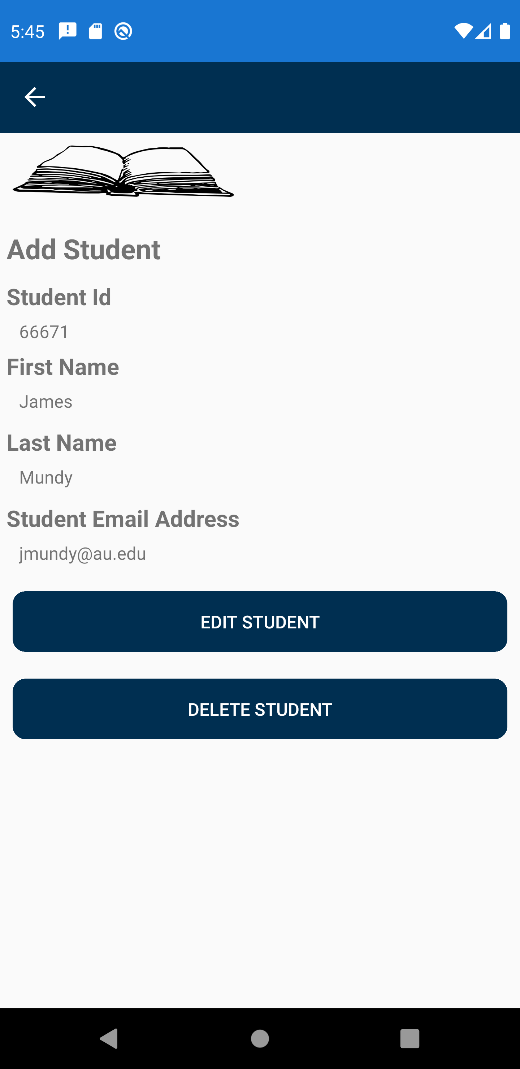
1. Tap on the 3 dots tool bar item. 



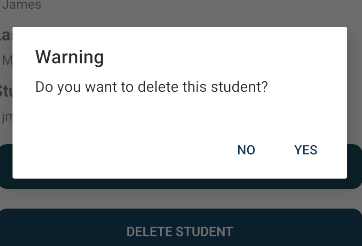
1. On the secondary tool bar, tap “Manage Students”.



1. Tap on a “Student” from the student list.



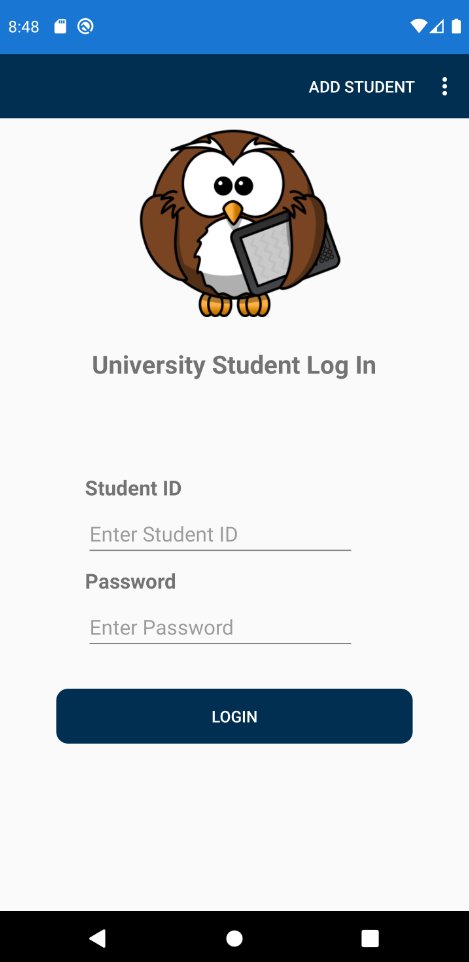
1. Tap “Delete Student”.



1. Tap “Yes” or “No”. Deleting a student will effectively delete all that student’s data from this application. Terms, courses, notes, assessments will be deleted.

## Logging In and Using StudyTrackr

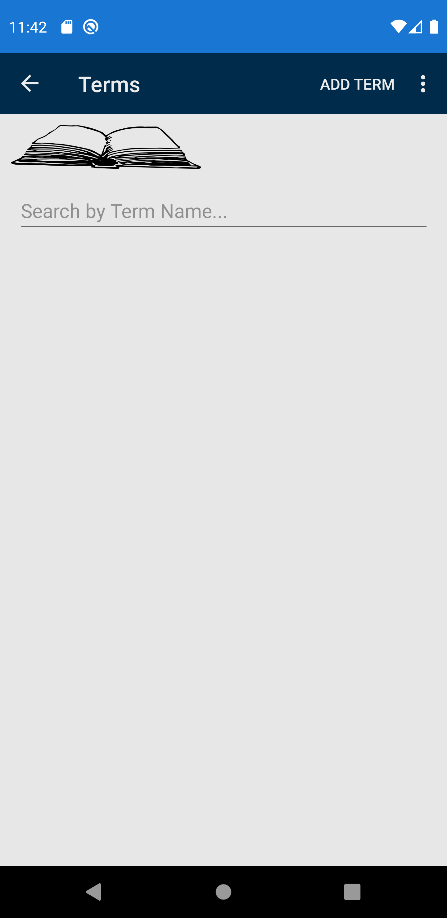
### Logging into StudyTrackr



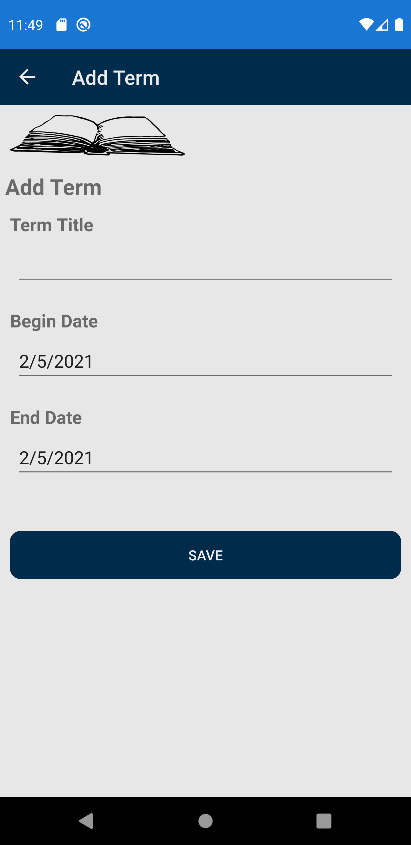
Enter the Student Id and Password for the account that was just created and tap “Login”.

## Managing Terms

### Creating a Term



1. Tap “Add Term” in the top right.

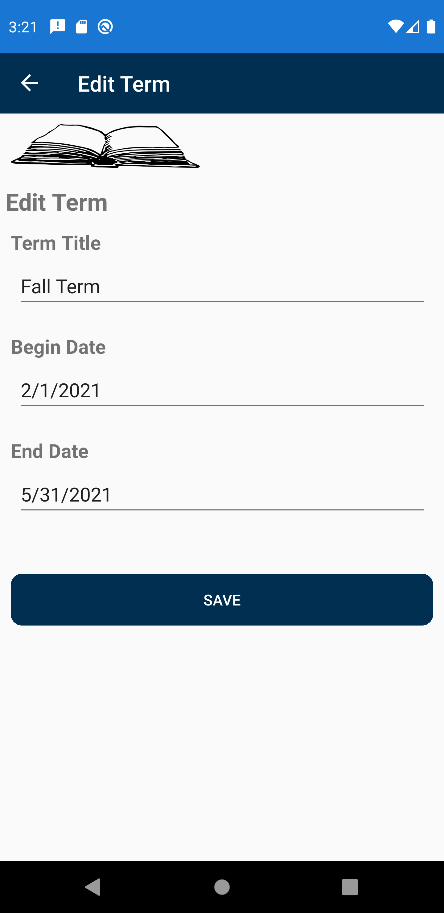


1. Enter a term title such as “Term 1” or “Summer Term”.
2. Choose the Start and End dates for the term.
3. Tap “Save”.

### Editing a Term



1. Tap on a Term on the Term Page.

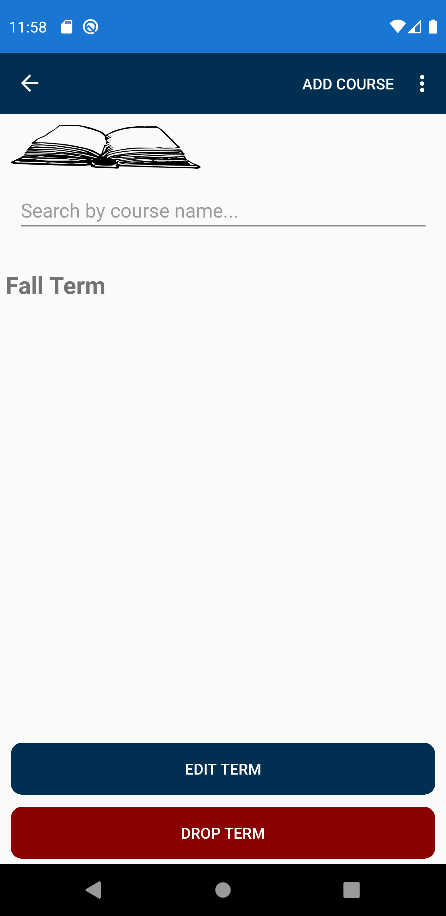


1. Make the desired changes to Term Title, Begin Date, or End Date.
2. Tap “Save”.

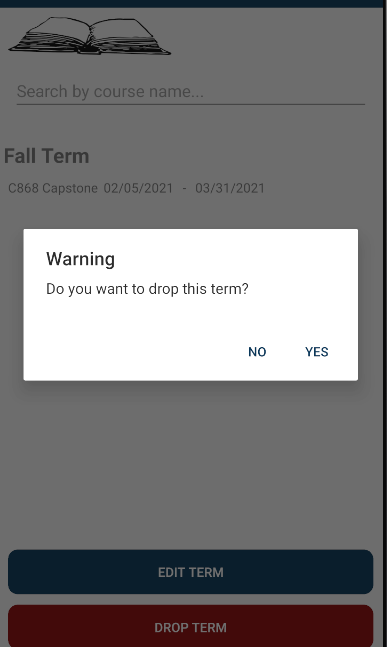
### Dropping a Term



1. Tap on a term from the Terms page.



1. Tap on “Drop Term”, a confirmation message pops up.



1. Tap “Yes” or “No”, if a Term is “Dropped” all notes, courses, and assessments created in that term will also be dropped.

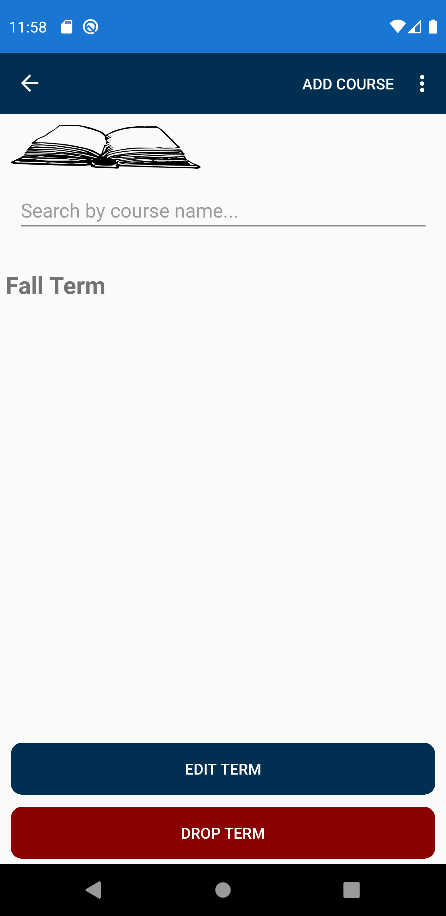
## Courses

Courses are used to hold information pertaining to a specific course and its assessments, notes, and instructor information.

### Adding a Course to a Term

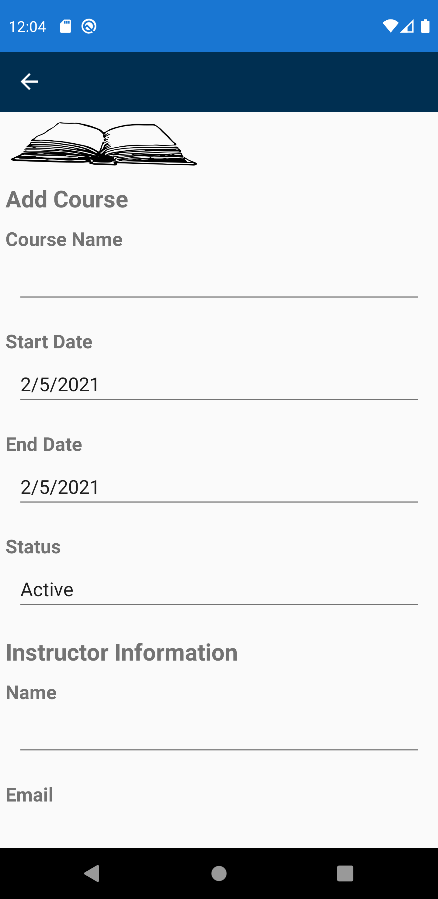
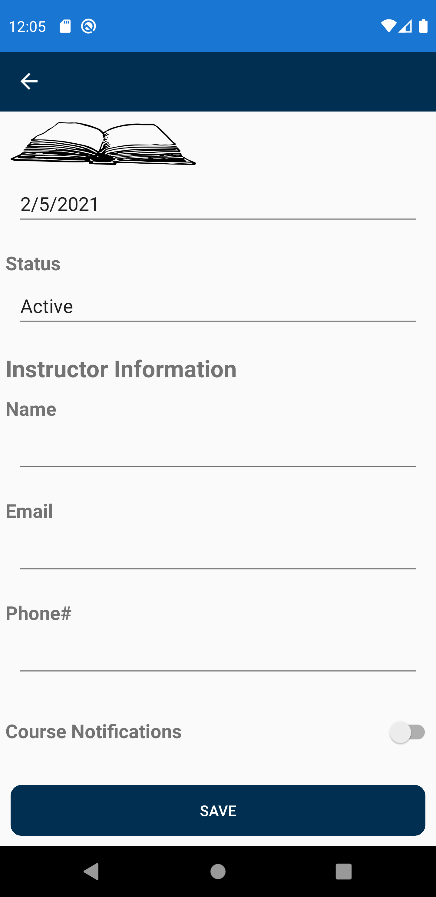


1. From the Term page, tap on one of the Terms.



*Note: The term can also be edited or Dropped from this screen as well.*

1. Tap “Add Course” in the top right. 

1. Enter the course name, it would be helpful to also include the course number.
2. Enter the Start and End dates for the course.
3. Set the Status to “Active”, “Inactive”, or “Complete”.

* Active means that you are currently enrolled and active in the course
* Inactive means that you have not yet started the course but are enrolled in it for this term
* Complete means that the course is complete

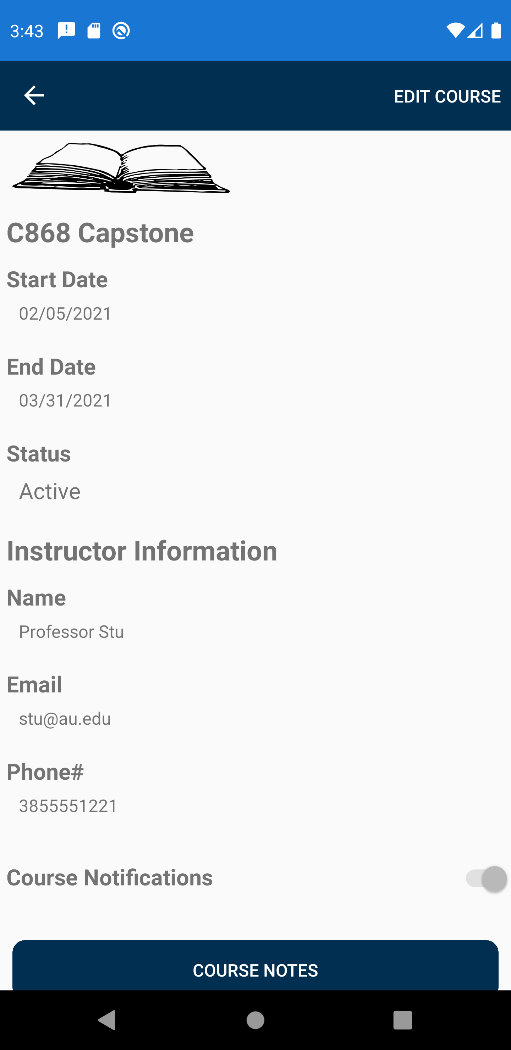
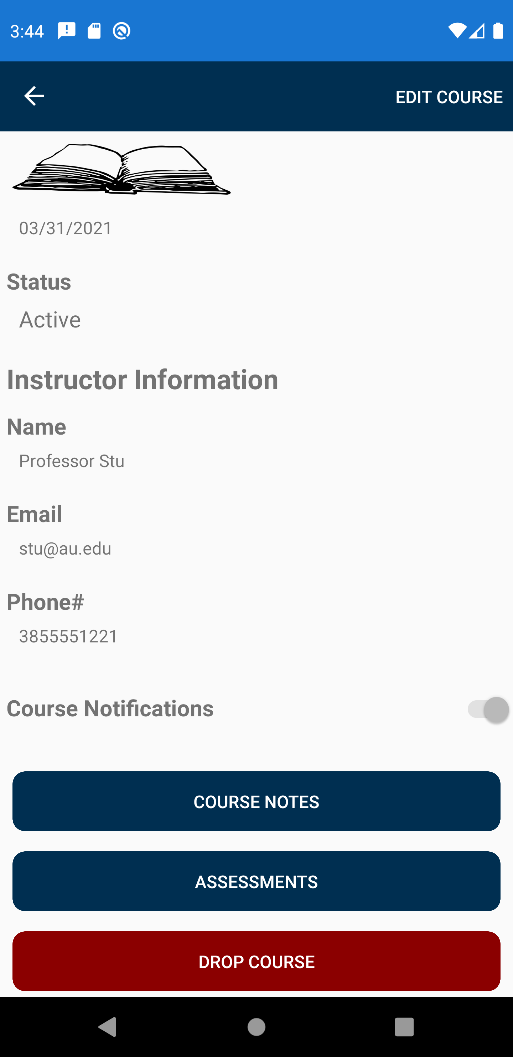
*Note: It is a good idea to change the course status as progress is made during the term so that you have accurate information.*

1. Enter the instructor’s name.
2. Enter the instructor’s email address.
3. Enter the instructor’s phone number.
4. Would you like to receive course notifications from the app regarding start and end dates? If so, toggle it to “on”.
5. Tap “Save”.

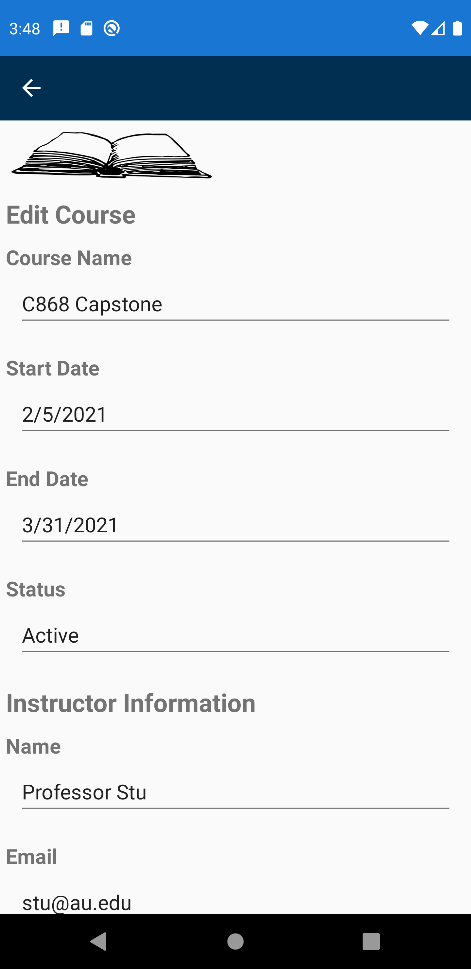
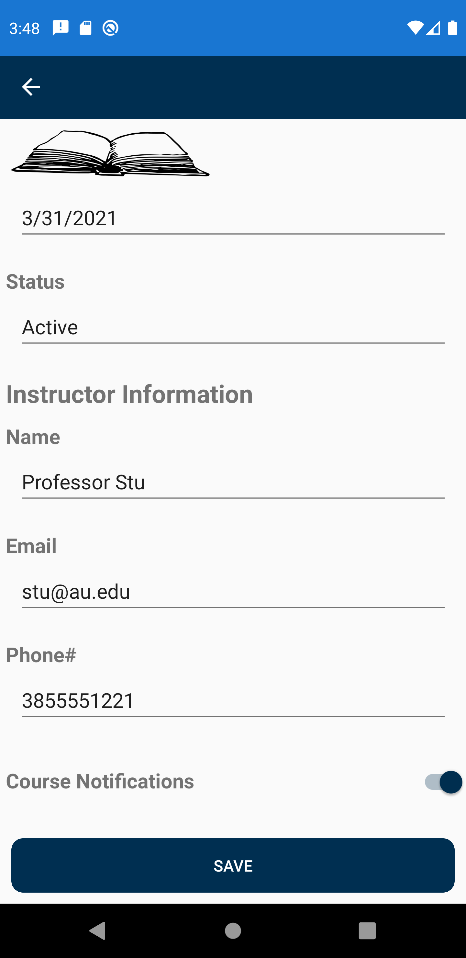
### Editing a Course



1. Tap on a Course in the Course list.

1. Tap on the “Edit Course” tool bar item in the top right. 

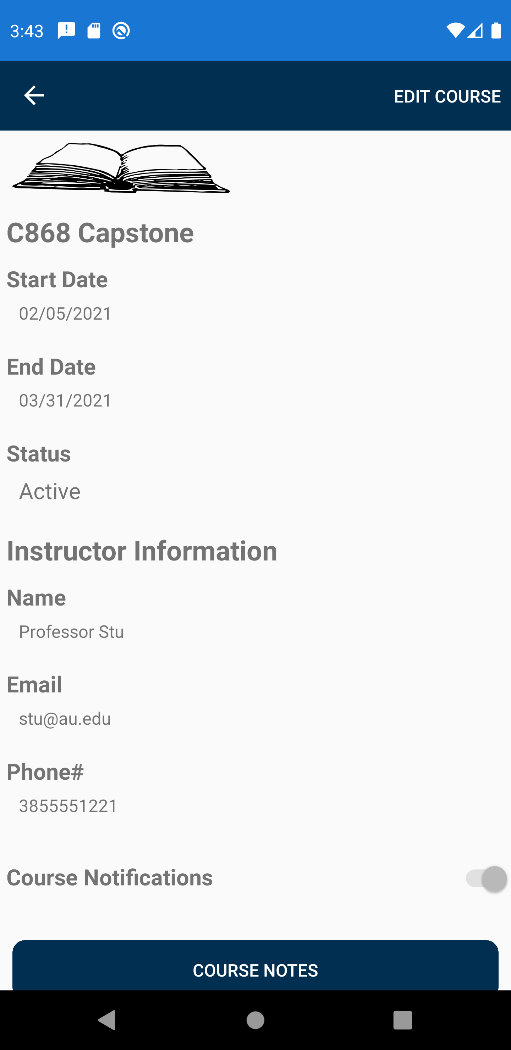
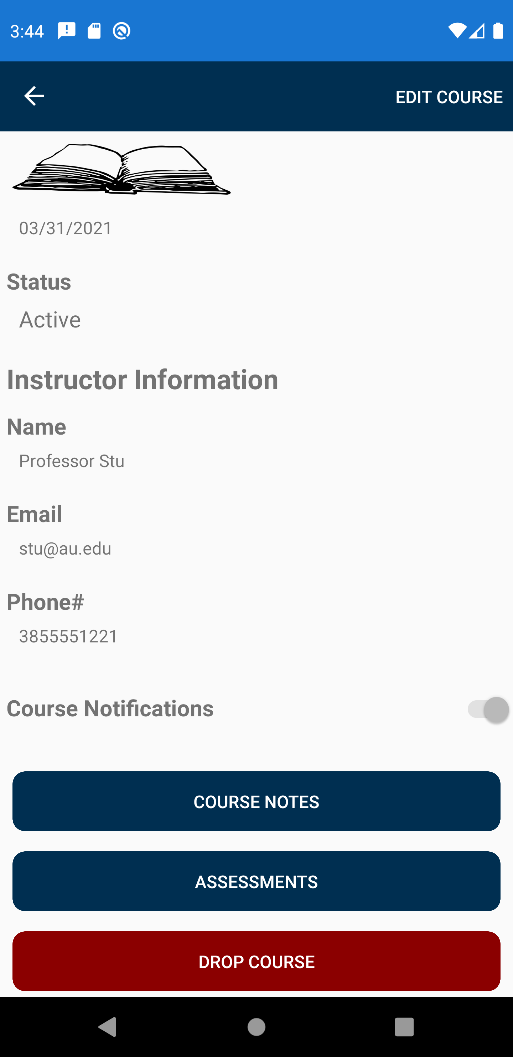
1. Make any desired changes in Course Name, Start Date, End Date, Status, Instructor Name, Email, Phone#, or Course Notifications.
2. Tap “Save”.

*Note: This is where the Course Status can be changed.*

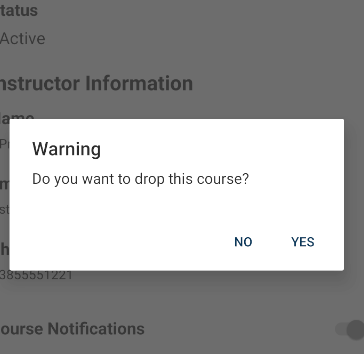
### Dropping a Course



1. Tap on a Course in the Course list on the Term page.

1. Scroll to the bottom of the Course page and tap “Drop Course”.



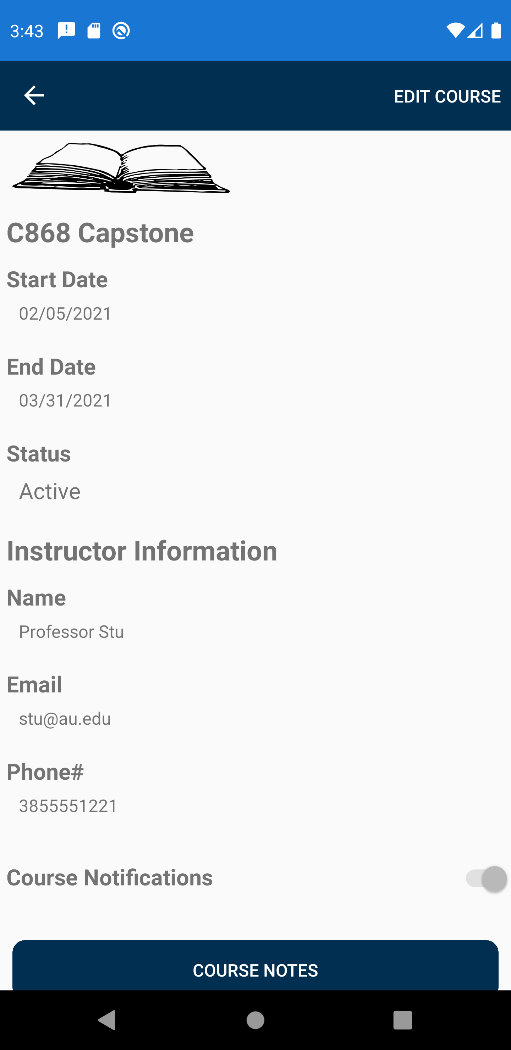
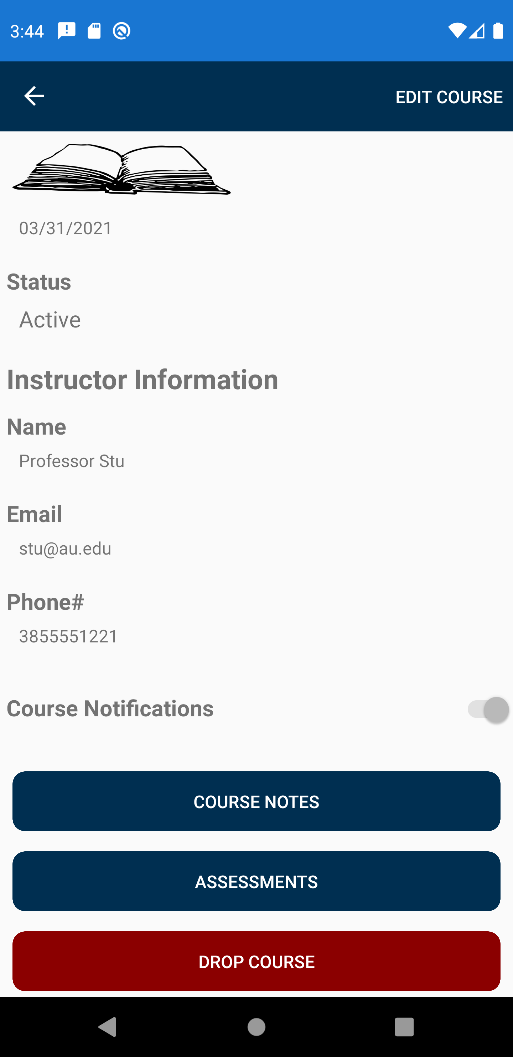
1. Tap “Yes” or “No”. If a Course is “Dropped” all Notes and Assessments associated with that Course will also be “dropped”.

## Course Notes

### Adding a Course Note



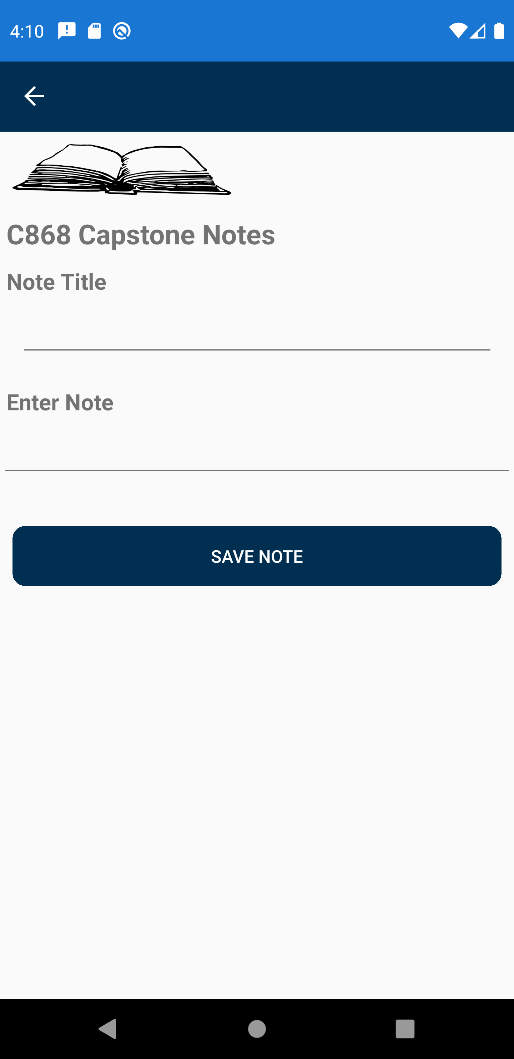
1. Tap on a Course from the course list on the Term page.

1. Scroll down to the bottom, Tap on “Course Notes”.



1. Tap on the “Add Note” tool bar item in the top right. 

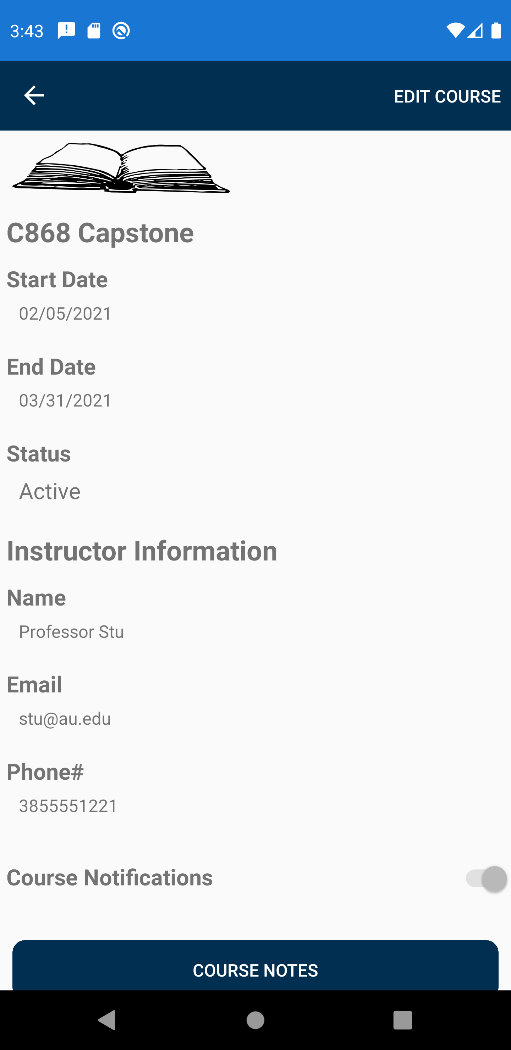
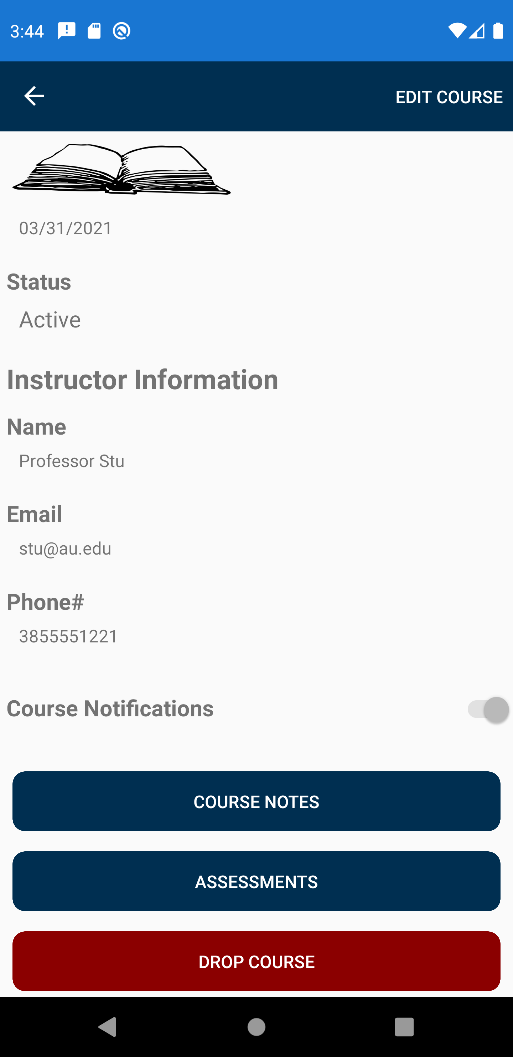


1. Enter a Note title and type out the note text under “Enter Note”.
2. Tap “Save Note”.

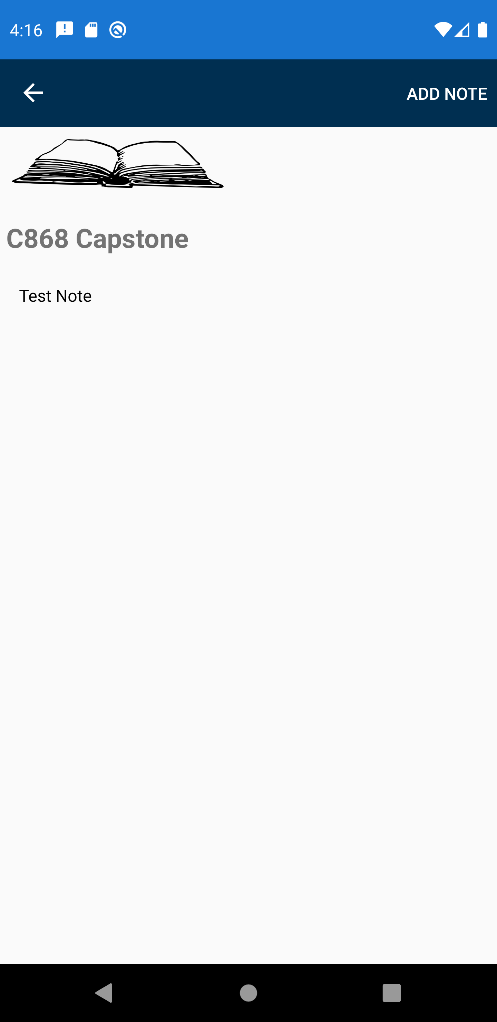
### Editing a Course Note



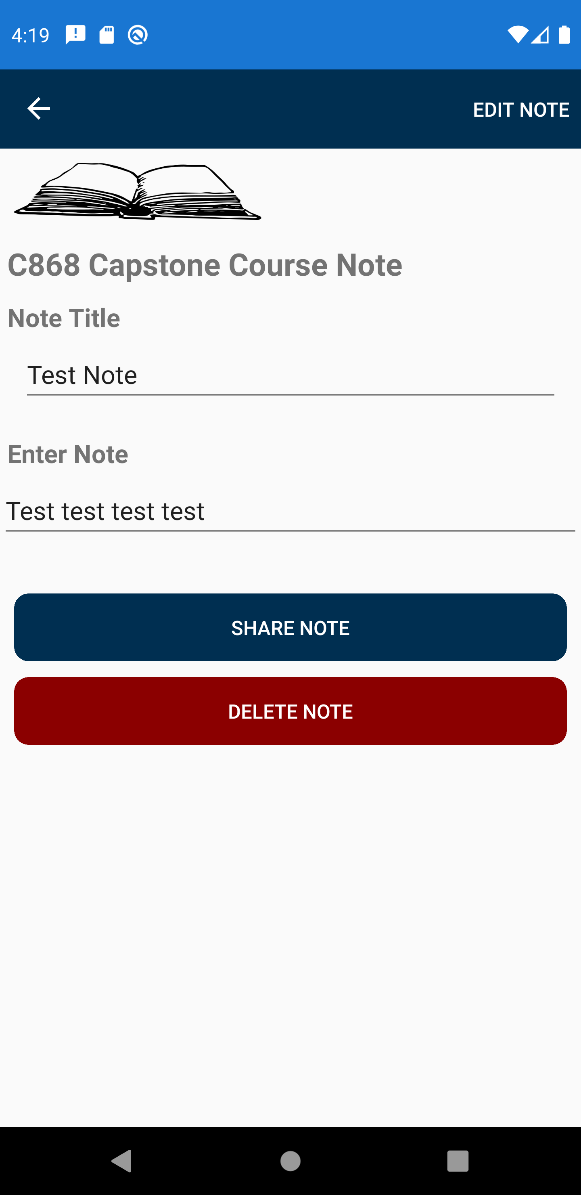
1. Tap on a Course from the course list on the Term page.

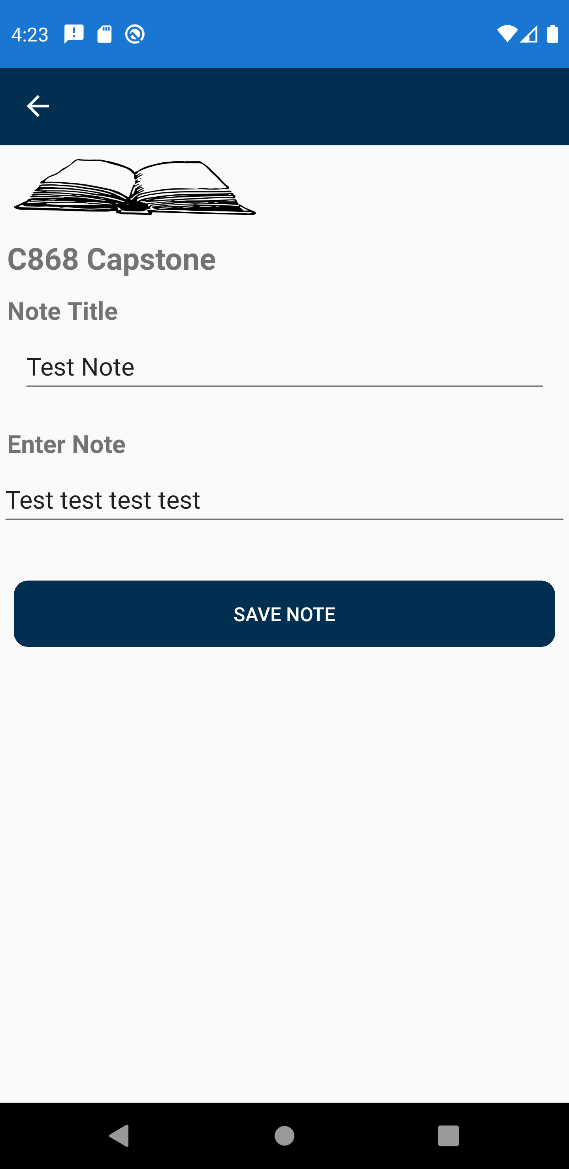
1. Scroll down to the bottom, Tap on “Course Notes”.



1. Tap on a Note in the note list on the Course Note page.



1. Tap on the “Edit Note” tool bar item in the top right, 

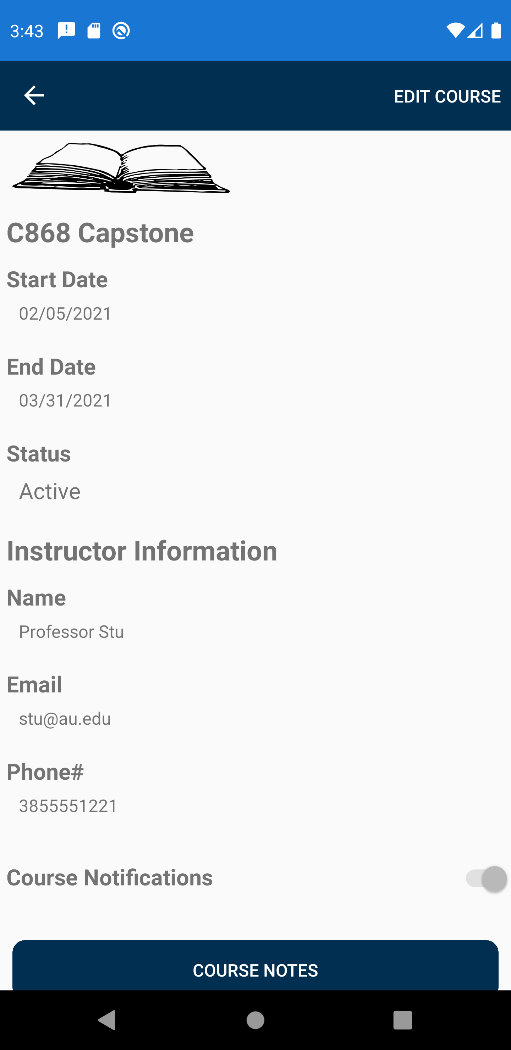
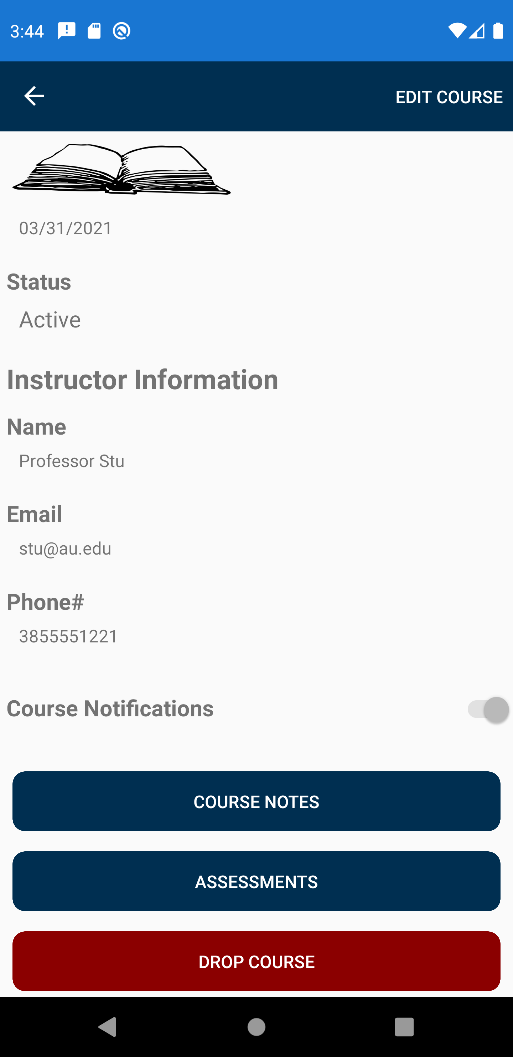


1. Make the desired changes in the Note Title or Enter Note text boxes.
2. Tap “Save Note”.

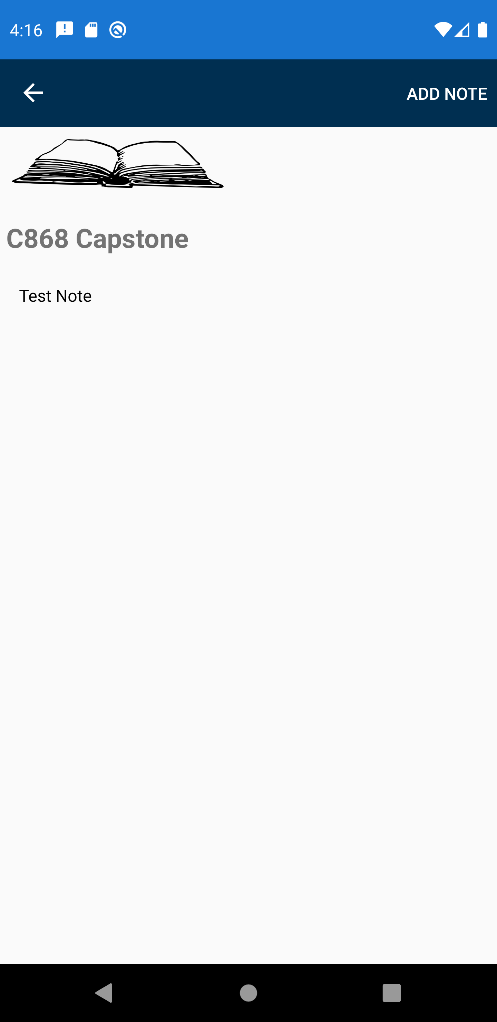
### Sharing a Course Note



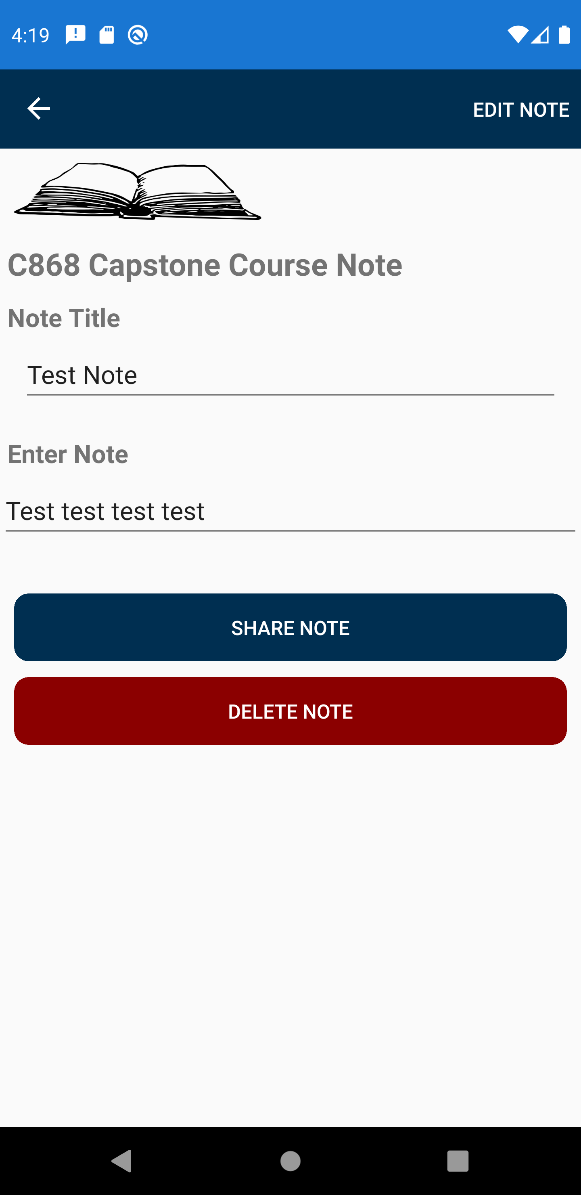
1. Tap on a Course from the course list on the Term page.

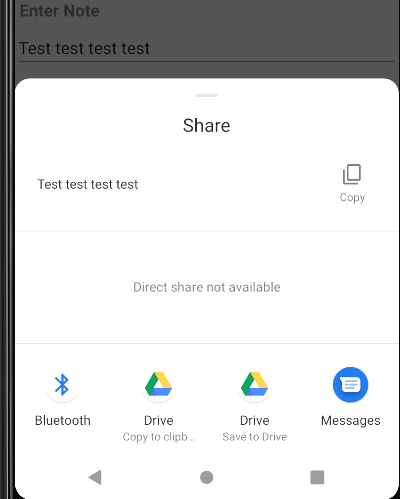
1. Scroll down to the bottom, Tap on “Course Notes”.



1. Tap on a Note in the note list on the Course Note page.



1. Tap on “Share Note”.

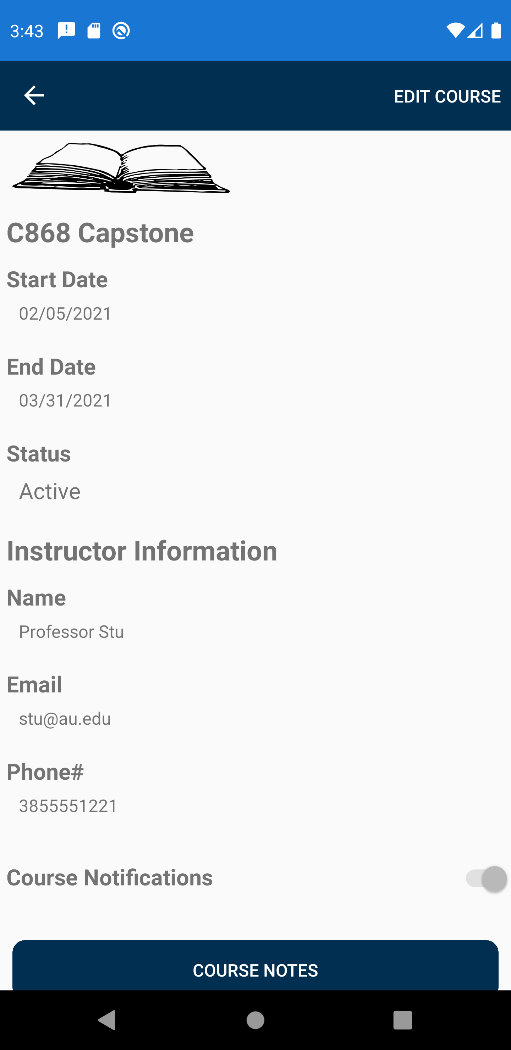
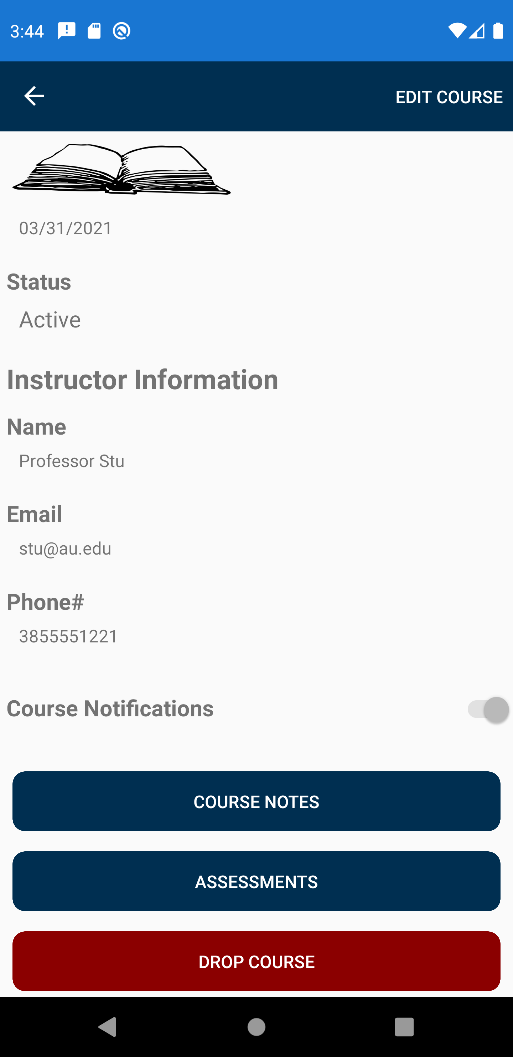


1. A share screen pops up, from here you can copy the note text or tap any app that has the functionality to share the note; for example, messages, email client, or another communication app.

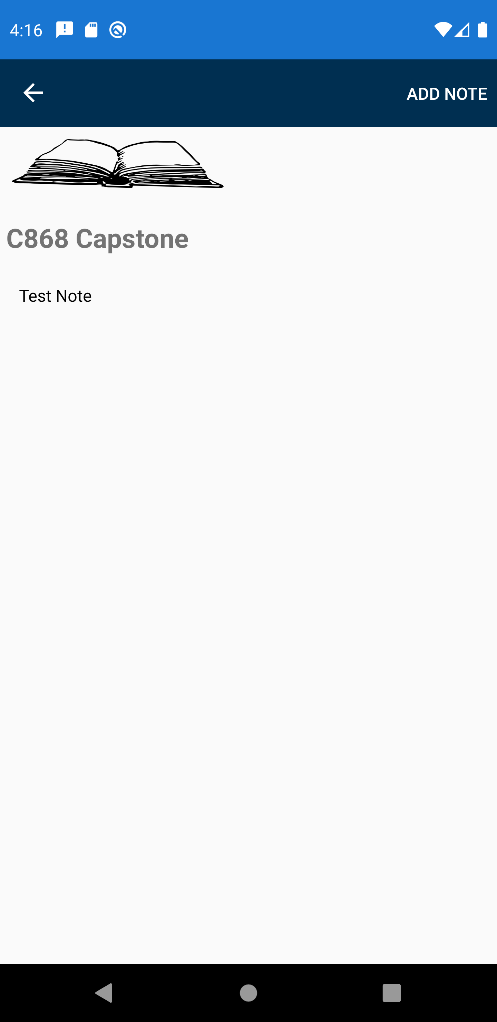
### Deleting a Course Note



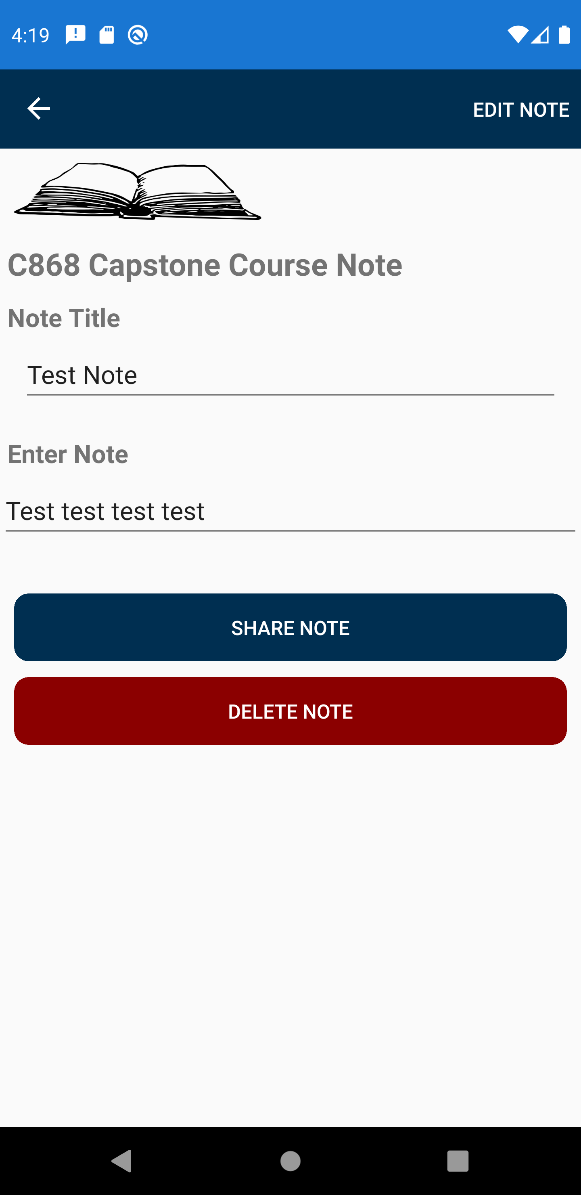
1. Tap on a Course from the course list on the Term page.

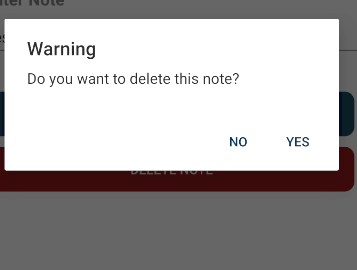
1. Scroll down to the bottom, Tap on “Course Notes”.



1. Tap on a Note in the note list on the Course Note page.



1. Tap on “Delete Note”.



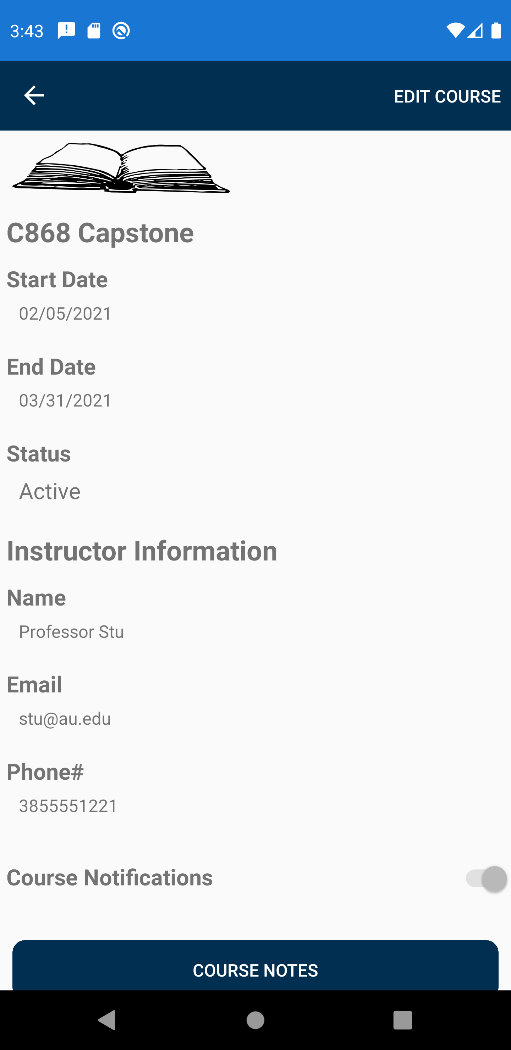
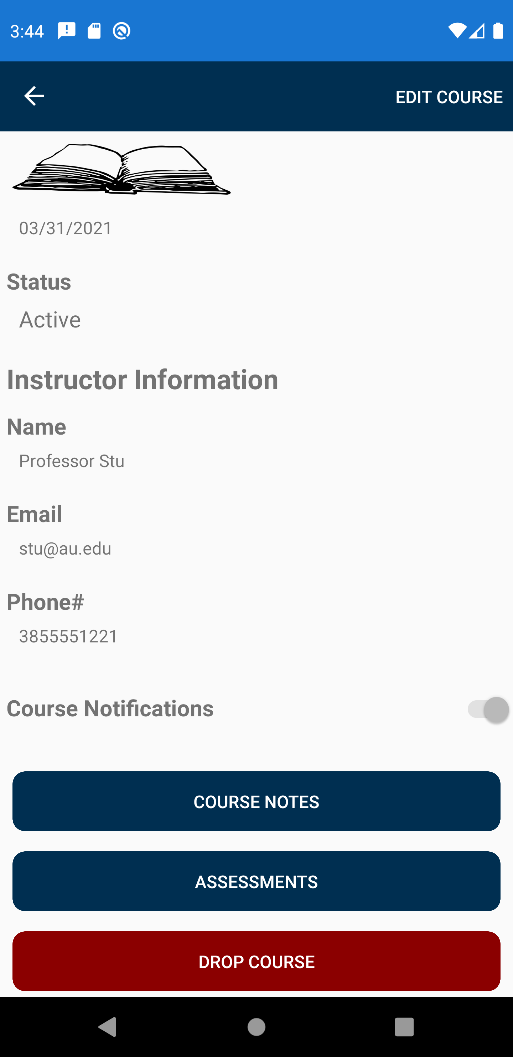
1. Tap “Yes” or “No”. If “Yes”, the course note will be deleted.

## Course Assessments

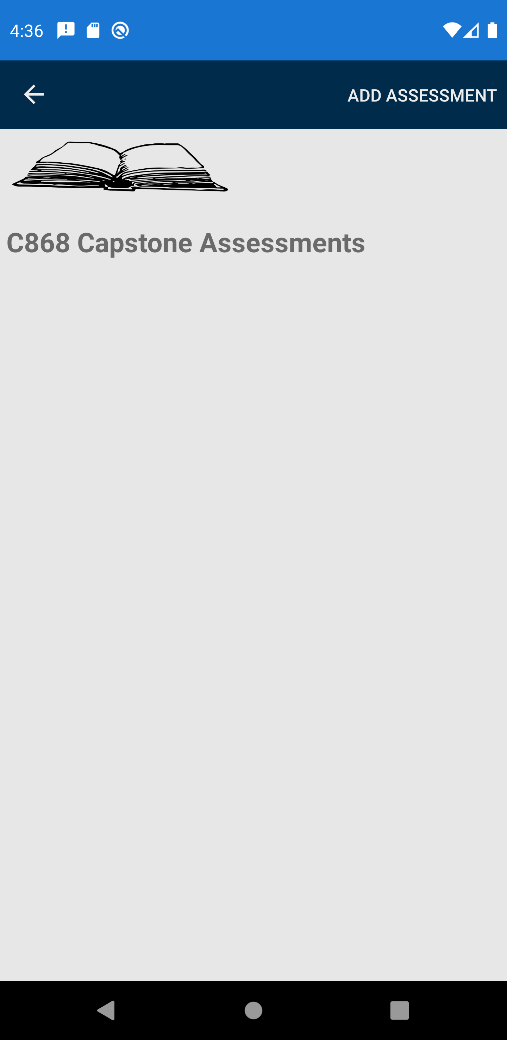
### Adding a Course Assessment



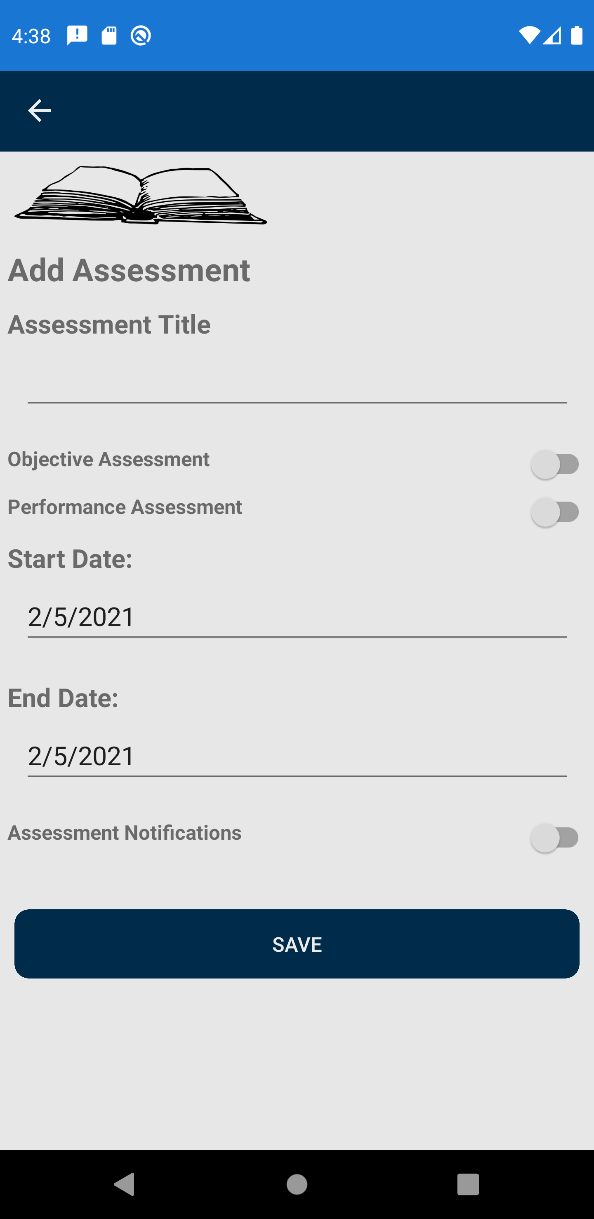
1. Tap on a Course from the course list on the Term page.

1. Scroll down to the bottom, Tap on “Assessments”.



1. Tap the “Add Assessment” tool bar item in the top right. 



1. Enter Assessment Title.
2. Tap Objective Assessment or Performance Assessment.

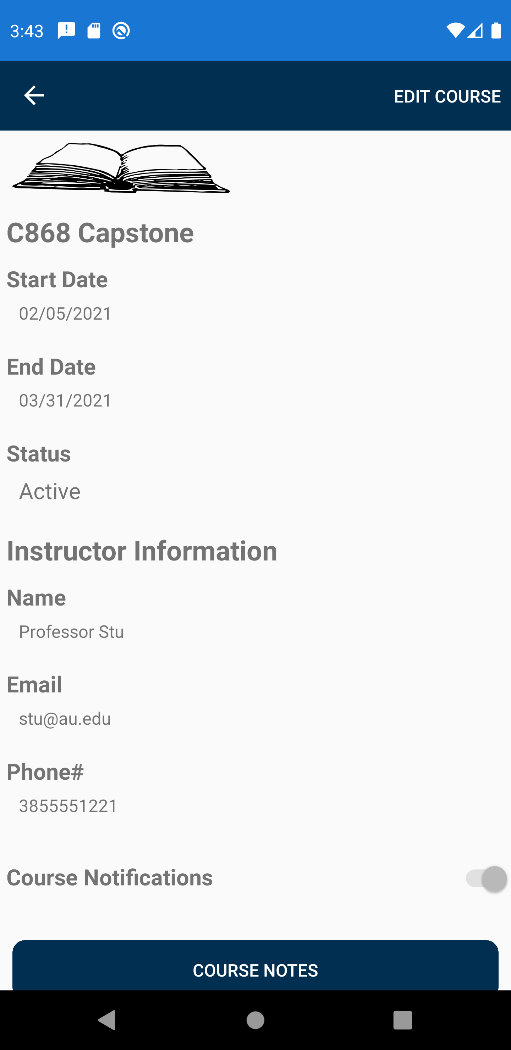
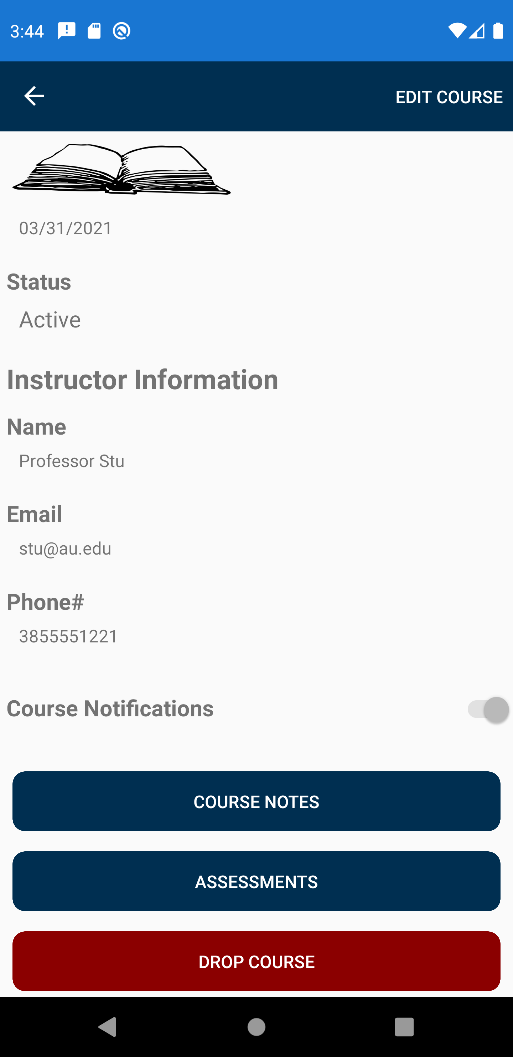
*Note: Courses can only have one of each type of Assessment.*

1. Select the Start and End Dates of the Assessment.
2. If a notification of the end and start date for this assessment is desired, toggle the Assessment Notifications switch.
3. Tap “Save”.

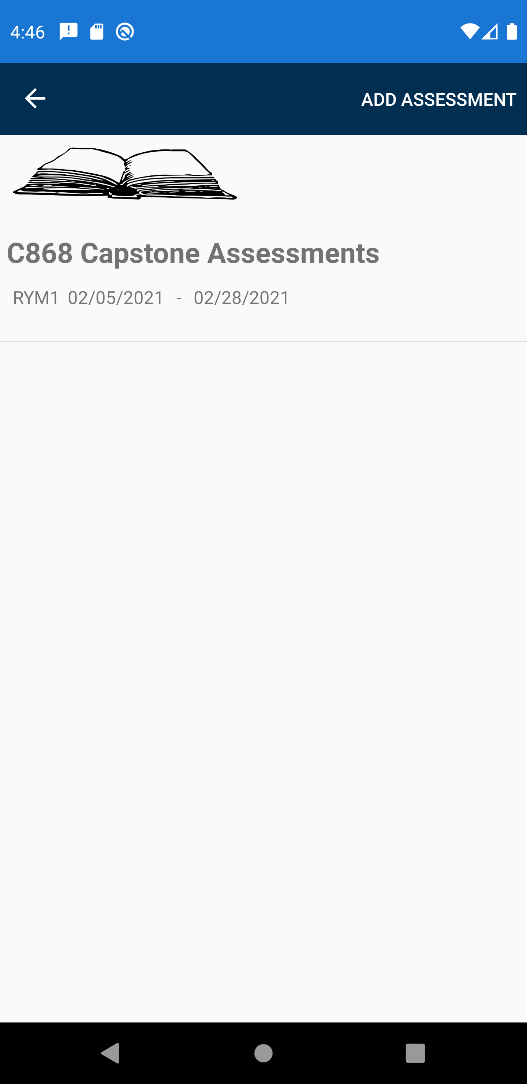
### Editing an Assessment



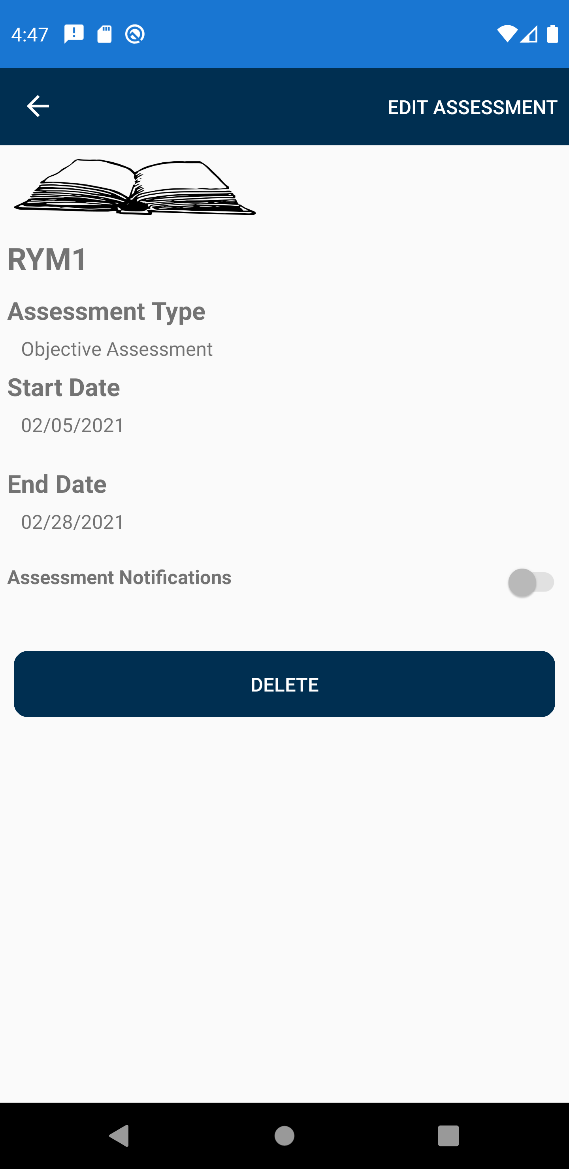
1. Tap on a Course from the course list on the Term page.

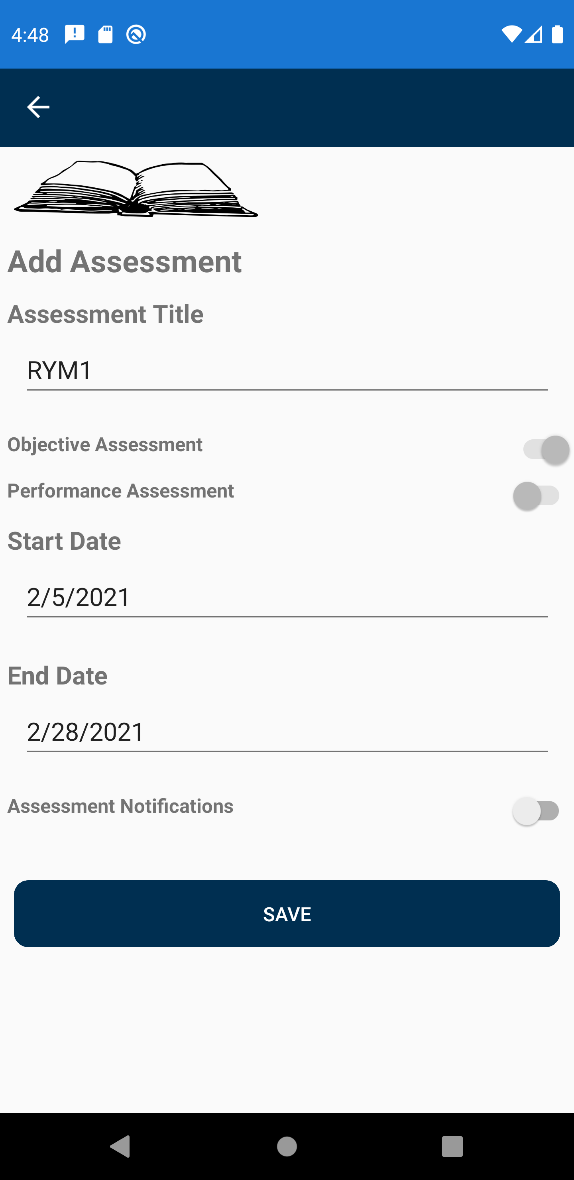
1. Scroll down to the bottom, Tap on “Assessments”.



1. Tap on an assessment in the assessment list.



1. Tap the “Edit Assessment” toolbar item in the top right. 



1. Make the desired changes.

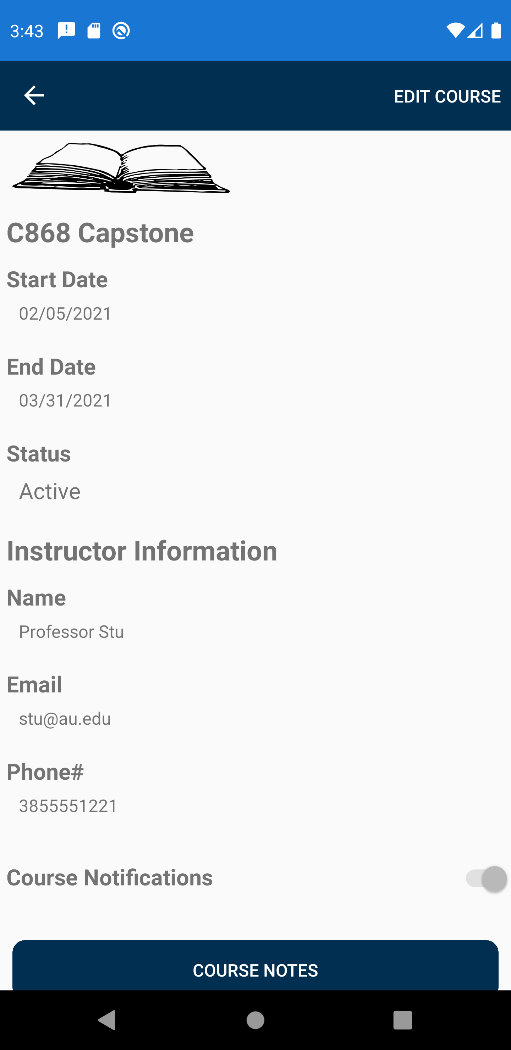
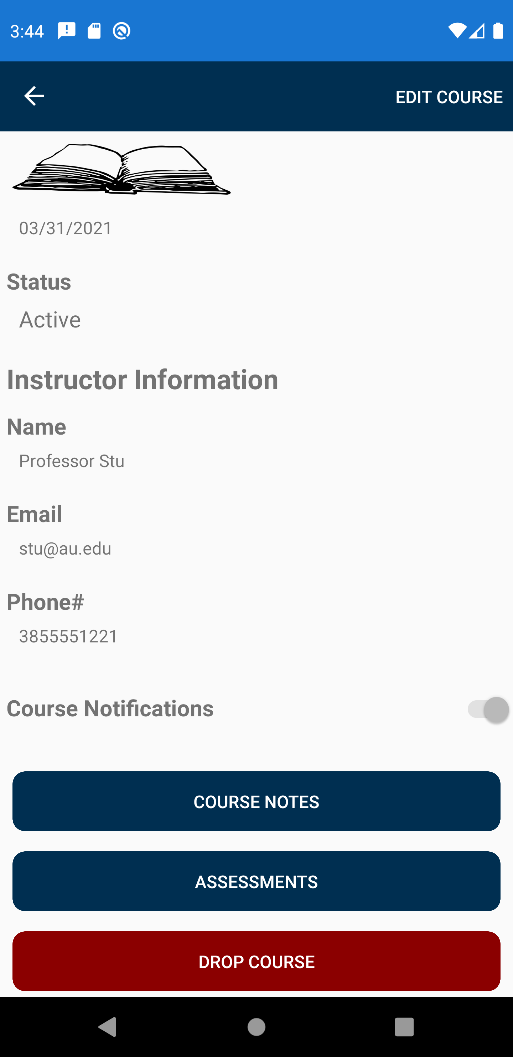
*Note: Cannot change Assessment type, if this change is desired, delete an existing assessment and recreate it.*

1. Tap “Save”.

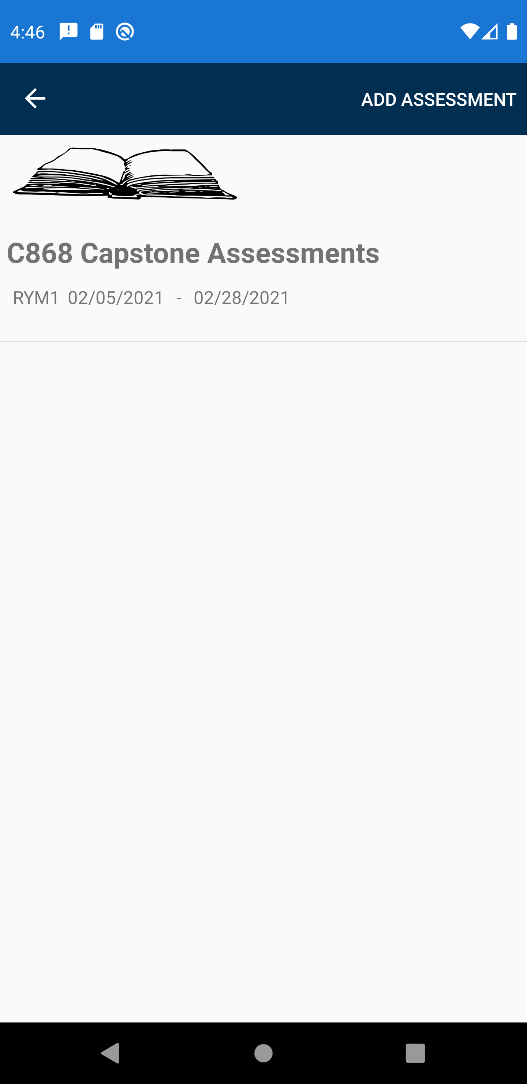
### Deleting an Assessment



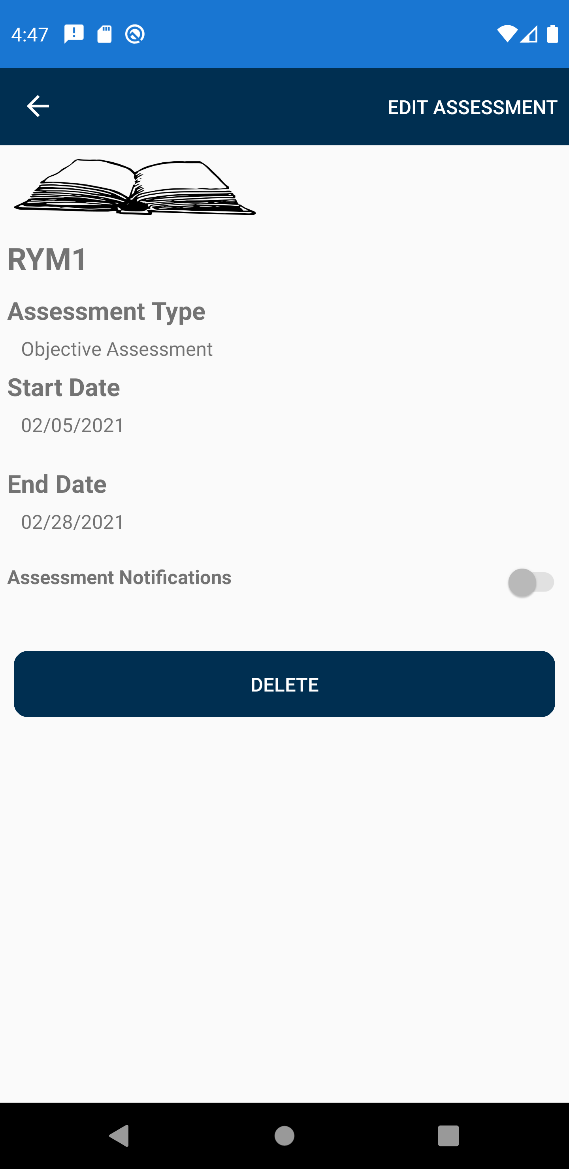
1. Tap on a Course from the course list on the Term page.

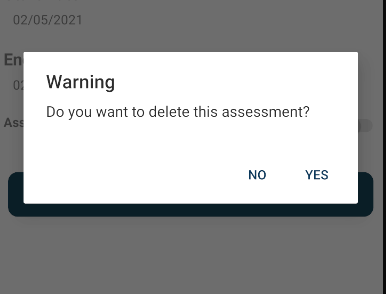
1. Scroll down to the bottom, Tap on “Assessments”.



1. Tap on an assessment in the assessment list.



1. Tap “Delete”.



1. Tap “Yes” or “No”. Deleting an assessment will only delete that assessment.

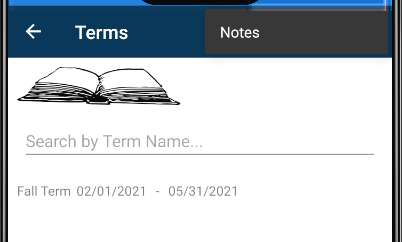
## Personal Notes / Course Notes

Personal Notes are a different type of note that can be added to each term. Through the same menu, a Course Note can also be viewed, edited, or added.

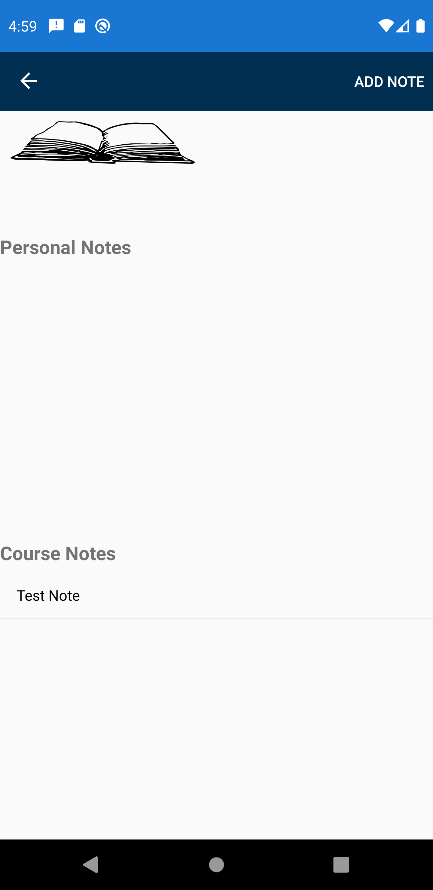
### Create a Personal Note / Course Note



1. From the Terms page tap the 3 dots in the tool bar on the top right. 

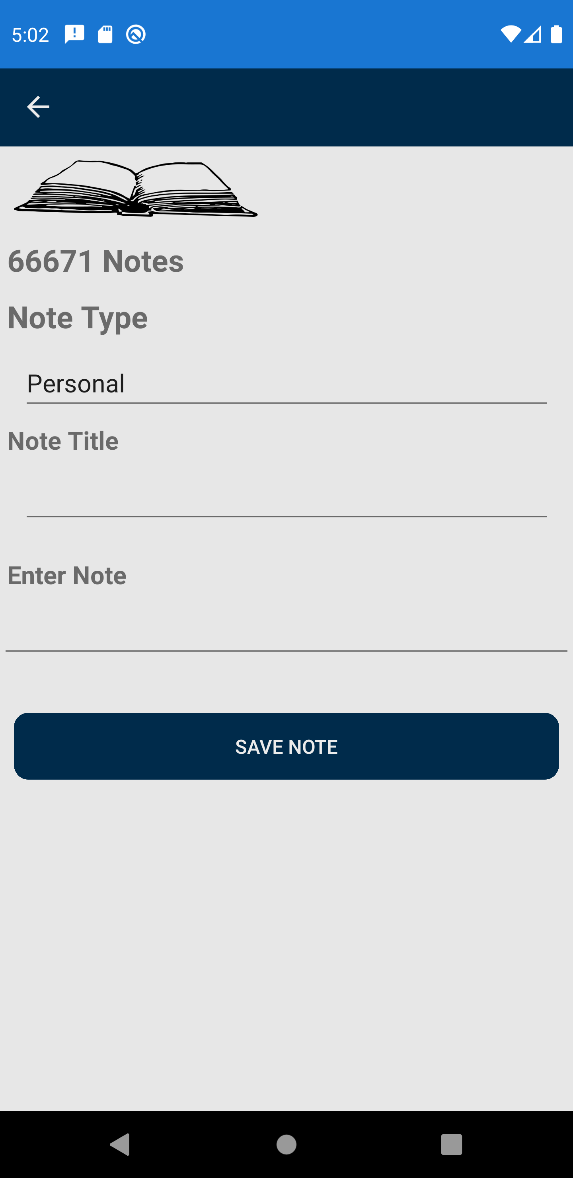


1. From the secondary menu, tap “Notes”.

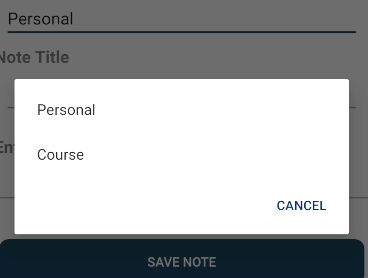


*Note: From this page you can view any Course Notes or Personal Notes that you have created, with the exception of Course Notes that were associated with a course that has been dropped.*

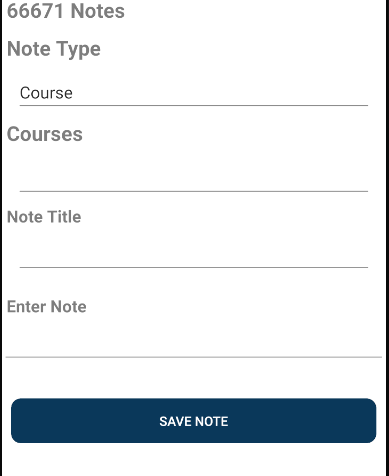
1. To add a Note, Tap the “Add Note” tool bar item in the right. 



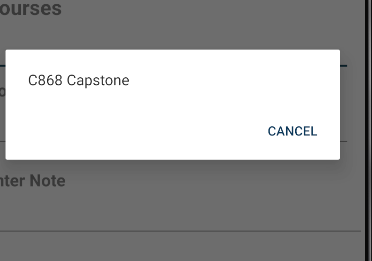
1. Choose Note Type. There are two types, Personal and Course. Notes default to Personal.



1. If Personal is chosen, nothing else further needs to be done.
2. If Course was chosen, a Course will need to be selected.



*Note: A “Courses” Picker will appear, and a Course will need to be selected. There will need to be at least one course created to save a course note.*

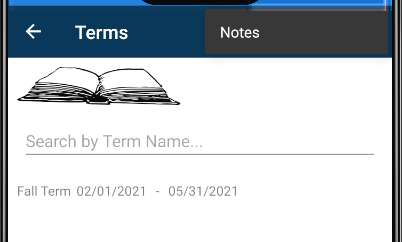


1. Type a note title and enter the note text into the Enter Note textbox.
2. Tap “Save Note”.

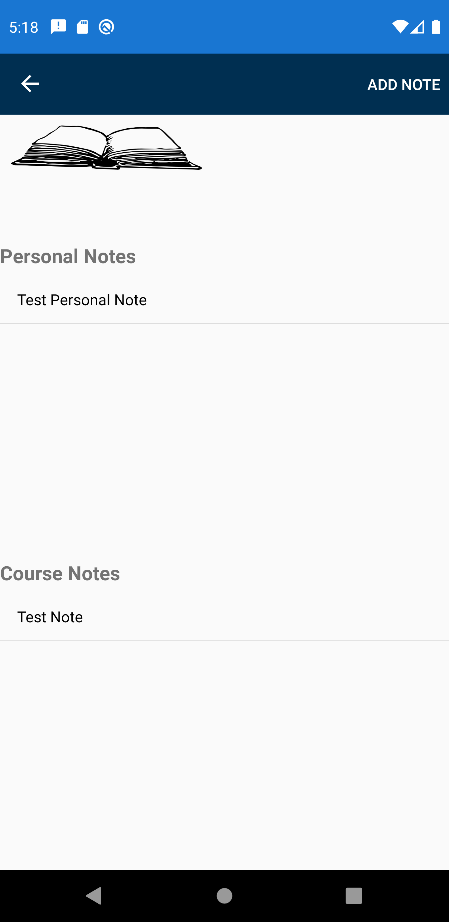
### Editing Personal / Course Note



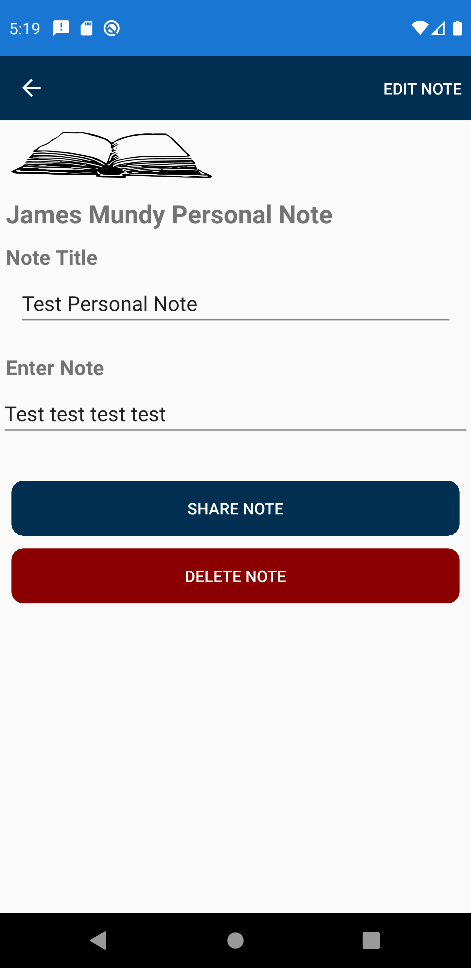
1. From the Terms page tap the 3 dots in the tool bar on the top right. 



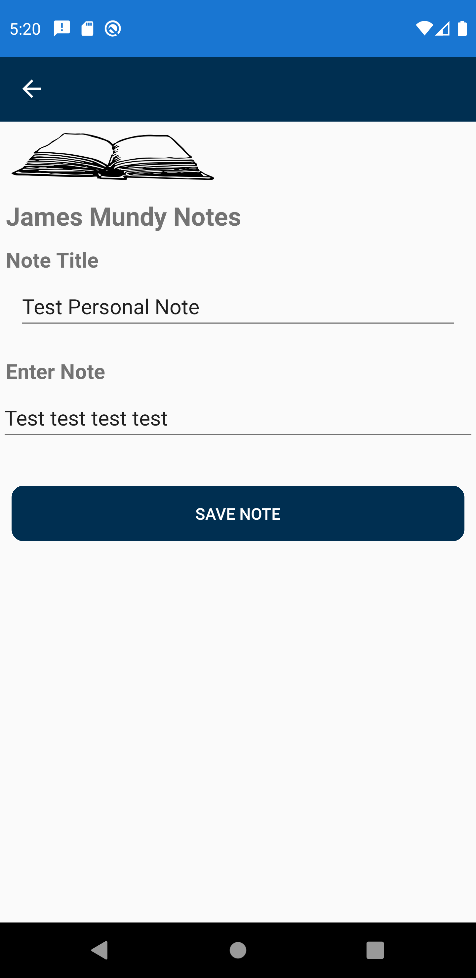
1. From the secondary menu, tap “Notes”.



1. Tap a note in the Personal Note or Course Note list.



1. Tap the “Edit Note” tool bar item in the top right. 

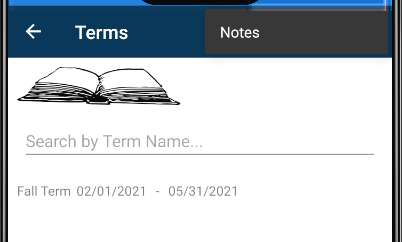


1. Make the desired changes.
2. Tap “Save Note”.

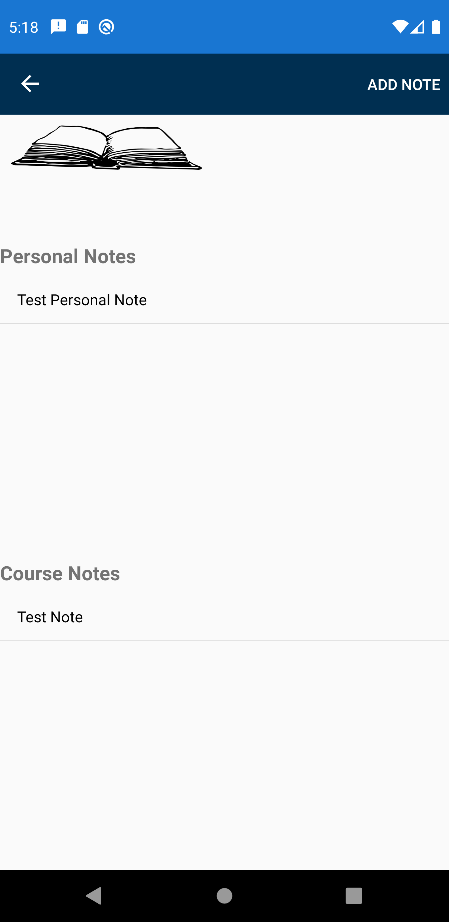
### Sharing a Personal Note / Course Note



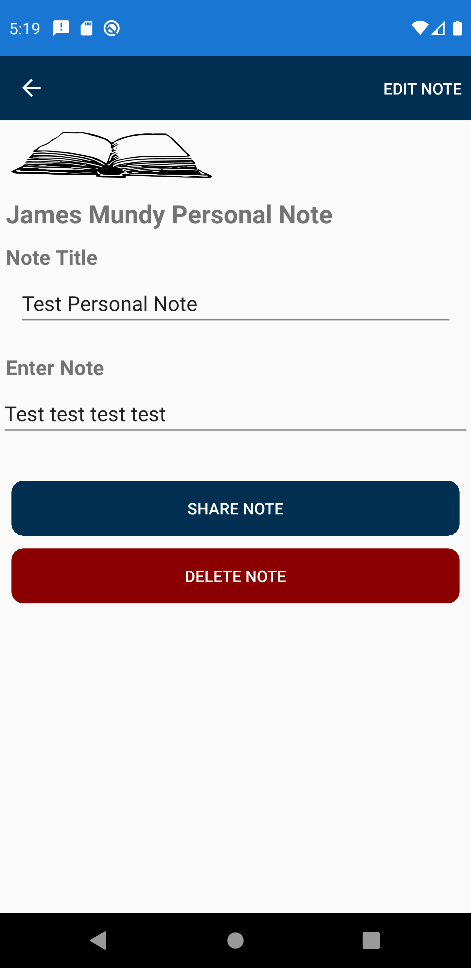
1. From the Terms page tap the 3 dots in the tool bar on the top right. 



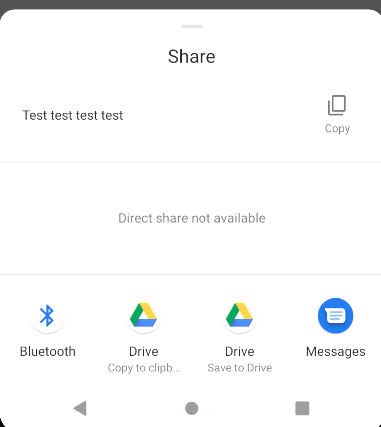
1. From the secondary menu, tap “Notes”.



1. Tap a note in the Personal Note or Course Note list.



1. Tap “Share Note”.

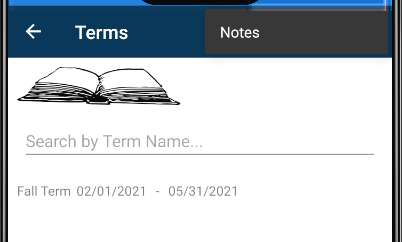


1. A share screen pops up; from here you can copy the note text or tap any app that has the functionality to share the note. For example, messages, email client, or other communication app.

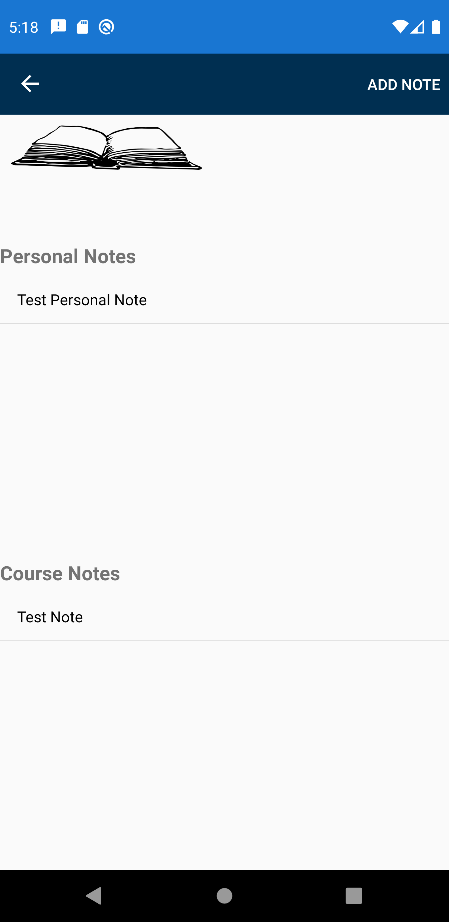
### Deleting a Personal / Course Note



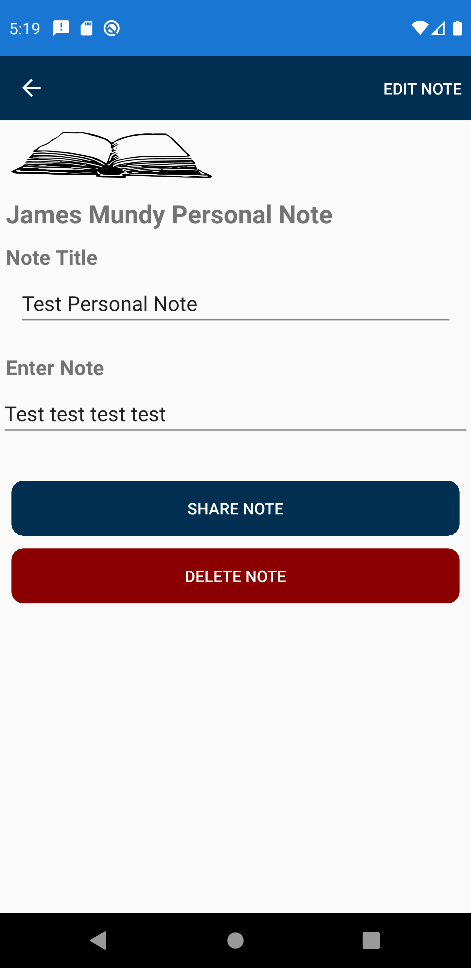
1. From the Terms page tap the 3 dots in the tool bar on the top right. 



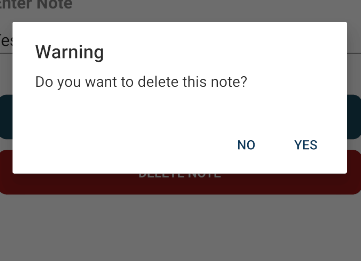
1. From the secondary menu, tap “Notes”.



1. Tap a note in the Personal Note or Course Note list.



1. Tap “Delete Note”.



1. Tap “Yes” or “No”. Deleting a note only impacts that note.

## Term Report

View a report pertaining to the current term. Contains time left in the current term and how many courses are active, inactive, or completed.

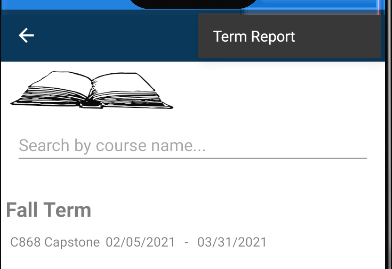
### View Term Report



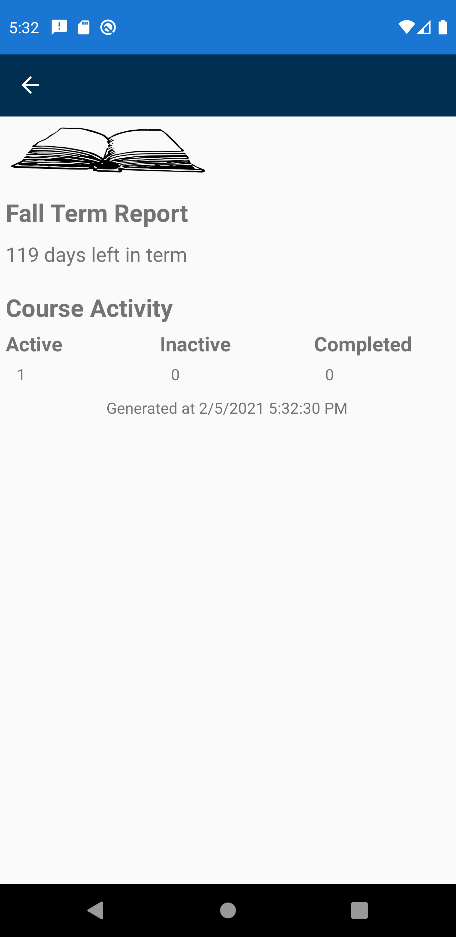
1. Tap on a Term from the Term List.



1. Tap on the 3 dots on the tool bar in the top right.



1. From the secondary menu, tap on “Term Report”.



1. The term report lists how many days are left in this term, how many courses are active, inactive, or completed.