FRAME & TELL

SAYEH GORJIFARD, RUNI GOSWAMI, ADENUGBE VORMAWOR

CS 165 - Spring 2014

INTRODUCTION

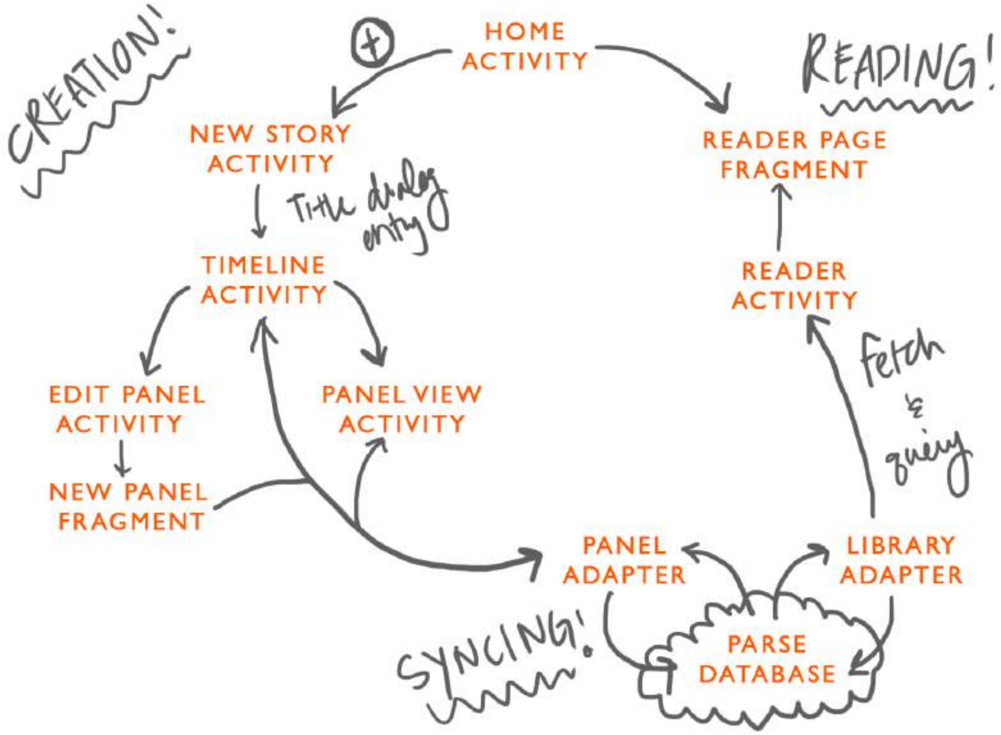
Frame & Tell is a mobile visual story creator and reader. It's also the easiest way for Android users to collaboratively create stories and comics. We felt the need for a collaborative comic creator and reader that allows users to

1. **Easily** create content
2. **Easily** view it
3. **Easily** share it with others without the need for manually keeping track of large image files.

All other current Android comic story creators and readers fail in one or more of those aspects, with cumbersome viewers and convoluted file sharing systems that require users to download reader-specific file readers.

ARCHITECTURAL DESIGN

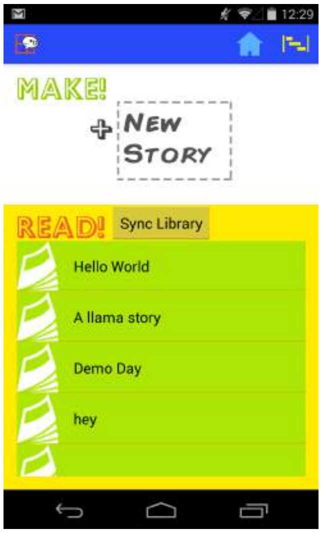
Design Diagram



Utilizes Parse.com for back-end development and cloud syncing.

Class Descriptions & UI

Home Activity – Launch pad for both the creator and reader functionalities. Allows users to either create a new story or sync to the cloud to read all existing stories in the database.

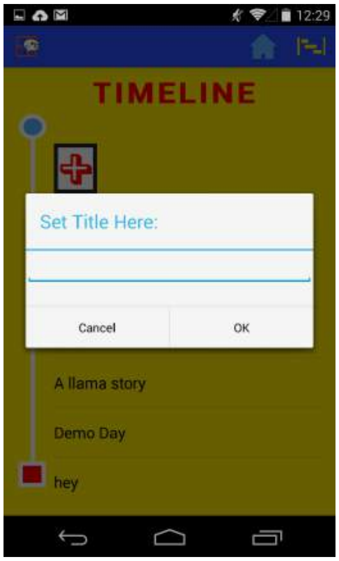


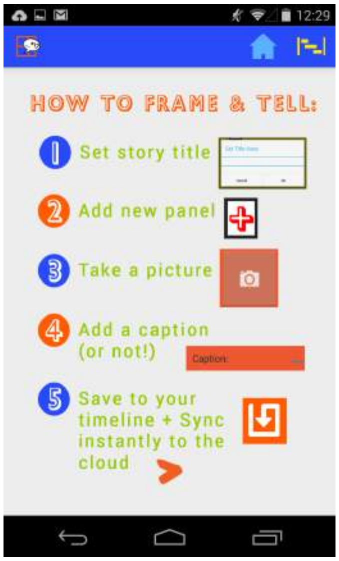
Panel – Panels are the basic objects of the app. Each panel has a title, caption, ParseFile (image file), object id and user id associated with it as well as getters and setters for each of those values.

Story – Multiple panels comprise a story object

Library – All the story objects comprise a library. Viewed on the Home Activity using a Parse Query and array adapter.

New Story Activity – Explains how to use the app, creates a new story object, and a launches a fragment to set its title.





HOW TO FRAME & TELL:

1 Set story title

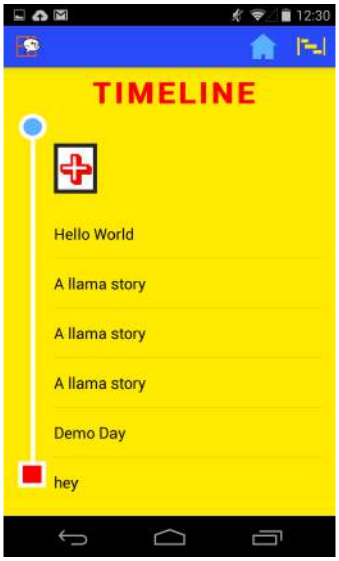
2 Add new panel

3 Take a picture

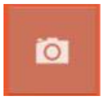
4 Add a caption (or not!)

5 Save to your timeline + Sync instantly to the cloud

Timeline Activity – Uses an array adapter and a Parse Query to display all the panels that have been created by all users.

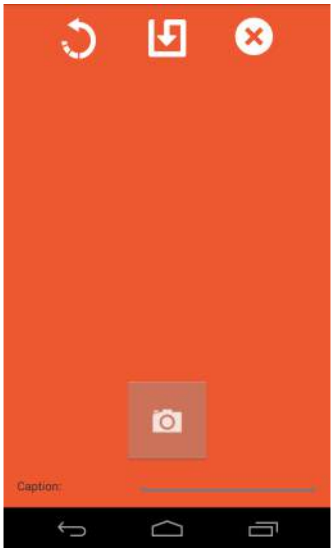


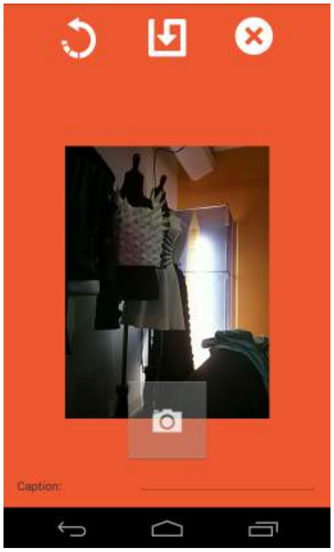
Camera Fragment – Simple fragment that directly uses Android's native Camera API to make an in-app camera rather than launching the camera service with an Intent. The shutter button:



takes the image data and saves a temporary local copy which is then saved to Parse in the Edit Panel Activity view.

New Panel Fragment - Activity that allows users to launch the camera fragment, set caption, and save a Panel to the cloud. Also allows users the choice to not add the Panel and return to the timeline if they don't like the image they have taken.





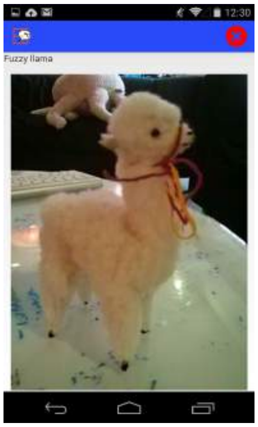
Edit Panel Activity – Container for the New Panel and Camera Fragments.

Frame and Tell Dialog Fragments – Dialog fragments for setting titles and caption.

Library Adapter – Array adapter using Parse Query to sort and visualize all created Panels by Story Title.

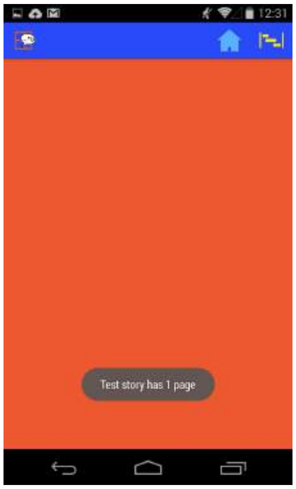
Panel Adapter – Array adapter using Parse Query to visualize all created Panels for the Timeline view.

Panel View Activity – Allows users to delete any existing panel object locally, from the cloud, and from all other users' phones.

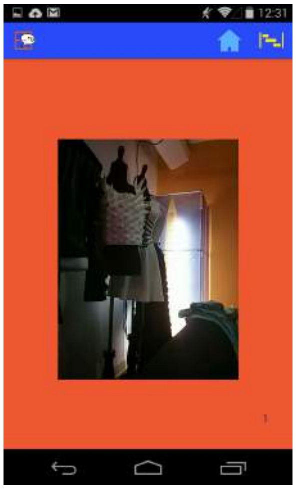


Reader Activity – Container for the Reader Page fragment.

Reader Page Fragment – Informs how many images have been added to a story, allows users to swipe through images in the story and view the captions associated as well as the "page number".



Test story has 1 page

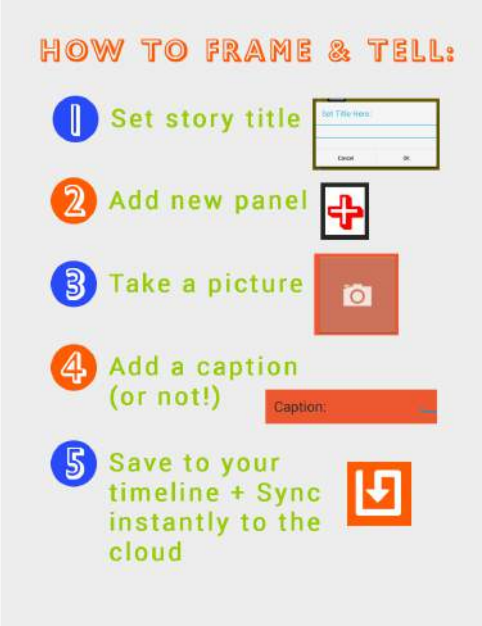


WHO DID WHAT

Sayeh – Most of the Parse back-end development and Panel creation activities.
Runi – UI/UX and some Parse back-end development.
Ade – Initial framework set-up, Reader activity, SVN and website management.

HOW TO RUN THE DEMO

1. Install the .apk from the project website or SVN
2. Log-in to Parse.com with username: runigoswami@gmail.com, password: parseL0gin and navigate to https://www.parse.com/apps/frame-tell/collections#class/Panel to view the datastore, analytics, and to view and delete objects on the server
3. Follow the How To instructions as they appear in the New Story activity and below:


HOW TO FRAME & TELL:

1. Set story title
2. Add new panel
3. Take a picture
4. Add a caption (or not!)
5. Save to your timeline + Sync instantly to the cloud

LESSONS LEARNT

We learned a great deal about collaborative coding, hacking to ensure maximum functionality, killing our darlings, redesigning on the fly, and Android development in the process of designing and creating a fully functional app in two weeks. We had hiccups right off the bat with clashing coding styles and visions for the app which only compounded when we independently started hacking away and created a Franken-app that would no longer merge on Git. Once we devised a system to collaborate and a very loose schedule for app development (another hiccup, we could probably use some of the scheduling apps other groups created), we were already one week in.

The code checkpoint halfway through really served as a turning point in Frame & Tell's development. We took Professor Campbell's suggestion to incorporate a back-end and image sharing to heart and went back to the drawing board, both figuratively and literally. After several hiccups with Google App Engine, we discovered Parse.com as an alternative platform for the back-end. This decision had its own pros and cons. On the one hand, getting data to the server was orders of magnitude easier than GCM had been. On the other hand, we had many fewer resources for help with Parse. Including "Parse" as a keyword in search leads to many red herrings as you can imagine. Moreover, the documentation for Android development in Parse is rather sparse, the number of Stack Overflow articles for it is sparser still, and the TAs, naturally, were as in-the-dark about it as we were.

Initially we were saving a local copy of all the ParseFiles by enabling local data syncing. Unfortunately we discovered that there is very specific bug that makes it impossible for some data from Parse Adapters to be retrieved locally. As of March 2014, they have been notified of this issue, but as of yet, have not fixed this bug. Instead of waiting on tenterhooks for another release of the Parse Android SDK, we redesigned yet again.

The final hiccup was a notification Wednesday morning informing us that Parse.com had gone down. After refreshing the Parse status page all day, we discovered that it was a bug with push notifications, which fortunately, we were not utilizing and fortunately data storage was still working all through the demo period!

CONCLUSION

We all learned a lot from this final project process and are incredibly proud of our final product. We've told all our friends with Android phones outside of the class to download the app and they've had a lot of fun over the past few days taking pictures, adding funny captions, and telling stories to each other as they occur. We've had people tell us they plan on documenting the stages of their finals period through the app and ask if we can develop an iOS version. If we were to do that and develop this app further, we would start thinking about scalability and using anonymous users with sharing between specific friends as well as the creation of a web-app as well. We would also consider looking deeper into caption overlays. Overall, we had lots of fun with Frame & Tell.