

Spark: Final Report
Streamlining Student Growth Tracking For Teachers

Bronson D. - Kevin K. - David A.H. - Melissa D.

Problem and Solution Overview:

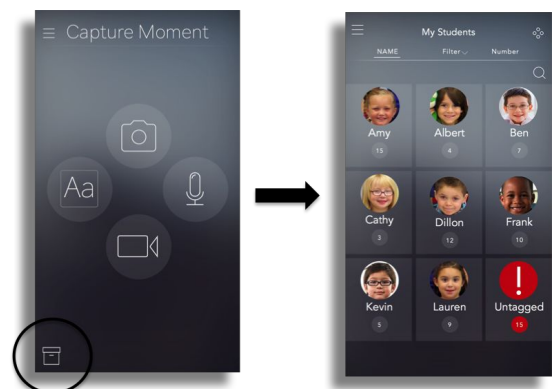
In our current educational framework, student accountability at the elementary school level is a major issue. Teachers, most of whom have thirty or more students in each of their classes, are expected to effectively track and engage with each of their students. Parents, on the other hand, lack the time and means to be involved with their children's education (teacher-parent meetings occur only an average of once every semester), and as a result, students' do not have enough support and guidance during one of the most critical phases of their educational development.

Spark aims to fix this need by introducing a two-fold solution, one that attempts to help teachers account for each of their many students in an efficient manner while also empowering parents to become more engaged in their child's education. We accomplish this by introducing the ability for teachers to capture moments in their students' educational journeys by way of camera, video, voice, and text. With these representative moments, teachers gain a more tangible way of referencing student growth throughout the year. Teachers can also share these moments with parents to better integrate and involve them in their child's education and hopefully empower parents to continue fostering an educational mindset beyond the school setting.



Tasks & Final Interface Scenarios:

Spark integrates three main tasks in its design.

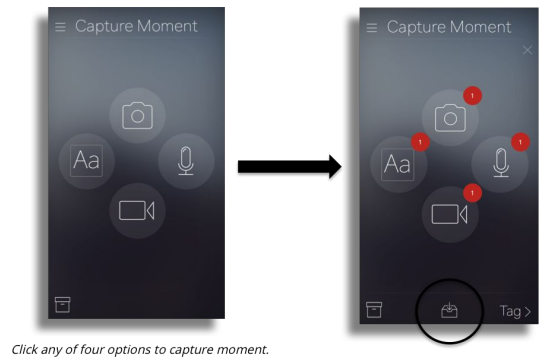


Simple Complexity Task: Easily Sort and Recollect a Child's Developmental History

We accomplish our simple task through the use of an archive, an easy to access database that teachers can use to see what their students have done throughout the year and to account for students that may be falling through the cracks.

Medium Complexity Task: Capture a moment in a child's education.

Having the ability to capture moments in a child's education is integral to how Spark solves the problem at hand. Doing so gives teachers a means to track student growth aside from the traditional means of taking notes, using grades, and relying on memory recall.



High Complexity Task: Share a moment in a child's education with his/her parents.

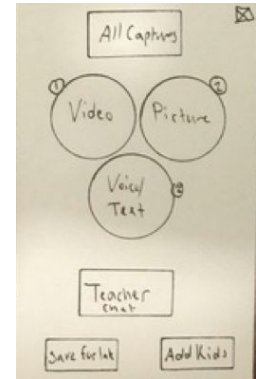
We implemented the high complexity task to allow for better parent engagement in their child's education. Sharing moments gives parents better insight on what their



kids are doing at school, thus integrating them more in their child's educational experience.

Design Evolution:

We wanted to make Spark as easy as possible to use for both parents and teachers. Since the idea of capturing important moments in a child's education is extremely important to Spark's fundamental goal, our original home page had this at the core of its design. Some of the main feedback we received during initial user-testing was that our lo-fi paper prototype was too feature heavy and confusing, so for our medium-fi prototype we simplified the home page even further.



We also added

an extra screen at the beginning that the user would see just after logging in that would serve as a home page; the limitation is that an extra page would be added before a teacher could navigate to transition page, but we thought that having a home page would be logical. However, after listening to and watching teachers' "in-the moment" reactions while they were using the medium fi prototype, many were confused by this additional home screen (they did not know how to advance to the Capture page and what they were supposed to do on it). Since Spark's core functionality was to make it as easy as possible to create and share moments, in our hi-fi prototype we decided to remove this first screen and took the user to the Capture page immediately after opening the app. For our hi-fi prototype, we kept much of the core functionalities of the medium fi prototype, but added features that teachers during user testing mentioned would be very helpful. The limitations

of implementing these features, however, was that we thought it might make our final app too feature heavy and confusing; in order to address this problem, for our high-fi prototype we focused on a really simple, clean design that had clear transitions.



Major Usability Problems Addressed:

For our final design changes, we focused on the results of our heuristic evaluation and attempted to address Heuristic Violations our evaluators found to be severe or catastrophic. Some violations needed simply more extensive refining of already established concepts while other violations required more extensive redefinition and discussion of our project goals and ideas.

Violations that required some refining included:

- Severity 3 Violation: [H2-2]: "Tag" terminology is confusing. This referred to



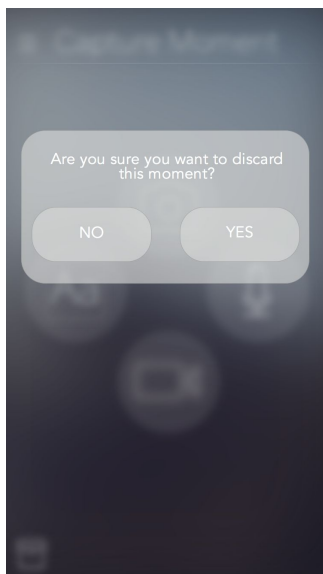
the screen in an individual student's archive of moments and how the word "Tag" seemed to imply a different sort of action as opposed to Filter, which Filters moments by "Tags".

This quick redefinition of one word helped in clarifying how the tagging and sorting system works, so users could maximize the usefulness of archiving moments and tagging them.

- Severity 3 Violation: [H2-2]: “Tagging” multiple students. The concept of tagging multiple students was also a concern that had consistently been raised in our needfinding. We decided to keep with our decision to only allow tagging of one student per moment, as our needfinding showed us lots of potential privacy issues with sending moments of one child to a parent of another. Though those problems did not necessarily apply to our prototype concerns, we had discussed this significantly, and this also allowed us to focus more on the experience of the tool as a whole instead of worrying over something that could be implemented later. For the purposes of the prototype, it was most effective to redefine this constraint as rooted in actual needfinding results while still thinking about possible solutions in the future and balancing the implementation of a working prototype.



- Severity 4 Violation: [H2-5] Confirmation of submissions. A violation deemed catastrophic by our evaluators was the fact that our application was missing confirmation screens for deleting and submitting moments to the archive. As per the standard, we worked to implement these useful error prevention measures and will continue to do so as our development progresses in order to ensure that users commit as few errors as possible.



- Severity 4 Violation: [H2-2] Editing Content. Another violation deemed catastrophic by our heuristic evaluators was the fact that once images/video/voice notes were associated with a given moment, they could not be “edited”. We

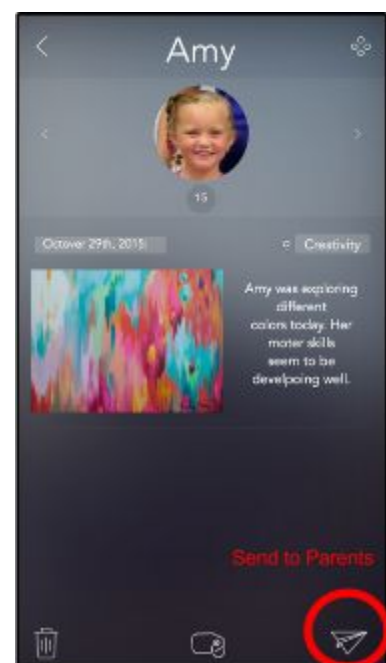
refined this need for editing by giving users the ability to edit in each media capture (i.e. within camera, take and retake a picture as much as you want to) and giving the ability to more fully add media to a moment later on. Editing content of a moment within the process of capturing it seemed to raise lots of technical complications, so we simply refined how to address this user need in a different way.

Beyond these violations, we also addressed other violations that required a little more redefining and discussion of concepts and ideas in our project:

- **Severity 3 Violation: [H2-5] Having untagged moments:** This issue arose multiple times during our needfinding and we constantly discussed the pros and cons of allowing users to capture moments as quickly as possible and the accumulation of untagged moments making our application's usefulness diminish. Teachers we interviewed also expressed concern over growing piles of untagged moments and not having enough time in class to tag all moments. Thus, we focused on allowing for both; allowing teachers to save untagged moments and being able to retag them later. Though not implemented for this prototype, we also discussed adding a reminder function after several untagged moments pile up in order to ensure these do not grow past a certain size.



- **Severity 4 Violation: Parent Communication not as "accessible".** Another concern deemed catastrophic was that people did not find it easy to send things to parents. This violation perhaps was deemed catastrophic given our original intentions, However, through our needfinding, we discovered that teachers would not send all moments to parents, especially since some might cast students in not the most positive light or areas they need to improve in, and parents might react negatively or worry unnecessarily with constant updates. By making the sending of moments to parents more sparse and more



special, the need to send to parents automatically was relegated to when one clicks on an individual moment. This also raised discussions of a future implementation of a Parent Interface Spark, that would focus and integrate more on the needs of that specific group.

- Severity 3 Violation: [H2-10] The concept of a “moment”: This violation also came up during before and after our needfinding. Our conceptual model of a “moment”, which included an array of different media types seemed to take a bit of extra processing for some users and evaluators. However, once grasped, people seemed to really understand what it was about and speak in terms of these moments. For this reason, we decided to redefine our approach to solving this problem and set goals of developing a short but clear tutorial upon first opening the app that primarily explains the concept of a moment. Given our time constraints, however, we were not able to implement this for our hi-fi prototype.

- Severity 3 Violation: [H2-7]

Filtering all students:

This violation refers to the original way we were filtering all students.

Previously, filtering was done for “All Moments” and it was harder to keep track of what moment, when filtered under a certain category, belonged to which

student. Now, filtering by category organizes “All Students” by the number of moments of the given category. We redefined this solution in this way in order to hinge on the well-received ability to view how many moments are associated for a given student, which enables teachers to better keep track of students across a variety of categories.



Ordering and number of moments per students changes

Prototype Implementation:

We created our final Hi-Fi prototype using Swift. Using Swift was a great way to create our app and truly discover what Spark felt like when used on an iOS device. Using Swift also gave us a great deal of design leeway, so we were able to make our Hi-Fi prototype look exactly like our Med-Fi prototype. Using Swift also saved us from having to hardcode as many features as we did in our Med-Fi prototype, such as the capture screen scenarios. Additionally, having Swift meant that we could access all of the iPhone's capabilities, making for camera, video, and voice capture to work just like in the actual product.

The hardest part about using Swift was our lack of knowledge going into this project. We learned the language on the spot and did our best to create our app. Another fallback was the inability to create a back-end data structure for storing moments captured using our prototype, but that was largely an issue of lacking experience using Swift. Lastly, due to our lack of knowledge regarding Swift, we were unable to implement auto-alignment correctly, so our prototype only works correctly (with regards to visuals) on the iPhone 6plus.

Our Hi-Fi prototype has a hard-coded archive with pre-populated student profiles. We also "simulate" the action of sharing these moments with parents (without actually doing so). Lastly, like we stated earlier, we do not actually store any captured moments due to a lack of having a back-end data structure - we just simulate the act of doing so.

Our app is missing the features that we hard-coded as well as the ability for parent-teacher chat. In the future, we would code in a back-end data structure to manage the storage of captured moments, and we would implement a parent-teacher chat interface either using e-mail, texting, or online chat services.

Summary:

Overall, we had a really great time building Spark. It's incredible to see how far our app has come since the original paper prototype we made during the first few

weeks of the quarter. We're very happy with the way our final product turned out, especially since none of us had any XCode or iPhone programming experience (or design experience, for that matter) to begin with. The final app successfully implements almost all of the major functionalities we had hoped it would do, and the iterative design processes we learned in this course helped us craft, hone and communicate our ideas to each other and to the teaching staff. Some of us have even been strongly considering continuing with this project next quarter with CS194H. The design thinking skills and insights into how humans interact with computers we gained during this quarter were invaluable, not only in terms of the tools we were exposed to but also because of how we learned to think, and applying them to a topic we all cared about was a rich and rewarding experience.