RFID Pathways
Continuity Report

RFID Team SEE 2017

UC San Diego
Jacobs School of Engineering

EnVision
Arts and Engineering Maker Studio

BIRCH AQUARIUM
at Scripps Institution of Oceanography
UC San Diego

HOLOGIC®

Client:
Birch Aquarium at Scripps Institution of Oceanography

Team Members:
Bryle Castro
Josh Duhay
James Wyatt Guidry
Samm Iwamasa
Matthew Rice

Project Managers:
Jesse DeWald (Lead)
Adam Johnson
Paul Llanura
Daniel Yang

Donor:
Hologic Corporate Philanthropy

Table of Contents

Abstract

To enhance the visitor experience, museums and aquariums are implementing new features into exhibits to make them more interactive and personalized so that the users can immerse themselves into the content. In collaboration with Birch Aquarium at Scripps Institution of Oceanography, our team was given the task of using RFID technology to create a more engaging, personalized, and interactive experience for the visitors. With RFID technology we keep track of visitor preferences in order to present them with information and content that matches their preferences. Our team developed a website to deliver this enhanced visitor experience. In addition to improving user experiences, our system also grants the administrator the ability to easily update the content that could be displayed on the terminals.

1. System Overview

1.1 Introduction and Motivation

Traditional museum displays are typically static signs mounted to a wall. They are not very engaging, are hard to update, and cannot easily cater to different languages, education levels, or personal preferences. We also observed that few visitors at the Birch Aquarium actually read the exhibit displays. The purpose of this project is to create a proof of concept to explore the potential of using digital sign displays and RFID (Radio Frequency Identification) technology to create personalized, interactive, and engaging experiences for visitors. This digital system allows for infinite content that can easily updatable by Birch staff.

RFID utilizes electromagnetic fields to identify and track tags attached to objects. These tags can be embedded into cards or tickets. By giving visitors their own RFID card, we can provide them with a personalized experience. To demonstrate the potential for personalization, we created the concept of a “Professional Pathway Experience,” which allows visitors to learn about exhibits from the perspective off one of the currently provided professions: Engineer, Biologist and Climatologist.

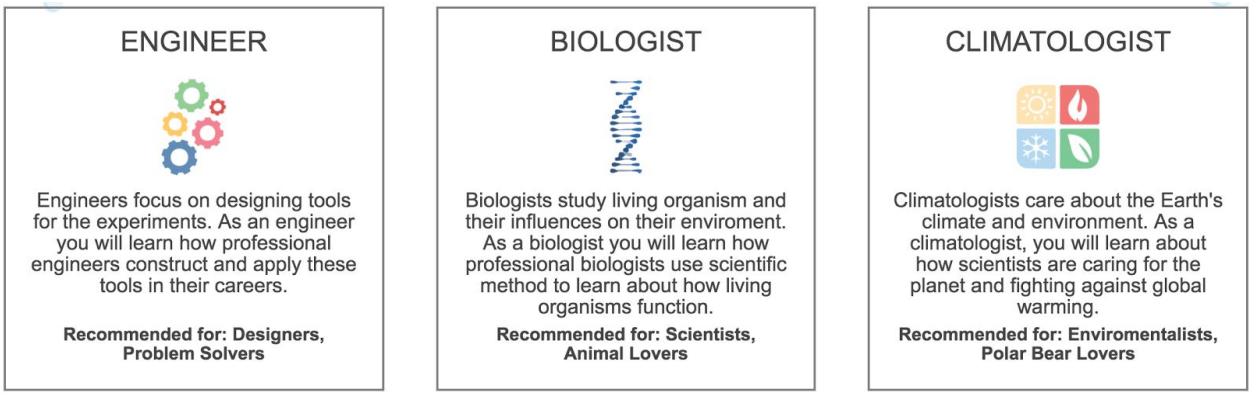

Figure 1. Professional Pathway Options

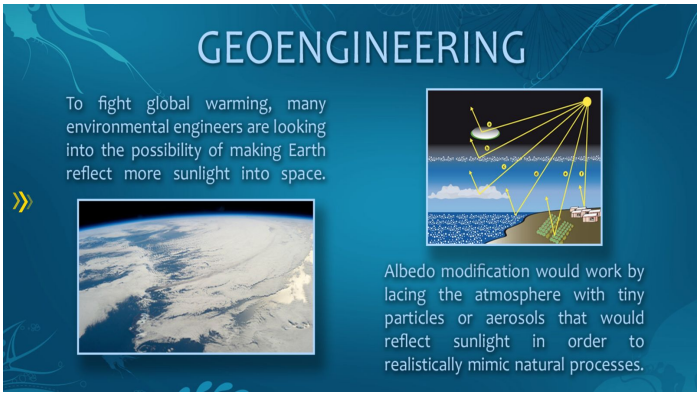

Figure 2. Engineer Professional Pathway Slide for Albedo Exhibit

We decided to deliver the Professional Pathway concept because we believed that it would provide a personalized experience for each visitor by giving them information that matched their interests. Our project aims to inspire greater insight into Scripps' research and educate visitors by targeting their individual interests and displaying information tailored to these interests.

1.2 RKAT Introduction

The Registration Kiosk, Admin, and Terminal system (RKAT) is the hardware, software, and website that our project consists of. The centerpiece of the the RKAT system is our website. The website contains the various Professional Pathway web pages, a registration web page, and an admin web page for updating terminal content.

A web-based solution was chosen to act as the backbone of the project because it builds upon the work of the previous team, allows for intuitive content updates, is scalable, and offers a centralized system.

The RKAT uses Firebase as its online database. It will store user information and preferences, terminal information, and statistical data. Firebase allows for real time data transfer to and from the terminals, as well as content updates from the admin web page. Content uploaded through the admin web page is kept in Firebase Storage. The terminals display the content via the Professional Pathway web pages.

Each terminal and registration kiosk consists of a touchscreen monitor, an RFID reader, and a Raspberry Pi (RPI). The RPI accesses the web and runs HTML files that are stored on it. The terminals also have a set of speakers. The terminal and registration kiosks will be placed on mounting stands; registration kiosks will be placed at the front of the aquarium while terminals will be placed next to various exhibits.

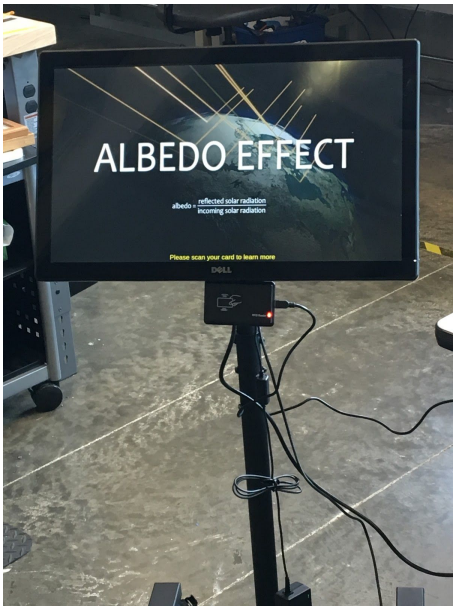

Figure 3. Terminal Display

Hardware and software design will be discussed in the following sections of the document.

1.3 Web Application Overview

The RKAT utilizes a website to serve as a centralized hub that connects the four subsystems, the registration kiosk, the terminal, the admin web page, and the online database. It is important to note that the registration kiosk and terminal are physically the same; the difference is which web pages are running, which you can select when the system boots up.

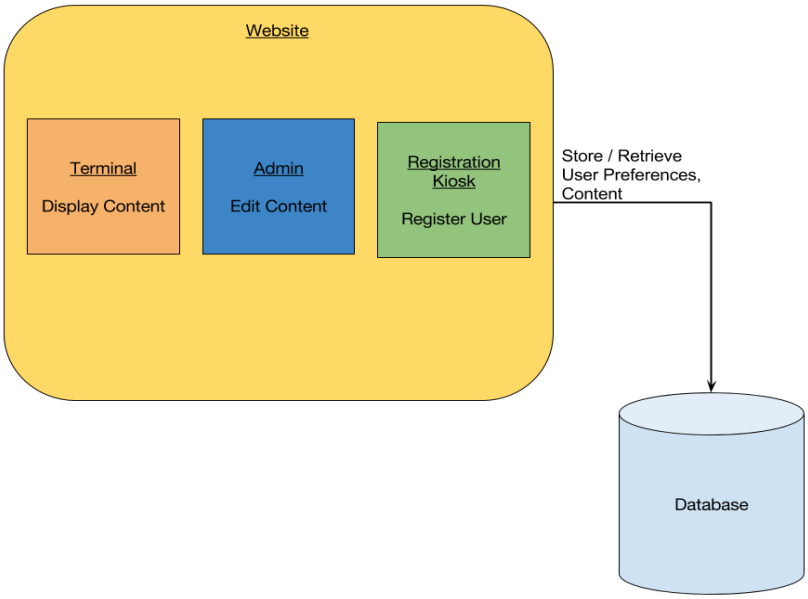

Figure 4. System Overview

1.4 Annotated Images

An annotated image is the type of image that is submitted through the admin update system and will be displayed on the terminals. It is a customized image that may or may not contain text that is created through an image-editing software.

For example, the following is an annotated image:

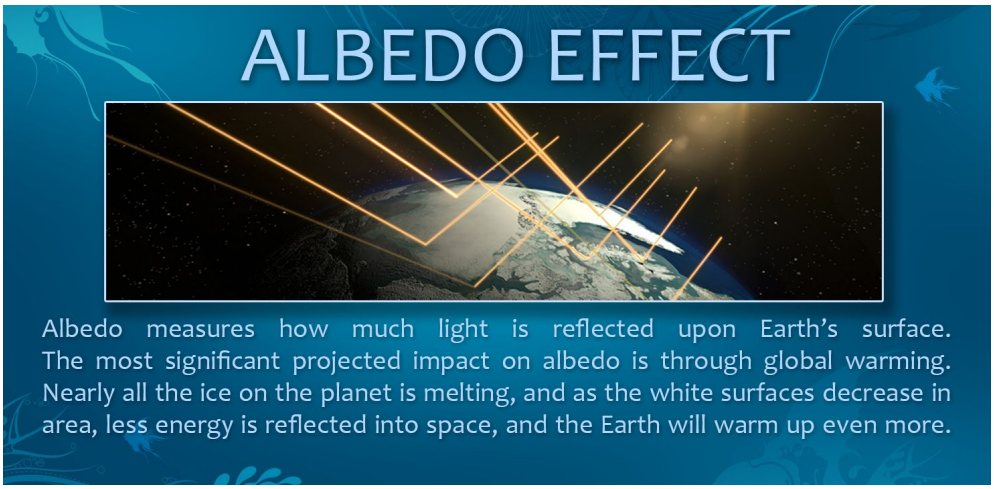

Figure 5. Annotated Image

Some pages display buttons, arrows, and text boxes. These are HTML elements that are not part of the annotated image. An example of a page like this can be seen below.

Figure 6. Annotated image with HTML elements (css arrows, buttons, greeting box)

2. Subsystem Design

2.1 Registration Kiosk

The registration kiosk records user preferences and information that will be used to deliver the user's personalized experience. When the user scans their RFID card the registration kiosk saves the unique tag ID. Next, the user is presented with a registration page where they enter their name, educational level, language, email, and select a Professional Pathway. The information is associated with the unique tag ID and stored into the database. Below is a flowchart of how a registration kiosk functions.

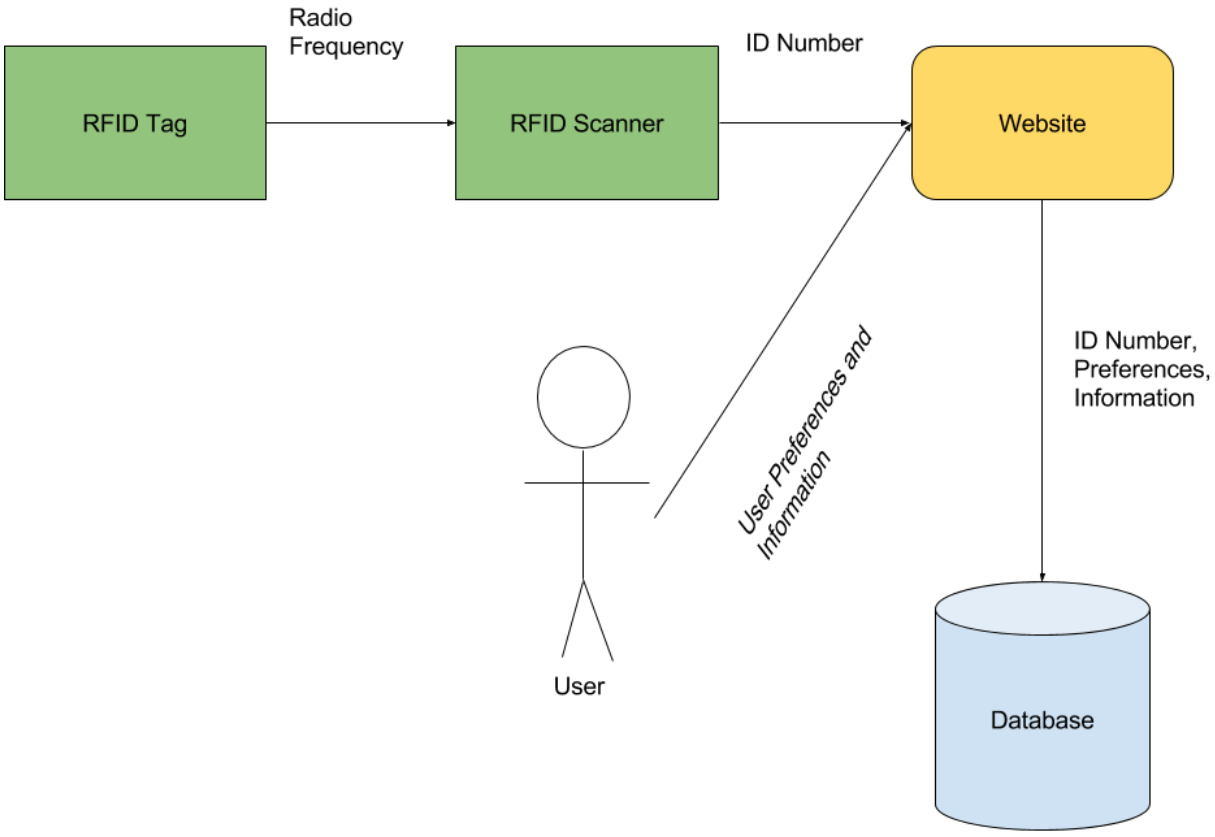

Figure 7. Registration Kiosk Overview

2.2 Terminal

The terminal delivers the personalized experience to the user based off of their registered preferences. A default screen, which will always be displayed on the terminal while not in use, prompts users to scan their RFID card. When the user's card is scanned, the terminal uses their unique tag ID to retrieve their preferences stored in the database, which the terminal then uses to load the correct Professional Pathway web page. Below is a flowchart of how a terminal functions.

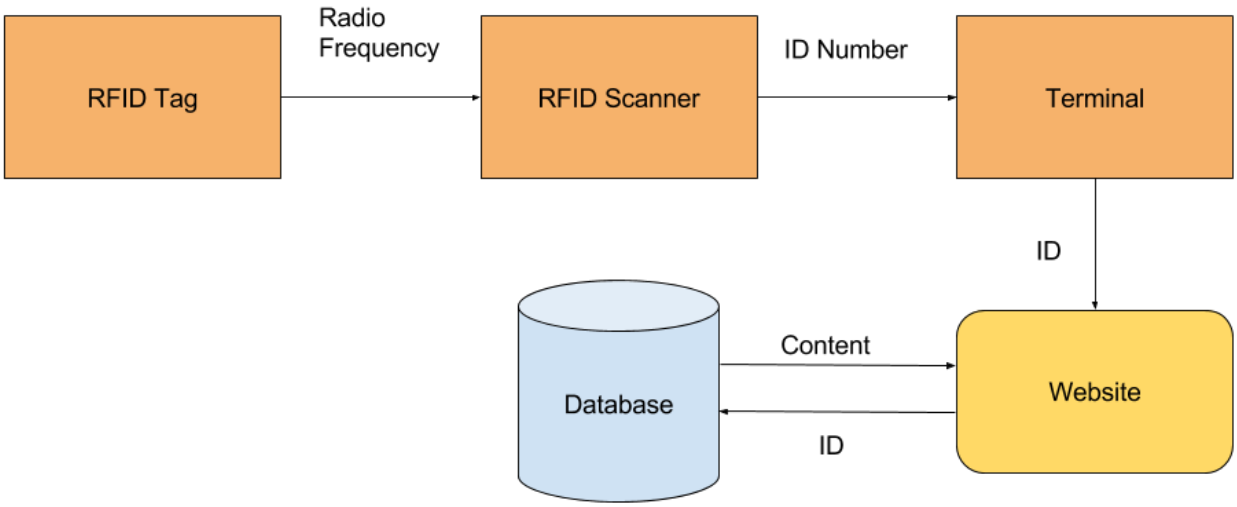

Figure 8. Terminal Overview

2.3 Admin Web Page

The admin web page allows Birch staff to update terminal content and view terminal analytics. The admin web page is hosted on Firebase, enabling staff to access the page from any computer with internet connection. To update the terminal content, staff can upload annotated images (1.4 Annotated Images) and videos using the appropriate naming convention (4.3.1 Naming Convention) via a file selection menu or by dragging and dropping the file(s) into the designated area. These updates are deployed instantly.

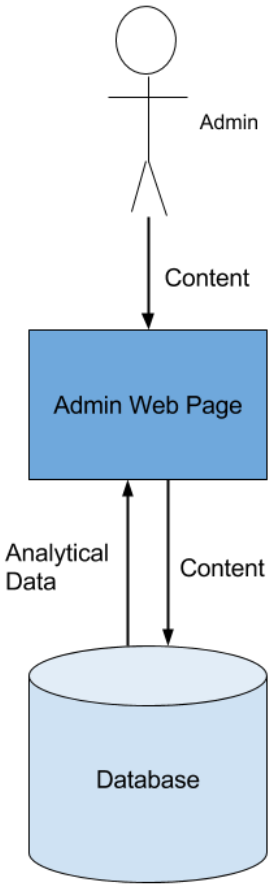

Figure 9. Admin Web Page Overview

3. Hardware Design

This section gives a detailed overview of the main components of our hardware setup. In this setup, we had the freedom of choosing our components for the RKAT, which granted us the flexibility to construct an advanced prototype with custom features that a pre-existing device may not have. The chosen parts will be explained in detail throughout the remainder of this section.

3.1 RFID Reader

The RFID reader used was the ONETAK 125K EM4100 ID Card Reader. When an RFID card is scanned, the reader will write the unique tag ID number followed by a carriage return to the selected input field. This USB reader allows for “plug and play” functionality as there is no custom code or drivers are needed to interface with the device. 13.56MHz RFID cards were used during development with this reader.

3.2 Raspberry Pi

The system uses a Raspberry Pi 3 (RPI) to connect to the internet in order to load our web application in chromium. Chromium will launch in kiosk mode, which runs in full-screen and disables certain features for enhanced security. On startup, an admin will be able to choose if it should act as a terminal or registration kiosk. The RPI will remain acting as that terminal / registration kiosk until it is shutdown or rebooted. We chose the RPI due to it’s low cost, large online support community, ability to act as an embedded system, and its customizability. (Appendix E: RPI Setup Guide)

3.3 Touch Screen

The terminal and registration kiosk units utilize a 21.5-inch Dell S2240T touch screen, LCD monitor with 1920 x 1080 resolution and 10 finger touch functionality to provide users a more interactive experience. It utilizes a capacitive touch screen, which provides excellent image clarity, high resistance to surface contaminants, liquids, and scratches, and a greater sense of familiarity for users who own smartphones and tablets. The monitor uses a standard 100mm x 100mm VESA mounting system, which offers a wide range of mounting options (6.3 Mounting Designs).

3.4 Mobile Monitor Stand

MI-879 TV Stands were used to provide the touch screen monitors a mobile mounting platform with adjustable height (36 inches to 52 inches) and tilt angle (0° to 30° from vertical), so that it has modularity for users of varying heights and ages. Each stand has a compatible VESA mounting plate that supports various VESA mounting hole layouts (75mm x 75mm to 200mm x 200mm) so that it can offer flexibility for installation modifications and monitor replacements. We settled with a height of 48 inches, with monitor attached, and a tilt of 30° from vertical. These were used as temporary mounting solutions given our time constraints.

3.5 Bill of Materials

These are the materials necessary for making one terminal/registration kiosk unit. Prices may vary due to sale offers.

Part	***Price***	***Link***
ONETAK 125K EM4100 ID Card Reader	$8.23	https://tinyurl.com/y8mw9uzo
13.56MHz RFID Cards (100 Pack)	$29.99	https://tinyurl.com/y7oa6aou
Dell S2240T touch screen	$299.99	https://tinyurl.com/y8w9qpjo
Raspberry Pi 3 Kit ● Power Supply ● microSD card (Pre-loaded with NOOBS) ● Pi Case ● HDMI cable	$59.99	https://tinyurl.com/yayqzxae
ARVICKA USB Speakers	$14.99	https://tinyurl.com/ybqkkgnr
Mobile Monitor Stands	$61.15	https://tinyurl.com/yaoml5ja
Total Unit Cost:	**$474.34**	

Monitor Price Range: ≅$41-$394 for monitors with a diagonal length ranging from 7-inches to 24-inches.

4. Software Design

This section gives a detailed explanation of the web pages used by the RKAT. The RKAT software was developed using HTML and CSS for page structure and styling, along with Vanilla JavaScript and jQuery for functionality and database communication. https://github.com/jguidry/SEE_2017_RFID_Pathways.

4.1 Registration Kiosk Software

4.1.1 Registration Kiosk Workflow

The Registration Kiosk has two purposes: to register user's RFID cards and to inform the user of the professional pathway feature and have them select their professional pathway. Below is a detailed technical flowchart of how the web pages on a registration kiosk function and interact with the database.

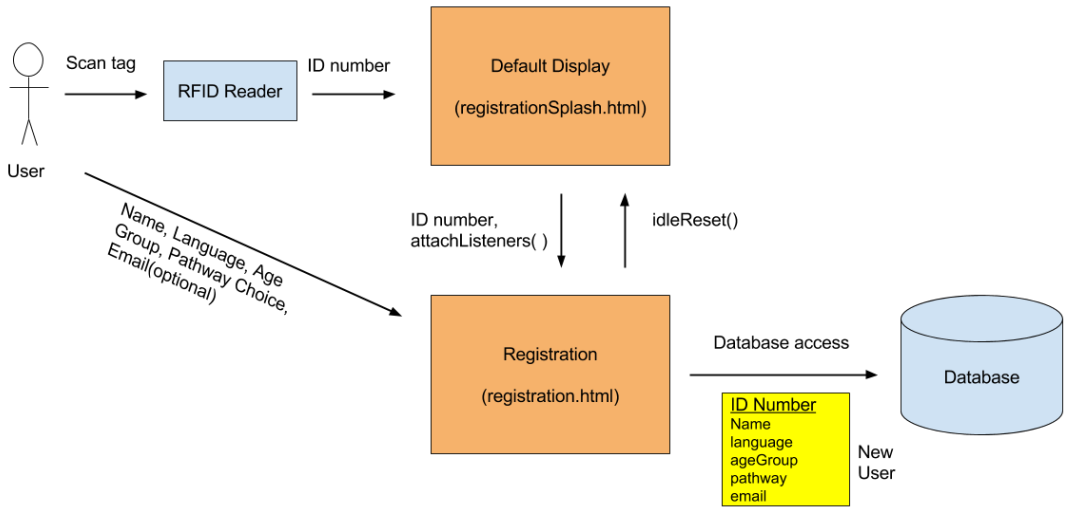

Figure 10. Kiosk Workflow

1. A user scans their RFID card to the reader and their unique user ID is inputted to registrationSplash.html as a datafield, followed by carriage return.

2. The user is taken to a registration.html where they can input their name, language of choice, age group, and their email. If the user doesn't enter a name, they are alerted, and are unable to continue until a name is entered.
3. The user is then directed to choose a professional pathway. Three options are currently provided: Engineer, Biologist, and Climatologist. Each option has a description below so the user can understand what their choice will bring. If the user doesn't select a pathway, they are alerted, and unable to continue until a pathway is selected
4. A final page is rendered, and once the user clicks "Finish", their card is then considered "registered" and the information entered is sent to the database.

Alternate Workflow:

1. If the user does not interact with the terminal, defined by the event listeners attached (attachListeners), for a specified period of time, the user is deemed "idle" and the idleReset function is called, ultimately redirecting the terminal back to the default display, registrationSplash.html.

4.1.2 Registration Kiosk Function Calls

writeToFirebase( ): Creates a new object in Firebase with the user's inputted information. Object's key is the RFID tag ID.

updateData( ): On clicking "Finish", the number of registered users in the database is incremented by 1.

idleReset( ): Redirects the terminal to the default display, registrationSplash.html, if the user is deemed idle after a specified period of time.

attachListeners( ): Attaches event listeners to the current webpage to determine if the user is interacting with the terminal or is idle.

4.2 Terminal Software

4.2.1 Terminal Workflow

The terminal's purpose is to deliver a personalized experience to the user. This is done via redirection to and from various interactive web pages, specifically the default display page and the Professional Pathway pages, and dynamically retrieving content from Firebase Storage.

When a user scans their registered RFID card at a terminal, the terminal will access the database to retrieve the preferences registered by the user in order to redirect the terminal to the correct Professional Pathway web page. Once redirected, the terminal accesses Firebase Storage to retrieve the content associated with the terminal and Professional Pathway. The terminal then displays this content into the correct sections of the web page. Below is a detailed technical flowchart of how the web pages on a terminal function and interact with the database.

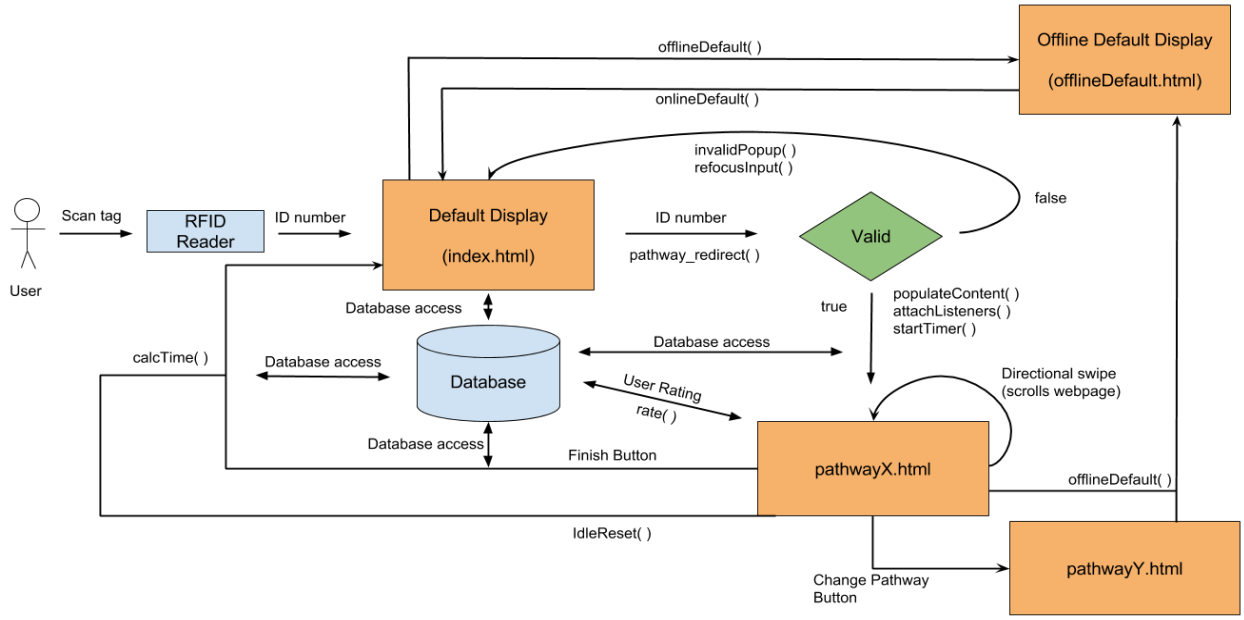

Figure 11. Terminal Workflow

Workflow:

1. The user scans their RFID card at the RFID reader and their unique user ID is written to an input field in the default display web page, index.html, followed by carriage return.
2. The website checks if the ID is registered in the database and therefore valid.
3. If the ID is valid, it retrieves the preferences and professional pathway from the database (pathway_redirect).
4. The website redirects to the correct professional pathway web page (pathwayX.html).
5. The annotated image(s) / video(s) for the Professional Pathway page are retrieved from Firebase Storage and loaded into the webpage. (populateContent)
5. Event listeners are bound to the web page for the idle reset functionality (attachListeners) and the interaction timer is started (startTimer).

6. Once on the Professional Pathway web page, the user can swipe in any direction to view content.

Alternate Workflow:

1. If at any point the terminal loses internet connectivity, the terminal is redirected to offline Default.html (offline Default).
2. Once connectivity is reestablished, the terminal is redirected back to the default display, index.html, (online Default).
3. If the tag ID read from the RFID card was deemed invalid, the user is prompted to register their card at a registration kiosk, and the terminal remains at the default display, index.html (invalidPopup and refocusInput).

Optional Actions:

1. If the user rates the terminal, the result is stored into the database and the terminal's average rating is updated (rate).
2. If the user presses the Change Pathway button and selects a pathway to go to, the terminal is redirected to that webpage.
3. If the user presses the Finished Button, the calcTime function is called to record the user's interaction time and calculate the average interaction time (seconds) at that terminal. The terminal is then redirected back to the default display, index.html.
4. If the user does not interact with the terminal, defined by the event listeners attached, for a specified period of time, the user is deemed "idle" and the idleReset function is called, ultimately redirecting the terminal back to the default display, index.html.

4.2.2 Terminal Function Calls

pathway_redirect( ): Validates that the ID number read from the reader was registered in the database, and if a valid ID, obtains the user's chosen Professional Pathway in order to build the correct link to redirect the web page to. Also records user statistics. Invalid input is delegated to invalid Popup( ).

invalid Popup( ): Displays a popup modal prompting the user to register their RFID card at a registration kiosk if the ID read by the reader was deemed invalid. The modal automatically closes after a specified period of time.

refocusInput( ): Clears the ID input field and refocuses the webpage to it for the next scan.

onlineDefault( ): Redirects the terminal to the default display, index.html, once online connectivity is reestablished.

offlineDefault( ): Redirects the terminal to the offline default display, offlineDefault.html, if the terminal ever loses internet connectivity.

rate( ): Takes the user rating and calculates the terminal's average rating. The result is stored into the database.

calcTime( ): Calculates the time (seconds) that the user was deemed to be interacting with the terminal. The average interaction time (seconds) for the terminal is then calculated and stored into the database.

idleReset( ): Redirects the terminal to the default display, index.html, if the user is deemed idle after a specified period of time.

populateContent( ): Retrieves content for the terminal from Firebase Storage and uses it to populate the correct sections in the Professional Pathway webpage.

startTimer( ): Starts the timer used to calculate user interaction in calcTime( ).

attachListeners( ): Attaches event listeners to the current webpage to determine if the user is interacting with the terminal, or is idle.

4.3 Admin Update System

The Admin webpage is hosted via Firebase at the following URL: https://birchaquarium-fd036.firebaseapp.com/

4.3.1 Naming Convention

The names of the content files (annotated images and videos) must follow the naming convention: *exhibit-pathway-location.extension*. For example, we have an exhibit called albedo, a pathway named engineer, and five different screen locations (top, bottom, middle, left, and right). If we were to change the annotated image of the top section of the albedo terminal for the engineering pathway, we would upload an annotated image with the name: *albedo-engineer-top.png*. For the default display's annotated image,

the naming convention is *exhibit-default-middle.extension*. Only *exhibit* and *extension* will change.

4.3.2 Supported File Extensions

Image Extensions: PNG, JPG, JPEG, SVG, BMP
Video Extensions: MOV, MP4, AMV, WMV, AVI, GIF

4.3.3 Uploading Content

Once an admin is logged in using valid admin credentials (Appendix G: Admin Credentials), the authorized admin can upload images and videos that are named based on our specific naming convention (4.3.1 Naming Convention). The upload system supports drag and drop functionality to upload multiple files for a chosen terminal. It also supports single file upload by using the upload button that is located in the middle of the drag and drop area. The system will notify the user when the files have been uploaded to the database via a progress bar at the bottom.

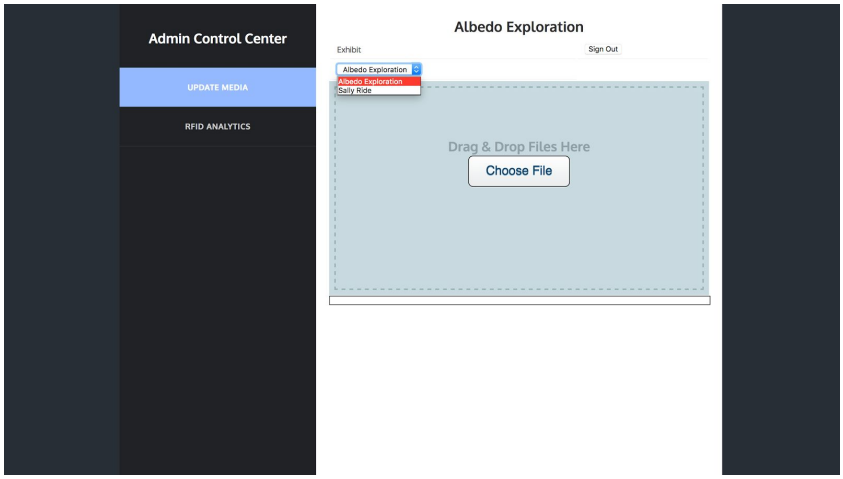

Figure 12. Uploading an Image

Note: Currently, for any terminal, the top section of any pathway of an exhibit will be the only section that will contain a video.

4.3.5 Analytics

The analytics tab of the admin website allows administrators to view real time statistical data about the terminals. The analytics page displays the average time users spend on each terminal and the average user rating for each exhibit. It also displays the number of registered users for each Professional Pathway, and the number of times the pathway has been used by users.

4.4 Database Structure

The system utilizes Firebase as its online, real time database to store user preferences, terminal information and system analytics.

Firebase Storage is used to store the content for terminals. The content in Storage is uploaded via the Admin web page into the desired terminal's folder, named T_#, where the # represents the number of the terminal. The terminals will use this stored content to populate the Professional Pathway web pages.

The database contains three main children, RFID, Terminals, and User_Data, under which all the information is kept. Each child will be described in the sections below.

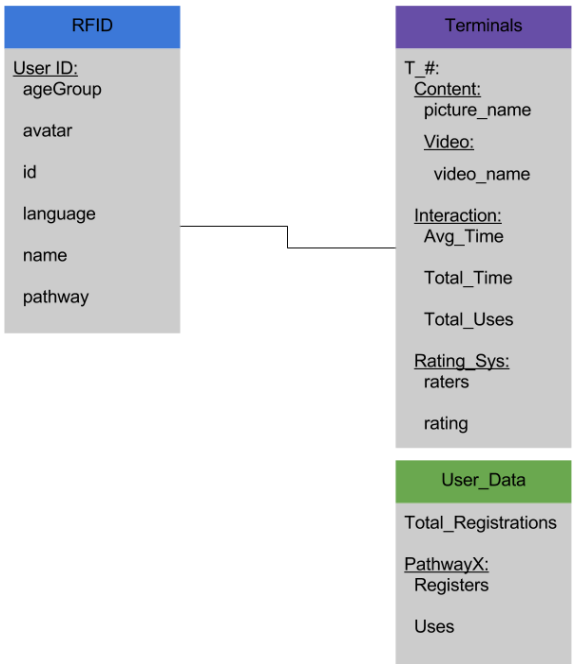

Figure 13. Database Schema

4.4.1 RFID

Stores information and preferences of registered users under the RFID tag's unique number ID.

4.4.2 Terminals

Contains information about each terminal. The information associated with a particular terminal is stored under a child named T_#, where # is the number of that terminal.

The content filenames and extensions are stored as a key value pair under the Content child. If the content is a video it is stored under the Video child, while pictures are stored normally under Content.

The average time (seconds) spent at a particular terminal by users is stored under the Interaction child and is calculated by: Total_Time / Total_Uses. Total_Time is measured in seconds, and represents the total amount of time spent interacting with the terminal for all users. Total_Uses represents how many users have scanned their RFID tag at that terminal.

The Rating_Sys child contains the number of users that have rated the terminal as well as the terminal's average rating. The rating field represents a user's rating of their experience with their personalized content at that particular terminal.

4.4.3 User_Data

Contains system analytics for Birch Aquarium staff regarding each Pathway offered. PathwayX represents a specific Professional Pathway and contains statistics about how many visitors have registered that pathway as well as how many times that pathway was 'used' by scanning at any terminal.

5. UI / UX Design

The user interface and user experience of this system has evolved by undergoing a process of heuristic research, prototyping, user testing, and design iteration.

5.1 Design Considerations, Heuristics and Research

The design of the system's interface and experience is critical in improving the educational experience for the visitors and keeping them engaged. It is also important in allowing Birch Aquarium staff to easily update the content of the terminals without needing advanced technical knowledge of the back-end programming.

Our design aims to be intuitive, engaging, and flexible. By being intuitive, visitors will know how to operate the terminals with little to no instruction.

The main design heuristics we took into consideration for our system are user control, flexibility, and efficiency of use. These heuristics are the most important because the system that we are providing is meant to be intuitive and engaging for the user, and if, for example, a screen appeared too convoluted or inflexible then the user would not want to use the system. The functionality of the system is the most important aspect of this design because our main task was to prototype ideas of personalized user experiences by implementing RFID technology, so user control and freedom was what was mostly taken into consideration. The visual aesthetics mainly complement the functionality but were not the main priority since Birch Aquarium staff will eventually determine what the visual content will actually show and say.

By researching general user interface heuristics, we found it important to have non-linear navigation for the terminals so that the experience was less fixed and gave more priority to the user's personalized choices. In regards to the visual content being displayed (buttons, captions, etc.), we aimed to avoid excessive and irrelevant information and to have non-competing dialogues so that the user wouldn't get overwhelmed.

5.2 User Testing and Feedback

Through user testing, we were able to expand upon the features we initially implemented based on research. The initial prototype of the pathway landing page included an annotated image, a change pathway button, and a finished button to demonstrate correct RFID redirection.

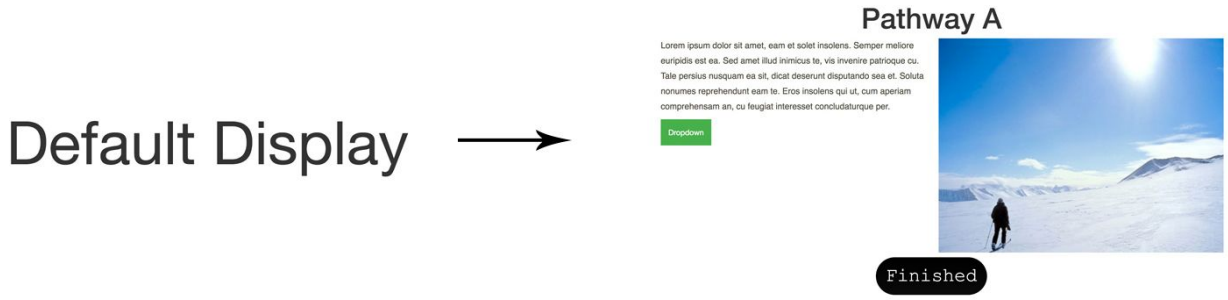

Figure 14. Terminal 1.0 – Basic scan redirection

Terminal

The terminal and its UI were our top priorities going in. We initially brainstormed ideas and tested users to see how our initial design could be improved upon to make it more interactive and personalized. We went around the UCSD campus and approached about 50 people from diverse backgrounds including students from a variety of different majors, professors and staff. Although, our user testing lacked feedback from younger populations such as elementary school students.

Through user testing, we mainly emphasized feedback on the functionality and intuitiveness of the design and not the educational content being displayed. We tested our users by first providing a brief description of what the experience will be used for and asking them to imagine themselves as visitors at the aquarium. They were then given an RFID card and presented with the kiosk and terminal to see if they could understand what to do without our help. We also tested different versions of the same subsystem to see which design decisions people preferred. We asked questions such as “Do you understand what the purpose of registration and professional pathways is?” and “How did you want your information to be presented to you? Reading then scanning/swiping or scanning/swiping then reading?” Our findings are discussed next.

Most criticisms of the UI were that it should be more flexible, and we remedied this by implementing full page four-directional swiping to give users more freedom in what content they’d like to see. To make swiping intuitive, swipe functionality must be

specified for the user by placing arrows on all four edges of the screen, and if that wasn't enough to indicate that the user can swipe, we also included a temporary help box that explicitly tells the user to swipe. To further enhance user engagement, we reduced the amount of information displayed on the default and landing pages because users wanted to swipe and interact first before reading information.

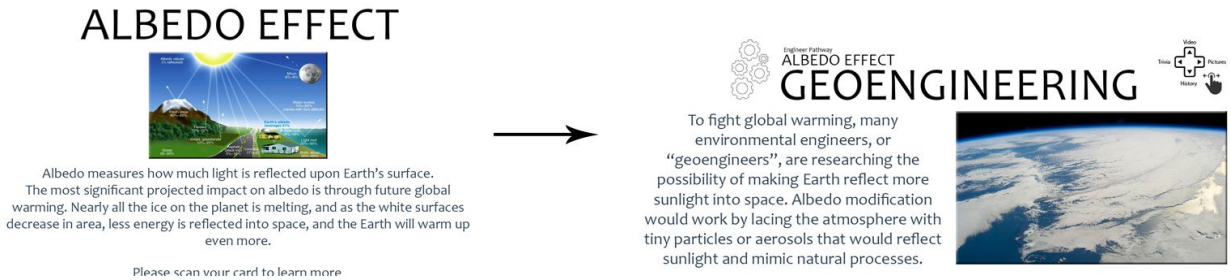

Figure 15. Terminal 2.0 – Annotated images

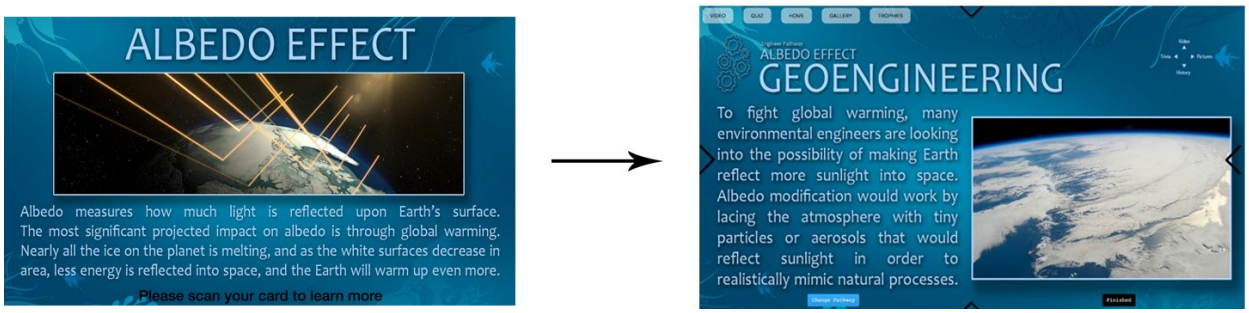

Figure 16. Terminal 3.0 – Aesthetic upgrade + directional swipe navigation

Figure 17. Terminal 4.0 – Optimized styling and content interactivity

Overall, the UI of the terminal eventually went through enough iterations that allowed for it to be intuitive and self-explanatory as well as engaging enough for the user. It's main features include: directional swipe navigation, nonlinear learning, arrow captions, and the ability to change pathways.

Kiosk

User testing the overall UX of the kiosk and the terminal revealed a critical flaw in our approach. Initially, we ignored the design of the kiosk since a previous team had already implemented it, but after realizing how confused users were during the registration process we began to rethink our approach. Users said they did not understand the purpose of registering and selecting a pathway, so we remodeled the kiosk to include an entire page dedicated to explaining the purpose of the professional pathways which also encourages users to want to scan their cards. Aesthetic consistencies between the kiosk and the terminal also suggest that the two subsystems are connected, so to further enhance the cohesiveness of the UX we implemented directional navigation for the kiosk as well (kiosk only has horizontal navigation) and used a similar color scheme of gold, blue and black for the buttons.

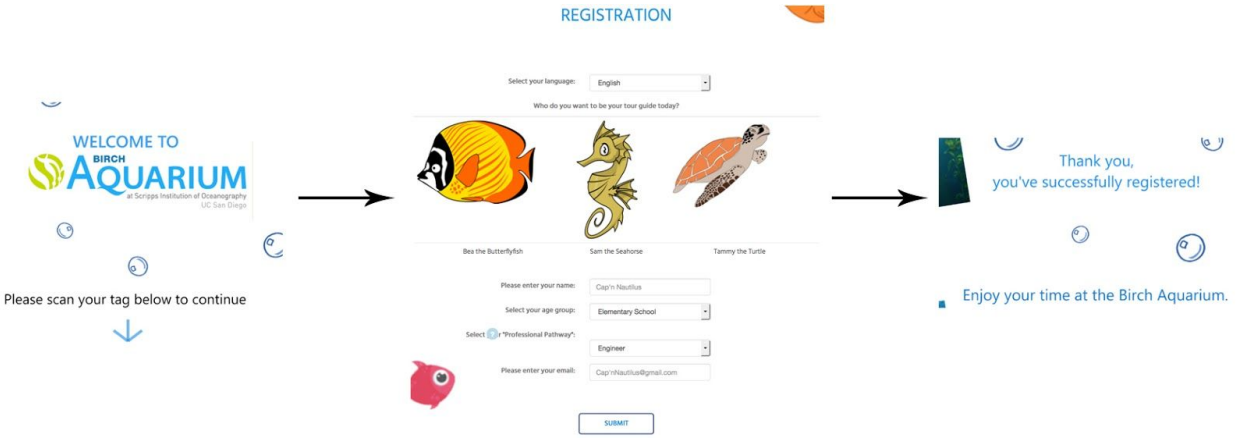

Figure 18. Kiosk 1.0 – Single page registration with basic information fields

Figure 19. Kiosk 2.0 – Multi-page registration with information fields spread out + instructions + information + aesthetic upgrade

5.3 Admin Web Page

The main priority for designing the admin portal was making it intuitive so that it did not require a staff member with advanced technical knowledge to update the terminals. We researched ways of uploading content that most people are familiar with, and we prototyped a solution that included a drag-and-drop or click-to-select box that the admin can use to upload the images and videos to the terminal. The solution we implemented works as follows: the admin selects a file to upload, and the system parses the filename to upload it to the correct spot in a terminal.

5.4 Conclusions and Potential Improvements

For the admin web page, it would be ideal to have a visual system rather than a filename/text-based system for file uploading. Instead of having the system parse filenames, a visual template with a real-time display of what the admin is uploading could improve the simplicity of the uploading process.

For the kiosk and terminal, after designing the UI / UX of our system we learned that it is sometimes necessary to provide very explicit instructions and information so that the user understands what to do, as long as it isn't overwhelming. The interface could be improved by providing more interactive components in each direction rather than an annotated image to be read; however, the content to be displayed is determined by Birch Aquarium staff and a team of content designers. Also, a future team should take advantage of having touch-screen capabilities and not limit themselves to simply pressing and swiping the screen (possibly something even physical such as incorporating a camera).

6. Conclusions

We were successful in creating a kiosk-terminal system that provides users a personalized experience through the use of an interactive interface that utilizes touch screen functionality and RFID technology. The RKAT is able to obtain content from an online database and distribute that content onto the terminals, as well as allow administrators the ability to update terminal content. The kiosks are able to read ID numbers from RFID cards and store them in an online database during the user registration process. RKAT terminals are capable of reading ID numbers from RFID cards and associate them with user preferences chosen from the registration process, then display content with their preferences. Interactive components are successfully satisfied with touch screen functionality.

We had a couple of minor issues while creating the system. Firstly, we had trouble implementing a successful, offline solution for the project due to the ability to create a robust system given the time constraints. Secondly, using firebase, we had difficulties decreasing the loading time of the content to a reasonable amount of time. Currently, it takes around 3 seconds to load the pathways on first load, and a little less the 1 second on subsequent scans (due to chromium's caching feature).

6.1 Future work

A lot of our decisions were motivated by our time constraint as well as building off of previous teams work.

- Currently, Birch Admins cannot add new terminals through the Admin web page. We heavily suggest that the next team develops a way to add new terminals to the system through the Admin web page.
- We suggested that RKAT is switched to a local area network (LAN) because we believe it will add additional security and robustness to the system.
- Our database is currently run on google's firebase. An experienced team could implement an SQl or personal database, which would improve the speed and the ability to search the database.
- Consider the fabrication of an encasement that can hold and secure most of the components needed to construct a physical terminal-kiosk unit.
- Implement other physical components, such as a camera, to initiate more interactivity amongst users

- Possibly use Local Forage: https://github.com/localForage/localForage instead of Local Storage to allow for offline usage as well as increased speed and security.

6.2 Creative/Interactive Possibilities

- Artifact Collection
    - Birch Aquarium visitors would be able to collect virtual artifacts from every RFID enabled exhibit they can interact with and play a mini scavenger hunt. Collecting artifacts in this way would promote social interaction among other visitors and further engage each visitor to interact with every exhibit.

- Quiz With Reward System
    - Birch Aquarium visitors could take quizzes on every RFID enabled terminal about the exhibit they are currently interacting with and obtain points based on their highest score. If an individual visitor accumulates enough points, they could use those points to trade for trinkets from the gift shop. This option would promote learning for each individual visitors by rewarding them for seeking out information on their own and then testing themselves about the information they had sought.

- Avatar Interaction
    - Birch Aquarium visitors would register at kiosk and be able to pick and choose from a selection of avatars. The avatars would then greet them by name and provide descriptions for them whenever they would interact with RFID enabled exhibits with their RFID tag. Avatar interaction would provide a fun, personalized, and ostentatious way of providing content for each Birch Aquarium visitor.

6.3 Mounting Designs

For terminal/registration kiosk mounting designs, there were two options: a mobile platform or a wall mounted installation. The best solutions was the mobile platform due to its portability and ease of installation/reinstallation into the existing infrastructure. Ideally, these designs would have an encasing or housing for the hardware components. These CAD designs were created using Autodesk Inventor 2018.

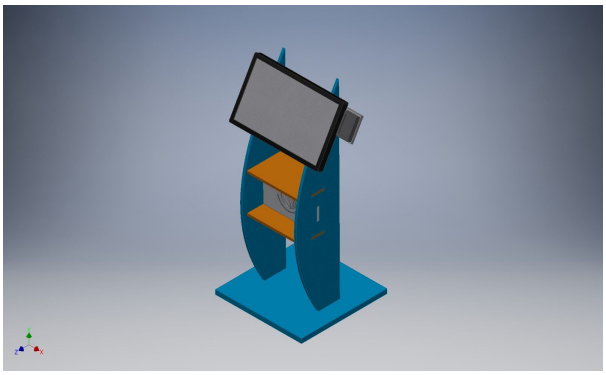

Figure 20. Podium Design

Pros:

- Aesthetically pleasing
- Does not require a lot of material

Cons:

- May be difficult to assemble
- Might not be structurally sound

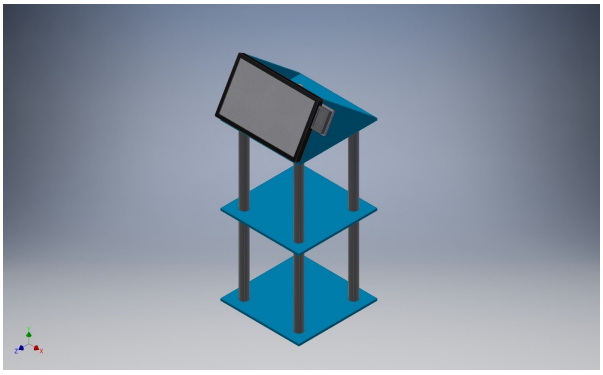

Figure 21. Bookshelf Design

Pros:

- Easy to assemble
- Uses very little material

Cons:

- Too much free space
- Prone to buckling (depending on geometry and material of poles)

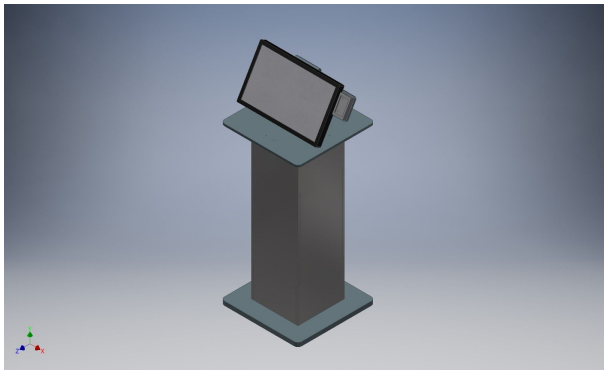

Figure 22. Pillar Design

Pros:

- Structurally sound
- Simple design

Cons:

- Requires a lot of material
- Not aesthetically pleasing

7. Appendix

Appendix A: RKAT GitHub Repository

The RKAT source code can all be found in one repository on GitHub: https://github.com/jguidry/SEE_2017_RFID_Pathways

Appendix B: Firebase Cloud Storage, Firebase Database, and Admin Hosting

Firebase is heavily relied on to store user data and content that would be used for populating RFID enabled terminals. The Firebase cloud Storage is used to store and retrieve media content such as images and videos. The Firebase database is used to store image and video extensions, user data, and user analytics. Firebase is also used to deploy Admin. This link is for the Firebase console: https://console.firebase.google.com/u/1/project/birchaquarium-fd036/overview .

Appendix C: Deploying Admin Website Through Firebase

To Deploy the Admin application, simply run

```
firebase deploy
```

If you want to deploy the entire application, you will need to edit the firebase.json as well as changed the file system to include all necessary files in a public folder.

To run the web application on a Raspberry Pi, first make sure you are in the "~" directory then run

```
./SEE_2017_RFID_Pathways/runApp.sh
```

This will launch the app in the browser (Note: this should automatically launch on the pi when it has internet)

Appendix D: External Libraries Used

fullPage js - A javascript library that helps performance UI, it was used to allow swiping to different directions.

Bootstrap - Bootstrap is an open source toolkit for developing with HTML, CSS, and JS. Bootstrap was used to simplify the styling of all the web pages used.

Keyboard js - Used to make a virtual keyboard pop up when a user registers using the RFID kiosk.

Quiz - Quiz application that was used to create the quiz for the albedo exhibit.

Drag and Drop - Used to create the drag and drop functionality to submit media files for the admin update system.

Appendix E: Raspberry Pi Setup Guide

This guide will help setup the web application on the pi. It assumes that the user has basic linux/bash knowledge as well as basic knowledge about the raspberry pi. It would also be useful if the user had a keyboard and mouse, otherwise the user would need to ssh from a different machine.

1. **NOOBS**
    a. Ensure that NOOBS with Raspbian running Debian Stretch is installed on the microSD card, if not, download it here: https://www.raspberrypi.org/downloads/noobs/
2. **Check Display Settings (If the pi isn't filling up the entire display)**
    a. Go to the Raspberry Pi Menu (top left corner) -> Preferences -> Raspberry Pi Configuration -> SetResolution
3. **Update and Upgrade the Pi software**
    a. Ensure you're connected to wifi
    b. Open the terminal (under the menu -> Accessories) then run:
    c. Run `sudo apt-get update` then
    d. Run `sudo apt-get upgrade`
4. **Disable Raspberry Pi ScreenSaver**
    a. Run `sudo apt-get install xscreensaver`
    b. Once that's installed, open the menu -> Preferences -> ScreenSaver -> Mode-> "Disable Screen-Saver"
5. **Clone Files**
    a. Open the terminal then run:
    b. `git clone https://github.com/jguidry/SEE_2017_RFID_Pathways.git`
6. **Make the app run on startup**
    a. Open the terminal then run:
    b. `sudo nano ~/.config/lxsession/LXDE-pi/autostart`
    c. Add this line to the bottom of the file:
    d. @bash /home/pi/SEE_2017_RFID_Pathways/runApp.sh
    e. Test if this works by running `sudo reboot`

7. **Disable the cursor**
    a. Open the terminal then run:
    b. sudo nano /etc/lightdm/lightdm.conf
    c. Find the line that says
    d. #xserver_command=X and uncomment it and add
    e. xserver_command=X -nocursor
8. **If you need to kill the process when in kiosk mode and restart**
    a. Open the terminal then run:
    b. killall -9 chromium-browser
    c. bash ./SEE_2017_RFID_Pathways/runApp.sh
9. **To update the code from the repository (hopefully not needed)**
    a. Open the terminal then run:
    b. cd SEE_2017_RFID_Pathways
    c. git pull origin master

Appendix F: Contact Information

Name	*Email*	*Phone Number*
Bryle Castro	brylecastro2013@gmail.com	(619)-480-6097
Josh Duhay	jeduhayl@ucsd.edu	(610)-730-5503
James Guidry	guidrywyatt@gmail.com	(916)-284-4911
Samm Iwamasa	siwamasa@ucsd.edu	(310)-713-4915
Matthew Rice	mhrice@ucsd.edu	(310)-245-6652

Appendix G: Admin Credentials

Username	*Password*
birchadmin@birch.com	fisharefriends

If username/password changes are needed, contact a team member above.