ITP 342 Homework 5

Goal

- You will update your **Flashcards** iPhone app.
- It will have a tab bar for two views – one to display a flashcard and one for the table of flashcards.
- The tab bar item that displays a table of flashcards will allow the user to add and delete cards.
- Add data persistence to the model by saving a plist file to your app's Documents folder. Do not use User Defaults.

Assignment

- Duplicate your high-level Flashcards (or Homework4) folder. Rename the copy to something like Flashcards_HW5.
- Update the View Controller (the one that displays a single card):
  - Embed it into a Tab Bar Controller.
- Create the Table View Controller in Storyboard:
  - Use the library to create a Table View Controller Scene.
  - Embed in a Navigation Controller.
  - Add a bar button item for Add.
  - For the Table View Cell, update the style and the reuse identifier.
- Create the class (Cocoa Touch or Swift) for the Table View Controller:
  - Enable the Edit navigation bar item and have it display on the left.
  - Create a private property for the model and use its singleton method.
  - Update the data source and delegate methods.
- Create a View Controller Scene in Storyboard to add a flashcard:
  - Add a navigation bar on top and add bar button items for Cancel and Save.
  - Add a label to tell the user what to enter.
  - Add a text view to input a question.
  - Add a text field to input an answer.
- Create the class (Cocoa Touch or Swift) for the Add View Controller:
  - Make it adhere to the UITextViewDelegate and UITextFieldDelegate protocols.
  - Implement the UITextViewDelegate and UITextFieldDelegate methods to enable/disable the Save button. It should be enabled when there is text in the text view and the text field.
  - Create a typedefBlock with no return type, and two arguments – NSString objects for the text of the Text View and Text Field.
  - Create a public property of the typedefBlock.
  - Create a public property to set the text of the label telling the user what to enter.
  - Create a public property to set the placeholder of the text field.

  - Create the appropriate IBOutlets.
  - Create the IBActions for the Cancel and Save buttons.
  - Implement the textFieldShouldReturn: method OR the touchesBegan:withEvent: method to have the keyboard dismiss properly.
- Update the Add View Controller in Storyboard:
  - Update the Class in the Identity Inspector.
  - Make it a delegate for the Text View and Text Field.
- Update the Table View Controller in Storyboard:
  - Update the Class in the Identity Inspector.
  - Create a private property for the model and use its singleton method.
  - Update the data source and delegate methods.
  - For the Table View Cell, update the reuse identifier.
  - Create a Present Modally segue from the Add button to the Add View Controller.
- Update the Table View Controller class:
  - Implement the prepareForSegue: method. In it, set the set the public properties.
- Add data persistence:
  - Update the model to add data persistence by saving a plist file to your app's Documents folder.
  - Verify that it works through the user interface.
  - Also verify that the file has been create/updated in the correct folder.
    In the init method in your file, log the filepath in the console. Using Finder, find your app folder that was printed out in console window using the NSLog statement above. In Finder, you can press the Shift-Command-G keys and enter the path in the text field. Look in the Documents folder. You should see a plist file.
- Handle deleting all cards:
  - Update the model to handle the deletion of all cards.
  - Update the class for the Question tab to handle no cards. Display an appropriate message such as "There are no more flashcards." for a single tap and "Please add more flashcards." for a double tap.

Submission

- Compress your project folder.
- Rename the zip to *Lastname**Firstname*HW5.zip where *Lastname* is your last name and *Firstname* is your first name.
- Submit the .zip file on Blackboard under Assignments.

Grading (60 pts)

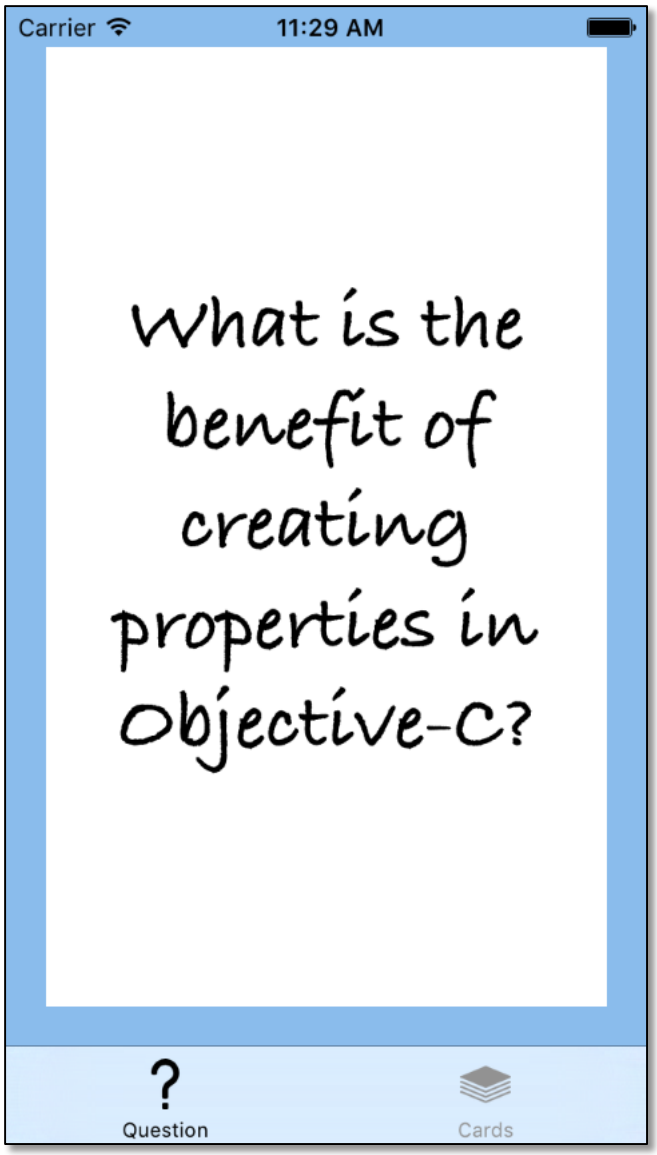
- Tab bar: 5 pts
- Table of flashcards: 15 pts
  - 5 for table shows questions
  - 5 for add button on right and edit button on left
  - 5 for delete working properly
- Add new flashcard: 10 pts
  - 5 for enable/disable Save button (text in text view and text field)
  - 5 for completion handler
- Delete all cards and handle on Question tab: 5 pts
- Auto Layout (can be portrait only; iPhone SE, 7, and 7 Plus): 5 pts
- App icon image and tab bar icons in Assets: 5 pts
- Data persistence: 15 pts
  - If a plist file exists in the Documents directory, then the model loads the data. If not, model creates at least 5 flashcards: 10 pts
  - Save model when add and delete a flashcard: 5 pts
- Do not share your code with other students. Do not accept code from other students. If you need help, ask the lab assistants and instructor.

Important Notes

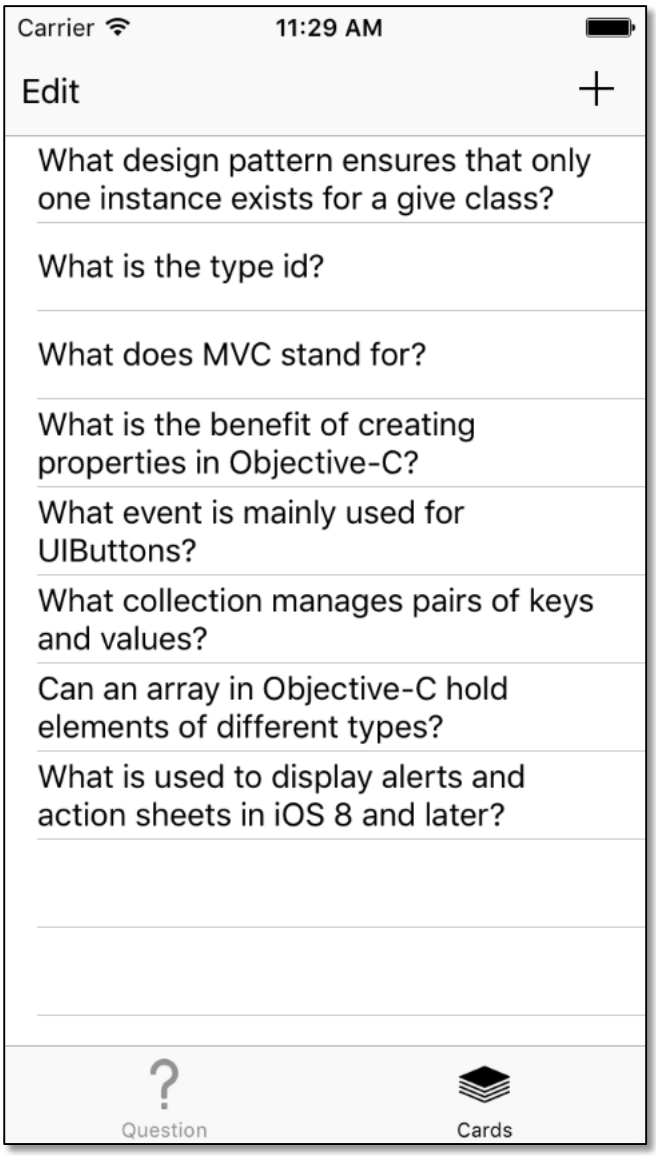
- If you are writing your project in Swift, please use Swift native types, like String instead of NSString, and Array instead of NSArray etc.
- Challenges are not graded, but are strongly recommended!
- Please do not submit a project including a challenge if it does not work or causes other bugs in your project that are graded.
- Code quality should be good with helper functions as necessary, no legitimate compiler warnings, and no blocks of repeated copy-pasted code or chunks of unused commented code. Consistent style and indentation is also a good practice to get in the habit of. Points will be deducted for these issues.
  - If your code is clear and easy to follow, it's easier for us to give more partial credit if something were to go wrong, so it's in your best interest to make your code readable.
- Your name should still appear in the top of each file.
- You cannot reuse the lecture week's project or sample project for your homework.
- We reserve the right to deduct points if your implementation is unsatisfactory.
- Please follow the spirit of the project.
- If you would like to deviate slightly from these guidelines, please check with us before doing so, even if your way requires more work. We can sometimes make exceptions but want to ensure you're able to receive points for the items we are looking to assess in this assignment.
- If you have a bug in your assignment that makes it difficult or impossible to test, we will not fix the bug to test your app and you will not receive points for features we cannot test.
- You are responsible for testing your app thoroughly on all supported devices before submission. We are happy to help work through issues and check against the rubric in office hours, but cannot be responsible for all of your testing prior to grading.
- If you have concerns about your grade, make a private post on Piazza or visit Office Hours. Regrade requests can result in a change in your grade either up or down, if we find something we missed in the previous grading.
- You have one week from when the assignment was graded to dispute the grade.
- Copying someone else's code is a violation of academic integrity policies and you will be reported. It is trivial for us to tell a project has been copied, in part or in full.

Example

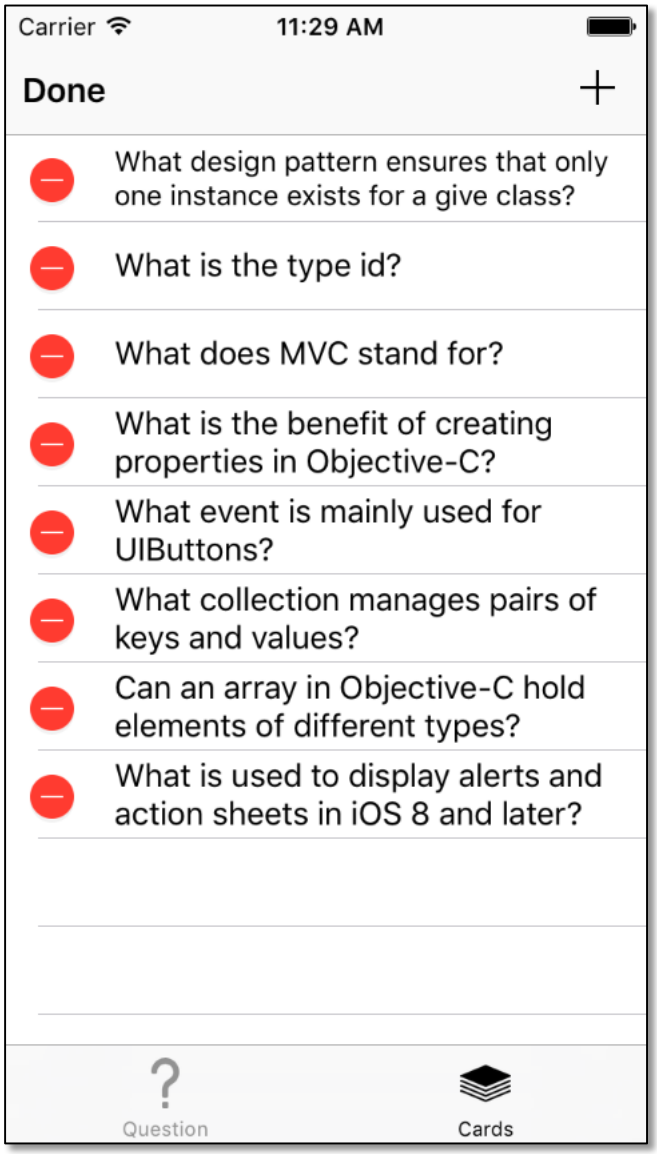
Question Tab



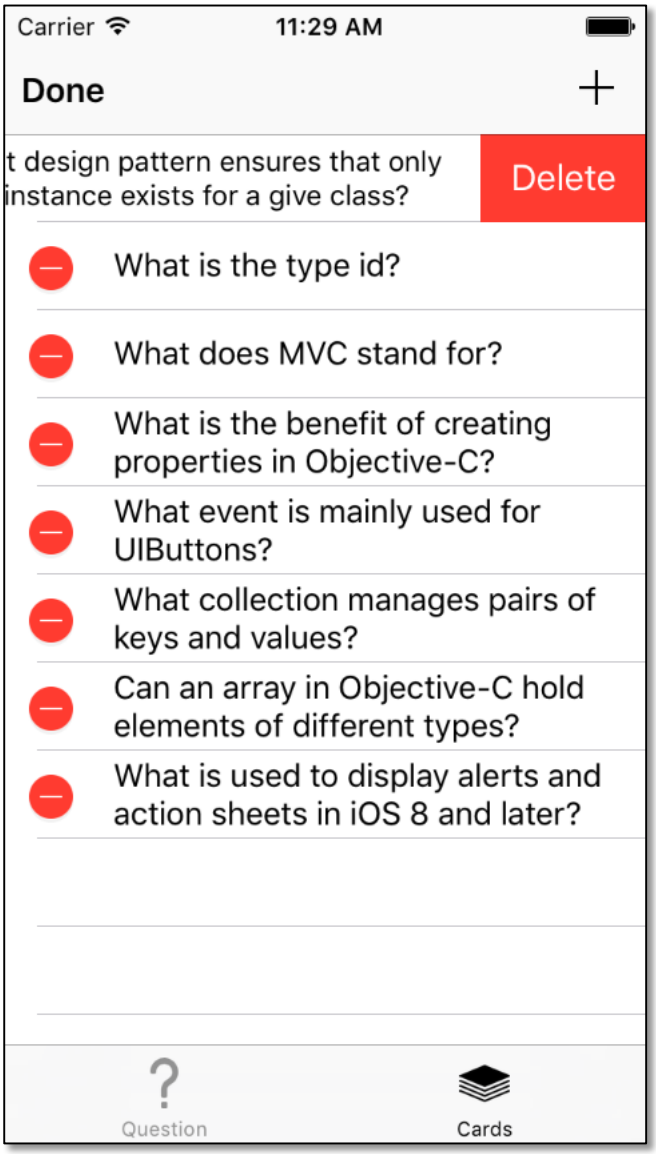
Cards Tab



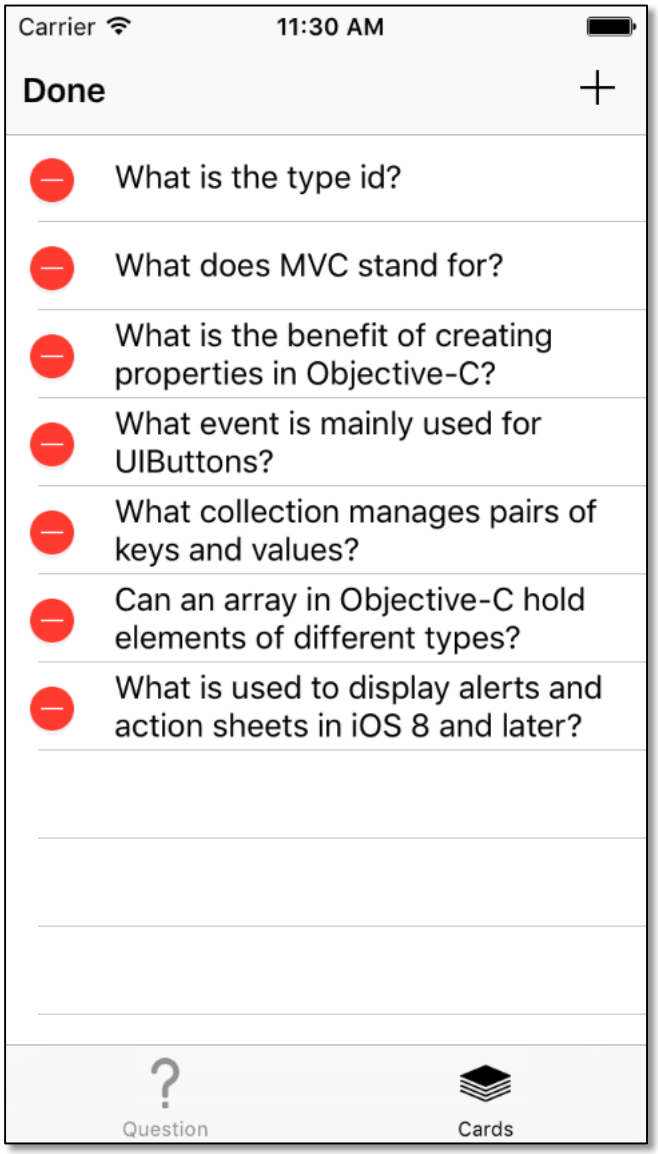
Edit touched



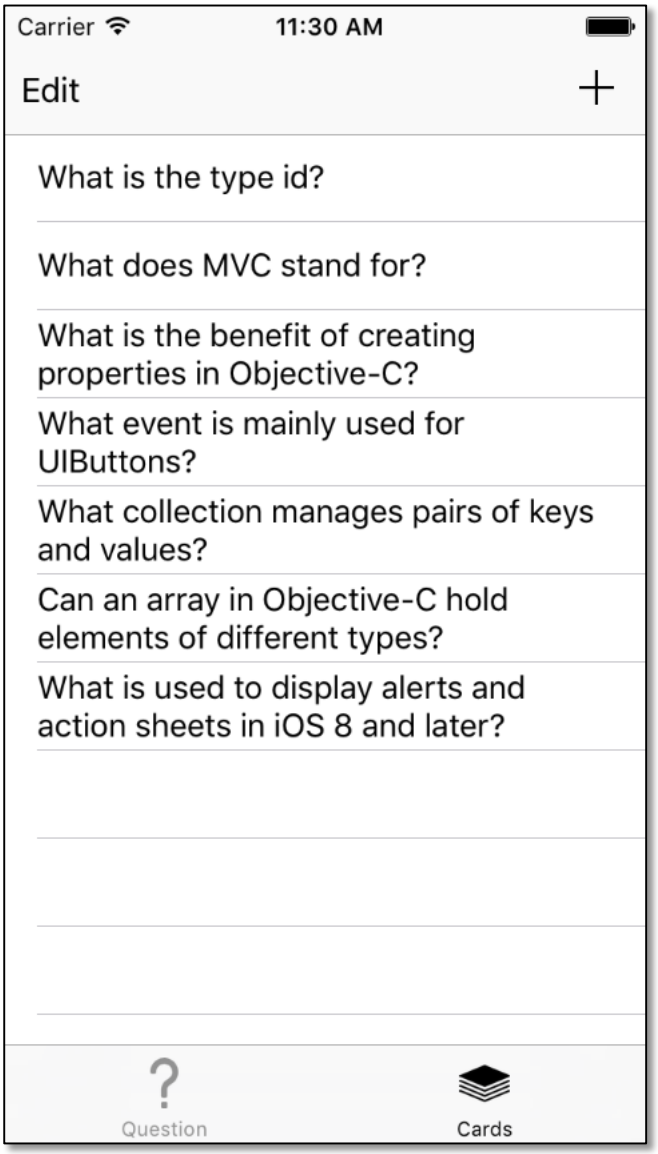
Minus icon touched for first element



Delete button touched

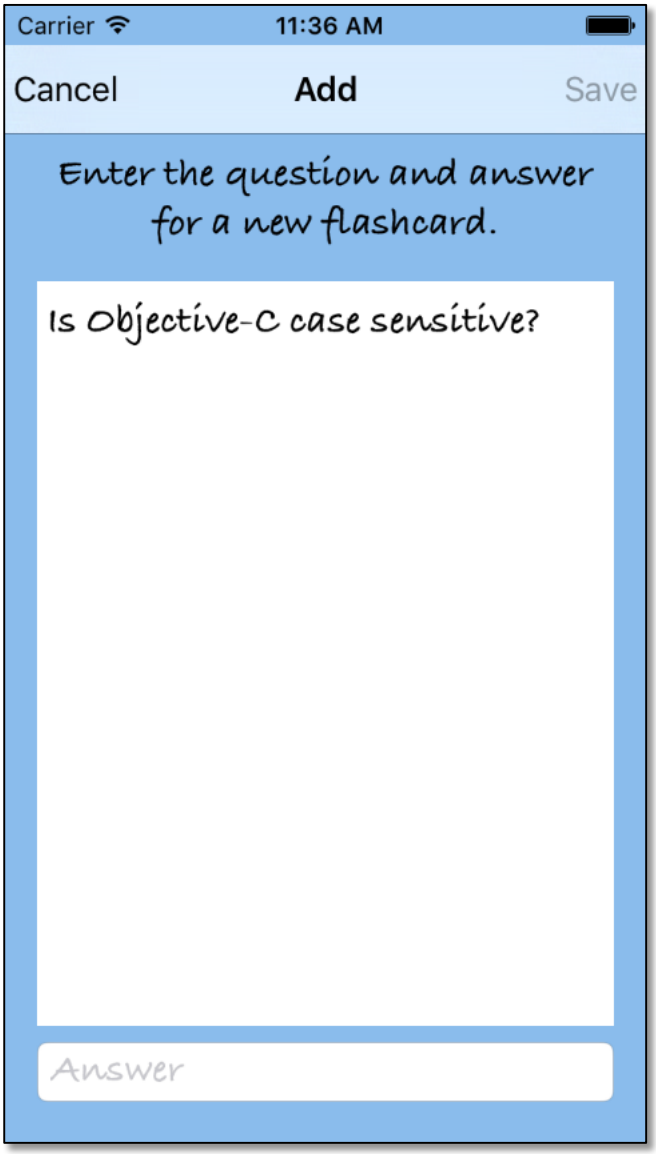


Model and Table is updated

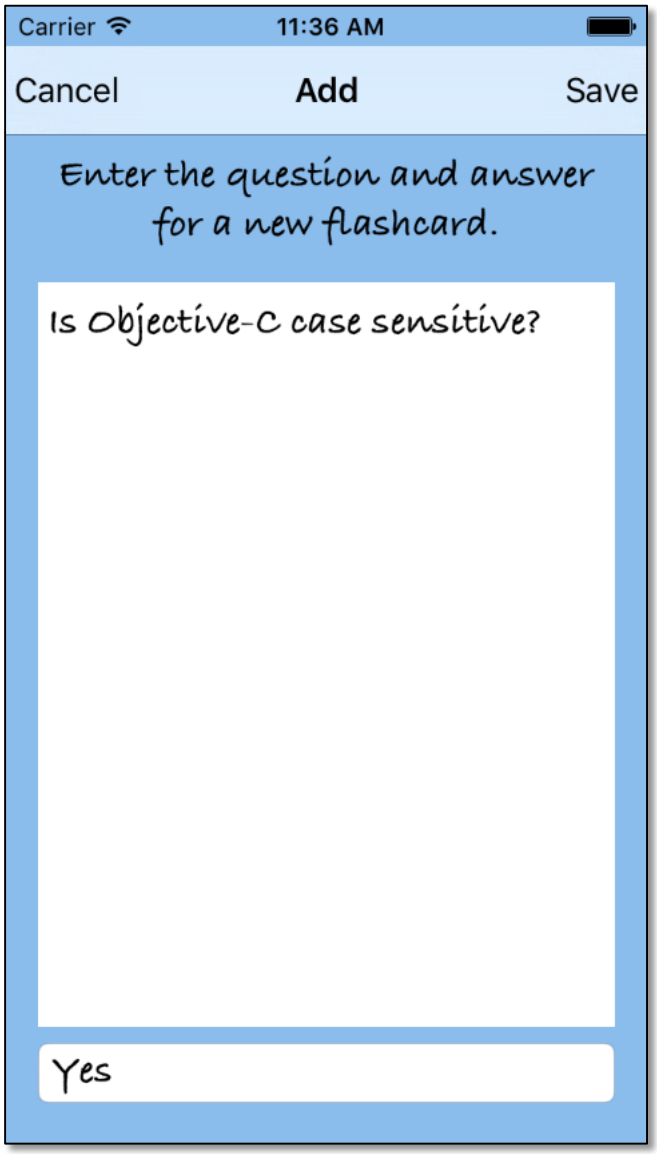


Add button (Plus icon) touched;
Save button is disabled;
Keyboard appears

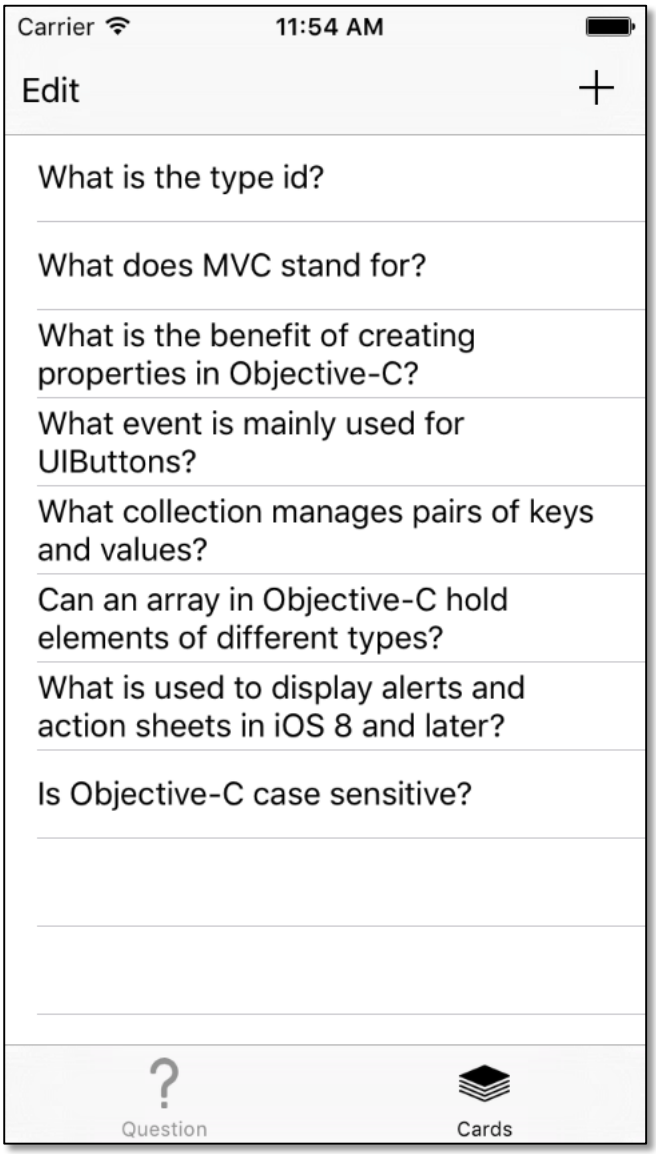
Add text in Text View;
Save button is still disabled;
Touch outside of Text View to dismiss keyboard



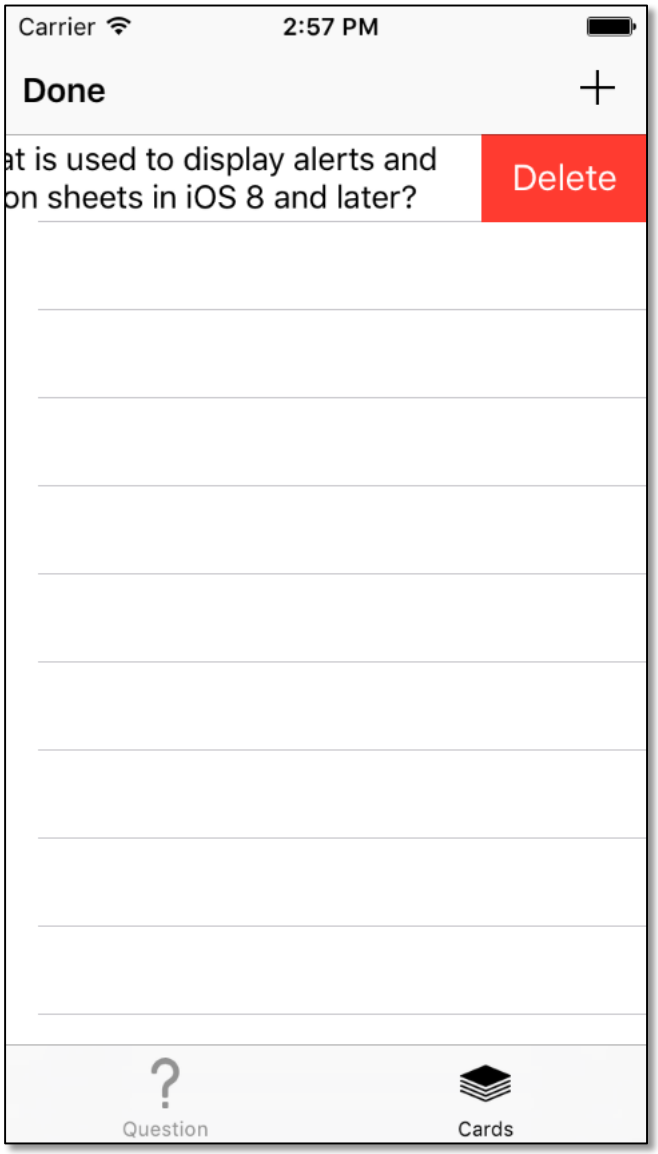
Add text in Text Field;
Save button is enabled;
Touch return/done button on keyboard or outside of Text Field and Text View to dismiss keyboard



Save button touched;
Model and Table is updated



Delete All Cards

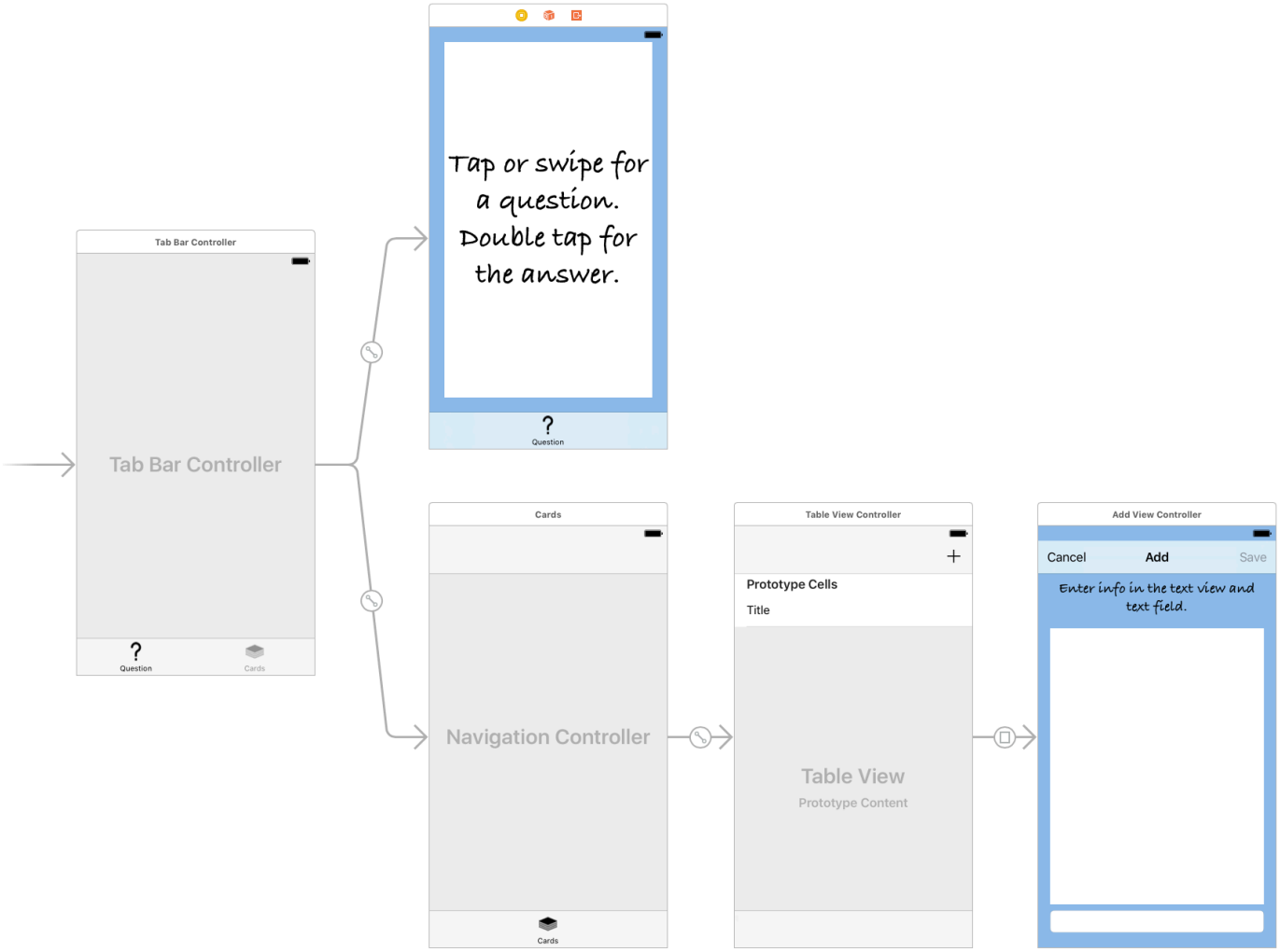


All cards deleted

Question Tab with Single Tap and no cards

Question Tab with Double Tap and no cards

Storyboard


Tap or swipe for a question. Double tap for the answer.
Question
Tab Bar Controller
Tab Bar Controller
Question
Cards
Cards
Navigation Controller
Cards
Table View Controller
Prototype Cells
Title
Table View
Prototype Content
Add View Controller
Cancel
Add
Save
Enter info in the text view and text field.