PictureBlocks : Constructing and Deconstructing Picture-Driven Literacy Development

Sneha Priscilla Makini

B.E., VISVESVARAYA TECHNOLOGICAL UNIVERSITY (2013)

Submitted to the Program in Media Arts and Sciences, MIT Media Lab,
in partial fulfillment of the requirements for the degree of

Master of Science in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY (MIT)

September 2018

© Massachusetts Institute of Technology 2018 . All rights reserved.

Author ....
Program in Media Arts and Sciences, MIT Media Lab,
Aug 17, 2018

Certified by ....
Deb K. Roy
Associate Professor of Media Arts and Sciences, MIT Media Lab
Thesis Supervisor

Accepted by ....
Tod Machover
Academic Head
Program in Media Arts and Sciences

Thesis Reader . . . .

Scot Osterweil
Research Scientist
MIT Comparative Media Studies/Writing

Thesis Reader . . . .

Roger Levy
Associate Professor
Department of Brain and Cognitive Sciences, MIT

PictureBlocks

Submitted to the Program in Media Arts and Sciences, MIT School of Architecture and Planning, on Aug 17, 2018 in partial fulfillment of the requirements for the degree of Master of Science in Media Arts and Sciences.

Abstract

Pictures play an important role in aiding literacy development amongst children. Present day educational apps for children take advantage of pictures in an instructionst manner - such as a flashcard, drag-and-drop, or fill-in-the-blanks approach. However, research indicates that following a constructionist approach rather than instructionst, where children actively construct meaningful projects playfully, leads to better engagement and learning. It is also universally established that children across the world enjoy creating and drawing pictures as a means of self-expression. Despite the evidence from the literature and the data, there is a lack of constructionist approaches towards picture-based learning apps for children.

The goals of this thesis are two-fold:

1. Successfully design and evaluate a picture-based, constructionist literacy learning app in order to address this gap.
2. Explore the unique affordances/implications that this exploratory approach has on children's self-expression and learning.

This app is called PictureBlocks, and it is designed for children between the ages of 5-9 years. The design of PictureBlocks is refined through several rounds of playtesting. Finally, a 15-day pilot study conducted in children's homes helps evaluate the app's design. Data analysis and findings also establish unique affordances and future implications for picture-based, constructionist learning apps.

Thesis Supervisor: Deb Roy
Title: Associate Professor of Media Arts and Sciences, MIT Media Lab

Acknowledgements

To quote Sir Isaac Newton – *"If I have seen further, it is by standing on the shoulders of Giants."*
This work would not be possible without the direction and well-wishes of several important people.

First and foremost, I express gratitude towards my advisor, Deb Roy, for helping me shape this research, and providing me with room to grow and explore. Thank you Deb, for your belief in me right from the start and motivating me in more ways than I have let on.

I would also like to acknowledge my readers - Roger Levy and Scot Osterweil, who took time out from their busy schedules. Prof Levy, your excellent and challenging class made me develop an enthusiasm for learning about language and Prof Osterweil, creating games for social change was one of the most enjoyable experiences during my time at MIT. Thank you both immensely for your valuable feedback and inspiration.

I also take this opportunity to thank Mitch Resnick for his crucial feedback in the early stages of PictureBlocks, my UROP Ini Oguntola for his significant contributions towards app development, Craig Ferguson from the Affective Computing group for impromptu help with Unity, Pedro Cuéllar and Huili Chen from the Personal Robots group for help with study recruitment, Jim Gray for his attentive feedback, and Fengjiao Peng and Lauren Fratamico for assistance with running the studies.

I am incredibly grateful to be a part of a group such as LSM - with wonderful colleagues who have together made the lab feel a lot like home. I am especially appreciative of my time spent with the talented learning team, who have gone above and beyond with their support and helping me with literacy related research.

Particularly, I would like to thank the following people - Andrew Heyward, Ann Yuan, Anneli Hershman, Bridgit Mendler, David McClure, Doug Beeferman, Eric Chu, Eric Pennington, Heather Pierce, Ivan Sysoev, Juliana Nazaré, Kevin Aiken, Keyla Gomez, Lauren Fratamico, Marc Exposito, Martin Saveski, Mina Soltangheis, Nabeel Gillani, Nazmus Saquib, Neo Mohsenvand, Prashanth Vijayaraghavan, Preeta Bansal, Raphael Schaad, Shayne O'Brien, Soroush Vosoughi and Winston Chen. Each of them has augmented my time at the lab, whether it be in the form of their guidance, availability, or friendship. I wish there were enough words to capture the late night working sessions, fun times at the offsites, 99 high-table hangouts, GetFit minutes, bonding over free food, group meetings, and inside jokes. Each of you mean a lot to me.

MIT would have remained a distant dream if not for consistently sound mentorship throughout my life. My sincere thanks to my school teachers, as well as Florian Fuchs, Robin Jeffries, Dilip K Sen and Bagirath RK for being remarkable role models, in addition to writing my letters of recommendation. Special shout out to Pratul Kalia for introducing me to open-source technology and GSoC, which has since triggered a chain of positive life events. Thank you to Jonathan Farr for helping me get through my Year Without a Summer.

For being a gem of a friend, I would like to recognize: my person - Fengjiao; my mirror image - Eric P; my first friend in Boston - Ashok; my friend in need - Jahnavi, my co-working partner - Nikhita; a thoughtful friend - Jules; my swim buddy - Lauren; my adopted brother - Prashanth; and my favorite person to annoy - Spots. Thank you all for helping me endure the Boston winters with your warmth and camaraderie.

I am forever indebted to Agnish Balasubramani, who has uplifted my morale every step of the way through to the finish line. Thank you for being my much needed PR person to boot and overall, an exemplary human being. My acknowledgements also go out to Abhilash Raj for being a great cheerleader, my friends from PRS and MCC for ensuring we always have each others backs, friends from MVIT for the craziness, and my friends from Zynga for all the good times.

Finally, I owe it all to my family and their faith in me, and whose sacrifices I do not wish to undermine with the brevity. To my grandparents - Krishna Rao and Nirmala Devi, whom I love more than anything else in this world, and whose love will always be my guiding north star. To my sister - Supriya, who is one of the strongest and kindest people I know. To my aunts and uncle - Suneetha, Suvarchala, and Sudheer who took care of me as one of their own, and continue to do so. And lastly to my mother Suvarna Latha and father Srinivasa Rao, who I hope would be proud of me.

Dedicated to my thatha (grandpa),

Mr. Nelapati Venkata Krishna Rao,

who passed away during my time at MIT,

but forever lives on in my heart.

Contents

List of Figures

List of Tables

1 Introduction

" 'What is the use of a book', thought Alice, 'without pictures or conversations?' "

Lewis Carroll, "Alice's Adventures in Wonderland"

If one carefully peruses this quote borrowed from Lewis Carroll's *Alice's Adventures in Wonderland*, they would find that Alice's assertions are indeed valid — especially when applied to the field of language and learning. While the written word has long been a source of information and communication, so has visual information and social interaction played a huge role in language acquisition and literacy. In fact, they are of even more importance during early childhood where children are still developing via communicating with their parents, interacting with the real world and beginning to read and write [63] [23].

Children's learning with books evolved to include pictures. Pictures are stimulating, engaging, and also provide a sense of context for the child when they encounter unknown words [34]. With the advent of hand-held technology, smartphone apps for children's learning have also grown to incorporate these picture-driven approaches, but in the form of structured puzzles or quizzes with pre-determined right and wrong answers, and less room for open-ended exploration [72].

With a majority of these apps, mirroring picture books, children are instructed to learn words with the help of pictures on the side, serving as a complementary aid and as an engagement factor. These interactions are usually instructionist[1] in nature - where the primary form of interaction for the child is to drag and drop objects or learn with flashcard styles [72].

[1] Refers to the learning theory of Instructionism which specifies that instruction needs to be improved in order for better learning results. It's usually focused on a teacher teaching with a pre-determined process.

However, research has indicated that while these structured, rewards-based apps may produce the right answers, it does not necessarily mean that children fully understand or are engaged [25]. Instead, children learn most effectively as an active learner when they are playfully engaged in constructing meaningful projects in the world [25]. This style of learning is referred to as a constructionist [2] approach [22] [51].

[2] Refers to the learning theory of Constructionism where the focus is on learning rather than teaching. It involves students drawing their own conclusions through creative experimentation and the making of social objects.

Despite the evidence from the literature supporting constructionist approaches towards learning in children, there are very few examples of such literacy apps. Plus, the majority of the popular literacy apps for children target very basic literacy skills and don't delve into higher order skills such as self-expression, story telling, reading fluency and so forth [72] - which have been found to be critical to the process of deeper literacy development [49] [57].

Creating with pictures - whether it be via drawing, making collages or using stickers, are universal examples of the ways in which children across the world engage in meaningful projects, as it both serves as a means of self-expression and helps children better understand objects and the world around them. As a result, aside from being included as an adjunct to words within literacy apps, pictures can serve as a powerful medium for children to engage in activities that foster these higher order literacy skills.

Therefore, the motivation behind this thesis is to address these gaps in order to promote deeper literacy learning via a picture-driven, constructionist approach. We combine the powerful benefits of pictures with those of constructionist principles in a mobile/tablet based app called PictureBlocks. Our app is meant to target children aged between 5-9 years old , because children:

1. stop drawing scribbles and progress to symbolic representations starting from around the age of five and onwards. [29]
2. start writing words and stories/narratives around that time. [18]
3. there are fewer literacy apps available for older children in the range of 6-8 years old, in comparison to apps for early childhood [72]

Figure 1.1: App Icon

The design of PictureBlocks also capitalizes on the technological advances in visually-driven forms of communication, as well as on the ease of access to large datasets of freely available icons online.

1.1 Contributions

1. App

   A self-expressive, mobile application for children which uniquely combines the power of pictures in literacy learning with that of a constructionist approach.

2. Design

   A set of design recommendations that are derived from the user experience of the app, and is backed by playtesting studies and a deployed pilot with children.

   Our design approaches and results can help other app developers, educators, and researchers reason about constructionist literacy app designs with a focus on pictures. To the author's knowledge, no such guidelines or published material examining such an app design and a careful evaluation of its elements currently exist.

3. Data And Analysis

   Data collected and subsequent data driven analysis can help provide findings in the ways of interaction with the app:

   - Words: Data on the words children type in order to get corresponding sprites, and how they type them.
   - Pictures: Types of pictures created by children using PictureBlocks.
   - Visualizations: The Picture Trees visualizations can help stakeholders in a child's learning understand their exploration of associated words. It can also serve as a useful tool for educators to analyze learning data.

   Finally, this collected dataset isn't limited to a specific set of research questions and can be used in a plethora of ways as our questions evolve in the future.

1.2 Terminology

Before we proceed, Table 1.1 defines some terminology that will be used throughout the rest of this document:

Term	Description
Sprite	Used to describe an image/icon representing a single word or object. For example, an icon of a 'cat' or a 'ball' is a sprite.
Picture	Used to describe a composite image, created by combining sprites together on a background image. For example, a forest themed background image with multiple animal sprites within it can be defined as a picture.
Imageability	The ease with which a word gives rise to a sensory mental image is called imageability. Words with high-imageability are therefore, referred to as imageable words.
Word Associations	Contextually similar words that arise in one's brain as a reaction to hearing or using a specific word.
Parent Sprite	A sprite when tapped, depicting three more sprites which are its word associations.
Child Sprites	Sprites which are word associations to a tapped sprite or a parent sprite.

Table 1.1: Terminology Table

1.3 Outline

The remaining chapters are arranged as follows:

- Chapter two reviews relevant literature. It also highlights present-day picture-driven mobile applications which focus on children's literacy learning.
- Chapter three describes the design and implementation of PictureBlocks and its supporting components, as well as the factors driving these decisions.
- Chapter four details the playtesting sessions, subsequent observations and iterations, and finally, the pilot study conducted with the PictureBlocks app.
- Chapter five explains the data collection and analysis, and reveals the subsequent findings from the pilot study.
- Chapter six concludes with a reflection of the results, limitations and suggestions for future research.

2
Background and Relevant Work

"For a small child there is no division between playing and learning; between the things he or she does just for fun and things that are educational. The child learns while living and any part of living that is enjoyable is also play."

Penelope Leach

In this chapter, we explore the state of present day literacy-based mobile apps and games for children. Our focus is on designing and exploring constructionist, picture-driven approaches to literacy learning. This section surveys the potential of apps within this genre - highlighting good design, areas of improvement and exploration, thereby informing the direction of this work.

2.1 *The State of Literacy Apps*

Literacy - a person's ability to read and write effectively, is vital for personal, economic and community development. Amongst children, the foundational reading and writing skills that emerge during early childhood have been found to have a strong correlation to later conventional literacy skills. Early literacy skills have also been linked to higher academic achievement, reduced grade retention and enhanced productivity in adulthood. [5] [27]

Key findings from a recent survey (2017) indicate that nearly all (98 percent) children in America, age 8 and under live in a home with some type of mobile device [61]. Out of them, 88 percent of parents with kids in the 5- to 8-year-old range have downloaded apps for them onto these devices. Furthermore, the digital divide between low and high-income households has virtually disappeared [61].

Despite this surge in digital access, 37 percent of children arrive at kindergarten without early literacy skills, and continue to stay behind their peers in later years unless there are any interventions [37] [31]. This deficit in later years is reflected in the data from the National Assessment of Educational Progress, whose most recent Nation's Report Card assessment (2017) states that only 37 percent of all fourth graders are reading at grade level, and 32 percent are below even basic levels of proficiency [9].

Due to their ubiquity, educational apps on smartphones have the potential to support children at different stages of literacy, and consequently design interventions in this undertaking. Therefore, it is important to design and develop high quality educational app content that is child-driven and contextually relevant.

To become proficient with language, children between the ages of zero and eight must master a number of different skills, with varying complexities [20]. However in a recent survey of top award-winning and best-selling literacy apps (2015) [72] , the age range and the types of literacy skills that a majority of these apps cater to been found to be severely skewed. Only 5% of the sampled apps were targeting 6-8 year olds, compared to apps targeting 0-5 year olds. The report also found that there was a preponderance of apps targeting a concentrated set of basic literacy skils, such as phonemic awareness and alphabet knowledge. Subsequently, this indicated a dearth in apps providing deeper literacy learning for children, by targeting skills such as self-expression, story-telling, comprehension, et cetera.

Moreover, an overwhelming majority of these apps follow an "instructionist" paradigm, where the main interactions are in the form of "puzzles, games or quizzes" which have predetermined right and wrong answers and/or rewards. In fact, the majority of the studied award winning and best selling apps (71%) incorporated these style of interactions. [72] However, the validity of such structured approaches have been questioned in light of findings that they are not viable tools for children's exploration or self-expression - resulting in lower engagement. [25]

Instead, the theory of constructionist learning put forward by Seymour Papert[22] [51] , has been shown to promote more meaningful learning and facilitate intrinsic motivation, with the child as an active learner [28][1].

[1] Active Learning is where the student is not just passively registering information without purpose or thought but actively involved "minds-on" by performing activities that require thinking and intellectual manipulation.

2.2 *The Role of Pictures*

Words are only one medium through which children derive meaning and express themselves. Visual information or pictures are intuitively understood and have several advantages ranging from understanding a story or a setting faster, to linking the pictures with information we already know sooner. Furthermore, the juxtaposition of visual information with written language together complement each other and can sometimes help fully express what the other cannot entirely convey i.e. visual information can help foster the written word, and vice-versa. [54] [24]

Children's books have already evolved to utilize this symbiosis effectively. The illustrations in picture books help:

- establish a setting or a mood to the story
- relate to the characters or objects children are already aware of
- inspire curiosity
- entice them to read and interact with the text
- foster their appreciation of art and beauty
- simulate and promote children's creativity
- contribute to textual coherence
- help reinforce text

All these qualities help children foster their literacy skills and extract a more enhanced understanding of linguistic language from books. [35]

Once children consume these stories and information, they get a chance to apply it to their real world. They use their active imagination to paint a picture verbally and/or literally to communicate with their peers and elders. One of the universal examples of constructionism amongst children is their love for making pictures - whether it is through drawing on a sheet of paper or a computer, creating a collage or putting stickers together. Making pictures not only cultivates self-expression and creativity, but stimulates a narrative in the child's mind, strengthening literacy skills. [67] [12]

Most literacy based smartphone apps emulate picturebooks in that they include a similar juxtaposing of pictures with words to aid children's learning. However, this similarity is limited in the fact that here, pictures are provided as a complementary aid to the instruction rather than being allowed to be manipulated by children in the learning process. These apps follow a fill-in-the-blanks or a drag-and-drop style approach where the children are not actively creating words and pictures but mostly consuming them. They are not harnessing the history of benefits that creating pictures has to offer.

As per the author's knowledge, there is no literacy-based constructionist app for children that uses both pictures and words for the purposes of meaningful self-expression. Therefore, there is a need to research how to go about designing such an app effectively, as well as explore what unique affordances are offered via this interspersion, that can contribute towards children's literacy learning.

2.3 *Related Apps and Games*

In this section, we review relevant apps and games, and understand their motivations and user experiences. Due to the large, growing number of mobile applications in the market, there is a lack of a standard, comprehensive data set evaluating these apps, and their approaches. For this purpose, we researched and shortlisted apps that focus on literacy and utilize pictures in some form.

While the app details are obtained from reviews by sources such as Common Sense Media [38] and Children's Technology Review [59], some of the listed limitations are the author's own views with respect to this line of picture-driven research.

SpeechBlocks

Also developed at the Laboratory for Social Machines, within the MIT Media Lab, SpeechBlocks is an example of a constructionist literacy app which allows children to explore spelling principles in an open-ended way. Children play with letter blocks on the screen by pulling them apart and putting them together using their fingers. Each sequence of letters joined together is pronounced aloud by a speech synthesizer, including nonsense words. [68][69]

Figure 2.1: SpeechBlocks

Initial SpeechBlocks studies show potential in terms of engagement, social learning, and developing a sense of agency. [68] Additionally, the play data collected from SpeechBlocks has helped provide a more descriptive view into children's learning processes and seems insightful for educators, parents, and researchers. [65]

SpeechBlocks demonstrates the importance of constructionist approaches to literacy by using letter blocks. This only further necessitates the exploration of such approaches by looking into the impact of applying both words and pictures together.

Endless Wordplay

Endless Wordplay includes a character called the Alphabot and uses a phonetic approach for word building and spelling. [26] Children can progress through levels of difficulty, with each level consisting of a set of rhyming words which are shaken out of place. These "movable alphabets" need to be dragged into place and upon completing a level, an animation acts out a sentence with the words just made by the child. For example, an animation of a fat cat sleeping on a mat to highlight the rhyming words it contains. [40]

Montessorium: Intro to Words (previously Alpha Writer)

The creators of this app have used Montessori methods while designing the app's experience. According to them, writing comes before reading in Montessori methods, therefore, the focus here is on helping children become better at putting phonetic sounds together to form words. This app includes movable alphabets and sounds out letter combinations (similar to SpeechBlocks). [46] The app provides different gameplays - kids need to usually spell words while provided with a hints such as a picture or how the word sounds. Other interactions includes a storyboard where kids can drag and drop pictures and words to create juxtaposed stories. [41] The presence of a storyboard fosters creativity, although, the pictures in this app are either only used as an accompaniment or dragged and dropped as stickers.

Pogg

Pogg is an alien character whom the app is named after, which promotes itself as a fun, educational app for young children (ages 1-8). [73] Kids are given the open-ended question of "What should Pogg do now?" upon which, the child can enter a word into the keyboard. Pogg then most likely will enact the typed out word - such as upon typing "Run", Pogg is animated to run immediately after. [66]

Pogg also comes with a picture mode where the child can tap on icons that represent these verbs, instead of typing them out. Pogg is said to be well received by the speech therapist and special education community as it helps autistic children with spelling, vocabulary and speech therapy. [73] [66] Pogg establishes a good word picture (animation) correlation, and is open-ended. The app allows for "tinkering" [36] with the letters and choice of words, but there is no "tinkering" with the animations/result itself.

The following are some example of apps that don't specifically target literacy, but still promote interactions between words and pictures.

Draw Something

Zynga's Draw Something is a turn based game for people aged 4 years and above. The player hand draws a chosen word onto a simple canvas area, and the opponent has to guess the word by looking at a combination of the drawing as well as a group of scrambled and additional dummy letters that contain the word. [50]

Each round has a timer and uses various gamified features such as coins (rewards) to buy bombs (hints), that help the guessser unlock letters. However, this app is not specifically meant for children as there is no upper age limit. [39]

ScribbleNauts

ScribbleNauts, with its catchphrase - "Write Anything, Solve Everything" , is an award-winning app that requires players to spell into existence any object that can help them solve puzzles. It includes more than ten thousand words and there could be several different pathways to solving a certain puzzle. Some solutions could range from creating a ladder to climb up a tree or fly to it by adding wings to your character. [7]

In newer versions of the game, players can even add adjectives to existing objects and transform their properties. In gameplay, the player enters a word for the corresponding object into a notepad to reach a certain objective, or to manipulate the environment to go to a next level, and receive rewards. Scribblenauts is only for children aged 10 years and above and while a great game, it is aimed for those who already have a significant vocabulary used to achieve the game's end goal.

Games by TinyBop

TinyBop makes several playful and experimental award winning games for children. On one hand, their Explorer's Library apps provide an interactive environment for children to learn science concepts about the human body and the solar system, with animated objects and some accompanying text. On the other hand, their Digital Toys apps are open-ended construction kits that focus on creative thinking, problem solving and story telling. [70] TinyBop games have the perfect balance between goal-setting and open-ended exploration. To the author's knowledge, however, there is no literacy focused game that is part of this collection.

2.4 *Exploring A New Design*

Research tells us that if a child also expresses themselves while constructing their learning environment, it helps them develop a richer understanding and is related to increased intrinsic motivation and self-efficacy. [75] [57] Especially in the area of literacy development, authentic self-expression is an important higher-order literacy skill that has close ties towards promoting lifetime literacy [64][49].

However, there are very few avenues for children that combine literacy learning as well as self-expression into one app or a game. Most smart phone apps for children focus on literacy and self-expression as two separate processes. Apps that target literacy use animations or pictures as a consumption aid, and are limited in ways that children can express their creativity through them.

On the other hand, apps that focus on self-expression and creativity as their end goals, have little learning with respect to literacy elements as part of the process. The related apps shortlisted in the previous section are taking positive steps in this direction. Yet, there are still areas of improvement that we addressed, as well as a lack of published research material analyzing their approach and effects.

With the surge in new ways of engagement, and advancements in graphics and visualizations, there are more and more alternative ways of applying pictures and words together besides juxtapositions such as in picture books. In a world where words typed can be replaced with emojis [2], stickers or gifs, and animojis can use our natural voices for an animated emoji [3], or AI can recognize our doodles and replace them with icons [4] - utilizing such novel methods can greatly enhance the

[2] Inc. Apple. *Use Emoji*. 2018. URL: `https://support.apple.com/en-us/HT202332` (visited on 07/31/2018)

[3] Inc. Apple. *Use Animoji*. 2018. URL: `https://support.apple.com/en-us/HT208190` (visited on 07/31/2018)

[4] Google. *AutoDraw*. 2018. URL: `https://experiments.withgoogle.com/autodraw` (visited on 07/31/2018)

way children engage with literacy apps, and at the same time nurture their creativity. Furthermore, combining such engaging picture-based features with playful, constructionist principles can together result in new modes of interaction and learning that need to be explored. This is the focus of our research.

We have already established the significance of pictures and why applying constructionist principles in conjunction with them towards educational apps shows potential. There are a myriad of ways in which we can create new interactions coupling pictures with literacy elements. Children have active levels of imagination, and we can gain insight into their purpose, processes and passions by studying their created pictures. Also, while children make pictures, there are no right and wrong answers promoting a less evaluation-oriented and more open-ended form of play. In this approach, we also take advantage of the easy accessibility to open-sourced datasets of icons available online, such as the FlatIcon database. [5]

[5] FlatIcon describes themselves as "..the largest search engine of free icons in the world." - `https://www.flaticon.com/`

Keeping these points in mind, we propose exploring the design of a new learning environment where picture creation (macro interaction) is the viable end-goal and a variety of word-picture interactions act as the smaller sub-goals (micro-interactions). While the end goal focuses on the product in order to boost engagement, creativity and self-efficacy amongst children, the sub-goals focus on the process itself by focusing on spelling principles, phonemic awareness, vocabulary expansion and so forth. This is the foundation of our app - PictureBlocks.

In the next chapter, we will explore PictureBlocks in further detail and survey the research behind its design and development.

3
System Design and Development

"A picture is worth a thousand words.
An interface is worth a thousand pictures."

Ben Shneiderman

How did we go about designing a picture-based, constructionist children's app for literacy learning? We address this question in this chapter and provide a detailed account of screen-flows and interactions within PictureBlocks. We also describe our app's software architecture and the challenges we encountered.

3.1 Setting

Imagine yourself in the shoes of a six-year old child in a classroom. Anytime you want, you can get a blank sheet for sticking stickers on. Your teacher has a sticker box filled with thousands of stickers, more than you can possibly think of. In a simple world, one could just peek into the box and take whichever sticker catches their eye.

However, here, you have to ask your teacher for a specific sticker, and they'll open this box and give it to you. You can get as many stickers as you want. The only catch is that you have to spell out the word for the sticker you want correctly, and if it is in the box - you will get it.

So at first, you begin by spelling words that you already know quite well, maybe something easy like 'cat' or 'ball'. Maybe you even want to make up silly words and spell them because it is fun saying them out loud, although you know that you won't get a sticker for them.

But with time, you want to get some new stickers like 'dinosaur' or 'dolphin', and you are not entirely sure on how to spell them. So, you go to the teacher and try to guess the spellings for these words, trying different variations until you get to the right one. You put these stickers together on the sheet and make pictures, stories, or anything you want. You are limited only by your own imagination.

In this scenario, your motivation to spell the word is driven towards the goal of getting the sticker (reward), which you will then use to make your own picture (project). This funneling helps the child get the sticker they want, while engaging them in spelling-related activities in order to do so.

This design and interaction forms the heart of our app - PictureBlocks.

With the surge in easier access to icons and illustrations online, PictureBlocks indeed has such a sticker box, with over 1,700 child-friendly icons. But, rather than children just selecting and dragging whichever ones they want, PictureBlocks helps them discover this sticker/icon by typing its word out instead.

In the smartphone app equivalent of this classroom scenario, a "keyboard" becomes the teacher, and checks if your word is correctly spelled. A "sprite icon" becomes the sticker reward received on correctly typing a word onto the keyboard. Finally, a blank "canvas" acts as the sheet where sprites can be placed and moved around in order to create pictures (Fig 3.1). In the next section, we present the PictureBlocks app and expound on these primary components.

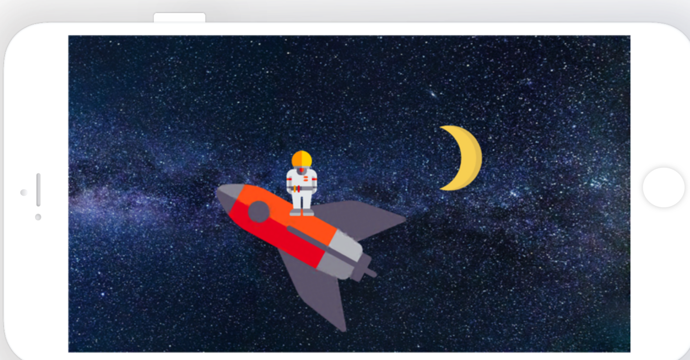


Figure 3.1: An example of sprites arranged together on a canvas, composing a picture.

3.2 *Presenting PictureBlocks*

The goal for children using PictureBlocks is to create and save pictures, which can be shared with friends within the app.

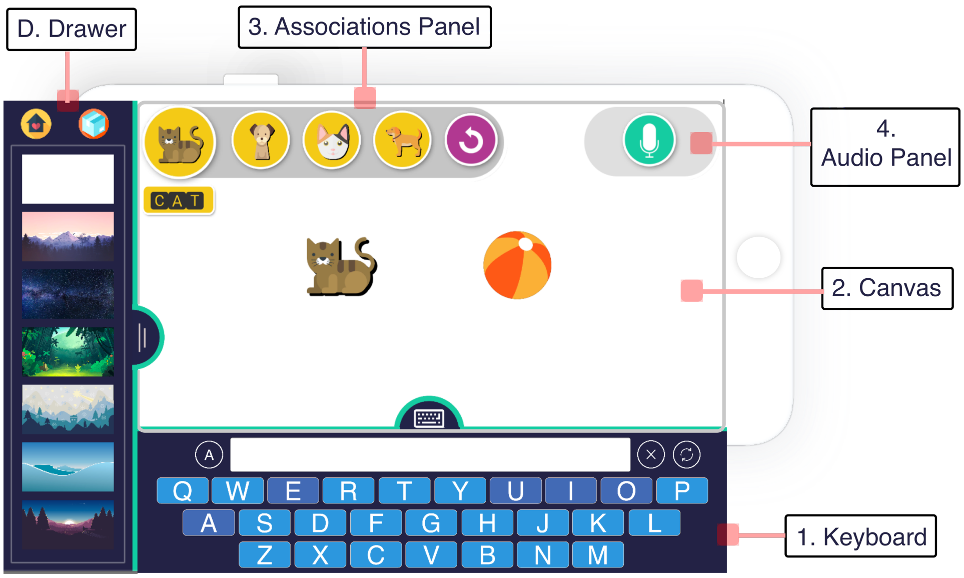


Figure 3.2: Main Game Area - PictureBlocks

Creating Pictures:

Figure 3.2 displays the main game area within PictureBlocks, and its salient features, that help with picture creation. Here, the Keyboard (1) and the Drawer (D) are retractable and can be closed and opened by tapping on their handles respectively. The 'cat' and 'ball' icons are sprites on the Canvas (2). The Associations Panel (3) and Audio Panel (4) toggle their visibility when a sprite is tapped or untapped.

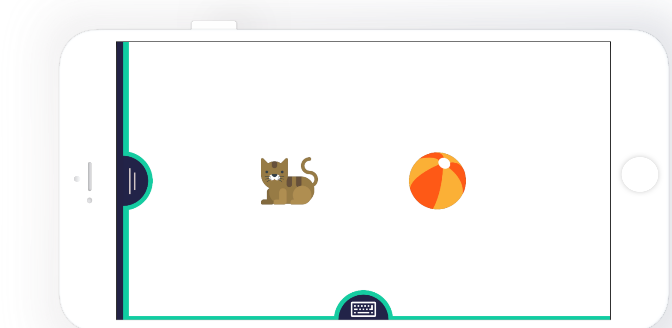


Figure 3.3: Retracted Game Area - PictureBlocks

Feature 1. Keyboard

There's a substantial amount of research on how the uncertain and the unpredictable intrigues and engages our minds, especially in the field of game design. [62] [1]

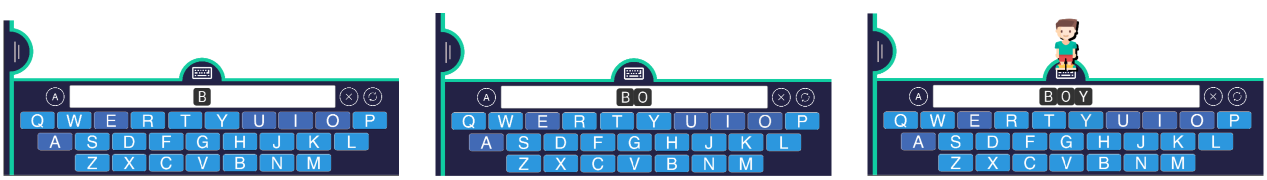
PictureBlocks contains an imageable vocabulary of around 1,700 words and corresponding sprite icons. However, none of these sprites are visibly accessible (drag-and-drop) or known to the player. The only way a player can discover a sprite within the app is by typing its corresponding word out using the Keyboard (Fig 3.2 (1)).

If the player types in a word that is valid and exists within the app's imageable vocabulary, a hovering sprite icon immediately appears on top of the keyboard handle. An important point to note is that this sprite appears only upon typing a correctly spelled word. Typing incorrectly spelled words displays nothing. Figure 3.4 shows a letter-by-letter play on creating an example sprite.

PictureBlocks also makes use of the phone's speech synthesizer for artificially speaking out the word being typed. This means that every time the player enters or deletes a letter from a word, the app speaks the resulting word out loud. This is to provide auditory feedback to the player so that they can cross-verify whether the word that they are trying to type sounds right.

Sometimes, a player may type in a word that is correctly spelled but the app might not have a corresponding icon for it. In such a case, a sad emoticon appears on top of the keyboard handle to indicate the same.

[1] Wendy Despain, author of 100 Principles of Game Design, writes "If the player knows exactly what reward they are going to get for a specific action, it removes any surprise from the equation [...] the experience leaves [the reward] feeling flat and unengaging".



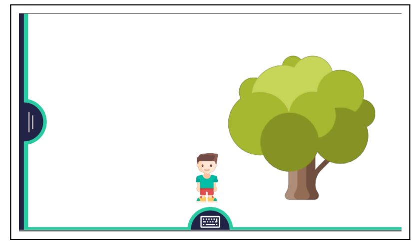
(1) (2) (3)

Figure 3.4: Sprite Creation using the Keyboard

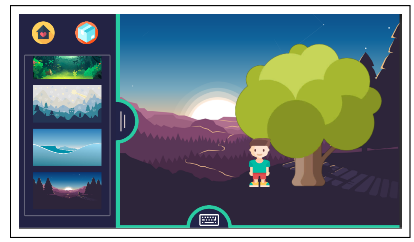
Feature 2. Picture Canvas

The Picture Canvas is the blank area in Figure 3.2 (2). It's a space for players where they can play with the sprites they create, and arrange/compose them together in order to form a picture (Picture Composition). The purpose of this feature is to provide a medium for children's self-expression and creativity.

Sprites created by typing words into the keyboard can be dragged with a finger onto this canvas area. These sprites can be scaled and rotated and overlapped with other sprites using other finger gestures. (Fig 3.5 (1)) Once a sprite is on the canvas, players can tap it to hear its word read out loud. The Drawer (D) provides a variety of images which can be used as picture backgrounds and fill up the canvas. (Fig 3.5 (2))



(1) Scaling the "tree" sprite to be bigger in size.



(2) Adding a background to the canvas.

Figure 3.5: Picture Composition

Feature 3. Associations Panel

Word Associations are words that arise in one's mind as a reaction to hearing a certain word. They can arise due to reasons such as word similarity, contrast or contiguity [8]. Table 3.1 shows a sample word and its associations.

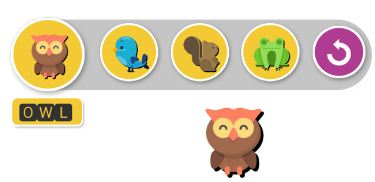
Word	Associations
restaurant	chef
	waiter
	dinner
	favorite
	fancy

Table 3.1: Example: Word Associations

Associations may reflect significant relations between objects/concepts in the real world, and therefore, are important in the field of psycholinguistics. Previous research also links the visual representation of a word with its associations by suggesting that the imageability of a word exerts a strong influence on word association. [19]

The vocabulary chosen for PictureBlocks is highly imageable. Ergo, with the hope of reflecting these existing structures/links between words - we were motivated to display word associations within the app. This feature is represented in Figure 3.2 (3) as the Associations Panel.

Within PictureBlocks, we display word associations in the form of sprite suggestions. When players create a new sprite, they are provided with a list of sprites that are related to it, rather than the words themselves.



The Associations Panel serves two purposes:

- For children to view and explore existing relationships between words/sprites.
- For children to discover what other sprites exist within the app.

We detail how we can explore word associations and their resulting network with the help of the Associations Panel in a later section (Section 3.3.4)

Feature 4. Audio Recording Panel

In the process of drawing, it is natural for young children to 'tell' a story through their work. [74] Often times, children also talk while drawing; and listening to their use of language helps understand the ideas on which the drawing is based. [10] [14] Previous work in this genre has also suggested opportunities for hybrid voice-visual tools that support children's emergent literacy. [55]

Therefore, we include an audio recording feature represented in the Audio Panel (Figure 3.2 - (4)) within PictureBlocks to allow for talking while creating, and thereby, help capture voice-visual play and narrative.

(a) Regular sprite

(b) Sprite with audio

Figure 3.6: Sprite Distinction

With this feature, players can record their own voices into the app while creating pictures on the canvas. Each sprite on the canvas can have an individual audio recording attached to it. Players can tap on sprites and click the record button, followed by their voice message. In order for the player to distinguish between sprites without audio vs sprites with audio, they are visually represented with differently colored shadows (Fig 3.6).

The length of each recording is limited to 10 seconds. These audio recordings can be deleted and re-recorded. If a sprite contains audio recording, then the player can simple tap on it every time they want to hear it. The addition of a voice recording persuades the need for the player's picture and voice to be shared and heard, leading us to our next feature - Social Sharing.

Saving and Sharing Pictures:

Feature 5. Social Sharing

Child-driven learning opportunities are even more favorable when they involve social collaboration. [6] [45]

Moreover, social interaction is an integral part of the process when children create pictures. As soon as a child completes making a picture, they usually want to show it to an adult or a peer. Children's drawings make their way onto refrigerators in homes, bulletin boards in classrooms and even newspapers sections.

Social interaction is key in both the processes of literacy development as well as picture creation amongst children. Therefore, in order to correctly design a picture-based, constructionist approach to literacy learning - this forms the final crucial feature of our app.

This Social Sharing feature fosters a mediated form of social interaction by allowing players to share their creations with friends within the app. Figure 3.7 depicts an area within the app (which we refer to as a Picture Dashboard) where the player's pictures are displayed and can be individually shared.

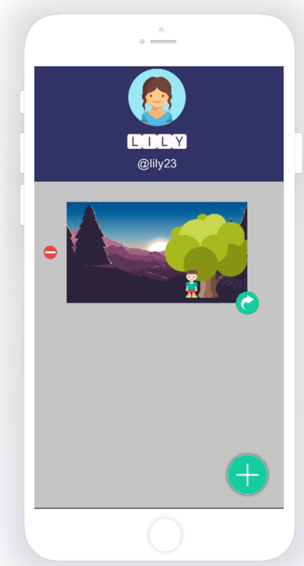
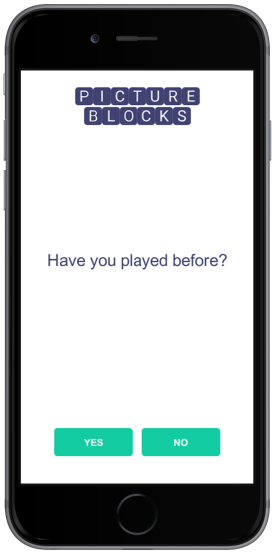
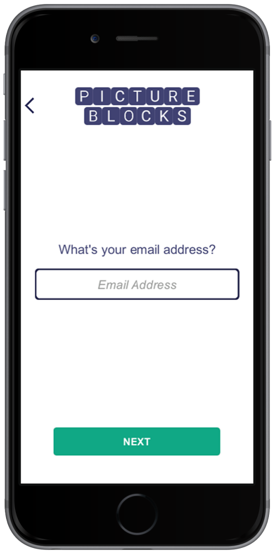


Figure 3.7: Picture Dashboard - Save and Share

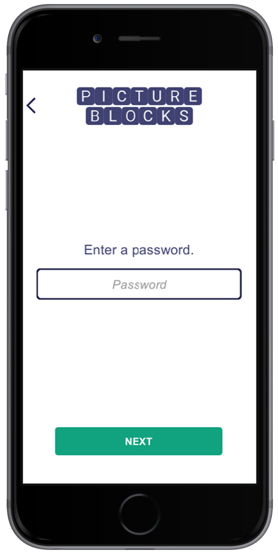
This form of picture sharing preserves information - as a child receiving this picture will not only get the image file of the picture shared, but can also open this picture on their own canvas to explore or modify its contents. We describe the Social Sharing feature in more detail in Section 3.3.3.



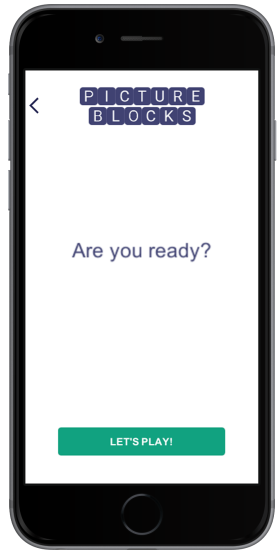
(1)



(2)



(3)



(4)

Figure 3.8: Parent Onboarding

3.3 *Detailed Screen Flows*

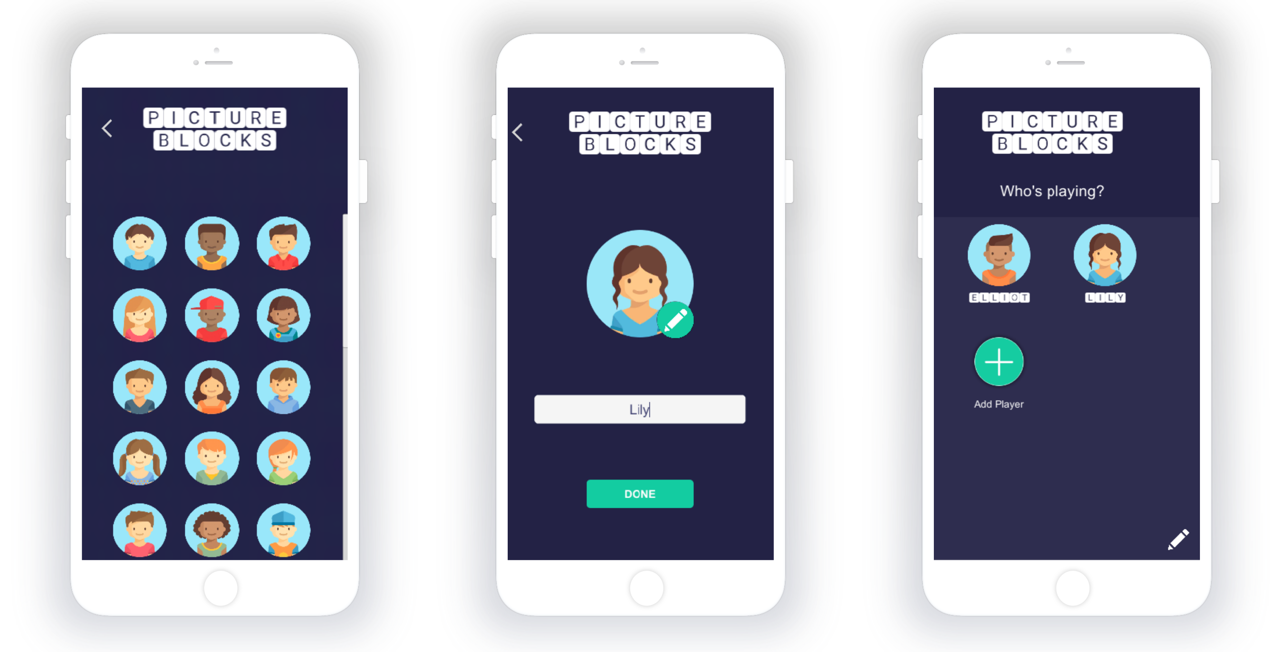
In this section, we provide a sequential screen by screen view into how the PictureBlocks app works.

1. *Parent Onboarding*

   On app download, parents go through a series of onboarding screens in order to register for the app. We request an email address and password that can be used to later login to the app. The inclusion of this registration feature is to ensure that in the case of accidental deletion or app update, children's picture data can be easily restored from a server by logging back in to the app. Figure 3.8 displays the sequence of screens in the parent onboarding process.

2. *Profiles*

   Post onboarding , this is the first screen the player is directed to each time the app is opened. In this screen, players can create a profile for themselves by selecting an avatar and typing in an username. Figure 3.9 displays a simple overview of how a player creates their profile within the app. Profiles can also be edited and deleted.



(1) Choosing an avatar

(2) Creating a profile

(3) Select Profile

Figure 3.9: Profile Creation

Each profile can access a unique instance of PictureBlocks. This is because siblings sharing a parent's phone may want to express their creativity in unique ways and might also be at different literacy levels. Therefore, they require their own profiles within the app.

3. *Dashboard*

On selecting their profile, the player is taken to their personal picture dashboard. The purpose of this screen is two-fold:

- Display a list of all the pictures they've made using the app, with most recent ones displayed first.
- Share the pictures they've made with friends within the app.

Initially, this screen is empty . Once a picture is created, it shows up on the player's dashboard (Fig 3.10). Tapping on their avatar (1) takes the player back to the Profiles screen.

To create a new picture, the player taps on the + button (2). To open an existing picture, the player taps on the picture itself. To delete a picture, the player taps on the delete button on the left of each picture (3). The player can tap the green share button at the corner of each picture (4) to open the share window.

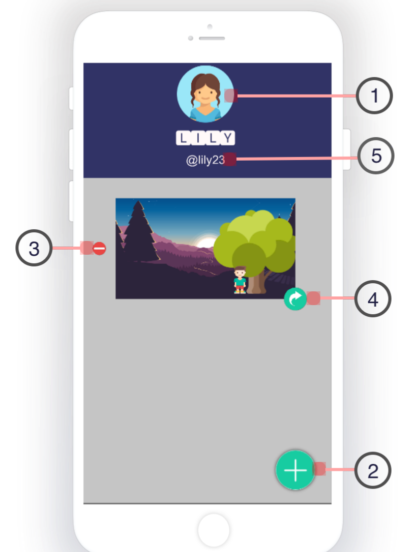
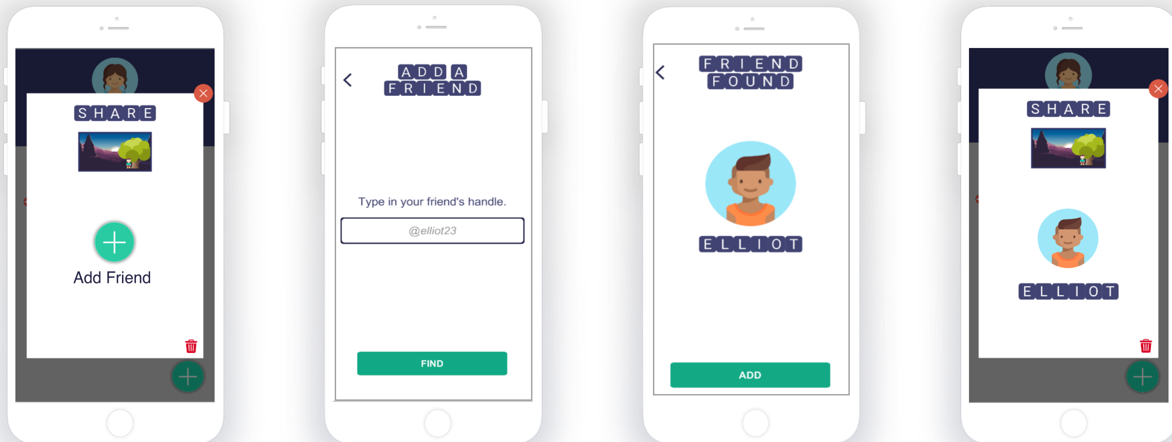


Figure 3.10: Sender Dashboard



(1) No current friends. (2) Find friend using handle. (3) Friend found. (4) Friend added.

Figure 3.11: Friend Addition

Each player gets a unique player handle displayed below their avatar (Fig 3.10 - (5)). This handle is used as a look-up to find and add friends within the app. On opening the share window, the above Figure 3.11 displays the screen flow to find and add a new friend using a player handle. In this example, Lily finds her friend Elliot using his player handle '@elliot22'. Lily can send Elliot her created picture by simply tapping on his avatar - upon which the app plays a 'whoosh' sound to indicate feedback.

Figure 3.12 displays Elliot's dashboard where he has received Lily's created picture. The received picture displays the name and avatar of the sender in a panel on top.

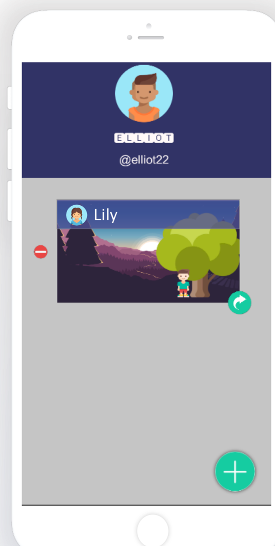


Figure 3.12: Receiver Dashboard

4. *Canvas*

This is the main screen in the app's interface. We previously provided a brief description of its components in Section 3.2.

Drawer

In Figure 3.2 , one of the components of the main screen is the retractable Drawer (D). In addition to providing a list of background images to select from, the Drawer serves multiple purposes.

The below Figure 3.13 displays two different states of the Drawer. The 'Save' button on the top left saves the current picture, and directs the player back home to the previous screen (Dashboard).

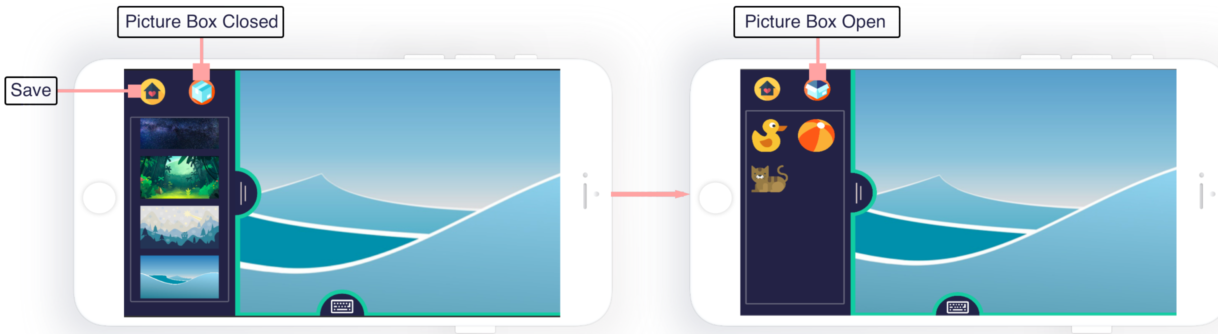


Figure 3.13: Picture Box Toggle

The drawer also includes the 'Picture Box' button which is used to toggle between opening and closing a picture box. This box is for players to save their favorite sprites or those sprites they want to create multiple copies of. For example, if the player wants to create twenty ducks in a row within their picture, it would be very tedious to type out the word 'duck' twenty times into the keyboard.

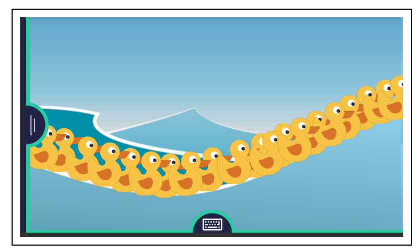


Figure 3.14: Saved Sprite Duplication

Instead, the Picture Box allows players to create multiple copies of a sprite by simply tapping on a saved sprite inside the box (Fig 3.14). On tapping a saved sprite, the keyboard automatically opens with the sprite created. An unlimited number of sprite copies can be tapped into creation. However, we designed the Picture Box to only hold a maximum of eight sprites.

To save a sprite to the Picture Box, players must drag and drop them from the canvas and into the drawer. A sprite can be removed from the Picture Box by long pressing on it and tapping the delete button that materializes (Fig 3.15).

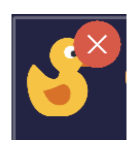


Figure 3.15: Saved Sprite Deletion

Semantic Network Exploration

When the player taps a sprite on the canvas area, two components appear on top - an Associations Panel and an Audio Panel (Fig 3.16). The Associations Panel displays word associations for the tapped sprite, introducing players to the semantic relations between them.

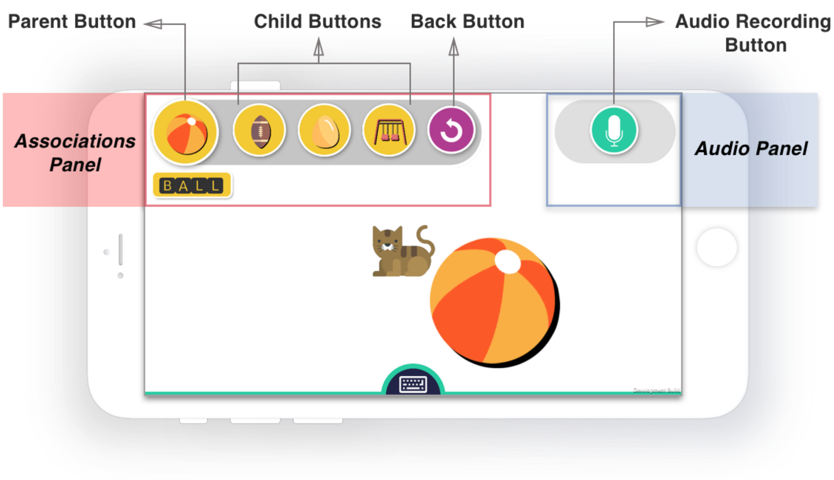


Figure 3.16: Canvas View Upon Tapping a Sprite

For the sake of convenience, we refer to the tapped sprite as a parent node, and its associations as child nodes. Each of these child nodes also have their own set of associations (grandchildren nodes). If a player taps on one of these child nodes, the panel now displays this child node's children, and so forth. In this manner, it is possible for a player to traverse through a semantic network of words. Figure 3.16 displays the Associations Panel for the tapped sprite - "ball" (parent node). The associations for ball are "baseball", "egg", and "swing" (child nodes). In figure 3.17, we present an example of how one explores the resulting semantic network:

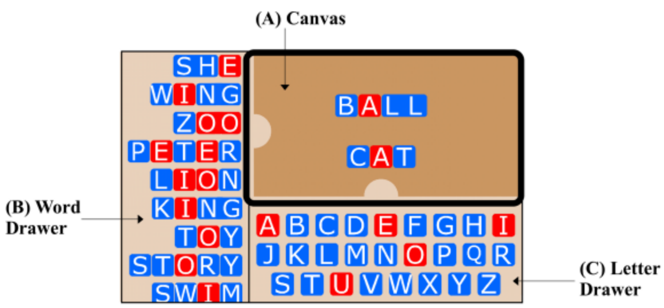
- **Action 1:** The player taps on the child button displaying an "egg" sprite. Immediately, the Associations Panel changes. The "egg" sprite has now replaced the parent node, and the child nodes are replaced with the word associations for "egg". These new child nodes are "eggs", "milk", and "chicken".
- **Action 2:** Next, the player taps on the "chicken" child button. The panel changes again to reflect this action. The "chicken" sprite now becomes the parent node, and the child nodes are replaced by a new set of word associations - "turkey", "pork", and "steak".
- **Actions 3,4:** The rest of the steps follow a similar pattern. The player taps on the "steak" button, and then on its "bacon" child button - with each tapped node becoming the new parent. The child nodes also get replaced by the new parent's associations.

In this example, the player has traversed a semantic network of words starting from the word "ball" and ended up at the word "bacon", while also being exposed to other sprites within the app during this process.

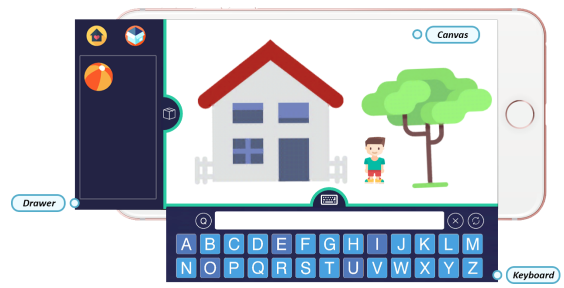
Figure 3.17: Exploration of a semantic network using the Associations Panel.

3.4 *Data and Design*

In this section, we explore some design inspirations for PictureBlocks, as well as decisions driven the data and design within the app.



(a) SpeechBlocks Interface Components



(b) PictureBlocks Interface Components

Figure 3.18: Component Design Inspiration

Design Inspirations

SpeechBlocks [68][69] serves as the original source of design inspiration for PictureBlocks. As seen in Figure 3.18, the play area for SpeechBlocks is divided into three areas - a word canvas, a letter drawer, and a word drawer. We follow a comparable separation of the interface components within PictureBlocks in the form of a picture canvas, drawer, and a keyboard respectively.

Similar to SpeechBlocks, PictureBlocks uses a speech synthesizer to read out the word while it is being created. It also depicts uppercase alphabets in the form of square letter blocks, distinguishing between vowel and consonant blocks using different colors.

We refer to SpeechBlocks' reasoning on using uppercase letters since they are more readily fitted on a square block [68]. Furthermore, research on invented spelling has shown that vowels have been shown to play a special role in writing acquisition [[60]], justifying the distinction between consonant and vowel blocks.

In SpeechBlocks, the primary interactions focus on phonological awareness by dragging and playing with movable letter blocks. Whereas in PictureBlocks, the focus is on Picture Composition and the letter drawer is modified to be a keyboard - with the letters fixed.

This transition from a letter drawer to a keyboard was done for two reasons: (1) in order to be easier to relate to a regular mobile phone keyboard. (2) to avoid crowding the canvas space by using both movable letter blocks and sprites. Finally, our aforementioned related apps and games (Section 2.3) also include interactions that serve as design inspirations for PictureBlocks.

Data Considerations

1. Vocabulary

   The vocabulary used within PictureBlocks is an important feature of the app's learning design. For the scope of this research - we only include nouns within this vocabulary because they are easier to visualize using static images, compared to verbs such as 'running' or 'drinking' - which would require animations.

   It was also critical to ensure the words were appropriate for children in the app's age range. Therefore, a good starting point for our vocabulary was to refer to an existing children's picture dictionary. For App V1, we constructed a vocabulary with 500 words chosen from the Longman Children's Picture Dictionary [32]. For App V2, we increased the vocabulary size to 1,711 words. These included a combination of the initial set of vocabulary words (500) and words scraped from the FlatIcon[2] website, further curated by a researcher. To implement this vocabulary, we built the Icon Scraper and Selector Tool (Section 3.5).

   Additional vocabulary lists were also used - (1) a 10k list of popular Google words [17] to identify real words which weren't in the imageable vocabulary. (2) a swearwords list to ensure that any swearwords typed into the keyboard would be prevented from being read aloud by the speech synthesizer.

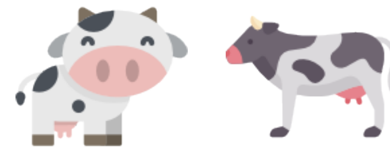
2. Icons

   For PictureBlocks, we rely on open-source images retrieved from the FlatIcon website to use within the app (more in Section 3.5 - Icon Scraper and Selector Tool). We assessed Google Images to search and filter for word-based icon retrieval dynamically. However, we discarded this approach since there was a wide variation in the image styles and required integrating advanced machine learning methods to determine similarity in artistic styles.

[2] FlatIcon describes themselves as "..the largest search engine of free icons in the world." - `https://www.flaticon.com/`

Icons were chosen keeping the following common characteristics in mind:

- They are multi-colored.
- They all have dimensions of 128x128.
- They follow a flat design (2D over 3D).
- Their style is more cute/cartoonish than realistic. For example, in Figure 2 we would select icon (a) over icon (b) for use within PictureBlocks.



(a) (b)

Figure 3.19: Sprite Icons Style

Background images for PictureBlocks are open-sourced vector graphics and illustrations carefully searched for and handpicked from different web sources. These were selected based on their variation in themes and overall adherence to app design.

3. Word Associations

For an initial set of word associations, we utilized the Nelson Norms dataset - the largest available dataset of free association norms. This contains association pairs for 5,019 words obtained by interviewing 6,000 participants. [48]. This dataset was further filtered only for those words that were within our app's vocabulary.

However, as the size of our vocabulary grew from 500 words to 1,711 words, we were limited by the vocabulary within the Nelson Norms dataset as our primary source of word associations.

Recent advances in computational linguistics have resulted in the availability of various tools that captures relationships between words (such as semantic similarity) [42] [52]. In order to accommodate a growing vocabulary, we halted use of the Nelson Norms dataset and instead, relied on these tools to help dynamically generate associations given a particular word.

Therefore, the set of word associations for our final vocabulary were obtained by creating a machine learning model as described in Section 3.5 - Word Associations Generator.

Design Iterations

PictureBlocks went through two iterations - Version 1 and Version 2. Our initial ideas for PictureBlocks were sketched out on paper, and went through several iterations and feasibility assessments keeping the time frame of the research in mind. Next, a prototype of the app was designed resulting in PictureBlocks V1.

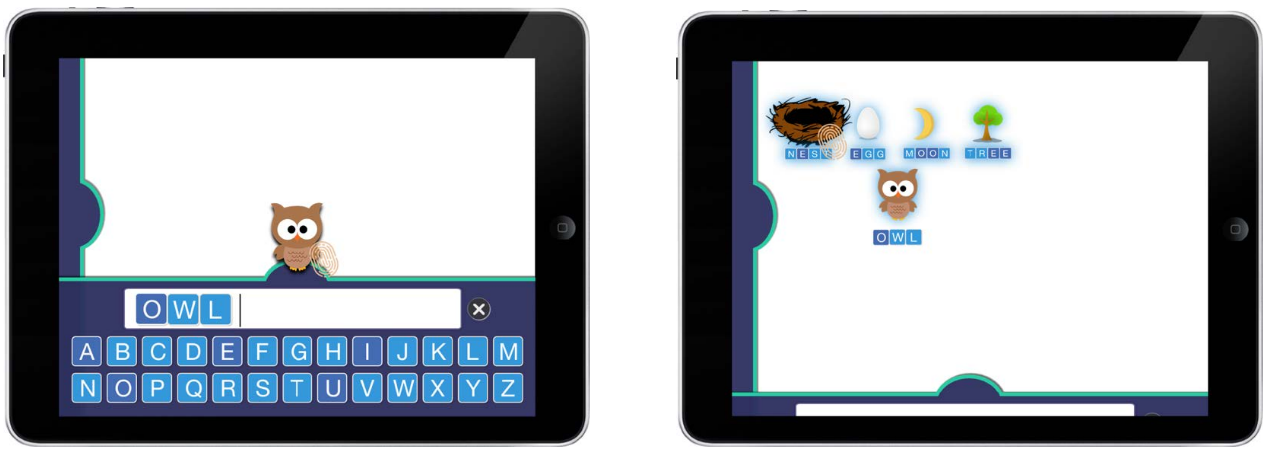


Figure 3.20: PictureBlocks v1 design prototypes

After several rounds of formal and informal playtesting with children, the app went through design changes, with new features being added and iterated upon (more detail in Table 4.2). This was based on the researcher's observations, direct feedback from players, and challenges that arose during these sessions. This resulted in PictureBlocks V2.

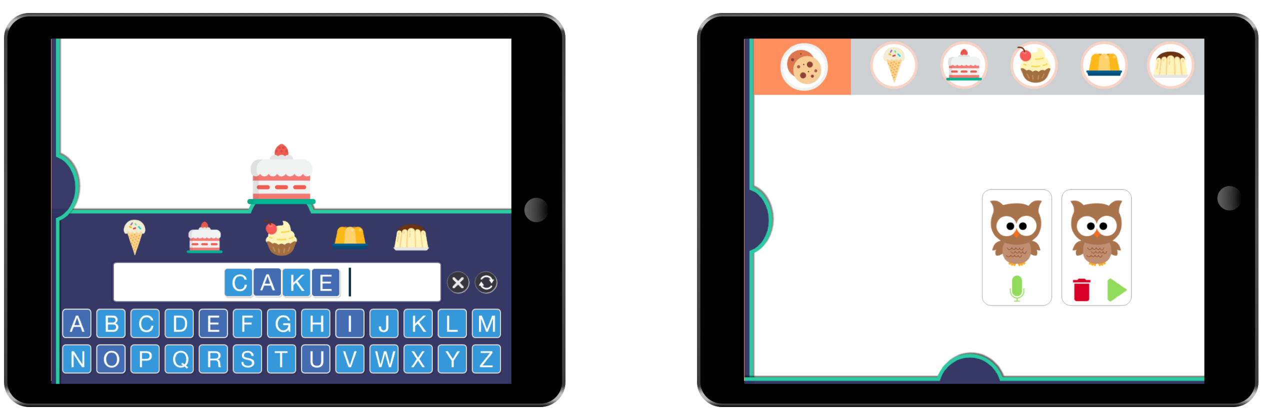


Figure 3.21: Design prototypes for PB v2 - Audio Recording Panel and Associations Panel.

Design Considerations and Decisions

This section explains some of the items that required careful consideration, and how we handled them.

Data Sharing Between Children

The Social Sharing feature involved children sharing picture with one another within PictureBlocks. This feature warranted cautious design considerations and came with its own set of questions:

1. How were we going to ensure privacy and safety? What kind of data could we display about friends?

   Children are assigned a unique player handle, displayed below their avatar and name on the Dashboard (Screen 3). One child could add another child as a friend only by knowing their player handle.

2. How can we simplify the process of adding friends and sharing pictures for such young users, who are not used to an online social network/sharing things on a regular basis?

   To make sharing easier to understand for young users, we designed a simple user interface with only two taps:

   - First, the player opens the sharing window displaying the friends' avatars with a single tap.
   - Next, the player taps on the avatar of the friend they want to send the picture to. The picture is sent!

3. What shared words, pictures or audio did we have to be wary of?

   The words within PictureBlocks were quality-checked via code and the respective picture icons were carefully handpicked prior to the experiment in order to ensure that they were safe for sharing. However, real-time voice recordings made by children were harder to validate with code.

   Therefore, we used a human in the loop to moderate the data being shared and ensure that inappropriate or abusive audio files were not being shared between children. For this purpose, we built the Audio Moderation Tool (Section 3.5).

QWERTY vs ABC Keyboard

We realized that by transitioning from a letter drawer to a keyboard, we had to choose between an "ABC" or a "QWERTY" keyboard. Since we were testing with a wide age range of kids (5-9 years), "QWERTY" could be confusing and time-consuming to use for younger children who were still only familiar with the sequential alphabet. However, for older children who were already familiar with typing or using apps and games, "QWERTY" would be a more appropriate option.

Eventually, we decided to design the app with both variations of the keyboard included. The button highlighted in Figure 3.22 is used to flip between the "ABC" and the "QWERTY" keyboard and children can use whichever one they feel comfortable with.

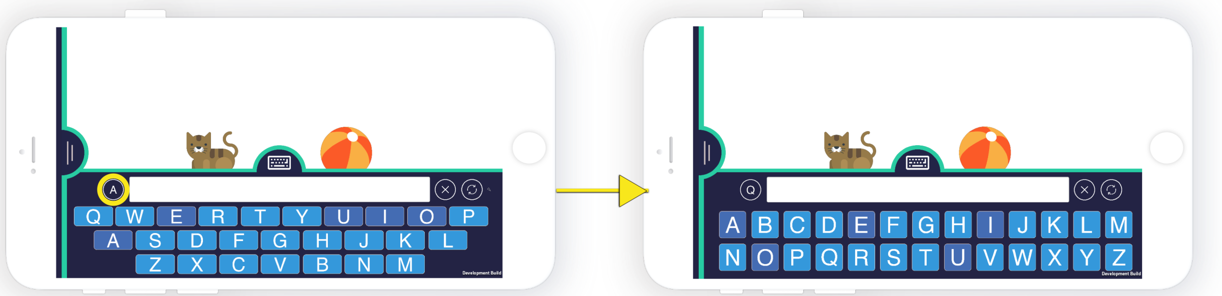


Figure 3.22: Flipping between "ABC" and "QWERTY" keyboards

Accounting for Homonyms

Homonyms are words with more than one meaning, and have different visual representations. To handle these words, we included a 'refresh' button on the keyboard (highlighted in Figure 3.23) which is used to loop between different sprites for the same homonym word.

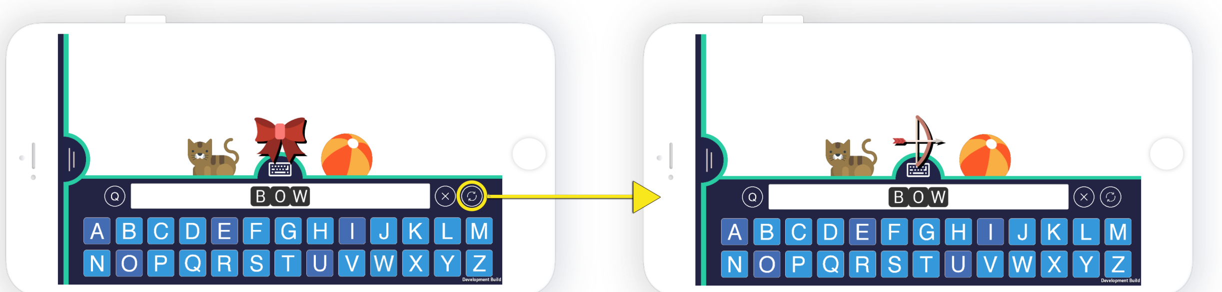


Figure 3.23: Cycling between Homonym Sprites

Picture Composition Vs. Drawing

While drawing (in conjunction with writing) has been proven to have a large set of advantages towards literacy [4] [11], it was harder to design our word-picture interactions around it. Moreover, if we compare drawings to well-defined and artist designed sprite icons, younger children have primitive drawing skills making it harder for the app to be as visually engaging. We believe that in future versions, drawing also needs to be integrated into the app and can contribute towards a more personalized experience for the child.

Representing Word Associations

In PictureBlocks V1, we represented word associations as smaller sprites that hovered over a tapped parent sprite. (Fig 3.24). The decision to display the associations (child sprites) hovering over the tapped sprite (parent sprite) was done to reinforce/highlight their relationship by decreasing the distance between them. User interface design principles state that spatial relationships are understood more quickly. [13]

Figure 3.24: Hovering word associations in PictureBlocks V1

On tapping the associated sprites, the corresponding word for that sprite would then be displayed. These associated sprites disappeared every time the player touched any other point on the screen.

However, during playtesting, we encountered the following challenges with this design:

1. **Crowded Canvas**: The hovering sprites cluttered the canvas since they appeared in various locations and in random numbers. Moreover, despite the presence of a shadow, there wasn't a noticeable distinction between associated sprites and the canvas sprites, further contributing to the clutter.

2. **Slowed Semantic Exploration**: During playtesting PB V1, children would need to tap a sprite to view its associations, then tap the associated sprite to view its word. After this, they would type the word association into the keyboard, sometimes letter by letter. They would have to repeat this process several times to explore semantic relationships between words, or find a word that they liked. This process was too slow, and there was a high chance that children would get distracted or forget the previously encountered words by the time they typed out a new word.

Therefore, in PictureBlocks V2, in order to de-clutter the crowded canvas, we moved all the associated sprites into a separate panel instead. This "Associations Panel" displays associations in the form of circle shaped buttons - which help them be better distinguishable than the sprites on canvas. We also reduced the number of options available to a fixed amount - three associations only. [3] The parent sprite is replicated and is also a part of the Associations Panel. By doing so, we still maintain the visually spatial relationship between the parent and its child sprites.

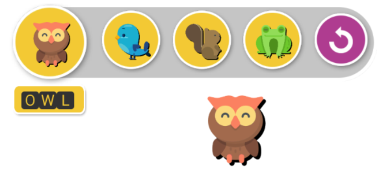


Figure 3.25: Word associations moved to a new panel.

[3] User interface guidelines specify to display a small number of navigational items (approx. 5) [13]. With the three child sprite buttons, single parent sprite button, and the single back button - we provide a total of five options.

Additionally, in order to allow for a faster exploration of words and their relationships, we decided to open up access to the entire semantic network using the Associations Panel. This allows the player to now traverse up and down the pathways between word associations and are not limited to exploring them one level at a time. Figures 3.21 and 3.26 show several iterations in the design for Associations Panel.

Figure 3.26: Design Iterations - Associations Panel

3.5 *Software Development*

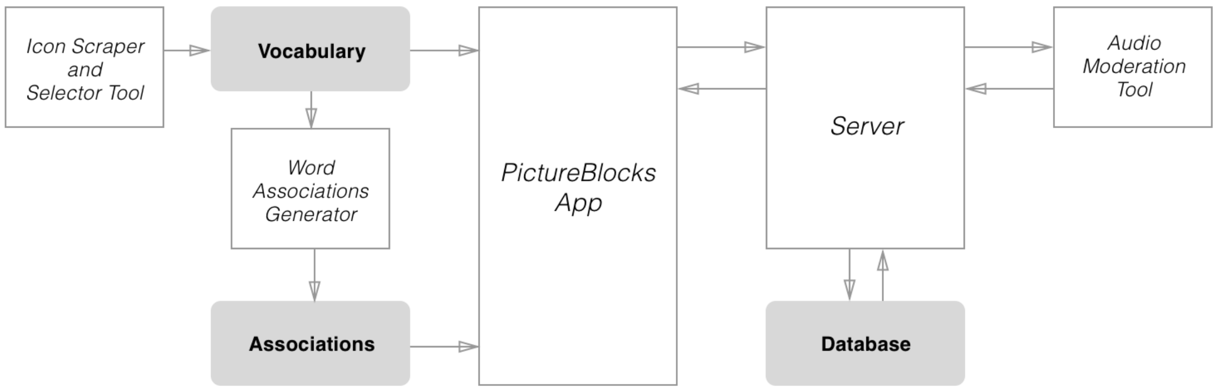


Figure 3.27: Software Architecture

The above figure (Fig 3.27) provides a brief overview of the software system architecture developed during this research. At the heart of this system is the PictureBlocks app developed using the Unity Game Engine, with other auxiliary software tools surrounding it that help support PictureBlocks. The following subsections explain each of the building blocks of this architecture in further detail.

Icon Scraper and Selector Tool

In the previous section, we discussed the construction of an imageable vocabulary with words and images to be used within PictureBlocks. While the initial playtesting version (V1) contained around 500 handpicked words and images, in order to construct a larger vocabulary, we had to speed up this process and bring more automation into place. However, there were several challenges that we encountered while doing so.

There was no standardized online children's picture dictionary since every website had different categories and structures of their own. Sometimes, icons for these words were not available on the FlatIcon website, or modern words (with icons available on FlatIcon) didn't make an appearance in these dictionaries. We evaluated several solutions towards this challenge and finally, the method we opted for was a reverse look-up. Instead of looking for the corresponding icons after we had constructed a vocabulary, we decided to go through all

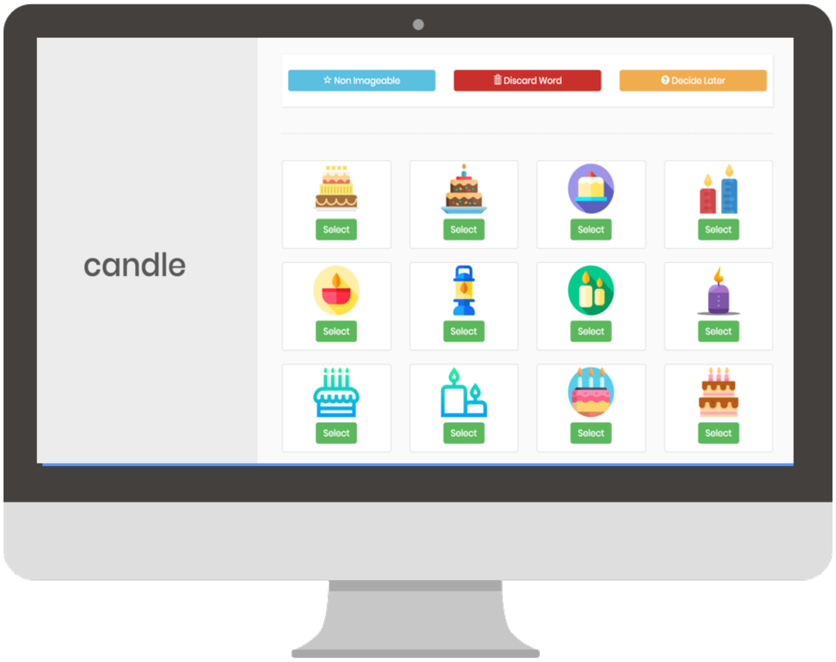


Figure 3.28: The Icon Scraper and Selector Tool. *For example: evaluation of the word 'candle' and corresponding icons.*

the icons available first and use it construct our vocabulary. Our job was made easier since FlatIcon website icons were already tagged with a word and a category.

FlatIcon is one of the largest available collection of free icons on the internet. This required ensuring that the website was scraped efficiently for icons. For this purpose, we built the Icon Scraper and Selector Tool using Flask and Python. It's intended uses were threefold - to help construct an imageable vocabulary using words on the website, to automate the icon download process, and to moderate every word and icon being downloaded to ensure it was sensitive to children's perusal. Initially, we wrote a script using Python and BeautifulSoup to scrape the FlatIcon website markup. We went through every icon amongst their available categories, and on encountering a unique word, we added it to a set. A final vocabulary of words was constructed by merging this set with our existing set of 500 vocabulary words. Finally, we obtained a list of color icon Image URLs (max of 20) for the words in this vocabulary and dumped these values into a JSON file.

The application front end for the Icon Scraper and Selector Tool is displayed in the figure. The tool iterates over every entry in our JSON file one at a time - displaying the word on the left, and the images from the corresponding URLs displayed in a grid view on the right.

The researcher clicks on the Select button below an image to match it with the word, and it automatically downloads the image into a folder. If a word is confusing or non imageable, we have three buttons on top which mark the word to be non-imageable (but a real word), discardable, or ambiguous and to be revisited later. When the researcher selects an image or clicks on any one of these buttons, each word is saved to a text file (imageable, non-imageable, discarded, and revisit) and the web page refreshes to show the next word in the JSON array.

We evaluated a total of approximately 13,500 words from the website and the process was fast and efficient since it involved clicking on only one button per word. (taking around a total of two working days for one researcher). We went through several iterations of the revisit list and paid close attention to every word and image to make it kid-friendly and suitable for our app. Our final vocabulary list consisted of a total of 1,711 words and icons.

Since our final vocabulary was comprised via manual annotation by a single researcher, we decided to validate the reliability of our vocabulary in terms of imaegability using an external source. Currently, the largest relevant research dataset is the Reilly English noun imageability dataset [56] [4] made of 2,877 nouns , each tagged with an imageability score and category.

[4] Reilly dataset ratings (2,877 nouns):

- low-img : 635
- med-img: 856
- high-img : 1385

A limitation of this is that the size of the Reilly noun vocabulary is quite small and not specific to children - therefore, we found that only 636 (out of 1,711) of PictureBlocks words were present within the Reilly word set. However, by cross-checking and calculating word imageability for the available words, we found that almost all reflected high-imageability ratings, with an average score of 575 (as seen in Fig 3.29).

Imageability Rating	Word Count
low-img	4
medium-img	48
high-img	584
Total Words:	636

Figure 3.29: Imageability ratings of a subset of PictureBlocks words - as evaluated by external dataset.

Word Associations Generator

In PictureBlocks V1, we used Nelson Norms [48] (word pairs data collected from human subjects) to generate associations for our 500-word vocabulary. For our new imageable vocabulary consisting of 1,711 words (as constructed above), we needed to find a way to dynamically generate these associations, while ensuring accuracy. Therefore, we trained a machine learning model, which would return N number of associations given a target word. A word can have certain associations based on many factors, but the most popular observation was based on semantic or contextual similarity.

There are many existing tools (at the time of this research) which are useful for such computational liguistics and NLP. To generate these associations, we use a combination of lexical data information (WordNet [42][43]) and word embeddings (GloVe [52]).

WordNet is a large English database that links words by their conceptual - semantic and lexical relations. This database is good for identifying hypernym/hyponym (super-subordinate) relationships with respect to nouns. Word embeddings are widely used in the area of distributional semantics, based on the underlying idea that "a word is characterized by the company it keeps". GloVe[5] is an unsupervised learning algorithm, which uses an explicit weighted-SVD strategy over a corpus to find word embeddings [6]. Applying this combination of methods on a target word, will therefore give us a richer set of associations encapsulating different relationships between words.

[5] GloVe differs from Word2Vec in that Word2Vec is a predictive model, whereas GloVe is a counts-based model.
[6] Another word embeddings tool is ConceptNet - a freely-available semantic network, designed to help computers understand the meanings of words that people use. It is built from a crowdsourced project that originated at the MIT Media Lab.

Word Similarity Model

For every pair of words within our app's vocabulary, we calculated two separate word similarity scores using WordNet and GloVe respectively. Our goal was to use a combination of these two scores to get a final similarity score that can be used to obtain word associations. To combine these scores, we assigned linear weights to each score, in order to find the ideal weight distribution that would result in the closest accuracy to the ground truth data.

Wordnet : We pre-parsed the WordNet database for every pair of words within our vocabulary, in order to generate a list of similarity scores. These scores were obtained by calculating the shortest distance between each pair of words in the hypernym-hyponym taxonomy graph (superordinate - subordinate relationships between words).

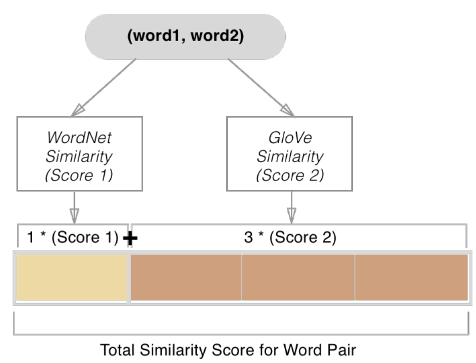


Figure 3.30: Word Similarity

GloVe : We trained a model on common crawl data from the web to generate word embeddings. This training dataset is the 'enwik9' corpus used in the Large Text Compression Benchmark, which contains the first 10^9 bytes of the English Wikipedia dump [33]. From the pre-trained word vectors thus generated, we use cosine similarity to quantify the degree to which two words are related. Therefore, this provides us with a similarity score.

The ideal weight distribution ratio for WordNet and GloVe scores respectively was found to be 1:3 (Fig 3.30).

Methodology

To obtain associations for a target word - each (target word, <word>) pair, is calculated and assigned their total word similarity score. For each target word, all such pairs are sorted in the order of decreasing word similarity scores. Finally, the highest N number of pairs are returned as word associations. In our case, we only retrieve the top five associations, hence N=5. From these five associations, we check for any mutual associations between a parent and child node (For example, (cat, dog) and (dog, cat)), and only represent the top three associations in our app's Associations Panel.

Evaluation

As ground truth, we refer to the free-association norms dataset specified earlier (Nelson Norms) [48], further trimming it to contain only the words within our vocabulary. For each target word within the ground truth dataset, we perform the following steps:

1. We obtain a set of its ground-truth word associations which could vary in number depending on the word. For example: the word 'wood' has more ground truth word associations in comparison to the word 'wool'.

2. We obtain a set of its predicted word associations. The number of predicted word associations that are obtained are chosen to be N (specified by the researcher). But to account for the variance in the number of ground truth word associations, we retrieve the Maximum(N, G), with N being the researchers depth of associations specified, and G being the number of ground-truth associations for that word. Therefore, we have an equal number of associations in both the ground-truth and predicted sets obtained.

3. We take an intersection of these two sets, in order to identify the common pairs amongst both the ground-truth and predictions.

4. Next, the accuracy of the predicted data for the target word is computed. This is done by taking the number of common associations (from Step 3) and dividing it by the total number of ground-truth associations. This tells us the percentage of the correctly predicted associations.

Finally, we average the individual accuracies across all the words in order to obtain an overall accuracy for our vocabulary. This average accuracy increases as we consider a larger number of predicted associations (N).

N word pairs	Average Accuracy
5	49.77%
50	83.56%
100	92.6%
150	95.55%
330	100%

Table 3.2: Accuracy of Predicted Word Associations

For example, if we compare the ground truth per word against only its top 5 predicted associations, we get an accuracy of 49.77%, with the accuracy leading to 100% with more word pairs considered (Table 3.2). These less than ideal accuracies maybe due to the fact that sometimes these textual relations alone do not entirely capture realistic human associations[7], or that our GloVe training dataset needs to be expanded or altered, by considering text from children's books. While our model can definitely be improved, from qualitative observations alone, we found that the relations between the words and their associations were sufficient for study inclusion.

[7] For example, the ground truth contains the pair (dolphin, football) indicative of the Miami Dolphins - an American football team. However, this pair was not captured within the predicted set of associations which relied on text trained data.

PictureBlocks App

The mobile app is the main component in our software architecture and is developed using the popular Unity Game Engine. We chose a game development engine over a traditional IDE such as Android Development Studio or Xcode since the design of PictureBlocks resembles an open-ended 2D game more than that of a regular app. The app was developed keeping both iOS and Android devices in mind utilizing Unity's cross-platform capabilities. [8] The programming language used is C#. Some external packages were coded in Java and Objective-C to interface with the Android and iOS speech synthesizers respectively. [9] An external open source library - SavWav was included to serialize and deserialize the audio recordings to a .wav format.

[8] The Unity project contains approximately 6000 lines of code excluding comments and blank lines.

[9] Development assets used:

- user interface icons
- sprite icons
- audio sounds for user experience
- SpeechBlocks font file
- kid avatar icons
- lists - swearwords, vocabulary, associations lists

The size of the app is approximately 175 MB. One of the challenges faced was controlling the size of the app primarily due to the large number of high-quality image files that it contained. First, we reduced the size of the app from around 700 MB to 350 MB by reducing the quality of the icons and choosing an appropriate quality that was neither too blurry nor too advanced for our screen size. Second, we were using two copies of each icon - one with a shadow and one without, which was doubling our size. We fixed this by writing a custom sprite shader that added shadows to sprites, and deleted all the duplicate icons with shadow from our app, thereby, reducing its size to 175 MB.

The user interface for the app has been developed to be responsive and scale to different screen resolutions of respective Android and iOS devices, including tablets. Unity allows development in two modes - we can edit the scenes using the editor tool on the front-end and/or we can create code scripts on the back-end. Usually, a combination of these two methods is used for the easiest design and development experience. Our project includes six scenes [10] - Onboarding, Login, Profiles, Friends, Dashboard, and Game, each corresponding to the respective screens that players interact with, within our app.

[10] A Scene in Unity contains the environment and menu items. It can be thought of as a unique level.

Each of our game scenes contain several different scripts [11]. For example, the PictureDrawer script handles interactions that occur within the Picture Drawer object and the PictureDrawerHandle script specifically is for identifying objects that enter and exit the Picture Drawer via its handle component. Our project also contains code that does not involve the scenes directly such as - scripts for interacting with the server, creating XML data files, creating log files, and managing screen resolutions. We save the following player data within a persistent data location in the app's folder, in order to be retrieved during gameplay as well as for data analysis:

[11] Scripts can be added to the Gameobjects within scenes in order to define variables and control their behavior.

Data	Description
Profile Folders	Each profile created on the app for each player has its own folder, with the profile IDs serving as folder names.
profiles.xml	A general XML file that contains information on the profiles that exist within the app, the player names and respectively chosen avatars.
pictures.xml	Within each profile, this XML file is present and contains information about the pictures created by the player, sprites within the pictures, and so forth.
drawer.xml	Within each profile, this XML file is present and contains information about the customized picture drawer for each player and the sprites that are saved within it.
Picture Folders	Within a profile folder, each picture created by that player in-game has its own folder, with the picture IDs serving as folder names.Deleted pictures are preserved with a deletion time timestamp appended to the folder name.
Picture Files	Within each picture folder, is the corresponding picture file in a .png format, named with the picture ID as well. The resolution of these pictures vary depending on the phone the app is installed on.
Audio Files	Within each picture folder, are the audio files recorded for sprites within that picture. These are named with the corresponding sprite ID, and saved in a .wav format. Deleted audio files are also preserved with a deleted time timestamp appended to the file name.
Log Files	Every player action is logged line by line into text files, which are named with a timestamp of file creation. A new log file is created every time the app is accessed, and the log files closes when the app is exited (correctly or incorrectly).

Table 3.3: Types of data stored within the PictureBlocks App

PictureBlocks Server and Database

Several PictureBlocks features tested during the pilot required the presence of an online server and database. For e.g.: sending and receiving pictures between kids, storing profile and app information for future logins, and finding friends. The server code offers API endpoints for the PB App and other tools to read and write to the database. The server is written in Python and hosted on a secure MIT server, whereas the database is managed with MySQL and hosted on a secure Amazon AWS server. The server code also handles the encryption and decryption of user account passwords within the app.

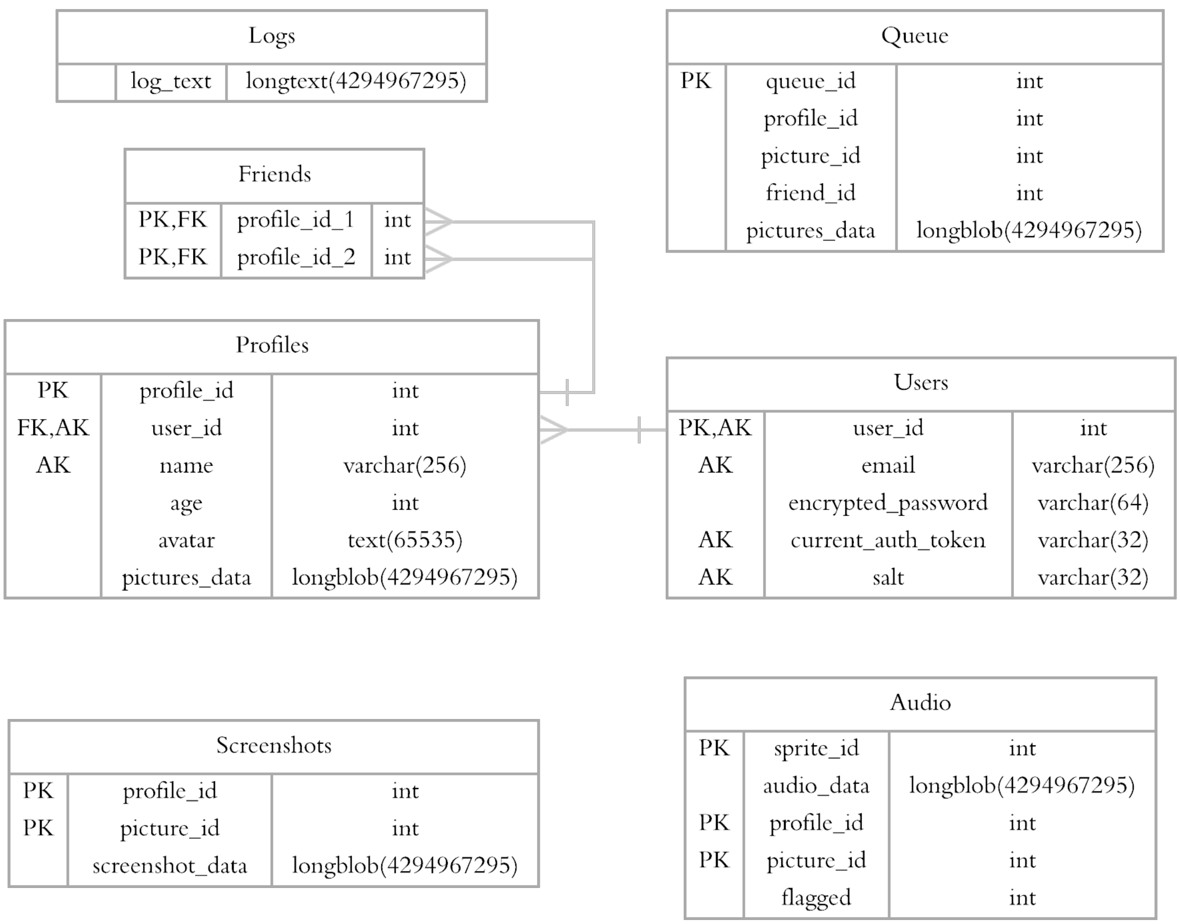
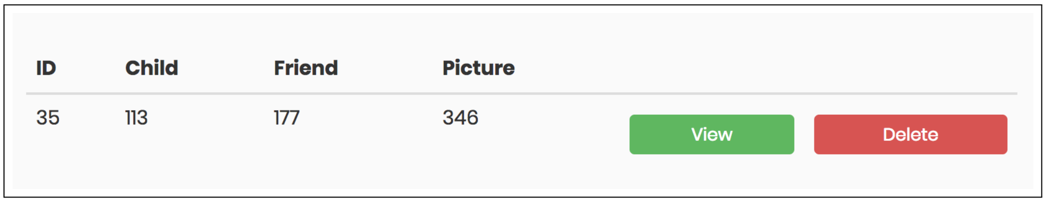


Figure 3.31: Database Schema

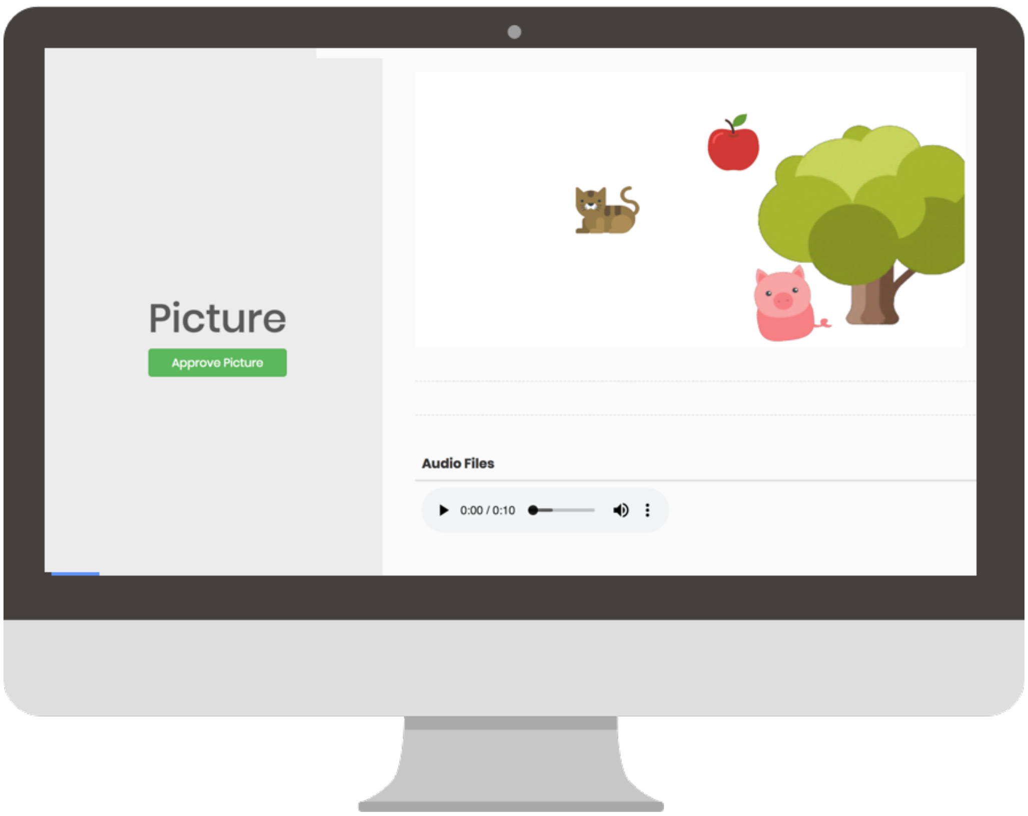
Audio Moderation Tool

Within PictureBlocks, the sharing of pictures between players raised concerns about the nature of the content that could be shared. While we alleviated most of these concerns by being conscientious in our selection of words and images that would be a part of the app - the audio recordings however, were much harder to validate using software and warranted human intervention. Therefore, we built an Audio Moderation Tool to be used by researchers, in order to oversee the audio data being shared between players.

Built using Python (backend) and Flask (frontend), this tool interfaces back and forth with the PB server in order to access the database. Every time a picture containing an audio snippet is shared from one player to another, it is inserted into a queue and an email notification is sent to the researcher to check the Audio Moderation Tool. Figure 3.32 displays this tool's front-end interface, where the researcher can view a list of queued items to be monitored. Upon selecting an item from the queue, the respective picture image and the audio snippets that need to be reviewed are obtained from the database and displayed for evaluation. If the data contains nothing inappropriate, the picture is approved and gets sent to the player it was originally meant for.



(a) Every shared picture containing audio snippets appear in a queue form for moderation.



(b) The composite picture is viewed for context and the audio components can be played individually before being approved.

Figure 3.32: Audio Moderation Tool

4
Experiments

"An empty canvas is a living wonder...
far lovelier than certain pictures."

Wassily Kandinsky

In the last chapter, we outlined how our design and development brought the PictureBlocks app to life. Iterations in the app's designs (Version 1 and Version 2) were implemented by conducting playtesting sessions at the MIT Media Lab and the Museum of Science, Boston. Finally, we ran a 15-day pilot study in children's homes, which has provided the data for the findings in this thesis.

4.1 Play Testing - Formal

A formal round of playtesting was held at the MIT Media Lab during the second week of December 2017. We had a total of 5 participants who were accompanied by their parents. The purpose of this playtesting was to solicit feedback on the user experience of PictureBlocks, as well as observe how children interacted with an initial prototype of the app. Furthermore, we were interested to check if the learning features of the app were being used as intended, and identify what new features we could introduce that would enhance the app's experience.

Gender	Age
Male	5
Male	5
Female	5
Female	5
Female	8

Table 4.1: Playtesting - Participant Details

Participants

The participants were recruited via several MIT Mailing Lists. Table 4.1 contains information about the participants. Since the design of the app at the time was open-ended, we chose a wide age range for the playtesting to see how children of different ages would engage with the app.

By opening up the app to a wider age range, the playtesting was also helping us determine the scope of the app, and the audience it was best applicable to. The participants were asked to come in an individual basis with each parent signing up for a 40 minute time slot using the YouCanBookMe website [1].

[1] YouCanBookMe is a calendar-integrated web application that helps create personalized booking pages in order for people to sign up.

Preparations

All the playtesting happened on one Apple iPhone device, with PictureBlocks pre-installed on it. This was connected to a screen recording software that would record what was happening within the app during the session. Cameras were also set up in two locations: one in front of the participant to capture their reactions, and the other placed above to observe how the children were moving their fingers on the screen (Fig 4.1). We also had corresponding consent forms and questionnaires for the playtesting approved by the MIT Internal Review Board (IRB)[2].

[2] Research involving human subjects require MIT investigators to undergo additional training. The Committee on the Use of Humans as Experimental Subjects validate all study material.

Procedures

Each playtesting session lasted no longer than 40 minutes, out of which 20 minutes was the time spent introducing PictureBlocks and playing with it. An additional 15 minutes were allotted to the feedback questionnaires with the remaining 5 minutes as buffer time.

Once they arrived, the parent and the child were placed in two rooms separated by a glass wall, so that the child could see the parent throughout the session and vice-versa. There was only one main researcher involved whose role was to introduce the app and its features, and communicate with the child during playtesting and feedback. During this time, the parents signed the consent forms.

Figure 4.1: Different camera viewpoints of a participant's play-testing with researcher.

While introducing PictureBlocks, the researcher guided the child to create their own profile and choose an avatar. The children were told that the objective of the playtesting session was to create a picture or scenery that they could think of - which would be printed for them to carry home. If the child was stuck, they were told to ask the researcher any questions, and the researcher would help guide the child on what to do next. Once the picture was created and saved, the researcher proceeded to ask feedback interview questions and finally print out the picture for the child to take home.

Data Collection

This round of playtesting utilized four types of data collection. Firstly, the cameras provided video recording that helped capture the mood of the session and participant reactions. Secondly, we had two structured questionnaires about the app and its user experience which served as a back and forth exchange between the researcher and the participant (Appendix A.1 and A.2). Thirdly, every interaction within the app was logged in the form of text files, for the purpose of recreating the gameplay (More in Section 5.1). Finally, we used screen recording software which helped us test this logging feature to ensure that everything that happened in the screen recording was reflected in the log files.

4.2 *Play Testing - Informal*

Further rounds of playtesting were conducted in collaboration with the Living Laboratory[3] within the Museum of Science, Boston. Children from birth to eight years old are welcome in this space to try out various activities and participate in studies with researchers.

[3] Living Laboratory is a research program within the Museum of Science that brings together scientists, museum educators, and visitors. Researchers can conduct studies with children who are visiting the museum.

Partnering with Museum educators, we were trained to talk with visitors about the scientific process, and had to modify our IRB forms to match the Museum's requirements. We conducted a few 3-hour long playtesting sessions for PictureBlocks in this space, between the months of February and March 2018.

We were allocated a corner within the Museum's Discovery Center space. When visitors approached us, we explained our research to the parents and confirmed if they were interested in their child (5-8 years old) participating in our research. Our interaction with each participant lasted no more than 15 minutes (as per the Living Lab's requirements). Since we were not collecting any specific information about the children and just observing their play, there were no consent forms involved and the playtesting was informal in nature.

The child was provided with instructions to use PictureBlocks and then allowed to freely play with it. Since the playtesting was in an open space, sometimes 2-3 children expressed interest in playing with the app simultaneously. In such a scenario, we provided them with individual phones in parallel and observed how they interacted amongst each other while playing. Finally, each child was given a participatory sticker.

Some of the items that were playtested during this collaboration include changes to the user experience, bug fixes, and features such as the semantic network, audio recording, and social sharing.

4.3 Observations and Iterations

Formal Playtesting

This round provided a lot of insight into the various modes of interaction within the app and unforeseen challenges. Overall, children were excited and engaged using the app, and highly enthusiastic while describing the app to their parents. The following modes of interaction were observed:

- **Story Telling**

  A participant verbally described a story around the picture while creating it. The story arc mentioned how a city was in danger due to a disaster (volcano explosion), and the superheroes would go in and rescue the city's people.

  We noticed that by looking at just the image file alone, the entire voiced narrative was not captured. This led us to examine how we could preserve the personalization within a child's picture - resulting in the addition of the Audio Recording feature.

- **Scene Setup**

  Children used the background images offered and constructed a scene around the selected background. Some children created a jungle themed scene by choosing the forest background, and then only creating animal sprites to populate it with.

- **Sprite Discovery**

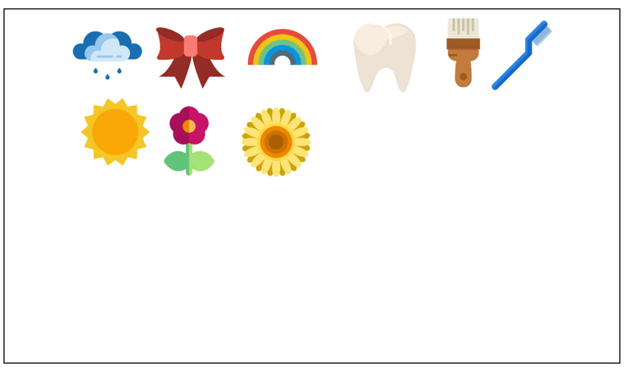
  On some occasions, the children were less concerned about the overall scene and displayed more curiosity to see what sprites the app contained. They bombarded the keyboard with all the words they knew how to spell and checked if the app contained sprite icons for the same.

- **Language Play**

  The oldest (Age 9) participant's interaction was to create compound words by using three sprites per compound word - a combination of two sprites to create a third sprite. For example: sun + flower = sunflower, or rain + bow = rainbow.

- **Favorite Characters**

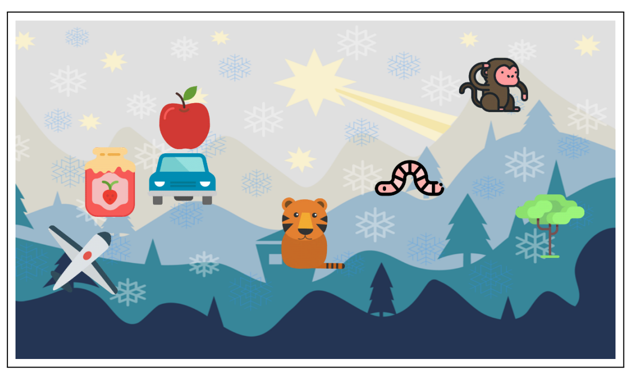
  Children who had a favorite sprite (For e.g.: a bunny), would try to make as many bunnies as possible, and were ecstatic playing with these few words alone.



(a) Language Play - Compound Words

(b) Scene Setup - Jungle

(c) Story Telling Scene - with spoken narrative



(d) Sprite Discovery

Figure 4.2: PictureBlocks Playtesting - Observed Modes of Interaction

Observation	Iteration
Few children were disappointed when the words they typed did not result in a sprite being created, especially for common ones such as "mat".	We decided to increase the app's vocabulary from 500 words to include as many words as possible, and built the Icon Scraper and Selector Tool.
There was a stark difference in the way children of different age groups or literacy levels used the app. Younger children were excited with the sprites created, whereas the older child started making compound words.	Due to this difference in gameplay, we decided to keep the wide age range (5-9 years) intact for the pilot - in case we get to see more interesting patterns amongst children who already were good at spelling.
Some children did not know where to begin with spelling a word, and needed a lot of guidance. This brought to attention the need for more scaffolding and ways to get them started within the app.	We realized that the blank canvas might be somewhat intimidating and added a cat (animal) and a ball (object) sprite for the child to start with. We evaluated and disregarded speech recognition as a form of spelling scaffolding.
The associations appearing on top of a sprite grabbed a lot of attention and were used extensively. However, they crowded the canvas and overlapped with other sprites, obstructing picture composition.	We moved the associations to a separate panel on top of the screen instead of crowding the canvas, and displayed only three instead of five associations.
Few children who had difficulty spelling , used the associations text to spell out words. This took a long time, as the associations were designed to disappear every time the keyboard was opened. Thus, the children had to keep going back and forth between tapping the sprite and opening the keyboard in order to spell.	Since the associations appeared to be helping with spelling, we decided to open up the entire semantic network to the player instead of only displaying the first level of the network. This panel was also changed to be visible when the keyboard was open.
The questionnaires were too long and the children were tired. The use of a paper form also added to their distraction as they kept looking into what the researcher was writing.	For the pilot, we changed the final interview to a semi-structured format in the form of a conversation, with the children's answers audio recorded instead of written down.

Table 4.2: Iterations to the app design based on playtesting observations.

Informal Playtesting

While our initial round of playtesting was held in a distraction-free environment, further rounds of playtesting were held at the Museum of Science, with diversions in the form of other exhibits or surrounding children. During this phase, we user tested for bugs, and iterated on designs for features such as Audio Recording and Associations Panel.

Children were very responsive to the Audio Recording feature and enjoyed adding funny noises to their pictures. Very often, they also deleted their audio recording if unsatisfied, and continued to record and play new ones while laughing along with them. Based on their interactions, we added a green shadow color to the sprites whenever they recorded audio to it. This was done in order to distinguish between sprites which included an audio snippet and those that didn't. Initially, the audio snippet could be played only by tapping the green Play button in the audio panel. Later, we modified this feature such that the audio snippet was played each time the sprite was tapped on as well.

Children also made extensive use of the Associations Panel, going back and forth between the sprites several times - which showed promise in exploring the unique affordances of this feature with a longer evaluation. Communication between siblings during playtesting also further advocated the presence of a Social Sharing feature within the app.

Several other interactions were conducted for generating feature ideas - such as the researcher pretending to be a button who would help with spelling words. In this case, the child would poke the researcher's hand and the researcher would spell the word out loud, letter by letter, which the kid would then type into PictureBlocks. While this method showed some potential for spelling based scaffolding using speech recognition, we also observed that children were less inclined to engage in critical thinking if such a button was readily available.

4.4 Pilot

We ran a small scale pilot with children to test PictureBlocks over a longer period. This study was divided into 3 parts: a pre-study meetup, followed by the study in the child's home environment, and ending with a post-study exit interview.

The study included 16 participants, with each child receiving a phone with the PictureBlocks app installed to take home for a minimum duration of 15 days. Both the pre and post-study events were conducted in a private space within the MIT Media Lab. Since the nature of the study was exploratory, no control group was present.

However, we did divide the children into groups for the pre-study meetup. This was done for the purpose of testing the Social Sharing feature among children whose siblings were not formally participating in the study. Participating siblings were asked to come in together to sign up and add each other as friends within the app, whereas single participants were grouped with other single participants for the pre-study meetup. The objective of this pilot was to explore how children played with PictureBlocks in a home environment, along with regular distractions given a period of time.

Participants

The study was publicized via flyers (Appendix C) put up in the Cambridge area, MIT mailing lists and parent groups on Facebook. Parents signed up for the study via YouCanBookMe.

Participants were between the ages of 5-9 years old. Though this age group constitutes a range of literacy skills, previous playtesting had indicated that children of different ages utilized the open-ended nature of PictureBlocks differently. This wide age range was also warranted to observe the interactions between siblings using the app.

Our study is exploratory in nature , and we chose not to conduct an assessment of the children's literacy skills pre and post study. This is because for an app such as ours with multiple features, it made it impossible to track all the variables affecting the participants' scores.

There were 5 female and 11 male children, with details in Table 4.3. Although some participants were from bilingual homes, they were all native English speakers and comfortable with English as a first language. With the exception of two children (parent education level unknown), all had parents who were well educated and had affiliations to Harvard or MIT. Out of the 16 participants, there were three sibling groups as grouped by color in Table 4.3:

1. P2 and P3 (Leonas and Viella)
2. P10 and P11 (Siddharth and Nikith)
3. P14, P15, and P16 (Tyler, Kalon and Khloe)

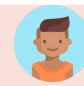
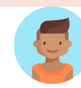
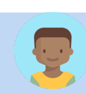
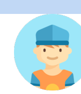
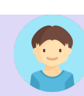
ID	Name	Avatar	Age
P1	Callum		5
P2	Leonas		5
P3	Viella		8
P4	Rohan		8
P5	Presley		7
P6	Julia		7
P7	Niven		5
P8	Ellery		7
P9	Isabella		9
P10	Siddharth		9
P11	Nikith		5
P12	Howard		7
P13	Jacob		5
P14	Tyler		7
P15	Kalon		5
P16	Khloe		5

Table 4.3: Pilot Participants

Preparations

Despite parents owning smartphones and tablets, we couldn't estimate how often these devices would be handed over to children on average. Therefore, we decided to provide each child with an individual phone that was cost-effective, but still offered a large enough surface area for picture composition.

We selected the Samsung Galaxy J7 Prime, to run the latest version of Android. Since several features of PictureBlocks required internet access and interfacing with an online server, our phones were set up with a data plan and a corresponding phone number using T-Mobile. The participants were also provided with a protective phone case and a wall charger. We installed PictureBlocks directly from the development environment onto each phone, and put restrictions in place to protect the app from accidentally being deleted.

Additionally, since participating parents would not want their young children to unsafely access other internet services and features on our study phone, we took precautionary measures and installed Kids Place[4]. This disallowed the child from exiting PictureBlocks and we provided the parents with a password for the same.

[4] Kids Place is a parental control app that restrict kids to only access approved apps [30].

Finally, we ensured that all our study materials went through several iterations to be reviewed and approved by MIT's IRB. This includes the COUHES[5] consent forms which requested additional permission from parents to allow their children to record audio and share pictures with other participants within PictureBlocks (Appendix B).

[5] MIT's Committee on the Use of Humans as Experimental Subjects.

Procedures

Once the participants were recruited, they were invited to attend a pre-study meetup at the MIT Media Lab accompanied by their parents. The duration of this meetup lasted for 90 minutes if the participants were part of a group or for around 45 minutes if the participants were siblings and already knew one another. When the participants came in as a group, the kids and the parents were separated into two areas right next to each other, with the kids often going back and forth to their parents.

There were two researchers present. While the first researcher helped the parent with consent forms and to create a user account on the PictureBlocks app, the second researcher administered a language barrier game (such as Pictionary) amongst the kids. This was to help foster a connection between the children and help them get to know one another.

The moment the PictureBlocks account was setup, the parent and the child together created a profile and chose an avatar for the latter. After all the children had their profiles created, one of the researchers provided instructions on how to use the app using a sample phone. The children were explained the user interface of the app , and demonstrated the entire flow from start to finish. Following this, the children added each other as friends within PictureBlocks and proceeded to take their respective phones home. If the participants were siblings, the above process was repeated with the exception of the language barrier games since the children were already familiar with each other.

The study officially began once the pre-study meetup was complete. The minimum study duration was 15 days, however some parents took longer to return the phones, thereby giving us additional play data for children who played with the app longer. However, for the sake of consistency, we only consider the first two weeks since the phones were given to each participant.

There were no requirements on the amount of time a child was required to play with PictureBlocks. Everyone was welcome to play with it if they wanted to, and for as long as they wished to. Furthermore, there was no stated objective to be achieved within the study period and the child was allowed to freely explore the app.

During this home-based pilot, while children were sharing the pictures they created with their friends within the app, we were concurrently moderating shared pictures that contained audio recordings. Recorded audio snippets could not be filtered for swearing etc, so this was done to ensure that children were not sending or receiving risky content amongst each other.

Once a picture with audio content was shared, the researcher would receive a notification to check the Audio Moderation tool (Section 3.3), listen to the audio recording, and then either approve or reject the picture from being shared.

At the end of the study, we invited each participating family back to the MIT Media Lab on a one-on-one basis in order to collect back the phones, and conduct an exit interview. This session took no longer than 15 minutes.

Once the families arrived, they would hand over the phone. Then, each kid was taken to a separate space and questioned about their experiences with PictureBlocks in a semi-structured format. They were all given a participation goodie bag containing a drawing book, stickers of their in-app avatars, and refrigerator magnets of a few selected pictures they made within the app.

Data Collection and Persistence

While the data from the exit interviews (Appendix A.3) served as auxiliary information, the primary data collected was in the form of logs, with every interaction in the app recorded line-by-line and saved in text files. More details about this form of multi-modal data collection is available in Section 5.1. These log files were backed up to the server every time the app was accessed, and were also accessible from a predefined folder location on the phone's storage. Screenshots of the pictures created and the audio recordings (both deleted and present) were also saved within this folder.

5
Data Analysis and Findings

"Not everything that counts can be counted,
and not everything that can be counted counts."

William Bruce Cameron

In this chapter, we provide a comprehensive view of the data collection and analysis for the Pilot (section 4.4). We then discuss the findings from this study, and try to posit where PictureBlocks stands in the realm of educational apps. Lastly, we reflect on case studies and draw conclusions from our results.

5.1 Data Logging

Applications collect data in-app in order to better understand their users and make improvements. This type of intrinsic data collection is especially useful in research relevant to children's technology - where there are several methodological challenges.

During studies, children might often be distracted or display different behavior due to the presence of a researcher. This problem is further elevated during studies in home environments, where gathering qualitative information via observations over a period of time is difficult.

Therefore, recording a child's every interaction within the app can help us gather data without interrupting their flow, and additionally capture quantitatively analyzable information. PictureBlocks is designed for such a passive collection of data during children's play.

	TIME 69.2626
Phone, App, User Specifications	VER 1 PB-VER 1 LANG ENG STARTTIME 1523646588 CHILD ID #229 0 SCR 720 1280
	TIME 2.088432
Picture-specific commands	PIC 1 NEW SVPIC 1 DPIC 1
	...
Timestamp for every interaction	TIME 2.288658
Sprite creation, manipulation and deletion commands	SPC 2 ball 0 0.7576215 75 0 0 0 60.02999 60.02999 CANVAS DT USER MV 2 2.624999 0 75 TAP 2 SCALE 2 15 15 ROTATE 2 0 0 60 DSP 2
	...
Keyboard Open and Close Letter String Creation and Deletion - SC, MERGE, DEL Speech Synthesizer - SPK	KOPEN SC BW 7 GRAS 373.963 640.384 0.001 m:1.000:-0.001:0.001:1.000 FREE ET USER SC BW 8 S 373.963 640.384 0.001 m:1.000:-0.001:0.001:1.000 FREE ET USER SC BW 9 GRASS 373.963 640.384 0.001 m:1.000:-0.001:0.001:1.000 FREE ET USER MERGE 9 7 8 SPK GRASS DEL 9 KCLOSE
	....
Drawer commands	DOPEN BG space DCLOSE
Finger coordinates	FR 0 0.4583338 -0.1527777 1 FU 0 -1.805556 3.041667
	....
App Pause or Exit	PAUSE END

Figure 5.1: Sample PictureBlocks Log featuring example commands.

Our software has been instrumented to record all touches and interactions within the app, which are saved into text files that we refer to as Play Logs. These logs are time-synced and include all the information necessary for us to reproduce a child's play session. This comprehensive data collection serves as a rich corpus of both qualitative and quantitative data.

An example log file is displayed in Fig 5.1. Each log file is named with the date and time when the app starts on the device. All logs begin with the identity of the player who is currently logged in and the specification of the device being used, such as screen dimensions. As players interact within the game, time-stamped touch points on the screen, and every interaction with the app and its gameobjects are recorded in these logs line-by-line. Logs also record the content of auditory feedback provided by the speech synthesizer.

Other captured information includes what pictures were made and deleted, words typed out by the child and any resulting sprites, children's audio recordings, sprites manipulated, and pictures shared with other children. For example, logs corresponding to entering and deleting letters using the keyboard can give us information on the way a child spells a word. Also, touch logs can provide meta-data on heatmaps and what parts of the user interface were frequently used, and how. Logs files are closed when the app is exited or paused.

This fine-grained level of data collection is to ensure that no potential data is lost during this study. Therefore, the collected data is not limited to a specific research question and can be applied to further analysis as questions evolve. The next section provides an overview of data-driven analysis methods built to explore this play data, also referred to as play analytics.

Previous play analytics in similar research have shown positive results and highlight "implications for researchers, educators, and parents by providing a more descriptive view of children's learning processes and literacy skills". [65] [47]

5.2 Analysis Tools

PictureBlocks acts as an unstructured playground for collecting contextual and behavioral data (captured in Play Logs). We have devised the following tools to explore this data both qualitatively and quantitatively:

1. *Log Parser Tool (Quantitative)*

   Play Logs are transcribed at an intricate level of granularity and therefore, need to be converted into meaningful abstractions of player actions on the game environment. This ensures easier data readability for the researcher. We built a Log Parser Tool that parses raw log files. It checks every command line-wise and converts them into significant events and actions - which mirror interactions within the game. This tool is written in Python and includes classes respective to Session, Picture, Sprite, Child, and so forth which extracts each log file into a corresponding Session object. We use this tool for implementing methods for quantitative evaluation of the data. For example: methods to calculate total gameplay duration of a player over two weeks, methods to check for incorrectly spelled typed words and actions taken post that, methods for determining the depth of player's exploration of the semantic network, et cetera.

2. *Log Replay Tool (Qualitative)*

   Children's play data, especially data consisting of potentially creative pictures and contextually similar information, cannot often be measured quantitatively alone. To evaluate some of our app's more qualitative features, a researcher has to go through the game play and the final pictures created, and annotate them. Therefore, we built a Log Replay Tool which takes as input the Play Logs and completely reconstructs everything that happens on the child's screen over the two weeks of study duration. This output of this tool are videos corresponding to each log file which reflect the data in real-time (i.e., it captures the exact time a child uses for an action such as to type a particular word). The video displays the PictureBlocks app and proceeds through different screens, with the child's touch points and taps being represented by fingerprint icons. This tool was built using Unity and the videos were screen-recorded using QuickTime Player [1]. Researchers evaluated these videos and annotated them for results using a custom built dashboard.

[1] Inc. Apple. *How to use QuickTime Player*. 2018. URL: `https://support.apple.com/en-us/HT201066` (visited on 07/31/2018)

3. *Association Trees (Qualitative)*

When a child explores a word's semantic network using the Associations Panel, it may be interesting and useful for parents and researchers to observe the words they encounter from a single starting point. We captured this exploration with the help of a visualization which shows how a child gets from word A to word B while using this feature within PictureBlocks.

To construct this visualization, we kept track of the buttons the child tapped on within this panel and mapped them to nodes in a network. Figure 5.2 depicts an example of exploring the Associations Panel beginning from the word "ball" and getting to the word "chicken". Here, the starting point is "ball", with subsequent taps on buttons for "egg" and "chicken" respectively. This interaction is mapped to the corresponding visualization shown in Figure 5.3. This visualization resembles a tree structure with nodes reflecting the tapped objects and leaves reflecting all of its associations. We refer to these visualizations in the rest of this document as Association Trees.

For each study participant, we generated these visualizations for all such explorations within the app. This was implemented by extracting the relevant data from the log files, and converting them to a JSON format. Each JSON file was then passed through our tool, which uses the D3 JavaScript Library to make these interactive web visualizations.

Figure 5.2: Associations Panel Exploration

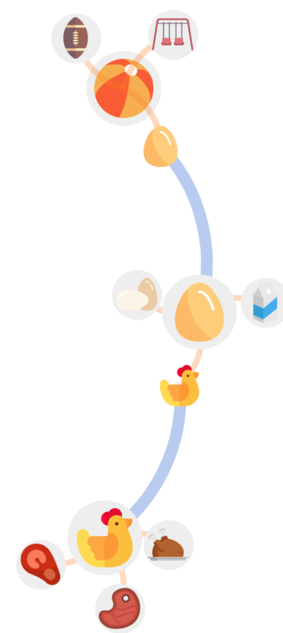


Figure 5.3: Association Tree

We applied the aforementioned analysis methods on the data from the Pilot study and highlight our findings in the rest of this chapter.

5.3 *App Usage and Overall Experience*

Out of the 16 participants, we do not consider two of the children's data towards our final results - P2 (Leonas) and P10(Nikith). In P1's case - the first week of play data was missing due to an unforeseen update in the app. For P10, the parent often logged into the kid's profile to make pictures - therefore, altering the data and rendering it inaccurate for our analysis. However, both P2 and P10 have siblings who were also participants in this study. Pictures shared undirectionally from P3 (Viella) to her brother P1 (Leonas), and from P11 (Siddharth) to his brother P10 (Nikhith) will still be considered for analysis of the Social Sharing feature.

Engagement

- **Time Spent in App:**
  We did not impose any requirements on how often children needed to play with PictureBlocks, so the play duration and sessions are voluntary. During these two weeks, some parents and children reported being away for school spring break, visiting relatives and/or having limited screen time at home, restricting their usage. This is similar to how any other app is used in a home environment, and we did not make any special adjustments to account for these breaks.

  For each child, we calculated the total time spent in app as the sum of all their individual session lengths, subtracting the durations for when the app was paused or not in the foreground. These total play durations per child are plotted in the Figure 5.4. Over the two weeks, P9 (Isabella) had the minimum total play time of 25 minutes whereas P7 (Niven) displayed the maximum total play time of 13 hours, 53 minutes.

  The median total play time among all kids is 3 hours, 23 minutes, and the mean total play time is around 4 hours, 6 minutes.

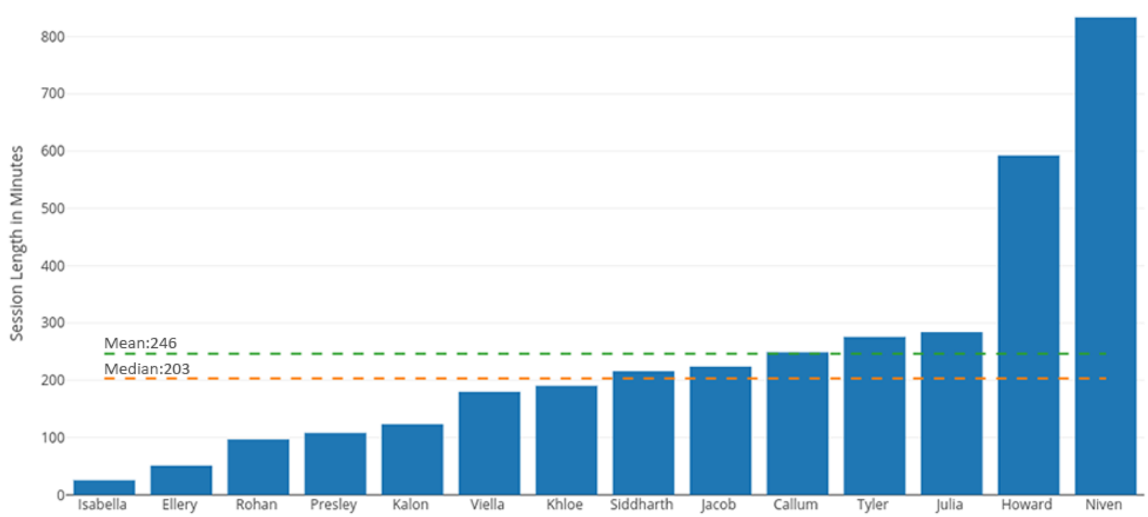


Figure 5.4: Total Play Duration over two weeks (minutes)

- **Player Retention:**

  Retention can be defined as the percentage of players who return to the app after an initial play session.

  Common app industry standards are to measure retention rates for Day 1 (D1), Day 7 (D7), Day 14 (D14), Day 30 (D30) and so forth. Sensor Tower is an app analytics platform that helps companies with mobile apps understand and improve their app's performance, user acquisition, and keep up to date with competitors [71] . We retrieved SensorTower data for two apps mentioned amongst our Relevant Apps and Games (Section 2.3) - Endless Wordplay and Endless Reader.

  Fig 5.5 displays D1, D7, D14 retention rates for PictureBlocks plotted against the expected retention rates for both these apps. Around eighty five percent of PictureBlocks players returned to the app after both D1 and D7, and fifty percent of the players returned to continue playing even after two weeks.

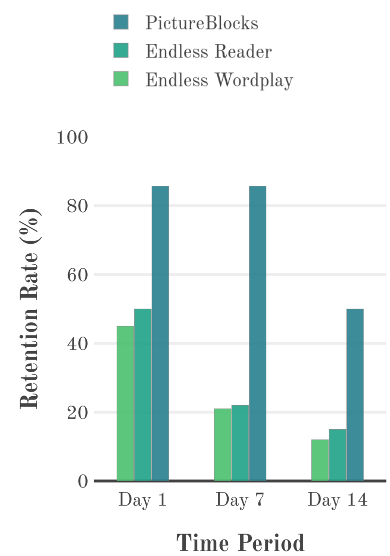


Figure 5.5: Day 1, Day 7, and Day 14 Retention Rates

- **Duration and Frequency of Play:**

  The duration of play is the length of the individual play sessions, each session beginning from when the player opens the app and ending when they exit or pause the app. The frequency of play is the number of play sessions which can provide insight into the "stickiness" of the app. [2]

  Figure 5.6 displays a mapping of the session counts and session lengths for each study participant. From this graph, we can observe that player P7 (Niven) had the maximum number of sessions (61) and player P1 (callum) had the longest session lasting a total of 2 hours and 12 minutes. The below table (Table 5.1 provides some insights from analyzing these metrics.

[2] "Stickiness" means how likely an app is to keep a player engaged in the long term, or in colloquial terms, how likely they are to "stick" to the app.

Median Session Length	11.3 minutes
Median Session Count	15.5 sessions
Average Session Length	12.45 minutes
Average Session Count	19.6 sessions

Table 5.1: Session Lengths and Session Counts

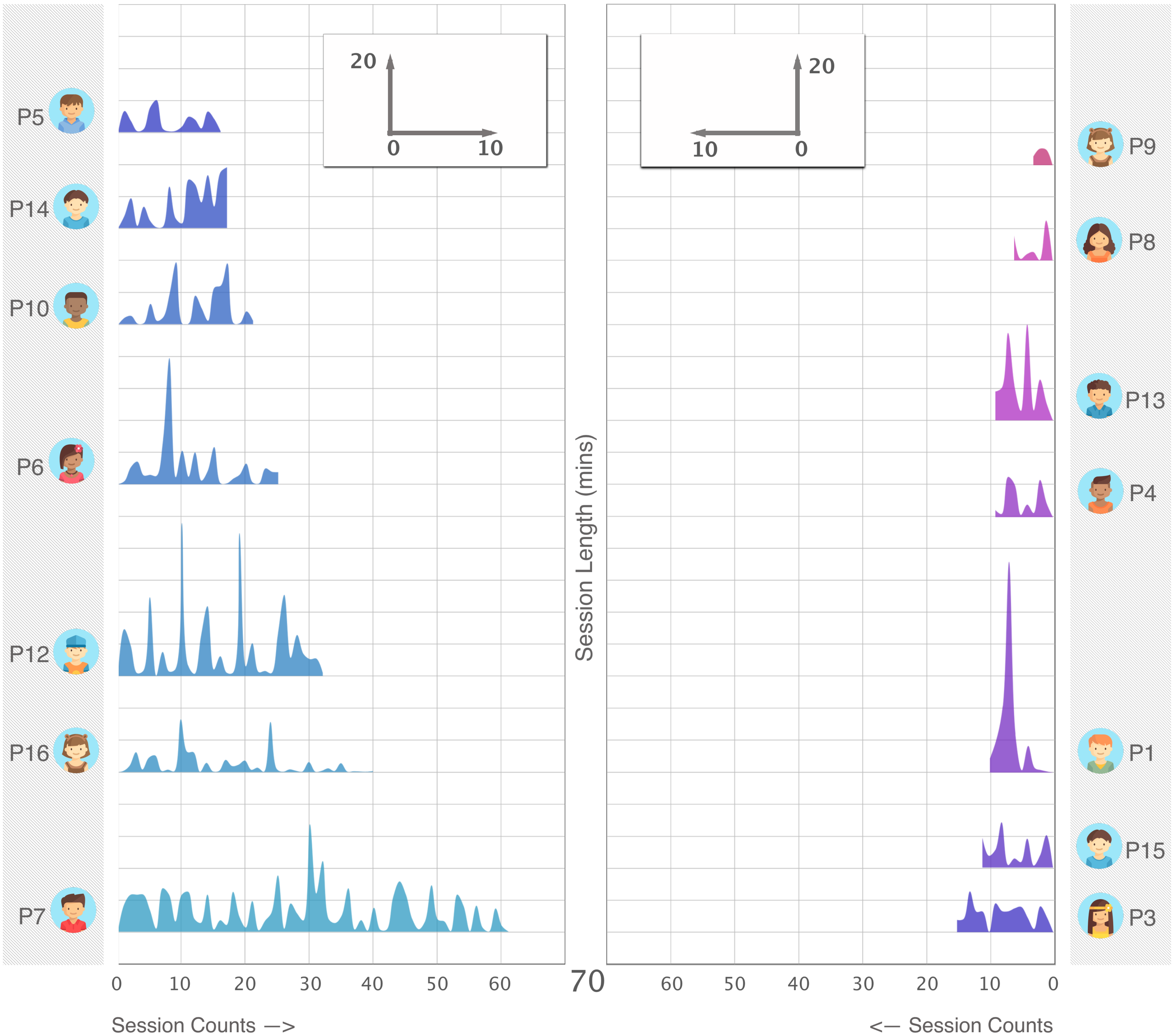
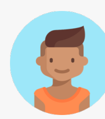
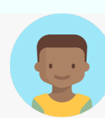
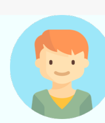
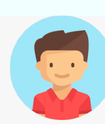


Figure 5.6: Play Session Counts and Session Lengths per Participant.

App Impressions

Overall, the response to the app was fairly positive. Children's positive descriptions of the app varied from "Good", "Really great!" to "How did you make this so awesome?". One of our older participants, Viella (Age 8), said she started off loving the app, but over time, it became less exciting for her to play with, as the only thing she felt she could do was to make pictures. The data also displayed a usage drop off for Isabella (age 9), who said she really enjoyed playing with the app, but didn't have enough time to play due to external factors.

Since we didn't follow a gamified approach with points, levels, right or wrong answers etc, we were interested to see what children perceived the goal or purpose of PictureBlocks to be. These are some of the answers we received - when we asked them what PictureBlocks does, and how they would describe the app to their friend:

"It's a very cool app. I can sleep with it in my bed, playing games with it." ..."It was great fun playing with it. I really loved it. Extra good, because you know, it does lots of fun stuff."		Jacob, Age 5
"I would describe it as fun, yeah it's really fun"..."I think PictureBlocks tries to create your imagination. It's trying to show us that we can be creative. We can be creative on our own and don't listen to others creativity. Do your own creativity."		Rohan, Age 8
"It's a fun game that you spell words and if the computer knows the word, the picture will pop up... and you can drag it or do whatever you want.. like make sounds for it. And it's really fun, because you get to do whatever you want."		Ellery, Age 7
"It helps you with spelling, and it tells when you spell a word, it shows a picture on it. And then you can get better at it."		Siddharth, Age 9
"This app tells you the letters so that you know what letters to type in for the (to get the) pictures."		Callum, Age 5
"(This app) creates picture magnets that I can take home!" (referring to the participation magnets we gave him at the end of the study)		Niven, Age 5

User Experiences

Exit interview answers on user experiences indicated that there was a variation in the things children liked or disliked about the app. For example, while one player loved the app's backgrounds, another wanted more background options. While one player thought some words were missing from PictureBlocks, another thought that it was fun because it already had several different types of words. These opposite likes and dislikes could be due to differences in players' personal tastes, ages or literacy levels. Table 5.2 provides a few reflections about user experiences, in the children's own words.

Table 5.2: Selected User Experience Quotes

Likes	Dislikes
"The fun part was that you could basically do whatever animal or everything you want on it.".	But the non fun part was that it's still getting updated and there are words which it doesn't know.
"I really like the backgrounds. You can pick the galaxy, sunset, forest background."	"Well the least favorite part is that why couldn't you put a grade section? Because then I would click on 8th grade and learn new words every time."
"It has most of the words - Like if you type in anything , like a keyboard it comes up. And like a radio and a TV."	"The backgrounds."
"The most fun part was making Halloween pictures. Because I like Halloween because it gives you candy."	"No. There was no boring part."
"Well, my favorite part about PictureBlocks was that it got to find our imagination and helped get a bigger and richer imagination."	"Sometimes if I try to drag an object to delete it, it goes underneath the drawer."

In Sections 5.4 and 5.5, we will dive further into the pictures players made, and how they interacted with individual features of the app.

5.4 *Picture Categorization*

Children composed and saved a total of 292 pictures throughout the course of this study. By observing the overall modes of interaction, we grouped their created pictures into the following seven categories:

1. **Sprite Exploration:** Initially, children often started out by exploring various sprites within the app, where the purpose was to discover as many sprites as possible.

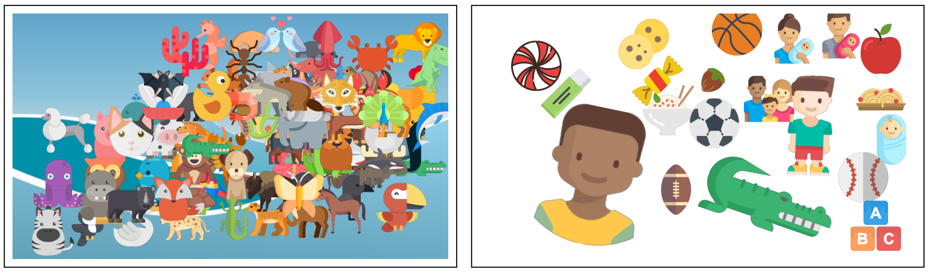


Figure 5.7: Picture Category - Sprite Exploration

2. **Collections:** Another common observation indicated that children liked making collections of similar objects, often times neatly arranging the sprites.

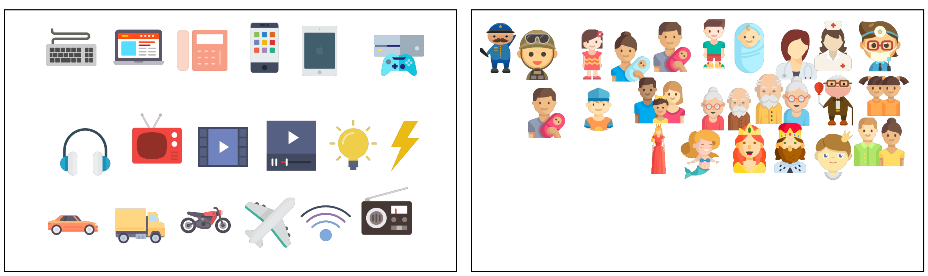


Figure 5.8: Picture Category - Collections

3. **Scene Setup:** Several children used the background images within PictureBlocks to create a scenery with relevant sprites. This category is for those pictures which follow a certain theme, but don't contain any visible narrative.

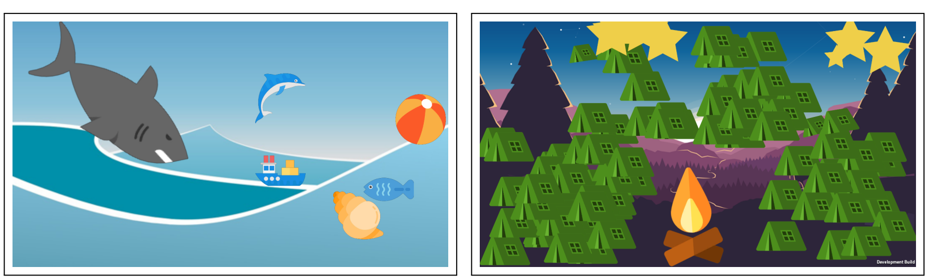


Figure 5.9: Picture Category - Scene Setup

4. **Narrative/Story Telling:** Sometimes, looking at a picture made it immediately clear that it contained narrative or a story around it.

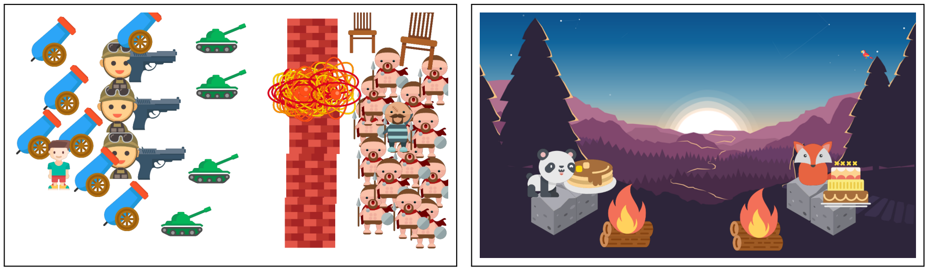


Figure 5.10: Picture Category - Narrative & Story Telling

5. **Repetition Play:** Repetition Play was more common amongst younger children. Upon discovering a sprite, they would like creating it several times irrespective of whether the duplication contributed to the overall picture or not.

Figure 5.11: Picture Category - Repetition Play

6. **Pattern Play:** Some children didn't just create repetitions of sprites, but they also rearranged them to form patterns.

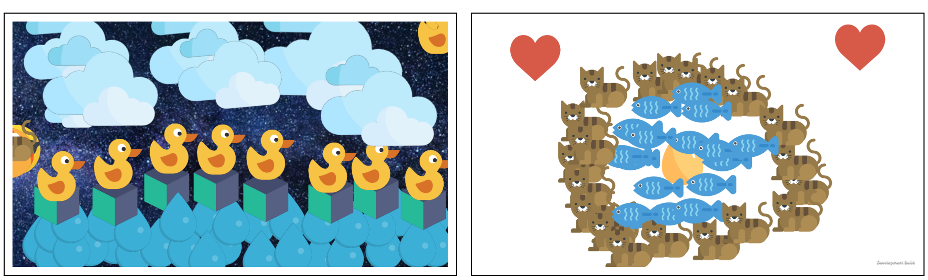
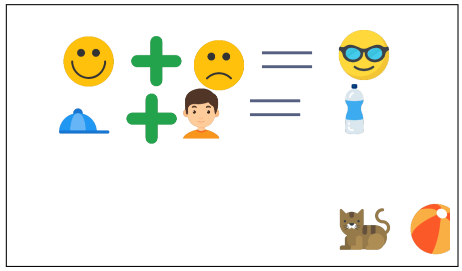


Figure 5.12: Picture Category - Pattern Play

7. **Mixed-Bag:** Finally, all other uncommon types of play constitute the "mixed-bag" category. Testing PictureBlocks with a larger population of children, the variation amongst these "mixed-bag" pictures may be indicative of newer categories arising in the future.

Picturization of the English nursery rhyme - "Hey Diddle Diddle"



Use of mathematical symbols to depict relationships between objects.

Figure 5.13: Picture Category - Mixed Bag

5.5 *Efficacy Of Design Features*

Each primary feature added to the PictureBlocks app was driven by certain design decisions. We breakdown the player's interactions with each of these features to better understand how they contribute to the overall app experience. We inspect the session videos reconstructed by the Log Replay Tool (Section 5.2) and provide findings on the features' user interface, data and behaviour.

Feature 1: Keyboard

The open-ended design of the keyboard - which allowed players to make whatever words and sprites they wanted, was the highlight of the app for most children. The idea of making objects they were passionate about, as well as making a variety of objects were both reflected in the keyboard's usage.

User Interface:

As observed from the video replays, this feature was easy to use and often the go-to starting point for most of the players when they opened a picture. Children also made use of both the QWERTY and the ABC keyboards, switching between them whenever convenient.

Data and Behaviour:

Children typed out an assortment of words including correctly spelled words, misspellings, nonsense words, and even entire sentences. In fact, one of the user suggestions at the end of the study was to include a spacebar button in the next version of the app so that they could type in words like "twin brother". Another noticeable pattern was that some children often bombarded the keyboard with words in order to test the limits of the app, and check if it contained a sprite for that word.

For every sprite created using the keyboard, children would have had to type in its correct spelling in order to get the respective sprite. Therefore, by inference, we can conclude that children made as many correctly spelled words as the number of sprites they made within the app.

The data also displayed evidence of children frequently correcting their misspelled words until a sprite popped up on the keyboard handle. This effort varied between player to player, and for different words. For example, let us look at Table 5.3 where P9 (Isabella) and P6 (Julia) are both are trying to type out the word "dinosaur".

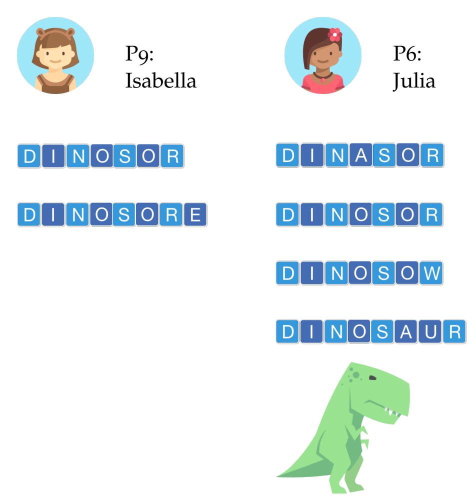


Table 5.3: Misspellings and corrections for the word "dinosaur"

Here, Isabella gives up after one or two tries and moves on to making the next word. On the other hand, Julia keeps trying different variations of the spelling until she discovers the dinosaur sprite. There are several such instances within our data where children have done both - typing variations of spellings and giving up, as well as typing variations of spellings until the sprite pops up. But the important common factor in both of these scenarios is that **the children are trying to spell**. As a result, not only do children sometimes discover the correct spelling, but they are also engaged in invented spelling activities while doing so.

The above types of data collected from the keyboard seem extremely useful from the point of view of stakeholders involved in children's learning. Parents, teachers, and researchers can gain insight into a child's spelling skills - what words they commonly misspell and how, what phonemes are often incorrect during the construction of a word and so forth. Other specific information such as a child's favorite words and their variations in effort are also decipherable from this data.

Feature 2: Picture Canvas

The canvas was designed to be used for picture composition - with the resulting pictures being saved into the player's Dashboard. As previously specified in Section 5.4, this behavior was indeed reflected and children composed a wide range of pictures.

User Interface:

There were only two minor impediments to user flow:

- Whenever a player dragged a sprite into the drawer handle for deletion, it would sometimes get hidden behind the drawer handle instead.
- Previous playtesting had indicated that a blank canvas might be intimidating, so we provided a cat and a ball as starting sprites. However, children would open a new picture and immediately proceed to delete the cat and ball, to make the canvas blank. This calls for identifying a new approach to handle this app element.

Data and Behaviour:

Apart from picture composition, we discovered a previously unconsidered but positive use of the canvas by analyzing our video replays - which we refer to as Canvas Play. From our original design perspective, the end goal for the children was the product they made - the composed picture. But for some players, the end goal was not always the resulting picture they created, but rather the process of playing around on the canvas itself.

Figure 5.14: Canvas Play

For example, P13 (Jacob, Age 5) loved superheroes, and every picture he made on PictureBlocks included at least one superhero sprite. On the canvas, Jacob would pretend play with the sprites. His finger gestures of moving the "superman" sprite wasn't to place it in a particular location, but rather to fly superman around. Other forms of play included making superman really tiny or large and flying him off into the distance.

Feature 3: Associations Panel

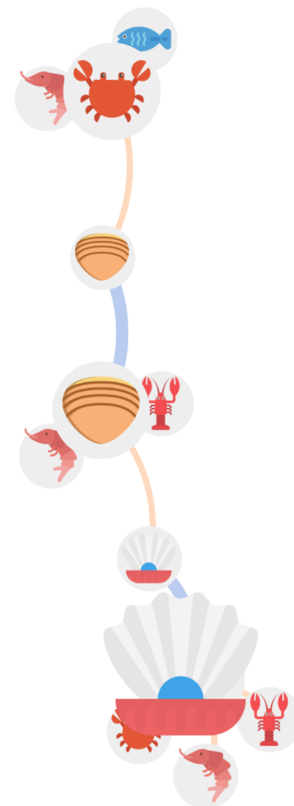


Figure 5.15: Association Tree : "lobster" -> "clam".

User Interface:

Children easily navigated through the Associations Panel without encountering any obstacles. Purely in terms of design, we believe that this panel can be a very useful exploration tool in mobile applications. Children's learning apps can embed this panel as an independent UI widget for the purpose of presenting players with word associations or other similar recommendations.

Data and Behaviour:

Association Trees (Section 5.2) help us visualize how a child explored the semantic network and capture their choices made from start to finish. Pictured on the right (Fig 5.15) is one of the association trees formed during a play session for player P3 (Viella). With the help of various such explorations, she finally creates the picture below (Fig 5.16).

This type of exploration and resulting picture suggests that one of the uses of the Associations Panel is to discover other sprites existing within the app. If the player in this case - Viella, already had an ocean theme in mind, then by exploring the panel, it is entailed that she was exposed to further semantically similar words that contributed to her overall picture.

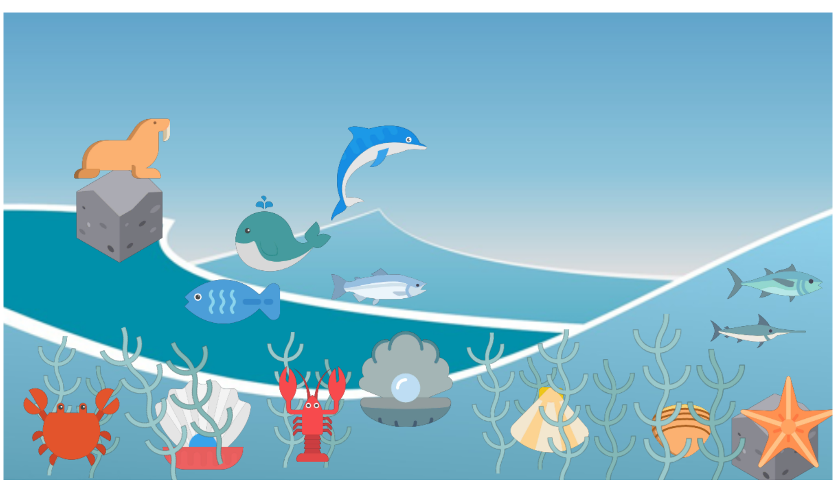


Figure 5.16: Artist - P3 (Viella)

The most powerful use of the Associations Panel was when it was used in conjunction with the Keyboard. Children, while exploring the semantic network with the help of the Associations Panel, would often encounter a sprite object that they would want to create on their canvas. There can be one of two things that happen next:

- The child already knows how to spell the word and the Associations Panel has helped them discover that word.
- The child has discovered the word using the Associations Panel but doesn't know how to spell it. S/he uses the text provided on the Panel to then type out the word into the keyboard correctly.

We provide an example of player P9 (Isabella) who used the Associations Panel to discover and then type out the word "husky". Fig 5.15 shows the association tree for this interaction where she went through 5 levels of the semantic network starting from the word "cat" to get the word "husky". Simultaneously, Fig 5.16 shows Isabella's screen as she is guided into typing out the word "husky" by looking at the text displayed on the Panel.

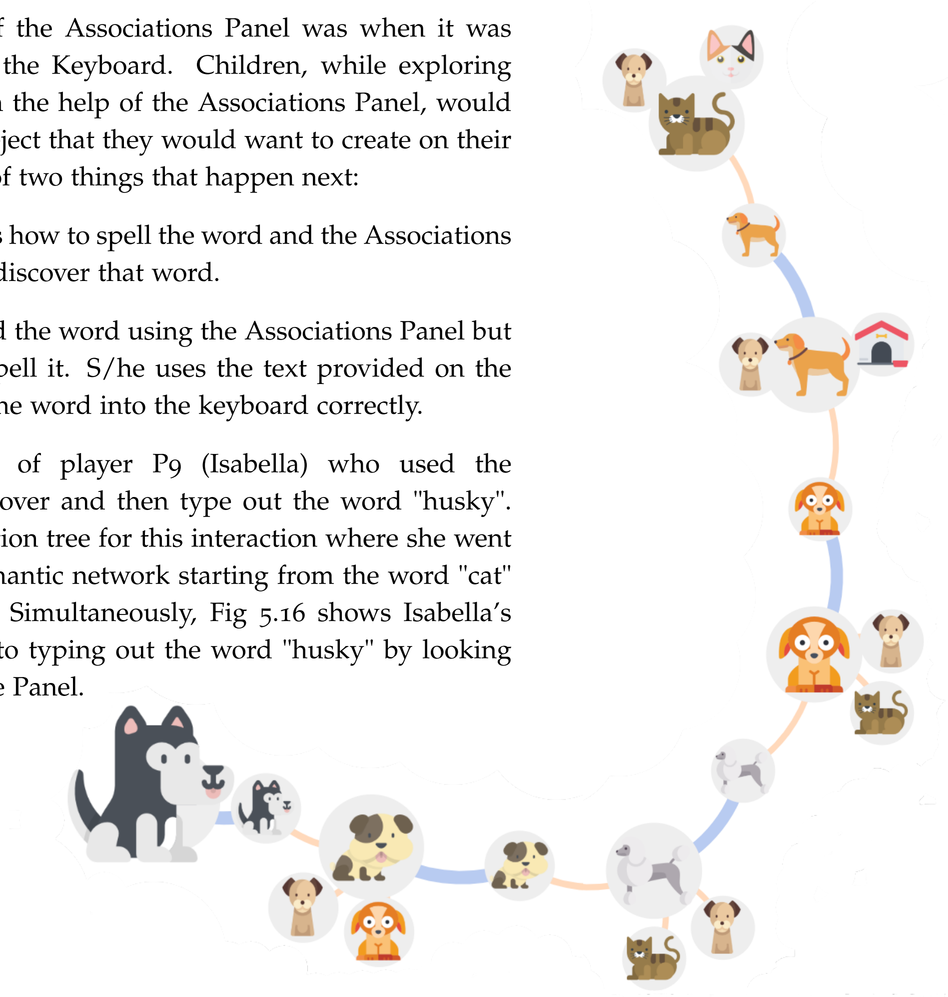


Figure 5.17: Association Tree: "cat" -> "husky"

Figure 5.18: Session Replay displaying the combined use of the Associations Panel and Keyboard to form the "husky" sprite.

Feature 4: Audio Recording

Children used the Audio Recording tool for everything from singing to screaming to telling a story. There were a total of 245 audio snippets recorded overall among the 14 kids.

User Interface:
Initially, some kids had difficulty figuring out how to record their voices correctly. They would hit the "Record" button and then the "Stop Recording" button almost immediately after. This caused the recordings to sometimes capture only the beginning of their sounds. However, by the time the study was over, all children had learned how to record an audio snippet properly. In future versions, including a first time user experience helping the kids to use different features of the app will be more effective.

Data and Behaviour:
We transcribed each audio snippet and although it is difficult to convey their essence completely using text alone, we have still provided some examples in Table 5.4. Different categories of audio snippets recorded in the study are listed below:

- musical sounds - singing songs, sentences, and even opera
- silly sounds - a lot of kids loved making silly sounds, and then laughing at themselves by replaying it afterwards.
- sprite sounds - this was a very common play amongst several kids. For e.g., children created animal sprites and made animal noises.
- dialogue sounds - some sprites had a dialogue/conversation with other sprites in the picture.
- storytelling sounds - the storytelling was captured as a narrative explaining the picture, or as part of the dialogue.
- scene sounds - audio that adds context to the picture made.
- self sounds - audio which spoke about themselves as a character (or their own in-game avatar)
- people sounds - several siblings and even parents used this feature along with the participants to make sounds.

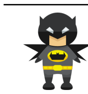
Sprite	Audio Recording
	"I am Batman. I am the dark knight."
	"I'm a frog. Ribbit. Ribbit. I'm a frog. Ribbit. Ribbit."
	"Dont' worry. Be happy."
	"The wheels on the bus go round and round."
	"Dabidabaloo"
	"Hello my name is Niven."
	"Hello my name is Sepalika." (Niven's mother)
	"Mmmmmmm donuts."

Table 5.4: Examples of Audio Recordings attached to Sprites

Qualitative analysis revealed that even though our tool allowed audio recordings to be attached to only individual sprites - most of the recordings made by children were more closely tied to the overall picture rather than the sprite. With every audio recording, children added more context to the composite picture they were creating, while simultaneously adding a unique voice to the sprite itself. Let us look at the below examples that supplement these observations.

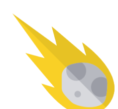
Composed Picture	Audio Snippet	Sprite
	*cackling noises*	
	ghostly howling sounds	
	growling sounds	
	"In your head, in your heaad, zombie zombie zombie" - *sings Zombie by The Cranberries*	
	*explosion sounds*	
	"Hey, wait for me! That is my rocket."	

Table 5.5: The role of sprite-specific audio recordings in the context of the composed picture

Feature 5: Social Sharing

We analyzed our data to check for instances where the app's Social Sharing feature was used. Keeping the siblings aside, the rest of the participants met each other only once at the beginning of the study. Therefore, it was hard to initially predict the popularity of this feature.

User Interface:
The number of pictures shared indicate the ease of using this particular feature. During the exit interview, participant P1 (Callum - only 5 years old) demonstrated this fact to the researcher by sharing a picture with his friend P7 (Niven) in under a minute.

Data and Behaviour:
We ask the following questions from the data:

- Did children use the Social Sharing module? If yes, how often were they sharing the pictures they were creating?

  Every participant both shared and received at least one picture from a friend, with the exception of one player (P9: Isabella). The maximum number of picture shares made were 106 (P15: Tyler). The following figure displays the total number of picture shares, and the total number of pictures created per player. One picture can be shared multiple times, and to multiple friends - therefore, this table reflects cases where the number of picture shares is higher than the total number of pictures created.

- How many pictures did they share and with whom?

  To identify the flow of information, we mapped each exchange of a picture between a player and their friends into the following network visualization (Fig 5.19). This graph depicts a social network formed within PictureBlocks as a result of using this feature. The graph is unidirectional, with each node representing a player. An edge is formed between two nodes if there is any feature interaction between them (sending or receiving a picture). Here, the color of the nodes indicate a player's activeness within the network. The higher the traffic of shared pictures to and from a node, the darker its color shade.

  Observing this network, we identify which player was interacting with whom, and also notice smaller sub-groups of nodes formed

Participant	Picture Shares	Created Pictures
P1	4	27
P3	9	13
P4	1	8
P5	2	7
P6	16	16
P7	32	27
P8	1	6
P9	0	5
P11	35	34
P12	9	104
P13	5	14
P14	106	17
P15	13	6
P16	6	8

Table 5.6: Number of Shared Pictures Per Participant

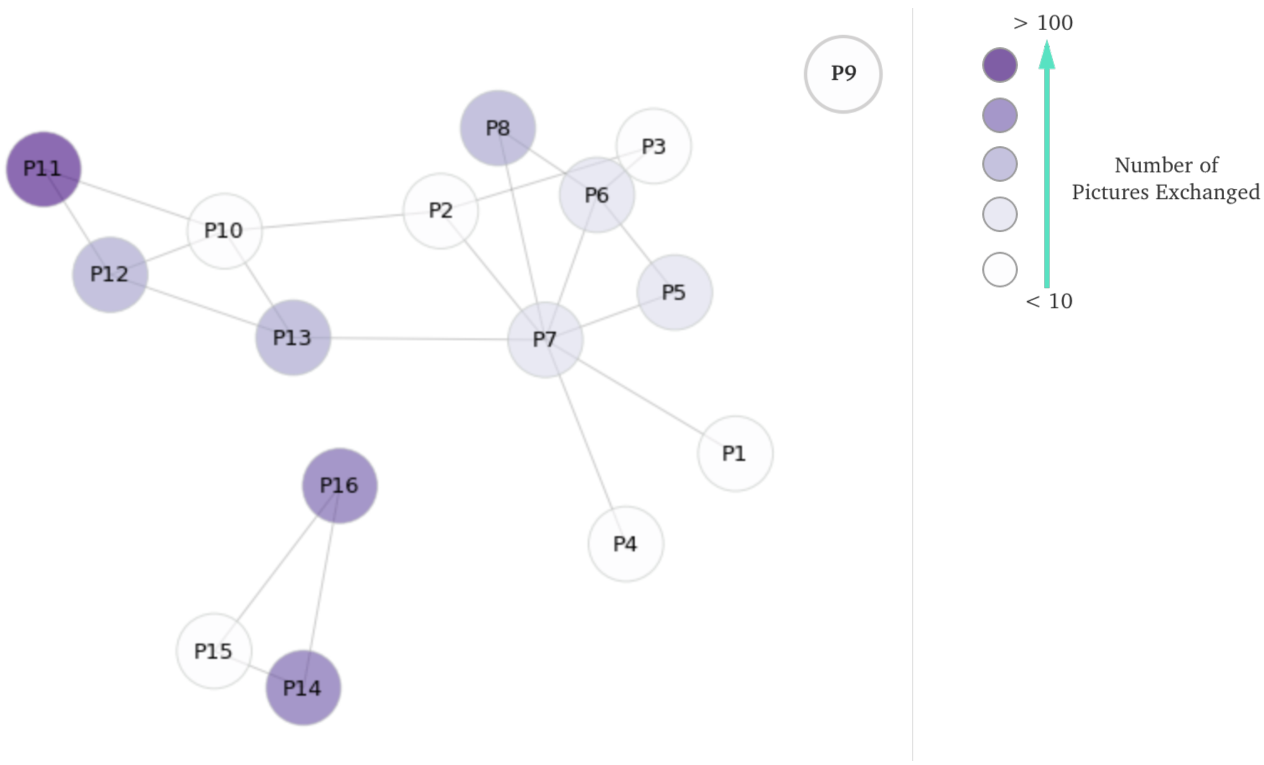


Figure 5.19: Picture Sharing Social Network

both within and outside the network. The majority of the nodes form a mesh like structure with P7 (Niven) having the most number of edges. Therefore, we can infer that he shared a moderate number of pictures (P7 node color), but with the most number of friends.

Nodes P14 (Tyler) , P15 (Kalon), and P16 (Khloe) together form a triangluar sub-structure that lies outside the main network. This makes sense since the three of them are siblings and were each other's friends within the app. Even though P15 (Kalon) and P16 (Khloe) are the same age (twins), they show a difference in the number of times they interacted with this feature. P9 (Isabella) is an orphan node that also lies outside the main network, as she did not share or receive a picture.

- What effect did receiving a picture have on a child?

Children reported almost always paying attention to a picture received from someone else. This was less often because they noticed the name of the friend who sent it, but more often because it stood out in their Dashboard as being something that was not created by them.

When a child received a picture, sometimes he/she discovered new words/sprites which were previously unseen by them or which they didn't know existed within the app. For example, P10 (Siddharth) mentions: "When P12 (Howard) sent me a picture, I was like oh there's a policeman (in the app). I tried to type it in, and it worked."

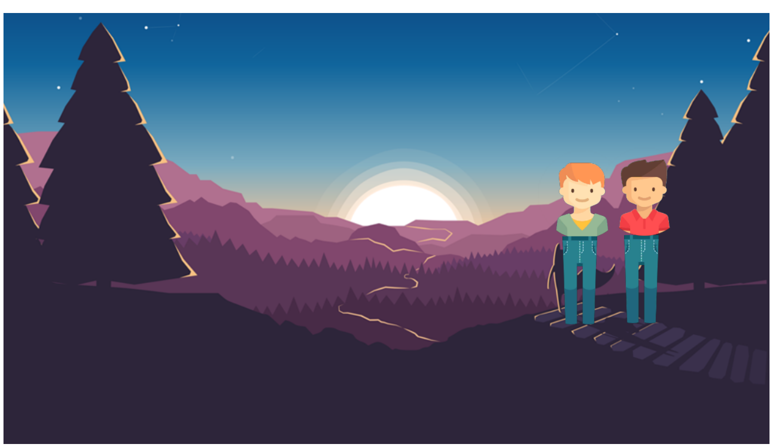
There were also several instances of remixing within the data, where a picture received from a friend was modified by the player into their own version.

P1: Callum: Age 5

P7: Niven: Age 5

To highlight both an instance of remixing as well as a social relationship between the participants, we look at the exchange of pictures between P1 (Callum) and P7 (Niven). Callum and Niven were not only friends within the app, but they were also neighbours and knew each other prior to the study.

1. Callum sends Niven a picture with multiple sprites. One of these sprites is also an avatar of himself within the app. (Note that the app only contains the current player's avatar, and does not allow friend avatars to be discovered by typing them out.)



2. Niven receives the picture and now has access to Callum's avatar on his app . He then modifies this picture, includes his own avatar next to Callum's - with the two of them standing next to each other like friends. He has "remixed" the picture.

5.6 *Interpreting Results*

Due to the exploratory nature of our research and our open-ended app design, there is no right or wrong way of playing with PictureBlocks or to demonstrate mastery. Moreover, despite there being a variety of app evaluation frameworks to choose from, there are very few concrete ways to measure abstract concepts such as self-expression, open-ended play and learning, and social pragmatics. For interpreting the success of PictureBlocks, a few of the frameworks we considered were the Four Pillars of Educational Design [25], Serious Game Design Assessment Framework (SGDA) [44], the Four P's of Creative Learning [58], and the Four Indicators of Learning. [53]. Most of these methodologies have several overlaps in the metrics they place value on.

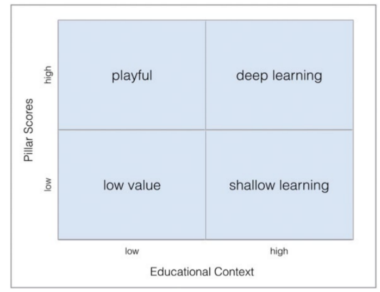


Figure 5.20: A grid for determining the pedigree of an app, as described in the Four Pillars Framework

We decided on mapping the PictureBlocks results against the Four Pillars of Educational Design framework, primarily because this rubric was also referenced in a recent report specific to literacy apps [72]. Using decades of research from the Science of Learning, the authors of this framework have identified four pillars that enable children's learning, thereby determining the potential "educational value" of an app. We analyze the findings from our study and provide descriptive observations against each of the following four pillars:

1. Active Learning

   Active Learning involves more than just tapping and dragging without purpose or thought. According to Hirsh-Pasek et al, [25], the activities within the app must be "minds-on" - such that they require thinking and intellectual manipulation on part of the child.

   The building block of play within PictureBlocks is the creation of a sprite and the app allows only three ways of creating sprites:

   - First, if the sprite words are typed out onto the keyboard letter-by-letter, with the correct spelling. By inference, we can assume that every time a child typed a word out onto the keyboard, their activity was probably "minds-on", as they had to come up with not only a choice of word to type in, but also think through the correct spelling for that word.
   - Second, if the sprites are saved into the Picture Box and tapped in order to create duplicate copies. This duplication of sprites can indicate mindless dragging and dropping, requiring the child to be less "minds-on". However, there are two important things to consider here: (1) In order to save the sprite into the Picture Box, the child would have had to type it out in the first place,

requiring them to think, (2) The duplication of sprites might be towards the construction of an overall picture that indicates a story or a narrative. For example, in the case of P12 (Howard) - he duplicated several soldier and army themed sprites in order to create a narrative around war. Therefore, just because a child is duplicating a certain sprite does not necessarily indicate that they are not "active".

- Third, if a friend sends the child a sprite that they haven't made. In this case, the child is not directly performing an action, but on observing the received picture, might observe new words/sprites that they haven't encountered before, or try to understand the context of the overall picture.

2. Sustained Engagement

   While reviewing our quantitative results pertinent to app retention, time spent in app, and average session lengths - PictureBlocks demonstrates good to excellent player engagement. However, an important point to be noted is that these results are only representative of a small-scale study run over a period of around two weeks. There exists a possibility that we might obtain different results with a longer study and/or with a larger group of participants. Specifically, with older children (ages 8-9), we observed either a sooner drop-off in play, or self-reported answers on how/why their engagement with the app reduced over time.

   Thus, for future app iterations, we can restrict the age range of PictureBlocks to be between 5-7 years old, and observe resulting engagement metrics. Or we could implement new features within PictureBlocks that cater to an older audience, to prevent them from gradually losing interest in the app.

3. Meaningful Learning

   The opposite of rote memorization, meaningful learning is the learning of personally relevant material with a purpose, and linking new information learned to prior knowledge.

   The open-ended design of PictureBlocks doesn't provide a structure or a static list of words/sprites for the child to memorize or create. Every child in the study created sprites using their own imagination, thereby suggesting that they typed words which were personally relevant. The sprite creation process also contributed towards the larger purpose of making a composite picture.

Furthermore, the pictures saved by a child onto their Dashboard or shared to another friend reflect that they were meaningful to the child in some manner, in order to generate such behavior.

4. Social Interaction

   Our app included a Social Sharing feature which promoted mediated social interaction between children. However, the presence of a social feature alone doesn't guarantee social interaction between children. Results from our data analysis also support the usage of this feature within the app, despite several kids only having met each other once.

   It was not a part of the explicit design, but the app also fostered several in person interactions between siblings, as well as between a child and their parent. During the exit interview, parents were not only aware of the words and pictures their children made within the app, but on various occasions, both the child and the parent acknowledged making a word together when the child experienced difficulty with spelling a word. Moreover, the audio data also included several parents' voice recordings, demonstrating participation in the picture creation process with their child. These interactions show promise towards supporting parent-child collaborative features in future iterations of our app.

   Between siblings, in-person interaction was even more noticeable as a few participants had siblings who were not a part of the study; yet these siblings also created profiles within the app, and shared pictures with the participants. Audio data also revealed siblings recording their voices and laughing at funny sounds together. While conducting exit interviews for the siblings together, each group of siblings who were a part of the study exclaimed that almost every time they made a new picture, they would send it to their sibling. Consequently, siblings demonstrated shared knowledge of each others created pictures.

Our findings from the PictureBlocks pilot reveal examples and usage statistics that suggest children's experiences often included Active Learning, Engagement, Meaningful Learning and Social Interaction - thereby, displaying conformation to all of the Four Pillars of Educational Design. Thus, the presence of these pillars within our app signifies preliminary success in enabling children's learning.

Moreover, the findings also display the presence of a range of literacy-based skills, varying in complexity. These include engagement in the following prominent activities such as:

- self-expression (creating meaningful pictures),
- basic speech production and spoken expression (Section 5.5 - Feature 4: Audio Recording),
- storytelling/narrative sequencing (Section 5.4.4),
- alphabet/letter knowledge, writing/typing individual letters, spelling, and phonemic awareness (Section 5.5 - Feature 1: Keyboard),
- vocabulary (Section 5.5 - Feature 3: Associations Panel),
- literary forms/genres. This is indicated by the variation in picture categories and stories (Section 5.4) and the use of voice-visual narrative containing songs, fictional and non-fictional items.
- motivation (reflected from sustained engagement and meaningful learning)

23 skills that enable children to become strong readers

1. basic speech production
2. alphabet/letter knowledge
3. phonemic awareness
4. writing/typing individual letters
5. lower-versus upper-case letter identification
6. print concepts (the form/function of print)
7. vocabulary
8. spelling
9. storytelling/narrative sequencing
10. recognizing sight words
11. reading comprehension
12. rhyming concepts
13. grammar
14. learning a foreign language
15. handwriting
16. reading fluency
17. written expression
18. spoken expression
19. sign language letters
20. sign language words
21. sign language phrases
22. literary forms/genres
23. motivation/love of reading

Figure 5.21: Reading Literacy Skills As Per Report

Figure 5.21 is borrowed from Joan Ganz Cooney Center's report (2015) on literacy apps that highlights 23 important skills constituting reading literacy [72]. However, the bulk of the sampled apps within this report only target around 8 out of 23 skills with most of them being fairly basic and lacking in higher order skills such as self-expression [21].

From the aforementioned activities (11 out of 23), the features in PictureBlocks help target not only basic skills such as phonemic awareness and alphabet knowledge, but may also nurture deeper knowledge-building experiences for literacy such as self-expression and literary forms/genres. In addition, our app doesn't have right or wrong answers, and engagement with the literacy skills such as spelling is open-ended in nature. Children can type words that they are interested in, and may come up with the correct spelling for words they don't know through exploration and invented spelling.

We believe that by applying a picture-driven, constructionist approach (especially via experimenting with digital interactions), we could create opportunities for children to both express themselves meaningfully, as well as engage in other literacy fostering activities within the same space.

6
Conclusion

"You can't connect the dots looking forward;
you can only connect them looking backwards."

Steve Jobs

6.1 Summary

The goals of this thesis were two-fold:

1) to address the gap in picture-driven, constructionist mobile applications that focus on literacy learning in children and,

2) to identify any unique affordances offered by this exploratory approach.

Therefore, we designed, built, and tested PictureBlocks, a mobile app targeted towards children between the ages of 5-9 years. Our design capitalizes on the advances in visual forms of interactions, as well as easier access to images online. By providing an open-ended playground for children's creation, and allowing them to choose amongst a plethora of picture icons, we created avenues for self-expression and explored new forms of play.

Our findings indicate that individual features of the app worked well as intended, but cohesively, they made a bigger difference in enhancing the overall experience of the app. Conclusively, the design of our app qualitatively indicates good engagement and displays a potential for learning, which we can shift our focus to for future studies. Through this work, we hope that app designers, educators and researchers will move towards adopting more picture-driven approaches to constructionist literacy learning.

6.2 *Limitations*

We designed, built, and studied the PictureBlocks app over a limited time period. This restricted the duration of our study to around two weeks. Furthermore, almost all of the work involved was conducted by a single person team - the author, with regular advice from a few expert sources.

In contrast, if we take into account the workings of a standard game studio, there is usually a division of effort amongst artists, game designers, developers, product managers, and other such personnel. There is also better access to resources with regards to generating child-friendly picture icons and designs.

Therefore, with more time and resources, we would have experimented with additional features in our app's design and also be able to study its effects over a longer duration. For example, we would have included more scaffolding techniques to help guide children towards correctly guessing a word's spelling. We would have also allowed more personalization of the sprites children created - enabling them to edit its colors and appearance, or even draw the sprite into the picture. By analyzing play over a longitudinal study duration, we could have also inspected for long-term changes in a child's spelling or word acquisition within the app.

Finally, the PictureBlocks app was tested with a small set of participants who were comfortable with English, the majority of whom were from families belonging to a high socioeconomic status (SES), with affiliations to Harvard or MIT. Consequently, this limited the scope of our study as the participants were not representative of all communities, with little variation in access to literacy resources.

Hence, the efficacy of our study can be improved by diversifying the range and number of our participants, and also include children from non-English speaking or low-SES families - where early literacy development is more greatly necessitated.

6.3 Recommendations

In addition to addressing the aforementioned limitations, we have some design and development recommendations for PictureBlocks, which can also be adopted by learning apps targeting picture-driven literacy development.

Experiment With Fresh Interactions

With advances in technology and modes of communication, as designers and developers, we have access to a potpourri of fresh interactions to choose from. Whether it is gifs, avatars, and animojis when it comes to visual forms of expression, or image recognition, speech-to-text synthesis and machine learning when it comes to technological evolution - we can find ways to integrate these growing number of ideas together elegantly. Especially in educational apps for children, where there are numerous 'chocolate-covered broccoli' approaches, we need to experiment with new ways of engagement, and observe how they impact children's behaviour and learning.

Specifically for PictureBlocks, the inclusion of verbs within the app's vocabulary can bring about new types of play - where adding a verb to an existing sprite can animate it with that action. For example, adding the verb "jump" to a monkey sprite can make the monkey jump up and down on the canvas. Or as another one of our participants P4 (Rohan) suggested - including a math keyboard and adding the number '123' next to the word 'cows' should make 123 cows on the canvas, because "that would be so much fun". For audio recordings, participant P3 (Viella) suggested making the sprite object's mouths move as if they are speaking in the player's voices. All of these are uncommon interactions and children, with their vivid imaginations, themselves seem to be good starting points for these ideas.

Provide More Visibility

If an app offers several features and interactions, it's occasionally easy for children to forget that some options exist or not realize its full capabilities. In such cases, even good learning apps can be underutilized or display repetitive patterns of a single type of use. Moreover, research has indicated that few children's learning apps actually include a parents section and present them with information about what information their child encounters within the app [72] .

As we prepare for deployment, including a first-time user experience within PictureBlocks instructing the child on how to use all of its features will be a positive addition to the app. Plus, we plan on incorporating a parents section within the app which can not only provide information about its contents, but also indicate what kind of words their child is having difficulty to spell using PictureBlocks.

Drive With Direction

Sometimes, children need a little push. Even if we want to move away from instructionist forms of learning, the presence of a teacher does not disappear. Rather, the teacher's role changes to that of a facilitator, guiding the child when they encounter a roadblock or are in need of some inspiration. Similarly, here, the app or other learning technology; while providing an open-ended environment with children taking the lead, should also be able to direct them when help is needed.

For PictureBlocks, we hope to include personalized scaffolding for children, where the app learns the behaviour of the child and deciphers hindrances in literacy skills or repetitive patterns in usage. Upon doing so, the app can provide relevant direction to each player, nurturing their individual strengths and improving on weaknesses. Also, we can add some structure to PictureBlocks' open-ended play by leveling the app. Adding levels of difficulty may boost motivation and cater to a wider range of children who are at different skill levels.

Create A Social Machine

Simply put, learning is inherently social. The benefits of collaborative learning have been known for decades for improving critical thinking skills [15]. Further, young children learn best when another caring individual joins the process. The proliferation in communication technology broadens the ways in which we can learn together with another individual, and allows for a range of mediated social interactions (some yet to be explored).

Therefore, by combining the merits of social collaboration with the unique offerings of technology, we can create a distinct "social machine" that enriches a child's learning experience by providing them with the best of both worlds.

Within PictureBlocks, we have already implemented such a "social machine" via the Social Sharing feature. This mediated form of social interaction has shown positive preliminary results in terms of engagement and has even displayed signs of scaffolding with the exchange of new words between friends.

However, despite this digital age, significant changes in children's learning cannot be achieved with technology alone. To reach the full potential of technology, we also need to enable more one-on-one interactions between children and other caring individuals. Having features within PictureBlocks which require children to engage with the outside world - such as taking pictures of real-life objects which get converted to sprites within the app, or motivating parents to make pictures with their children together will create more sustainability.

Within our Laboratory for Social Machines, we are already working on these efforts and hope to establish a "Family Learning Network" among communities in the Greater Boston Area. As part of this network, we introduce a new role of a family learning coach - someone who can analyze data from our literacy focused apps, and form a relationship with the child and parent to promote more visibility and collaboration in the child's learning processes. Our hope is to integrate PictureBlocks within this network of smart, expressive apps. The below diagram (Fig 6.1) depicts this network and displays the role of PictureBlocks and the relationships between stakeholders in a child's learning trajectory.

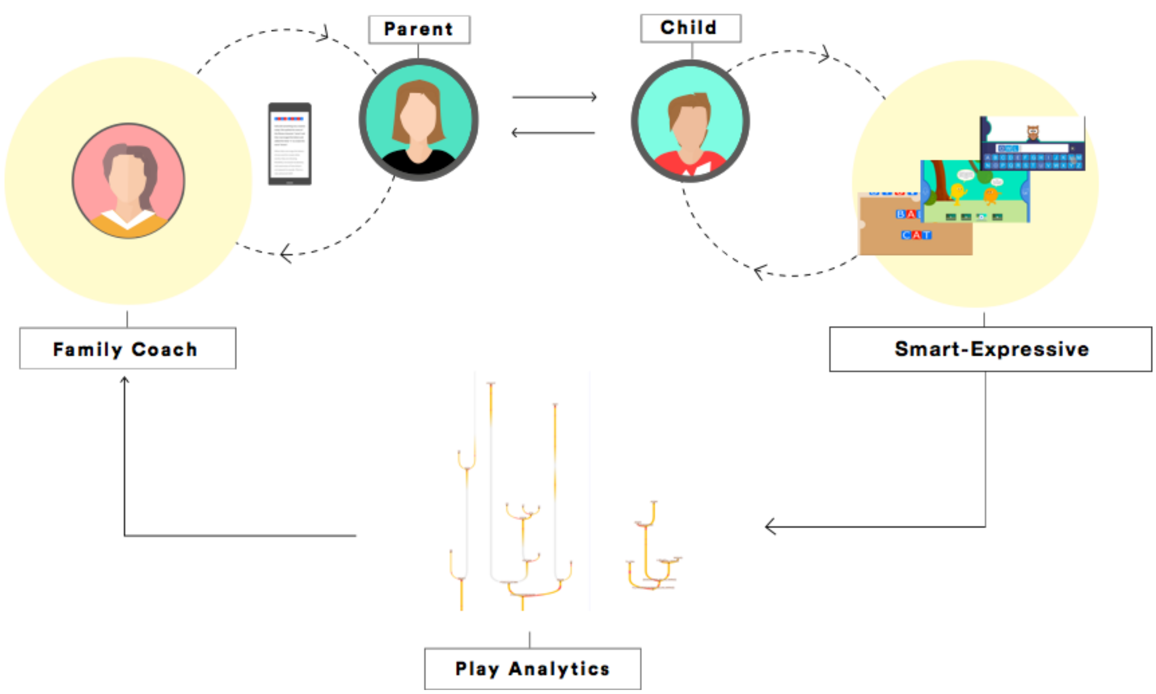


Figure 6.1: The Family Learning Network depicting the interactions between the literacy apps and the stakeholders involved in a child's learning trajectory.

6.4 Reflection

Designing an app for children's literacy is hard. Designing an app for children's literacy that is engaging is even harder. Designing, developing, playtesting, studying, and writing about an engaging app for children's literacy is the hardest of them all. With distractions galore in the form of apps and other entertainment, sometimes kids may get tired of even the most carefully constructed app that takes into account best practices and established research. On the other hand, a simple app could go viral overnight if it contains the right combination of the right things at the right time. Especially in the genre of serious apps, and games for purpose, it is an acknowledged challenge to strike a balance between fun and learning. Between creativity and literacy. Between consumption and creation.

With the help of the humble yet powerful picture, we hope that through this work, we have managed to take tiny sprite steps towards bridging these gaps. With PictureBlocks, our goals were never for children to simply type or learn new words alone, but rather to use their words as a medium for exploring their passions, communicating their ideas, and exhibiting their creativity. This is the true purpose that language was created for in the first place, and not just learning language for the sake of learning. Our hope is that this research has laid the groundwork for picture-driven constructionist approaches within literacy apps, and that in the future, we can run non-pilot, controlled experiments to potentially assess these approaches with respect to traditionally implemented ones.

Personally for the author, the biggest revelation was the surprising variety of ways in which children played with PictureBlocks. Each word, sprite, voice recording, and picture created was a reflection of their individuality - making the author laugh, be surprised, or even confused, but mostly forming a meaningful connection with every child, by just looking at their play. This further reinforces the idea that there is no one-size-fits-all design when it comes to children's learning. Every child is unique and if an app can make a positive difference in even one of their lives, it should be considered a success.

Bibliography

[1] Inc. Apple. *How to use QuickTime Player*. 2018. URL: https://support.apple.com/en-us/HT201066 (visited on 07/31/2018).

[2] Inc. Apple. *Use Animoji*. 2018. URL: https://support.apple.com/en-us/HT208190 (visited on 07/31/2018).

[3] Inc. Apple. *Use Emoji*. 2018. URL: https://support.apple.com/en-us/HT202332 (visited on 07/31/2018).

[4] Marcia Baghban. "Scribbles, labels, and stories: The role of drawing in the development of writing". In: *YC Young Children* 62.1 (2007), p. 20.

[5] W Steven Barnett. "Preschool education for economically disadvantaged children: Effects on reading achievement and related outcomes". In: *Handbook of early literacy research* 1 (2001), pp. 421–443.

[6] Elena Bodrova and Deborah Leong. "Tools of the mind: The Vygotskian approach to early childhood education". In: (2007).

[7] Warner Bros. *ScribbleNauts Unlimited*. 2018. URL: https://itunes.apple.com/us/app/scribblenauts-unlimited/id1030902626 (visited on 07/31/2018).

[8] Roger Brown and Jean Berko. "Word association and the acquisition of grammar". In: *Child Development* (1960), pp. 1–14.

[9] Nation's Report Card. *National Achievement-Level Results*. 2018. URL: https://www.nationsreportcard.gov/reading_2017/#/nation/achievement?grade=4 (visited on 07/31/2018).

[10] Elizabeth Coates and Andrew Coates. "Young children talking and drawing". In: *International Journal of Early Years Education* 14.3 (2006), pp. 221–241.

[11] Anne Haas Dyson. "The Emergence of Visible Language: Interrelationships Between Drawing and Early Writing." In: (1983).

[12] Johanna Einarsdottir, Sue Dockett, and Bob Perry. "Making meaning: Children's perspectives expressed through drawings". In: *Early child development and care* 179.2 (2009), pp. 217–232.

[13] Wilbert O Galitz. *The essential guide to user interface design: an introduction to GUI design principles and techniques*. John Wiley & Sons, 2007.

[14] Keith Gentle. *Children and art teaching*. Taylor & Francis, 1985.

[15] Anuradha A Gokhale. "Collaborative learning enhances critical thinking". In: *Volume 7 Issue 1 (fall 1995)* (1995).

[16] Google. *AutoDraw*. 2018. URL: `https://experiments.withgoogle.com/autodraw` (visited on 07/31/2018).

[17] Google/first20hours. *Google-10000-English*. 2018. URL: `https://github.com/first20hours/google-10000-english` (visited on 07/31/2018).

[18] Gilbert R Gredler. "Snow, CE, Burns, MS, & Griffin, P.(eds.)(1998). Preventing reading difficulties in young children. Washington, DC: National Academy Press, 432 pp., 35.95". In: *Psychology in the Schools* 39.3 (2002), pp. 343–344.

[19] Annette M de Groot. "Representational aspects of word imageability and word frequency as assessed through word association." In: *Journal of Experimental Psychology: Learning, Memory, and Cognition* 15.5 (1989), p. 824.

[20] Lisa Guernsey and Michael H Levine. *Tap, click, read: Growing readers in a world of screens*. John Wiley & Sons, 2015.

[21] Lisa Guernsey et al. "Pioneering literacy in the digital wild west: Empowering parents and educators". In: *Washington, DC: Campaign for Grade-Level Reading* (2012).

[22] Idit Ed Harel and Seymour Ed Papert. *Constructionism.* Ablex Publishing, 1991.

[23] Willard W Hartup. "Social relationships and their developmental significance." In: *American psychologist* 44.2 (1989), p. 120.

[24] Audrey A Hazamy. "Influence of Pictures on Word Recognition". In: (2009).

[25] Kathy Hirsh-Pasek et al. "Putting education in "educational" apps: Lessons from the science of learning". In: *Psychological Science in the Public Interest* 16.1 (2015), pp. 3–34.

[26] Originator Inc. *Endless Wordplay*. 2018. URL: https://itunes.apple.com/us/app/endless-wordplay/id727871636 (visited on 07/31/2018).

[27] Connie Juel. "Learning to read and write: A longitudinal study of 54 children from first through fourth grades." In: *Journal of educational Psychology* 80.4 (1988), p. 437.

[28] Yasmin B Kafai and Mitchel Resnick. "Introduction". In: *Constructionism in practice*. Routledge, 2012, pp. 13–20.

[29] Donna Darling Kelly. *Uncovering the history of children's drawing and art*. Greenwood Publishing Group, 2004.

[30] kiddoware. *Kid's Place - Parental Control*. 2018. URL: https://play.google.com/store/apps/details?id=com.kiddoware.kidsplace (visited on 07/31/2018).

[31] Valerie E Lee and David T Burkam. *Inequality at the starting gate: Social background differences in achievement as children begin school.* ERIC, 2002.

[32] Pearson Education Limited. *Longman Childrenś Picture Dictionary*. 2018. URL: https://www.pearsonelt.com/catalogue/primary/lcpd.html (visited on 07/31/2018).

[33] Matt Mahoney. *About the Test Data*. 2018. URL: http://mattmahoney.net/dc/textdata.html (visited on 07/31/2018).

[34] Morton S Malter. "Children's preferences for illustrative materials". In: *The Journal of Educational Research* 41.5 (1948), pp. 378–385.

[35] Miriam Martinez and Janis M Harmon. "Picture/text relationships: An investigation of literary elements in picturebooks". In: *Literacy Research and Instruction* 51.4 (2012), pp. 323–343.

[36] Sylvia Libow Martinez and Gary Stager. *Invent to learn: Making, tinkering, and engineering in the classroom*. Constructing modern knowledge press Torrance, CA, 2013.

[37] Jeff McQuillan. *The literacy crisis: False claims, real solutions*. Heinemann, 1998.

[38] Common Sense Media. *App Reviews*. 2018. URL: https://www.commonsensemedia.org/app-reviews (visited on 07/31/2018).

[39] Common Sense Media. *Draw Something App Review*. 2018. URL: https://www.commonsensemedia.org/app-reviews/draw-something (visited on 07/31/2018).

[40] Common Sense Media. *Endless Wordplay App Review*. 2018. URL: `https://www.commonsensemedia.org/app-reviews/endless-wordplay` (visited on 07/31/2018).

[41] Common Sense Media. *Montessorium: Intro to Words App Review*. 2018. URL: `https://www.commonsensemedia.org/app-reviews/alpha-writer-by-montessorium` (visited on 07/31/2018).

[42] George A Miller. "WordNet: a lexical database for English". In: *Communications of the ACM* 38.11 (1995), pp. 39–41.

[43] George A Miller et al. "Introduction to WordNet: An on-line lexical database". In: *International journal of lexicography* 3.4 (1990), pp. 235–244.

[44] Konstantin Mitgutsch and Narda Alvarado. "Purposeful by design?: a serious game design assessment framework". In: *Proceedings of the International Conference on the foundations of digital games*. ACM. 2012, pp. 121–128.

[45] Maria Montessori. *The Advanced Montessori Method...* Vol. 1. Frederick A. Stokes Company, 1917.

[46] LLC Montessorium. *Montessorium: Intro to Words*. 2018. URL: `https://itunes.apple.com/us/app/montessorium-intro-to-words/id394038232` (visited on 07/31/2018).

[47] Juliana Nazare et al. "Bilingual SpeechBlocks: Investigating How Bilingual Children Tinker with Words in English and Spanish". In: *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*. ACM. 2017, pp. 183–193.

[48] Douglas L Nelson, Cathy L McEvoy, and Thomas A Schreiber. "The University of South Florida free association, rhyme, and word fragment norms". In: *Behavior Research Methods, Instruments, & Computers* 36.3 (2004), pp. 402–407.

[49] Penny Oldfather and Allan Wigfield. "Children's motivations for literacy learning". In: *Developing engaged readers in school and home communities* (1996), pp. 89–113.

[50] OMGPOP. *Draw Something Classic*. 2018. URL: `https://itunes.apple.com/us/app/draw-something-classic/id488628250` (visited on 07/31/2018).

[51] Seymour Papert and Idit Harel. "Situating constructionism". In: *Constructionism* 36.2 (1991), pp. 1–11.

[52] Jeffrey Pennington, Richard Socher, and Christopher Manning. "Glove: Global vectors for word representation". In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014, pp. 1532–1543.

[53] Mike Petrich, Karen Wilkinson, and Bronwyn Bevan. "It looks like fun, but are they learning". In: *Design, make, play: Growing the next generation of STEM innovators* (2013), pp. 50–70.

[54] Meredith M Pike, Marcia A Barnes, and Roderick W Barron. "The role of illustrations in children's inferential comprehension". In: *Journal of experimental child psychology* 105.3 (2010), pp. 243–255.

[55] Hayes Raffle et al. "Jabberstamp: embedding sound and voice in traditional drawings". In: *Proceedings of the 6th international conference on Interaction design and children*. ACM. 2007, pp. 137–144.

[56] Jamie Reilly and Jacob Kean. "Formal distinctiveness of high-and low-imageability nouns: Analyses and theoretical implications". In: *Cognitive science* 31.1 (2007), pp. 157–168.

[57] Mitchel Resnick. "Computer as paint brush: Technology, play, and the creative society". In: *Play= learning: How play motivates and enhances children's cognitive and social-emotional growth* (2006), pp. 192–208.

[58] Mitchel Resnick. "Give P's a chance: Projects, peers, passion, play". In: *Constructionism and creativity: Proceedings of the Third International Constructionism Conference. Austrian Computer Society, Vienna*. 2014, pp. 13–20.

[59] Children's Technology Review. *Objective Reviews of Children's Interactive Media*. 2018. URL: `http://reviews.childrenstech.com/ctr/home.php` (visited on 07/31/2018).

[60] Donald J Richgels. "Invented spelling, phonemic awareness, and reading and writing instruction". In: *Handbook of early literacy research* 1 (2001), pp. 142–155.

[61] Victoria Rideout. "The Common Sense census: Media use by kids age zero to eight". In: *Robb, M., Ed* (2017).

[62] Katie Salen, Katie Salen Tekinbaş, and Eric Zimmerman. *Rules of play: Game design fundamentals*. MIT press, 2004.

[63] S Jay Samuels. "Effects of pictures on learning to read, comprehension and attitudes". In: *Review of Educational Research* 40.3 (1970), pp. 397–407.

[64] Joseph Sanacore. "Promoting lifetime literacy through authentic self-expression and intrinsic motivation". In: *Journal of Adolescent & Adult Literacy* 40.7 (1997), pp. 568–571.

[65] Mina Soltangheis. "From children's play to intentions: a play analytics framework for constructionist learning apps". PhD thesis. Massachusetts Institute of Technology, 2017.

[66] Autism Speaks. *Pogg*. 2018. URL: `https://www.autismspeaks.org/autism-apps/pogg` (visited on 07/31/2018).

[67] Susan Steffani and Paula M Selvester. "The relationship of drawing, writing, literacy and math in kindergarten children". In: *Reading Horizons* 49.2 (2009), p. 4.

[68] Ivan Sysoev et al. "SpeechBlocks: A Constructionist Early Literacy App". In: *Proceedings of the 2017 Conference on Interaction Design and Children*. ACM. 2017, pp. 248–257.

[69] I Sysoev et al. "Exploring SpeechBlocks: Piloting a Constructionist Literacy App with Preschool Children". In: ().

[70] Inc. Tinybop. *Apps Index*. 2018. URL: `https://tinybop.com/apps` (visited on 07/31/2018).

[71] Sensor Tower. *What is Sensor Tower?* 2018. URL: `https://help.sensortower.com/article/23-what-is-sensor-tower` (visited on 07/31/2018).

[72] Sarah Vaala, Anna Ly, and Michael H Levine. "Getting a Read on the App Stores: A Market Scan and Analysis of Children's Literacy Apps. Full Report." In: *Joan Ganz Cooney Center at Sesame Workshop*. ERIC. 2015.

[73] Ricky Vuckovic. *Pogg - kid's game for language*. 2018. URL: `https://itunes.apple.com/us/app/pogg-kids-game-for-language/id330964517` (visited on 07/31/2018).

[74] Susan Wright. "Graphic-Narrative Play: Young Children's Authoring through Drawing and Telling." In: *International Journal of Education & the Arts* 8.8 (2007), pp. 1–28.

[75] Jennifer M Zosh, Meredith Brinster, and Justin Halberda. "Optimal contrast: Competition between two referents improves word learning". In: *Applied Developmental Science* 17.1 (2013), pp. 20–28.

Credits

The following sources deserve credit for their work that was used directly within this thesis:

- Ivan Sysoev and Anneli Hershman for their initial work on SpeechBlocks and its design components, that served as inspiration for PictureBlocks.
- Juliana Nazaré for her work on the SpeechBlocks font used within PictureBlocks.
- Marc Exposito for his well designed illustration on Family Learning Networks (Fig 6.1).
- Figures 2.1 and 3.18 are borrowed from the research paper - "SpeechBlocks: A Constructionist Early Literacy App" [1].
- Figure 5.20 is borrowed from the research paper - "Putting education in 'educational' apps: Lessons from the science of learning" [2].
- Figure 5.21 is borrowed from the research paper - "Getting a Read on the App Stores: A Market Scan and Analysis of Children's Literacy Apps. Full Report." [3].
- The authors on the FlatIcon website whose open-sourced icons made all the sprites and buttons within PictureBlocks possible.
- The Personal Robots group within the MIT Media Lab whose mailing list for parents helped recruit the majority of the participants for our studies.
- Ini Oguntola for doing extraordinary work as a UROP and coding all the server stuff within PictureBlocks, and then some.
- Agnish Balasubramani, for his quick and wonderful prototypes that helped enhance PictureBlocks' user experience.
- All the children who participated in playtesting sessions and the pilot for PictureBlocks, and whose pictures added to the beauty and playfulness of this thesis.

[1] Ivan Sysoev et al. "SpeechBlocks: A Constructionist Early Literacy App". In: *Proceedings of the 2017 Conference on Interaction Design and Children*. ACM. 2017, pp. 248–257

[2] Kathy Hirsh-Pasek et al. "Putting education in "educational" apps: Lessons from the science of learning". In: *Psychological Science in the Public Interest* 16.1 (2015), pp. 3–34

[3] Sarah Vaala, Anna Ly, and Michael H Levine. "Getting a Read on the App Stores: A Market Scan and Analysis of Children's Literacy Apps. Full Report." In: *Joan Ganz Cooney Center at Sesame Workshop*. ERIC. 2015

A
Questionnaires

A.1 Playtesting - App

Child Questionnaire **App**

Name: ID

Age:

Thanks for playing! I'm going to ask you some questions about what you thought of the app, and some questions about words and pictures. Are you ready?

1. What did you think about the game, PictureBlocks?

   a. What did you like? What was fun?

   b. What did you not like? What wasn't fun?

   c. What would you change about it?

2. Were there any parts that were boring to you?

   a. What was boring?

   b. How could we make it more fun?

3. Was this game too easy, too hard, or somewhere in between?

4. What was your favorite word to make using PictureBlocks?

5. Was there any word that you wanted to make but couldn't?

6. Would you be interested in playing with PictureBlocks at home?
   If yes, what kind of words would you try to make the next time?

7. Do you like to draw and make pictures? How do you think Picture Blocks is different from that? Which one would you prefer?

Child Questionnaire **App**

8. When you saw a picture appearing when you typed a word, how did it make you feel?

9. When you finished saving your picture, how did it make you feel?

10. Do you want to share the picture you made on PictureBlocks with someone?

    a. With whom?

    b. Would you tell your friend about it?

    c. How would you describe it to them?

11. Have you played anything like this before? What?

12. Did you have any difficulty spelling any words? Which ones/what happened?

13. When you tapped on a picture, you saw some other pictures on top. What reason do you think that was for?

14. What do you think is the point of Picture Blocks?

    a. Did you learn anything new?

    b. Anything else you want to share with me?

Thank you so much for your help, you had some really great ideas for how we can improve PictureBlocks!

A.2 Playtesting - UX

Child Questionnaire **UX**

Name: ID

Age:

You're almost there! I'm now going to ask you some questions about the design of the app. The design of the app means the buttons, the colors, how easy it was to play with, and things like that. Are you ready?

1. What do you think of the colors of the app? ______

2. Was there any part of the app that made you really frustrated?
   a. What happened? ______
   b. How could we make it better? ______

3. Did anything seem confusing while playing ? What and why? ______

4. What did you think about the pictures?
   a. Did you like them? Why? ______
   b. Did you not like them? Why? ______
   c. What would you change about them? ______

5. What did you think about the keyboard?
   a. What did you like or not like? ______
   b. What would you change about it? ______

6. Which keyboard did you like using - the QWERTY or ABC one? Why? ______

7. What was the easiest part of the game? Why? ______

Child Questionnaire **UX**

8. What was the hardest part? Why? ______

9. What did you think about the backgrounds? ______

10. What did you think about the pictures that appeared on top of other pictures?
    a. Do you think those pictures were related in any way? ______
    b. If not, why did they not make any sense? ______
    c. What was your first thought when you saw those pictures? ______
    d. Were you curious about how to make those pictures that appeared on top? ______

A.3 Pilot - Exit Interview

EXIT INTERVIEW

1. What did you think about PictureBlocks?

2. What do you think PictureBlocks does?

3. What was your most favorite thing about the app?

4. What was your least favorite thing about the app?

5. What do you think this panel is for?
   (researcher points to Associations Panel)

6. Did you share any pictures with your friends/siblings?
   Explain.

7. If you had to change something about the app or
   add a new feature, what would you add?

8. Did you learn anything new with PictureBlocks?
   Explain.

9. How would you describe PictureBlocks to your friend?

10. Would you like to play with it again?

B
Consent Form

Promoting Literacy through Playful Words' Apps and Tools for Children and Families,
Phase 3 (applicable to apps not integrated into the social learning system)

**The following privacy policy will serve as an in-app consent form that will be signed by families during the on-boarding process inside of the Playful Words app.*

Privacy Policy and Consent to Participate in a Research Study

We have created literacy learning apps so that children can explore words, sentences, and stories through a playful learning experience. The purpose of this study is to help us understand how children use our apps in the home. We understand how important privacy is to our community, especially children and their parents. We wrote this privacy policy to explain what information we collect, how we use it, and what we're doing to keep it safe. If you have any questions regarding this privacy policy, you can contact us at playfulwords@media.mit.edu.

Consent to participate in a research study

Your child's participation in this study is completely voluntary and you are free to choose whether they can be in it or not. By accepting these terms on the Playful Words apps, you consent for your child to be part of this study. If you choose your child to be in this study, you may subsequently withdraw them from it at any time without penalty or consequences of any kind by emailing us at playfulwords@media.mit.edu.

If your child is a participant in this study, we ask that you, the parent, give us permission to collect information about how your child interacts with our Playful Words apps and allow a researcher to access and analyze the data collected for research purposes. Our app may be designed such that it involves your child recording audio snippets using their voice while playing with the app. The app may also allow sharing of information in the form of pictures, words, stories, or audio between children who are friends within the app. If any information is being shared amongst children, it will be moderated by the researchers of the study before being shared for safety reasons. We ask that you, as the parent, also provide your consent for such information within the app to be shared and collected. You can also choose for your child to opt-out of the study at any time without consequence, by emailing us at playfulwords@media.mit.edu.

AMENDMENT APPROVED 23-MAR-18

APPROVED 14-Sep-2017 - MIT IRB PROTOCOL # 1708070811 - EXPIRES ON 13-Sep-2018

There will be no monetary compensation given for participation in this study. If you have any questions or concerns about the study, please email the research team at playfulwords@media.mit.edu.
The duration of the study can be between one and six weeks, but your child is welcome to continue playing with the Playful Words apps after the study is complete. Please note that we will still be collecting data on your child's activity within the app after the study, unless you opt-out. Since we are looking to see how children play with the Playful Words apps in their home environments, there is no prescribed amount of play per day during the study.

If you are borrowing an Android/iOS device from the Laboratory for Social Machines research group, you are responsible for the use and care of the device during the study. However, you are not liable for any major damages, or lost or stolen devices. If there is any damage to the device outside of regular use, or if your device is lost or stolen, please immediately contact the research team at playfulwords@media.mit.edu.

What information do the Playful Words apps collect?

We collect some data on where you click and which parts of the app you use. Our apps may also collect data such as audio snippets of your child's voice recorded in-app or information shared between children. Altogether, this data helps us figure out ways to improve the app, to understand how your child plays with the app, how children interact with each other using the app, and to conduct research to find patterns about your child's progress. Please note that all click data is de-identified and anonymous. The data is associated with a unique device number, and does not contain any personally identifiable information about your child. The data will be stored on a secure server, and will only be accessed by the researchers on this project. All information will be stored for up to three years after the completion of this study. After three years, all data from this study will be deleted from the server.

We may also collect other information you (the parent) directly provide to researchers, such as when you send us an email, fill out a form, provide or request information, or otherwise submit information during the study. Such information may include personal information such as your name, email, and telephone number. In addition, from time to time we may collect demographic and personal information you provide in connection with your participation in surveys and other activities in connection with the Service. Please note that no personally identifying information will be linked with your child at all and the app will not collect any data outside of this study (i.e. calls, SMS, etc.).

AMENDMENT APPROVED 23-MAR-18

APPROVED 14-Sep-2017 - MIT IRB PROTOCOL # 1708070811 - EXPIRES ON 13-Sep-2018

Although some of the Playful Words apps use an outside service provider, such as the Acapela Group, for speech synthesis, our apps do not share any personally identifiable information collected with the Acapela Group, and the Acapela Group does not collect any personally identifying information about you.

How does the Playful Words Team use the information it collects?

- We may use the information in research studies intended to improve our understanding of how people learn with our Playful Words apps. The results of this research are shared with educators and researchers through conferences, journals, and other publications.
- We analyze the information to understand and improve our apps, such as determining which pre-made words, images, or stories are most popular, which sounds are most used, and how long users spend in the app.
- Other than the cases described above, we will never share personally identifiable information about you with any other person, company, or organization, except:
  - As required to comply with our obligations under the law.
  - For technical reasons, if we are required to transfer the data on our servers to another location or organization.

Playful Words is a project from the MIT Media Lab's Laboratory for Social Machines. If you have any questions about this policy or any concerns about your privacy, please email us at playfulwords@media.mit.edu or contact us at: Laboratory for Social Machines, MIT Media Lab, 75 Amherst St. Cambridge MA 02139. Tel: 1-617-324-4914.

You are not waiving any legal claims, rights or remedies because of your participation in this research study. If you feel you have been treated unfairly, or you have questions regarding your rights as a research subject, you may contact the Chairman of the Committee on the Use of Humans as Experimental Subjects, M.I.T., Room E25-143B, 77 Massachusetts Ave, Cambridge, MA 02139, Tel: 1-617-253 6787.

❏ I Agree

(by clicking “I Agree”, you are acknowledging that you have read the privacy policy and consent to participate in the Playful Words study)

❏ I Do Not Agree

AMENDMENT APPROVED 23-MAR-18

APPROVED 14-Sep-2017 - MIT IRB PROTOCOL # 1708070811 - EXPIRES ON 13-Sep-2018

C
Flyer


Massachusetts Institute of Technology
mit media lab
Picture Blocks
Does your child enjoy making pictures and learning new words?
We are a part of the Social Machines Group at the MIT Media Lab and we are interested in exploring how children associate pictures and words.
Child participants in this study will use the PictureBlocks app at home for 14 days. At the end of the study, they will answer a questionnaire and get to take home their created pictures and participation goodies.
Participants must be 5-9 years old. We will provide the mobile phone with the app installed for the study.
How do I participate?
To participate or if you have any questions, please contact Sneha:
Email snehapm@mit.edu.
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu
Contact: snehapm@mit.edu

D
Gallery

Selected pictures and stories created by children with PictureBlocks.

Figure D.1: Callum, Age 5

Figure D.2: Jacob, Age 5

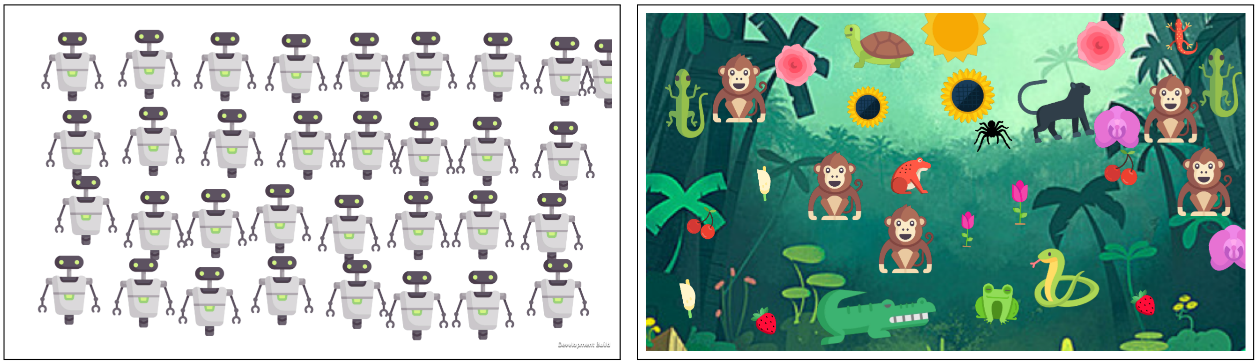


Figure D.3: Viella, Age 7

Figure D.4: Niven, Age 5

Figure D.5: Julia, Age 7

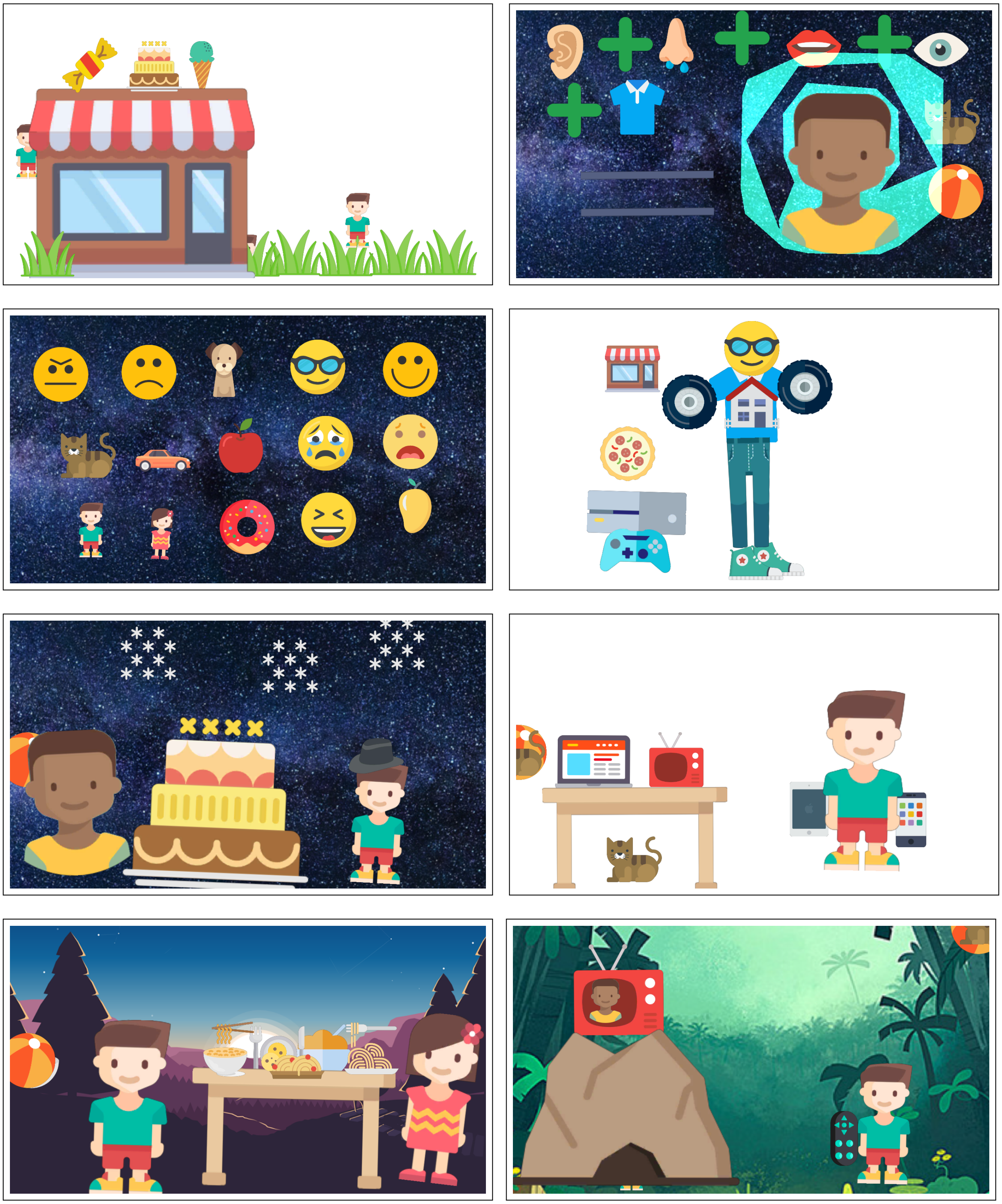


Figure D.6: Siddharth, Age 9

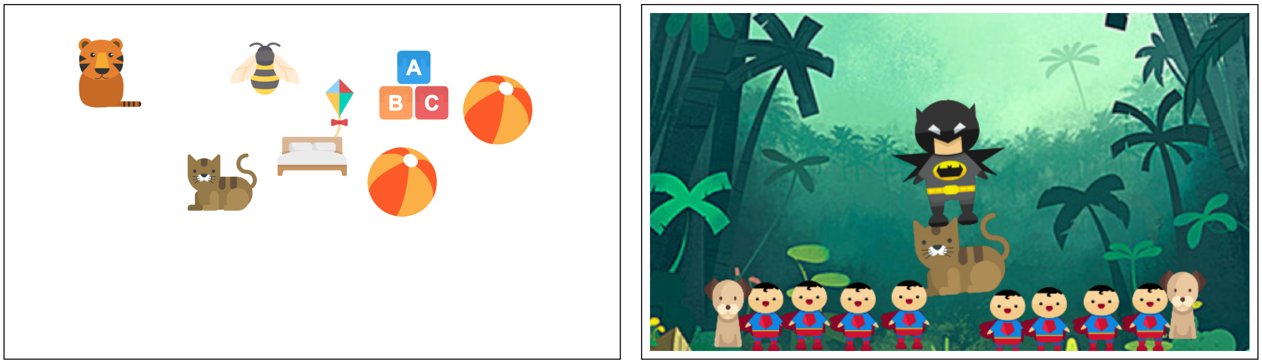


Figure D.7: Presley, Age 7

Figure D.8: Ellery, Age 7

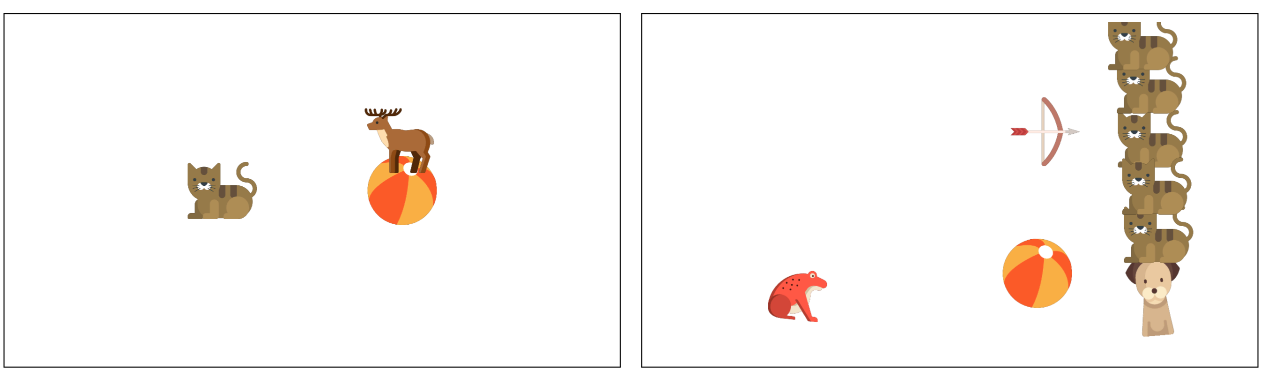


Figure D.9: Rohan, Age 8

Figure D.10: Howard, Age 7

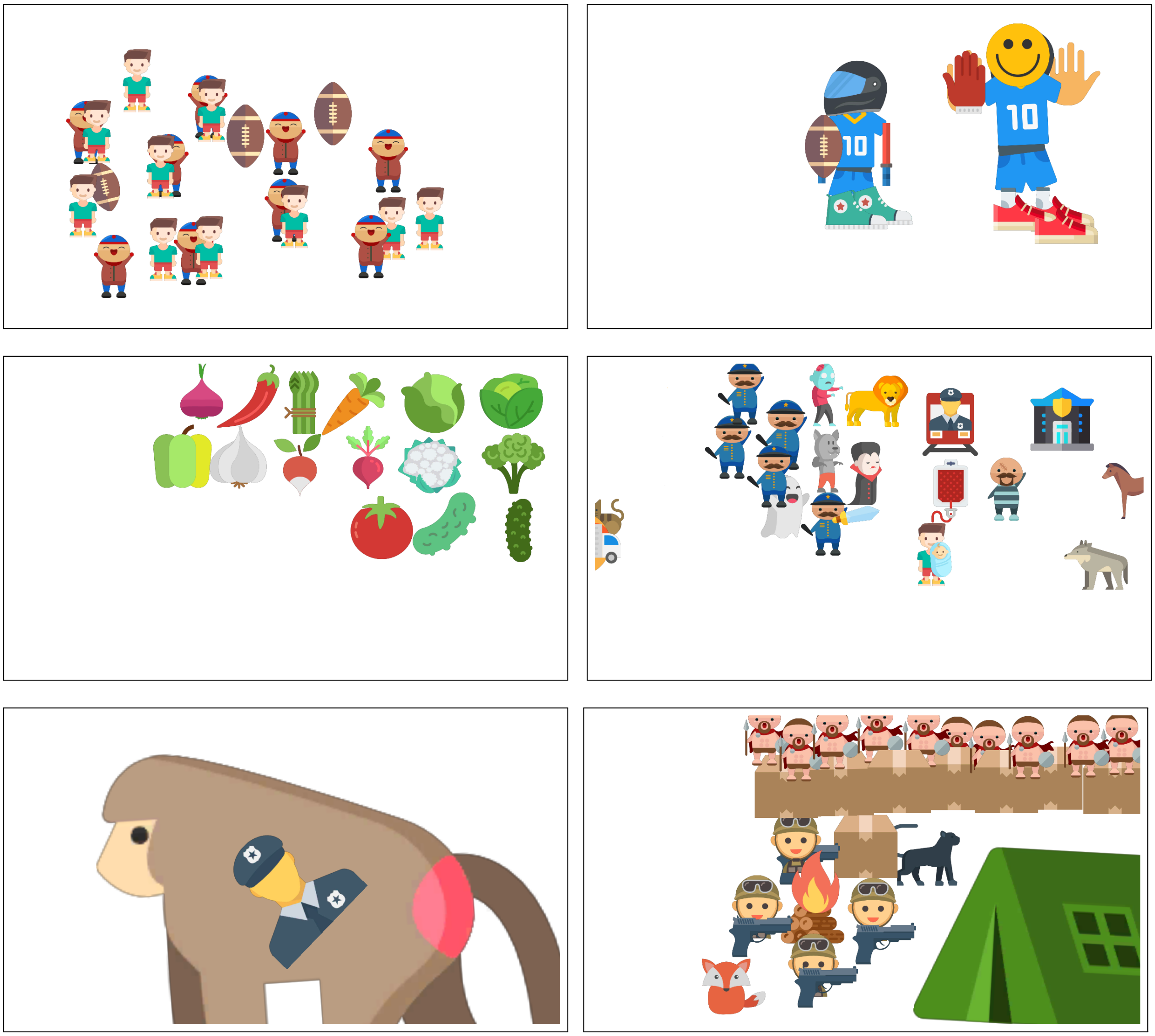


Figure D.11: Howard, Age 7

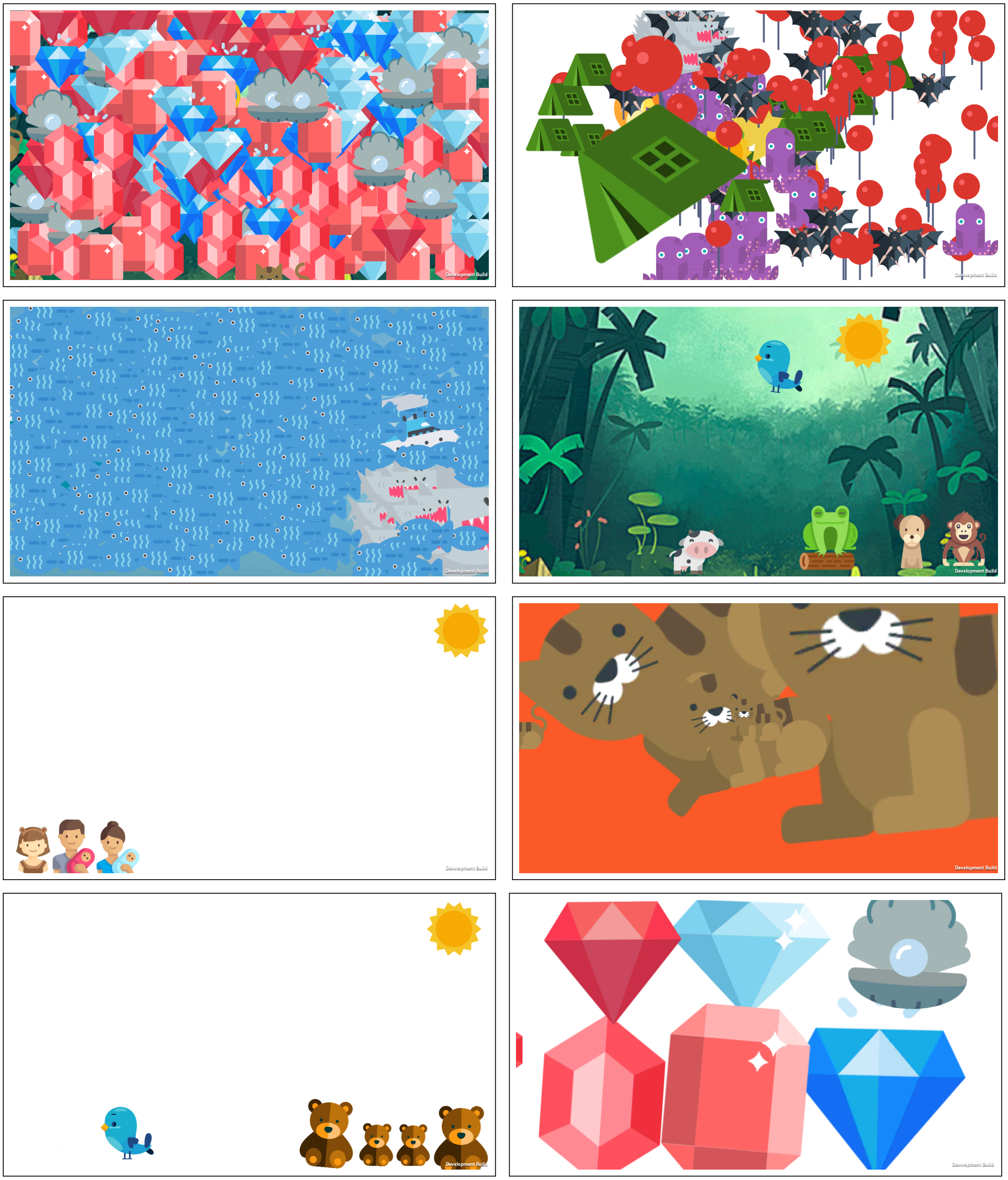


Figure D.12: Tyler, Khloe and Kalon, Ages 7, 5, 5