A Comparison of Three Approaches to Assist Users in Memorizing System-Assigned Passwords

Michael Clark
Brigham Young University
clark.michael.c@gmail.com

Scott Ruoti
The University of Tennessee
ruoti@utk.edu

Michael Mendoza
Imperial College London
askmichaelmendoza@gmail.com

Kent Seamons
Brigham Young University
seamons@cs.byu.edu

***Abstract*—Users struggle to select strong passwords. System-assigned passwords address this problem, but they can be difficult for users to memorize. While password managers can help store system-assigned passwords, there will always be passwords that a user needs to memorize, such as their password manager's master password. As such, there is a critical need for research into helping users memorize system-assigned passwords. In this work, we compare three different designs for password memorization aids inspired by the method of loci or memory palace. Design One displays a two-dimensional scene with objects placed inside it in arbitrary (and randomized) positions, with Design Two fixing the objects' position within the scene, and Design Three displays the scene using a navigable, three-dimensional representation. In an A-B study of these designs, we find that, surprisingly, there is no statistically significant difference between the memorability of these three designs, nor that of assigning users a passphrase to memorize, which we used as the control in this study. However, we find that when perfect recall failed, our designs helped users remember a greater portion of the encoded system-assigned password than did a passphrase, a property we refer to as *durability*. Our results indicate that there could be room for memorization aids that incorporate fuzzy or error-correcting authentication. Similarly, our results suggest that simple (i.e., cheap to develop) designs of this nature may be just as effective as more complicated, high-fidelity (i.e., expensive to develop) designs.**

I. Introduction

Passwords are ubiquitous, but their limitations are legion [12], [37], [30]. In particular, users struggle to select strong passwords [15]. This is particularly problematic when using those passwords to secure high-value assets, such as a cryptocurrency wallet [38].

To address this problem, users can instead use random, system-assigned passwords. The key challenge with adopting system-assigned passwords is the ability of users to memorize those passwords. While password managers can help users store many system-assigned passwords [28], there will always be a few passwords that users have to memorize, such as the master password for the password manager, the password for email accounts they use for account recovery, or the password for highly-sensitive materials such as a cryptocurrency wallet [20]. Failing to memorize these passwords puts users at significant risk, as demonstrated by recent events where there is strong evidence that the LastPass breach has led to the theft of users' cryptocurrency passphrases stored in their password manager [20].

There are many promising approaches to help users memorize system-assigned passwords, one of which is the *method of loci* [18], [22], also known as a *memory palace*. The method of loci works by envisioning a fixed journey through a fixed environment and placing objects to remember at fixed positions (or *loci*) in that environment. Each fixed position acts as a memory cue for whatever was stored there. Virtual environments can work as memory palaces [22] and have been used to remember passwords [18], [13] with promising results.

Prior research has found that systems incorporating memory palaces helped assist users in memorizing system-assigned passwords [18], [13]. However, these systems also incorporated many other design features (i.e., confounding factors), so it remains unclear to what extent memory palaces and how they are presented to users impact the memorability of system-assigned passwords [13].

In this paper, we start filling this knowledge gap by comparing three approaches based on memory palaces. First, we investigate the performance of a two-dimensional scene with objects placed inside it in arbitrary (and randomized) positions. Second, we explore the impact of fixing an object's position within the scene. Third, we analyze the effect of transitioning to a three-dimensional, navigable representation of the scene. By comparing the performance of these three designs, we can investigate what aspect of the design most impacts memorability. *Critically, the goal of this research is not the creation of a best-in-class system but rather a scientific comparison of these three design approaches.* This distinction is important as attempting to create a best-in-class system would nearly certainly introduce confounding factors into our research, as was the case in past research [11], [18], [13].[1]

[1]Our study design ensures fidelity and entropy are consistent between treatments, and tests for the influence of 2D vs 3D design and location memory. In some prior work with comparisons, fidelity was not as consistently controlled between treatments, which leaves open the possibility that fidelity differences may have influenced the results. Additionally, experiments intended to demonstrate a best-in-class system typically integrate memory techniques along with the presentation of the system, leaving open the question of whether the technique or the system design had more influence on the observed memorability.

Symposium on Usable Security and Privacy (USEC) 2024
26 February 2024, San Diego, CA, USA
ISBN 979-8-9894372-5-2
https://dx.doi.org/10.14722/usec.2024.23xxx
www.ndss-symposium.org, https://www.usablesecurity.net/USEC/

To compare these three approaches, we conducted a 300-participant Amazon Mechanical Turk workers study. In this study, participants learned a system-assigned password using one of these three approaches. They then had their memory tested in follow-up sessions seven [5], [18], [13] and thirty [27] days later. As a baseline for comparison, we also had each participant memorize a system-assigned passphrase.

Surprisingly, our results show no meaningful difference in memorability between the passphrase and our three approaches. This suggests that passphrases may be just as effective at helping users memorize system-assigned passwords as these designs. It also suggests no significant difference between fixed or random positioning of elements within a scene. Similarly, there is no difference between three-dimensional, interactive scenes and two-dimensional, static images.

While users incorrectly entered their passphrase and *3D path* (the sequence of scenes, objects, and object states) at roughly the same proportions (i.e., failed the memory check), we did notice a difference in the portion of the passphrase or path that was correct. That is, when participants failed to remember the entire password, they were, on average, able to remember a significantly larger portion of the 3D path than the passphrase. We label this effect as *durability*. While durability is less desirable than memorability, it is nonetheless a helpful property. In particular, memory techniques with high durability could be used to memorize system-assigned passwords if error-correcting codes were included in the process of encoding the password. We believe this is an intriguing area for future research into system-assigned password memorability.

II. Background and Related Work

In this section, we review relevant background from research into human memory. We then describe related work for memorizing system-assigned passwords. Finally, as the designs we use incorporate graphical cues, we discuss graphical authentication systems.

A. Memory

Memory tasks test recall, such as entering a password at a prompt; recognition, such as selecting familiar faces from a list (as in Passfaces [8]); and implicit memory, such as relearning information since forgotten [39]. Recognition is generally considered easier than recall for most memory tasks [36]. Testing recall is typical in "something you know" authentication, though graphical authentication systems often instead test recognition, and at least one research system tests implicit memory [4].

It is possible for humans to recall extremely long sequences of unpredictable data, such as more than 70,000 digits of π [31], or the sequence of cards in a shuffled deck of 52 playing cards ($\lfloor \log_2 \prod_{i=1}^{52} i \rfloor = 225$ bits of entropy). Mnemonic techniques are trainable, and Dresler et al. show that even a short period of training results in similar brain connections to memory champions under fMRI [14].

Most participants who place highly in the World Memory Championships use the "method of loci", by imagining a familiar environment and placing variable items to remember at fixed locations (loci) in the environment [24]. Legge et al. found the method of loci was equally effective when providing a constructed virtual environment for naïve participants [22]. Some systems inspired by the method of loci use a variable path instead (such as [11]); it is unclear if such systems enjoy the same memory benefits.

Landauer shows that time spent rehearsing influences recall, and provides a general estimate of approximately 2 bits able to be recalled per second focused on memorizing [21]. Hyde and Jenkins [19] show that recall does not improve when the participant expects a follow-up.

Other relevant factors in memorability include bizarre imagery and chunking. McDaniel et al. show [25] that bizarre imagery can aid memory when distinctive both in context and individual experience, particularly when mixed with non-bizarre items. Miller shows [26] that memory appears to operate best on "chunks" of information, and that 7 ± 2 may be a reasonable upper bound on the number of chunks.

Spaced repetition, which involves repeatedly memorizing information with delays in between, has been shown effective in password memorization by Bonneau and Schechter [6] and Blocki et al. [3]. Comparing the link method — telling a story that ties the next item with the previous item — with the method of loci for memorizing passwords, Haque et al. found the method of loci to be superior [18]. Das et al. found superior memorability for a system inspired by the method of loci when implemented as a first-person 3D view vs. a 3rd-person 3D view or a 2D top-down view [11]; however, there were confounding factors in the representation fidelity between those conditions.

B. System-Assigned Password Memorization

Prior experimental research has shown that when supported by various memory techniques, most users can recall a 56-bit system-assigned password in later follow-up sessions. Using spaced repetition, 61 of 104 participants (59%) remembered their password after around 17 days [6]. Using video training and the method of loci, 15 of 28 participants (58%) remembered their password after 7 days, and with the addition of a memory game to reinforce the training, 133 of 164 participants (81%) recalled their password after around 17 days [13]. Unfortunately, each of these systems incorporated many design features (i.e., confounding factors), leaving it unclear precisely what part of the system led to the high recall rates [13]. *In this paper, we seek to address this problem by completing a controlled A-B comparison of three different approaches for method of loci.*

Without support from memory techniques, recall rates tend to be somewhat lower. A comparison of three system-assigned password generators at 30.8, 35.7, and 38.8 bits of entropy found that only four of nineteen participants correctly recalled them two weeks later [23]. Another study of three system-assigned password generators at 47.2, 47.5, and 49.8 bits of entropy found only four of forty participants recalled any password one-week later [9].

A large study of eleven random password generators at from 29.3 to 39.2 bits of entropy with 1476 participants found that 49% of the 410 participants who didn't write down their password were still able to recall it three days later [34]. This study tested recall after a distractor survey and provided the password to fewer than 10% of participants who did not recall

it after five attempts. This may have aided recall, acting as a single instance of spaced repetition.

C. Graphical Authentication

Biddle et al. classify graphical authenticators into three categories [2]: recall-based, where the user must reproduce the same drawing used during registration; recognition-based, where the user recognizes some subset of images learned during registration from a displayed set including distractor images; and cued-recall systems, where the user must select specific locations within a larger scene. Our designs, Psychopass [10], and the methods of Haque et al. [18] and Doolani et al. [13] are relevant examples of cued recall graphical authentication systems, each incorporating a memory journey.

Many other graphical authentication methods are designed to secure against observation ("shoulder-surfing") attacks. These systems have very different design goals from the systems we tested, and are in many cases subject to brute-force guessing attacks due to their design [40]. Because these systems have very different design goals, they are not relevant to this research.

Alsulaiman and El Saddik propose a 3D password where users move around in a virtual environment, and the sequence of actions they perform chosen from the set of possible actions represents the password [1]. SeedQuest, the system we fork to create our designs, represents a high-fidelity commercial prototype of this previously primarily theoretical paper. From our discussions with SeedQuest's lead developer, we believe SeedQuest represents an independent rediscovery of the same principles. George et al. implement a similar system in a virtual reality environment [16].

III. System Design

In this paper, we wanted to understand how different design approaches impact the memorability of system-assigned passwords. To this end, we forked SeedQuest, a pre-existing commercial open-source tool. SeedQuest is designed to help users memorize a system-assigned password (i.e., a *seed*) by representing that password as a sequence of objects displayed within a scene, similar to the idea of a memory palace. In total, we created three variations of SeedQuest, each of which allowed us to test a different approach:

1) Our first approach involved showing a two-dimensional scene that has objects placed inside it in arbitrary (and randomized) positions.
2) Our next approach involved maintaining the two-dimensional scene and objects but fixing the objects at specific positions within that scene.
3) Our final approach involved a transition to a three-dimensional, navigable representation of the scene. This design is indistinguishable from SeedQuest's public implementation other than the shorter secret and sequence lengths.

Figure 1 shows an example of these three different approaches. Critically, we used screenshots of the 3D assets in the 2D designs to avoid introducing confounding factors (such as fidelity differences) that might obscure to what extent the differences between the methods impact memorability, as has been a problem in prior work [11], [18], [13]. As this meant that we avoided adding unique features that would have improved the usability or memorability of specific designs, we do not expect our systems to represent best-in-class approaches for memorizing passwords. Instead, our focus was on the precise and scientific comparison of our three approaches based on memory palaces.

In the remainder of this section, we describe SeedQuest in greater depth. We also give more details about each of our three designs.

A. SeedQuest

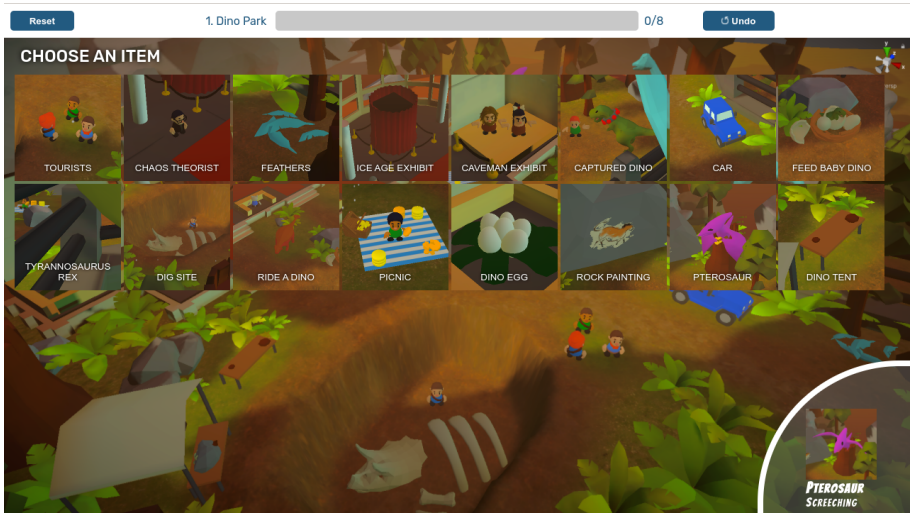
We used SeedQuest to build our three designs for two reasons. *First*, as an open-source project, SeedQuest was readily available and we could make changes to the code. Complimenting that availability, Consensys, the organization sponsoring SeedQuest's development, was helpful in answering our questions about the tool and provided some assistance in getting our code running. However, Consensys did not fund this research, nor did they have any input on the formulation of research questions, selection of system designs, creation of study instruments, execution of the study, analysis of the results, or writing of the paper. *Second*, SeedQuest is a relatively high-fidelity prototype, close to release quality. Such a system would be prohibitively expensive to develop in a lab setting and provided a unique opportunity to contrast the benefit of that fidelity against simpler and cheaper lab prototypes.

SeedQuest itself is heavily inspired by the concept of a memory palace. SeedQuest utilizes 16 distinct scenes, within which 16 scene-appropriate, interactable items are displayed to the user (see Figure 1). Each of these interactable items can be in one of 4 different states—for example, a lamp could be on or off, turned on its side or upside down. Many items and states employ bizarre imagery to make them more memorable [25]. The sequence in which scenes, objects, and states are selected is used to encode a system-assigned password.

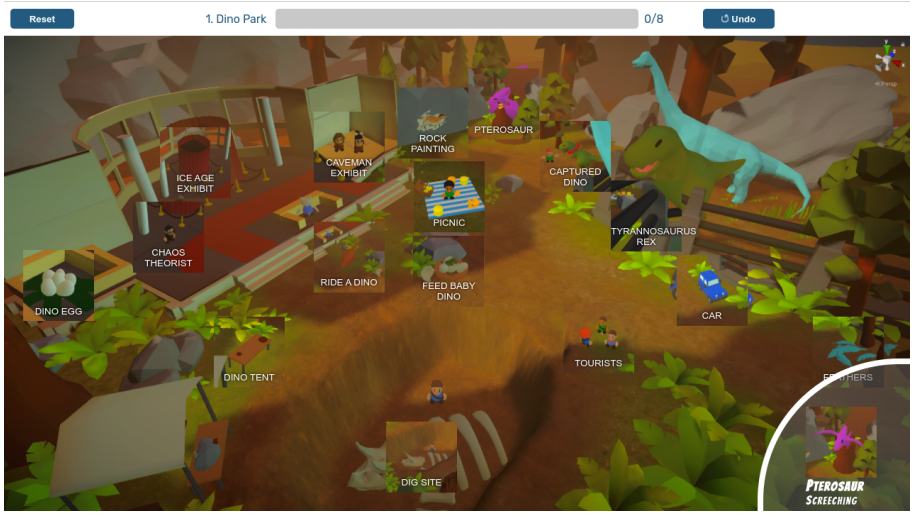
SeedQuest is intended to help users memorize the seed used in a *Brain Wallet*, a tool that uses the cryptographically strong seed to generate public and private keys for a cryptocurrency wallet. Past efforts have encoded this seed as a passphrase using the BIP-39 standard [29], resulting in a sequence of 12 words pulled from a set of 2048 possible words. SeedQuest seeks to improve the memorability of this seed by encoding it in users' memory using an approach similar to a memory palace.

On loading SeedQuest, it presents the user with a welcome screen, asking them to select between two modes: *Encode*, where they memorize a sequence of scenes, objects, and object states representing the system-assigned seed or *Decode* where they recover the seed by enter their sequence of scenes, objects, and object states. While SeedQuest is focused on cryptowallet seeds, there is no reason it couldn't be used to encode arbitrary system-assigned passphrases.

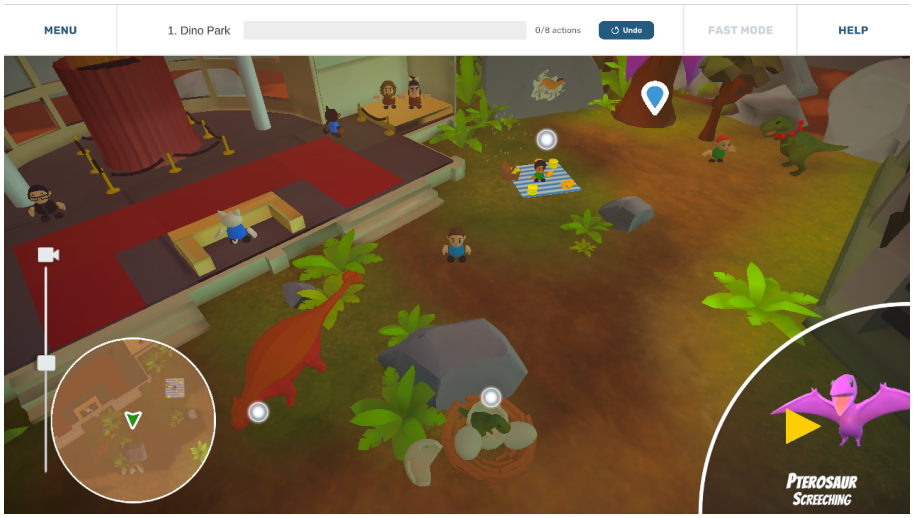
1) Memorizing the SeedQuest Path: In Encode mode, the user first enters a 12-word BIP-39-encoded passphrase generated by their cryptowallet. SeedQuest then encodes this passphrase as a sequence of 6 scenes, 3 interactable items per scene, with each item in one of four possible states. We refer



(a) A two-dimensional scene with objects placed in arbitrary and randomized positions



(b) A two-dimensional scene with objects placed in specific positions within the scene



(c) A three-dimensional scene with objects placed in specific positions within the scene

Fig. 1: The three approaches investigated in this paper based on a memory palace

to the sequence of scenes, objects, and object states as the *SeedQuest path*. Because random bits are mapped directly to scenes, items, and states, SeedQuest uses selection with replacement.

Users are then trained on the SeedQuest path. First, they are shown the list of 16 possible scenes and asked to select the scenes in order, with the appropriate scene indicated to the user and other options being unavailable to select (see Figure 2). Next, within each scene, users are trained on the sequence of items they must select and their desired states. This is done by displaying the item in its desired state in the lower right corner of the screen and having the user select that item (see Figure 1). After clicking the item, the user is shown a menu with the four possible states for that item (see Figure 3) and is required to click on the correct state. Hovering over any state previews the change, and the user may leave by pressing escape or clicking "×" in the upper right corner of the menu.

Throughout this process, users are shown a progress bar. Once the user has interacted with all the appropriate objects and set them to the appropriate state, they are presented with a popup asking if they are ready to move to the next scene. After completing all scenes, users are returned to the main menu, where they can choose to take the training again by clicking "Encode" and re-entering their BIP-39-encoded seed.

2) Entering the Memorized SeedQuest Path: In decode mode, users repeat their SeedQuest path to retrieve their BIP-39-encoded seed, which can then be entered into their cryptwallet. Users first select six scenes (in order), and then within each scene, they select the appropriate objects (in order) and the state for those objects (using the same menu as in the training process). SeedQuest provides no redundancy in the encoded seeds, so users must enter the sequence of scenes, objects, and states exactly to successfully retrieve their seed.

B. Three SeedQuest Variations

We forked SeedQuest to prepare it for use in our study. First, in alignment with prior research on system-assigned passwords [6], [13], [23], [9], [34], we modified SeedQuest to work with 5-word BIP-39-encoded passphrase, giving 55-bits of randomness. This passphrase was then encoded using 2 scenes with 4 interactable objects per scene, retaining the four possible states for each object. Second, due to restrictions on requiring Amazon Mechanical Turk crowdworkers (our study population) to install new software, we converted SeedQuest from a desktop application to one that runs in the browser using WebGL. As part of this process, we optimized textures to minimize download time.

Based on this modified version of SeedQuest, we created three designs based on the idea of a memory palace. We first describe Design 3 as it mostly closely aligns with SeedQuest's original design.

1) Design 3—Three-dimensional, navigable scenes: Other than the changes described above, this design matches SeedQuest exactly as described above (see Figure 1c). Of note, this representation acted somewhat like a video game, with users able to move their camera throughout the three-dimensional scene. We hypothesized that this design would perform the best, with the rationale that its game-like nature would aid users in memorizing the assigned SeedQuest path.

2) Design 2—Two-dimensional, static scenes: To investigate how important the three-dimensional, navigable representation of the scene was, we created a version of SeedQuest that replaced the three-dimensional scenes with a two-dimensional screenshot of the three-dimensional scenes. Unlike the three-dimensional version, the entire scene was visible at all times to the user (i.e., they did not move the camera through the scene. Additionally, the objects were represented as two-dimensional tiles the user would interact with. Otherwise, the scenes, objects, and object states were the same as in Design 3 (see Figure 1b). We hypothesized that this design would perform worse than Design 3 but better than Design 1.

3) Design 1—Randomized item locations: To investigate to what extent the placement of objects within the scene, as opposed to simply using thematically correct objects, impacted memorability, we created a version of SeedQuest that displays the two-dimensional scene from design 3, but moved all interactable object tiles to the top of the screen, with the order of the object randomized and re-randomized on each interaction. Otherwise, the scenes, objects, and object states were the same as in Design 2 (see Figure 1a) We hypothesized that due to the removal of locating objects within the scene, this design would perform the worst.

IV. Methodology

We conducted a longitudinal, between-subjects user study to measure the performance of our three designs. This study was conducted using Amazon Mechanical Turk and ran in June and July 2021. In total, 300 crowdworkers participated in our study. Our study sought to answer the following research questions:

1) Does a three-dimensional scene that you can move around improve memorability over a two-dimensional, static screenshot of that same scene? *(Comparing design 3 to 2)*
2) Does fixing the position of objects within a scene improve memorability over randomly placing objects in a list at the top of the scene? *(Comparing design 2 to 1)*
3) Do SeedQuest paths improve memorability over a BIP-39-encoded passphrase?
4) Does more time elapsing between usage of the memorization aid impact memorability?

To answer these questions, our study includes a between-subject component, where each participant is assigned to use one of our three system designs (addressing RQ1 and RQ2). Our study also includes a within-subject component involving each participant memorizing the BIP-39-encoded passphrase, acting as a control condition for our study (addressing RQ3). Finally, our study measures recall at both seven and thirty days for both the SeedQuest- and BIP-39-encoded passphrase (addressing RQ4).

This study was approved by our Institutional Review Board. All source code and resources used to conduct the study can be found at https://github.com/Iiridayn/seedquest-2d, https://github.com/Iiridayn/bip39generator, and at https://github.com/michaelmendoza/seedQuestAssets/tree/byu-fixes-06-2020.

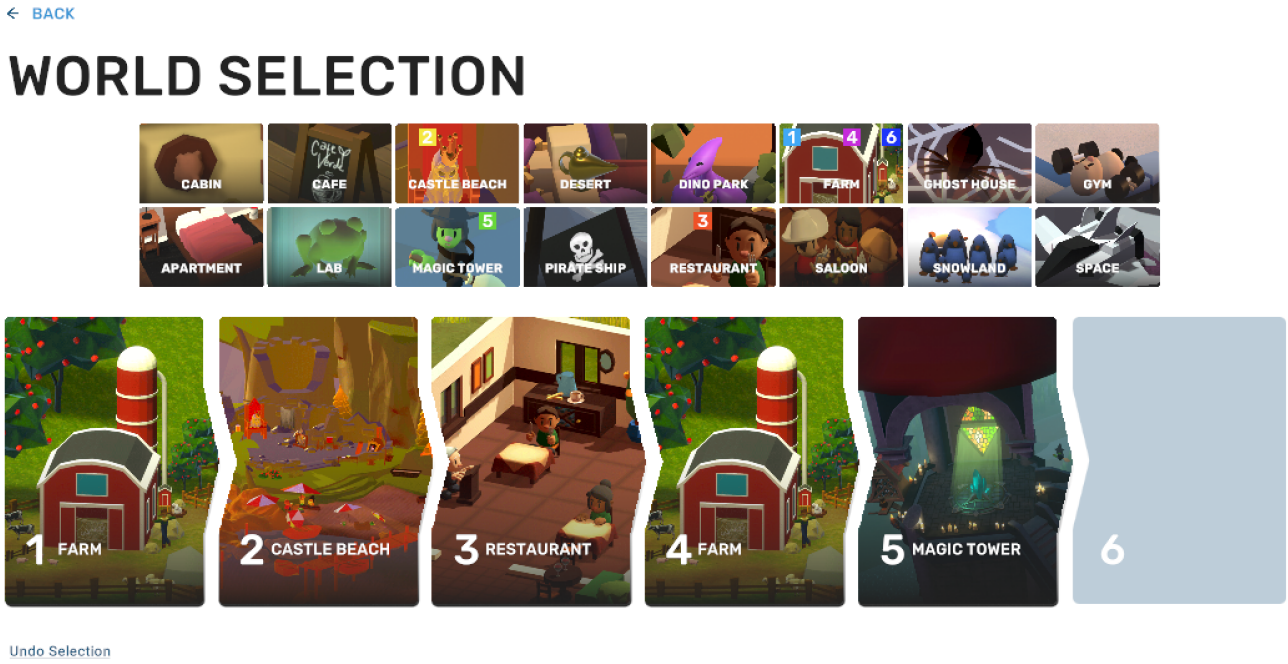


Fig. 2: The SeedQuest Scenes interface in encode mode, with five of six scenes selected

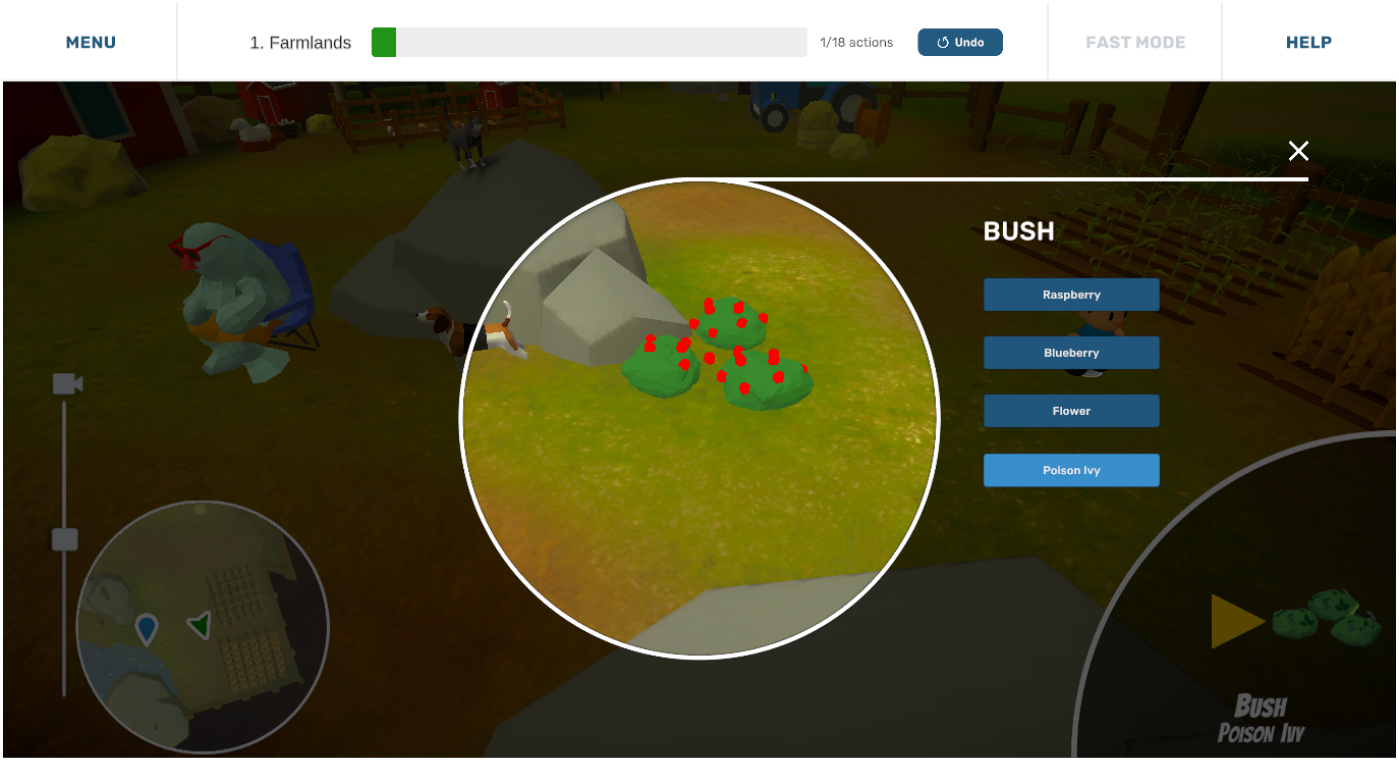


Fig. 3: SeedQuest, when selecting an interactable item

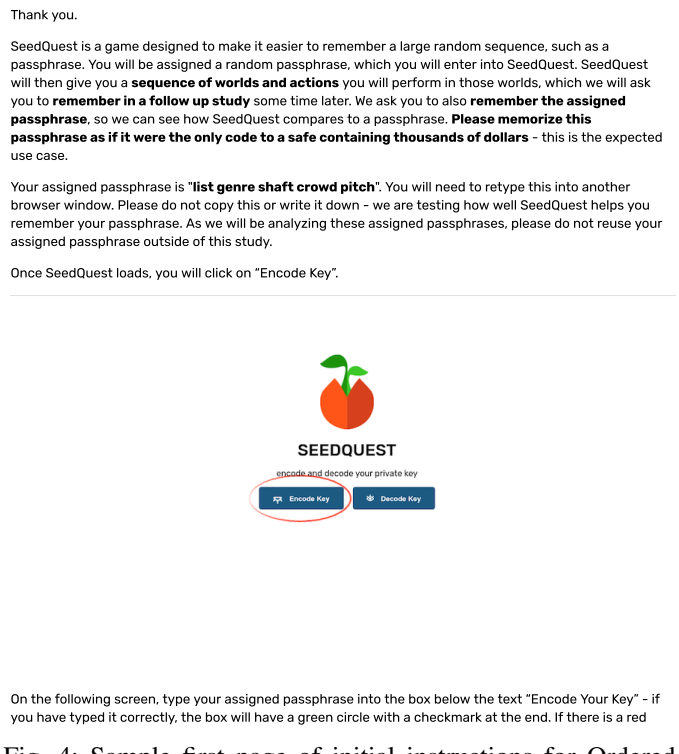


Fig. 4: Sample first page of initial instructions for Ordered condition

A. Study Design

Our study was composed of three parts: (1) an initial registration session, (2) a follow-up recall session at 7 days, and (3) a follow-up recall session at 30 days. Participants were paid USD $2 for completing the registration session, and USD $3 for each follow-up session they completed. Compensation was based on an estimated 20 minutes needed to complete the initial registration session and the first follow-up session, with pay weighted towards the follow-up to encourage participants to return. This represents a target payment of $15/hour, well within acceptable norms for crowdworker compensation.

We deliberately did not offer participants a bonus for successfully recalling their assigned authenticators. First, our goal was to measure human memory, and motivating participants to guarantee recall would likely have lead to more writing down their authenticators, and would have introduced a bias in our results. Second, a bonus would have communicated to participants that we desired recall as an outcome introducing bias due to *demand characteristics*, where participants change their behavior in response to what they believe experimenters want to see. Throughout we were careful to use neutral language ("Our goal is to determine how effective the approach used by SeedQuest is…") to avoid presenting a specific outcome as desirable. Third, Hyde and Jenkins show [19] that for situations similar to our experiment intent to remember something has no impact on the participant's ability to remember. For these reasons, we felt offering a bonus for successful recall would not be wise.

1) Registration Session: Before beginning the study, participants provided their informed consent to participate in the study. After completing this consent form, participants were randomly assigned to one of three conditions, one condition for each study design (addressing RQ1 and RQ2).

Next participants were shown introductory text describing SeedQuest to them (See Figure 4). They were also assigned a randomly generated 5-word BIP-39-encoded passphrase that they were told to memorize for follow-up studies that would occur at unspecified times in the future (addressing RQ3). At the end of the instructions, participants were instructed to select the "Encode" option in SeedQuest and to enter their assigned passphrase. This would take them through the training process intended to help them memorize their SeedQuest path as described in §III-A1.

Once participants completed this training, they were provided a link to click to complete the task and receive payment. We did not test memorability immediately after the registration session. Our rationale was that this improves ecological validity, as we believe it is unlikely that real-world users would immediately try and enter their SeedQuest path to recover their passphrase.

2) Follow-up Sessions: Seven and thirty days after the registration session, we invited participants to participate in a follow-up session where they would attempt to repeat their memorized SeedQuest path. Participants were not aware of when these follow-ups would occur, only that they would at some point. We chose to use two follow-up periods as this allowed us to measure the impact that different periods of time have on memorability (addressing RQ4).

We choose seven and thirty days based on prior work. First, seven days is a common waiting time before a follow-up session [6], [18], [13]. Second, the forgetting curve between 28 days and 360 days is much flatter than between 1 day and 28 days, meaning that 30 days is a good approximation for even longer periods of disuse [27].

Both follow-up sessions were identical, and we invited all participants who had completed the registration setting and who had not been excluded due to data quality issues (see §IV-C). Participants were first asked to enter as much of their passphrase as they could remember, entering a "?" as a placeholder for any words they couldn't remember. Next, participants were instructed to use SeedQuest's decode mode to enter their SeedQuest path. We used this order to prevent the decode operation from cueing recall of the passphrase, especially the screen at the end that shows the recovered passphrase.

After completing the decoding procedure, whether correctly or incorrectly, participants were asked to complete a post-task survey (see Appendix VIII). In the survey, we asked (1) whether they had written or saved their passphrase anywhere, with a note that we would not change compensation based on their response, (2) the System Usability Scale [7] questionnaire, (3) basic demographic questions (gender, age, education, occupation). All questions were optional. The survey also automatically recorded the passphrase decoded using SeedQuest.

B. Study Development

We conducted two pilot studies; one with a convenience sample at our institution, and the other with 9 Mechanical Turk

	3D	Ordered	Random	Total
Registered with study software	193	190	185	568
Completed SeedQuest path, even if wrong mode	66	134	120	320
Completed Encode mode	63	126	115	304
Submitted the HIT	61	122	116	300
Multiple accounts	0	4	2	6
Fake completion code	5	0	0	6
Multiple registration	12	8	1	21
Failed to complete but submitted something	0	2	2	4
Total excluded	17	14	5	37
Used the demo passphrase instead of assigned	1	22	17	40
Demo passphrase, but with a typo	0	5	4	9
Created own path then learned that	2	5	10	17
Learned the assigned passphrase with a typo	1	1	4	6
Completed as expected	40	75	76	191
Total invited back	44	108	111	263
Account closed	1	2	0	3
First follow-up pool	43	106	111	260
Returned	37	87	94	218
Submitted	34	85	89	208
Completed Outro Survey	32	80	85	197
Faked decoding path	1	4	1	6
Invited back	31	76	84	191
Account closed	0	8	13	21
Second follow-up pool	31	68	71	170
Returned	26	60	64	150
Submitted	25	60	60	145
Completed Outro Survey	25	59	60	144

TABLE I: Detailed information about study participation

workers. The lab pilot study identified a few minor issues, such as accessing other parts of the study by fiddling with the study URL. All identified issues were fixed before conducting the live pilot study, which revealed no new issues.

C. Participant Recruitment and Dropout

We recruited $n = 300$ participants using Amazon Mechanical Turk (MTurk). To ensure quality data, all were required to have at least 100 accepted HITs with a 95% approval rate and be in the United States of America. We selected this number based on a desire to have 50 participants in each of our three conditions, assuming a 50% dropout rate as was reported in similar prior work [6], [34].

Of these 300 participants, we excluded 43 participants due to data quality concerns:

- 27 participants completed the study multiple times, including six who used the same passphrase each time.
- 10 participants failed to submit valid completion codes.
- 6 participants found a way to avoid completing the SeedQuest encode step.

Table I shows detailed participant counts for each condition, including dropouts and exclusions. After removing these participants, we were left with 257 participants whom we invited to the two follow-up sessions. Of those participants, 191 completed the first follow-up session and 144 the second follow-up session.

D. Participant Demographics

We did not collect participant demographics in the registration session (there was no post-task survey).

In the 7-day follow-up study, we had more male participants (117; 59%) than female participants (80; 41%). The ages of the participants were as follows: 18–24 years (7; 4%), 25–34 (85; 43%), 35–44 (64; 32%), 45–54 (24; 12%), 55–64 (15; 8%), and 65+ (2; 1%). Over 82% of participants had a college degree. The participants had a wide variety of occupations, with the most common being computer[2] (42; 21%), MTurk (27; 14%), business (21; 11%), and sales (8; 4%).

For the 30-day follow-up study, we had more male participants (90; 63%) than female participants (54; 37%). The ages of the participants were as follows: 18–24 years (8; 6%), 25–34 (60; 42%), 35–44 (46; 32%), 45–54 (17; 12%), 55–64 (12; 8%), and 65+ (1%). Over 79% of participants had a college degree. The participants had a wide variety of occupations, with the most common being computer (29; 20%), MTurk (17; 12%), business (15; 10%), sales (12; 8%), and clerk (11; 8%).

E. Limitations

Our results are subject to some common limitations due to our study design. First, although Mechanical Turkers are similar to the general population in many ways [17], [32], since we conducted our study there have been concerns raised about the quality of MTurk-collected data [35]. We tried to address this by thoroughly checking our responses (see §IV-C), however, this could limit the generalizability of our results.

Second, we tested three variants of a SeedQuest and cannot claim that our results generalize to all possible graphical memorization systems. Similarly, we cannot guarantee how memorability will change outside the 7–30-day window we tested.

Specific to our study design, memory interference between the passphrase and the path is possible. We expected the path to take more advantage of different areas of memory than we found in practice, and so we did not control for this case. We note that our memorability results are similar to some of the related work discussed in Section II-B.

V. Results

Our experiment had several surprising outcomes. First, we expected participants to remember their SeedQuest path better when interactable items had a stable position in the scene, but found no such effect. This is surprising because we expected the position of items to strengthen recall and aid recognition. Second, participants recalled a significantly higher percentage of their SeedQuest path than their passphrase, while having no significant difference in their perfect recall percentages. Other results were less surprising; we will touch on them only briefly.

Throughout the tables and figures, we refer to the three designs as 3D (Design 3), Ordered (Design 2), and Random (Design 1).

A. RQ1 and RQ2—SeedQuest Design Differences

Using logistic regression, we find no significant impact of system design on SeedQuest path recall during either follow-up. This contradicted our assumption that Design 3 (three-dimensional, navigable scenes) would fare the best followed

[2]In these demographics, the "computer" occupation class represents users who selected "Computer/IT professional, Programmer, Data Scientist, Statistician" from the occupation dropdown.

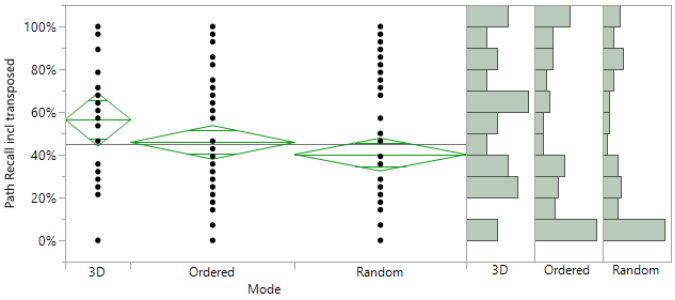
	3D	Ordered	Random	Total
Perfect Memory	4	10	8	22
Only Transposition Errors	0	2	0	2
Missed Only One State	2	4	2	8
Only Option Errors	0	2	5	7
Missed Only One Item	3	9	10	22
Correct Scenes, missing 2 Items	3	2	6	11
Correct Scenes, missing > 2 Items	12	11	5	28
One scene and 3+ items, some transposed	4	10	7	21
Remembered *something*	3	18	18	39
No memory	3	18	28	49
Total	34	86	89	209

(a) 7-Day follow-up

	3D	Ordered	Random	Total
Perfect Memory	3	4	4	11
Only Transposition Errors	0	2	2	4
Missed Only One State	0	4	2	6
Only Option Errors	1	1	2	4
Missed Only One Item	4	4	7	15
Correct Scenes, missing 2 Items	2	8	3	13
Correct Scenes, missing > 2 Items	8	10	8	26
One scene and 3+ Items, some transposed	0	3	5	8
Remembered *something*	7	11	14	32
No memory	0	13	14	27
Total	25	60	61	146

(b) 30-Day follow-up

TABLE II: SeedQuest path memory errors



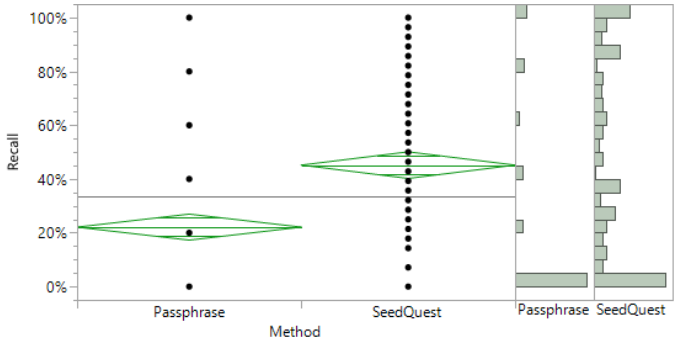
Dots represent individual data points. The diamond gives the 95% confidence interval. The line in the background is the overall mean. The histogram on the right provides a clearer view of the shape of the data.

Fig. 5: 7-Day follow-up recall rates when allowing for incorrectly ordered items

by Design 2 (two-dimensional, static scenes) and then Design 1 (two-dimensional, randomized item locations).

Table II provides more details on the types of errors that participants made when entering their passphrase. Based on these results, we find suggestive but inconclusive evidence (3-Way ANOVA, $p = 0.0878$) that during the 7-day follow-up, Design 3 did outperform the other designs in memorability when transposition errors were allowed (see Figure 5).[3] Digging in further, Tukey-Kramer HSD shows convincing evidence that Design 3 was best at helping users remember scenes ($p = 0.0031$), with a linear regression finding convincing evidence ($p = 0.0011$) that Design 3 led to a mean increase of 0.31 (0.13–0.50, 95% CI) scenes recalled when compared to Design 1.

[3]A transposition error refers to a participant correctly selecting an item in the path, but doing so in the wrong order.



Dots represent individual data points. The diamond gives the 95% confidence interval. The line in the background is the overall mean. The histogram on the right provides a clearer view of the shape of the data.

Fig. 6: 7-Day follow-up partial recall rates

B. RQ3 and RQ4—SeedQuest Path vs Passphrase Memorability

We decided to combine RQ3 and RQ4, comparing aggregate SeedQuest path memorability against passphrase memorability, as the differences in perfect recall between SeedQuest designs were not statistically significant (see Section V-A).

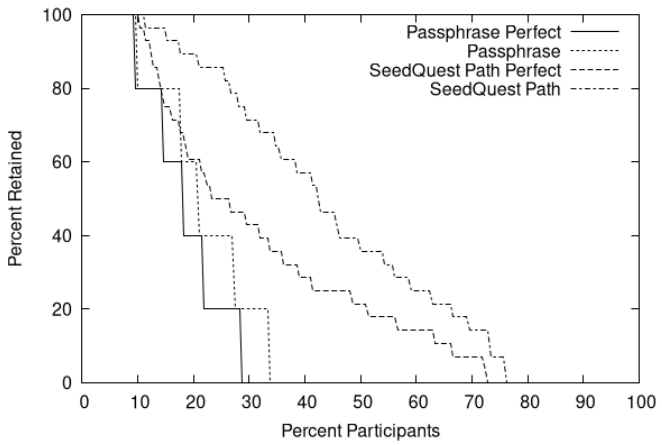
We found no significant differences between the perfect recall of the assigned passphrase and the SeedQuest path. In the 7-day follow-up, 21/218 (9.6%) of participants perfectly recalled their assigned passphrase, and 22/209 (10.6%) of participants perfectly recalled their assigned SeedQuest path. In the 30-day follow-up, 14/150 (9.3%) perfectly recalled their assigned passphrase and 11/146 (7.5%) perfectly recalled their assigned SeedQuest path. These results are similar to Brumen [9], of 10% after 7 days. Using Fisher's Exact Test, we find no evidence ($p = 0.4418$ 7-day follow-up, $p = 0.7776$ 30-day follow-up) that SeedQuest paths lead to higher perfect recall than passphrases.

However, using logistic regression, we find convincing evidence ($p < 0.0001$) that participants recalled an average of 23% more of their assigned SeedQuest path than their passphrase (with a 95% confidence interval from 16% to 30%), during the 7-day follow-up (see Figure 6).[4] We similarly find convincing evidence ($p < 0.0001$) that during the 30-day follow-up, participants recalled an average of 31% more of their assigned SeedQuest path than their passphrase (with a 95% confidence interval from 24% to 38%). We refer to this increased partial recall as *durability* and plot it in Figure 7.

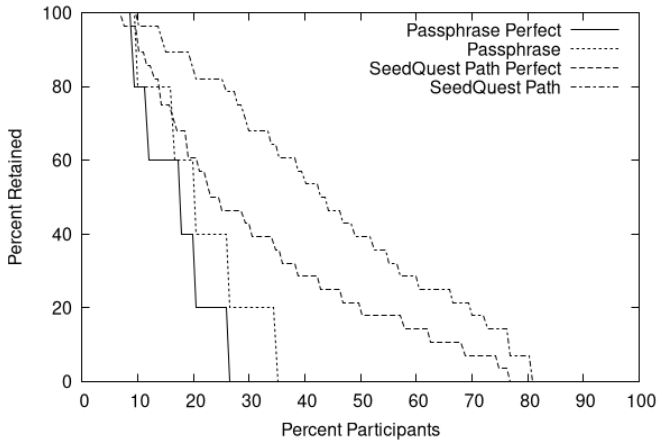
114 participants chose to repeat the encoding mode multiple times. Using logistic regression, we find convincing evidence ($p < 0.0001$) of a correlation, that the probability of perfect SeedQuest path recall after 30 days rises by 23.5% for each repetition of encoding mode.

Table III provides more details on the types of errors that participants made when entering their passphrase.

[4]Table III provides more details on the types of errors that participants made when entering their passphrase.



(a) 7-Day follow-up



(b) 30-Day follow-up

Passphrase/SeedQuest Path Perfect refers to the portion of items that were in the correct position. *Passphrase/SeedQuest Path* refers to the portion of items that were correctly recalled, but placed in an incorrect position.

Fig. 7: Durability comparison for passphrases and SeedQuest paths

Perfect	Transposed	3D	Ordered	Random	Total
0	0	18	62	64	144
0	1	0	0	4	4
0	2	5	0	1	6
0	3	1	0	0	1
1	0	2	6	2	10
1	1	0	1	3	4
1	3	1	0	0	1
2	0	1	1	2	4
2	1	1	0	0	1
2	2	0	1	2	3
3	0	2	2	1	5
3	1	0	0	2	2
3	2	0	0	1	1
4	0	1	5	5	11
5	0	5	9	7	21
		37	87	94	218

(a) 7-Day follow-up

Perfect	Transposed	3D	Ordered	Random	Total
0	0	13	44	40	97
0	1	0	4	4	8
0	2	3	0	1	4
0	3	1	0	0	1
1	0	1	1	3	5
1	1	0	1	3	4
2	0	1	0	0	1
2	1	1	1	0	2
2	2	0	1	0	1
3	0	1	1	1	3
3	1	0	1	4	5
3	2	0	0	1	1
4	0	2	1	1	4
5	0	3	5	6	14
		26	60	64	150

(b) 30-Day follow-up

For each word participants got correct and in the correct position, we added one point to "Perfect". For each word participants got correct but in the wrong position, we added one point to "Transposed".

TABLE III: Passphrase memory errors

C. Other results

1) Perceived Usability: Surprisingly, using 3-way ANOVA we found no significant difference in SUS scores (mean=56.2, $p = 0.6427$ at the 7-day follow-up, mean=58.8, $p = 0.3600$ at the 30-day follow-up) between the SeedQuest treatments. This suggests that participants found all three designs equally easy to use.

2) Writing Down the Password: Table IVa shows perfect memorability of the passphrase and SeedQuest path, grouped by treatment and by if participants reported writing it down. Table IVb shows the same for the 30-day follow-up.

Interestingly, those who answered "Yes" to "Did you write or save your passphrase anywhere?" had less recollection of both their assigned passphrases and their assigned SeedQuest paths (see Table IV).[5] For example, using logistic regression, we find convincing evidence ($p = 0.0003$) of a 19.6% reduction in the probability of perfect passphrase recall during the 7-day follow-up with a 95% confidence interval from 9.2% to 30.0%, and suggestive evidence ($p = 0.0620$) of a 12.9% reduction in the probability of perfect passphrase recall during the 30-day follow-up with a 95% confidence interval from -0.1% to 26.4%, among those that answered "Yes". Similarly, we find convincing evidence ($p < 0.0001$) of a 21.2% reduction in the probability of perfect SeedQuest path recall during the 7-day follow-up with a 95% confidence interval from 11.5% to 30.8%, as well as convincing evidence ($p = 0.0002$) of a 25.0% reduction in perfect SeedQuest path recall during the 30-day follow-up with a 95% confidence interval from 12.3% to 37.6%.

VI. Discussion

Below we discuss some of the more interesting takeaways from our study results.

A. Spatial Memory Unused

We expected that participants would remember navigating the 3D environment (Design 3) or where on the screen they

[5]We did not ask if participants referred to what they had written, only the question as stated.

Treatment	Wrote	Participants	Passphrase	SeedQuest
3D	null	2	0	1
3D	no	29	5	3
3D	yes	1	0	0
Ordered	null	1	0	0
Ordered	no	54	4	8
Ordered	yes	25	2	2
Random	null	1	0	0
Random	no	55	7	7
Random	yes	29	0	1
Sum by treatment:				
3D		32	5	4
Ordered		80	6	10
Random		85	7	8
Totals:		197	18	22

(a) 7-Day follow-up

Treatment	Wrote	Participants	Passphrase	SeedQuest
3D	no	23	3	3
3D	yes	2	0	0
Ordered	no	46	4	4
Ordered	yes	13	1	0
Random	null	1	0	0
Random	no	44	5	4
Random	yes	15	1	0
Sum by treatment:				
3D		25	3	3
Ordered		59	5	4
Random		60	6	4
Totals:		144	14	11

(b) 30-Day follow-up

The number of participants who reported writing down their SeedQuest path against the number who perfectly recalled.

TABLE IV: Participant recall based on whether they reported writing the SeedQuest path down during the initial study

found the interactable objects (Design 2), producing better memorability in these two treatments than if location of objects was randomized in the scene (Design 1). That we saw no evidence of this effect in either system is surprising.

This suggests that participants relied entirely on the visual characteristics of the interactable items and states, or on the word labels or a word encoding of those visual characteristics. Because the passphrase is also encoded as words with possible imagined visuals to pair with the words, it is further surprising that the 5-chunk passphrase was no better than the path of at least $2 + 2 \times 4 = 10$ chunks.

It is possible that both techniques contained distinct advantages and disadvantages, the sum of which canceled out, resulting in no discernible advantage between the path and passphrase. Alternatively, the hierarchical SeedQuest design may have been treated as two groups of 5 chunks, perhaps somehow equally memorable. Future experiments will be needed to control for these confounding variables.

B. Technique, Not Representation

Miller showed advantages in chunking for encoding values in memory [26]. However, this and other experiments in random password memorization (see Section II-B) have not found improved recall rates despite experimenting with different chunking and representations. Instead, the experiments with the highest recall percentages have leveraged memory techniques. This suggests that, as a research community studying password memorability, we might find easier wins by focusing on ways to embed memory training into our systems instead of searching for better ways to represent random values.

C. High-fidelity 3D Environments

We found no significant improvement between our 2D faithful clone and the high-fidelity 3D virtual environments. This suggests that we may be able to investigate the same research questions using cheaper 2D lower-fidelity lab prototypes.

However, there was suggestive but inconclusive evidence of some kind of difference. Further analysis detailed in Section V-A shows that the scenes were more memorable for the original 3D system. If 3D autonomous movement triggered deeper processing of the scene itself, it would provide a possible reason for this effect. Deeper processing may be achieved in a number of ways; one such could be adding some multiple choice questions afterwards such as "would you expect to see a person in a cowboy hat in that scene?". Further work could investigate the underlying cause behind the superiority of the 3D system only for scene recall.

D. SUS Scores

We found no significant difference between the SUS scores for all SeedQuest variants. This shows that cheaper lab prototypes may be equally usable to the full high-fidelity system. However, we failed to identify which parts of the system design caused the SUS scores to be low (a commonly accepted average SUS score is 68 [33]). Future research could identify which factors resulted in low usability.

E. Recording the Passphrase

Participants who claimed to have recorded their passphrase somewhere were less likely to remember it during the follow-up sessions. This is an unexpected result, but we found only statistical correlation rather than cause and effect. It is possible that those participants spent less effort memorizing, and when they returned they tried honestly to recall their passphrase and path from memory. We feel this correlation is noteworthy but not practically significant.

F. Future Work

Longitudinal research into *which* design factors help users remember random passwords is uncommon, and we would like to see more of it. This study revealed some unexpected areas where changes did not influence memory; an interesting area for future work would be to make further changes to SeedQuest-like systems until something does break, as we attempted in this study.

Another fascinating research area is indicated by our result showing that participants recalled a higher percentage of their path than their passphrase. Can SeedQuest-like systems enable error-correcting authentication? How many redundant bits would we require to reach acceptable recall rates, considering the added cost of memorizing the redundant bits?

We did not research the optimal performance of the SeedQuest system itself, opting instead to attempt to break it

to see how it works. Future work could use it as intended, comparing it against other "best system" approaches such as in prior work [6], [18], [13].

Another interesting question is the interference effects between multiple simultaneous uses of a single SeedQuest-like system or SeedQuest-like systems and passphrases. We did not design our experiment to test this, preferring to find what makes SeedQuest work, but it would be an excellent area for future work to address, especially should SeedQuest perform well in a "best system" approach.

However, our results in the context of the broader research literature point more strongly towards memory techniques rather than entropy representations. Are there interaction effects between certain techniques and representations, or is there a "best technique" that appears to apply to most or all systems? Further research is needed to explore training alternatives that increase memorability. Notably, those who repeated encoding mode four or more times all recalled their paths perfectly after 30 days — while these participants self-selected, it suggests that few repetitions might be necessary for high memorability with SeedQuest.

SeedQuest systems use a nested hierarchy of scene-item-state, and use selection with replacement to select from each of these layers. We suspect further design disruptions, including reversible selection without replacement and a flat hierarchy may yield better memorability. One such design would have blocks of bits select items from a group; for example, the first 5 bits would select a single item from 32 items, requiring $\frac{55}{5} \times 32 = 352$ distinct items for a similar 55-bit study (SeedQuest has $16 \times 16 = 256$ items and $16 \times 16 \times 4 = 1024$ item states, so this is feasible, especially if using lower fidelity designs). The items would not need to be clustered in a larger environment. We also found that users sometimes selected scenes and interactable items out of order. While Shay et al. [34] showed that passphrases do not benefit from allowing out-of-order entry, future work might evaluate if SeedQuest-like systems benefit from orderless entry.

SeedQuest was also initially designed to help users memorize sequences of 132 bits; 128 random bits total. We have not found any related scholarly work investigating the memorization of passwords of more than 100 bits, though we are aware of people who use them. Future work could establish a baseline memorability for similar high-entropy passwords, possibly contrasting with SeedQuest as originally designed.

VII. Conclusions

Our study resulted in several surprising outcomes with practical significance.

First, the memory-palace-based designs resulted in a higher percentage of the random bits recalled over passphrases despite statistically indistinguishable rates of perfect recall. This implies that SeedQuest-like systems are a better fit for "fuzzy" or error-correcting authentication, such as systems that retrain the user on their password after authenticating them if they fail to enter their password perfectly. We do not know which features of the SeedQuest design led to this outcome; however, our results find no evidence of an impact from either 3D autonomous motion or fixed positions in a scene.

Second, we found no evidence that the high-fidelity 3D design improved memorability over the 2D designs. This implies that similar inexpensive lab prototypes may be as effective when investigating passphrase memorability. Unexpectedly, presenting items in a re-randomizing list was just as memorable as the high-fidelity 3D representation.

Acknowledgment

The authors thank Rand Al Rabadi for her help with implementing the 2D versions of the system.

This work was partially supported by the National Science Foundation under Grant Nos. CNS-1816929 and CNS-2238001. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] F. A. Alsulaiman and A. El Saddik, "Three-dimensional password for more secure authentication," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 9, pp. 1929–1938, 2008.

[2] R. Biddle, S. Chiasson, and P. C. Van Oorschot, "Graphical passwords: Learning from the first twelve years," *ACM Computing Surveys (CSUR)*, vol. 44, no. 4, pp. 1–41, 2012.

[3] J. Blocki, S. Komanduri, L. Cranor, and A. Datta, "Spaced repetition and mnemonics enable recall of multiple strong passwords," 2015.

[4] H. Bojinov, D. Sanchez, P. Reber, D. Boneh, and P. Lincoln, "Neuroscience meets cryptography: designing crypto primitives secure against rubber hose attacks," in *Proceedings of the 21st USENIX Security Symposium*, 2012, pp. 33–33.

[5] J. Bonneau, C. Herley, P. C. Van Oorschot, and F. Stajano, "The quest to replace passwords: A framework for comparative evaluation of web authentication schemes," in *2012 IEEE Symposium on Security and Privacy*. IEEE, 2012, pp. 553–567.

[6] J. Bonneau and S. Schechter, "Towards reliable storage of 56-bit secrets in human memory," in *23rd USENIX Security Symposium USENIX Security 14)*, 2014, pp. 607–623.

[7] J. Brooke *et al.*, "SUS: A 'quick and dirty' usability scale," *Usability Evaluation in Industry*, vol. 189, no. 194, pp. 4–7, 1996.

[8] S. Brostoff and M. A. Sasse, "Are passfaces more usable than passwords? a field trial investigation," in *People and Computers XIV—Usability or Else!: Proceedings of HCI 2000*. Springer, 2000, pp. 405–424.

[9] B. Brumen, "System-assigned passwords: The disadvantages of the strict password management policies," *Informatica*, vol. 31, no. 3, pp. 459–479, 2020.

[10] P. Cipresso, A. Gaggioli, S. Serino, S. Cipresso, and G. Riva, "How to create memorizable and strong passwords," *Journal of Medical Internet Research*, vol. 14, no. 1, p. e10, 2012.

[11] S. Das, D. Lu, T. Lee, J. Lo, and J. I. Hong, "The memory palace: Exploring visual-spatial paths for strong, memorable, infrequent authentication," in *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, 2019, pp. 1109–1121.

[12] M. Dell'Amico, P. Michiardi, and Y. Roudier, "Password strength: An empirical analysis," in *Proceedings of the 29th IEEE Conference on Computer Communications*. IEEE, 2010.

[13] J. Doolani, M. Wright, R. Setty, and S. T. Haque, "Locimotion: Towards learning a strong authentication secret in a single session," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–13.

[14] M. Dresler, W. R. Shirer, B. N. Konrad, N. C. Müller, I. C. Wagner, G. Fernández, M. Czisch, and M. D. Greicius, "Mnemonic training reshapes brain networks to support superior memory," *Neuron*, vol. 93, no. 5, pp. 1227–1235, 2017.

[15] D. Florencio and C. Herley, "A Large-Scale Study of Web Password Habits," in *Proceedings of the 16th International Conference on World Wide Web*. ACM, 2007, pp. 657–666.

[16] C. George, M. Khamis, D. Buschek, and H. Hussmann, "Investigating the third dimension for authentication in immersive virtual reality and in the real world," in *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 2019, pp. 277–285.

[17] J. K. Goodman, C. E. Cryder, and A. Cheema, "Data collection in a flat world: The strengths and weaknesses of mechanical turk samples," *Journal of Behavioral Decision Making*, vol. 26, no. 3, pp. 213–224, 2013.

[18] S. T. Haque, M. N. Al-Ameen, M. Wright, and S. Scielzo, "Learning system-assigned passwords (up to 56 bits) in a single registration session with the methods of cognitive psychology," in *Proceedings of the Network and Distributed System Security Symposium (NDSS 2017), USEC*, vol. 17, 2017.

[19] T. S. Hyde and J. J. Jenkins, "Recall for words as a function of semantic, graphic, and syntactic orienting tasks," *Journal of Verbal Learning and Verbal Behavior*, vol. 12, no. 5, pp. 471–480, 1973.

[20] B. Krebs, "Experts fear crooks are cracking keys stolen in lastpass breach | krebs on security," 2023, retrieved 2023-09-14 from https://krebsonsecurity.com/2023/09/experts-fear-crooks-are-cracking-keys-stolen-in-lastpass-breach.

[21] T. K. Landauer, "How much do people remember? some estimates of the quantity of learned information in long-term memory," *Cognitive Science*, vol. 10, no. 4, pp. 477–493, 1986.

[22] E. L. Legge, C. R. Madan, E. T. Ng, and J. B. Caplan, "Building a memory palace in minutes: Equivalent memory performance using virtual versus conventional environments with the method of loci," *Acta Psychologica*, vol. 141, no. 3, pp. 380–390, 2012.

[23] M. D. Leonhard and V. Venkatakrishnan, "A comparative study of three random password generators," in *2007 IEEE International Conference on Electro/Information Technology*. IEEE, 2007, pp. 227–232.

[24] C. R. Madan, "Augmented memory: a survey of the approaches to remembering more," *Frontiers in Systems Neuroscience*, vol. 8, p. 30, 2014.

[25] M. A. McDaniel, G. O. Einstein, E. L. DeLosh, C. P. May, and P. Brady, "The bizarreness effect: It's not surprising, it's complex." *Journal of Experimental Psychology: Learning, Memory, and Cognition*, vol. 21, no. 2, p. 422, 1995.

[26] G. A. Miller, "The magical number seven, plus or minus two: Some limits on our capacity for processing information." *Psychological Review*, vol. 63, no. 2, p. 81, 1956.

[27] R. S. Nickerson, "A note on long-term recognition memory for pictorial material," *Psychonomic Science*, vol. 11, no. 2, pp. 58–58, 1968.

[28] S. Oesch and S. Ruoti, "That was then, this is now: a security evaluation of password generation, storage, and autofill in browser-based password managers," in *Proceedings of the 30th USENIX Security Symposium*. USENIX, 2020.

[29] M. Palatinus, P. Rusnak, A. Voisine, and S. Bowe, "Mnemonic code for generating deterministic keys," BIP 39, September 2013, retrieved 2021-06-01 from https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki.

[30] S. Pearman, J. Thomas, P. E. Naeini, H. Habib, L. Bauer, N. Christin, L. F. Cranor, S. Egelman, and A. Forget, "Let's go in for a closer look," in *Proceedings of the 24th ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017.

[31] G. W. Records, "Most Pi places memorized | Guiness World Records," 2018, retrieved 2021-08-17 from https://www.guinnessworldrecords.com/world-records/most-pi-places-memorised.

[32] E. M. Redmiles, S. Kross, and M. L. Mazurek, "How well do my results generalize? comparing security and privacy survey results from mturk, web, and telephone samples," in *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2019, pp. 1326–1343.

[33] J. Sauro, "Measuring usability with the system usability scale (SUS)," Feb. 2011, retrieved 2021-09-02 from https://measuringu.com/sus/.

[34] R. Shay, P. G. Kelley, S. Komanduri, M. L. Mazurek, B. Ur, T. Vidas, L. Bauer, N. Christin, and L. F. Cranor, "Correct horse battery staple: Exploring the usability of system-assigned passphrases," in *Proceedings of the eighth symposium on usable privacy and security*, 2012, pp. 1–20.

[35] J. Tang, E. Birrell, and A. Lerner, "Replication: How well do my results generalize now? the external validity of online privacy and security surveys," in *Proceedings of the Eighteenth Symposium on Usable Privacy and Security (SOUPS 2022)*, 2022, pp. 367–385.

[36] E. Tulving and D. M. Thomson, "Encoding specificity and retrieval processes in episodic memory." *Psychological Review*, vol. 80, no. 5, p. 352, 1973.

[37] B. Ur, F. Noma, J. Bees, S. M. Segreti, R. Shay, L. Bauer, N. Christin, and L. F. Cranor, "'I added '!' at the end to make it secure': Observing Password Creation in the Lab," in *Proceedings of the Eleventh Symposium On Usable Privacy and Security*, 2015.

[38] M. Vasek, J. Bonneau, R. Castellucci, C. Keith, and T. Moore, "The bitcoin brain drain: Examining the use and abuse of bitcoin brain wallets," in *International Conference on Financial Cryptography and Data Security*. Springer, 2016, pp. 609–618.

[39] R. W. Weisberg and L. M. Reeves, *Cognition: From Memory to Creativity*. John Wiley & Sons, 2013.

[40] Q. Yan, J. Han, Y. Li, and H. Deng, "On limitations of designing usable leakage-resilient password systems: Attacks, principles and usability," 2012.

APPENDIX

VIII. SURVEY QUESTIONS

In the outro survey during both follow-ups we asked participants the same questions.

1) Text field, "Recovered Passphrase". We pre-filled this field with their first recovered passphrase.
2) Radio selection, "Did you write or save your passphrase anywhere? (this is for analysis only and will not change your compensation)".
3) The 10 question SUS Questionnaire [7].
4) Dropdown, "Gender", with the options "Male", "Female", and "Non-binary".
5) Dropdown, "Age", with the options "18 – 24", "25 – 34", "35 – 44", "45 – 54", "55 – 64", and "65 or older".
6) Dropdown, "Highest Completed Formal Education", with the options "Some High School", "High School / GED", "Some University", "Associates Degree", "Bachelors Degree", "Graduate Degree", and "Other".
7) Dropdown, "Current occupation — primary source of income", with the options listed below.

We derived the list of occupations from the 2018 US BLS Standard Occupational Classifications, presenting the following options:

- Architect, Engineer, Surveyor
- Art, Design, Entertainer, Journalist, Sports
- Business: Executive, Management, Advertising, Marketing, PR, or HR
- Clerk, Teller, Operator, Courier, Secretary, Data Entry, etc
- Computer/IT professional, Programmer, Data Scientist, Statistician
- Construction, Mining, Drilling
- Education: Teacher, Librarian, Curator, etc
- Fishing, Farming, Forestry
- Food preparation and service
- Healthcare assistant, Massage Therapist
- Homemaker

- Installation, Repair, Mechanic
- Legal: Lawyer, Judge, etc
- Maintenance, Pest control, Cleaning, Landscaping
- Medical professional, Dentist
- Mechanical Turk Worker
- Military
- Production: Assembly, Baker, Butcher, Machinist, Caster, Printer, Laundry, Tailor, Woodworker, Plant operator
- Protection: Law enforcement, Firefighters, Security, etc
- Retired
- Sales, including Retail
- Scientist or Technician
- Service: Usher, Embalmer, Barber, Tour Guide, Childcare, etc
- Social Worker or Religious Professional
- Student
- Transportation: Pilot, Trucker, Driver, Sailor, Traffic engineer, etc
- Unemployed
- Other