tiny tales

creative storytelling for young children

Angel Ibarra, Deanna Gelosi, Larry Wu, Lucy Lou, and Matthew Tang

Table of Contents

Introducing tiny tales

Children between the ages of two and five are actively developing their language skills every day. Many products and tools are available to help young learners develop their phonics and letter recognition skills, but few value developing their **storytelling** and **creative abilities**.

Our team aims to address this problem space of early childhood language development, storytelling, and creativity with an engaging and easy-to-use app that gives children tools to become **creators of personally meaningful stories**.

We hope you enjoy **tiny tales**.

Designed by Team 17 (Simplex)
Angel Ibarra
Deanna Gelosi
Larry Wu
Lucy Lou
Matthew Tang

Brainstorming

Fall 2020 was an entirely remote semester due to the COVID-19 pandemic, so processes for user interface design and development that might normally take place in person transitioned online. Over two months ago, we started brainstorming possible project ideas that related to this semester's theme of **equity and inclusion**, which resulted in 50 unique ideas.

Over the course of a few of our twice-weekly team meetings, we narrowed down our top candidates to four ideas:

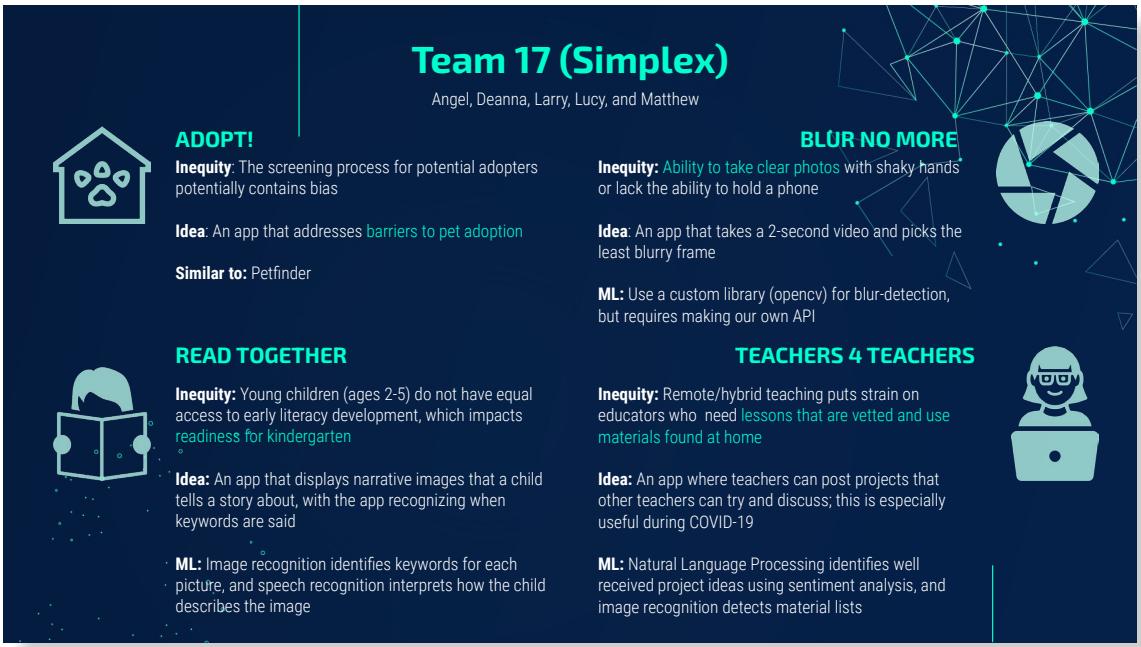


Figure 1: A slide presented to Professor Eric Paulos during a check-in, including our top four ideas and how they would address the theme of equity and the machine learning requirement.

We decided to pursue **Read Together**, a storytelling app for young children. The idea felt compelling to us because it felt both specific and impactful. Our first pass at research revealed that while there were many tools and technologies available to support early learners in their language development skills, few of them took the approach of supporting **language development through storytelling**. Also, one of our team members has existing connections with early childhood educators and parents of young children, so conducting future interviews with key stakeholders and user testing felt manageable.

Our initial idea was to present **pre-generated images with prompts to children**. The questions prompted children to describe what they saw and identify key characteristics of the image and if they responded correctly, they would receive positive feedback. We felt this initial idea promoted language development in a creative way since the child was free to answer the question however they pleased. They would start by describing different aspects of the image while eventually getting to the correct key word. Since machine learning APIs were a part of the project from the start, we imagined using an **image recognition API** to identify what was in each image and **voice recognition** to determine whether the child said key words that were associated with the image. Though the idea was still rough at the time, it felt like an interesting problem to invest in while still meeting the criteria for the project.

Intermediate Sketches

To communicate our vision for the app, we created a sketch to demonstrate how a child would engage with Read Together.

1. The child opens up Read Together.
2. They are prompted with an image on the screen and a question to respond to.
3. The child responds to the prompt using their own words and phrasing.
4. The app determines whether the child said any keywords that match with what's in the image using image and voice recognition.

Figure 2: A sketch describing a possible interaction between a child and the app Read Together.

Overall, we were pleased with how this initial idea used machine learning and how the child could respond in many possible ways. This felt like a **playful take on the game iSpy**, and could result in an interesting interaction for the child. One concern was that the app felt too much like a game with **binary answers**, and we had concerns with whether **voice recognition** would work with young children if the data set had not been trained to include younger voices. We also wondered how engaging the app would be for the child and how the child would **understand the prompt** if it was written if they are learning how to read themselves.

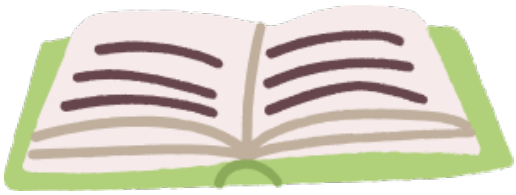
Stakeholder Interview Findings

Our interviews with key stakeholders included a total of five people: Cassandra*, Yuri*, Brenda*, Peter* and Isioma*. **Cassandra** is a mother of two and has 21 years of experience in Early Childhood Education. **Yuri** is a mother of a five-year-old child who co-develops STEM learning experiences with preschool educators and facilitates hands-on activities in preschool classrooms. **Brenda** and **Peter** are parents of a two-year-old child with informal experience with working with young children through babysitting and summer camps. **Isioma** is the site director for a child development center in San Francisco, where she leads a team of teachers and oversees multiple classes of young children.

After these interviews, we noticed the following recurring themes:

- Learning happens when **centered around the child's interests**
- **Creativity** and **storytelling** are often overlooked when developing a child's language skills
- **Independent tasks** for children are valuable learning opportunities as long as the task is developmentally appropriate
- **Screentime** is a concern, especially during this pandemic

As a result, we decided to transition our initial idea of Read Together to **a creative storytelling app** that was built on the child's interests. If we allow the child to create their own stories instead of limiting them to default options, we believed that they would stay engaged longer because they would be narrating stories that interested them. This resulted in children having the power to exercise their creativity while keeping language development the focal point of the app.



Selected Quotes

"He's learning best when he's working on something he's interested in, when he wants to do it." *-Yuri, in reference to her son*

"A child watching stories all day doesn't know how to tell stories, they have to be asked to play out their own stories." *-Cassandra*

"Give parents resources so that their kids don't become addicted to technology, such as recommendations for timing. Give parents tech literacy on how to not create a demon child." *-Brenda*

"If you understand the child, you can provide the best channel for them to learn." *-Isioma*

*pseudonym

Final Idea and API

The feedback from our stakeholder interviews immensely impacted on our final design idea. We built upon our initial idea and imagined the following user flow:

The child picks and uploads their own images that they want to narrate (with help from a parent or guardian as necessary). The app uses Google's Cloud Vision API to generate stories by **identifying key features** in the uploaded images and then placing **images in sequence** that share keywords to build story continuity. When a photo isn't available, a photo is selected from a list of seeded photos that comes installed with the app. Once the child selects a story to narrate, the app records the screen and audio and the video can be saved and played back.

We considered using other Google Cloud APIs, but concluded that **Vision API** provided the most seamless integration and best performance. The speech recognition API was considered, but we concluded that it would not work well for young children. We also considered a text to speech API in earlier stages of development but decided against it since it didn't fit as well with our idea. Our goal was to identify an algorithmically generated way to sequence of photos, so we ended up going with Cloud Vision and building an algorithm using the tags to intelligently generate photo sequences.

Then, **tiny tales** was born.

Competitive Analysis

Most early childhood literacy apps focus on **phonics** and **letter recognition** but we did manage to find one other creative storytelling app for children.

Target User Group: *Imagistory* is a creative storytelling app for children in grades pre-K through 3rd grade.

Functionality: The app offers six picture books for kids to narrate over. Each recorded story is saved and can be viewed and shared at a later time.

Usability: A simple UI and limited text makes the app easy to use; it clearly designed with young children in mind.

Summary: *Imagistory* is an app designed for kids to use their imagination and craft their own story given preset picture books. However, this limited number of picture books **restricts the amount of creativity kids can have with the storytelling**. We will differentiate ourselves from this app by allowing our users to add their own images to make a story that builds on their personal interests.

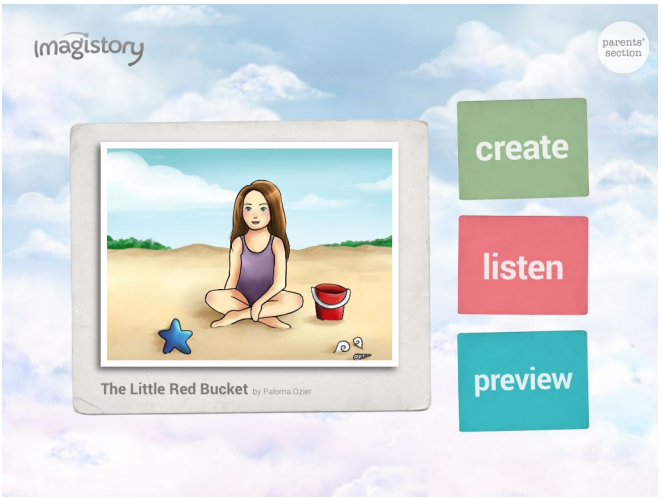


Figure 3: *Imagistory* is a creative storytelling app for children available in the App Store.

Scenarios

When creating *tiny tales,* we considered stories like these to drive our design and development. We wanted it to be an open-ended tool that learners could use in a variety of ways.

Case 1: Daily Recap
Billy, a preschooler, goes on a field trip to the Zoo. His parent is a chaperone for the day and takes photos of Billy next to some of his favorite animals. When they arrive at home, his parent helps Billy add his photos from the zoo to *tiny tales*. Once uploaded, multiple stories are generated from the pictures and Billy can choose which story he would like to tell first. He selects a story that starts with his favorite bird - a penguin! After he finishes his story, Billy shares it with his family.

Case 2: Surprise!
Valeria, a rising kindergartner, has been playing with *tiny tales* for about six months. She has made many stories using the app and loves to add photos of her toys, her pet dog Slimy, and the birds in her backyard. One feature she loves in *tiny tales* is when a random image pops up in her stories. She finds this to be so funny! Her stories often take surprising twists and turns, like the story that started with Slimy and a bone and ended with a bathtub full of candy. She even drew a picture of the story for her friends and often talks about filling her own bathtub with candy one day.

Overall, our aim behind *tiny tales* is that it's a tool for kids to simply express themselves in a creative way, whatever form that takes.

Initial Draft

We designed our initial draft and subsequent wireframes in **Figma.** The goal of the first draft was to imagine how a user would create their own unique story.

The initial draft was inspired by *Imagistory*. We used similar tile images that represent different stories, as well as a button to switch between story creation and listening modes.

Visit our Figma designs at bit.ly/team-simplex-figma.

Figure 4: The initial draft home screen of *tiny tales*.

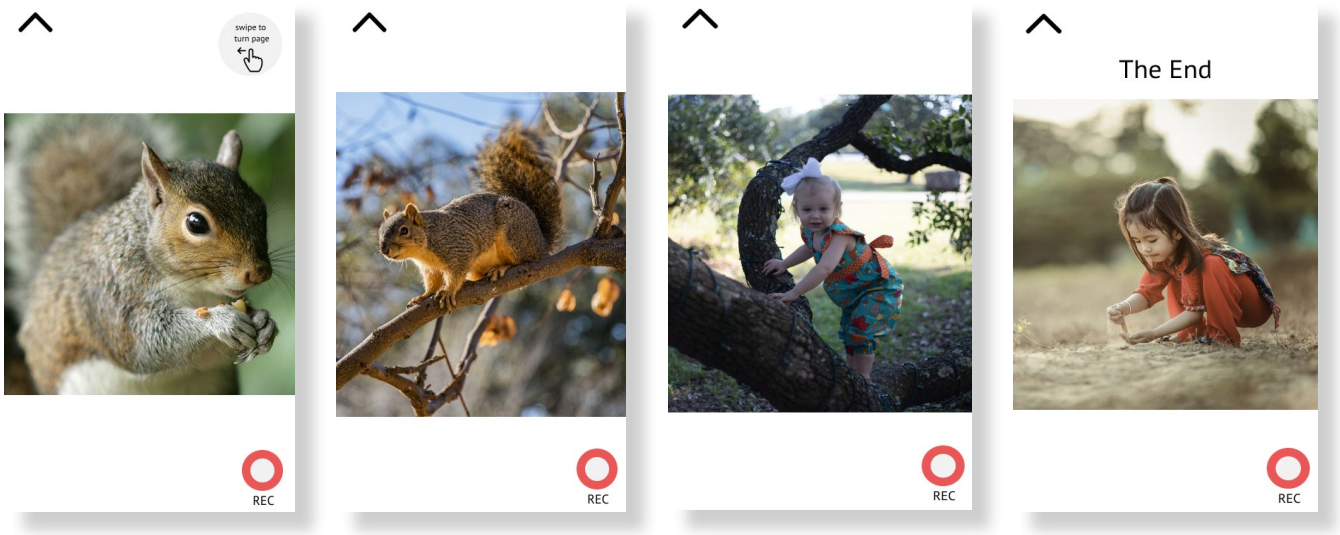
Landing Page

On the main page, there are four squares with images. Each image represents a different story starting point.

In this draft, we designed a prototype for a story that starts with a picture of a squirrel (the image in the top left hand corner).

Activity 1: Tell a Story

Our primary function in the app was to **tell a unique story**. Once the user selects an image on the main screen, the app launches the story. The first image of a squirrel transitions into an image of a squirrel in a tree since they both share the identification of a squirrel. The third image is of a child and a tree, since both the second and third images have a tree in common. The fourth image is of a child, connecting the last two images together.



Figures 5-8: The squirrel story, which users narrate over and swipe between screens to move to the next image.

Activity 2: Save

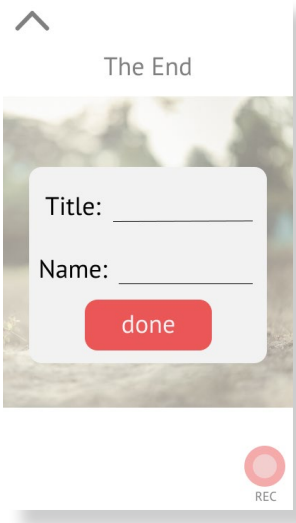


Figure 9: Users can give their story a title and add their name to each recorded story.

The user gives a **unique title** to their story as well as add their name. This information is saved alongside the story and can be played back at another time.

Activity 3: Listen

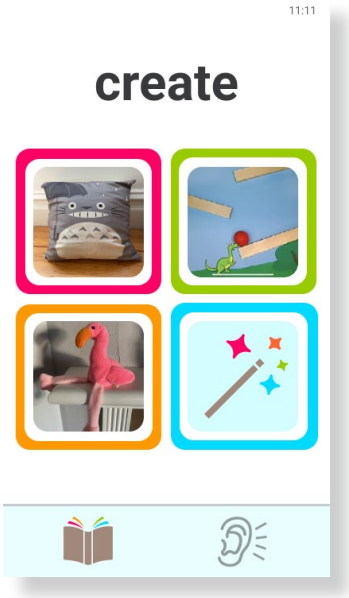
Figure 10: On the *Listen* page, users can watch and listen to pre-recorded stories.

Back to the main screen of the app, users navigate to the *Listen* tab by selecting the ear icon. On this page, they can listen to **previously recorded stories**, just like the example squirrel story.

This initial draft helped us flesh out what form the app might take, which stayed true to form until the very end. In subsequent wireframes, we added more details to make the experience more user-friendly.

Wireframes

Landing Page



In our low-fidelity prototype, we added additional functionality to make the user experience easier and more intuitive. On the home screen, we made a bottom navigation bar so users can toggle between the *Create* and *Listen* pages.

The **magic wand** is a new feature that randomly picks the starting image.

Figure 11: The low-fidelity home screen of *tiny tales*.

Press Record

A new feature we implemented on the story create mode was a button to start recording your story. This allows the user to **choose when to start** their story instead of it starting automatically.

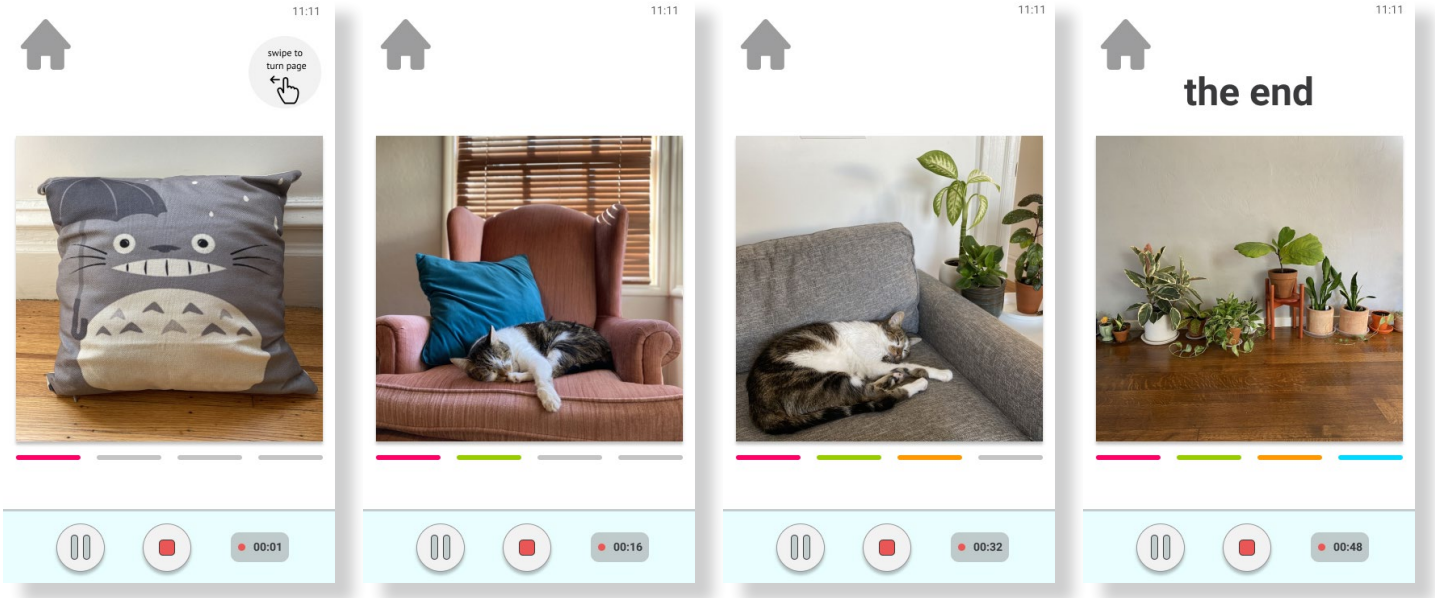
Also, we added a *Home* button to return back to the landing page.

Figure 12: Press record to start recording a new story.

Activity 1: Tell a Story

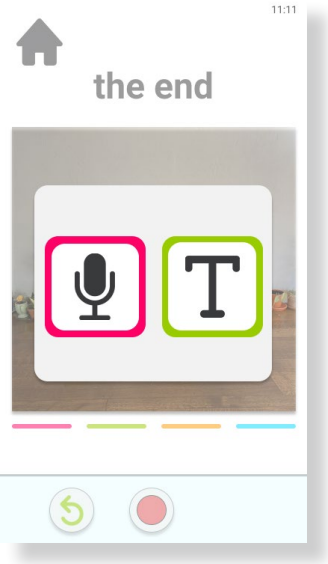
Additional recording features are now available in this version of the app. Pause and play buttons are located in the bottom navigation bar and a timer displays the recording time. **Colorful bars** underneath each image show progress in the story.

In this story, the pillow connects the first and second images, the cat connects the second and third, and plants connect the third and fourth.



Figures 12-15: The Totoro pillow story, which users narrate over and swipe between screens to move to the next image.

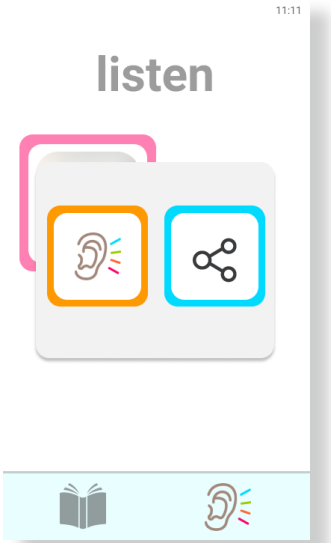
Activity 2: Save



Since our users do not read and write, we included an option for users to **say their story title** instead of typing it.

Figure 16: The updated save screen which adds audio recordings for titles.

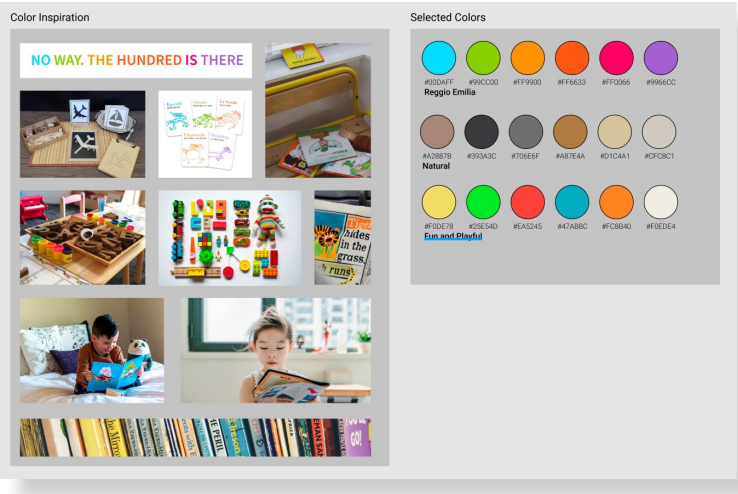
Activity 3: Listen



In addition to listening to the story, we added a feature that allows users to **share their story** with friends and family.

Figure 17: The updated *Listen* page which adds sharing features.

Color Theme



The low-fidelity prototype used a new color scheme that drew from **children's books and toys** with joyful colors and **natural materials**. The colors were also inspired by the branding of the **Reggio Emilia Approach**, an educational philosophy for early childhood and elementary school aged children that centers on self-directed, experiential learning.

Figure 18: Color inspiration for the initial wireframe was informed by childrens books and toys.

User Testing

We used these wireframes during our user testing, and we were fortunate to work with two young learners: Kyle* and Yuji*. **Kyle** (son of Brenda and Kyle) is 2 years old, and **Yuji** (son of Yuri) is 5 years old. We used our initial wireframes to see how our users and their parents would interact with the application. We also asked the parents to send us photos to seed the Figma wireframe to best simulate the experience. We conducted these user tests over Zoom by sharing our screens and asking the child and parent to direct the cursor.

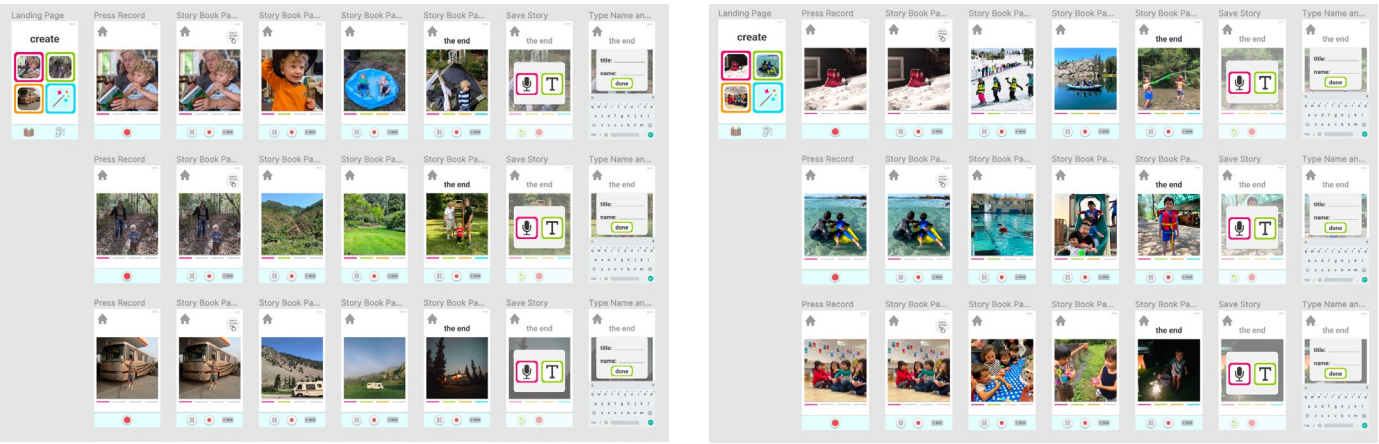


Figure 19: Kyle and Yuji's custom wireframes, respectively, using images provided by their parents.

After conducting the user tests, we found out the following about our application and design:

- The parents were **very engaged** with the child during the storytelling process, always prompting the child with questions.
- Children had trouble coming up with **titles for their stories**.
- For the listen feature, users **struggled with the bottom navigation bar** to switch between screens.
- Users had similar trouble getting to the **sharing button** and recognizing the share symbol that we had selected.
- The **recording button** was unused during tests.

Based on this feedback, we discovered some issues that our application and redesigned the application address issues, leading us to our final design. We focused our final redesign on improving the user experience design. We realized that the while it may be nice to give stories titles, it's not an important feature of the design. We also learned that in Yuji's user test, he retold what he remembered from the photos rather than abstracting away and reimagining a new story. This prompted us to decide to add additional photos to the app to insert randomly throughout the story to break up the retelling of existing stories and hopefully interject new ones.

*pseudonym

Final Design

Our final design kept the same structure as previous wireframes with some notable differences. We created a splash screen and onboarding page to introduce users to the app. We added buttons to the homescreen to add photos and refresh the tile images. We also updated the colors and theme of the app to center around **an orange cat** and its green book.

Splash Screen + Onboarding

Figures 20 and 21: A splash screen and onboarding page for *tiny tales,* respectively.

Landing Page

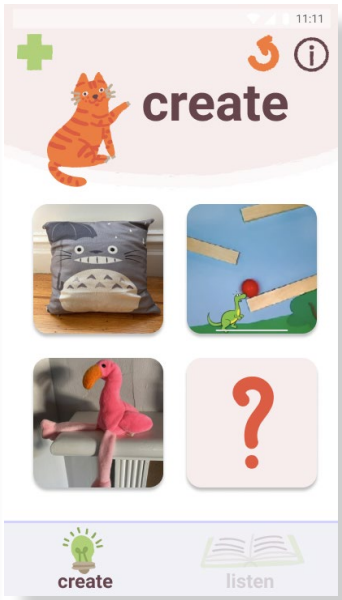
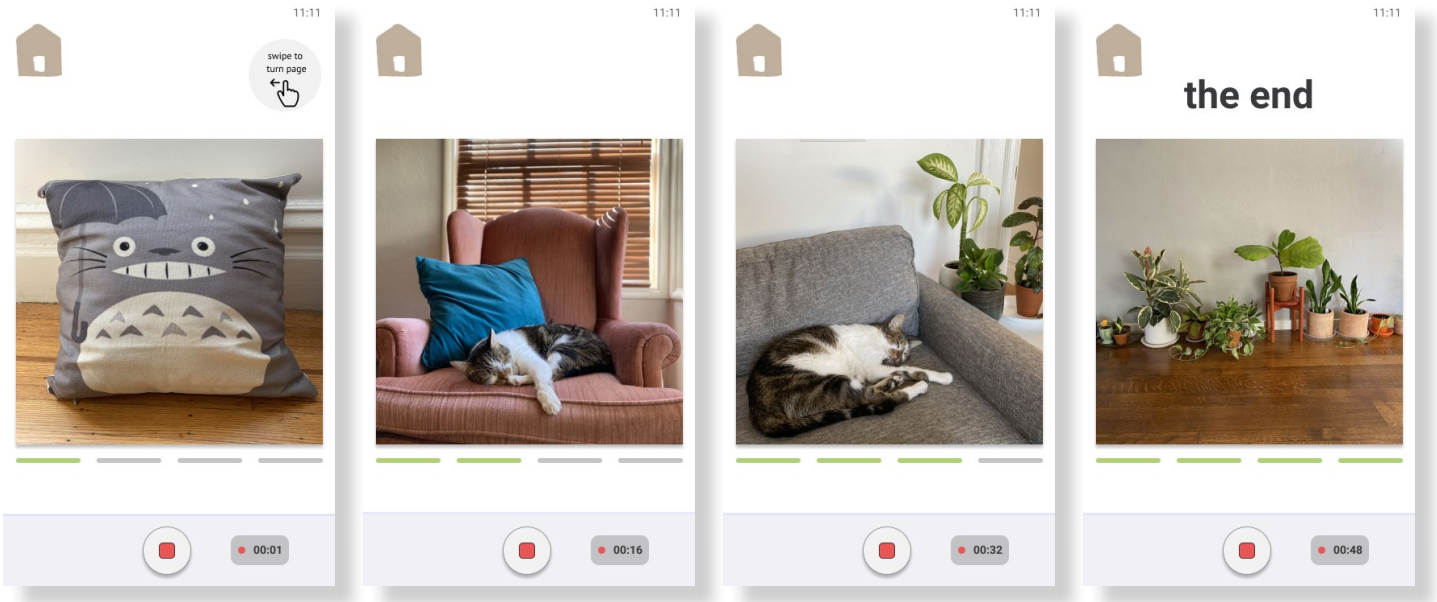


Figure 22: Updated landing page includes new buttons.

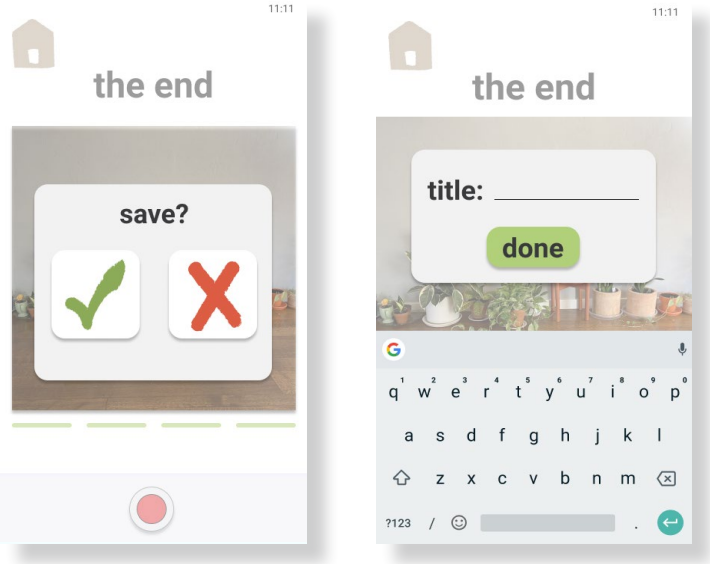
The new landing page includes improved iconography to represent the two main screens: *Create* and *Listen*. New buttons include ***Add Photos***, ***Refresh Landing Page***, and ***Onboarding Screen***. The fourth tile image is a question mark, and still represents a random starting photo.

Activity 1: Tell a Story



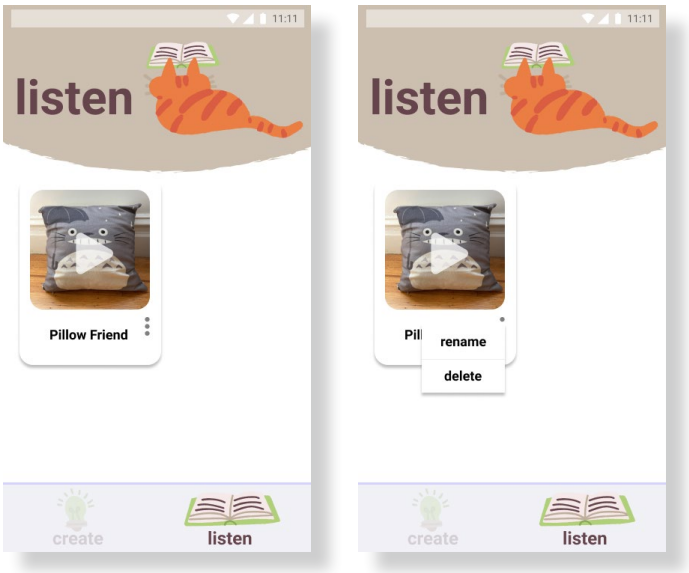
Figures 23-26: The Totoro pillow story, which users narrate over and swipe between screens to move to the next image.

Activity 2: Save



Figures 27 and 28: The updated save screen removes audio recording for titles as well as author name.

Activity 3: Listen



Figures 29 and 30: New *Listen* page includes options to rename and delete recordings. Videos save to the phone's camera roll.

The changes made to the app that resulted in our final design were a combination of our group's efforts with the feedback from our young user testers, their parents, and experts in the field of early childhood education. See **Appendix A** for a one-pager interaction guide and watch the final two minute video for the app here: https://www.youtube.com/watch?v=GDWZCZ9Lgtc&feature=emb_title

Technical Challenges

Front End Challenges

New UI designs resulted front end learning opportunities. The implementation of the cardview using recycler view to dynamically display the created stories was a new challenge. Also, the creation of an options menu popup in each individual cardview was another puzzle. Both of these were challenging because they required extra research which resulted in the creation of separate adapters and xml files.

Cloud Vision API Challenges

The Cloud Vision API wasn't as complicated as expected to implement. There were great examples on the official GitHub documentation. The biggest challenge was to update and refactor the code. The example code used deprecated libraries and was for an older version of Java, which required us to replace some imports and functions with their respective new ones. We originally thought that permissions issues and async issues could take a while, but the examples on GitHub made this issue much easier to solve.

Algorithm Challenges
The design of an algorithm to generate a story required that the backend team work together to find a solution that would be fairly simple to implement. We did not run into many challenges when coming up with a way to use the labels from the Google Cloud Vision API to algorithmically pick images for the story. We created our own data structures and functions in Tools.java to implement the algorithm. One issue we ran into was the differentiation of user images and stock images and how we could have preferences in the algorithm for each case, but some minor tweaks to the algorithm resulted full implementation with little difficulty.

Screen Recording Challenges
There were some great online resources that provided examples of how to record the phone's screen and user's audio at the same time. However, when recording the phone's screen, the window size was fixed and the entire phone screen recorded. This included the status bar at the top and control buttons on the bottom of the screen. Unfortunately, we couldn't find a solution to this problem at this time. We even looked into video editing APIs, but the best that they can offer is to trim not crop the video.

Conclusion

Our team came a long way from our initial brainstorming session to our final product. Along the way we gained great insight from our user interviews and testing, which ultimately shifted our final design to be a **playful, open-ended tool** for young learners to express themselves creatively while developing literacy skills. Since households across socio-economic backgrounds are likely to have a smartphone, we hope this tool also fills a need for families who otherwise **do not have access to creative storytelling resources**.

After speaking with early childhood experts and parents of young children, we felt motivated to approach the challenge of developing early literacy skills from a new direction. With creativity and personally meaningful ideas at the core, we designed a tool that supports learners where they are at. Our goal for this project was to create a tool for kids to use to express themselves in a unique way that promotes creativity and language development through storytelling. Storytelling has profound value that is further enhanced when encouraging children to use their diverse background as a foundation for communicating their powerful ideas.

Future directions of this app include continuing to fine tune the image-selection algorithm to find a balance between user- and app-supplied images. Also, based on final conversations with Yuri after filming our product video, we discussed the option for users to have more control over the sequence of photos. Perhaps another mode of *tiny tales* could be a story sequence entirely selected by the user so they can tell the exact story they want to. More user testing and conversations with experts would help refine this app even further. We're excited about what we accomplished in CS160 with *tiny tales* and thank you for joining us in that journey.

GitHub Repository

https://github.com/cs160-berkeley/project-fa2020-team-simplex

Team Simplex presents

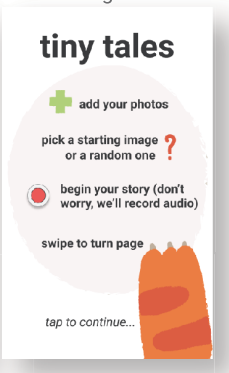
tiny tales

creative storytelling for young learners

A tool for young children to develop their **creativity** and **immagination**, *tiny tales* uses **personal photos** to make novel story sequences for children to narrate.

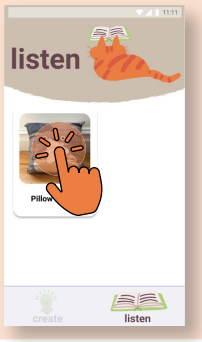
Splash Screen

Onboarding



Task 2: Listen

Listen and watch recorded stories.



Primary Activities

Landing Page

Add Photos
Onboarding
Refresh Starting Images
Starting Image
Random Starting Image

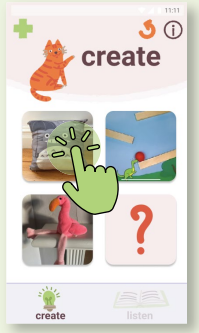
Saved Stories

Recorded Story
Share

Task 1: Record

Optional: Add Photos

Select Image

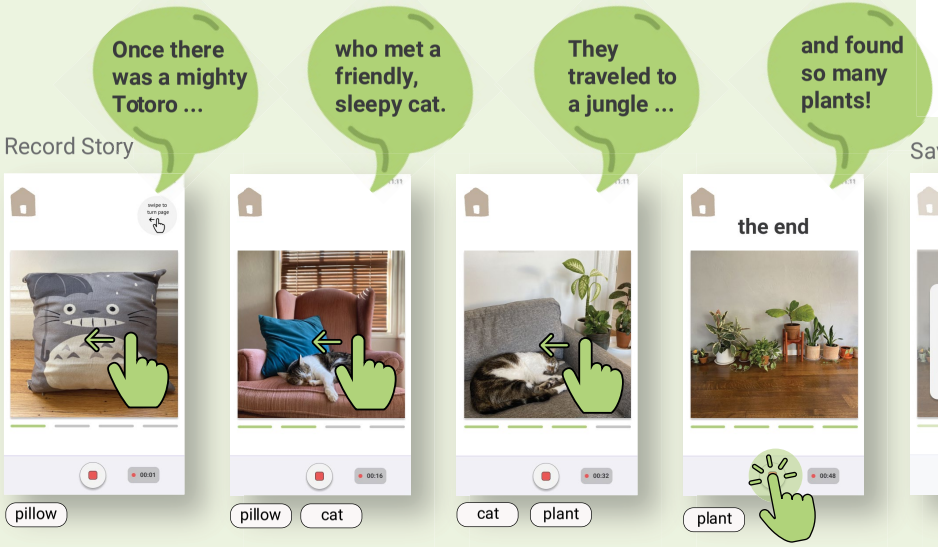


Press Record

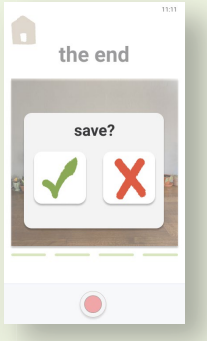
Task 3: Share

Save and share stories from the camera roll.

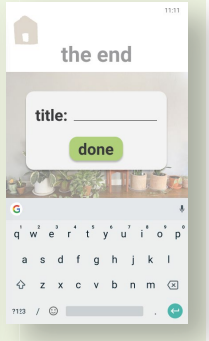
Record Story



Save



Title



After the starting image, related images are selected to provide story continuity.

Google Cloud Vision API labels to images (both preloaded and ones chosen by the user) which allows images with related labels to be placed in sequential order.

Appendix A.