Developing Applications to Compare Methods of Teaching Emotions

Elizabeth "Lizzie" Siegle

An Undergraduate thesis submitted in partial
fulfillment for the
degree of Bachelor of Arts

in the

Computer Science Department of
Bryn Mawr College

Advisor: Professor John Dougherty

May 2018

Abstract

e will explore which form of graphic (static image, gif, or video) is most effective in teaching emotions in web and mobile applications. We will compare users' answers for different sections in the web applications and iOS application we develop. The questions are meant for people who have Autism Spectrum Conditions (ASC), but have been tested on current college classmates.!We will explore which form of graphic (static image, gif, or video) is most effective in teaching emotions in web and mobile applications. We will compare users' answers for different sections in the web applications and iOS application we develop. The questions are meant for people who have Autism Spectrum Conditions (ASC), but have been tested on current college classmates.

1 Background

1.05 Dedications

Thank you to my advisor, Professor John Dougherty; Haverford Digital Scholarship Librarian and Visiting Professor Andy Janco; my parents, brother, and grandmother; the Twilio Developer Network team; and mentors Tomomi and Bear.

1.1 Overview

Social cognition is a complex process where individuals acquire, understand, and use social knowledge to quickly and accurately respond to verbal and nonverbal social information. Studies have shown that social cognition is extremely important in human relationships [14]. Understanding both themselves as well as the people around them can lead to success in other aspects of one's life

1.11 Problem

Autism is used to describe "a wide range of symptoms that span across an individual's sensory, cognitive, motor, language, and social-emotional development." [2] These skills are all important in the development of every human being, regardless of whether or not they are diagnosed with ASC, or Autism Spectrum Conditions. ASC can be distinguished by deficiencies in social interaction such as communication, as well as stereotyped repetitive behaviors or habits.[7] Those could include rocking and hand-flapping.

Lacking social cognition and having a deficit of social knowledge is one common trait in individuals with Asperger syndrome/High Functioning Autism (AS/HFA), which may be a reason why people diagnosed with AS/HFA are oftentimes behind their more social-thinking peers in terms of functioning in social situations.[14] Being behind in social situations often leads to being behind in other situations and ways.

Social skills are a behavioral manifestation of social cognition. Having a social cognitive deficit means one has social difficulties in the initiation of communication, listening to and processing of subtle sensitive cues, abstract and inferential thinking, understanding the perceptions of others, gestalt processing, and/or humor. [20] Individuals with AS/HFA, or anyone with a social cognitive deficit, are thus limited in the jobs, environments, activities, and opportunities available to them because many social situations with other people would make them uncomfortable: they would not understand some of what was happening around them, and others would not understand what they were doing or why they were reacting in the way they were. This is why it is important to teach everyone social cognition.

1.12 Non-Verbal Communication

Linguists believe that each distinct language has two sublanguages. Expressive language encodes messages by translating them into words or other symbols. Receptive language decodes messages so that their meanings are accurately understood with their intended meanings. [13] This means that speaking and writing

words is expressive and understanding those words is receptive. Communicating how one feels internally through emotions is a form of expressive nonverbal language. Albert Mehrabian studied face-to-face interactions of a wide range of people and found that fifty-five percent of a message's emotional meaning is conveyed through "...facial, postural, and gestural means", and the other thirty-eight percent of a message's emotional meaning is communicated by the mood of a voice. Words contribute to a mere seven percent of a message's emotional meaning. [13] These studies show the importance and value of nonverbal communication and social cognition in understanding how someone is feeling and what they are thinking: correctly identifying and reacting to facial, postural, and gestural means can make one more welcomed, respected, and wanted by those around them. Conversely, if one incorrectly sees and reacts to facial, postural, and gestural means, they could become social outcasts because "serious misunderstandings can occur if [someone fails] to interpret nonverbal messages correctly, or if [they] send nonverbal messages that do not accurately reflect [their] emotions." [13] Furthermore, recent surveys show that the average person spends less than forty minutes per day communicating verbally with others. [13] (This does not mean they are not communicating at all: they are merely communicating non-verbally as well.) These studies and statistics show the value of learning emotions.

1.13 Social Thinking

Related to social cognition is the concept of "Social Thinking" which emphasizes teaching and studying the reasoning behind socialization "...without implicitly targeting discrete social skills." [14] In other words, "Social thinking is a way to train your brain to help you figure out the people around you." [21] Our project will focus on both social cognition as well as Social Thinking to teach emotions to users of our applications.

1.14 Teaching Social Skills

Much research has been conducted to show the benefits of using visual aides with Autistic people as well as those without disabilities. This is because visual images "(a) can make abstract verbal concepts more concrete, (b) remain stable over time, while auditory information can be missed as students' attention fluctuates, and (c) provide a more powerful means to engage attention. [2] Those who work with people with Autism, and teachers in general, have taken note, often deciding to use visual aides in lesson plans.

2 Related Work

Fortunately, however, other studies have shown that social skills can be taught. [14] Already, people have been trying to teach social skills in schools. Figure 2.0 is an image of a poster in a classroom from Stanbridge Academy, a kindergarten-high school for students with mild to moderate learning differences and social communication disorders in San Mateo, California. [16]

Figure 2.0: Poster from Stanbridge Academy

Figure 2.0 shows an image of a smiling person and students must select the emotion from a series of options that best fits that of the smiling person. This poster is one way to teach social knowledge, and is one way of addressing how "...children with ASD need a structured educational approach with explicit teaching." [11] As some of the students who attend Stanbridge have ASD, this poster is one example of a structured, clear, and simple educational approach with explicit teaching.

Similarly, much books like *The Social Skills Picture Book* and *The Social*

Skills Picture Book for High School and Beyond by Dr. Jed Baker help teach people what is an appropriate thing to say or an inappropriate thing to say in different situations.

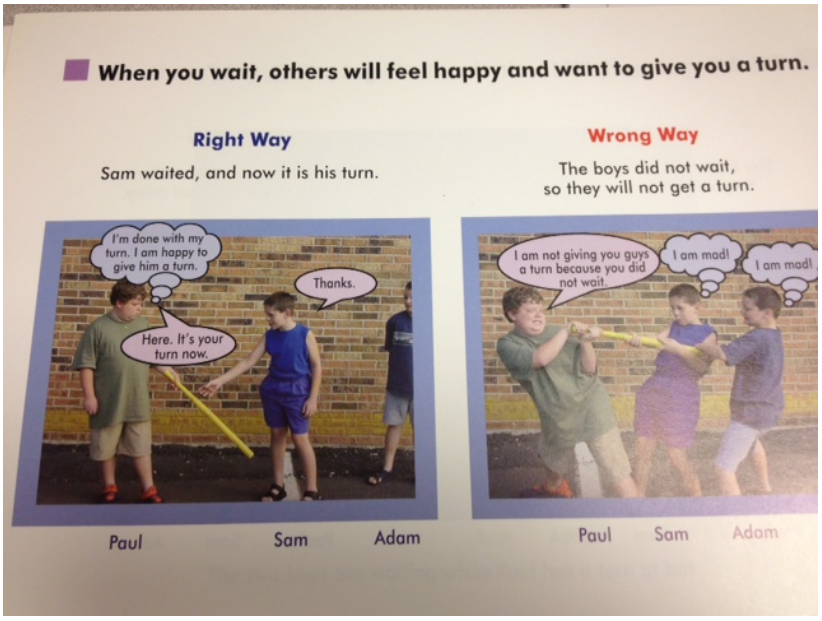


Figure 2.05: Page from *The Social Skills Picture Book for High School and Beyond*

Those books, a page from one which is shown in Figure 2.05, provide example situations where readers learn when to make supportive statements, when to be funny or not, when to introduce oneself, how to start a conversation with someone they know as well as someone they do not know, where to sit or stand in relation to other people, how to handle difficult situations, how to interrupt a conversation politely, how to make new friends or work in a group, and more. [3] These are situations some people take for granted, but many people, regardless of whether or not they have been diagnosed with Autism, could use assistance with.

2.1 Related Technical Solutions

Much work has been done in the field of iOS development to address the teaching of social skills. Some similar applications include *BodyLanguage*, which is meant to guide users on how to gesticulate, greet people, and remain calm while

considering what their body language conveys, [5] or *Micro-Expression Trainer* (shown in Figure 2.1) which shows users what an emotion looks like and explains facial features that match said expression. It focuses on anger, contempt, disgust, fear, happiness, sadness, and surprise, which it believes to be the seven universal emotions.*Micro-Expression Trainer* costs $3.99 and lacks sound or audio.

Figure 2.1 Screen from Micro-Expression Trainer

Another iOS application that teaches emotions is *AutismXpress*, shown in Figure 2.2. It has twelve faces with basic feelings, and the user chooses one to see a fun animation and hear some sound effect that goes along with the feeling.

Figure 2.2 Screen from AutismXpress

TouchLearn is one iOS and iPad app that displays four faces with different emotions, and the user must select the face that matches the one given emotion.[18] Questions and levels can be added to further and hone the learning and teaching of emotions and social skills to different types of people. One iOS and iPad application available to the public is Avokiddo Emotions which is more interactive, geared towards younger children. It involves dressing up animals and seeing their reactions to different actions like being poked, hearing an alarm, dancing, and more. [1] Another iOS and iPad application available to the public is Emotionary+ which guides users past five core emotions, helping them learn more specific ones based on one of those five primary ones. [9]

Speech Language Pathologist Lois Jean Brady advocates for “iTherapy”, or the usage of Apple products like the iPhone, iPad, or iPod Touch and iOS applications to help students, both ones with and without Autism, to achieve their personal educational goals. [5] She points out that Apple products support applications involving voice output, text-to-speech, sign language, sentence generation,

and other forms of communication to help gamify and reinforce repetition and usage, making learning fun for students and people of all ages and backgrounds.

In short, much prior work has been done to teach emotions to people on the Autism spectrum. Applications have been made based on prior social thinking research to specifically teach emotions, idioms, sign language, vocational skills, language comprehension, and more. [5] With the rise of technology, work in that sector has been done primarily through the development of iOS applications. However, not much work has been done to do this on the web platform. That is one way that our project differs from work that is already out there. Additionally, Many other applications include some sort of graphic like a static image or video, but none we have seen include static images, gifs, and video, so that is one other way ours stands out.

3 Our Solution

Our project differs from other applications currently available to the public in a few ways. Firstly, our apps use different forms of graphic (static image, gif, or video with sound.) There's also different levels of questions to test different aspects of emotion recognition. The levels build on each other but still let us compare which graphic type the user performs better on. It also is primarily on the web platform, but has a small iOS application. Our applications also differ from others already out on the App Store in that they use different technologies like APIs, which we will go into more detail about later on.

Our hope for our applications is that they tackle this problem of individuals lacking social knowledge by digitizing and gamifying the teaching of social skills through application development, which we will go into greater detail when discussing the technical methodology used. This should address the problem because people of all backgrounds, but particularly ones diagnosed with ASD, often learn better through a structured educational approach.

3.1 Our Applications

Our project is the development of a series of web applications to make the teaching of emotions more accessible to everyone–regardless of whether or not they have Autism or ASD, or have a social cognition deficit. Thus, anyone and everyone can learn the skills needed to be comfortable in any and every social environment.

3.2 Web and Mobile Platforms

Most of our applications are made for the web platform because that is the most accessible: most people are able to somehow get to a public library that provides free access to computers. Whereas the mobile platform, though it allows users to use applications wherever they are with Wi-Fi or data, is limited to those users who own smart phones or tablets. Since social skills and cognition can be taught, applications teaching social skills should and could make it easy and accessible for anyone, regardless of background or prior social knowledge, to learn and develop social skills. The purpose of our series of web applications is to compare which application is better at teaching emotions. So far, there are three web applications with the same questions and possible answer choices, but the applications differ by the type of graphic displayed for each question. One has static images of faces displaying different emotions, one has short gifs that lack sound of faces displaying different emotions, and one has a short video with sound showing faces that display different emotions. This is the only difference between each web application: the questions, question order, possible answers, and possible answer orders remain the same to best compare which media type the user scores better on. Their score is based on the number of questions they get right.

3.3 Distinctions within our Applications

That distinction in media type is important because most applications available to the public that teach emotions or similar social skills only use static images. We

hope to test whether or not a user gets more questions right on a certain media type. Each application also has multiple levels or styles to teach different social skills: one asks the user to identify the emotion shown, one asks the user what the person could be thinking, and one asks the user what they themselves could say to the person shown. Each question aims to teach a particular emotion using multiple choice questions.

3.4 Emotions

The emotions we chose to teach and focus on are the basic emotions advocated by Ekman, Friesen, and Ellsworth. They include anger, disgust, fear, joy, sadness, and surprise. These emotions differ from the basic emotions advocated by other researchers and psychologists in multiple ways. The main one is that Ekman has done extensive work with emotions and people with Autism and was mentioned in an article called *The Development of Emotion Recognition in Individuals with Autism.*[6]

There is no authoritative list of basic emotions. Different people believe different emotion theorists.

3.4.5 Emotions in our Applications

We decided to focus on anger, disgust, fear, sadness, joy, and surprise because we believed they were most important to learn and distinguish, and also that there is slight overlap and similarities between many of the eight primary emotions. Some complaints about Plutchik's model are that it is too simplified, not reflecting larger emotional nuances. Regardless, it is generally believed that his Wheel of Emotion is a good starting point when deciding what emotions to target when designing an application or other product.

Given extra time, we, or other developers, could add on to these applications to further the teaching of emotions and social skills in multiple ways. Other possible levels could be added on to include questions about recognizing and identifying language like sarcasm, idioms, metaphors, humor, or irony. Similarly,

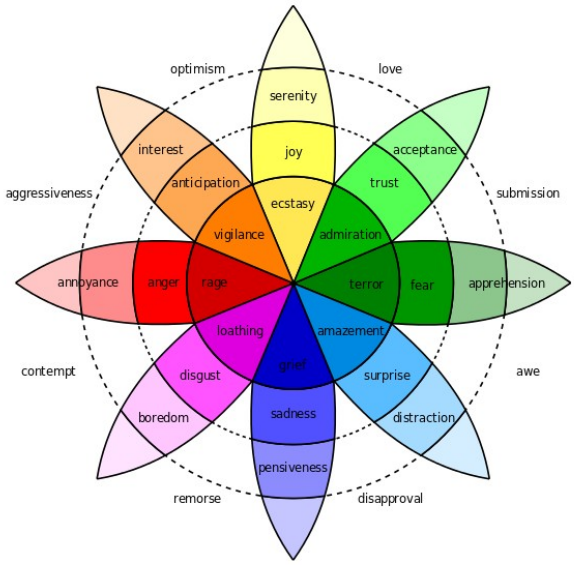


Figure 3.4 Plutchik's Wheel of Emotions with the eight primary emotions and the emotions the primary ones combine together to form

another method might be to develop some mobile application to mobilize the game that syncs with the web version, thus letting users learn on-the-go. We are not adding these features yet, but they reflect the ways for the project to be continued, improved, and added on to.

To test the results, we will look human subjects to volunteer to use the applications after submitting an IRB form. These subjects will sign a waiver and can be anyone. It would be preferred to test on subjects who have been diagnosed as being on the Autism spectrum, but that requires more paperwork and more time searching for subjects. These subjects will go through each question of the applications and we will see which media questions the user performs better on. We predict that most users will perform better on questions with a video graphic as opposed to one with a static image or a simple, no-sound gif, but are unsure that this prediction is true: that is why we will compare each application that has a different, distinct media type. The UI design of these applications is very simple, as demonstrated by images in figures 3.41-3.44.

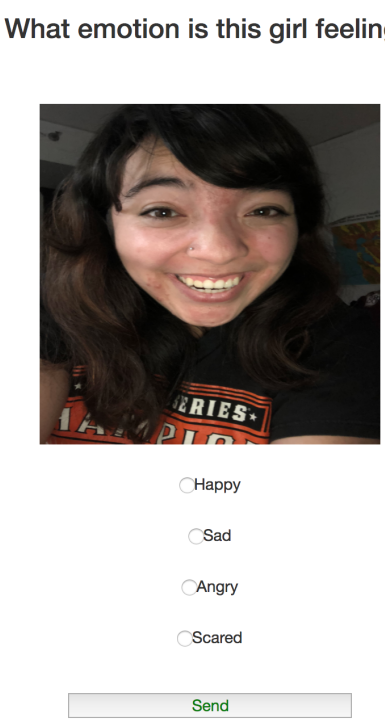


Figure 3.41 guess emotion from image" of web application

Figure 3.42 "what might they say based on their emotion" of web application

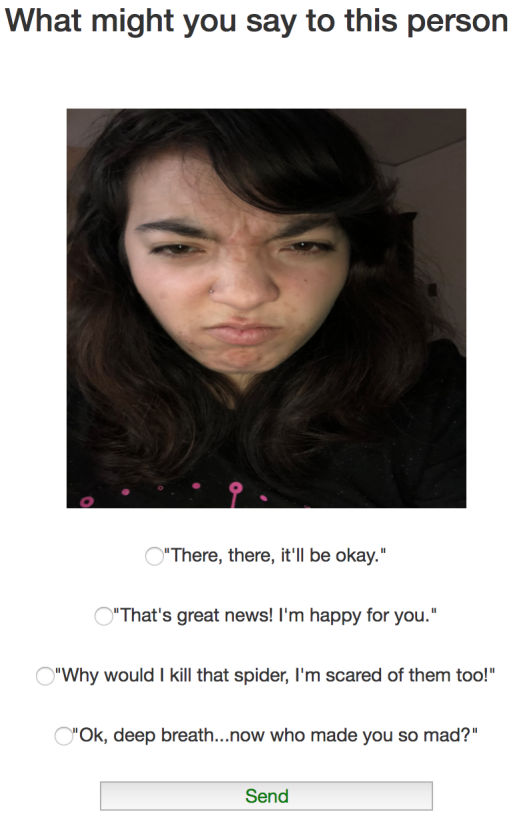


Figure 3.43 "what might you say based on their emotion" of static image application

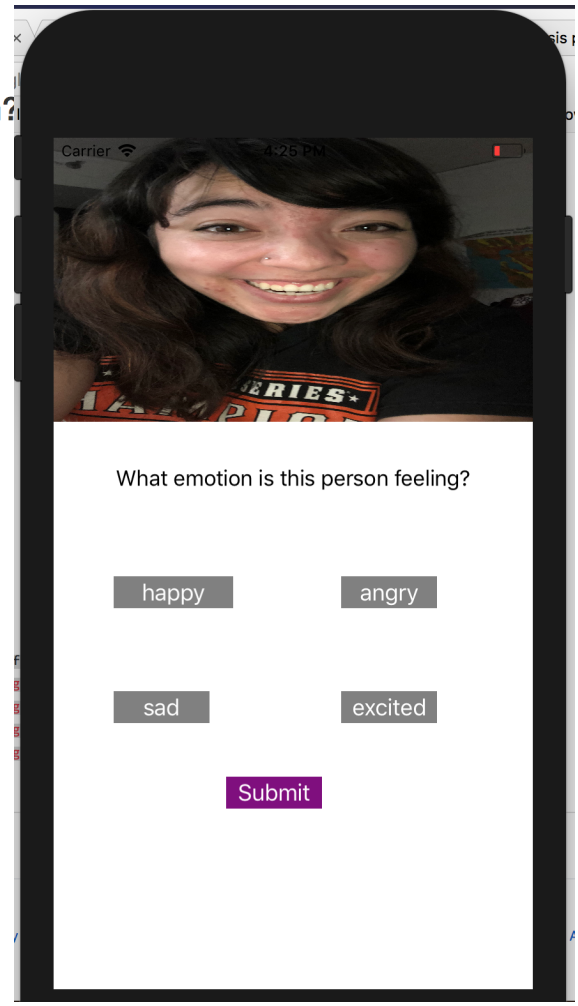


Figure 3.44 "guess their emotion" of iOS application

Example Questions from Static Image Apps

4 Web Application Development

4.1 First Version of Web Application

First, we drew out with pen and paper the design and flow of our application screens. We knew we wanted to make our applications accessible to more people, so we knew what languages to start with: HTML, CSS, and JavaScript for the web. We drew out what the homepage would look like, what buttons would go where, and where each button would lead to. Figure 4.1 shows our first sketch of the flow of what was the whole project at the time.

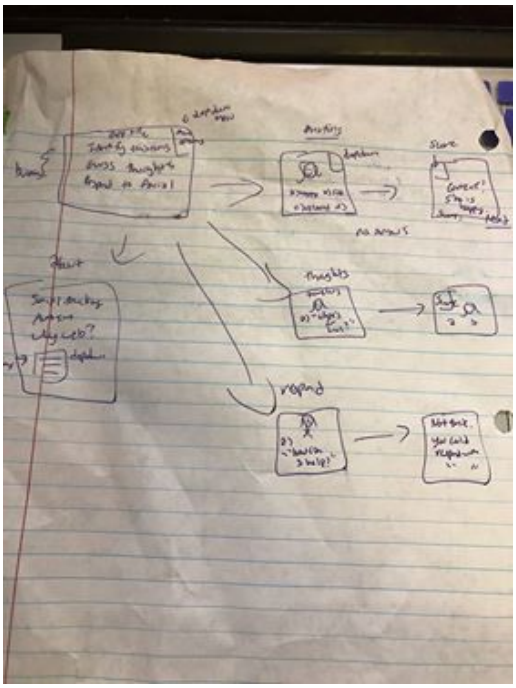


Figure 4.1 Our initial drawing of the flow of one application, October 2017

From there, we began developing static web pages in HTML, CSS, JavaScript, and jQuery (which we will look at more in the next section) that ran locally on our machines.

Static web pages almost solely use client-side languages and are more simple. This was alright for testing each application, but it was difficult to save answers from the web pages. In November 2017, we found a script on Rails Rescue blog

[17] that could save responses to a Google Form. First, we parsed the input for the form names:

```
var $inputs = $form.find("section, question, answer");
```

Then, we serialized the data in the form.

```
var serializedData = $form.serialize();
```

Lastly, we fired off an ajax request to our Google Form's URL with the parsed answers we just serialized.

```
request = $.ajax({
        url: "https://script.google.com/macros/s/AKfycbwWYK6rOZ4XAF92hQ
        type: "post",
        data: serializedData
    });
```

Answers were being saved to a Google Form as early as October, but it was inconsistent. The answers did not always save in the correct question order, and there were sometimes blank spots, but they were saved.

4.2 Dynamic Pages and Servers

In order to better save answers, we decided to make our web pages dynamic rather than static. Dynamic web pages have code written in a server-side language, and we needed a server to save our answers to. Most of our time has since been spent working on the back-end code to power the applications. Currently, the web pages do not look as pretty or complex as they could look because we have been focused more on the back-end code than front-end code, and we will work on front-end code in the near future.

In order to make the web pages dynamic, we converted our web applications to Django. Django is a free and open-source Python web framework that handles dynamic web actions related to servers such as user authentication and forms. [19] This helps us get the web application hosted on Digital Ocean, a cloud computing platform which makes the application accessible by anyone in the

world. More specifically, this project is deployed to the world wide web on Digital Ocean droplets, which are cloud servers for personal use. [8] Anyone can view and partake in this project by visiting http://esiegle.digital.brynmawr.edu/ in their browser. Google Chrome will tell you "Deceptive site ahead. Attackers on 104.131.74.54 may trick you into doing something dangerous like installing software or revealing your personal information (for example, passwords, phone numbers, or credit cards)." Get past it by clicking *details* followed by *visit this unsafe site*

4.25 Web Application Programming Setup

Initially, the static web pages were developed in Sublime Text. Once we migrated over to using Digital Ocean, we began solely using *vim* because we had to SSH in to Digital Ocean.

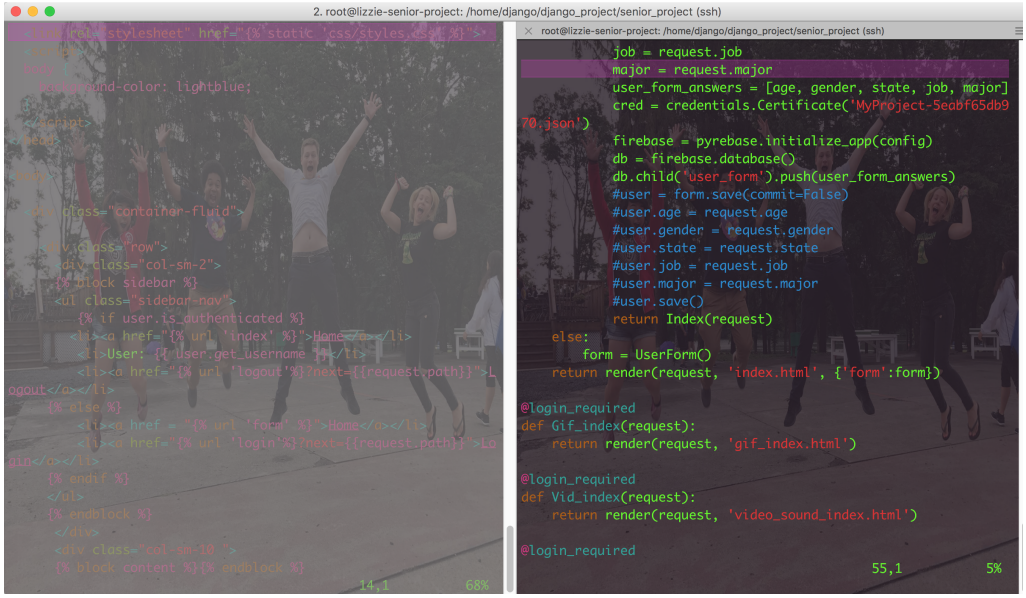


Figure 4.3 Answers for each web app saved to Firebase

4.4 Web Application Design

For this project, the web pages are designed with HTML and CSS, using Bootstrap, a "free front-end framework for faster and easier web development" that "includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and...optional JavaScript plugins" [4] Bootstrap let us write less code for design. Then, the web pages are rendered in Django.

Django Forms are used to receive background information on the user on the first page, handling automatic form verification for us and checking that each question is answered. For each question about emotions, the answer options are represented by radio buttons, and the answer responses are saved with the JavaScript language and jQuery library. According to the jQuery website, it "...is a fast, small, and feature-rich JavaScript library...[making] things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers." [12] In other words, jQuery makes it easy to access the HTML webpage in JavaScript when communicating between both the client and server sides. We also use jQuery to check that each question has an answer: otherwise, a screen pop ups telling the user the first question that they did not answer.

4.45 Serving Static Files in Django

The transition from the initial design of the static website that saved data with the Google Form to the Django web app was not seamless because Django apps must contain a special section solely for static files. In a folder called *static*, we had to place all of our styling code. Then, we connected the styling code (CSS) to each web page or each template with code like the code snippet below. This particular code snippet is from a template file called *base_generic.html*. Many of our app's pages inherit from this file so that they all have the same design and the same features. This reduces the amount of code written or copy-and-pasted.

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/
bootstrap/3.3.7/css/bootstrap.min.css">
```

```
<link rel="stylesheet" href="//maxcdn.bootstrapcdn.com/
bootstrap/3.2.0/css/bootstrap-theme.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/
jquery/1.12.4/jquery.min.js"></script>
{% load staticfiles %}
{% load static %}
<link rel="stylesheet" href="{% static 'css/styles.css' %}">
```

4.5 Current Web Application Back-end

The answers from the radio buttons are currently saved to a Google Firebase database. Google Firebase offers numerous developer products across multiple platforms, but the one our project uses is the real-time database. This "cloud-hosted NoSQL database...lets you store and sync data between your users in realtime", providing real-time syncing for the radio button answers once they are converted to JSON data.[10] We decided to use Firebase over other database services because of its extensive documentation, ease of use for cross-platform applications (time-permitting, the web applications will sync with iOS applications), and ease of use for smaller applications and prototypes. Additionally, according to the 2018 StackOverflow developer survey, Firebase is used by 14.5% of all developers and is the ninth most popular platform, ahead of Azure, Heroku, and the rest of the Google Cloud Platform. [15]

Each level is on one web page to minimize the number of calls we make to Firebase. The user just scrolls down to see each question for that level. Currently when a user finishes a level, the *submit* button checks that each question is answered, and then publishes the answers (which have been converted to JSON form to be read by Firebase) to Firebase. Figure 4.3 displays how Firebase saves the answers for each separate level in the Firebase console (accessible via the web.)

Figure 4.3 Answers for each web app saved to Firebase

Specifically, we use the Python wrapper called Pyrebase as it came with more methods. We access the last answers saved on the last "submit" button clicked with

```
db.child('static_part_1').order_by_key().limit_to_first(1)
.get()"
```

where "static_part_1 is the web page or level, "order_by_key" kept the answers in the order they were clicked on the page, and "limit_to_first(1)" returns just the answers submitted on the button click instead of all of the ones underneath "static_part_1".

4.55 Additional External APIs in Web Applications

After clicking the *submit* button, the user is taken to a page that shows a gif and a graphic based on the number of answers the user got correct. After three levels, the user is directed to the next web application (first they have static images, then silent gifs, and lastly videos with sound.)

External APIs used include Cloudinary to render and position gifs and videos in Django (which developers can not do with CSS in Django applications), the Giphy API to render gifs based on the user's score, and PubNub's EON graph framework to render graphs showing the user's score after each level.

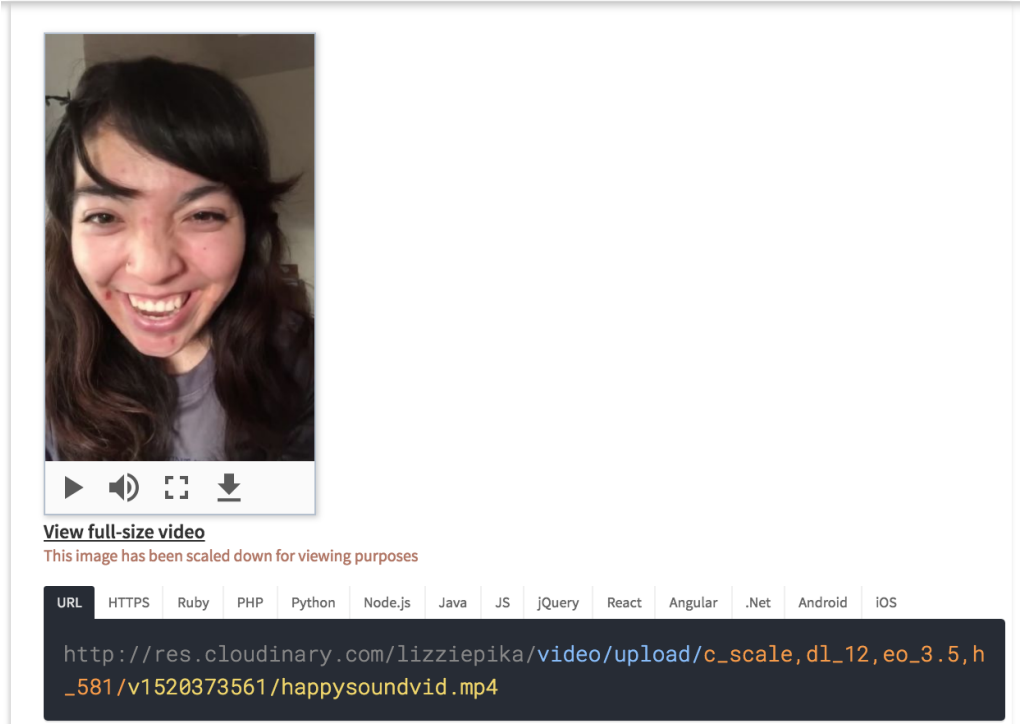


Figure 4.55 Cloudinary URL

In summary, the languages and libraries we have implemented so far are Python, JavaScript, HTMl, and CSS. The external libraries, APIs, and services we are using include Firebase database, Digital Ocean for servers, Cloudinary API, PubNub, Giphy, and the Django web framework.

5 iOS Application

5.1 iOS Application Design

The iOS Application is written in Swift and saves data to Firebase. The design is roughly the same as the web applications, such that the background is the same color, the images are centered, Images are rendered locally in Swift, so the Cloudinary is unnecessary. The data visualizations are rendered with PubNub EON in a webview so that JavaScript is written for an iOS application.

6 Challenges and Solutions

6.1 Django Challenges

We had never used Django before so a lot of time was spent learning it and working to get our applications on it. It took time to get the Django app up-and-running so that we could view our web pages. A surprising and interesting fact about Django is that it can sometimes distort images and graphics. A large challenge of using Django was that when we tried to convert our first static web pages to Django, the images rotated ninety degrees. They could not be rotated back with CSS, so we spent much time Googling different ways to rotate them, like with CSS or JavaScript or a JavaScript framework. Finally, we decided to use an API. In order to rotate them back, we implemented Cloudinary, which provides APIs for image and video manipulation, cloud storage, and file upload in the cloud across platforms. Because of Cloudinary, not only is every picture

oriented correctly, but the gifs and videos appear directly in the web page. Without Cloudinary, no gifs or videos showed up in our Django application. (Figure 0.7 shows the Cloudinary console.)

6.2 Database Challenges

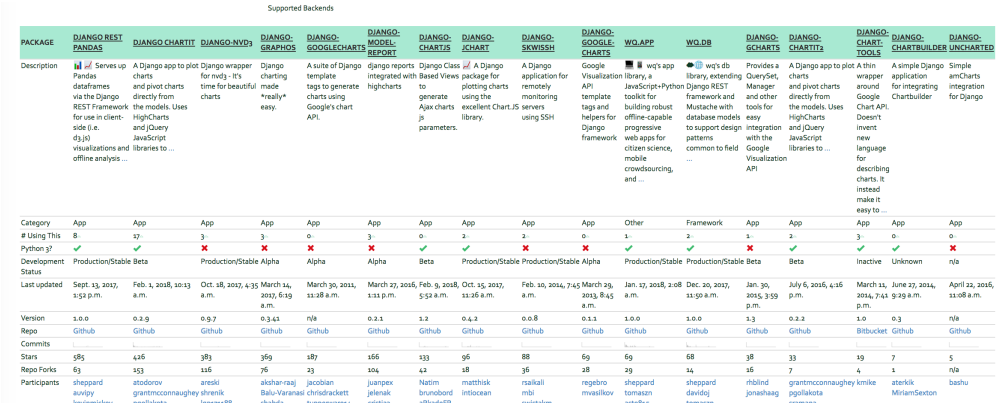
We encountered difficulties parsing the Pyrebase object that was returned after we retrieved it from the Firebase database. We had to import *json* to call *json.dumps* to turn that object into a string.

The web applications are done and ready to be tested by users. We plan on making the web pages look fancier and more aesthetic, and are currently drawing out more prototypes as their design at the moment is extremely basic because we were focusing on the backend code that powers the applications rather than the frontend design. The only backend changes we may make are to implement a login system so Firebase does not use randomly-generated strings for each user, or to pull the answers from Firebase and display the results on a web page to let users know how they performed.

The iOS application is incomplete at the moment. It has the first level (recognizing emotions), but does not save to Firebase yet or have any other levels or any other media types besides static images. Additionally, we successfully implemented a video chat feature, but are unable to successfully merge it with the Django application so far.

6.3 Data Visualization

A small challenge we faced was how to create and render charts in django. There were roughly eighteen options, as shown in Figure 6.3.



Supported Backends

PACKAGE	DJANGO REST PANDAS	DJANGO CHARTIT	DJANGO-NVD3	DJANGO-GRAPHOS	DJANGO-GOOGLECHARTS	DJANGO-MODEL-REPORT	DJANGO-CHARTJS	DJANGO-JCHART	DJANGO-SKWISSH	DJANGO-GOOGLE-CHARTS	WQ.APP	WQ.DB	DJANGO-GCHARTS	DJANGO-CHARTIT2	DJANGO-CHART-TOOLS	DJANGO-CHARTBUILDER	DJANGO-UNCHARTED
Description	Serves up Pandas dataframes via the Django REST Framework for use in client-side (i.e. d3.js) visualizations and offline analysis ...	A Django app to plot charts and pivot charts directly from the models. Uses HighCharts and jQuery JavaScript libraries to ...	Django wrapper for nvd3 - It's time for beautiful charts	Django charting made *really* easy.	A suite of Django template tags to generate charts using Google's chart API.	django reports integrated with highcharts	Django Class Based Views to generate Ajax charts.js parameters.	A Django package for plotting charts using the excellent Chart.JS library.	A Django application for remotely monitoring servers using SSH	Google Visualization API template tags and helpers for Django framework	wq's app library, a JavaScript+Python toolkit for building robust offline-capable progressive web apps for citizen science, mobile crowdsourcing, and ...	wq's db library, extending Django REST framework and Mustache with database models to support design patterns common to field ...	Provides a QuerySet, Manager and other tools for easy integration with the Google Visualization API	A Django app to plot charts and pivot charts directly from the models. Uses HighCharts and jQuery JavaScript libraries to ...	A thin wrapper around Google Chart API. Doesn't invent new language for describing charts. It instead make it easy to ...	A simple Django application for integrating Chartbuilder	Simple amCharts integration for Django
Category	App	App	App	App	App	App	App	App	App	App	Other	Framework	App	App	App	App	App
# Using This	8	27	3	3	0	3	0	2	2	0	1	2	1	2	3	0	0
Python 3?	✔	✔	✖	✖	✖	✖	✔	✔	✖	✖	✔	✔	✖	✔	✔	✔	✖
Development Status	Production/Stable	Beta	Production/Stable	Alpha	Alpha	Alpha	Beta	Production/Stable	Production/Stable	Alpha	Production/Stable	Production/Stable	Beta	Beta	Inactive	Unknown	n/a
Last updated	Sept. 13, 2017, 1:52 p.m.	Feb. 1, 2018, 10:13 a.m.	Oct. 18, 2017, 4:35 a.m.	March 14, 2017, 6:19 a.m.	March 30, 2011, 11:28 a.m.	March 27, 2016, Feb. 9, 2018, 1:11 p.m.	5:52 a.m.	Oct. 15, 2017, 11:26 a.m.	Feb. 10, 2014, 7:45 a.m.	March 29, 2013, 8:45 a.m.	Jan. 17, 2018, 2:08 a.m.	Dec. 20, 2017, 11:50 a.m.	Jan. 30, 2015, 3:59 p.m.	July 6, 2016, 4:16 p.m.	March 11, 2014, 7:41 p.m.	June 27, 2014, 9:29 a.m.	April 22, 2016, 11:08 a.m.
Version	1.0.0	0.2.9	0.9.7	0.3.4.1	n/a	0.2.1	1.2	0.4.2	0.0.8	0.1.1	1.0.0	1.0.0	1.3	0.2.2	1.0	0.3	n/a
Repo	Github	Github	Github	Github	Github	Github	Github	Github	Github	Github	Github	Github	Github	Github	Bitbucket	Github	Github
Commits																	
Stars	585	426	383	369	187	166	133	96	88	69	69	68	38	33	19	7	5
Repo Forks	63	153	116	76	23	104	42	18	36	28	29	14	16	7	4	1	n/a
Participants	sheppard auvipy	atodorov grantmcconnaughey	areski shrenik	akshar-raaj Balu-Varanasi	jacobian chrisdrackett	juanpex jelenak	Natim brunobord	matthisk intiocean	rsalkali mbi	regebro mvasilkov	sheppard tomaszn	sheppard davidoj	rhblind jonashaag	grantmcconnaughey pgollakota	kmike	aterkik MiriamSexton	bashu

Figure 6.3 Django charting options

First, we successfully implemented the Django-GraphOS library for the first two levels (one of which is shown in figure 0.9). However, we could not render the chart with the third level of the static image application, and then the first two charts stopped rendering. Sometimes even though all six questions were answered and pushed to Firebase, and they were also successfully retrieved from Firebase, a dictionary key error was produced on 'q4', or question number four. This inconsistency of Django-GraphOS as well as testing out each possible charting library or API shown in figure 6.35 took over forty-eight hours before we finally decided to use PubNub's EON chart framework which rendered the chart from the client with JavaScript instead of the server with Python.

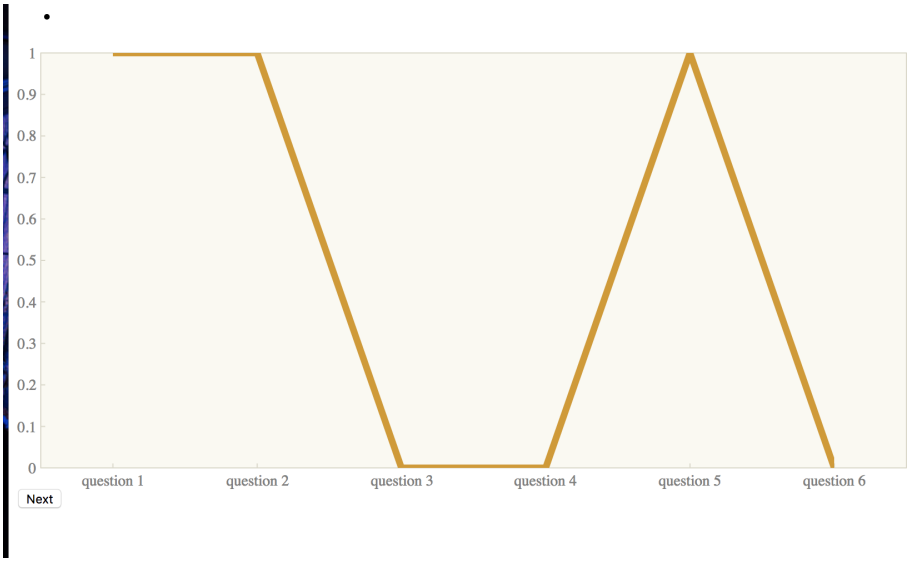


Figure 6.35 Django graphOS for one level

7. Testing the Applications

7.1 Hypothesis

We hypothesized that users would perform the best on the video application because a video gives more information.

7.2 Background of Users

So far, we have been unable to get users who are on the Autism spectrum. Instead, we have been testing on people we know who volunteer. By filling out the initial form, we have received some information about them. Many users so far have been computer science students, engineers, or prospective computer science students.

7.3 Results of User Testing

Surprisingly, users have performed best on the static image applications for each section or level. They tend to perform worst on the video application.

7.4 Possible Explanations of Results

We believe that users perform worse on the video application precisely because the video provides more information, context, and detail. Eyes can move, mouths can move, and bodies can move. Sound is emitted. With so much extra information, we think users find that there are multiple possible answers. For our users who do not have clear challenges recognizing emotions based on their background and them not being diagnosed as being on the Autism spectrum (that we know of), they should not find some of the questions difficult.

8 Goals

After usage of these applications, we would love to see improvement in interpersonal relationships, play and leisure interactions, and social skills; however, this is difficult to measure. We aim to target skills like awareness of feelings, recognition of non-verbal communication, starting a conversation, and making small talk. We hope to one day get the IRB permission to test our applications in schools like Stanbridge or just on people who have been diagnosed with Autism. Outcomes with those users will be based not just on how users score on the questions, but also through questionnaires which ask about their background which could explain some of their answers. This is similar to how studies with ASD/HFA children measured progress and results. They targeted the same skills and more, such as politeness, introducing oneself to others, maintaining a conversation, ending a conversation, making small talk, negotiating with others, responding to teasing and bullying, hygiene, dining etiquette, and dating etiquette. [14]

9 Appendix

This web project's droplet can have backup versions in case one the droplet (server) goes down, is configured with 1 GB memory. Readers, users, or anyone with access to the internet, can visit the project at the following public network or IP address: 104.131.74.54.We edit the project by SSH-ing on our local machine to our Digital Ocean droplet.

Bibliography

[1] *Avokiddo Emotions*. Accessed:2018-02-17.

[2] Dr. Jed Baker. *Social Skills Picture Book*. Future Horizons, 2001. ISBN: 1885477910.

[3] Dr. Jed Baker. *Social Skills Picture Book for High School and Beyond*. Future Horizons, 2006. ISBN: 9781932565355.

[4] *Bootstrap Get Started*. Accessed:2018-04-02.

[5] Lois Jean Brady. *Apps for Autism - an Essential Guide to Over 200 Effective Apps for Improving Communication, Behavior, Social Skills, and More!* Future Horizons, 1996. ISBN: 9781935274490.

[6] *Development of Emotion Recognition in Individuals with Autism*. Accessed:2018-03-24.

[7] *Diagnostic and Statistical Manual of Mental Disorders*. Accessed: 2018-03-24.

[8] *Digital Ocean Droplets*. Accessed: 2018-02-17.

[9] *emotionary+ by Funny Feelings*. Accessed:2018-02-17.

[10] *Firebase Realtime Database*. Accessed:2018-02-17.

[11] MD; Marc Woodbury-Smith MD; Bryan King MD; James McCracken MD; Matthew State Fred Volkmar MD; Matthew Siegel. “Practice Parameter for the Assessment and Treatment of Children and Adolescents With Autism Spectrum Disorder”. In: (2014).

[12] *jQuery*. Accessed: 2018-02-17.

[13] Jr. Elisabeth A. Martin Marshall P. Duke Stephen Nowicki. *Teaching Your Child the Language of Social Success*. Peachtree Publishers, 1996. ISBN: 1561451266.

[14] Janine Y. Rachman Pamela J. Crooke Ryan E. Hendrix. "Brief Report: measuring the effectiveness of teaching social thinking to children with Asperger syndrome (AS) and High Functioning Autism (HFA)." In: (Accessed:2018-02-11).

[15] *Stack Overflow Developer Survey Results 2018*. 2018.

[16] *Stanbridge Academy*. Accessed:2018-02-17.

[17] "Step by step setup to send form data to Google Sheets". In: ().

[18] *Touch and Learn-Emotions*. Accessed:2018-02-17.

[19] *What is Django*. Accessed:2018-02-17.

[20] Michelle Garcia Winner. "Assessment of Social Skills for Students with Asperger Syndrome and High-Functioning Autism". In: (2001-2002).

[21] Michelle Garcia Winner and Michelle Crooke. *Socially Curious and Curiously Social-A Social Thinking Guidebook for Brigh Teens and Young Adults*. Think Social Publishing, 2011. ISBN: 9780884272021.