

Project Name*Emoji Passwords by S5 Software***Date**April 6th 2017

Name	Email	Student Number
Theodore Kachulis	tedkachulis@cmail.carleton.ca	100970278
Mathieu Schmid	mathieu.schmid@carleton.ca	100970437
Parth Patel	parthpatel5@cmail.carleton.ca	100963711
Mike Stupich	mikestupich@cmail.carleton.ca	100973305

Part 1: Sample Password Method Logs Analysis

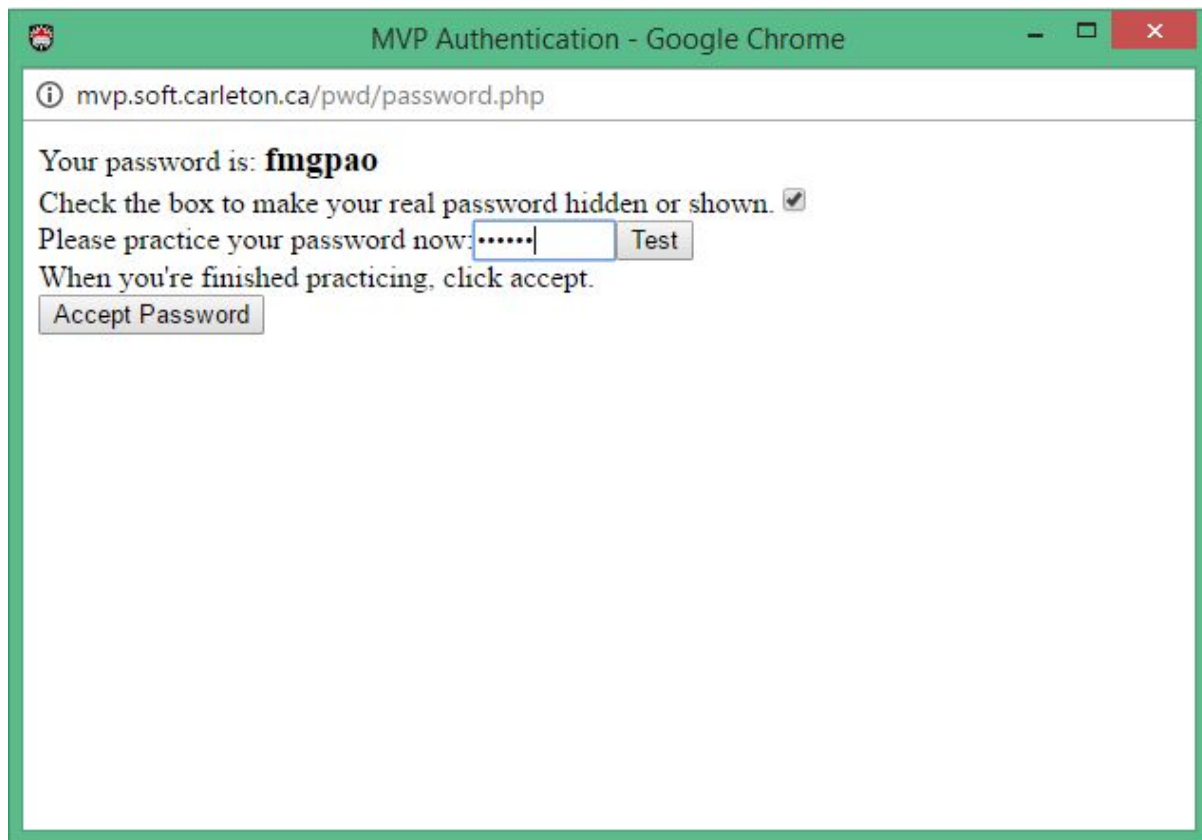
Text28: Passwords of 6 random lower-case letters.

Pros:

- Randomly generated passwords prevent the use of rainbow-table databases which use commonly used passwords
- Every user will have experience with text-based passwords
- All lowercase letters and no numbers remove the mix up between commonly mixed up letter (l and I, O and 0, etc..)
- 6 digits passwords follows Miller's Magic Number 7 plus or minus 2 which deals with lowering the user's cognitive load.

Cons:

- Hard to remember a randomly generated password
- Roughly 165,765,600 possible passwords, half as many of the other two password systems. Brute force will find this password before the other two.



The screenshot shows a web browser window with the title "MVP Authentication - Google Chrome". The address bar displays "mvp.soft.carleton.ca/pwd/password.php". The main content area of the page contains the following text and form elements:

Your password is: **fmgpao**

Check the box to make your real password hidden or shown. ☒

Please practice your password now:

When you're finished practicing, click accept.

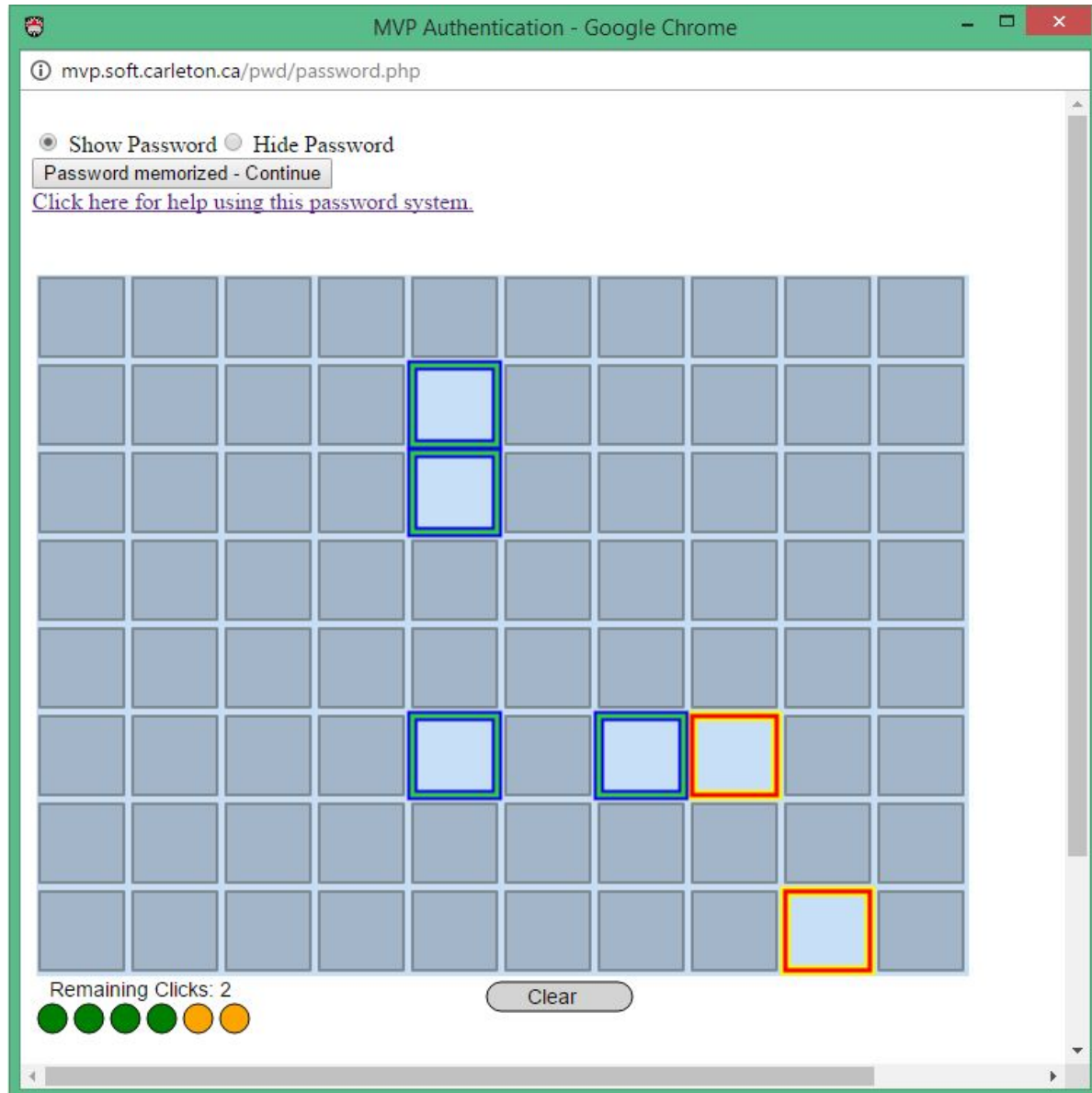
Blankpt28: Passwords of 6 random tiles, chosen from a blank grid of 80 tiles.

Pros:

- Prevents use of rainbow-table databases which use commonly used passwords
- Order doesn't matter in clicking tiles

Cons:

- Incredibly hard to remember tiles with no reference points
- Users have no familiarity with passwords systems such as this one



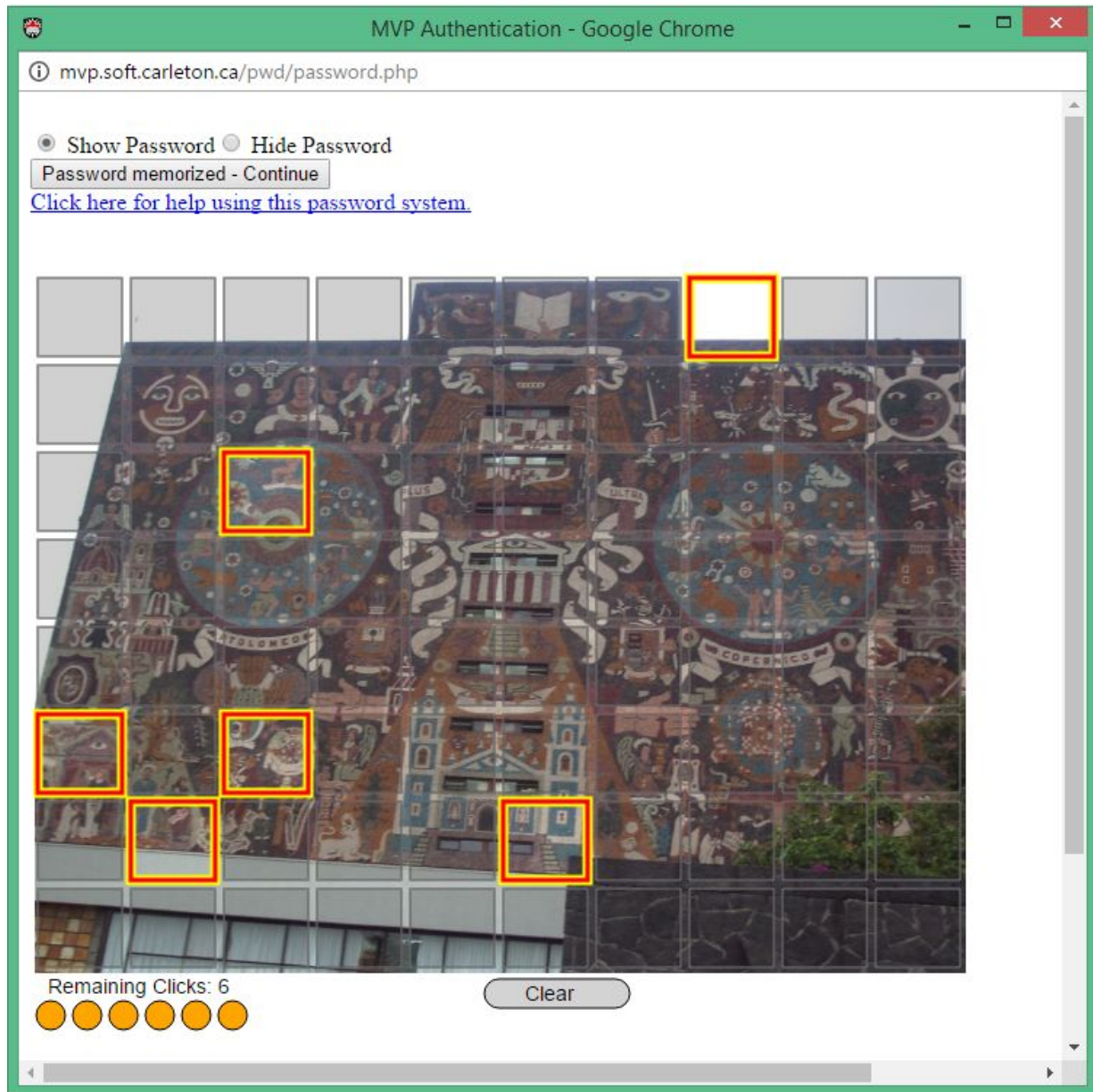
Imagept28: Passwords of 6 random tiles, chosen from a image made of 80 tiles.

Pros:

- Easier to remember point locations with image references
- Prevents use of rainbow-table databases which use commonly used passwords
- Order doesn't matter in clicking tiles

Cons:

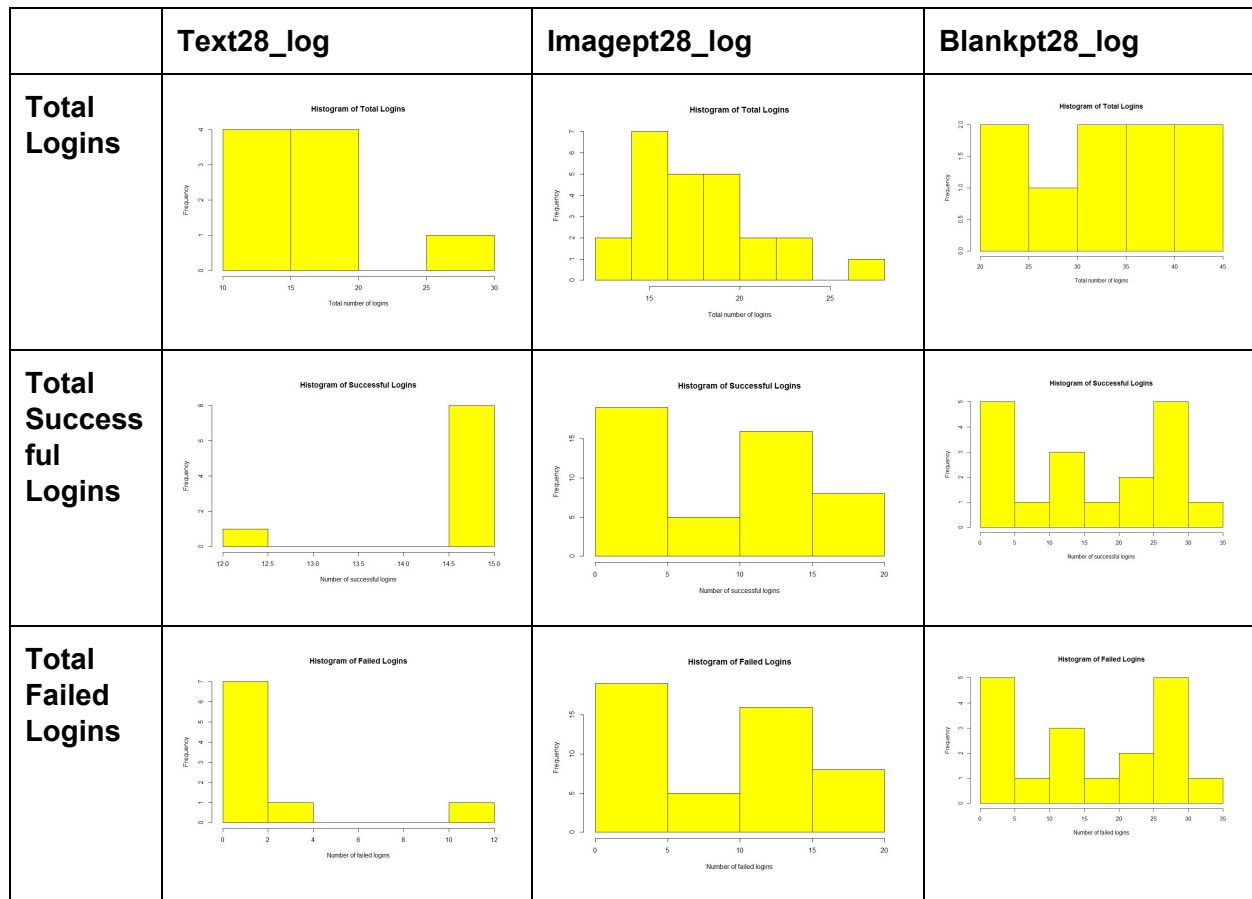
- Users have no familiarity with passwords systems such as this one
- Image references help with remembering the password, but 6 random points are still difficult to remember



Mean Median and Standard Deviation of Text28_log
Mean of users events that are logins 16.88889 Mean of successful logins per user 14.66667 Mean of failed logins per user 4 Median of users events that are logins 16 Median of successful logins per user 15 Median of failed logins per user 2 Standard deviation of logins per user 4.226241 Standard deviation of successful logins per user 1 Standard deviation of failed logins per user 4.636809
Mean Median and Standard Deviation of Blankpt28_log
Mean of users events that are logins 33.33333 Mean of successful logins per user 27.55556 Mean of failed logins per user 7.428571 Median of users events that are logins 34 Median of successful logins per user 30 Median of failed logins per user 6 Standard deviation of logins per user 7.745967 Standard deviation of successful logins per user 4.558265 Standard deviation of failed logins per user 5.12696
Mean Median and Standard Deviation of Imagept28_log
Mean of users events that are logins 18.20833 Mean of successful logins per user 15.29167 Mean of failed logins per user 3.5 Median of users events that are logins 17.5 Median of successful logins per user 15 Median of failed logins per user 2.5 Standard deviation of logins per user 3.501294 Standard deviation of successful logins per user 1.805286 Standard deviation of failed logins per user 2.328315

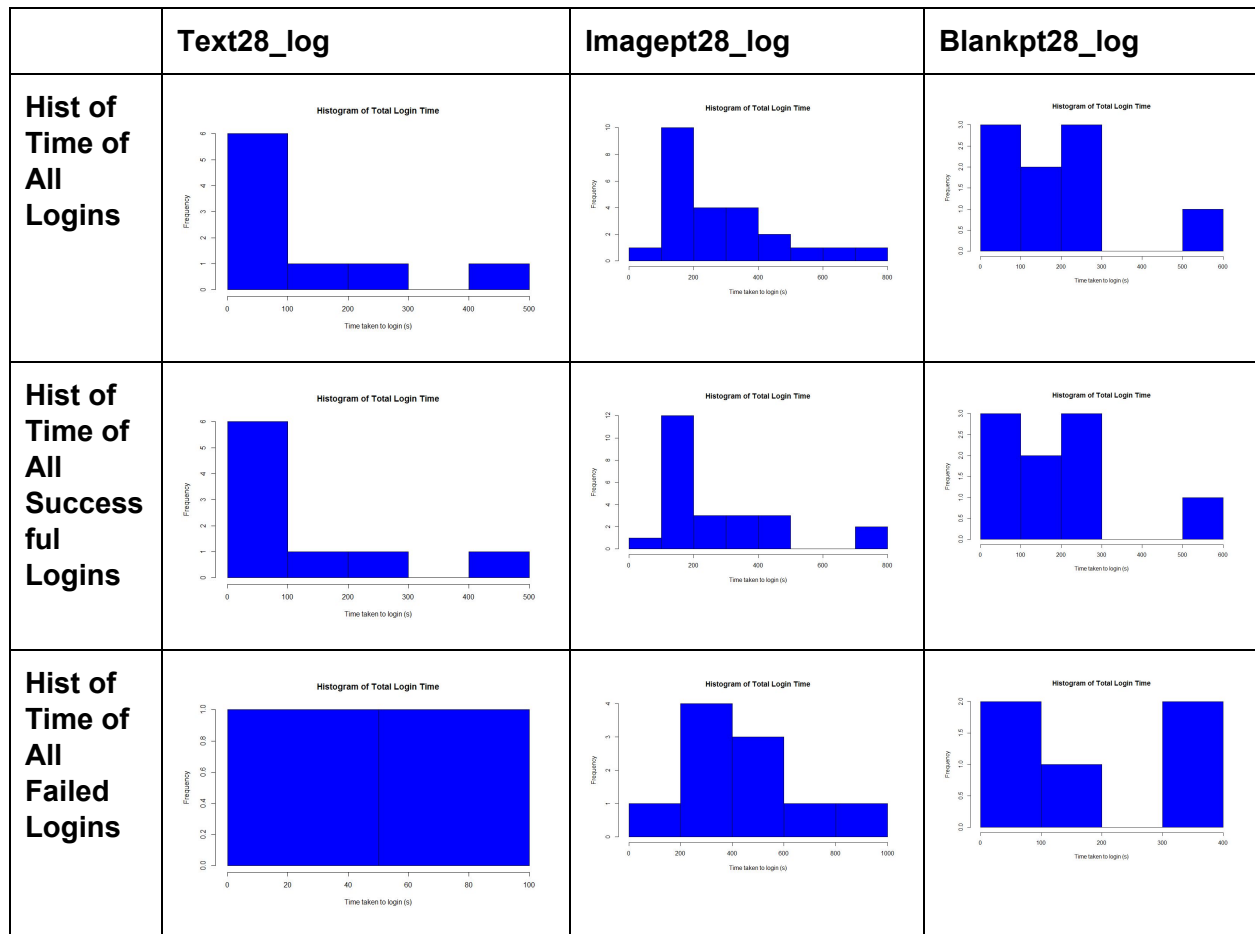
All comparison graphs and R source code available here:

<https://drive.google.com/open?id=0B41LGpA8OtX7anIFdWtpaWQ1VFk>



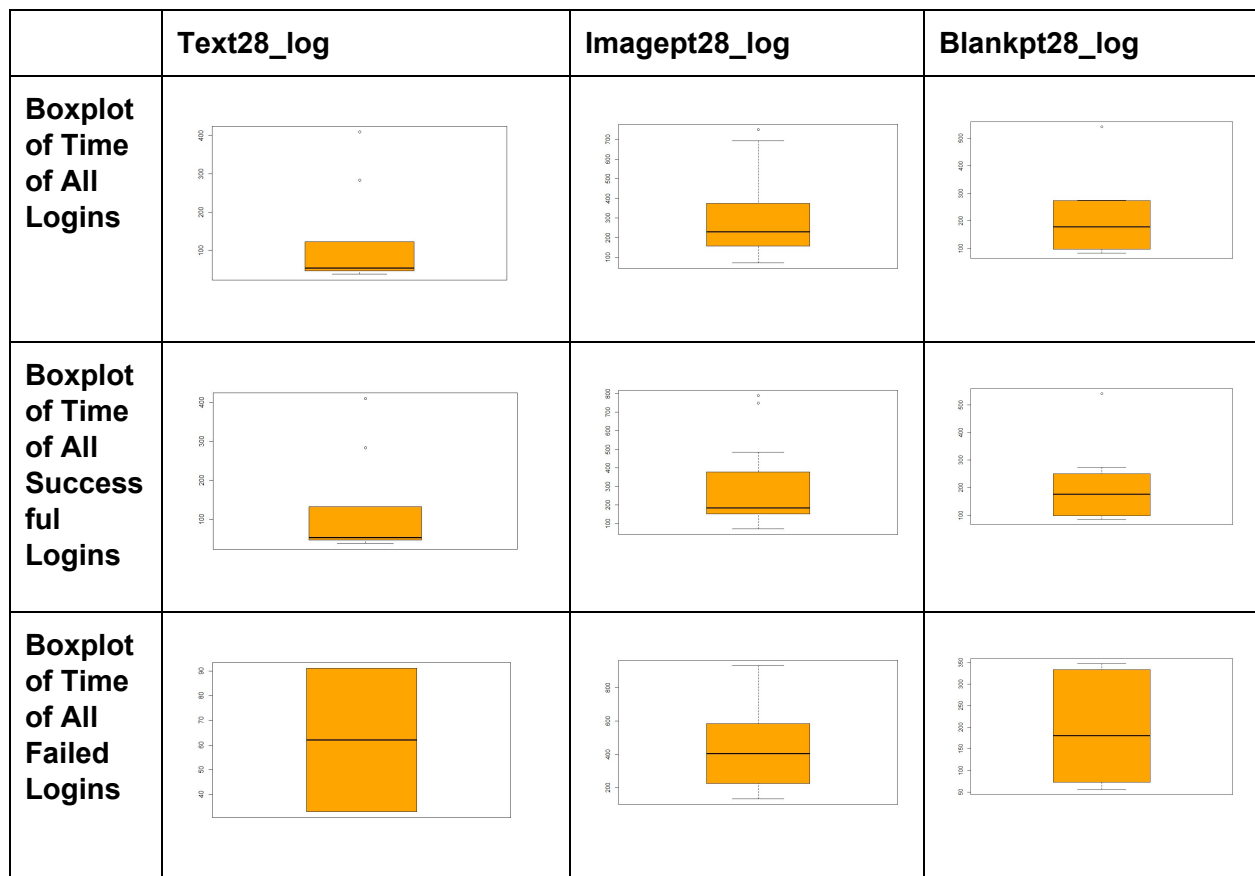
These are the histograms of all the users logins, both successful and failed on the three different password schemes. By analyzing the above histograms we can determine:

1. From the Total Logins, we can see that users logged in the most on the Blank password scheme, followed by the Image scheme, and the Text scheme was logged in the least on.
2. From the Successful Logins, we can see that the Text and Image scheme had much fewer successful logins than the Blank scheme.
3. From the Failed Logins, we can see that the Text scheme had almost all users between 0-2 failed logins, whereas the Image scheme had a more evenly distributed failure rate, between 0-7 failed logins, however there were the most users that failed the login 7 times. And finally we can see that the Blank scheme had some users that seemed to get it, and had between 0-2 failed logins, however there was quite a large distribution of failed attempts, with some users experiencing between 10-14 failed login attempts.



These are the histograms of all the time taken by the users to login on the three different password schemes. By analyzing the above histograms we can determine:

1. From looking at the histograms of the total times for all logins, we can see that users felt they were ready to login the fastest when using the Text scheme, followed by what seems to be fairly even time for the blank and Image schemas.
2. From looking at the time until successful logins, we can see that once again, the Text scheme proved to be the fastest, followed by the Blank and Image schemas.
3. The time to failed login histogram shows that the Text scheme users took very little time to attempt to login which would most likely mean they felt confident in what they were typing, whereas the image and blank scheme users varied quite a bit, with the image users taking even longer. Perhaps this is due to users of the image scheme feeling as if they might remember a part of the password if they keep looking.



These are the boxplots of the time it took users to login on the three different password schemes. By analyzing the boxplots we can deduce most of the same conclusions as were stated from looking at the histograms. One thing box plots do show us is the difference in the average times for the different schemes. We can see that the Text scheme has a fairly condensed box, with most of the users taking slightly under 100s. We can see that the Blank scheme and the Image scheme have much larger boxes, which shows that users were having more issues remembering their passwords, making things take longer.

We believe that this largely comes down to what users have grown accustomed to using as a password scheme. While we are not fans of the Blank password scheme, we do believe that the Image password scheme would be an effective password scheme that would match the memorability of a text password if we were accustomed to it. However, as the charts show, at the moment users are not nearly as proficient with

remembering a randomly generated image password as they are a text password. So for the time being, text based passwords are a much more reliable way for users to protect their data.

To conclude, we can quite clearly see that the Text scheme is the best, followed by the Image scheme, and then the Blank scheme. This agrees with our preliminary assessment of the password schemes, where we found the Text password to be simplest.

Part 2: Design, Implementation, Statistical Inference – 60%

We have decided to design a password scheme much like phone or bank pin system. Like a pin system, it consists of an array of characters that is the user's password. However, instead of the traditional choice of using numbers or letters, we have chosen emojis as the characters for the password. Our main reasoning behind this decision involves the memorization of a randomly generated password. When a user is given a randomly generated password that consists of letters and/or numbers, it is often difficult to memorize it, since it is just a random combination with no meaning. Our emoji offers some more context and imagery that letters and numbers do not provide.

Using the underlying principles of the Image Superiority effect^[1], which states that pictures and images are more likely to be remembered than words, we hope to make it easier for users to remember randomly generated passwords. The main method of memorization would be by combining their given emojis to create a kind of mental storyboard to ingrain in their memory the specific emojis and the order in which they are to be entered. We will investigate this alternative memorization technique, and how it compares to the widely used text-based password memorization.

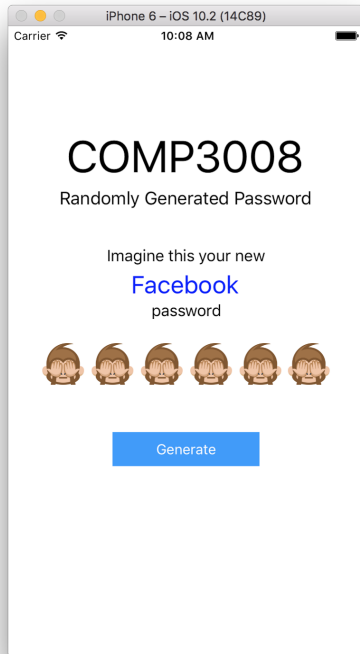
The security of our emoji-based password system is comparable to that of letter based system. We have a bank of 30 emojis, which we use to fill 6 positions in a password, giving us 30^6 combinations, which converts to 2^{30} bits. This is a larger

bitspace than a 6 character single-case letter password. Our decision to have 6 characters in the password is a result of a compromise between memorability and security. Having fewer than 6 emojis would compromise security, giving less than 2^{28} bitspace. Having more than 6 emojis, on the other hand, makes it much harder to remember a password, which is a hindrance to users. Coincidentally, having 6 characters fits the Miller's Magical Number Seven, Plus or Minus Two^[2] in Interaction Design. Miller's theory suggests that the 7 ± 2 objects in question are some sort of "short term" memory that hold new perceptions yet to be processed. It can be thought of as a buffer between perception and processing of new information.

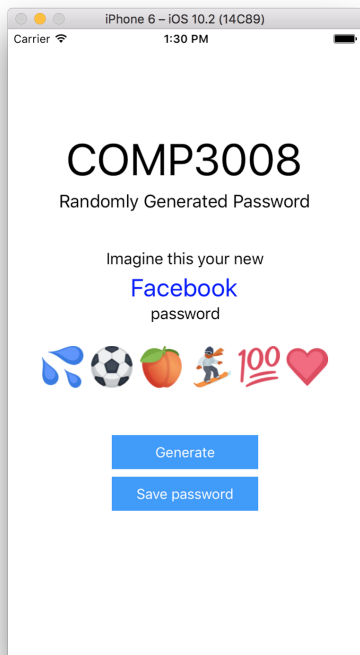
When using emojis, people use a lot of different face emojis. When using them in messages, for simple reading purposes, there is no issue with being able to differentiate between different "emotions". The issue does arise when a user is faced with the challenge of committing the face emojis to memory. We found that using different face emojis is actually detrimental to users' ability to accurately recall them for entering a password. This is due to the fact that all face emojis have the same shape and colour. When forced to memorize and recall these similar looking emojis, the users are forced to spend unnecessary time and effort to remember little details. This is an avoidable problem, as there is a vast collection of other, less popular, emojis that have unique shapes and colours. This reduces the similarity between the emojis used to generate the password.

An effective way of remembering a randomly generated sequence of emojis is to use the emojis to create a storyboard-like scene and walk through the scene when trying to recall it. This process is called the method of Loci^[3], or more commonly known as a memory palace. If we were to implement the system with only face emojis, the users would be restricted to only emotions to create their story (for example, happy crying winking). By incorporating different emojis, such as food items and vehicles, users can create more memorable stories. Such as, "soccer ball caught fire, so I cried and went home in the car". This type of method is used and taught by numerous memory experts.

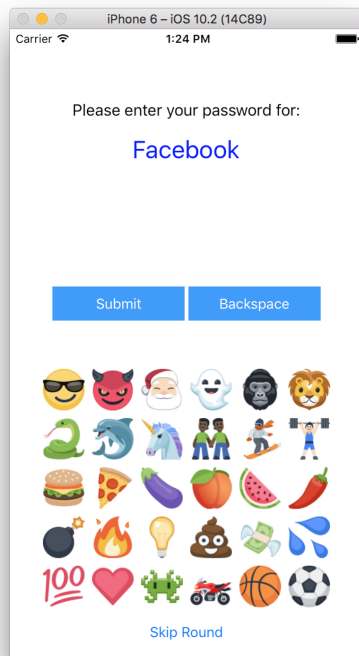
Password Scheme Implementation



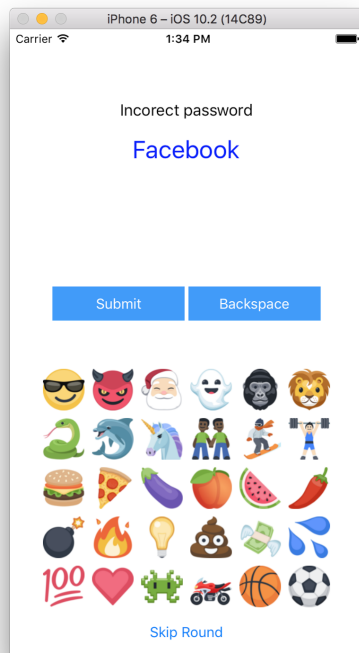
This is the initial screen the user will see when starting the simulation. This is running on an XCode simulator of the iPhone 6 on iOS 10.2. It states which system the password will be generated for, in this case, Facebook. All of the emoji slots are hidden by the monkey hiding his face emoji until the user clicks on the Generate button.



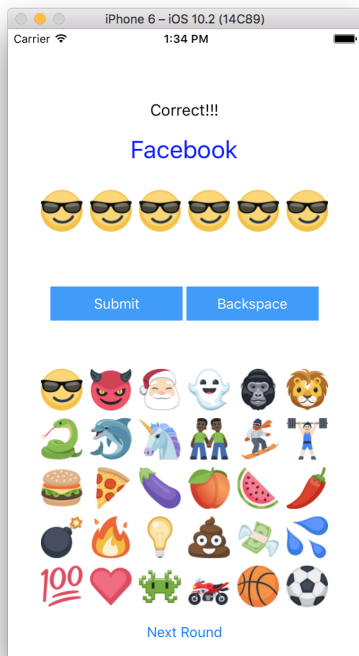
Every time the user clicks the Generate button, a new series of 6 randomly generated emojis is shown. The password is picked from a bank of 30 emojis. One thing which the user can do to help with memorizing the password is to keep generating it until they find one which they can make quick story out of.



When the user saves their desired password, they are brought to a new screen which contains a keyboard of the bank of 30 emojis. The user must now enter the password which they just memorized. At any point on this screen the user can click the 'Reset' button to advance to the next step of the simulation



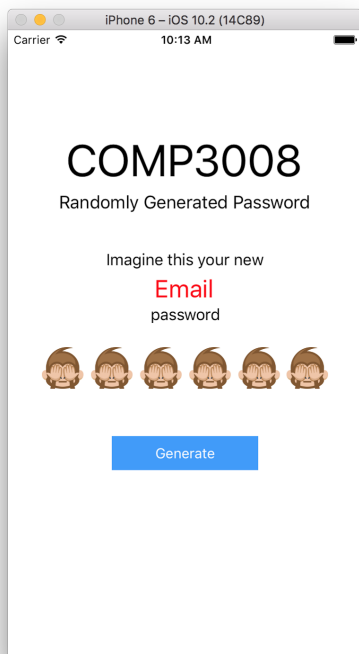
When the user enters an incorrect password or not enough characters, they are given an appropriate error message.



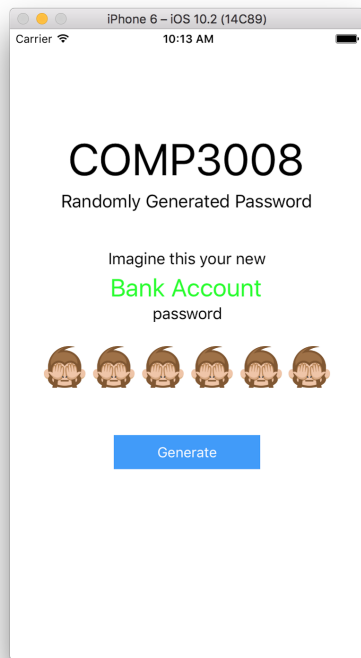
When the user enters the correct password, they are finished with this step of the process and now need to click on the 'Reset' button to advance to the following task.

Source code: <https://goo.gl/3is7di>

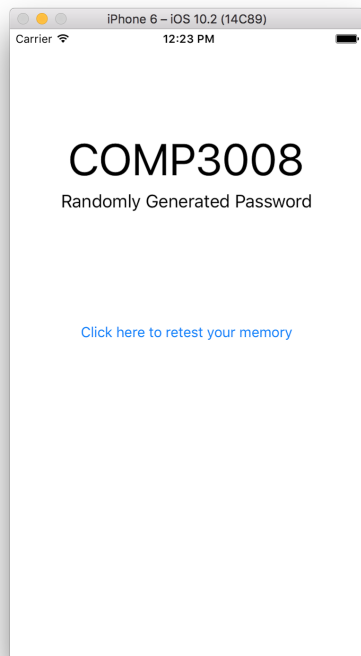
The code for before the multiple systems and retesting was implemented can be found above.



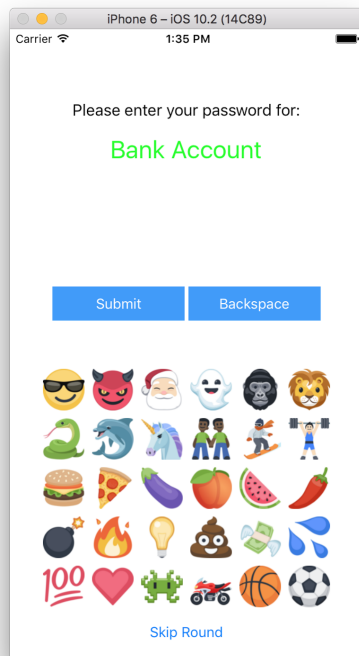
The three passwords they need to memorize are for Facebook, Email and their Bank Account.



Each system is colour coded in order to help the user associate a password to a different colour



When the user is done all three password setups, they are prompted to start memory testing portion of the simulation



Here they will be asked to re-enter the three passwords they memorized in the previous steps in a random order.

Source code:

The fully documented source code for this software can be found on my github here:
<https://github.com/schmidyy/EmojiPassword>

For a direct .zip download click this link:

<https://github.com/schmidyy/EmojiPassword/archive/master.zip>

Please note that in order to run this, you will need XCode 8.x (Mac OS X only) running an iPhone 6 or 7 on iOS 10.2 emulator. If there are any issues running the software, you can contact me directly at mathieuschmid@cmail.carleton.ca.

Further instructions can be found in the README.md

COMP3008 Password Scheme Survey

This is a survey for the second project of COMP3008. We are comparing memory efficiency for different randomly generated password schemes.

Please answer every question as accurately as possible.

There are 15 questions in this survey.

A note on privacy

This survey is anonymous.

The record of your survey responses does not contain any identifying information about you, unless a specific survey question explicitly asked for it. If you used an identifying token to access this survey, please rest assured that this token will not be stored together with your responses. It is managed in a separate database and will only be updated to indicate whether you did (or did not) complete this survey. There is no way of matching identification tokens with survey responses.

Memory

* I retain more information when it is presented in the form of images/visual aids rather than from text sources.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

?

1. Strongly Disagree

2. Disagree

3. Neutral

4. Agree

5. Strongly Agree

* It is significantly more difficult to memorize 6 random objects rather than 5.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

?

1. Strongly Disagree

2. Disagree

3. Neutral

4. Agree

5. Strongly Agree

* It is easy to remember a randomly assigned text password

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

?

1. Strongly Disagree

2. Disagree

3. Neutral

4. Agree

5. Strongly Agree

✳ Creating a word out of randomly generated letters helps me remember it.

☐ 1 ☐ 2 ☐ 3 ☒ 4 ☐ 5



- 1. Strongly Disagree
- 2. Disagree
- 3. Neutral
- 4. Agree
- 5. Strongly Agree

✳ Creating a story out of randomly generated images helps me remember them.

☐ 1 ☐ 2 ☐ 3 ☒ 4 ☐ 5



- 1. Strongly Disagree
- 2. Disagree
- 3. Neutral
- 4. Agree
- 5. Strongly Agree

Emojis

✳ I use emojis on a day-to-day basis.

☐ 1 ☐ 2 ☐ 3 ☒ 4 ☐ 5



- 1. Strongly Disagree
- 2. Disagree
- 3. Neutral
- 4. Agree
- 5. Strongly Agree

✳ I like the idea of being able to use emoji's as part of my passwords online. For example, "tedrocks123🐼🐼".

☐ 1 ☐ 2 ☐ 3 ☒ 4 ☐ 5



- 1. Strongly Disagree
- 2. Disagree
- 3. Neutral
- 4. Agree
- 5. Strongly Agree

Security

✳ I consider randomly generated password safer than user chosen passwords

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5



- 1. Strongly Disagree
- 2. Disagree
- 3. Neutral
- 4. Agree
- 5. Strongly Agree

✳ I would feel secure using this emoji based password as my facebook password.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5



- 1. Strongly Disagree
- 2. Disagree
- 3. Neutral
- 4. Agree
- 5. Strongly Agree

✳ I would feel secure using this emoji based password as my email password.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5



- 1. Strongly Disagree
- 2. Disagree
- 3. Neutral
- 4. Agree
- 5. Strongly Agree

✳ I would feel secure using this emoji based password as my bank account password.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5



- 1. Strongly Disagree
- 2. Disagree
- 3. Neutral
- 4. Agree
- 5. Strongly Agree

Password Comparison

🌟 I feel safer using this emoji-based password over a randomly generated text-based password.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5



- 1. Strongly Disagree
- 2. Disagree
- 3. Neutral
- 4. Agree
- 5. Strongly Agree

🌟 I feel safer using this randomly generated emoji-based password over a text-based password created by myself. (Safer, not easier to remember)

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5



- 1. Strongly Disagree
- 2. Disagree
- 3. Neutral
- 4. Agree
- 5. Strongly Agree

If my iPhone or Android device let me use a pin of Emoji characters instead of digits or text, I would choose this feature.

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☒ No answer



- 1. Strongly Disagree
- 2. Disagree
- 3. Neutral
- 4. Agree
- 5. Strongly Agree

User Info

What was your tester User ID?

Link to our survey:

<http://hot.soft.carleton.ca/comp3008limesurvey/index.php/694139?lang=en>

Part 5: Testing of the Password Scheme

The participant list can be found by looking in our signed consent form.

Field summary for Name		
What was your tester User ID?		
Answer	Count	Percentage
Answer <input type="button" value="Browse"/>	15	100.00%
No answer	0	0.00%
Not completed or Not displayed	0	0.00%



We had 15 survey participants as required by Robert Biddle's instruction. All 15 logged their user ID's as seen below.

The following page is a brief analysis of results, and our formal conclusions regarding the average participants opinions about our system.

Following the analysis, the next pages of this report will be a display of data based on the survey responses given by our participants.

Please keep in mind that the responses are based on a Likert 1 (Strongly Disagree) to 5 (Strongly Agree) format.

The survey results above have lead us to several conclusions about people's feelings towards our password system, and the concept of emoji's an an authentication key.

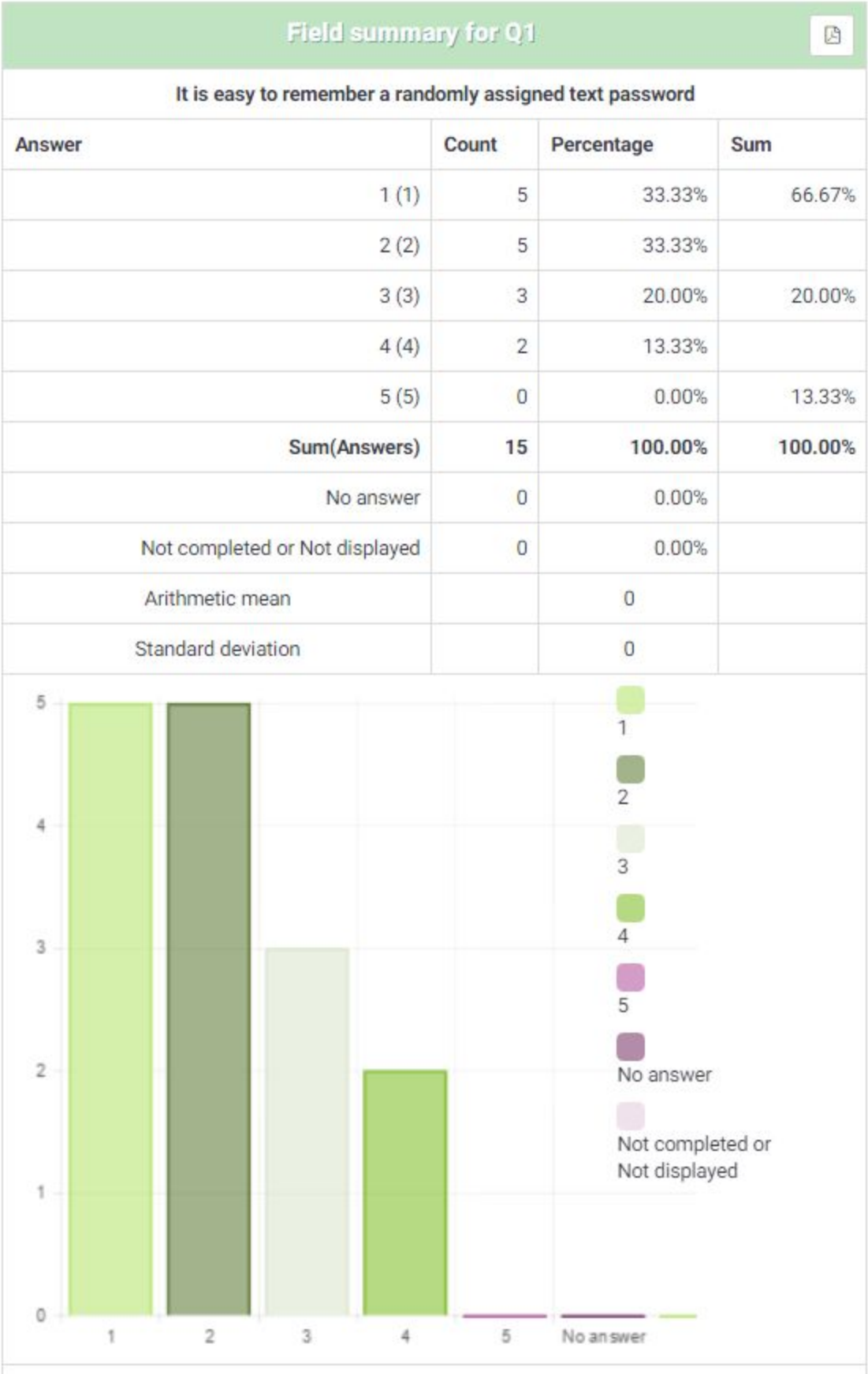
Most people were able to agree that images are lighter on the cognitive load, and easier to remember than text passwords. We also learnt that almost everyone uses emoji's on a daily basis - this is good because people are already familiar with the information they are being presented. However, this can also be a bad thing.

Since users are consistently using these emoji's, they have developed a sense of informality regarding emoji's. I believe this is reflected in the answers users generally gave when asked about using an emoji based password for their social, communication, and banking accounts. Users have a hard time taking the images/emojis as seriously as a text based password, regardless of the bit level security provided.

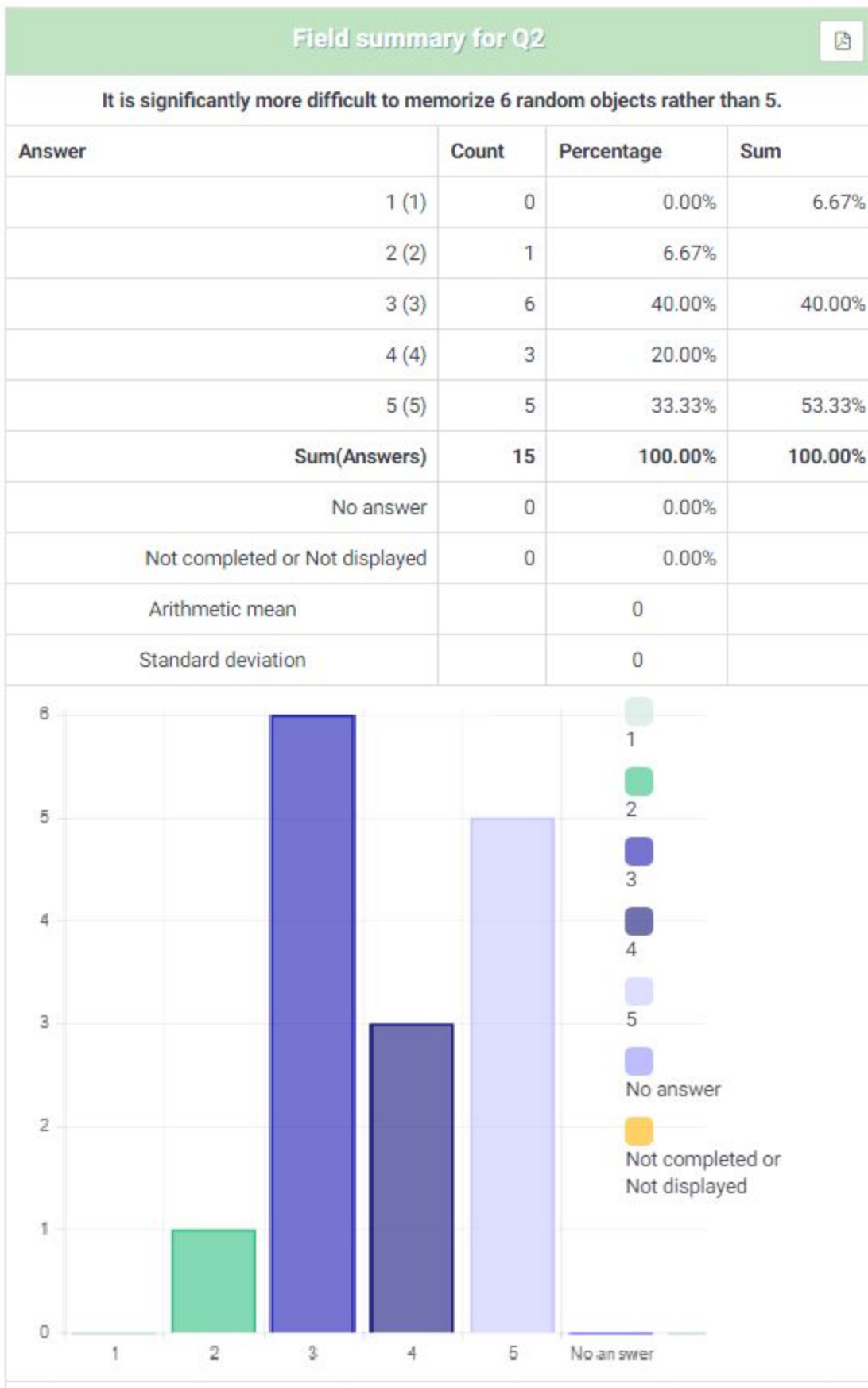
Though users were able to consistently re-enter the pins after being shown their new password for the first time, many of them had trouble at the end of the simulation process when asked to re-enter all three emoji pins. I don't see this as too much of a surprise though - many other teams experienced similar results, and it is concluded that asking someone to remember three separate combinations at once after a very brief time is a very difficult task. After finding this out, I wanted to see how users did entering the three at the end of our simulation, in comparison with other simulations. My findings were that though the participants generally had trouble with ours, they seemed to have less issues than with many other groups. The majority of people were able to remember at least one combination, which suggests that given more time, the user could become accustomed to using all three for their personal accounts.

Overall, we found that our participants enjoyed both the interface, and the method. The aesthetic designs of the emoji's was a contributing factor to the overall enjoyment of our simulation.

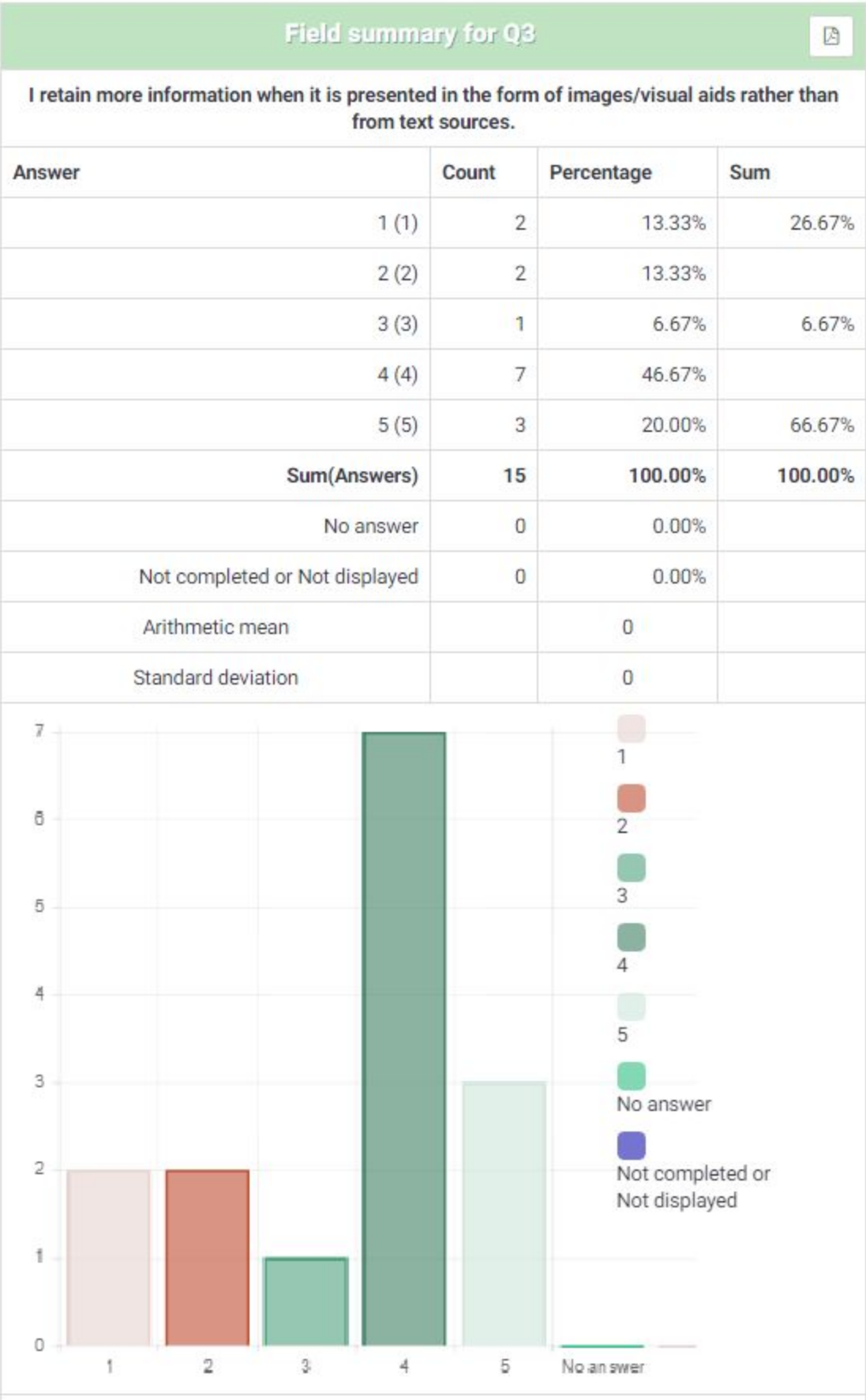
Question 1



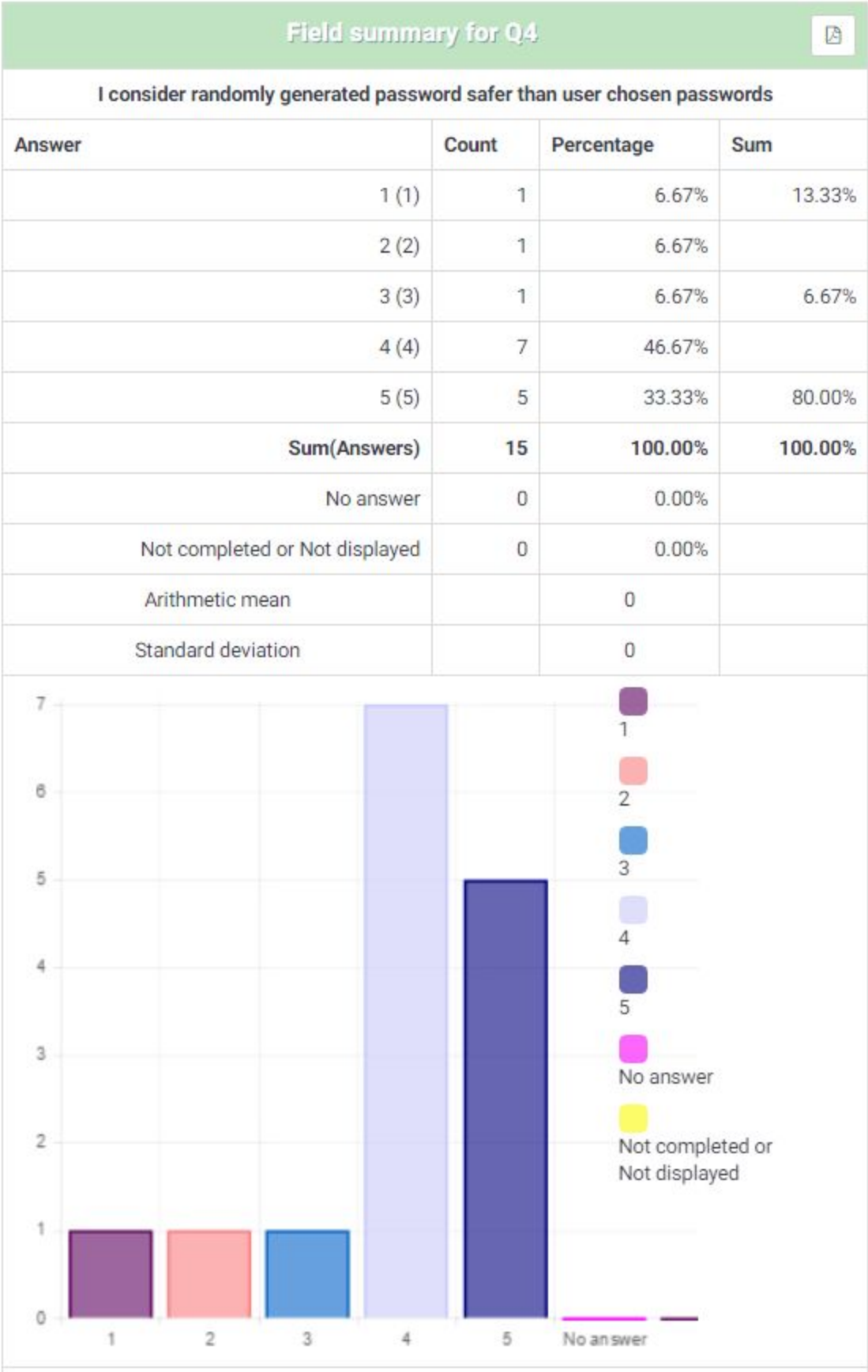
Question 2



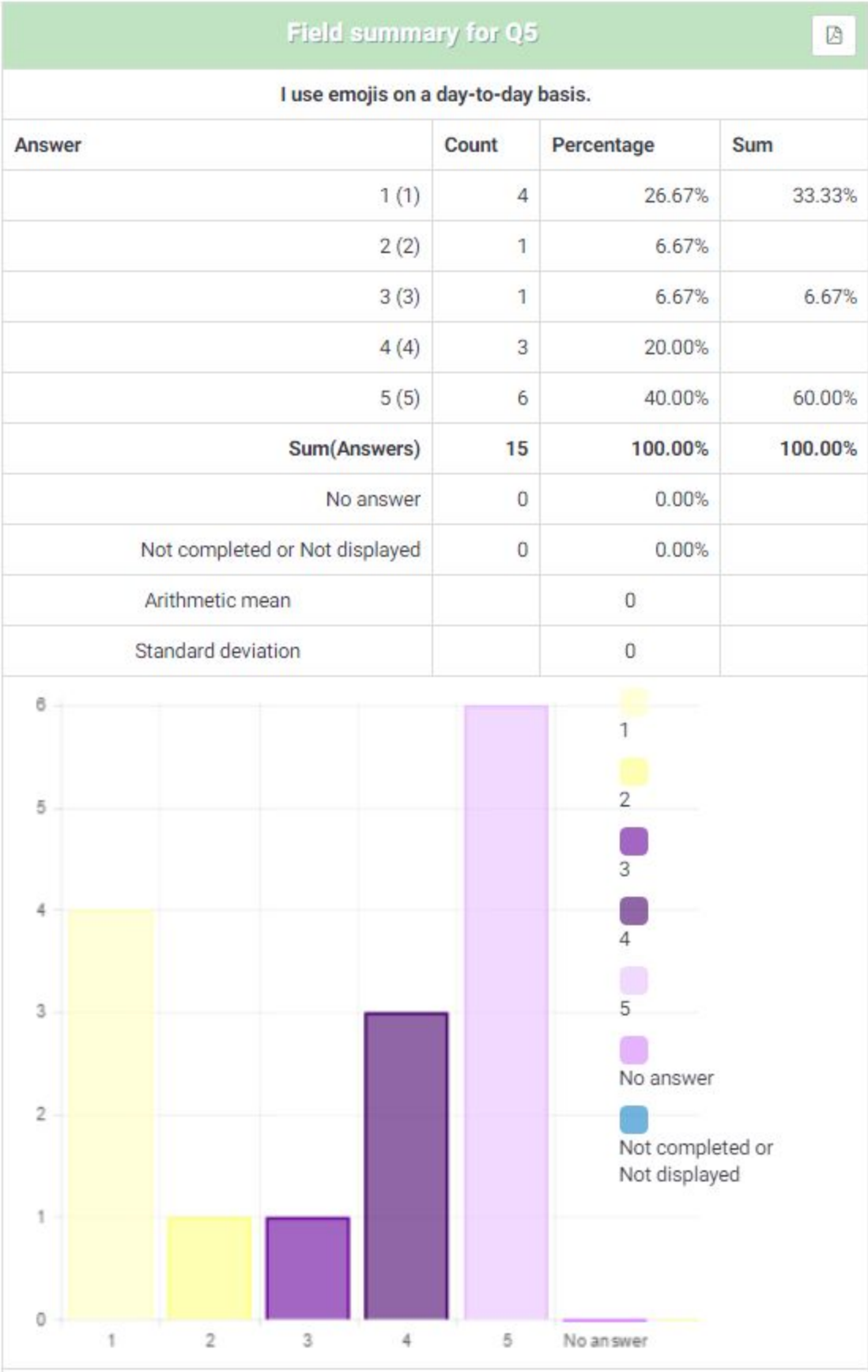
Question 3



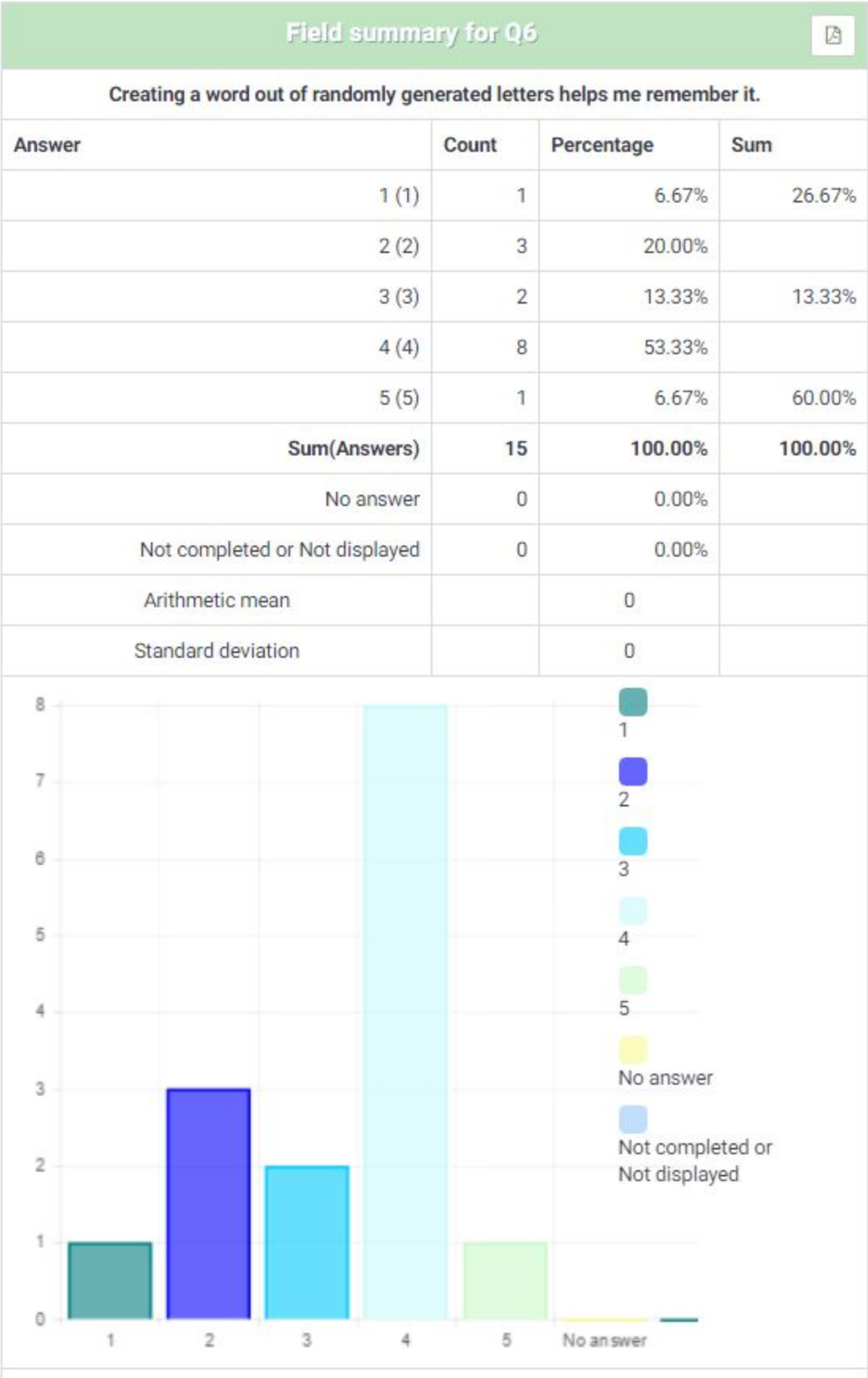
Question 4



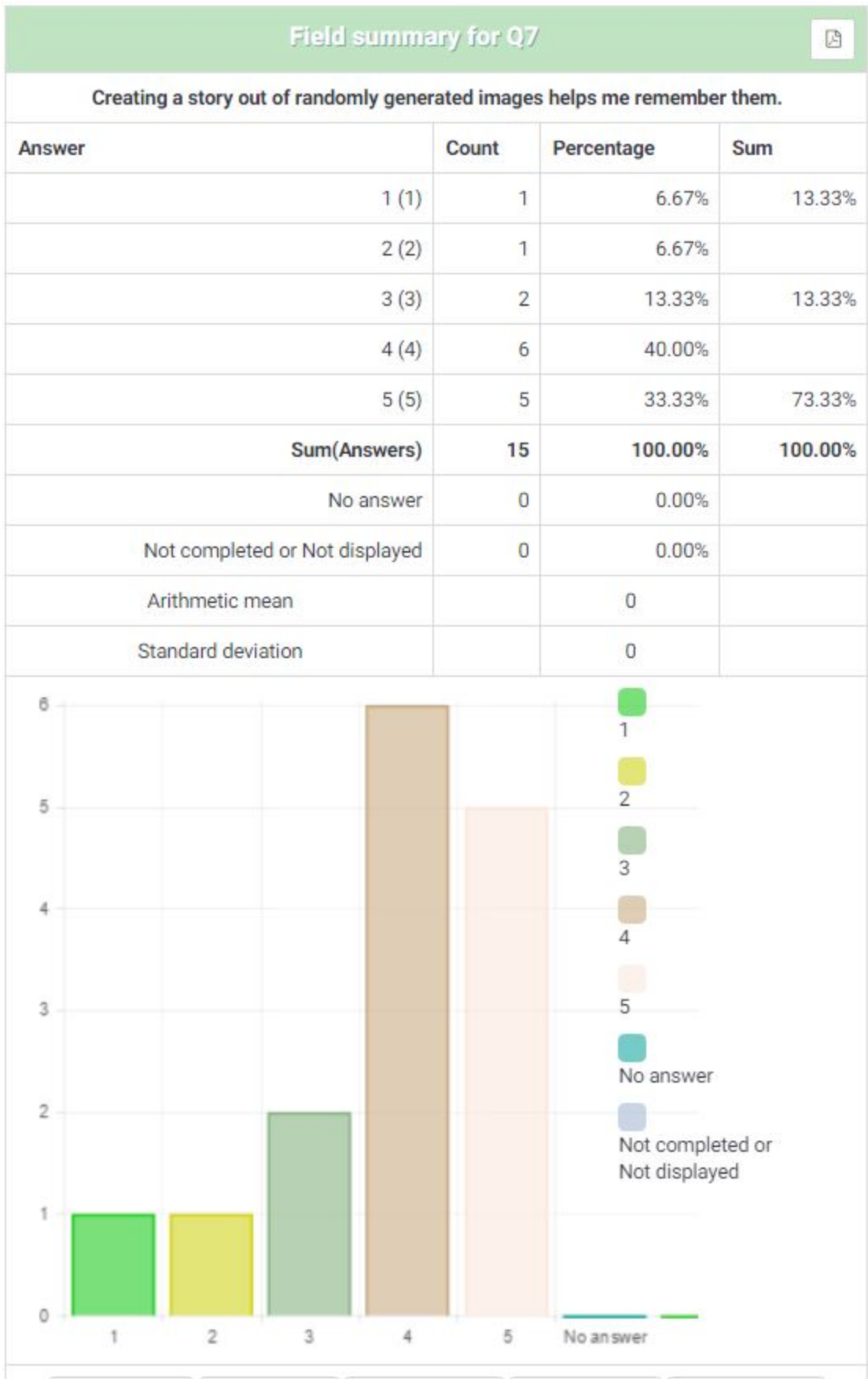
Question 5



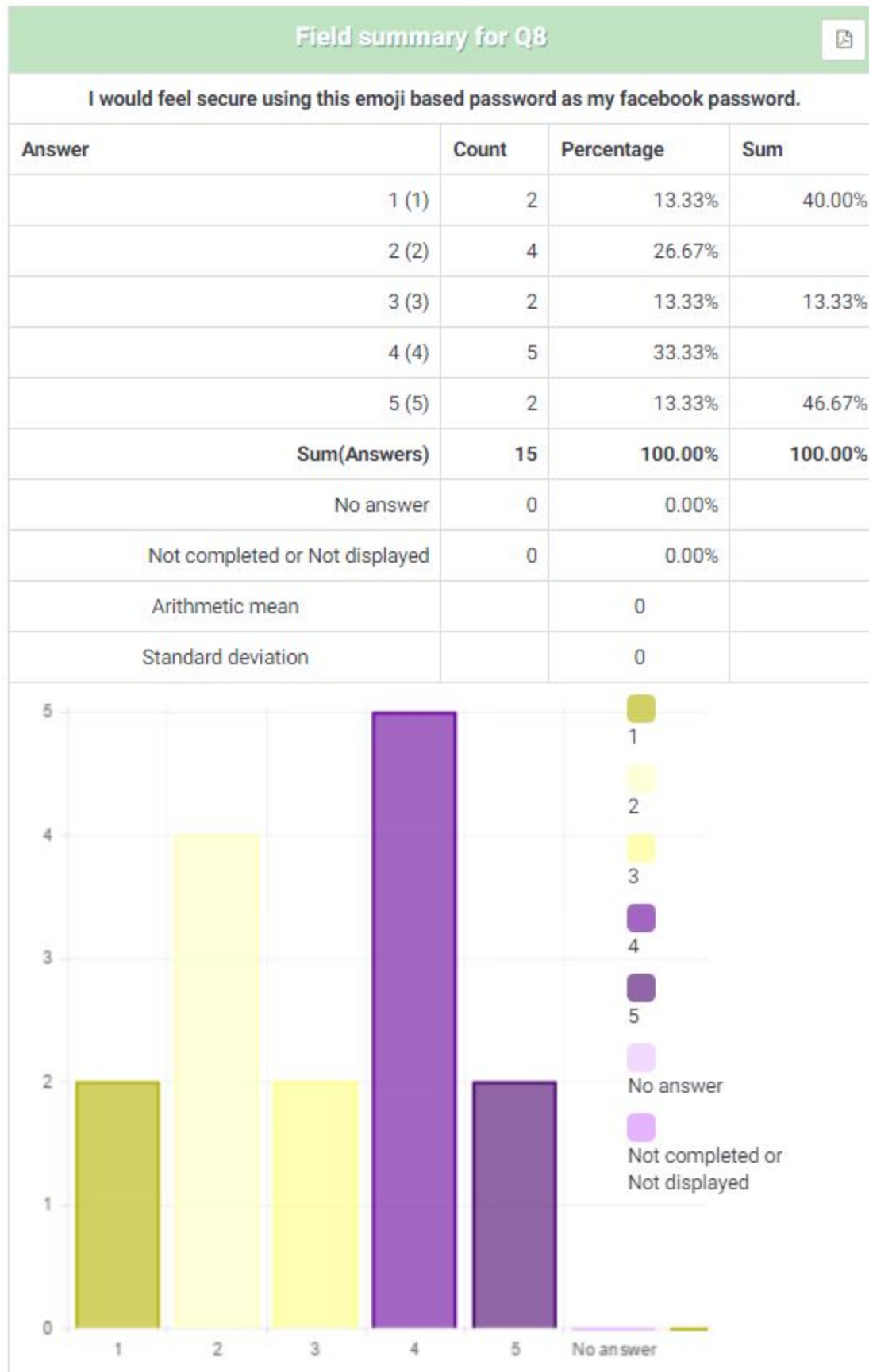
Question 6



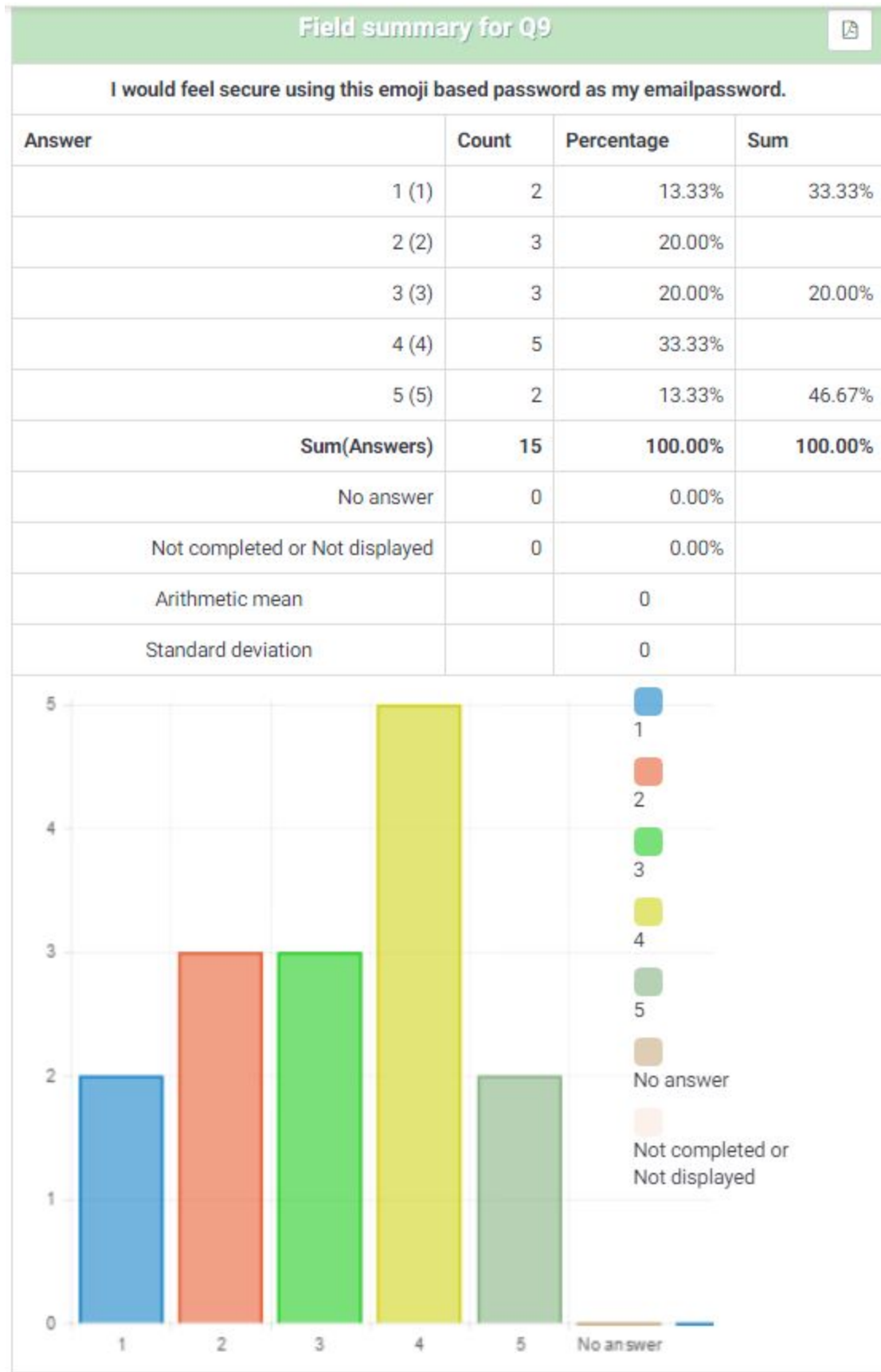
Question 7



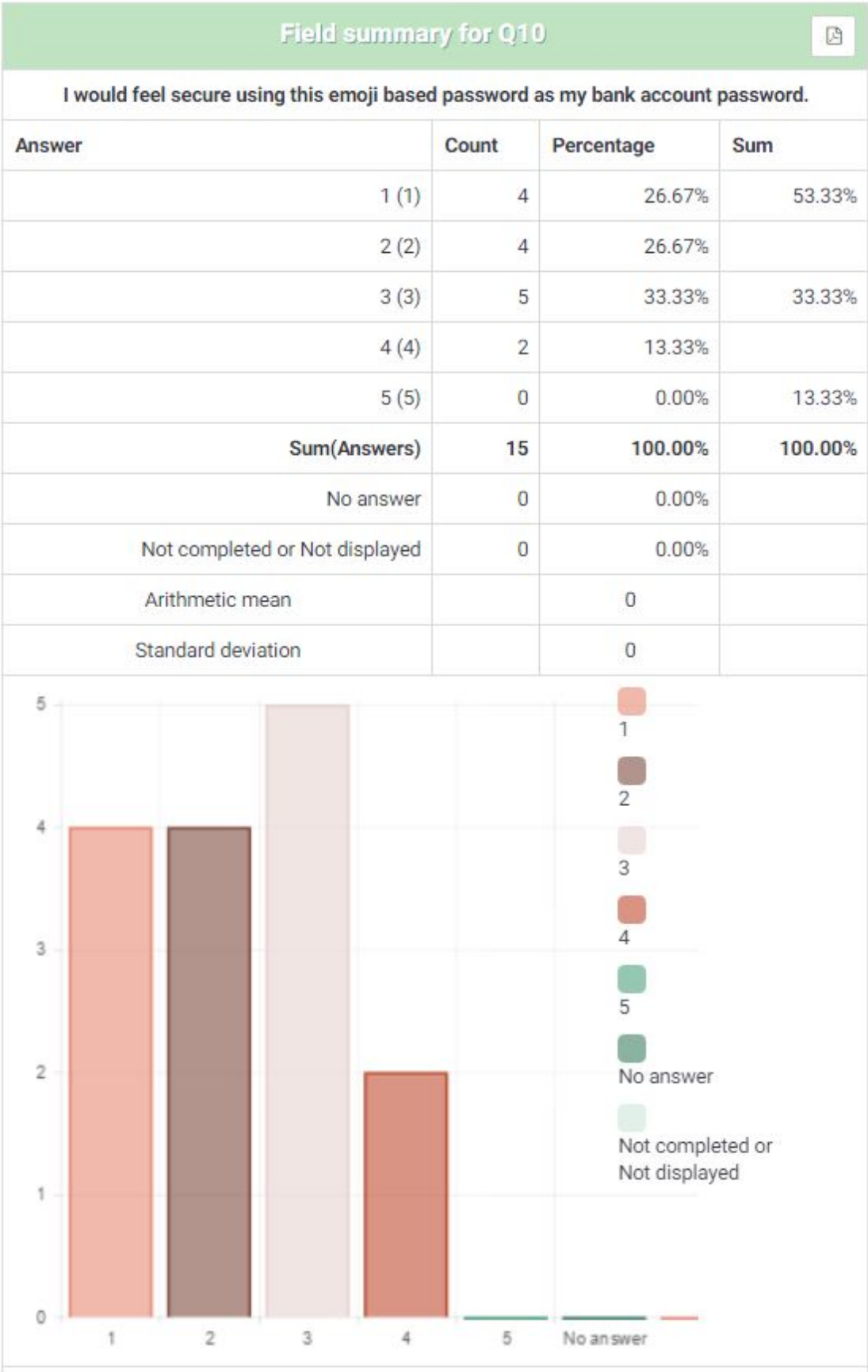
Question 8



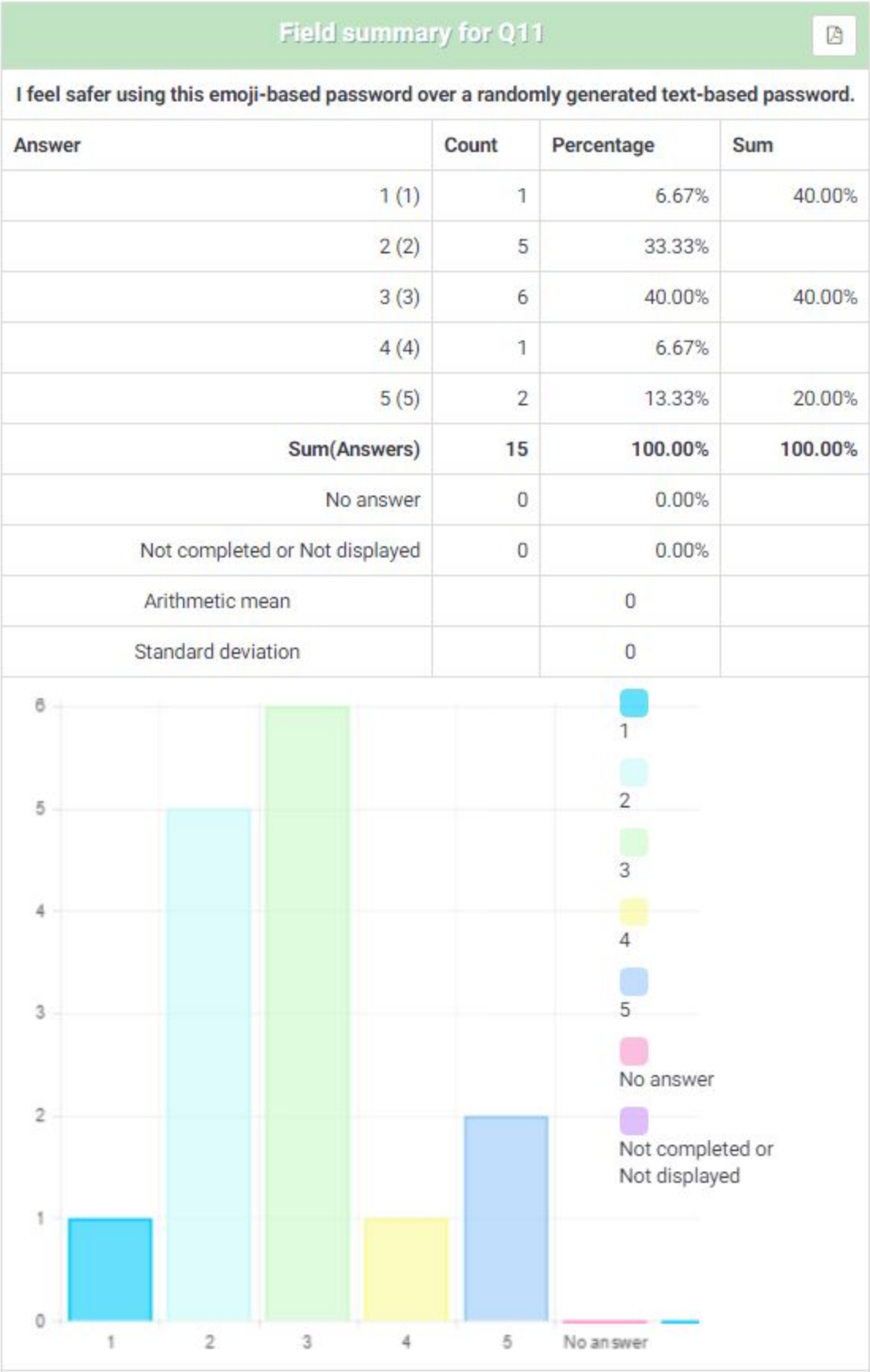
Question 9



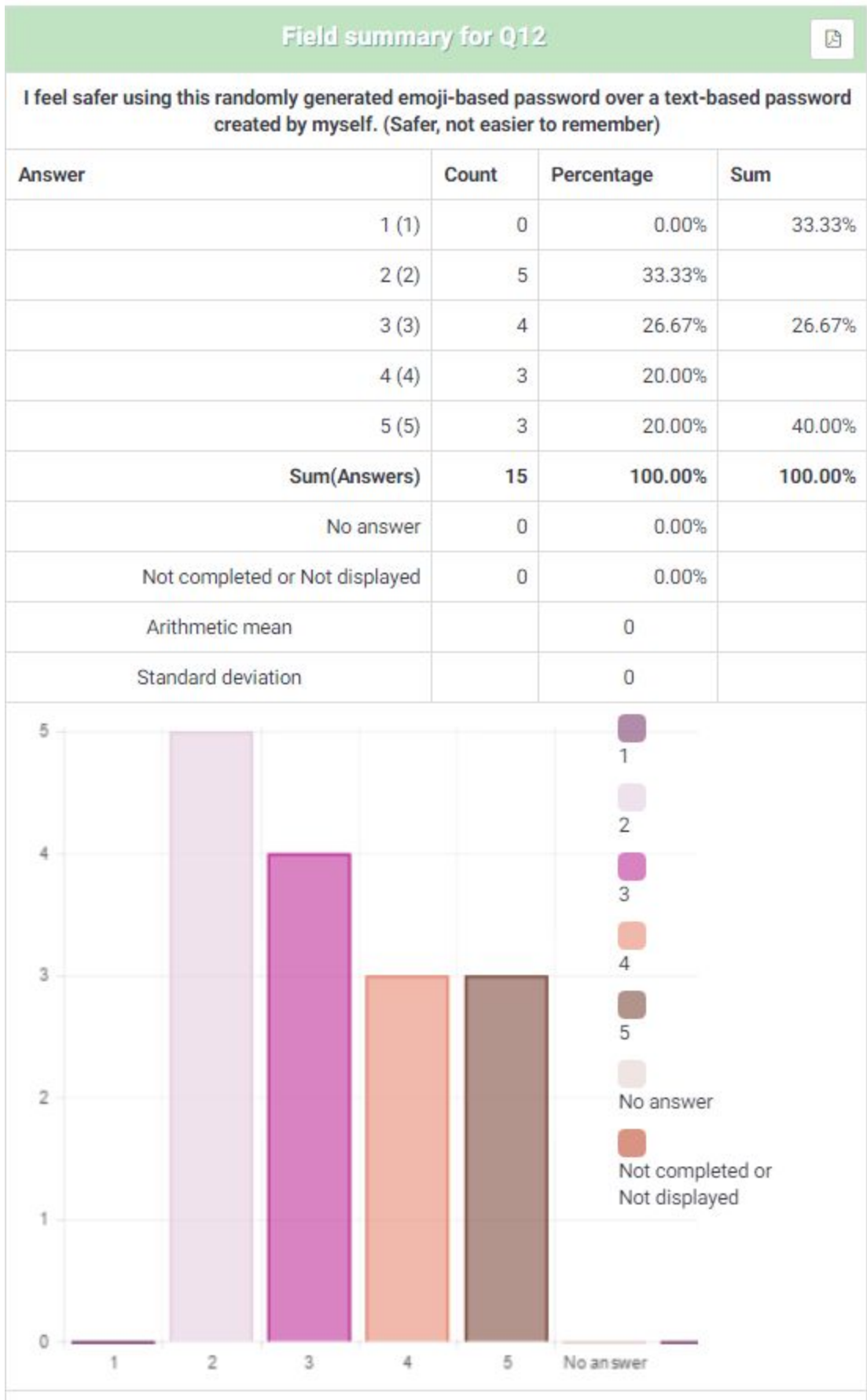
Question 10



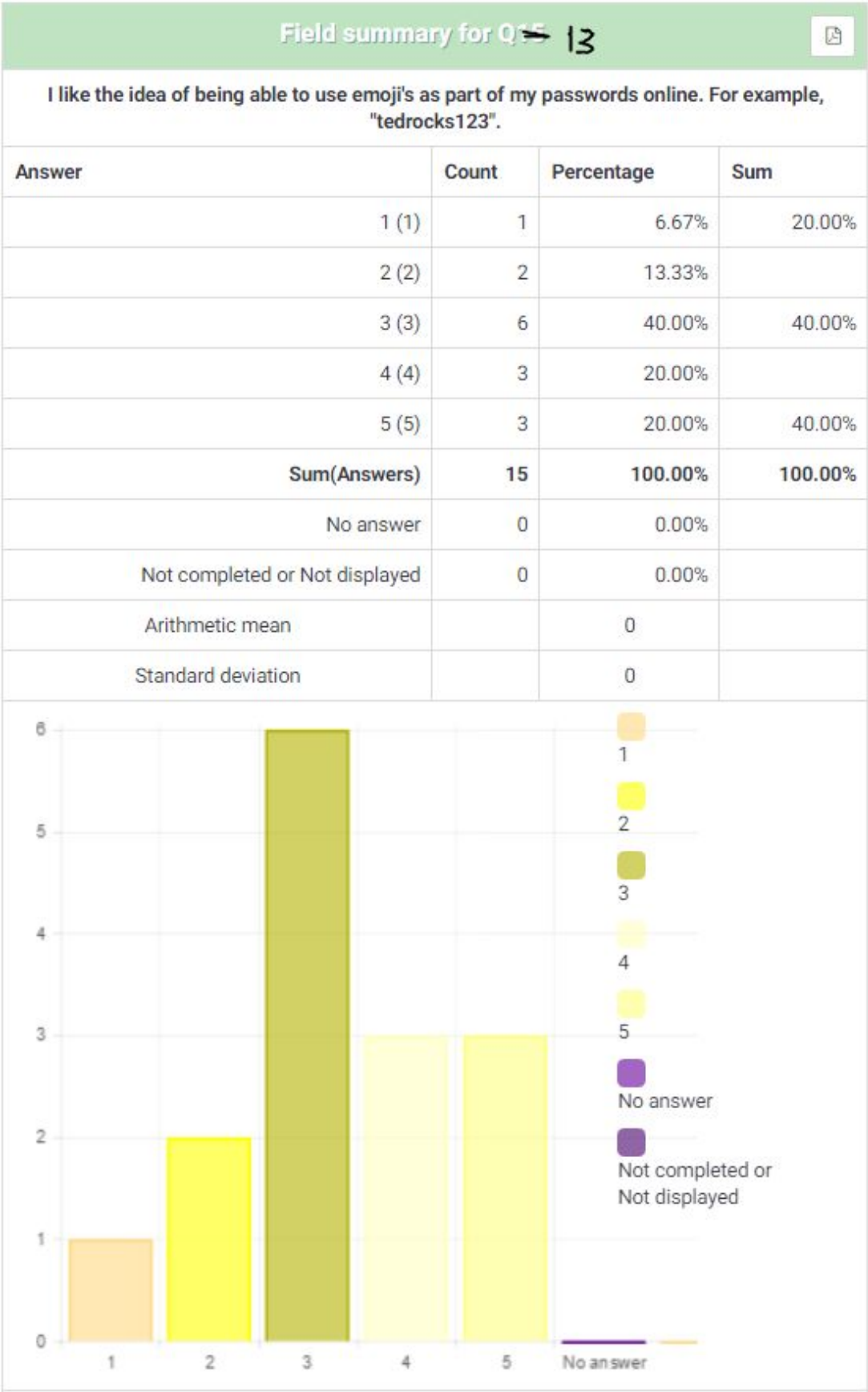
Question 11



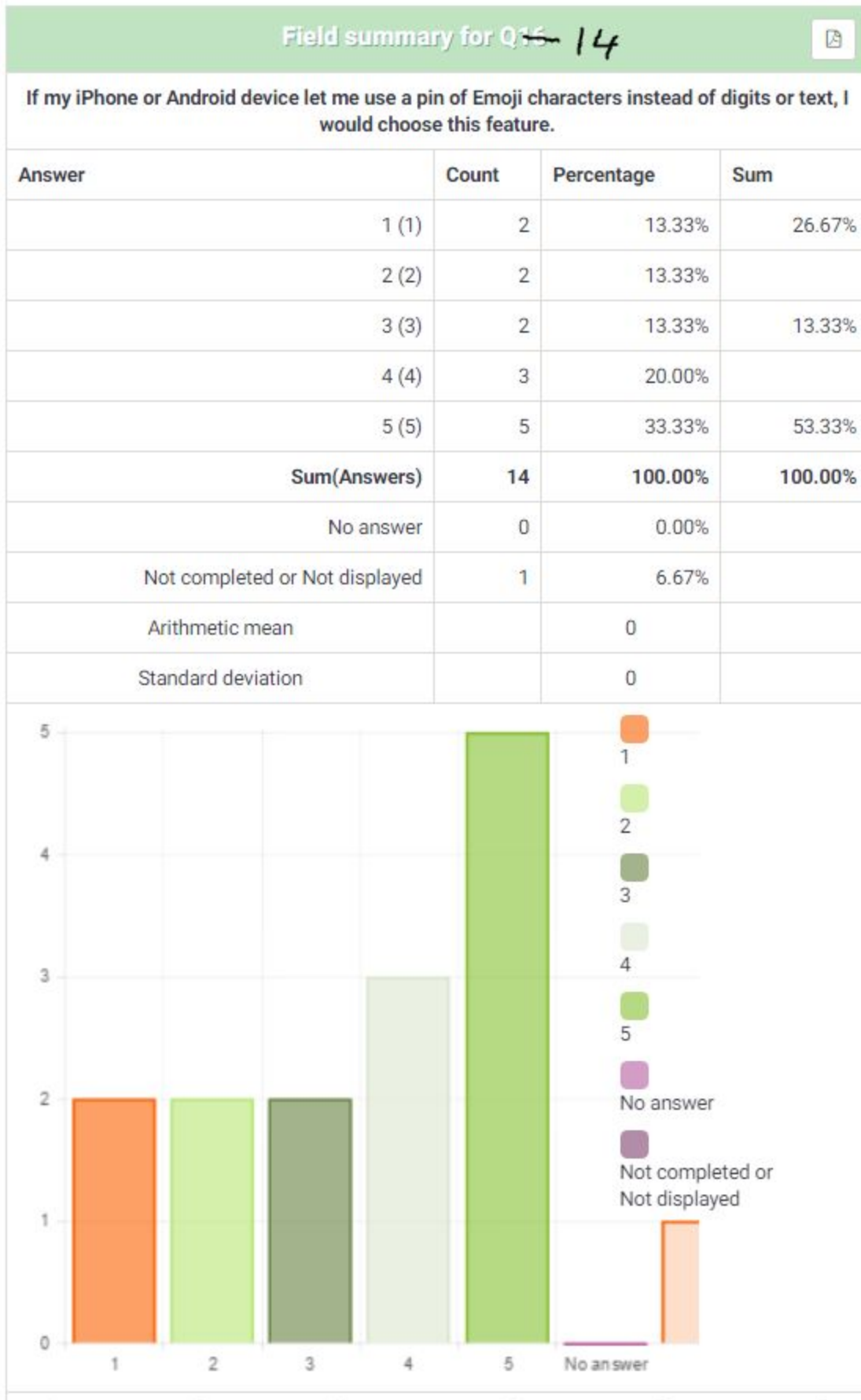
Question 12



Question 13



Question 14



Part 6: Analysis of Excel Data for Emoji Password System

Mean Median and Standard Deviation of Text28_log
Mean of users events that are logins 16.88889 Mean of successful logins per user 14.66667 Mean of failed logins per user 4 Median of users events that are logins 16 Median of successful logins per user 15 Median of failed logins per user 2 Standard deviation of logins per user 4.226241 Standard deviation of successful logins per user 1 Standard deviation of failed logins per user 4.636809
Mean Median and Standard Deviation of Emoji Password
Mean of users events that are logins 6 Mean of successful logins per user 4.75 Mean of failed logins per user 1.909091 Median of users events that are logins 6 Median of successful logins per user 5 Median of failed logins per user 1 Standard deviation of logins per user 1.779513 Standard deviation of successful logins per user 1.864745 Standard deviation of failed logins per user 1.221028

Comparison graphs and R source code available here:

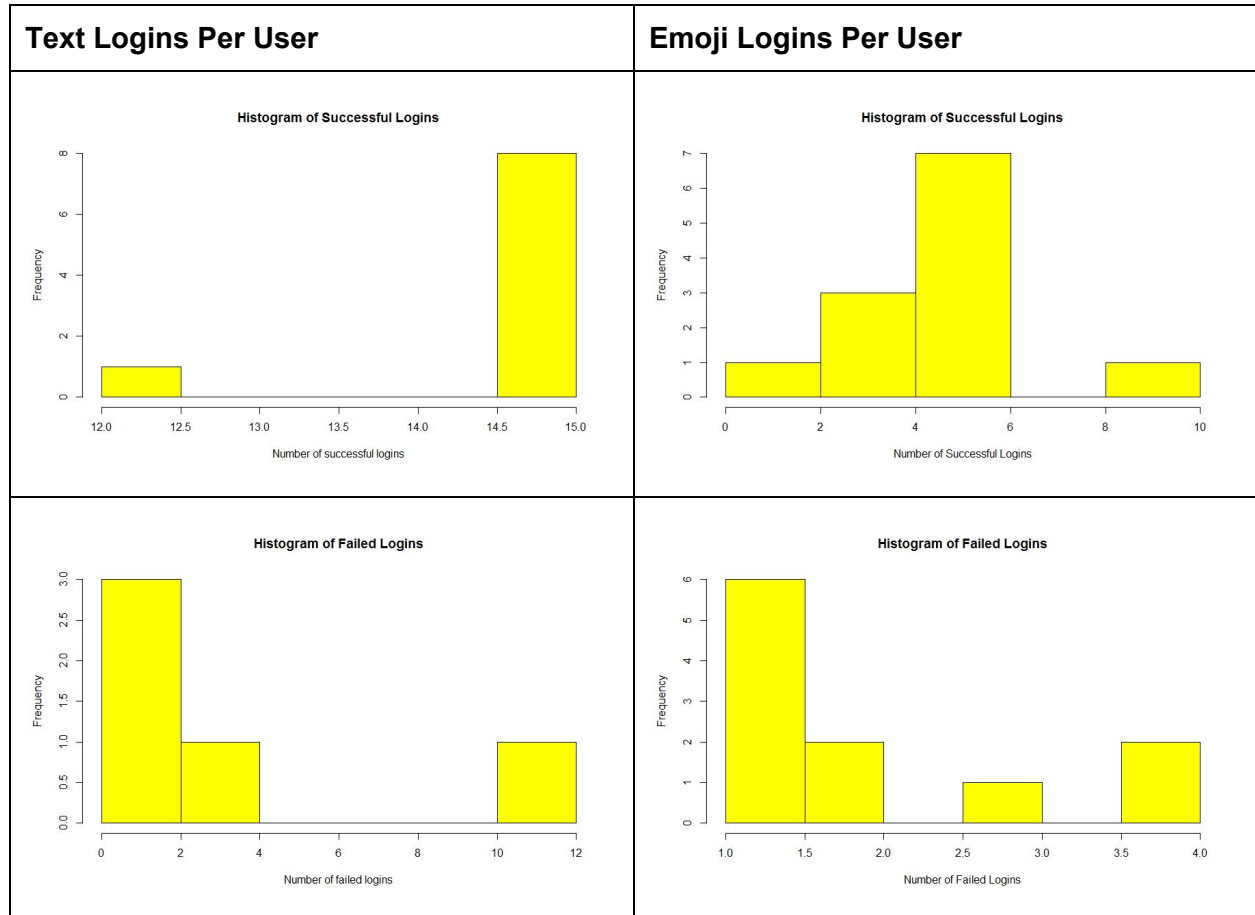
<https://drive.google.com/open?id=0B41LGpA8OtX7anIFdWtpaWQ1VFk>

Looking at the mean, median and standard deviation, we can make several quick conclusions about our emoji password scheme in comparison to the Text scheme.

We can see that the ratio of successful logins per user to the unsuccessful logins per user in the Text scheme is approximately 3.75. We can also see that the ratio for the emoji scheme is approximately 2.5 successful logins to every failed login. This shows that on average the text based password was easier for users to remember than the emoji password.

We can also see that the standard deviation of failed logins per user is MUCH higher for the Text based password. This tells us that while most users have more success using the text based password scheme, some users have a much more difficult time remembering randomly

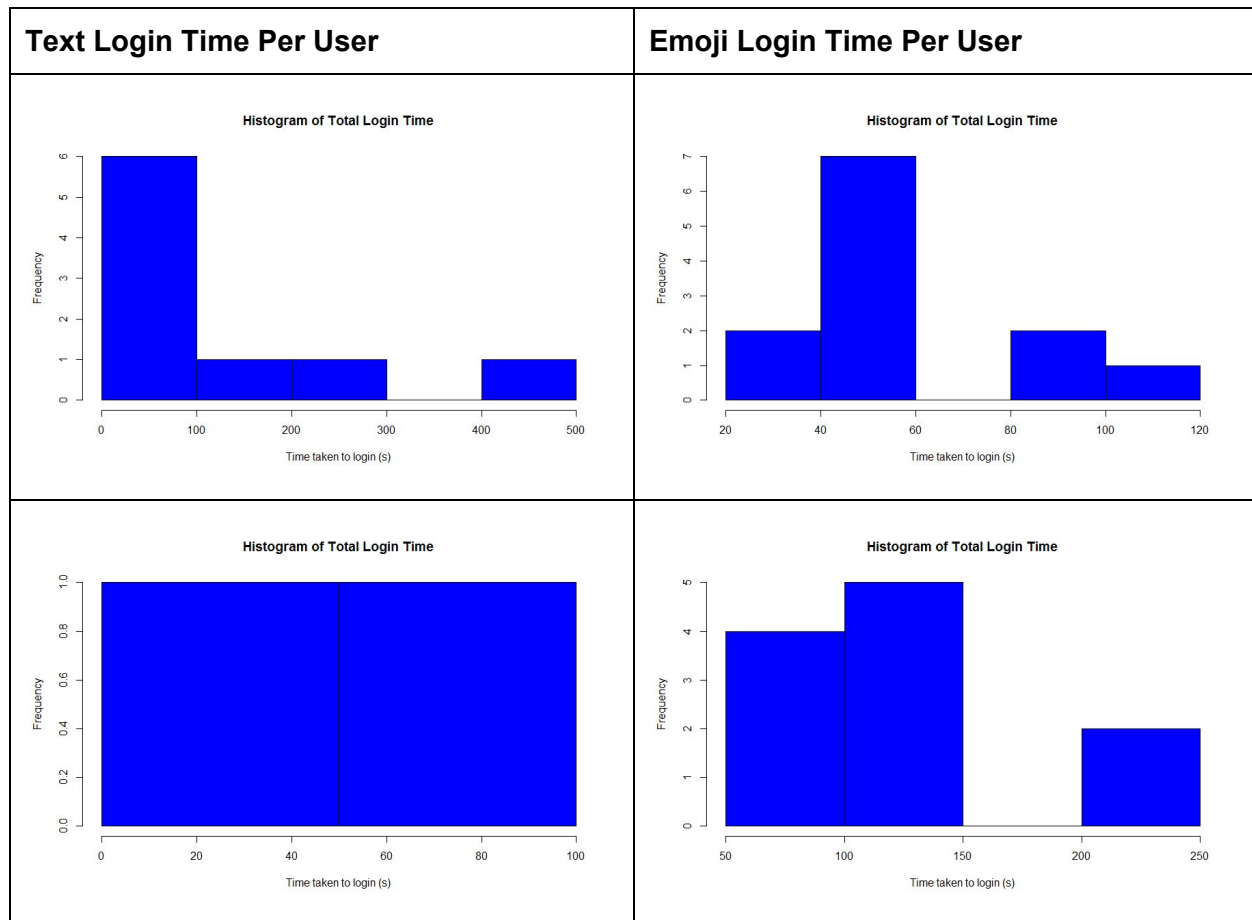
generated letter passwords. The emoji password proves to be more consistent, with all of our users having a similar success-failure rate.



Comparing the histograms of the number of successful and failed logins per user, we can see a couple things which help to prove the points made above about the mean and standard deviation of the password schemes.

1. We can quickly see that the Text scheme has almost all of its users with 15 successful logins, and 0-2 failed logins, whereas the emoji scheme has most of its users with 4-6 successful logins, and 1-2 failed logins. This reinforces what we could see about the mean of the text scheme having a much better ratio of success to failure than the emoji scheme.
2. We can also quickly see the outliers of each scheme. If we look at the text scheme, we can see that 1 user had 12 successful logins, a 20% less success rate. We can also see that that same user (most likely) had MANY more failed logins, 10-12 compared to the usual 0-2. This change in usability of the password scheme is much more drastic than the emoji password scheme. In our scheme there was slightly more variance in the number of successful logins, however there was much less variance in the number of

failed logins, with the highest number of failed logins at only 4, much less than the text scheme.



Comparing the average time taken for users to login using both password schemes, we can see some differences between the two schemes when a user is successful and when a user is unsuccessful.

1. Using the text password scheme, the majority of the users took less than 100 seconds, however some users seemed to be very stumped and took all the way up to 500 seconds before they successfully logged in. When this is compared to the average login time of the emoji scheme, we can see a large difference. The users of the emoji scheme were much quicker in entering a password when they felt they remembered it.
2. In the case of entering unsuccessful passwords on the other hand, users of the text scheme took much less time than users of the emoji password. In our opinion, this is most likely due to the image recognition of the emoji password scheme. When users are entering their password, they are more likely to see something does not look right, compared to entering a text password, where letters don't have that same image recognition feeling to them.

Conclusive results:

The results of our tests show that unfortunately our password scheme is not quite as user friendly as the Text28 scheme. This is shown through our data analysis of the log files of both schemes. However, while they do show that the Text scheme is better, there is still some room for discussion. Our group believes that the reason the text scheme performed better is because not only were the users which we had test our system do numerous other password scheme tests during the class of April 3rd, but most importantly every users has had at least a decade of experience using randomly generated text passwords. This will help them remember them better because it is something their brain has become accustomed to. We believe that users had been using emoji passwords always, we would see the reverse. To this extent, we believe that our password scheme is even, if not better than the Text28 scheme. Additionally, it is a safer password scheme as it has even more possible combinations than the Text28 scheme.

Sources

- [1] https://en.wikipedia.org/wiki/Picture_superiority_effect
- [2] https://en.wikipedia.org/wiki/The_Magical_Number_Seven,_Plus_or_Minus_Two
- [3] https://en.wikipedia.org/wiki/Method_of_loci