An iOS Application for Assessment of Handwritten Examinations

Department of Computer Science
Golisano College of Computing and Information Sciences
Rochester Institute of Technology
Rochester, NY 14623

***Abstract*—The iOS application proposed in the paper lets instructors grade handwritten examinations on their iOS devices. It lets users add annotations, text, or grades to the selected PDF Documents. If the user has selected multiple documents, the user has the ability to view them per page or per document. Automatically adding up the grades and writing them out to a table helps instructors avoid calculation errors and analyze performance on the test. The scanned copies of the examination also discourages the students from making changes to graded examinations. The application uses the PDFKit framework provided by Apple to manipulate the selected PDF document, and the UIDocumentViewerController class to access and update the documents stored in the iCloud drive. A group of participants were asked to grade sample examinations using the application and fill out a survey. Their response was used to improve the application further.**

***Index Terms*—Mobile application; Education technology; iOS**

I. Introduction and Background

Almost all instructors at universities use handwriting to grade student examinations. The traditional approach has a lot of drawbacks associated with it. The instructor has to carry all the examinations with them and also store the old examinations at least for some time if the tests are returned. The students have the ability to change the marks after the examination has been graded. Altering the grade for a question can reduce the readability of the examination. Sometimes an instructor may realize after grading a few examinations that a particular question may need to be curved or graded according to an updated rubric. In addition to that, manually adding up the marks for an examination increases the odds of introducing errors in the final grade.

All of the above mentioned problems can be solved by using a mobile application that can open scanned examinations and lets the user add annotations, text and grades. There are already a lot of mobile applications available that let users annotate and add text to PDFs. The Files[2] app by Apple is one such application that lets users work with different documents including PDFs. There are other applications available on the App Store like the Schoology[3] app, which is a suite of related education tools. The Files[2] app has some of the functionalities that we think would solve the problem mentioned above but its lacks the functionality to add grades, automatically calculate the grade for an examination and generate helpful data. On the other hand the Schoology [2] app is a comprehensive tool which allows the instructors to create a virtual classroom with exams, assignments and lessons. The students can create accounts and then connect with the instructors.

The paper builds on top of the work done by Mr. Roland Burbulis, who was a graduate student at the Rochester Institute of Technology. His application had the ability to add and delete annotations, text and grades, although it could not be used because the updated document could not be saved back to iCloud. The first objective of the paper is to add the functionality that would allow the user to save the updated document so that it can be ready for actual use. The second objective is to have a bunch of qualified individuals test the application and provide important feedback and use the feedback to improve the application further.

There are a lot of applications available on the Apple App Store that let users view and edit PDF documents. The Files[2] app from Apple is one such application that lets users open documents stored on their devices or on their iCloud drive. The application has other tools available for the users to edit the documents like an annotation tool. The user can also share the edited documents directly from the app.

The Adobe Acrobat Reader[1] is an application that lets users create, view, annotate and share PDF documents. The user can access their documents from any cloud file providers like Dropbox, iCloud etc. The app also provides support for filling forms, scanning in documents etc.

Thus, the App Store does not have any applications available which fulfill the objectives proposed in this paper.

II. Architecture

The proposed iOS application lets instructors at universities scan paper based examinations and grade them on their iPads. Thus the first step in the process is to make the scanned examinations available on their devices.

Once the examinations have been scanned, the application needs to have the ability to access and display the documents stored in the iCloud drive. For this, we use a view controller which is a subclass of the UIDocumentBrowserViewController. This allows the user to select a document which is stored in their iCloud drive. The app is configured appropriately so that the user can only open documents which have the .pdf extension.

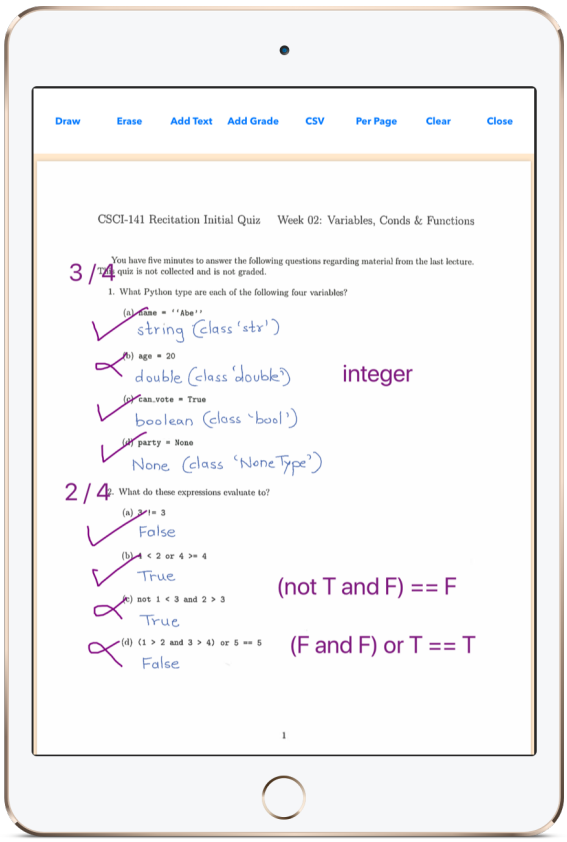

Fig. 1. The Application Interface

The application has a custom model class called the EZPDFDocument class, which is a subclass of the UIDocument class. By subclassing the UIDocument class, and implementing the two methods loadFromContents:ofType:error: and contentsForType:error:, we can save the updated document to the iCloud drive. The custom subclass has a property of type PDFDocument which will point to the PDFDocument instance created with the URL obtained from the UIDocumentBrowserViewController subclass.

The PDFKit framework provided by Apple is used for manipulating the PDF document displayed to the user. The PDFKit framework allows the user to add annotations, text and grades to the documents. It also provides a PDFView subclass which can be used to display the selected PDF document by setting the the document property on the instance. The second view controller includes a PDFView and a series of buttons at the top that allows the user to add annotations, grades or other remarks.

III. Method

We tested the application by asking nine people to grade a dummy examination on an actual device and fill out a survey. Most of the participants had experience grading handwritten examinations and homework. The dummy examination had a question of almost every type and was already graded by an instructor. The survey had 15 questions related to user interface, robustness and ease of use of provided tools.

- Did the application ever crash?
- Select all the operations which may have caused the crash
- Rate the ease of use of all the provided tools
- How important was the perPage tool?
- Does the application lack any features?
- Keeping in mind the constraints of grading on a tablet vs grading on paper, do you like the overall look and feel of the application?
- Do you think we should add a tutorial so that users can understand how the app works quicker?
- Would you use the app to grade examinations in the future?
- Do you have any additional comments about the working of the application?
- Rate the overall experience

IV. Results

Out of the 15 questions on the survey, eight offered five precoded responses. The answers to these questions have been converted to a bar graph by taking the average of the values for all participants.

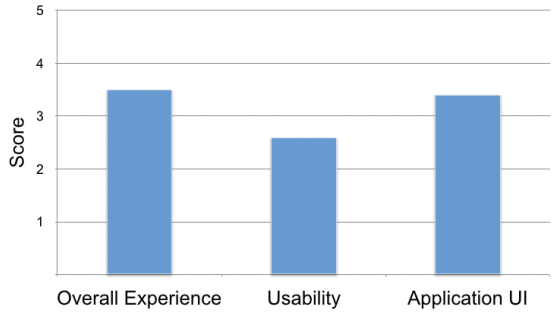

Fig. 2. Average Scores

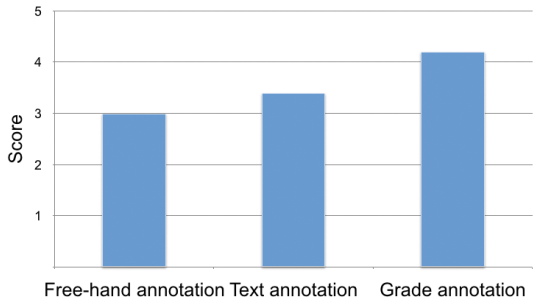

Fig. 3. Average Scores for Tools

It is clear from the above graph that the grade annotation tool is much more easier to use than the free-hand annotation tool. Some of the participants reported that it was too cumbersome to use fingers to add free-hand annotations.

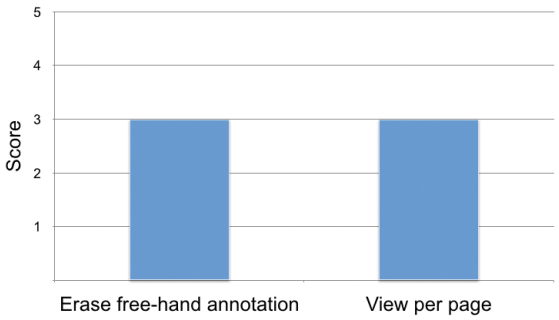

Fig. 4. Average Scores

The app is quite robust at this stage, although it crashes if the user zooms in on a spot in the document. One of the things mentioned in the survey a lot is how the app does not allow the user to scroll through the document while one of the annotation tools has been selected. So the user has to constantly switch between the different modes, which can be very inconvenient. In addition to that, if the user wants to move an annotation, they have to delete it and add a new one at the new spot.

V. Conclusion and Future Work

We have been able to resolve some of the most important problems which had been preventing instructors from using the application. The users can now save the updated document back to iCloud, and also add annotations to the whole document. The CSV file generated can be shared with others through email or saved to the Files folder on the device.

The survey has provided us with valuable insight into the things we need to improve in the application. The ability to scroll through the document while adding annotations will be very helpful. That can be implemented by adding a narrow scrollable area near the right edge of the screen. In addition allowing the user to move the added text and grade annotations would make the grading process a lot faster. As some of the survey participants felt that using their fingers to add annotations was very cumbersome, the app should have support for a stylus or Apple Pencil in the future.

Acknowledgment

The author would like to thank Dr. James Heliotis, professor in the computer science department at the Rochester Institute of Technology for the application idea and his support and guidance during the process, and Mr. Rolandas Burbulis for building the first version of the application.

References

[1] Adobe, Inc. (2018). Acrobat Reader (Version 18.11.01) [Mobile application software]. Retrieved from Apple App Store.
[2] Apple, Inc. (2018). Files (Version 12.0) [Mobile application software]. Retrieved from Apple App Store.
[3] Schoology, Inc. (2018). Schoology (Version 6.0.0) [Mobile application software]. Retrieved from Apple App Store.