UNIVERSITEIT TWENTE.

Final Project Thesis

Developing a Tool for Learning Concept Maps

Author:
M.C. van den Enk
[s1004654]
m.c.vandenenk@student.utwente.nl

Supervisors:
dr. A.H. Gijlers
a.h.gijlers@utwente.nl
dr. L. Bollen
l.bollen@utwente.nl

June 22, 2017

Contents

Part I

Introduction

Project Description

Over the centuries, knowledge has been fundamental to any learning process. Socrates already stated that knowledge is the only true virtue, and the tragedian Aeschylus regarded memory as the mother of all knowledge. Moreover, it was not only regarded as important by ancient thinkers, but is still regarded as such by modern scholars on education. The taxonomy of learning by Bloom, Engelhart, Furst, Hill, and Krathwohl (1956), a revision of the taxonomy by Krathwohl (2002), and the three stages of skill acquisition by J. Anderson (1982), propose that all learning should start with memorising factual knowledge. Furthermore, von Glaserfield (2001), one of the main founders for critical constructivism, expresses a need for training students so that they permanently possess facts and are able to repeat them flawlessly whenever they are needed, while also understanding what is placed into their memory. Ericsson and Kintsch (1995) adds to this by stating that in order to perform complex tasks, people must maintain access to large amounts of information, and that solely encoding knowledge is not sufficient. Despite all of this, Karpicke (2012) argues that "[r]etrieval processes, the processes involved in using available cues to actively reconstruct knowledge, have received less attention" (p. 158), whereas basic research on learning and memory has emphasised that retrieval must be considered in any analysis of learning.

Traditionally, when students have to gain complex and meaningful knowledge – for example knowledge about a historical event or a chapter in a psychology textbook, they are asked to read the relevant chapter from a provided textbook. However, Mayer (2008) states that many students have difficulty gaining knowledge in this manner. He breaks reading for comprehension down into four separate skills, which are integrating, organising, elaborating, and monitoring. Integrating refers to relating a text to one's prior knowledge, for which evidence exists that rich background knowledge leads to better inferences about the text, and thereby to better comprehension. This need also has been stressed by Ausubel (1968), and forms different problems between individual readers having access to different background knowledge. After integration, the reader has to organise the text, so that the important ideas and the relationships among them are identified. This is mainly a problem for less experienced readers, possessing fewer strategies to quickly identify important parts and thereby spending too much time on reading unimportant information. While organising a text, the student also has to make necessary inferences while reading, or has to elaborate, which is quite difficult for readers when not prompted to do so. Finally, students have to monitor their comprehension, which refers to evaluating their understanding of the text and if necessary adjusting the reading strategy. This is again quite difficult for the average reader, however this can be trained.

While intergrating is something more dependent on the curriculum design, organising and elaborating can be facilitated by a technique called concept mapping, and monitoring by so-called flashcard systems. Furthermore, the flashcard system might be helpful for the integration of a next topic with the current. This research aims to develop a new tool combining these learning tools. In this chapter, concept mapping and flashcard systems be explored on a practical level in order to establish their definitions together with a summary of arguments in favour or opposition of using them as tools for studying textual material, while also describing their current applications within education. Furthermore, a new tool called the flashmap system is introduced.

Concept mapping

A Concept map is a learning tool deviced by Joseph Novak in 1970's, based on constructivist theories of learning. It was originally intended for assessing the structure of student conceptions, before and after instruction, in order to map their prior knowledge and compare it to what they learned during the instruction. This expanded on the notions of Ausubel (1968), who stated that what the learner already knows is most important, and that this had to be ascertained before teaching. Although the use of concept maps as an assessment tool remains prevalent (Cañas & Novak, 2012; Chung, O'Neil Jr., & Herl, 1999; Hwang, Wu, & Ke, 2011; Ruiz-Primo & Shavelson, 1996), over time, students began to use it as a tool to comprehend textual material by organising and elaborating on the included concepts (Cañas & Novak, 2012; Eppler, 2006; Hwang et al., 2011; Karpicke & Blunt, 2011; Nesbit & Adesope, 2006).

Definition

One definition provided by Burdo and O'Dwyer (2015) states that "concept maps are hierarchical representations of knowledge. Construction of them involves linking concepts [...] through the use of linking phrases into propositional statements" (p. 335). The concepts are typically nouns or verbs with or without modifying adjectives or adverbs, and linking phrases specify the relationship between two concepts. Ruiz-Primo and Shavelson (1996) also mention these elements in their own definition, yet Cañas and Novak (2012) and Eppler (2006) include a few extra features, such as the concepts being ordered in hierarchical fashion. They describe two different kinds of links, which are hierarchical links to indicate ranking between the concepts, and crosslinks to indicate relationships between concepts in different segments or domains of the concept map. The latter aims at relating concepts residing within different knowledge domains, enabling better connections to prior knowledge of the learner. According to Eppler (2006), concept maps are always top-down and show systematic relationships among sub-concepts relating to one main concept, however Cañas and Novak (2012) state that they can also be cyclical as long as the concepts still have a conceptual hierarchy. Finally, most of the above mentioned articles describe the links between concepts to be directed. In conclusion, the definition of concept maps used within this thesis will be:

A concept map refers to a directed graph, in which the nodes consist of concepts, and the edges of – either hierarchical or cross- – links labeled with linking phrases, forming several propositional statements about a knowledge domain.

An example of a concept map is displayed in figure 1.

For this study, the more interesting aspects of concept maps are the use of concept mapping for elaborating, and of demonstrating meaningful relationships between concepts to learners. The first use of the concept map is known as generative use, and the second as supplantive (Smith & Ragan, 2005).

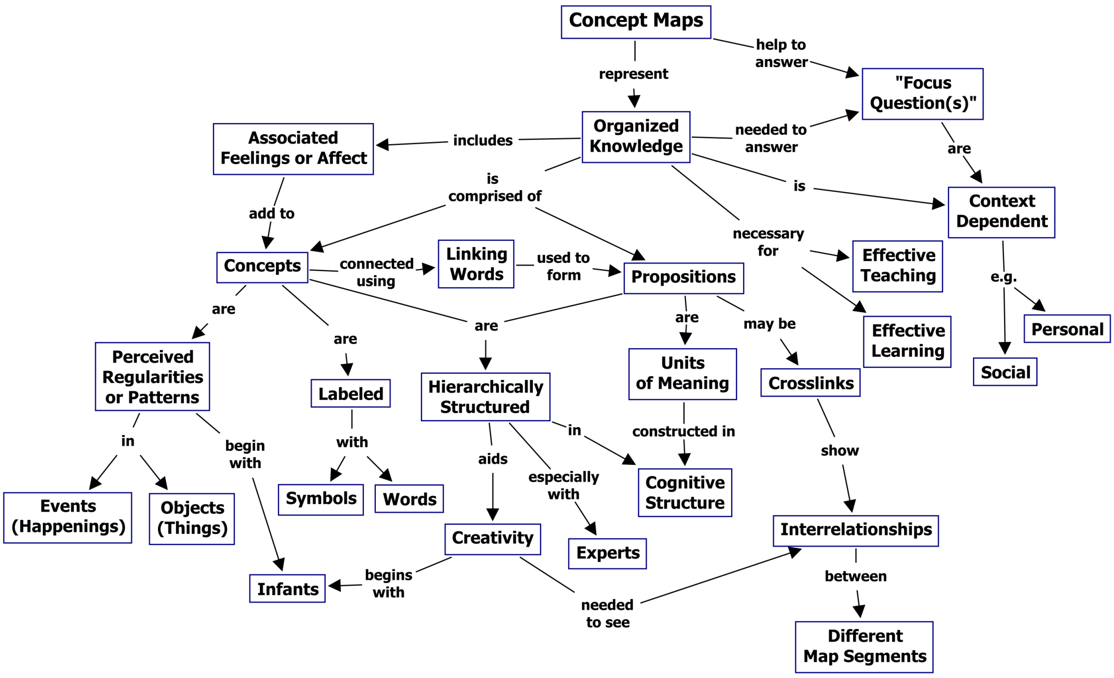


Figure 1: An example of a concept map by Novak and Cañas (2008)

Effectiveness

Multiple studies, both qualitative and quantitative, have demonstrated that concept maps can promote meaningful learning (Cañas & Novak, 2012; Hwang et al., 2011; Nesbit & Adesope, 2006; Subramaniam & Esprivalo Harrell, 2015; Singh & Moono, 2015). One of the positives of the concept map is that it does not provide learning by means of disconnected facts, but rather as a cohesive narrative placing emphasis on the connections between the concepts. However, most studies state that merely studying a concept map (supplantive use) is not sufficient, and that constructing the concept map (generative use) is essential for using it as a learning tool. However, no studies were found testing this hypothesis, and yet Blankenship and Dansereau (2000) have found that expert generated concept maps do help students gain conceptual understanding. Still, this study did also indicate that greater maps (more than 20 nodes) used within textbooks lead to *map-shock*, which Moore, North, Johri, and Paretti (2015) defines as “a type of cognitive overload that prevents students from effectively processing the concept map, thereby inhibiting their ability to learn from it” (p. 3). Finally, Eppler (2006) enlists some of the main advantages and disadvantages in comparison to other visualisation formats (mind maps, conceptual diagrams, and visual metaphors). A positive aspect is that students can gain information rapidly, because of the systematic, proven approach to provide an overview and the emphasis on relationships and connections among concepts. On the other hand, the technique of concept mapping is not easy to apply by novices and requires exstensive training, since otherwise the maps tend to turn out to be idiosyncratic. Furthermore, although better understandability is provided, the overall pattern does not necessarily assist memorability. Finally, the quality of concept maps can be assessed through evaluation rules, however this turns out to be quite a time consuming task for the tutors.

Flashcard system

In contrast to concept maps, a flashcard system is not intended for meaningful knowledge encoding, but rather for the rehearsal of knowledge so that it keeps active and as such is prevented from being forgotten.

Definition

In the context of language learning, Nakata (2011) defines flashcard systems as learning tools in which "target items are presented outside meaning-focused tasks, and learners are asked to associate the L2 [foreign language] word form with its meaning, usually in the form of a first language translation, L2 synonym, or L2 definition" (p. 17). This form of learning is also referred to as a *paired-associate format*, which refers to learning by the learner being presented by cues and having to recall an associated counterpart. Besides vocabulary learning, it can also be used to memorise word definitions or topographical information. In order to be more inclusive of other use cases, the following general definition is proposed:

> *A flashcard system refers to any system in which a learner is presented with cues and has to recall their counterparts from a paired-associate format.*

The most simple form of a flashcard system is a system where the learner has a stack of cards, with each containing a retrieval cue on one side and the correct associated response on the other side. A learning session then consists of going through the whole stack each day and trying to come up with correct answers. Efficiency can then be increased by repeating difficult cards more often, or skipping reviewing certain easy cards for multiple days. This way, the learner only focuses on the pairs which require more practice. Finally, the size of the stack of cards can be increased over multiple days in order to improve the spreading of cognitive load. Next to these paper flashcards, there is also a multitude of digital flashcard systems available (Hwang et al., 2011; Nakata, 2011; Edge, Fitchett, Whitney, & Landay, 2012), which allow for automating the rescheduling of flashcards and thereby providing better access to more advanced algorithms.

Effectiveness

Flashcard systems have not been completely free from criticism by other researchers. Hulstijn (2001) for example describes flashcards as a relic of the old-fashioned behaviourist learning model, and McCullough (1955) states that the main emphasis of flashcards is memorisation, not comprehension. However, Zirkle and Ellis (2010) state that it is still important for teachers and students to understand and utilise memory in such a way that a store of knowledge is produced that remains flexibly retrievable in a variety of contexts over a period of time. Flashcards have been found to be both a time efficient tool for learning large numbers of facts and an effective tool for these facts to be more resistant to decay in comparison to traditional teaching methods (Nakata, 2011). Furthermore, Kornell and Bjork (2008) state that "perhaps no memorisation technique is more widely used than flashcards" (p. 125). Their effectiveness also has been demonstrated accross studies in different contexts, for example that of language learning (Chien, 2015; Macquarrie, Tucker, Burns, & Hartman, 2002; McCullough, 1955; Nakata, 2011), word recognition (Joseph, Eveleigh, Konrad, Neef, & Volpe, 2012), psychology courses (Burgess & Murray, 2014; Golding, Wasarhaley, & Fletcher, 2012), and geography (Zirkle & Ellis, 2010). Therefore, many authors support pursuing research into flashcards and its effective application into classrooms.

Comparison of the two tools

In summary, most studies describe concept mapping as a tool for meaningful encoding, whereas flashcards are described as a tool for rote memorisation, and therefore imply that the former approach leads to more comprehension than the latter. A recent study by Karpicke and Blunt (2011) researched this hypothesis by having participants study a science text with four different learning conditions and prompting them afterwards with verbatim and inference questions and metacognitive predictions. Within the first condition, students only had to read the text and then answer the quesions. The second group studied the text repeatedly in four consecutive study periods. Students within the third group studied the text in one initial study period and then created a concept map after being instructed in concept mapping. The final group studied the text in an initial study period and then had to retrieve as much from the text as they could on a free recall test. The time spent on concept mapping and recalling was equal. When analysing the results, it was found that the retrieval group performed highest on both the verbatim and the inference questions, whereas the repeated study and concept mapping groups performed about equally well and the study once group performed the worst. Interestingly enough, the retrieval group judged their own learning the lowest, and the repeated study group the highest. The same effect of concept mapping and retrieval practice was found again in a second reproduction study, and also in another study by Burdo and O'Dwyer (2015). It is theorised that during elaboration, subjects attain detailed representations of encoded knowledge by linking concepts together in meaningful ways, but that during retrieval, subjects use retrieval cues to reconstruct meaning en thereby already organise the content in a meaningful way. Karpicke and Blunt (2011) conclude that these insights could pave the way for the design of new educational activities with retrieval practices in mind.

Flashmap system

It can be concluded that both of these tools are helpful for studying, since concept maps help students organise by drawing hierarchical links and elaborate on the content by drawing cross-links, and flashcards help students monitor their understanding of the text and retain the knowledge in order to facilitate integration with a following topic where the knowledge may prove relevant. The object of this study is therefore to create a new learning tool, referred to as Flashmap system, that combines both the visual overview of concept maps with the retrieval mechanism of flashcard systems. It will present incomplete parts of a concept map, in which the student has to fill in the missing parts of propositions represented by that map (see figure 2). These parts will consecutively be repeated according to algorithms already used by digital flashcard systems. Thereby, the system should make the relations between the concepts explicit to the student, increasing the organisation of the knowledge and reducing the segregation of facts. Because of this, the system might have the potential to bridge the gap between the two systems, and therefore make meaningful and effective rote memorisation possible, facilitating the needs stressed by both Karpicke (2012) and Zirkle and Ellis (2010) of more meaningful retrieval. Furthermore, by having the students memorise the concept map and graduately expanding on it, the generally experienced map shock occuring with expert-generated concept maps might also be mitigated (see also Tzeng, 2010).

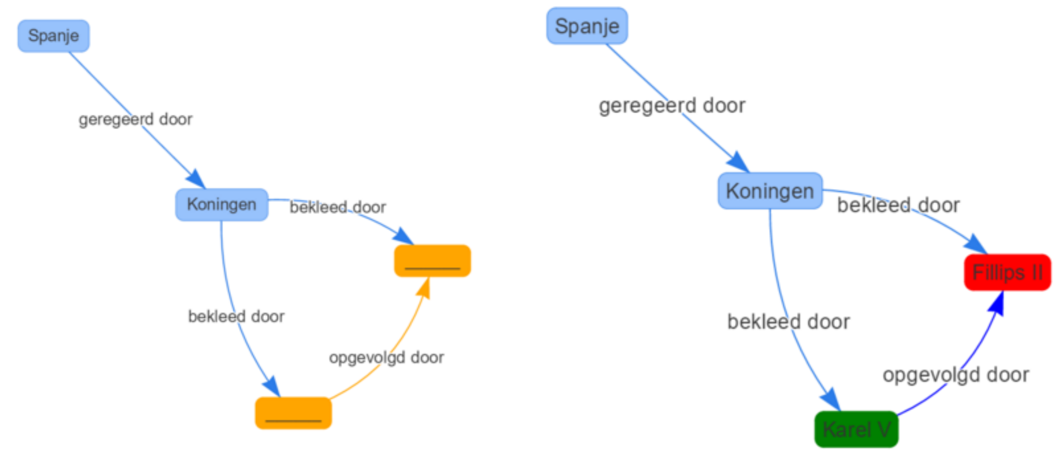


Figure 2: A display of the flashmap system, where the user has to think of the concepts fitting in the orange nodes on the left, and has to indicate which nodes were correct on the right

Evaluation

This project does not only aim to develop a flashmap system, but also to evaluate it by comparing it to a similarly functioning flashcard system. Since the retrieval practices have already been established to be more effective than concept maps (see the Comparison of the two tools section), the flashmap system is only evaluated in comparison to a similarly functioning flashcard system. The group approached for participating within the evaluation constists of Dutch high school teachers of the Stedelijk Lyceum has been found willing to participate, with their students using either the flashmap or the flashcard system for self study parallel with classroom instruction. The content of the instruction is Dutch literature during the sixteenth and seventeenth century, to be learned by the students for a school exam. For example, the students have to learn what the influence was of the Dutch War of Independence on the *Spaanschen Brabander* by Bredero. Because of the content existing mainly of concepts with meaningful relations it fits to the concept map technique, and thereby the flashmap system could be significantly beneficial over the flashcard system.

The research aims to investigate the following questions: Regarding high school students learning for Dutch literature using the flashmap system in comparison to them using the flashcard system...

Ia. ...is the learning gain larger?

Ib. ...is the learning gain larger controlled for the time spent on the system?

IIa. ...do they perceive the system to be more useful?

IIb. ...do they perceive the system to be easier to use?

Whether the flashmap system is more effective or efficient than the flashcard system is measured by the learning gain of high school the students, referring to the knowledge obtained by a student over the course of an instruction. Sequentially, the efficiency of the system is determined by the learning gain controlled for time spend on the system.

For measuring the affectiveness of the systems, the Technology Acceptance Model by Davis, Bagozzi, and Warshaw (1989) will be used (see figure 3). This model predicts the use of an information system by measuring the Perceived Usefulness and the Perceived Ease of Use of the user. These variables are mediators between External Variables and Attitude towards using,

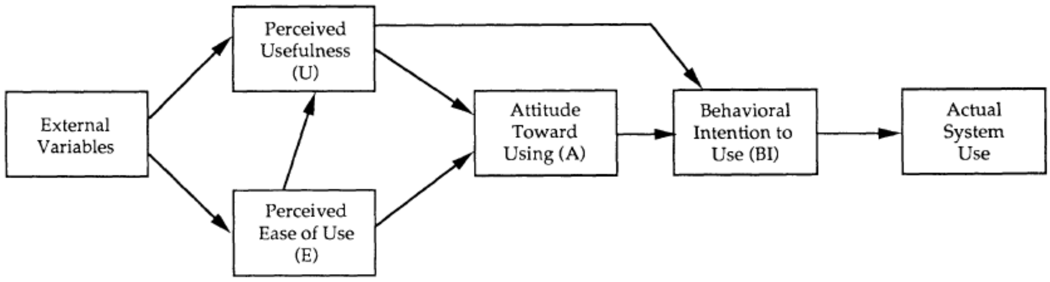


Figure 3: The Technology Acceptance Model by Davis et al. (1989)

leading to Behavioural intention to use, which in turn leads to the Actual system use.

Answering the research questions has both practical and scientific relevance. From a practical perspective, it has potential to overcome the criticism from various authors about flashcard systems and answer the need for meaningful rote memorisation. From a scientific perspective, it can provide new insights on the way students learn texts. Finally, it makes way for new research opportunities, for example what the difference in effect is of when the user develops their own concept maps or flashcards to be used within the system, or when they are created by the teacher or an expert.

The following chapter will elaborate further on the cognitive theories underlying concept mapping and flaschard systems on page 13, after which the design and development of the flashmap will be described in part II. Finally, the research conducted within this project and its results will be described in part III.

Cognitive theories

This chapter aims to explain the effectivity and inner workings of both concept mapping and flashcard systems by elaborating on the physiology of the relevant parts of the brain and the relevant cognitive theories, since these provide relevant information for the design of the flashcard and flashmap system. However, a general overview of the types of learning will be addressed, and the type of learning involved within this project, in order to provide focus on the specific cognitive theories relevant for the software design.

Types of learning

According to Squire (1987), there are multiple varieties of memory, which can mainly be categorised into declarative and nondeclarative knowledge, sometimes also referred to as respectively explicit and implicit knowledge (J. Anderson, 2015). Declarative knowledge also refers to memories that can be explicitly recalled, entailing facts such as definitions, paired associations etc., but also the events where these facts were acquired. Nondeclarative memory involves every memory which can be demonstrated in action, but not in conscious recall per se. Subcategories of these memories are procedural skills, priming, conditioning, and nonassociative memories. Because of the nature of this study, the cognitive theories discussed below are mainly focused on declarative knowledge, although most theories also are relevant to nondeclarative memory in some degree.

Furthermore, Smith and Ragan (2005) describes declarative knowledge as one of Gagné's types of learning outcomes, and relates declarative knowledge to Bloom's levels of recall and understanding, meaning that declarative knowledge does not only encompass rote memorisation of facts, but also understanding the meaning behind this fact. This is also in line with the essay written by von Glaserfield (2001) on radical constructivism, in which it is stated that whatever it is that students are to place into memory they should also understand. Another category of learning outcomes applicable to this context is that of intellectual skills, mainly that of concepts. These, according to Smith and Ragan (2005), help the learners simplify the world and can make them into more efficient thinkers. From a cognitive perspective however, there is not a great difference in dealing with declarative knowledge or concepts, because both relate to explicitly recallable memories and thereby can both be considered as being explicit (Squire, 1987).

Storage and retrieval

Although the whole brain is involved in storing memories, the most prominent areas facilitating the process of memorising are the frontal lobes and the hippocampus (J. Anderson, 2015) (see figure 4). The prefrontal regions are responsible for the creation and retrieval of memories, whereas the hippocampal and surrounding areas allow permanent storage of these memories. Because of this dynamic, Atkinson and Shiffrin (1968) conceived a modal theory of memory, displayed in

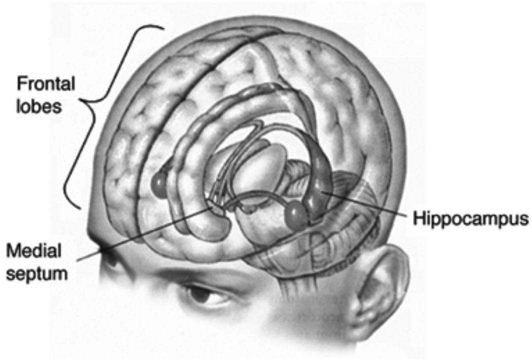


Figure 4: The brain areas mainly involved in storing and retrieving declarative knowledge (White, 2003)

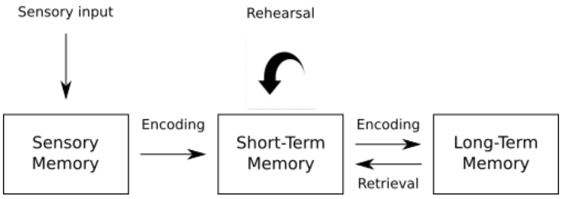


Figure 5: The modal model of memory proposed by Atkinson and Shiffrin (1968)

figure 5. In this model, information is perceived as sensory input, and is then shortly stored in the sensory memory. If the perceiver has paid enough attention to the input, it is then transfered (or encoded) into short-term memory. When the input is strong enough, that is, rehearsed often enough within short term memory, it can be more permanently stored in long-term memory. If not, the input fades away from memory and is forgotten. When a memory exists in long-term memory, it has to be retrieved into short-term memory in order to be remembered and used.

Karpicke (2012) describes two seperate learning practices based on the modal model of memory, namely encoding and retrieval practices, where encoding practices are focused on meaningful encoding or construction of knowledge, and retrieval practices are more focused on the reconstruction and rehearsal of knowledge. He states that both practices are essential to enhancing learning. The flashcard systems described in the Flashcard system on page 9 are a famous retrieval practice, which emphasise drilling the same pairs by association over and over again. Concept maps, described in the Concept mapping on page 7 are often regarded as a encoding practice, since the student has to connect diverse concepts within one topic by meaningful relations.

The following sections will elaborate on cognitive effects with regard to both encoding and retrieval practices, and relating them with their relevance to the effectiveness of concept mapping and flashcard systems respectively.

Cognitive effects with regard to encoding practices

The brain as an associative network

The brain is structured as an associative network, where neurons function as nodes, and synapses function as edges. When something has to be retrieved from memory, neurons signal relevant neighbouring neurons through the synapses in order to activate the relevant parts of the brain. More generally speaking, when stimulated with a retrieval cue, the brain can then use neural pathways to find a corresponding item in the brain. These networks are sometimes referred to as *semantic networks*, and the implication for retrieval as *spreading activation* (J. Anderson, 2015). This effect has also been found on a cognitive level, for example Kintsch, Welsch, Schmalhofer, and Zimny (1990) has found that material is often not literally encoded, but rather as a set of abstract meaning units representing certain associations between concepts.

Elaborative processing

Because information is retrieved in the brain via related nodes and edges in the semantic network, strong neural pathways facilitate the retrieval process. One way of creating these pathways is elaborative processing (Karpicke, 2012; J. Anderson, 2015), which focuses on meaningful processing of the content. Craik and Lockhart (1972) conducted an experiment where students were to freely recall from a list of words after the students had to train the words by one of the following techniques: answering questions about structural details (e.g. is it in capital letters); about phonemical details (e.g. the word rhyming on another word); whether the word fits into a certain category; and whether the word fits in a certain sentence. They found that the more meaningful the task was, the higher the retrieval rate was, which was later confirmed by Barclay, Bransford, Franks, McCarrell, and Nitsch (1974). Furthermore, research conducted by Nelson (1979) presented students with paired associates that where either semantic or phonetic (in this case rhymes), and students showed a significantly higher recall of semantic associates. These studies demonstrate the importance of meaningful processing for retention.

Implications for concept mapping

Reflecting on the previously described theory of associated networks, it appears that a semantic network is very similar in structure to concept maps, and thereby the maps provide an accurate representation of the way information is retrieved from the brain. For example, Cañas and Novak (2012) states that "the widespread use of concept maps is based on the notion that a concept map is a reflection of the builder's cognitive structure and thus portrays his or her understanding of the domain depicted in the map" (p. 1). Nesbit and Adesope (2006) speculate that because of this, more and better retrieval cues are created when learning from or generating a concept map. Furthermore, a concept map highlights the meaningful relations between the concepts, rather than just teaching the concepts themselves.

Cognitive effects with regard to retrieval practices

According to Karpicke (2012), a lot of educational practices have placed an emphasis on finding optimal ways to encode knowledge and experiences, but that retrieval practices have received less attention. Nevertheless, basic research has indicated that retrieval is still important to consider in any analysis of learning. This is mainly due to the fact that information is not stored exactly and indefinitely, but rather that memories are forgotten over time. Two theories explaining why forgetting occurs have been proposed and debated over, namely it occuring because of interference by other redundant memories, and it occuring because of decay of existing memories.

Interference and Decay

Theory of interference The theory of interference (sometimes also referred to as the *fan effect*) being responsible for forgetting has been demonstrated in an experiment by J. Anderson (1974). The participants were asked to memorise sentences in the form *A <person> is in the <location>*, where sometimes multiple persons where associated with only one location, and some locations with only one person. They found that if a sentence contained locations or persons with multiple associations this had an impact on the recognition time for that sentence, and even more so if both the location and the person had multiple associations. The explanation for this phenomenon is that since memories are retrieved by means of spreading activation and only limited activation can spread from one source (J. Anderson, 2015), the activity has to be divided over different branches in the semantic network, increasing the retrieval difficulty of the correct node.

Theory of decay The effect of decaying memories takes place in the connections between neurons, and therefore it is important to first examine how neurons communicate signals. Figure 6 displays a schematic representation of a neuron in which it can be seen how the soma (cell body) is connected via an axon to the dendritic tree of other cells. The neuron can transmit stimuli by creating an action potential in the nucleus, transmitting this signal through the axon to the terminal button in the connected telodendrion (in the image refered to as the terminal aborization). There, neurotransmitters are released from vesicles, and after they have crossed the synaptic cleft there is a certain chance of being received by postsynaptic receptors. When this is the case, the nucleus of the receiving cell is triggered via the connected dendrite to also create an action potential, and the whole process is repeated (Bliss & Collingridge, 1993). The strength of a certain connection between neurons is therefore dependent on the action potential generated by a nucleus, the amount of telodendria over which the action potential has to be distributed (hence the aforementioned fan effect), the amount of neurotransmitters in the terminal button, and the amount of postsynaptic receptors in the dendrite of the next neuron.

Long-term potentiation One widely studied effect with regard to the increase and decrease of action potential and strength of memory traces is called long-term potentiation (LTP) (J. Anderson, 2015; Bliss & Collingridge, 1993; Pavlik & Anderson, 2005; White, 2003). Whenever a neurotransmitter is received by by a receptor, not only is the next nucleus activated to release its action potential, but also more receptors are activated, so that the postsynaptic membrane is able to receive more neurotransmitters at the next activation. Furthermore, another process is activated altering the metabolical profile of the neuron, causing it to create proteins for more stable increased sensitivity towards stimuli. It is also speculated that there might be a retrograde effect, causing presynaptic modifications such as the creation of more neurotransmitters in the presynaptic vessicles (Bliss & Collingridge, 1993). This all results in an increased sensitivity in the postsynaptic neuron towards action potential in the presynaptic neuron, which then again increases the strength of this particular memory trace. Over time, if a specific neural pathway is not used, the effects of LTP decrease again, causing its strength to decrease and thereby causing decay. This also is a predictor for the *testing effect*, the effect of retrieval strengthening memory more than extra opportunities for further encoding, even when the retrieval is only carried out internally without any outward response (Edge et al., 2012).

Although both the effect of interference and decay have been proposed as separate theories and have been debated, they are not mutually exclusive, and J. Anderson (2015) therefore conludes that forgetting results both from decay and from interference.

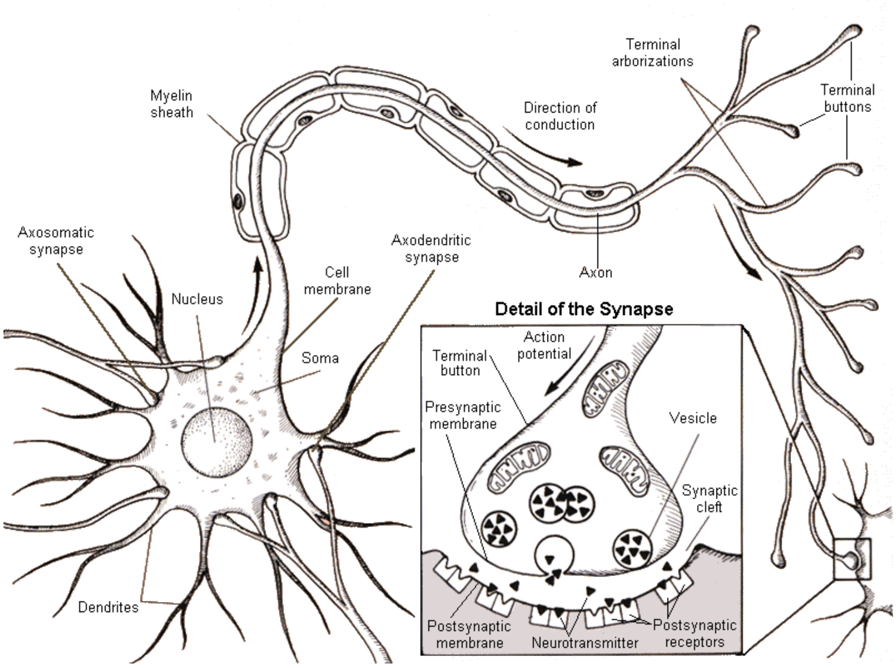
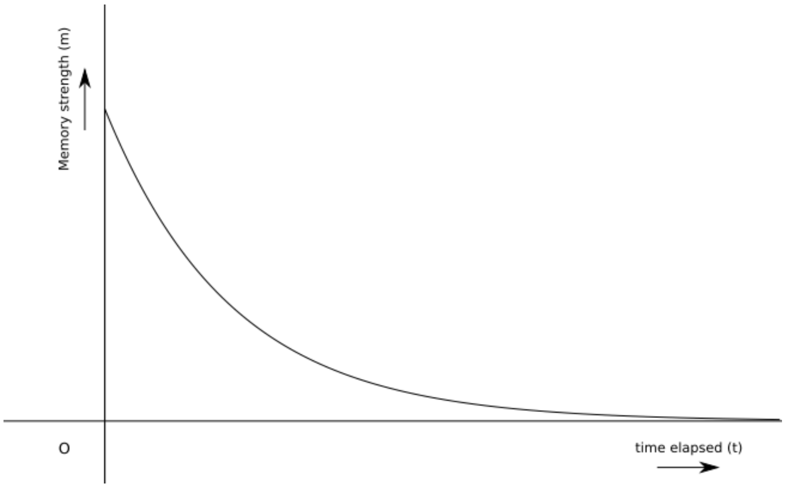


Figure 6: A schematic image of a neuron with a closeup of a synapse (Matsaridis, 2013)

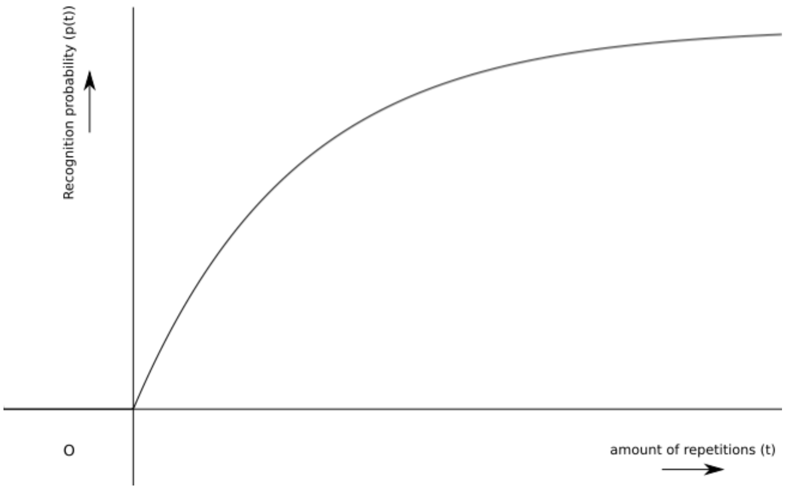
Power laws of forgetting and learning

Now that the relevant theories for learning and forgetting have been discussed, it is important to investigate with which rate people learn and forget. Already in 1885, Ebbinghaus discovered the power law of learning, referred to as the inversal exponential nature of forgetting (Edge et al., 2012; Pavlik & Anderson, 2005). The implication of this model is that memory not only systematically deteriorates with delay, but also that this loss is negatively accelerated, meaning that the rate of change gets smaller with increasing delay (J. Anderson, 2015). Wickelgren (1974) already proposed the formula $m = \lambda(1 + \beta t)^{-\psi}$, where m is memory strength (the probability of recognition), t is time, λ is the state of long-term memory at $t = 0$, ψ is the rate of forgetting, and β is the scaling parameter (see figure 7a). This formula has also found to be accurate by Wixted and Carpenter (2007). Finally, the effect has been directly related to LTP in the rat hippocampus by stimulating neural pathways directly with electrical signals (Raymond & Redman, 2006).

A similar effect has been found for the effectiveness of repetition: Newell and Rosenbloom (1981) have proposed a power law of learning, stating that a learning curve is inversal exponential (see also R. Anderson (2001) and Wixted and Ebessen (1991)). Murre and Chessa (2011) propose $P = p(t) = 1 - e^{-\mu_i t}$ as a function describing this power law, where P or p is the probability of recognition after t iterations and μ is the learning rate of student i (see figure 7b). The power law states that repetition has a positive effect on retrieval probability. This effect however does not increase linearly but inverse exponentially, with an asymptote at a certain amount of repetition. Again, this effect has also been demonstrated in the context of LTP in rat hippocampi (Barnes, 1979). The stronger memory trace from a higher repetition rate does not only result in a higher recall probability, but also in a more gradual retention curve, allowing memories to persist longer.



(a) The power law of forgetting, with m as the probability of recognition and t as the time passed since learning



(b) The power law of forgetting, with p(t) as the probability of recognition and t as the iterations of learning

Figure 7: The power laws of learning and forgetting

Spacing effect

The spacing effect is a well known effect occuring within paired-associate learning, and demonstrates that repeated items are better remembered when both occurences are separated by other events or items than when they are presented in immediate succession (Verkoeijen & Delaney, 2008; Logan, Castel, Haber, & Viehman, 2012; Siegel & Kahana, 2014; Xue et al., 2011; Karpicke & Blunt, 2011). This effect has been demonstrated with diverse populations (Verkoeijen & Delaney, 2008; Logan et al., 2012), under various learning conditions (Verkoeijen & Delaney, 2008; Logan et al., 2012), and in both explicit and implicit memory tasks (Verkoeijen & Delaney, 2008). Items in immediate succession are called massed items, and items in separated succession are called spaced items. However, Wahlheim, Maddox, and Jacoby (2014) adds to this that the spacing effect only takes place when a student detects the repetition of an item, and therefore the lag should not be too long.

Two theories have been presented explaining this phenomenon, namely the contextual variability theory and the study-phase retrieval theory (Siegel & Kahana, 2014). The first theory entails that because context is not static but continuous, and that therefore spaced items are studied in a greater variety of contexts and as such are easier to recall in yet other contexts than massed items due to the so-called encoding-specificity principle (J. Anderson, 2015). This principle entails that the probability of recalling an item depends on the similarity of the context during the encoding. The study-phase retrieval theory entails that additional retrieval cues for the repetition of an item are generated by earlier occurences and their associated contexts being associated with the repeated item. These theories are not mutually exclusive (Siegel & Kahana, 2014).

Inspired by the power laws of learning and forgetting, Karpicke and Bauernschmidt (2011) conducted an experiment to test for a relative spacing effect, which entails that the intervals between the repetition expand, remain constant, or contract. From their findings they confirmed the effect of absolute spacing, namely that longer gaps between items do have an effect on long-term retention, yet they did not find a relative spacing effect. However, this has not been tested for spacing with longer intervals, such as intervals spanning multiple days or weeks.

Implications for the flashcard system

It can be concluded that the flashcard system derives its effects mainly from the testing effect by having students actively retrieve information instead of simply encoding it, and from the spacing effect by students going through the items interspersally instead of by immediate succession. The key question however is how often a single card has to be repeated. On the one hand, overlearning can occur, where the student repeats an item too often resulting in diminished learning effects because of the power law of learning, and also only on the short term (Rohrer, Taylor, Pashler, Wixted, & Cepeda, 2005), which is inefficient. On the other hand, if the intervals are too long, students forget the items inbetween intervals, resulting in the spacing effect not applying anymore. In order to solve this problem, most modern digital flashcard systems apply a system called *adaptive spaced-repetition learning* (e.g. the Pimsleur system, the Leitner system, Supermemo, and Anki (Edge et al., 2012)). In this system, exponentially expanding intervals are used, not because of a relative spacing effect, but rather to increase the average (absolute) spacing with each new repetition. This creates a stronger memory trace every time, but also takes into account the further decreasing risk of forgetting because of the slower declining retention curve (see figure 8).

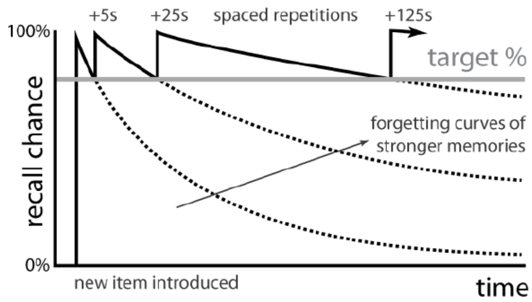


Figure 8: Adaptive spaced-repetition learning (taken from Edge et al. (2012))

Conclusion

Overall, this chapter has discussed several cognitive theories related to the storage and retrieval of explicit (or declarative) knowledge in and from the hippocampus. Related to encoding practices, it has now been established that the brain works as an associative or semantic network, and that meaningful or elaborative processing is important for the later retrieval of memories. This seems to fit with the structure and process of concept mapping, although more research is needed in this area. Furthermore, the theories of interference and decay have been discussed in order to explain forgetting of memories, together with Long-Term Potentiation and its effects on the rate of forgetting and learning. In addition, articles were discussed demonstrating that spaced rehearsal is more effective than massed rehearsal. This has finally led to the conclusion that adaptive spaced-repetition learning is an effective method to expand absolute spacing, which entails that items are repeated with exponentially increasing intervals.

Part II

Design Report

The Project Description mainly described the needs which the Flashmap System might be able to accomodate, and on page 10 generic features of such a system is described. Although the term Flashmap System is intended for describing any system including these features, when having to evaluate the idea one has to evaluate one or multiple specific implementations of that idea. Therefore this part will specify the design features of the specific tool developed within this project, along with arguments in favour of and against these choices and their considerations, and the process with which they are incorporated within the tool itself. This description will follow the different steps of the Generic Model (Plomp, Feteris, & Pieters, 1992), which is displayed in figure 9.

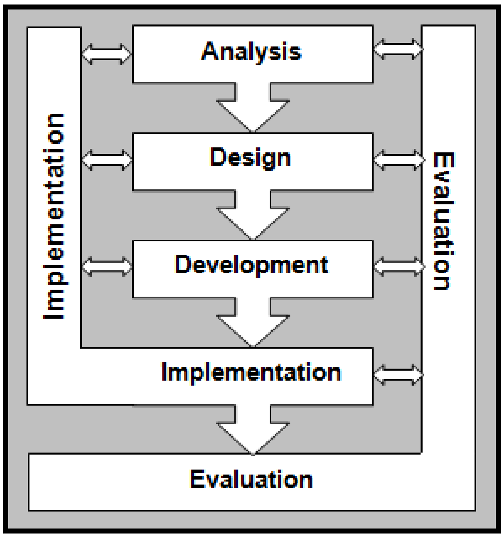


Figure 9: The generic model by Plomp et al. (1992)

The first step consists of analysing the context, learner and task characteristics in order derive their possibilities and requirements. Based on these together with general design guidelines provided by educational literature, design choices can be made and argued for within the next step. When these choices have been made, they can be developed and incorporated within the product. When the product then is finished it can be implemented within the context and evaluated. As can be seen in the model, the implementation and evaluation play an important part throughout the design process, rather than them being seperate steps at the end, and will therefore already be addressed during the description of the first three steps. Furthermore, these seperate steps will be described in the next part, which describes the research conducted within this project. Finally, rather than addressing the design and development in seperate chapters, they will be described alongside each other within the chapters describing the separate facets of the product. INSERT SEPARATE FACETS OF THE PRODUCT

Analyses

Before designing an educational product, it is important that the designer first acquaints himself with the extrinsic factors important to this product. In order to discover the important characteristics of these factors, Smith and Ragan (2005) enlist three types of analyses to be conducted, together with steps for conducting them. These are an analysis of the context, of the learner, and of the learning task. Although these analyses are more targeted towards instructional design, and therefore more focused on a specific group being taught specific content, these analyses still provide relevant information for the design choices and the evaluation. However, the steps are adjusted and generalised or even omitted in order to fit the design of the more generic learning tool. The information gathered in order to conduct these analyses mainly stems from meetings with one of the teachers. This might not be the most reliable source of information because of the lack of triangulation, and should therefore not be taken as insight in the curriculum of Dutch Literature courses in secondary education, but rather as context information relevant to the design. The most important findings are described within this chapter, along with their implications for the design of the software.

Analysis of the learning context

As already stated in the Evaluation section on page 11, the evaluation of the flashmap system will be evaluated within the Dutch secondary school Stedelijk Lyceum, with students having to learn about the Renaissance genres in Dutch Literature. Although the general needs for a flashmap system are shortly provided within the Evaluation on page 11, it is still important to investigate the specific needs of the context where the programm will be implemented. Therefore, this context will be further investigated within this section, starting with the Needs Assessment (Smith & Ragan, 2005).

Needs assessment

There are multiple reason why teachers think it is important to learn about Dutch renaissance literature: one could argue that this way the knowledge is passed onto a new generation, keeping it relevant; understanding the history of literature is important for understanding modern literature; and literature can provide certain insights to the individual reading it (Slings, 2007; Dirksen, 2007). Furthermore, the school is extrinsically motivated to teach the subject matter, since subdomain E2 and E3 within the Dutch national exam programm state that a student has to recognise and distinct between literary text genres, apply literary concepts in the interpretation of literary work, provide the outlines of the literature history, and place literary works in this historical perspective.

The teacher did confirm the need for better retention and comprehension of the content already stated within the Project Description, since she indicated that most of the time the students only learned the night before the exam in order to get a high (enough) grade and consequently forget everything again. Both using a flashcard system and the flashmap system should accomodate this need. Furthermore, she deemed them to become more familiar with the Dutch Renaissance writers or work to be the most important, entailing that students recognise impotant names or that they can distinguish between different genres. Based on these statements, the goals within the context are in line with students memorising and understanding all of the facts, without them being too ambitious. Finally, the teacher provided a test from the previous year to offer some more concrete examples of what she wanted the students to know, of which an English translation is included in the appendix on page 88. From this test, more goals can be extrapolated, such as students having to not only distinguish different genres, but also having to define them or provide characteristics, and recognise the application of these features in both examples of the time periods as well as modern examples. Furthermore, they have to be able to relate the famous writers and writings to the genres.

The learning environment

The Stedelijk Lyceum is an open denominated school organisation, consisting of 7 schools on different locations. The school approached within this project has been approved by the Dutch Inspection of Education (*Kwaliteitsonderzoek in het kader van het onderwijsverslag 2016, Het Stedelijk Lyceum - locatie Kottenpark, HAVO, VWO*, 2015). The course on Dutch renaissance literature consists of two different types of learning activities, which are classroom instruction, and individual learning at home by the students. There are two sessions of classroom instruction, both lasting 50 minutes, in which the 100 students are divided over the three teachers in static groups on separate locations. These lessons take place over the course of two weeks, with one lesson provided in one week. Within these lessons, the teachers transfer knowledge and provide excersises for the students. Outside of the lessons, the students still have to study the textbook Laagland individually (van der Meulen & Kraaijeveld, 2010), which contains all of the materials which will be prompted on a final written assessment. As already stated before, the teacher indicated this activity mostly to take place on the evening before the assessment, and only on a superficial level. Finally, this assessment takes place in the second week after the final instruction, and will be similar to the example test included in the appendix on page 88.

The teacher stated that the course mainly consisted of the rote memorisation of facts, and that she was still doubtful whether the students would actually be willing to participate in the evaluation of the Flashmap system. Yet, she did see the general use of the tool for achieving the learning goals, and therefore still seemed to be enthusiastic in cooperation and encouraging the students to participate. The only two technical problems are that there is not too much time for extra activities within the lesson plan and the teachers being quite busy themselves, and that the technological possibilities within the classroom are limited. Within the classroom, only a couple of computers are available for use, and still run relatively old software. Therefore, the activities envolved in using the flashmap have to target the individual learning of students, since they have more time outside of the lesson plan, and mostly do possess the hardware and software necessary to run the software.

Analysis of the learner

Physiological characteristics

The physiological characteristics of the respondent's brain provide important implications for the design of the software. They are enrolled in grade 4 of Dutch secondary education, and therefore should be around the age of 16-17, with some deviations due to students either having skipped or repeated a grade. Therefore, the students are generally considered to be either at the end of puberty, or the beginning of young adolescence. The Cognitive theories chapter on page 13 already provides general theories about the learning process within the brain. However, during late puberty and early adolesence, the brain is still heavily in development, especially the prefrontal cortex. (Blakemore, Burnett, & Dahl, 2010). In order to map out the changes in the adolescent brain, Giedd et al. (1999) performed a longitudinal MRI study of the brain development during this period, where three themes emerged within the adolescent development of the brain:

1. After a peak in growth of both brain cells, connections and neurotransmitters during childhood, one can see a decline in adolescence;

2. The connectivity between different regions of the brain increases;

3. A new balance is formed among frontal and limbic lobes.

The first theme is a result for the brain becoming more streamlined after having collected a lot of information during late childhood, making it more efficient (see also Interference and Decay on page 16). This is also known as peak plasticity, after which a decrease can be observed. Powell (2006) describes this phenomenon as *Use it or lose it*, since the brain rigorously selects the specific memories which are activated during this time. The second theme refers to the strengthening of specific memories, which are enhanced during that period. Here the flashcard system proves to be a useful tool, since it focuses on repeating specific associated pairs that the learner wants to remember.

Finally, during adolescence a shift is made from “cold” to “hot” cognition, where the former relates to hypothetical, low-emotion reactions, and the latter to high arousal decision making, strongly influenced by peer pressure and real, direct consequences. This is highly related to the prefrontal cortex being heavily developed, resulting in the teenage brain to rely more on the amygdala which is the more emotional, impulsive area of the brain. This means that for students to be motivated to learn, they either need a strong intrinsic motivator, or they have to rely on what Powell (2006) describes as an “external prefrontal cortex”, which can be either a reward or a person reminding them to study (e.g. the teacher or a parent). Therefore, extrinsic factors such as the usefulness of the system to passing the school test, a voucher for icecream, and the teacher are used to motivate the students to use the system.

Cognitive characteristics

All students should have learned about the relevant time period in their history classes prior to this course (e.g. the Spanish War, the Lutheran reformation etc.), providing the relevant knowledge to understand the context of Dutch renaissance literature. Additionally, the students have received a similar instruction on Dutch medieval literature, which is also relvant for concepts in the renaissance literature, such as the *Mecenas*, the *Lyriek* and *Rederijkers*. Therefore, these concepts form the root concepts from which to start within the concept map.

However, one difference between students is whether they chose technical or society-related subjects. This makes up for different specific aptitudes within this specific Dutch literature course. Furthermore, some of the students are also enrolled in classical subjects, and because the Dutch renaissance literature has a lot of connections with classical genres, these students might have an advantage in prior knowledge. Both technical and society-oriented profiles, and classical and non-classical profiles are therefore accommodated for within the concept map.

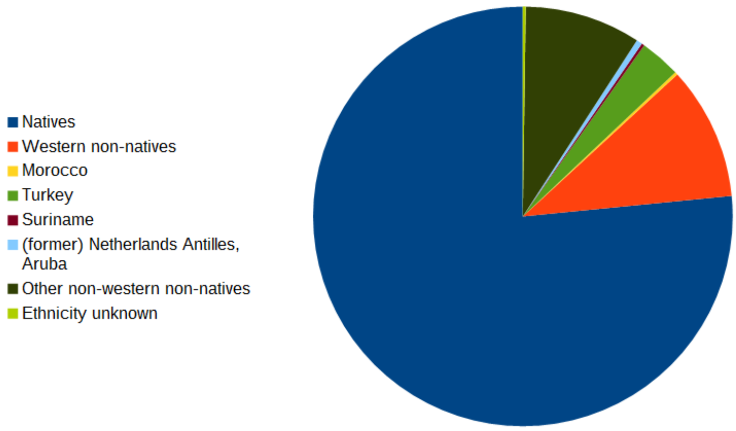


Figure 10: The distribution of ethnicities among 16-17 year old vwo students of education type vwo3-6 (*Leerlingen, deelnemers en studenten; onderwijssoort, woonregio*, 2016)

Social characteristics

Among other data, the CBS offers descriptives of students, categorised per national district. This descriptive data entails information about the age, sex, type of education, and ethnicity of students, and the interactions among these variables (*Leerlingen, deelnemers en studenten; onderwijssoort, woonregio*, 2016). From this data, descriptives about 16-17 year old students from Enschede enrolled in vwo grade 3-6 was extracted, displaying their age and ethnicities. 23% of the 16-17 year old students are enrolled in VWO, where 44% is male and 56% is female. The distribution of ethnicity is visualised in figure 10. 77% of the students are native and 31% are non-native. 9% of the students is western non-native, and 22% is non-western non-native. The CBS defines a non-western non-native is an non-native originating from Afrika, Latin-Amerika and Asia (except for Indonesia and Japan) or Turkey. The most prevalent non-western ethnicity is Turkish, with 8%.

Grever, Pelzer, and Haydn (2011) provide information on the perspectives on learning history by Dutch, English and French high school students. Within this study, students were asked several questions about what kinds of history, which periods of history are important or interesting for the students, and what the meaning of history is for their personal lives and what they believe to be its relevance for society. For Dutch students, this study found that the history of ones own family generally ranks high, and after that the history of the country where the parents come from (both for natives and non-natives). This means that native students might be more interested in learning about the subject than non-native students. Furthermore, the history of ones own religion is mostly important for Moroccan and Turkish students (which are mostly muslim), so the history of christianity is generally not that interesting towards most students. The study also found that the time period of early modern history is the least interesting for students, no matter the gender or nationality, despite that in the Netherlands the most important topic is the rise of the Dutch republic and the Golden Age (the content of the subject used within this study). Finally, the study states that there were no significant differences in perceptions of pre-vocational students and HAVO/VWO students in these respects, although one might expect Gymnasium students to be more interested in the classical revival of art during the renaissance than the Atheneum students.

Finally, the CBS also offers statistics about religious denominations (*Religieuze betrokkenheid;*

kerkelijke gezindte; regio, 2015), which state that 57% of the people in the province of Overijssel is affiliated to a church. These affiliations are split up in different religions: 22% Roman-Catholic, 8% Protestant, 12% Dutch Reformed, 7% Continental Reformed, 4% Islam, and 5% miscellaneous (see figure 11a. 43% is not affiliated to any church, however this does not necessarily entail that they do not have a religious worldview. Data is also offered on how frequent people visit the church: 14% visits every week or more often, 4% two or three times a month, 4% once a month, 8% less than once a month, and finally 70% (almost) never (see figure 11b. This would indicate that although there is a majority affiliated with a certain church, most of the people do not actively take part in their respective community. There are also more specific statistics available about the region of Twente only (*Religie; naar regio; 2000/2002 of 2003*, 2004), however these are older and might already have changed significantly over the last 13 years. Yet, they state that Twente is more religious than the overal province of Overijssel. Unfortunately, there are no statistics available about Enschede only. Finally, the school of the target group has an open (i.e. non-religious) denomination, whereas the other large school in Enschede has a christian denomination, so one might expect mainly the students without any strong religious views to choose for this school. Still, since christianity is generally prevalent within the region, students are highly likely to be familiar with the christian themes relevant to understanding the literature from the renaissance period.

Analysis of the task

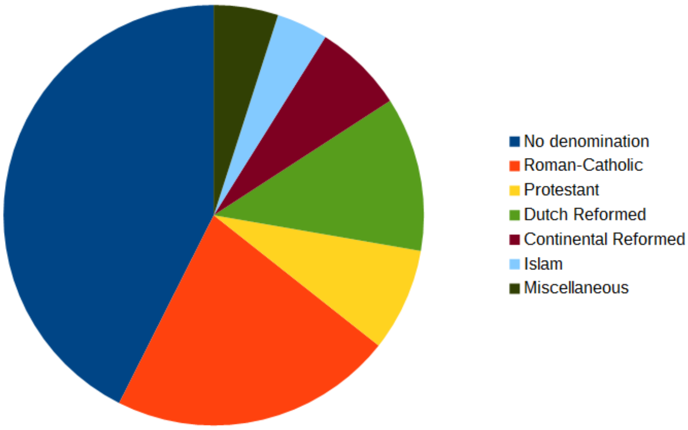
Finally, the characteristics of the task itself will be investigated in order to learn how to design a tool in such a way that it facilitates or augments the learning process. Smith and Ragan (2005) enlist primary steps for performing a learning task analysis, which are writing a learning goal, determining the types of learning of the goal, conducting an information-processing analysis of that goal, conducting a prerequisite analysis and determining the type of learning of the prerequisites, writing the learning objectives for the learning goal and each of the prerequisites, and writing the test specifications. However, within this project the instruction has already been written (van der Meulen & Kraaijeveld, 2010), and only has to be used to create a concept map. Still, knowledge of the underlying structure of the instruction might prove to be helpful for finding the relevant elements, and it is also useful to investigate the specific uses of the instruction within the context of this project.

Learning goals

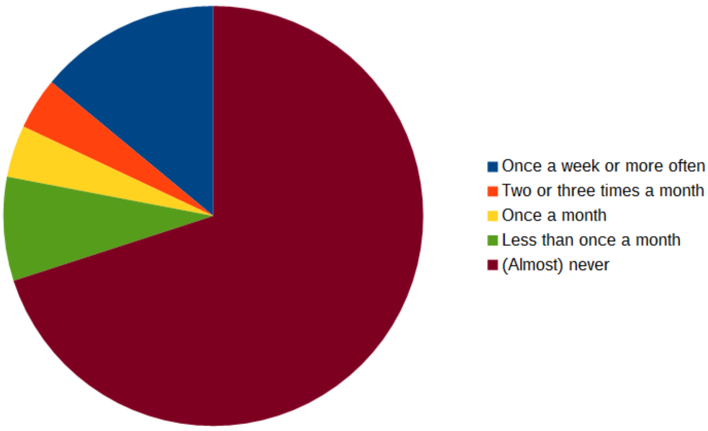
The direct learning goals of the instruction can be found in paragraph 13.4 of Laagland, the specific instruction for the Dutch renaissance literature, where the previous paragraphs only provide the prerequisite knowledge necessary to understand this paragraph. The different chapters describe the *emblematiek*, the *lyriek*, the sonnet, and the different theatrical genres (the tragedy, the comedy, and the *klucht*). One of the goals of this instruction is that the students are able to describe these genres, and are able to differentiate between the subgenres or terminology within these genres. However, the students also have to be able to relate these genres to the general context described in the previous chapters, which consist out of the political, the socioeconomic, and the cultural backgrounds.

Types of learning

Attaining these skills are mainly intellectual in the typology defined by Smith and Ragan (2005), because the students mainly have to be able to describe and discriminate between defined



(a) A distribution of church affiliations



(b) The distribution of frequencies of church visits within the group of people affiliated to a church

Figure 11: Denominations within the region of Twente (*Religieuze betrokkenheid; kerkelijke gezindte; regio*, 2015)

concepts. However, there is also a certain amount of declarative knowledge learning involved because students have to first learn and memorise certain definitions or conceptual organisations. Furthermore, within the book there are not only abstract concepts being defined, but also declarative knowledge such as names of important authors (Vondel, Bredero etc.), books or plays (e.g. the *Klucht van de koe*), and certain historic events such as the migration of calvinists from Antwerpen to Amsterdam in 1585.

Concept map

Because the information has already been defined within the textbook (both the new content and prerequisite content), the information-processing and prerequisite analysis activities have been replaced by translating the content of the instruction within the textbook to a concept map. Within this map, not only the relevant concepts, names and events are presented, but also the relations between them, providing a more meaningful representation. Furthermore, the concept map also contains information about the order in which the concepts have to be learned, because of the direction of the relations. The data used for the concept map is uploaded on github[1]. A direct visualisation is too extensive to be feasibly included within this report, however a digital visualisation is available[2] (after a short initial rendering time due to its size). The Client design and development chapter on page 57 will elaborate further on the design choices for the concept map. Finally, this map is directly shown to the students within the flashmap condition during the experiment.

Flashcards

The activity of specifying the learning objectives is replaced by formulating the flashcards, because the flashcards already form the specific knowledge-based learning objectives. They already contain the most important types of information which should be included in an objective, namely the statement of the terminal behaviour (the answer itself), the conditions of demonstration (given this question, the student can reproduce the correct associated answer). The standards or criteria for these objectives are globally defined, namely that the student has to be able to demonstrate that he knows the correct concept corresponding to a parent node and edge label. The flashcards are directly based on the previously defined concept map. Within this activity, edges and their corresponding parent nodes were transformed to a question, and the child node formed the answer to that question. For example, the nodes *Strijdliteratuur* and *Actualiteit*, respectively connected by the edge *verwees naar*, is translated to a flashcard "Q: Waar verwees de Strijdliteratuur naar?" → "A: Naar de actualiteit" (*Translation:* To which did the war literature refer? To actual events). Sometimes, multiple edges from one node to several child nodes having the same label or falling within the same category were translated to only one single flashcard. The data for the flashcards can be found again on github[3].

Test specifications

The assessments conducted before and after the students have used the learning tool consist partly out of the questions from the flashcards for measuring knowledge reproduction, but also partly of questions targeted to measure the comprehension levels of the students (see Bloom et al., 1956). On both assessments for all questions, the students are asked to fill in an answer in a textbox.

[1] `https://github.com/mcvdenk/MasterThesis-Software/blob/master/database/concept_map.json`
[2] `http://www.mvdenk.com/thesis/concept_map/`
[3] `https://github.com/mcvdenk/MasterThesis-Software/blob/master/database/flashcards.json`

In order to answer the questions for comprehension, a student has to be able to draw relations between not directly linked nodes, and thereby requires a higher degree of mastery of the content. It does however not yet contain any questions where students have to apply the content within different context, or have to think outside of the content directly taught, since these questions would rate on even higher levels on the taxonomy of Bloom. Finally, the questions are phrased according to the specified action verbs related to the level of learning. A more detailed elaboration of the test construction and analysis can be found in the Instrumentation section on page 65, and all of the comprehension level questions are included on github[4].

[4] `https://github.com/mcvdenk/MasterThesis-Software/blob/master/database/itembank.json`

Defining the general use cases

Supplantive or generative

The first important design choice which has to be made is whether the students are supplied with a concept map or flashcards, or that they generate the content themselves. The dichotomy of generative versus supplantive instruction is described in further detail by Smith and Ragan (2005), where the implications of both sides are enlisted for the learner, the task and the context.

One of the aspects of generative strategies is that the learner requires a higher amount of prior knowledge, a higher aptitude, and a wider and more flexible range of cognitive strategies, because the content still has to be (partly) researched and constructed. This can be a disadvantage, because the learner might not possess these skills and therefore the instruction may not be suitable or highly inefficient using generative strategies. On the other hand, greater mental effort generally leads to greater depth of processing and therefore better, more meaningful learning, which was also stressed by Cañas and Novak (2012) and Nesbit and Adesope (2006). Furthermore, learners experience a higher motivation and a lower amount of anxiety when using generative strategies, and their attribution of success is internal rather than external.

Furthermore, when using more generative strategies, the learning task becomes more complex and ill-structured, and therefore requires more instruction and time to complete. It also leads to a higher focus on cognitive strategies, but less so on the learning goals. These goals can also not become universal, since each student creates their own flashcards or concept map, and therefore decides on their own learning content.

The most important factor for this design choice is feasibility. The teacher already stated that there is only limited time available during the lesson to introduce them to the software, so there is no time for extensive instruction on how to create concept maps, let alone creating the maps within the classroom. Additionally, students do not have much time at home to spend on creating the maps, and it is also known from both interviews with the teacher as literature that they will probably have only a low amount of intrinsic motivation. Finally, when the students have to create their own maps, it cannot be guaranted that they will include the nodes relevant for the goals of the instruction, and might become either to narrow or to extensive in certain branches. The same arguments are valid for letting students create their own flashcards. Therefore, despite of the benefits that a more generative approach may have for the learning process, the content will be supplied to the students instead.

Choice of platform

The next design decision is centered around the choice of platform or medium going to be used in order to support the learning tool. In the section **??** on page **??** it is described that students generally prefer to use traditional or written flashcards, despite the many advantages of digital flashcards. However, with the flashmap tool this is not a feasible option, since the tool has to dynamically generate different graphs based on the general concept map and the profile of the students. Of course it would be possible to provide the concept map digitally and the flashcards in written form, however this would introduce an extra variable to the research design. Finally, with written flashcards one can only use rather crude methods for rescheduling the cards, instead of using the more precise algorithms possible within a digital tool.

There are various options for the specific implementation, for example a computer program or an app. Of these options, a web application is the most convenient, since it is accessible for any device with a modern web browser, and immediately stores the usage data on a centralised server so that it is immediately accessible for the researcher. Furthermore, adjustments or fixes can be applied during the research, without all users having to update to the newest version.

For the client, HTML, CSS, and Javascript are used, importing the vis.js library for visualising the concept map dynamically[5]. Furthermore, Python is used for the server logic, communicating with the webclient through a websocket using JSON messages. The choice for Python is mainly based on preference by the programmer. Finally, MongoDB was used as a database engine since it stores data in a format very similar to JSON, which is also used by vis.js to represent concept maps.

The server implementation will be further elaborated in the Server design and development chapter on page 47, and the client implementation in the Client design and development chapter on page 57.

Supported user actions

The final design decision related to the general ideation of software is deciding which use cases should be supported, which are generally displayed within a UML use case diagram ((OMG), 2015). For the flashmap software, the use cases are divided in cases related to the registering and login process (see figure 12), and the cases related to the main use of the software (see figure 13).

Login use cases When opening the webapplication, the user is first prompted with a login screen. Here, th e user can either enter an already existing username to continue this session, or he can enter a new name in order to register as a new user. When the user is registering as a new user, a form is presented asking for information on gender and birthdate as descriptive information, and asking for the specific code the user received on the informed consent form in order to validate that the user indeed signed this form before partaking in the research (see section Procedure on page 63). After that, another form will be prompted for the pretest (section Instrumentation on page 65). When the user has met certain criteria, a posttest similar to the pretest will be prompted, followed by a questionnaire and a debriefing text. When none of these criteria are met, the user can access he main use cases.

Learning use cases The main use cases entail requesting items for review, requesting the learning progress, or logging out. When requesting items for review, the user can receive a due or new flashcard or flashmap, depending whether there are any old items due for review and the experimental group the user is in. Alternatively, the user can also be prompted whether a certain section of the instruction material has been read, since the rehearsal of items cannot be meaningful when the user is not familiar with the content. These prompts take often place at the beginning of a session so that the user does not have to interrupt a session. Furthermore, they prompt two sections ahead of the material currently being learned or reviewed by the user from the flashmap

[5] `http://www.visjs.org/`

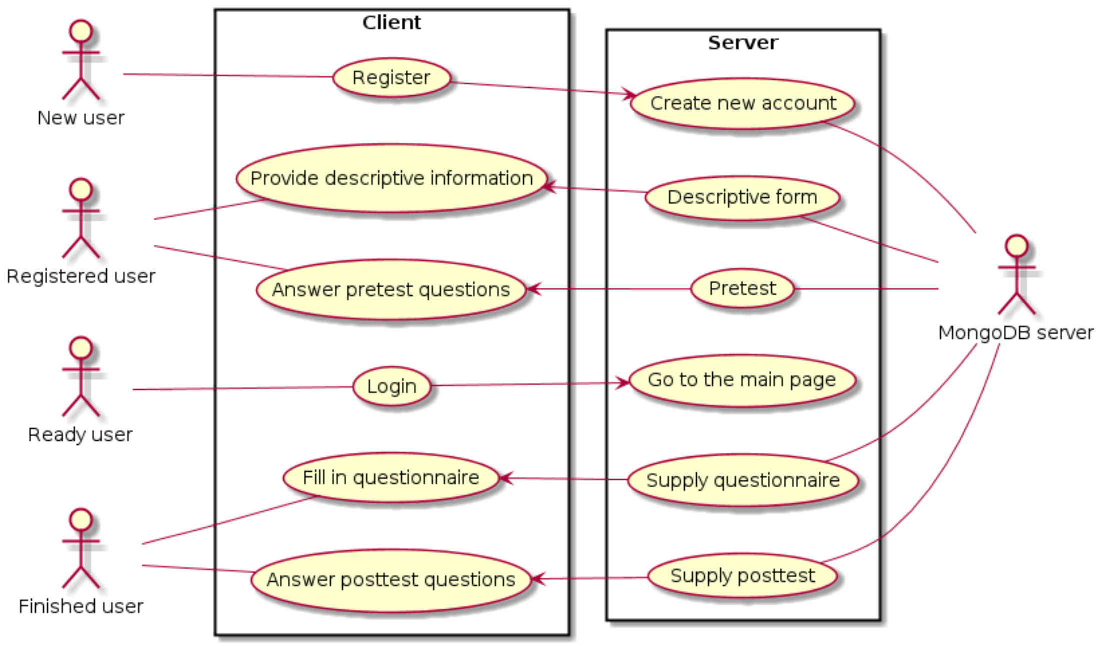


Figure 12: An UML use case diagram for registering as new users or logging in as existing users

or flashcards in order to guarantee that the user is familiar with the material before learning the items. The user is prompted to read a section at most once per day. After the user has submitted a response, he can undo this response if he is not content with it. For example, the user could after seeing the correct response decide that he thought of a similar enough answer, but after deeper reflection still decide that his answer was not sufficient. In this case, he could use the undo option in order to be presented with the previous response again and select the 'incorrect' option.

Learning progress When requesting the learning progress, the user is presented of an overview of what has already been learned and what is still left as either unseen items or items due for review. This provides an indication for the user so that he can estimate how much time he still needs to invest into the software, but also could stimulate the user by seeing the number of new or due items lowering and learned items increasing.

Finally, the user can return to the login screen by logging out.

Detailed description of the client server interaction

Based on the previous description of use cases, there are two sets of complex interaction between the client and server, which are again the interaction for the login and registering process, and the interaction for the learning process. These are described as UML activity diagrams ((OMG), 2015) in figure 14 and figure 15. These diagrams are elaborated on below together with the specific network messages belonging to the interaction step. Each network message is a simple JSON message consisting of a *keyword* field — containing the main function which has to be performed by the other party — and a *data* field containing a dictionary with necessary supplementary data.

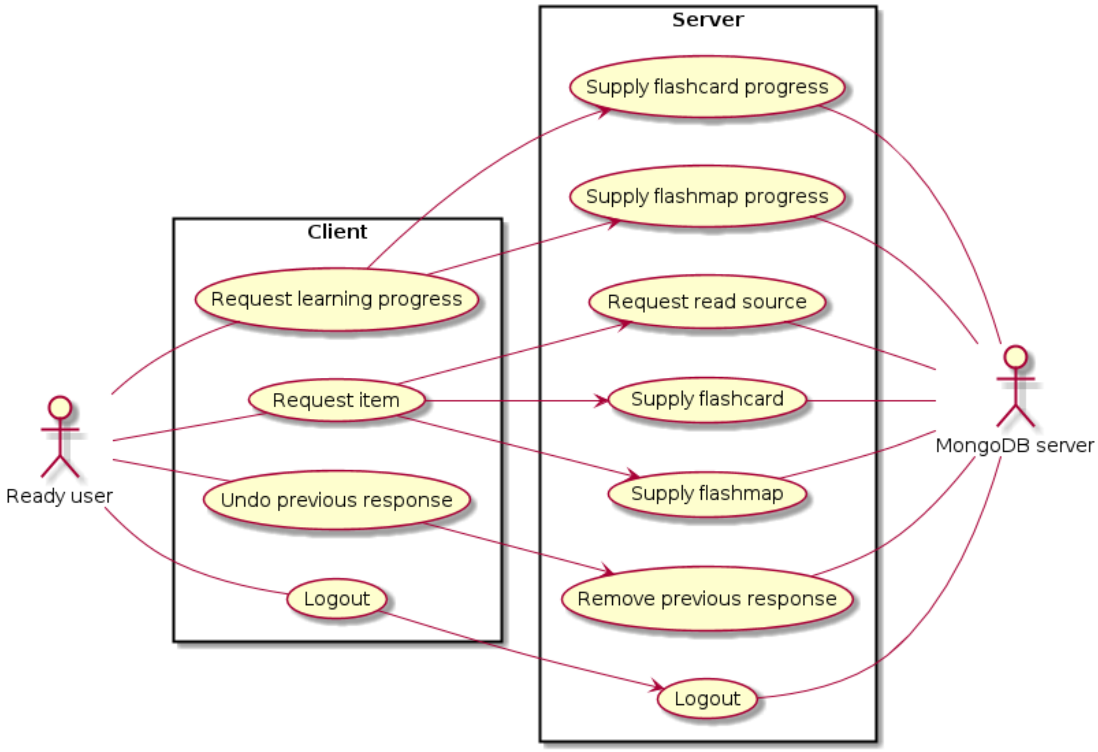


Figure 13: An UML use case diagram for main uses of the application

Login activities

The exact reasoning behind the different activities can be found in the Procedure and Instrumentation sections of the Methods chapter on pages 63 and 65.

Authenticate When the user logs in, the client sends a message with keyword "AUTHENTICATE-REQUEST" and data containing a name field with the username. When a user with this username does not exist yet, the server creates a new user with a randomly assigned condition (either flashcard or flashmap, also known as control or experimental). When this user already exists, the server fetches this user from the database.

Descriptives The server then checks whether the user already has set description fields. If not, the server returns a message with keyword "DESCRIPTIVES-REQUEST", on which the client responds with keyword "DESCRIPTIVES-RESPONSE" with the data fields gender, birthdate, and code.

Pretest When the previous condition is met, the server will check whether the user has a registered pretest. If this is not the case, it will create a new test by randomly selecting 5 items from the flashcard dataset and 5 items from the itembank, which it will then send to the user with the keyword "TEST-REQUEST". After the user has answered the questions, the client sends the responses to the server with the keyword "TEST-RESPONSE".

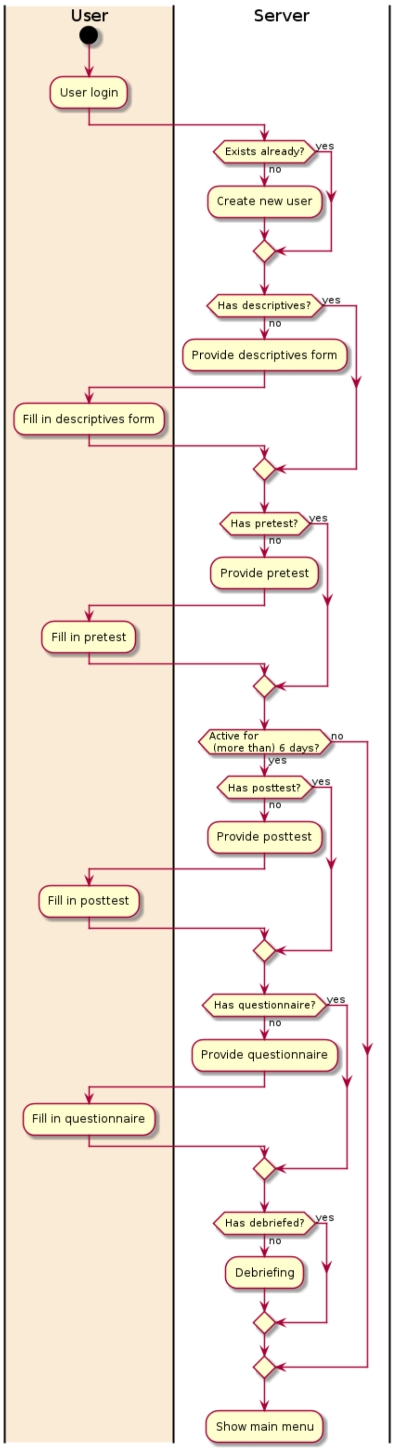


Figure 14: An UML activity diagram displaying the server-client interactions when a user logs in

After the experiment If the previous condition is also met, the user will be directed towards the main application with an "AUTHENTICATE-RESPONSE" message from the server, unless he has used the software for at least 15 minutes on 6 days. In that case the checks described below will be performed.

Posttest In this step the server checks whether the user also has a posttest entry. When this is not the case, it will send a similar test message to the pretest message, with the exception that the flashcards and test items from the pretest are excluded from the random selection in the posttest.

Questionnaire Sequentially, if the user has no questionnaire entry, the server will construct a new questionnaire by shuffling the Perceived Usefulness item, randomly selecting for each item whether it is positively phrased or negatively, copying this item set but with the opposite phrasing, and finally shuffeling the second set. The same is done for the Perceived Ease of Use items. This questionnaire is then sent to the client with the "QUESTIONNAIRE-REQUEST" keyword. The client sends a filled in version back with the "QUESTIONNAIRE-RESPONSE" keyword to the server with an extra textfield for what was good about the software, what could improved about the software, and an (optional) emailadress of the user for an interview at a later time.

Debriefing Finally, if the user has not debriefed before, the server sends a message with the keyword "DEBRIEFING-REQUEST" to the client, which will show a debriefing message to the user and returning a message with the keyword "DEBRIEFING-RESPONSE". When all the checks are met, the user will be directed to the main application with the "AUTHENTICATE-RESPONSE" message.

Learning activities

In the main application view, the user can either review items or view his learning progress. If he chooses the latter, a message will be sent to the server with the "LEARNED_ITEMS-REQUEST" keyword, to which the server will respond with a "LEARNED_ITEMS-RESPONSE" message containing information on the learning progress (see the Learning progress section on page 60). If the user wants to review items, the client will send a "LEARN-REQUEST" message to the server and the process described below is performed.

Aimed time reached First, the server checks whether the user already spent 15 minutes learning today. If this is the case, the client will display a message that the user has spent an sufficient amount of time on learning for today. This will not be directly be displayed as the activity diagram suggests, but rather it will show this message together with the next item.

Selecting an item After this, the server will check whether there is any item already due for review. If this is the case, the server will sent the item which is due for the longest time to the client with the keyword "LEARN-RESPONSE". If not, the server selects a new item from the database. It is then checked whether the user already read the section in the book related to this item. If not, the server sends a "READ_SOURCE-REQUEST" to the client, which prompts the user whether he has read the source supplied in the source field of the message. If so, the client sends a "READ_SOURCE-RESPONSE" message back to the server, which adds the supplied source to the list of read sources for the user. When the user has read the section , the server will sent a "LEARN-RESPONSE" message with a new item from the database. If there is no new item left, the server sends a message to the client with the keyword "NO_MORE_INSTANCES".

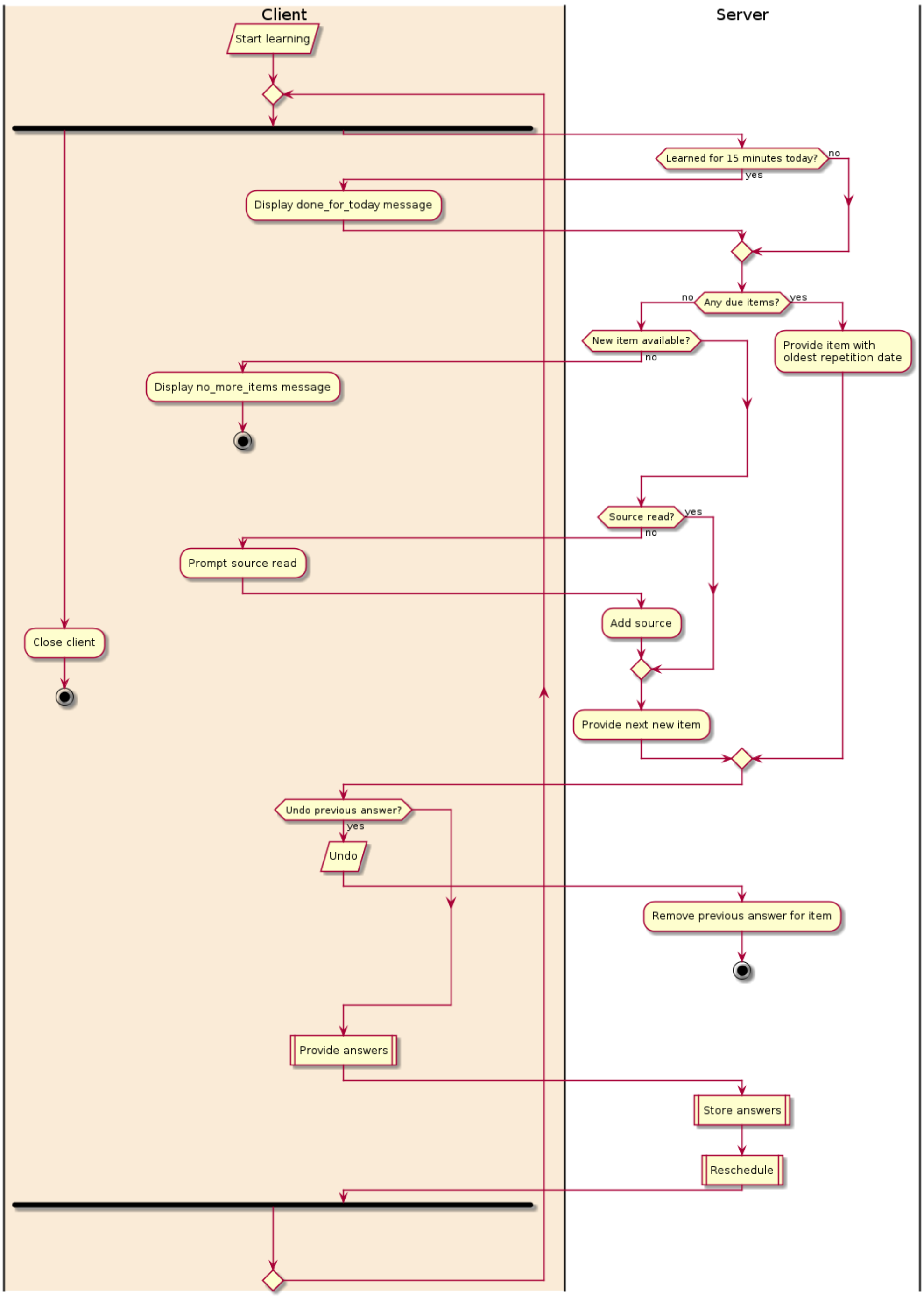


Figure 15: An UML activity diagram describing the general server-client interactions related to reviewing items. The *Provide answers* activity is described in more detail on page 58, and the *Store answers* and *Reschedule* activities in figure 20 on page 55

Validating the item When the user has reviewed the item, the client sents a "VALIDATE" message to the server with an 'id' field with the item id and a 'correct' field with information whether the user was able to think of the correct response for the item. This response will then be stored in the database by the server, and the item will be rescheduled for when it is due for the next review. After this, the server will repeat the learning cycle as if the client just sent a "LEARN-REQUEST" message.

Undo previous response When the supplied item is not the first item within the current learning section, the user can choose to undo the response of the previous item. The client will then send back a "UNDO" message to the server, which will then remove the previous response and also go back to the beginning of the learning cycle.

Design frameworks

This chapter describes the general design frameworks used for the implementation of the use cases defined in the previous chapter. These frameworks are Concept map construction design features (Novak & Cañas, 2008), flashcard learning design features (Nakata, 2011), and the two factor theory for website design (Zhang & von Dran, 2000). The aim is to implement these features from the next sections within the flashmap software in order to enhance the quality of the tool. Within the Server design and development and Client design and development chapters it is described when the sofware does not adhere to one of these original design features.

Concept map construction design features

Unfortunately, little is written about the design of clear concept maps for supplantive use. Only one technical report by (Novak & Cañas, 2008) is written on how students should construct concept maps, however the main focus of this document is on how this process can be scaffolded rather than discussing good design features of concept maps. The only two features mentioned are:

- Cross-links are important in order to show relationships between subdomains in the map (also mentioned by Eppler (2006))
- One should avoid "String maps", which are maps mainly consisting of large sentences within the nodes

The developed concept map therefore focuses on meaningful relations between concepts rather than hierarchical structure, and divides nodes into smaller nodes whenever one node becomes to large.

Finally, Eppler (2006) describes that concept maps are top-down diagrams rather than radial diagrams, therefore the graph layout will be hierarchical with top-down orientation.

Flashcard learning design features

Nakata (2011) describes a framework for developing flashcard applications based on several design features generally present in all major flashcard systems, and findings of earlier studies. The framework is split up in features aimed at creation and editing of flashcards, and at learning of flashcards. However, the creation and editing features are not relevant to the design of the flashmap system, since the content is already developed for the students. Furthermore, the Flashcard systems reviewed by Nakata (2011) are aimed at vocabulary learning, so not all of the guidelines mentioned in the review can be fully generalised to the flashmap system. That being said, there is still quite some overlap between the functionality of these reviewed systems and the flashmap system, and even if a principle is not (completely) applicable it is still relevant to know why this is not the case and what should be the guideline instead.

The flashcard learning features can be split up in two subcategories, which are features related to how the cards are presented (Presentation and retrieval modes, Retrieval practices, Increasing retrieval effort, and Generative use), and features related to the rescheduling of cards for review (Block size, Adaptive sequencing, and Expanded rehearsal). All features are expounded in the following subsections.

Presentation and retrieval modes

It is recommended to use two different modes, namely the presentation mode — where users can familiarise themselves with not seen before flashcards —, and the retrieval mode — where the user tries to actively retrieve a target when shown the associated cue. The presentation mode is introduced, because retrieval of unfamiliar targets would only result in unsuccessful performance and negative effects on the motivation of the user. Nonetheless, it was decided to only include the retrieval mode, since the students have already familiarised themselves by listening to the teacher explanation and reading the book, which can already be regarded as a presentation mode.

Retrieval practices

Retrieval practices relate to the ways the system prompts the user to recall a target from memory. Within vocabulary learning, there are four different categories of retrieval practices divided into 2 axes: reception versus production, and recall versus recognition.

Reception and production The first axis of reception and production relates to which part of the associated pair should be retrieved. Reception means that the meaning of the word should be retrieved, whereas production refers to retrieving the target word. For example, when having the pair ”Goedenmiddag” and ”Good afternoon” while learnig Dutch vocabulary, with reception ”Good afternoon” would have to be retrieve when being shown ”Goedenmiddag”, whereas in productive recall ”Goedenmiddag” would have to be retrieved when being shown ”Good afternoon”.

Recall and recognition The second axis refers to how the retrieval takes place, namely whether the student tries to recall the cue from memory (recall), or whether he chooses the correct answer from a list of possibilities (recognition). For example, when presented with ”Goedenmiddag”, when using recall the student should think of the correct answer on his own, where when using recognition the student should choose the correct answer from a list, e.g. ”Good morning”, ”Good afternoon”, or ”Good evening”.

According to Nakata (2011), it is difficult for students to aquire both the word form-meaning connection and the word form of a word simultaneously, mainly because of limited cognitive resources. Therefore, good vocabulary learning software should split these tasks into separate exercises: one using reception or recognition (or both), and one using specifically productive recall.

In the case of using flashcards for learning texts, reception and production could be interpreted as prompting the question versus prompting the answer. Recall and recognition still work in the same way, namely as an open prompt or a multiple-choice prompt. In flashmaps however there are not 2 elements but 3 elements per prompt — the parent node, the edge, and the child node — and this makes way for 6 options rather than 2 options for the reception-production axis, which are displayed in table 1. Within the table they are ordered from leaning towards reception to

	Parent node	Edge	Child node
1	X	X	
2	X		X
3		X	X
4	X		
5		X	
6			X

Table 1: The different possibilies for the reception-production axis in the flashmap system, with an hypothesised ordering from more leaning towards reception to more towards production. The shown elements for each retrieval mode are indicated by an X, whereas the other elements would be the targets for retrieval.

learning towards production, with the underlying hypothesis that retrieving the child node is more in allignment with reception whereas retrieving the parent node is more in allignment with production. This of course also depends on the concept map. The recall and recognition axis still stays the same, where in the latter case the possible options could be displayed next to the flashmap.

Only one retrieval practice was chosen for this experiment in order to keep the amount of variables to a minimum and thereby make a better comparison between the conditions. For the flashcard condition, the receptive recall was chosen, since this was the skill most relevant to the test itself. The flashmap equivalent was retrieval practice 1 from table 1 combined with recall (in contrast to recognition). Other retrieval practices are however still worth investigating in further research.

Another consideration here not mentioned by Nakata (2011) how the answer retrieved by the student should be compared with the actual answer. In the case of recognition, the user could simply click on the answer he thinks to be the correct answer, which can then be compared to the correct answer. Within the case of recall however, the student either has to merely indicate whether his response was in allignment with the correct response, or he has to type his answer which is then compared to the correct response by the software. The latter option was chosen, since this is easier for students using a mobile plattform to only press a button in comparison to typing on the screen keyboard. Furthermore, typing out an answer can lead to unintended rigidity of the system, such as an answer being marked as incorrect with spelling errors or omitted articles, or a highly increased complexity when trying to take into account these alternative correct responses. The only downside to this decision is that within the results there is an increased unreliability in how leniant the students were with their response being correct or incorrect.

Increasing retrieval effort

The design feature of increasing retrieval effort is strongly related to the choices made with regards to the previous design features. It entails that over the course of multiple presentations of one associated pair, the challenge of retrieval should be increasing. Nakata (2011) describes that this can be achieved by starting with the presentation mode before introducing the retrieval mode, and by gradually shifting from recognition modes to recall modes and from receptive modes to production modes. However, since the presentation mode is omitted and only one retrieval practice is chosen within this project, increasing retrieval effort is not feasible in this way. This again could be incorporated in later prototypes where other retrieval practices are incorporated.

Generative use

Generative use of words refers to presenting words in novel contexts. In the case of vocabulary learning, one can use specific words in different sentences which underly the different meanings of the words. Generative use also enhances the elaborative processing of certain concepts. Within the flashmap system, this is incorporated by presenting the concepts together with different edges, sometimes because a concept switches from child node to parent node, but in other cases because multiple unique edges direct from or to a concept. Within the flashcard system, this is also the case but more implicit, because every edge (or group of similar edges) being translated to flaschards, and thereby incorporating the concepts in multiple questions and answers. The only concepts appearing in one instance would be the concepts with only one outgoing edge (root node, K=1), since they only appear in one question, or only one incoming edge (leaf node, K=1), since they only appear in one answer. Of these, the root node promblems can be eliminated by creating more direct subconcepts in the hierarchy or omitting the direct subconcept and linking its subconcept direct to the root node. The leaf node problems are more difficult to eliminate, but still can be linked to other concepts by creating more cross-links generating more incoming or outgoing edges on the lower levels. However, this is not always possible, since some leaf nodes cannot be meaningfully connected to other nodes. In this case it is also worth considering whether this node could be eliminated altogether.

Block size

Within the Spacing effect section on page 19 it is described how repetition of an item is more effective when interleaved by other items (spaced items) than when repeated in immediate succession (massed items). The block size is therein defined as the length of items after which items are repeated again. When using massed items, one would have a block size of one, whereas when interleaving each repetition by 8 other items, the block size would be 9. The Implications for the flashcard system section however describes why it is better to use adaptive spaced-repetion learning instead of fixed block sizes, which is also recommended by Nakata (2011). Therefore, no specific fixed block size will be used within the scheduling algorithm.

Expanded rehearsal

According to Nakata (2011), expanded rehearsal is widely believed to be the most effective. The main difficulty for choosing the right slope for the expanded rehearsal is to balance between overlearning — repeating items too often, reducing the effectiveness of each repetition — and the forgetting curve — repeating items too little, leading to the students forgetting the card and thereby frustrating the user and even ineffective studying.

The first system which implemented this system is the Pimsleur system, where over the course of a 30-minute audio lessons words would be presented in a progressive series of exponentially expanding intervals with a base value of 5 seconds (Edge et al., 2012). This means that the first repetition would take place after $5^1 = 5$ s, the second after $5^2 = 25$, the third after $5^3 = 125$ s, etc. This system thereby already took into account the decreasing slope of the forgetting curve because of the power law of learning (see figure 8 on page 8), preserving a steady retrieval chance with a decreasing amount of repetitions. This curve is possibly too flat and leads to increased overlearning, however overlearning can also lead to a benefitial confidence within the user because of the higher percentage in correct retrievals. The Pimsleur system is also rather simple, reducing the amount of variables in the research. Because of these reasons, it is chosen as the basis for the flashmap scheduling algorithm.

Adaptive sequencing

An adaptive sequencing algorithm takes into account the learners' previous performance on individual items when rescheduling an item for the next review. Within the original Pimsleur system, one would always increase the time interval for the next review independent of whether the student could correctly recall the item or not. However, this does not account for the flashcards which are more difficult or forgotten at the time of the new review, resulting in the user not being able to keep up with those cards.

The first system to implement an adaptive sequencing element is the Leitner system, which is also the most basic adaptive sequencing system. The user has a number of piles, each representative of a expanded time interval, and a stack of physical flashcards. Each time a flashcard is answered correctly, it moves to the next pile, resulting in a larger time interval before the next repetition, and when answered incorrectly it would be moved back to the first pile. The rationale behind this system is that when an item cannot be retrieved it is forgotten, and the expanded rehearsal should therefore be reset to the lowest value.

The main problem with the Leitner system is that when it was introduced, managing the flashcards and different piles was quite a hassle. This problem was resolved with the introduction of digital flashcards, since the computer could take care of the scheduling and bookkeeping of the flashcards and their reviews. The Leitner system is therefore still prevalent in most digital flashcard systems (e.g. superMemo, Anki, and FaCT).

When combining the Pimsleur and Leitner system as one system, one gets the formula $i = 5^c$ for scheduling the flashcards, where i is the time interval in seconds, and c the amount of times the flashcard was correctly retrieved for this item since the last incorrect retrieval (or the total amount of retrievals when there were no incorrect retrieval).

Edge et al. (2012) introduced an even more sophisticated system, namely the adaptive spaced repetition system. On top of the Pimsleur intervals and Leitner adaptive sequencing, it also adapts the time intervals based on the amount of answers correctly and incorrectly answered by the student for each time interval. This results in specific time intervals better catered towards the user's ability. This is not included within the scheduling algorithm used within this project, because it adds another variable to the experiment. It can still be included however when aiming to improve the prototype.

The ARCS model

In the Analysis of the learner section, it was already found that it is likely for the students to have a low intrinsic motivation for engaging into the subject matter. Therefore, it is also important to include a framework within the motivational domain next to the previous framework focusing mainly on the cognitive domain. A commonly used and well researched model for incorporating motivational features is the ARCS model (Keller, 2000), which is an acronym for Attention, Relevance, Confidence, and Satisfaction. The factors were applied within the application where possible, however many are also applied within the context outside of the application. They all can be expressed in three subcategories, described in the following sections.

Attention

In this instance, the category of attention mainly refers to gaining attention at the beginning of an instruction, whereas keeping the attention is covered by the other categories. It can be gained by simple unexpected events, or mentally stimulating problems engaging the learner. Furthermore, variety is important for the effect not wearing down.

Perceptual arousal This mainly refers to the simple events — such as whistles or strange imagery — in order to capture the interest from the learner. In order to achieve this for drawing the user towards using the system, the researcher went to the classroom in the last lesson where the subject material was taught, where the attention was mainly gained from the effect of having someone else than the teacher appearing in the classroom. Within the system itself it was harder to achieve, since the content of the instruction was dynamically generated and therefore it is more difficult to find fitting content for each study session. It would then also add extraneous cognitive load, which is undesirable both for the learning achievement as well as the experimental setting.

Inquiry arousal This relates to the deeper, mentally stimulating problems which can be offered to the student in order to activate engagement with the topic. Related to the content itself, this is mainly done by the teacher within the lessons and by the instructional material. The researcher also tried to stimulate inquiry by stating that this was an early opportunity for the students to experience what research on a university looks like. Finally, by continuously asking questions users are also stimulated to actively participate in the instruction.

Variability Finally, if the setup for every instruction is the same, learners will eventually get disengaged because of the predictability. Using this system instead of the usual chapter reading in preparation of an exam might be an example of breaking such a predictable pattern. On the other hand, one of the downsides of drill and practice is its repetitive nature, so this could demotivate or disengage the students.

Relevance

Gaining arousal is not enought to keep the learner engaged over a long period of time. One method therefore to keep the attention is for the user to understand why it is relevant to engage in the learning activity. The subject matter however is not that interesting towards the average high school student. Therefore, the presentation of the system by the researcher will mainly focus on how it can help students to effectively and efficiently prepare them for the exam, since this is likely to be their direct goal.

Goal orientation The first step in establishing relevance is to relate to the needs and goals of the learner. In order to achieve this, the introduction by the researcher mainly delves into how a general flashcard system works and how it benefits learning, making the process more effective and efficient than purely reading through the book. Added to this, it is mentioned that some of the flashcard and test questions will be repeated on the actual test used for grading the students, making it extra attractive because of the sneak preview.

Motive matching This relates more to provide learners with appropriate choices, responsibilities, and influences. One way to do this is by modeling, derived from the theory of planned behaviour (Ajzen, 1991). In order to convince the students that the software is reliable, the researcher adds an anecdote from his own experience of using the system, and that by using it he had a guarantee of being well prepared for exams because of how the algorithm works.

Familiarity This subcategory mainly refers to tying the content to the learners' experiences. Unfortunately, the content of the instruction does not lend itself well for this subcategory, hence it is also not used in the instructional material and thereby in the flashmap and flashcards. However, the teachers made great effort within the lessons itself to explain how the renaissance genres and techniques are still prevalent in todays writing, with examples known to the learner.

Confidence

The learner must, next to feeling the subject matter is relevant, also be confident that he is able to learn the subject and to perform the learning activities successfully. This confindence can be boosted by creating the right expectations and providing positive feedback related to the learning activity.

Learning requirements One way the confidence can be boosted on beforehand is by assisting in building a positive expectation for success within the learner. In this case this is done by the researcher first acknowledging that learning and comprehending the core message of a text can be difficult, but that the system can assist this process greatly and makes it easy to do. This is also done within the initial presentation of the software. Furthermore, within the presentation as well as within the software it is made clear that the researcher could support individual learners at any moment if they would get stuck using the application. Within the software this was achieved by including a separate help page with explanations on how to use the software, by including a small text above the main content explaining what the user should do within the current step, and an email address in the navigation menu for contacting the researcher himself.

Learning activities Additionally, the learning activities themselves can also support or even enhance the students' beliefs in their competence. Within the flashcard system this is done by having the relatively flat expanded rehearsal slope from the Pimsleur system, which generates more overlearning and thereby increases the amount of correct retrievals. A more steep slope would result in more failed retrieval attempts, increasing the frustration within the user and thereby requiring a more stoic attitude.

Success attributions Finally, the learner also has to attribute his success to the use of the system in order to be motivated to use it. However, according to Logan et al. (2012), judgement of learning of students using spaced items does not appear to be higher than when using massed items. Therefore, a separate overview is included within the system displaying the progress the learner has made and how much items are still left.

Satisfaction

Finally, in order to sustain motivation after the student is attentive towards, understands the relevance of, and is confident about performing the learning activities, satisfaction is required. This is mainly related to positive feelings stemming from the reward system, and are generally categorised in intrinsic and extrinsic motivation.

Self-reinforcement Intrinsic satisfaction is probably the most powerfull satisfaction, since it works directly on the short-term. This feeling can be instigated for example when the learner is intrigued by the subject, but also when he experiences some form of achievement. It is already stated within the Analysis of the learner that most students will most likely not be highly interested within the subject matter, and therefore the first category of intrinsic satisfaction will only be met in rare cases. However, successful retrieval does lead to a sense of achievement, next to it being a confidence boost and effective learning tool. Therefore, the flashcard system itself might already facilitate self-reinforcement.

Extrinsic rewards An important extrinsic motivator for the student is to pass the test at the end of the course, for which this system is an effective aid. However, as the teacher already stated before the experiment, this motivator is probably not enough motivation for the students to spend extra free time in such a tool. This is mainly because students often overestimate their own abilities and thereby deeming the extra preparation as not necessary, and them being less able to oversee long-term consequences because of their prefrontal cortex still being in development. In order to create an extra reward, the students are rewarded with a coupon for icecream if they successfully committed to he whole experiment. This entailed filling in the pre- and posttest and questionnaire, and spending 15 minutes every day over the course of 6 days on the system. This reward of icecream seemed to be what the students were most excited about during the initial presentation by the researcher.

Equity Finally, it is imporant that each learner has a feeling of fair treatment. This entails that there was an appropriate amount of work, that the work they did is related to the final test, and that there is no favouritism among students. Using the system might feel as extra work to the students in comparison to just reading the textbook, even if it attributes to a higher chance of passing the test. This is an extra reason to emphasise the added value of using the software during the presentation. Furthermore, the consistency of the learning content and objectives and the test is guaranteed by deriving the concept map directly from the instructional material, by including flashcards and pre- or posttest question on the school test, and by letting the teacher confirm that the flashcards cover what the students have to know for the test and that there are no extraneous flashcards. Finally, equity is guaranteed by the fact that all students have anonymous accounts and thereby exactly the same general treatment during the learning procedure. The only contingency is that some students are using the flashcard system while others using the flashmap system, which might provide one group with advantages over the other group.

Server design and development

In this chapter, the general model behind the server will be elaborated. The formal documentation with descriptions of the server classes and their methods can be found on page 91, which also includes an UML class diagram ((OMG), 2015). An HTML version of the documentation is also available on `mvdenk.com/thesis/doc/`, and the entire source code is available at `https://github.com/mcvdenk/MasterThesis-Software/tree/master/server/`. The description of the model is divided in generic data entries — entailing the supplanted data by the developer such as the concept map or test items — and the user attributes, objects and methods — entailing user specific data such as birthdate or how often a user responded correctly or incorrectly to a certain instance. Only the conceptually relevant methods will be described below, since this keeps the thesis more accessible for readers not familiar with programming paradigms, and these methods are already described within the documentation. For example, the to_dict() method is prevalent in a large amount of classes, but merely serves the purpose of converting the class into an object that can be sent over the network connection to and rendered within the client and therefore is not conceptually relevant and is not elaborated. The two classes left out of the description are the Controller class, which parses client messages, operates the server and translates the server output to a new network message, and the LogEntry, which represents a single incoming or outgoing network message. These classes are also not conceptually relevant, and their functionality can already be derived from the previously described UML Activity Diagrams in figure 14 on page 35 and figure 15 on page 37.

Generic data entries

There are four classes of generic data: the concept map (containing nodes and edges), the flashcards, the test items and the questionnaire items. The concept map and flashcard model is illustrated in figure 16.

Concept map

The ConceptMap class is mainly a container class consisting of nodes and edges, and certain useful methods for performing standard queries on the concept map. As described in the general definition of the concept map by Cañas and Novak (2012), a Node object represents a concept, and an Edge object represents a relation between two concepts. The Node and Edge were originally intended to be embedded within the ConceptMap class, however this makes them impossible to refer to by other classes in MongoEngine (used for interacting with the MongoDB database). Furthermore, it could theoretically be possible for certain Nodes to exist in different concept maps. Therefore, they are implemented as separate classes in this server model. The example concept map in figure 17a is used to demonstrate the different functions in of the class.

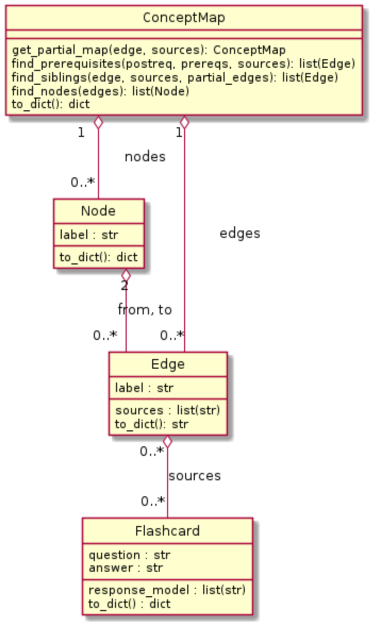


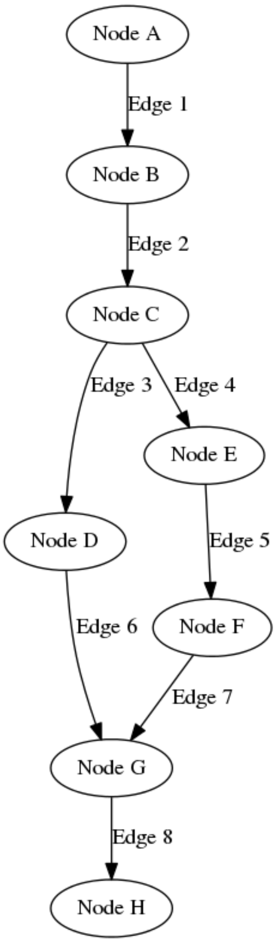
Figure 16: A UML class diagram illustrating the ConceptMap, Node, Edge, and Flashcard classes

Nodes The concepts represented by the nodes are not only an abstract ideas (such as Renaissance Literature), but also more concrete concepts such as a time period (e.g. the Golden Age), a person (e.g. P.C. Hooft) or objects or inventions (e.g. the printing press). Nodes are simple classes only containing a label describing what it should represent and a unique identifyer string.

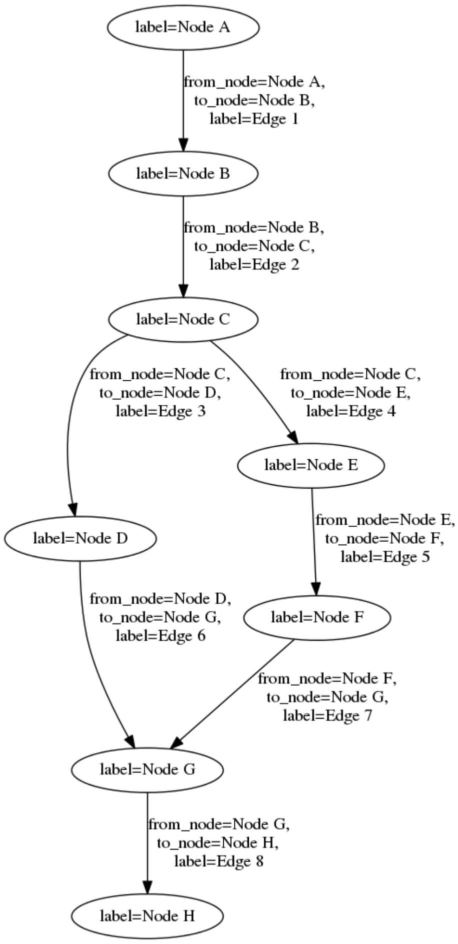
Edges An Edge also contains a label describing the specific relation between two concepts and an id, but also contains the references to the id's of the nodes it refers to, one being the 'from' node and the other the 'to' node, and a list of sources (usually only containing one source), which are the sections of the instructional material the edge is derived from. The model does not make any destinction between a hierarchical and cross-link, since they are the same from a graph rendering perspective. However, this destinction might still be considered for more sophisticated hierarchical rendering or searching algorithms. The Edge class is chosen as the equivalent to the Flashcard in the Flashmap system rather than the Node, since a relation between concepts is more specific when teaching about concepts. Furthermore, within an Edge object there is already a clear question (the from_node) and answer (the to_node).

Figure 17b demonstrates how the different classes and attributes represent the concept map from figure 17a.

Methods The most important method of the ConceptMap class is the get_partial_map() function, which will provide a new ConceptMap object containing all the parent nodes and edges which directly or indirectly link to a given Node (the parent nodes and edges), plus the nodes and edges linked to by the direct parent nodes (the sibling nodes and edges). This concept map can then be displayed to the user when showing a specific flashmap instance for review. The reason why the parent nodes are returned rather than the child nodes is that in the instructional material the concepts are introduced top-down rather than bottom up, so building up the concept map from parent to child alligns better with the order in which the students read about the concepts.



(a) An abstract example of a concept map



(b) The different attributes demonstrated within the example map from figure 17a

Figure 17: An example illustrating the ConceptMap class

Additionally, the sibling nodes are also returned so that they can be prompted at the same time, and that the user has more context for deciding which concept should be filled in the missing node. Figure 18a and figure 18b demonstrate the get_partial_map() function with Node D and Node G from the example map in figure 17a.

Flashcards

A Flashcard class represents a traditional flashcard by simply having a question and answer entry. It addtitionally has an response model entry in order to also function as a test item. In most cases, this is a list only containing the answer entry, however in some cases the answer entry is split into multiple response entries. Finally, since each flashmap is based on the concept map, each Flashcard object also contains a list with Edges the card is derived from. This has the advantage of being able to compare the flashcards with the concept map, but also indirectly relates the flashcards to the sections within the instructional material.

Test items

A TestItem object represents an item on the pre- or posttest. It is very similar to a Flashcard object, with the exception that it directly links to the text sources, and does not contain an answer field since this is never displayed to the user.

Questionnaire items

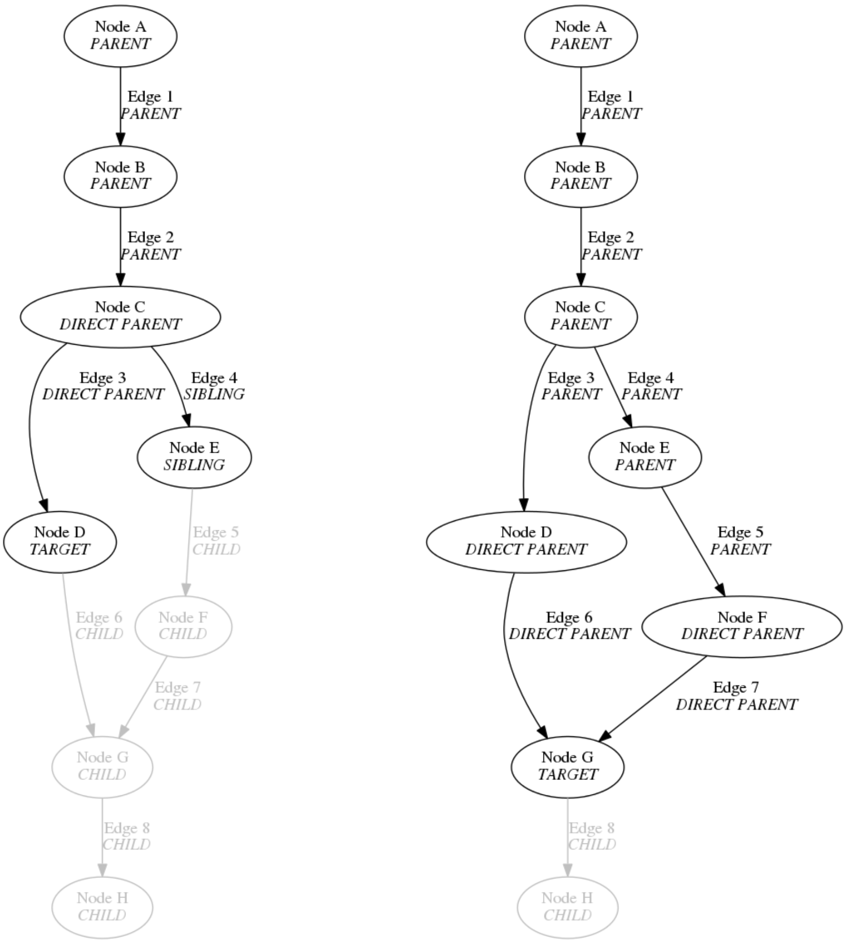
The QuestionnaireItem class represents items from the Technology Acceptance Model questionnaire (Davis, 1989), and contains a usefulness entry categorising the item as either a Perceived Usefulness item or as a Perceived Ease of Use item, and a positive and a negative phrasing entry. Both phrasings are included instead of only the standard positive phrasing, so that one of these phrasings can be selected when presenting the item to the user, avoiding only one type of phrasing causing a bias within the user towards that specific phrasing.

User attributes, objects and methods

The main user attributes are mainly based on the aforementioned generic data entries, namely the FlashcardInstance or FlashmapInstance objects — storing the learning progression of certain Flashcards or Edges from the ConceptMap —, the Test objects — representing a pre- or posttest and containing the responses to sets of TestItems —, and the Questionnaire object — containing the responses to the QuestionnaireItems. Next to these objects, the user also has certain descriptive attributes, and objects related to logged in sessions. The User class and its attributes and embedded classes are illustrated in figure 19.

Descriptive attributes

The descriptive attributes are either used for the program to function, or for generating descriptive statistics as controll variables in the results section. They contain the username, the condition, the gender and birthdate of the user, the code he received on his informed consent form, an email address, and a debriefed field. The methods defined within the User class are mainly used for communication with the Controller class, with the exception of the undo() function which removes the last entered response.



(a) The result of get_partial_map() function with Node D from the example map from figure 17a

(b) The result of get_partial_map() function with Node G from the example map from figure 17a

Figure 18: Example for using the get_partial_map() function of the ConceptMap class

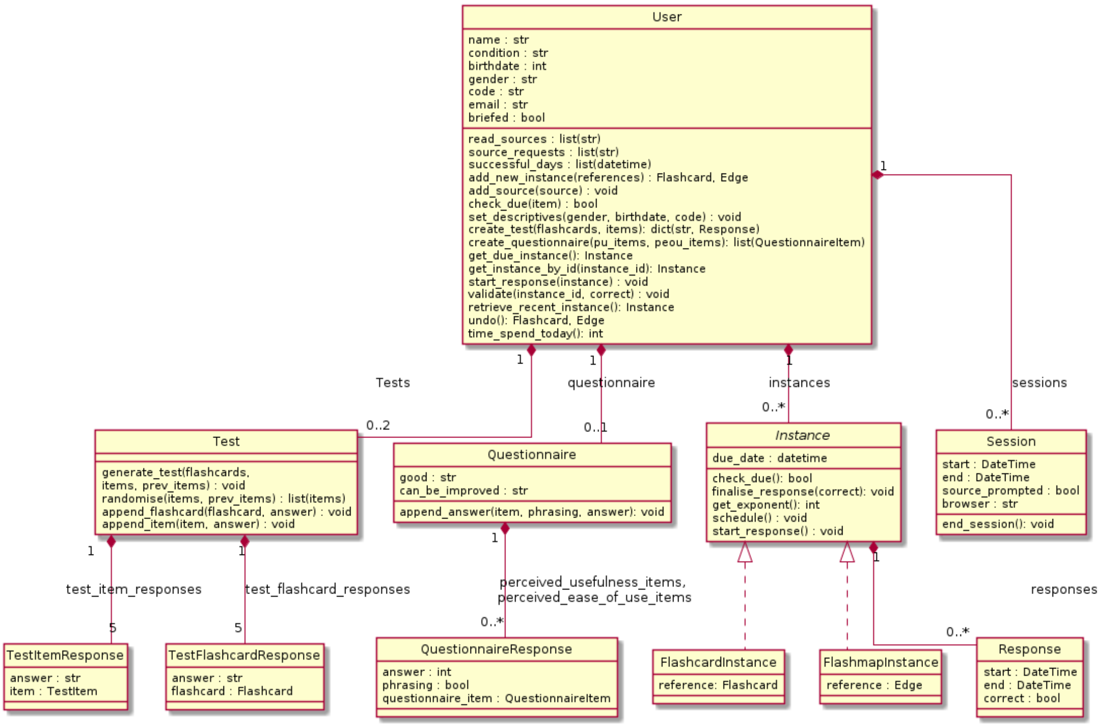


Figure 19: A UML class diagram illustrating the User class and its embedded classes. For readability, the TestItem, QuestionnaireItem, Flashcard and Edge classes are referred to inline within the attributes instead of as separate classes with aggregation links.

Username First and foremost, every user has a unique username which can be used to log in to the application. This name is chosen by the user himself, so he can decide to use his proper name or use an alias to remain anonymous. Even so, within the released data this field is removed in order to safeguard the identities of the user.

Condition This field determines whether the user is partaking in the "FLASHCARD" group (the control group) or the "FLASHMAP" group (the experimental group). This field is set when the user registers a new account by the formula $len(users)$ mod 2, which entails that every new user is assigned to the opposite group relative to the user before. This is in order to ensure that the initial users are equally divided over both groups.

Gender The gender field of the user is included in order to check after the experiment whether the distribution of genders is equal within both experimental conditions. This variable is prompted before the pretest.

Birthdate The birthdate field of the user is included in order to check after the experiment whether the age distributions of both experimental conditions are the same. This variable is prompted before the pretest.

Code Before the start of the experiment, every student participating had to sign an informed consent form together with a parent or caretaker in accordance with the Ethics Committee at the University of Twente. To ensure that every user in the system corresponded with students who handed in a double signed form without the user having to enter his own name into the system, every form contained a code which was prompted towards the user before the pretest.

Email At the end of the questionnaire, the user was prompted whether he wanted to participate in an interview. If this was the case, he could fill in his email address here in order to be contacted at a later date. This variable is also ommitted in the published data.

Debriefed This field is a simple boolean value indicating whether the user has been debriefed. It initiated with a False value, and was set to True when the server received a "DEBRIEFING-RESPONSE" message from the client (see figure 14 on page 35).

Sessions

A Session object represents the time between the user logs in and out. This is represented by a *start* and *end* value, indicating the time interval and the moment that the user was active. Additionally, the Session object has a variable *source_prompted*, indicating whether the user was prompted to read a section within the instructional material. The object finally contains a string *browser*, which is the string retrieved from navigator.platform in the client window.

Instances

The instances class is the most relevant class for the functioning of the flashcard system, containing a record of responses and the scheduling function. The name instance is used in order to distinguish between an abstract Flashcard or Edge, which is the same for each user, and the specific instance of this idea within the memory of a user. An Instance object can be either a FlashcardInstance or a FlashmapInstance, depending of the experimental condition of the user.

Attributes Each Instance object has a *due_date* and *reference* attribute. The *due_date* attribute indicates when this instance is due for review by the user, and initiates at the current date and time. This date is set by the scheduling function, which is elaborated below. The reference attribute contains either a Flashcard or an Edge object. As already indicated in the previous chapter, each time the user has no more due instances a new instance is created referring to a new item. The *reference* field refers to either a Flashcard or an Edge, depending on the specific subclass of Instance.

Responses The Response class represents each individual time the user reviews an instance, containing a *start* field for when the Flashcard or Edge was prompted, an *end* field for when the server received the response from the user, and a *correct* field for whether the user was able to think of the correct answer or to_node when being prompted.

Schedule The scheduling algorithm uses two different methods from Instances, namely get_exponent() and schedule(), and is illustrated in figure 20. In the Design frameworks chapter on page 39, it is described that to calculate the interval until the next review, one needs the number of correct responses since the last incorrect response. This is done by the get_exponent() function, which loops through the responses in descending order of *end* date, increasing a counter (exp, initiated with value 1), until a response with a False *correct* value is found. The schedule() function then takes 5^{exp} seconds as the interval until the next review of the instance as specified in the Adaptive sequencing section on page 43. Examples:

- When there are no responses, the interval is $5^{1+0} = 5$ seconds;
- When there are two correct responses, the interval is $5^{1+2} = 125$ seconds;
- When there are two correct responses, followed by one incorrect response and then three correct responses, the interval is $5^{1+3} = 625$ seconds.

This interval is then added together with the *end* value of the last response and sets this as the new *due_date* value.

Tests and Questionnaire

Finally, each User also contains Test entries and a Questionnaire entry, which involve response entries and constructor functions.

Responses The responses consist either of FlashcardItemResponses and TestItemResponses, or QuestionnaireResponses, which all refer to a database item (a Flashcard, TestItem or QuestionnaireItem) and contain an answer string or likert-scale value. Each QuestionnaireItem also contains an entry indicating which *phrasing* was used when prompting the item, and a Questionnaire also contains string entries for what the user thought was good about the application, and what could be improved.

Constructors are used to create randomly generated Test and Questionnaire objects based on the items in the database. The Test constructor creates a random set of 5 to be filled in FlashcardResponses and a random set of 5 TestItemResponses, which are used as the Knowledge and Comprehension questions. The constructor can also be given a list of Flashcards and TestItems to be excluded from the selection. This makes it possible to generate posttest which do not contain any items present within the pretest. The Questionnaire constructor creates a

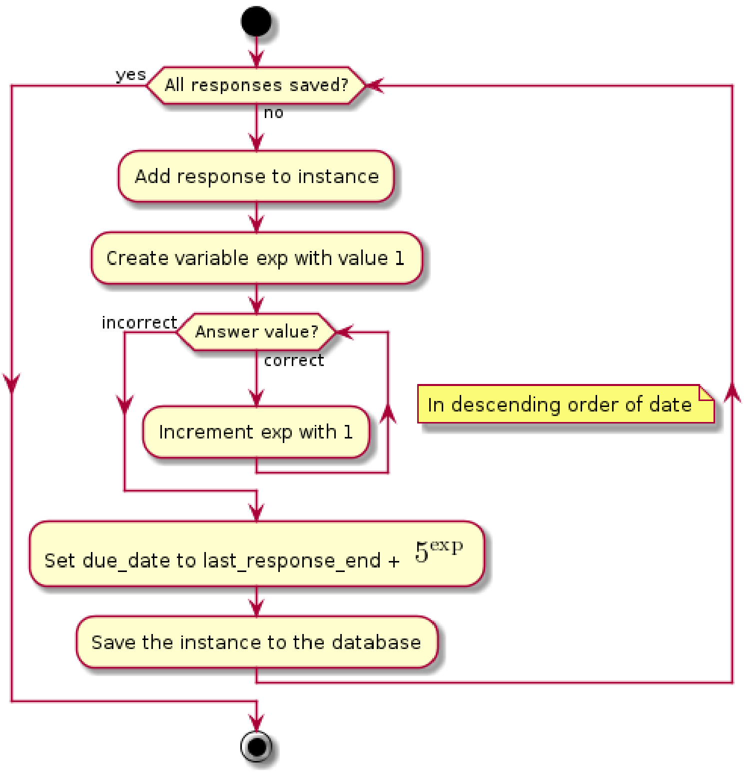


Figure 20: An UML activity diagram showing the scheduling and saving of a list of responses within instances

shuffled set of Perceived Usefulness items with randomly assigned phrasings, a shuffled set with the same Perceived Usefulness items but with opposite phrasings, and similar two sets containing Perceived Ease of Use items.

Unittests

In order to remove programming errors and unexpected behaviour, all server methods are rigorously tested using unittest, which can be found on `https://github.com/mcvdenk/MasterThesis-Software/tree/master/server/unittests`.

Client design and development

Within this chapter, the front end of the webapplication and its design choices are expounded. Firstly, the general page elements are explained, which are mainly defined within the HTML and CSS of the application (see the index.html and style.css within the Source files of the client appendix on page 112. Sequentially, the learning process interface is elaborated in a separate section, since this encompasses the main functionality of the application. This process is mainly defined within client.js (see the client.js section, also within the Source files of the client appendix). The complete source code for the client is also available on `https://github.com/mcvdenk/MasterThesis-Software/tree/master/client`. Finally, the other views are described, such as the login screen or the learning progress overview. All full screenshots referred to within this chapter are included in the Screenshots of the client appendix on page 125.

Page elements

Each page is represented by the same HTML file defining 4 different page elements, which are the navigation menu, the instructions panel, the main viewer, and a button panel. Within the different views of the application, they generally preserve the same functionality and layout, and will be explained below after the description of the colour scheme.

Navigation menu

The navigation appears as a centered div at the top of the screen, displaying buttons for the pages of the applications plus a button to contact the developer for help. An image of this menu is included in figure 21.

Instructions panel

The instructions panel is the next element is placed below the navigation menu, and is reserved for providing the user with extra instructions where needed. It does not have a background colour, but it does have a fixed height in order to keep all elements at the same place independent of the length of the instruction. The instruction has a centric text-alignment.

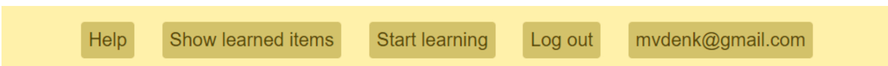


Figure 21: The navigation menu

Main viewer

The main viewer is the center element, and expands from the instruction panel to the button panel. Within this container, the main content of the specific view is displayed, such as the flashcard or concept map, the questionnaire, or the login form. In order to stand out from the rest of the page, it has a separate background with rounded corners. The background colour is somewhat lighter in comparison to the general background colour in order for the text to be better readable. The main viewer is also the container for visjs, which is a javascript library for rendering graphs in browsers.

Visjs Since the content contained within the graph is dynamic because of the partial maps returned from the server, generated automatic layouts of graphs are necessary. Visjs is capable of two models for automatic layout, namely hierarchical and force-directed. As described in the Concept map construction design features section on page 39, the initial idea was to render the graphs as hierarchical. Upon trying this with different subgraphs however it was found that automatic assignment for the different nodes on different hierarchical levels was not correctly done by visjs. This is mainly due to the two options for hierarchical layouts, namely hubcentered and directed. The idea of a hubcentered hierarchy is that the levels of the nodes within the hierarchy are based on the amount of other nodes directly or indirectly linked to this node. This works especially well for tree graphs, but because of the cross-links a concept map is not a tree graph. The other option, directed hierarchy, should make advantage of the directed edges by determining the levels of the nodes based on the direction of the edges. Unfortunately, this is implemented within visjs as only the root and the leaves being determined whether there are only incoming or outgoing edges, whereas the rest of the nodes are still placed based on the hubcentered layout, unlike in other graph layout engines such as DOT.

Because this rendering leads to more confusing graphs, the force-directed layout was chosen instead, despite this resulting in a more cyclical graphs common in other visualisation techniques such as mind maps. This layout engine attempts to position the nodes in such a way that all edges are about equally long and there are as few crossing edges as possible. This is done by assigning forces among the set of edges and the set of nodes, for example for having all nodes an inverse gravity force and all edges a spring force.

The other options include options for assigning colours fitting within the existing colour scheme, and for the user being able to reposition nodes if for example the edge labels are not readable because of overlapping with other edges.

Button panel

Finally, within the footer of the page, a button panel is included. Here the user can choose to for example show the correct answer to a flashcard, or confirm that he has read a certain section within the instructional material. The layout of this panel is exactly the same as that of the navigation menu.

Learning process

The core functionality of the client is reviewing the user instances. In general, every time an instance is reviewed, first the question or incomplete flashmap is prompted, the user thinks of the correct answer, the client shows the correct answer, and finally the user indicates whether his answer was correct or incorrect. Furthermore, the client can prompt whether the user has read a certain section from the textbook, indicate that the user is finished with learning for today, or state that there are no more instances left to review. Finally, the user can also undo his last submitted response. These use cases are elaborated below.

Read source

When a new user starts learning, he will first be asked whether he has read section 13.1 from the instructional material (see figure 24). Within the main viewer the question "Did you read section 13.1 already? If no, read this now." is displayed in Dutch. The user can then press the "Read" button in the button panel, which will lead to prompting the first instance. This screen is simular for each subsequent section prompt.

Prompt

The prompt of an instance is dependent on the experimental condition of the user. If the user is a flashcard user, he will see the prompt such as in figure 25. The main viewer contains the specific flashcard question, and the button panel contains a button with the label "Show answer". The flashmap users get to see a partial incomplete flashmap within the main viewer (figure 26), which they can drag around and zoom in and out on. The cues which have to be retrieved are indicated by orange empty nodes. The button panel is the same for both conditions. After the user has at least responded to one instance, an "Undo" button appears left to the "Show answer" button (see figure 27), with which the user can reanswer his previous response. The instructions element also shows instructions on what the user should achieve (to retrieve the correct answer from memory).

Show answer

After the user has pressed the "Show answer" button, the show answer prompt will be shown. Flashcard users get to see the correct answer in the main viewer below the question, with "Incorrect" and "Correct" buttons in the button panel to indicate whether the correct answer could be retrieved (figure 28. Flashmap users get to see the correct answers within the previously empty nodes, which will also turn green indicating that the user retrieved them correctly (figure 29). When the user did not retrieve an answer correctly, he can click on that node, turning it red (figure 30). After the user indicated the correct and incorrect retrievals, he can click on a "Next" button in the button panel. The instructions element again contains instructions on what to do within this screen.

Finished learning and No more instances

Finally, when the user has spent 15 minutes on the system or when there are no instances left to review, the user gets to see a screen such as in figure 31. The main viewer contains information on why the user is finished. When the user is finished because there are no more instances left in the sections he already read but there are still instances available in following sections, it also shows which section the user could read for the next instance, and presents a button to continue. Finally, if the user spent 6 days on the system, this prompt will also inform him that the next day he can take the posttest and fill in the questionnaire.

Other views

Next to the main functionality described in the previous section there are also other views for accommodating the other use cases. For new users, these are the login screen, the descriptives screen, and the pretest, for regular users there are the help screen and the learning progress screen, and for the users which are finished there are the posttest, questionnaire, and debriefing screens.

Login screen

The main viewer in the login screen containing a simple form with a textfield for the username, and a submit button for logging in (figure 32). Furthermore, a text within the instructions panel refers the users to the researcher's email adress for when they require further instructions or when the logging in does not function.

Descriptives, Test and Questionnaire forms

These screens also contain basic forms, all containing questions or items and either textfields or radiobutton selection panels, depending on whether the question or item is open or closed, and a submit button at the bottom. (figures 33, 34 and 35, and 36 and 37). The date field within the descriptives form is checked to be a valid date before the user can submit. Furthermore, the questionnaire item contains an email field for when the user wants to sign up for a later interview, which is a voluntary field and can be left empty. The instructions element again provide instructions for how to fill in the forms.

Debriefing

Finally, when the user is finished using the system and filling in the posttest and questionnaire, the application presents the debriefing information (figure 38). This entails a thank you message, that the user will receive the coupon for ice cream soon, that he is able to keep using the system, that he can request his personal data gathered during the experiment from the researcher at any time, that he will receive an email for making an appointment for the interview when he filled in his email address, and that for further questions he can always send an email to the researcher's email address. When the user has read this information he can click the "Read" button.

Help

The help screen (figure 39) contains some global information about the experiment and on what conditions the user can receive the icecream coupon. It also states that the system will notify the user as soon as he is ready for today.

Learning progress

Finally, the user can request information about how much progress he made. Flashcard users are presented with how many cards are ready to be learned right now, how many are never reviewed, how many are new (less than exponent 2), how many are in the learning stage (less than exponent 6) and finally how many cards have been learned for the long term (more than exponent 6). Figure 40 shows a learning progress screen of a new flashcard user, and figure 41 an shows this overview for a user having correctly reviewed one flashcard but incorrectly reviewed another flashcard. The flashmap user is shown with a different overview, namely the part of the concept map containing the edges already reviewed by the user (figure 42), which will expand during the use of the system.

These views can still be improved in order to better convey the progress towards a user, which would contribute to a higher self-reinforcement. However, this has not yet been implemented due to time constraints.

Part III

Research

Methods

Research design

Research questions Ia, b, IIa and b will be investigated using intervention-based research. Because of the systems being used for self-study by the students, they can be individually assigned to a condition, and this enables the use of a true experimental design. Since this will provide the most valid and reliable results, this research design is implemented in this experiment.

Additionally, research questions a and b will also be investigated using open questions on the questionnaire form and by conducting interviews with a sample of the participants.

The quantitative and qualitative results will be mixed for the purposes of triangulation and expansion as described by Schoonenboom (2014). The interviews and logs could provide insight in the degree of which the systems were used the intended way and in why students had certain perceptions on using the systems. Both triangulation and expansion will be on a partial level of mixing, will take place concurrently, and the quantitative data will be dominant, since the qualitative data exists only to triangulate and expand the quantitative data.

Respondents

100 15 to 17 year old tenth grade Dutch high school students will be approached. They already have to prepare themselves for an exam on the same topic and thereby have incentive to learn. To increase the response rate, the students will be rewarded with a € 5 voucher for participation. The participants will be assigned to either the flashcard or the flashmap condition at random when they create a user account within the webapplication.

Procedure

Review concept map, flashcards, and item bank In order to verify whether the content offered within the system is in alignment with the learning goals set by the teachers, one of the teachers is asked to review the content. The feedback received from the teacher is afterwards incorporated by altering the dataset. This can be seen as the focus evaluation of the product (Nieveen, Folmer, & Vielgen, 2012).

Approval ethical committee Before the actual experiment can take place, the research setup first has to be approved by the ethical committee of the University of Twente. This takes place before the system is introduced to the students.

Presentation Within the school curriculum there are two instructions planned for the topic of Dutch renaissance literature. At the end of the instruction, the researcher introduces the experiment and the system to the participants. This is meant both to attract students to participate, but also to provide a briefing next to the written briefing. Within the presentation, the benefits are stated (better preparation for the exam, preview for the exam), it is stressed that participation is voluntary and that the data will be collected anonymously, the informed consent form will be introduced, and finally the reward (icecream vourchers) will be announced, together with the conditions for receiving the reward. Information with regards to the seperate experimental conditions will be limited to the researcher explaining that there are two different versions, without elaborating what these versions entail. This is in order to prevent prejudgements from within the user, making it a double-blind experiment. It could however of course happen that the participants learn about each others version during the experiment, and unfortunately this cannot be prevented.

Informed consent form The informed consent form, included within the Informed consent form appendix on page 137, contains a letter, the written briefing, and a form which has to be signed by both the parent or caretaker and the participant. It also contains a code with which it can be verified that a user within the system did indeed sign this form. The briefing contains a description of the research, the advantages of participation, and the procedure of the experiment.

Division of respondents Users will be assigned alternately to the control group or the experimental group. This pseudo-random assignment increases the validity of the experimental design without risking one group becoming larger than the other group. This however only holds up for the initial group, because dropout rates between the group could vary resulting in differently sized finished groups.

Descriptives Before the experiment itself starts, the gender and the birthdate of the participants are prompted. This provides descriptive statistics necessary for the measure of generalisability of the results.

Pretest Another form prompted towards the user before the start of the experiment is the pretest, measuring how much the user already knows and understands about the subject. This test is elaborated within the next section.

Experiment The experiment itself consists of the participants having to review instances for 15 minutes over the course of 6 days. The 15 minutes are estimated to be the amount of time necessary to review one section in the instructional material, and the 6 days are chosen as a balance between having covered a large enough portion of the material to measure a significant learning gain without the participant investment being too large resulting in no student wanting to participate. The 6 days of learning ideally take place subsequently, since then students have a higher chance of retrieving repeating instances correctly. However, it is likely that participants could forget about the experiment or be too busy to invest the 15 minutes, and therefore they are allowed one non active day during the experiment. Each session consists of being presented by questions or incomplete concept maps, trying to retrieve the correct answer or missing concepts from memory, and indicating whether the correct answer or missing concept was successfully retrieved.

Posttest The seventh day the participant logs in to the system he is prompted with the posttest in order to measure the level of knowledge and comprehension after the experiment. This test uses the same itembank as the pretest, and is thereby also elaborated within the next section.

Questionnaire After filling in the posttest, the participant is also asked to fill in a questionnaire based on the Technology Acceptance Model, which is further elaborated within the next section. The form also contains two open questions, namely to describe what the participant thought was good about the system, and what he thought could be improved. Finally, he can fill in his email address if he is interested in being interviewed afterwards.

Debriefing Finally, the participant is presented with the debriefing text from figure 38 on page 134. This states that the user will soon receive the voucher, that he is allowed to keep using the system, and that he can contact the researcher if he has questions or when he wants to see his personal data. The participant is now finished with the main experiment.

Scoring sample items After all the results have been gathered, a small sample of the responses are reviewed by both one of the teachers and the researcher in order to establish an inter-rater reliability. This will take place after the school test itself, but before the teacher has scored the test administered by the school itself to remain unbiased. Furthermore, the sample will be a random anonymous selection of responses in order to minimise any halo effect. Finally, the samples are filtered on non-empty items which do also not exactly correspond to the response model, since these can be automatically scored. If the inter-rater reliability is too low, the scoring rubics will be altered in order to differentiate better among correct or incorrect responses, and the procedure is repeated. Otherwise, the rest of the responses is scored by the researcher.

Interview Those who volunteered for the interview will be sent an invitation by email for an open group interview, also elaborated in the next section.

Icecream vouchers The vouchers and the list containing the names of students having participated within the experiment will be handed over to the students after the scoring of the test administered within the school in order to avoid influencing the teachers during the marking of the tests.

Instrumentation

Test

A test (either pre- or posttest) consists out of a knowledge section and a comprehension section, derived from Bloom's Taxonomy (Bloom et al., 1956). The knowledge section aimed at measuring whether the rote memorisation was effective, and the comprehension section served the purpose of measuring whether the explicit relations within the flashmaps scaffolded the comprehension. Only these two levels were chosen, since higher levels are more time consuming to create and to answer by the students, and the first step is to only measure whether students can already generalise from only rehearsing questions on the knowledge level to questions on the comprehension level.

The itembank used for the knowledge questions were the flashcards themselves, and the itembank for the comprehension questions a seperately constructed set of 10 items. Both the knowledge questions and the comprehension questions are based on a concept map encompassing the instructional material created by the researcher. The pretest and the posttest both randomly select 5 questions from each bank with non-overlapping items. By randomly selecting the questions, the overall knowledge and comprehension are measured instead of specific subsets measured seperately on the pretest and posttest. This eliminates the specific item difficulty variable from the learning gain, increasing its validity. The random variable however also increases the variability of scores, decreasing the significance when comparing the conditions.

Finally, a rubic is created in order to quantisise the openly formulated answers, consisting out of possible answer categories for each item.

Questionnaire

The questionnaire consists out of items based on the Technology Acceptance Model (TAM) (Davis et al., 1989), containing items measuring perceived usefulness and perceived ease of use. DeVellis (2003) describes that the validity of a questionnaire can be increased by formulating the items mixed positively and negatively, and by repeating the items. Therefore, the items from TAM are translated to Dutch in both a positive and a negative phrasing. For each participant, 2 sets of items are created per TAM category: for each item, a phrasing is selected at random after which the set is shuffled; another set of shuffled perceived usefulness items is created with each item having the opposite phrasing of its counterpart in set 1. This results in 4 sets of items encompassing both phrasings of all items. For each item, the participant can indicate whether he completely disagrees (-2), he disagrees (-1), he neither agrees nor disagrees (0), he agrees (1), or he completely agrees (2).

Data collection during experiment

During the experiment itself more data is created in order to provide statistics about the usage of the system. These statistics are mainly contained within the Response objects in the database.

Correctness retrievals For each response within an instance, information is stored about whether the instance was retrieved correctly by the participant. This is not only useful for the rescheduling of the instance, but also in order to gain more insights in how often a participant was able to retrieve instances correctly, and to verify whether the success rate is indeed around 90% as stated by Edge et al. (2012).

Retrieval time Furthermore, the start time (when an instance is sent from the server to the participant) and the end time (when the response was received by the server) are stored for each response in order to measure how much time was spent on each instance, and on the system in general.

Active days Finally, a list is stored for each participant containing the dates that the participant was active. An active day is here defined as a day where the participant is active for 15 minutes or where the participant reviewed all of the available instances within the read sections. This data is useful for the system to know when to prompt the posttest, but also serves as a statistic on participation rates.

Interview

The interview is open instead of structured or semi-structured, since the interview should provide the interviewees the opportunity to tell about their own experiences from using the system, and since the data covering specific questions is already gathered within the usage data, the questionnaire and the two open questions. The researcher will use topics based on TAM in order to verify during the interview whether the main area is covered, with an open question at the end asking for general possible improvements. Furthermore, the interview will take place as one group interview, since the school organisation only has limited availability options for providing seperate rooms fit for interviews and the group interview being the most efficient option. At the start of the interview, the interviewees will be disclosed about the aims of the research in order to establish the different versions. Results will however not be disclosed until after the experiment in order to leave the interviewees unbiased in their opinion about the perceived usefulness. Finally, with the interviewees permission, the interview is recorded for later analysis. If not, the researcher takes notes during the interview.

Analysis

Inter-rater reliability

In order to verify whether the rubics is a reliable measure for quantificating the open answers on the test questions, first the inter-reliability is calculated. This is done by calculating both the proportionate agreement as the more reliable weighted Cohen's kappa (Cohen, 1968) between of one of the teachers and the researcher which marked 10 randomly seleceted knowledge items and 12 randomly selected comprehension items. The items were selected from both the pretest and posttest from random participants, and shuffled in order to minimise any halo effect.

Test scores

In order to determine the score on a test, an item matrix is created with the response categories from the rubic as columns, the participants as rows, and the cells containing a 0 or 1 indicating whether the participants answer contained the response category. Each possible answer category from the rubic is chosen for the columns over the test items in order to more accurately determine item difficulties. The raw item difficulties are then calculated by the sum over the columns, whereas the raw person abilities are determined by the sum over the rows. Furthermore, the absolute and relative learning gain are calculated for comparing the conditions. The absolute learning gain entails simply subtracting the pretest person abilities from the posttest person abilities, resulting in a list of scores indicating how many more points each person scored on the posttest than on the pretest. The relative learning gain is calculated by the posttest score minus the pretest score divided by the maximum possible posttest score minus the pretest score. The main advantage of using relative gain over absolute gain is that each user can have a different maximum test score because of the items having different maximum scores and them being randomly selected for either the pre- or posttest, and the relative learning gain implicitly controls for this where the absolute gain does not.

For each subcategory, two histograms are plotted: one for the item difficulty (expressed as the average amount of correct answer on an item per person) and one for the person ability (expressed as the average score of a person divided by the amount of items). The correction is necessary in order to compare the two groups on an equal level, and the numbers are not indicatory of the percentage of correct answers on the test. Both the pretest and the posttest histograms are plotted in one graph for each condition, and the learning gains of both conditions are also plotted in one graph for visualising the differences.

Participation statistics

For determining the participation rates, a matrix is constructed containing the different days of the experiment as columns, the participants as rows, and the cells containing a 0 or 1 indicating the participant being active on that day. The participation rates are then calculated by the sum over the rows, and the amount of active participants from the sum over the columns. These matrixes are constructed seperately for the control group, the experimental group, and the group of combined experimental conditions. The statistics are depicted as histograms for the amount of active days for a participant, and as a scatterdiagram showing the amount of activity for each day of the experiment, combined with a step diagram showing the cumulative amount of finished participants. The days are indicated by a number from 0 to 28, where 0 is the day of presentation, day 21 is the day of the school test, and 28 is the last day with any activity.

Instance statistics

This category contains all statistics derived from the Instance and Response objects in the database collected during the experiment. For all of the below subcategories, item matrices are constructed for the control group, experimental group, and the group of participants from combined conditions. These matrices consist of the Flashcards or Edges in the columns, depending on the condition, and the participants in the columns. Within the combined conditions group, flashcards are mapped to the first edge in their list of sources. Furthermore, for in the number of responses and time spent categories, the double flashmap instances are removed. These are the instances presented together in one flashmap to the user. Like in the previous categories, item scores are again calculated by sum over the columns, and person scores by sum over the rows. Additionally, a relative score is calculated by dividing sum by the amount of entries: for the item score the relative score is the sum (absolute score) divided by the amount of participants, and for the person score it is the absolute score divided by the amount of items. Finally, mean scores are calculated for both rows and columns. The relative score is meant for being able to compare both groups, whereas the mean score provides additional insight in the score per reviewed instance. Finally, histograms are depicted for both the item scores as the user scores.

Reviewed instances In order to determine how much of the instructional material is reviewed by each participant, the first subcategory investigated is the amount of reviewed instances. The cell of the item matrices contains whether an instance for a participant contains any response. The relative score proves here the most useful, since this is the ratio of covered material.

Number of responses This statistic contains information on the number of times an instance is retrieved by a participant, which is included in order to compare whether both groups had an equal amount of opportunities to rehearse the instances. Since the ratio of retrieved instances is already known from the previous category, the mean number of responses per instance is the most useful score for comparing the experimental groups.

Exponents The exponent statistic is directly derived from the get_expontent() function from Instance, indicating the amount of responses since the last incorrect retrieval. This is included as a measure of learning progress of an instance. For the same reasons as for the previous category, the mean values are the most useful measures for comparing the groups.

Responses marked as correct The ratio of the number of correct retrievals and the total number of retrievals is included in order to establish whether both groups had the same difficulty of retrieval from memory, and to verify whether participants indeed retrieve the instance correctly for 90% of the retrievals as indicated by Edge et al. (2012). The only relevant values are here the mean values, for the other values do not make sense.

Time spent Finally, the amount of time spent is calculated for each instance by the sum the retrieval time of each response, where the retrieval time is calculated by subtracting the time the instance was sent to the client from the time the instance was received on the server. The reason why this is chosen as the heuristic for time measurement is that this is the most intensive use of the system, whereas the times in between measure only the validating of an retieved answer or an intermission. Furthermore, the responses are filtered on only having retrieval times shorter than 5 minutes, since any longer retrieval time probably indicates an intermission between learning. For the same reason, the session time is also an unreliable measure. The absolute and mean values are here the most insightful, since the absolute values provide a measure of how much time the average user spent in total on the system, and the mean values are useful for comparing an individual flashcard with a flashmap.

Questionnaire scores

The last quantitative category contains the scores on the questionnaire, divided in the perceived usefulness score and the perceived ease of use score. Again, an item matrix is constructed, with the questionnaire items (either usefulness or ease of use) represented by the columns and the participants by the rows, and the cells containing how the participant rated the item. This score is determined by the rating from the positive phrasing minus the rating from the negative rating, which should result in twice the positive rating if the participant rated the negatively phrased item with the inverse rating of the positive phrased item. This result thereby ranges from -4 to 4.

Classical test theory

In order to establish Cronbach's alpha, classical test analyses are conducted over all of the above described statistics, using the CTT package from R (Robitzsch, Kiefer, & Wu, 2017). Furthermore, if Cronbach's alpha was lower than 0.7 — indicating a reliability classified by DeVellis (2003) and van Berkel and Bax (2006) as questionnable at best—, the items indicated by the package as the most weak items are omitted until the alpha is higher than 0.7 as described by van Berkel and Bax (2006) or participants are omitted since all of the items they responded to are omitted.

Item response theory

In order to provide an alternative measure of reliability, the EAP score from the TAM package in R (Willse, 2014) is calculated, based on item response theory (IRT). This serves the determination of the reliability of the tests and the reliability of the questionnaire. Additionally, this package calculates the person ability score (θ's) and the item difficulties as dependent items, in contrast to classical test theory which calculates both values independently. The 1 parameter logistics IRT model or the Rasch model is used to determine these variables over the 2PL or higher models, since the sample is expected to be too small to provide any reliable estimators for other variables than item difficulty (e.g. item discrimination). Finally, the TAM package supports defining fixed item difficulties, which is necessary in order to compare two sets of test scores. The item difficulties from the combined pretest matrices are used for setting the item difficulties when determining the IRT person abilities of the posttest matrices.

Descriptive statistics

For each category described above, the descriptive statistics will be provided as sample size, minimum value, maximum value, mean value, variance, the skew, and the kurtosis, the t- and p-values for the normality of the distribution (D'Agostino & Pearson, 1973), and the reliability from either classical test theory or item response theory as described in the previous sections.

Comparison tests

The tests used for comparing the control and experimental group are the Mann-Whitney U (Mann & Whitney, 1947) test and Welch's t-test (Welch, 1947). If the p-value for the normality test is significant ($p < 0.05$) for both distributions, the parametric Welch's t-test is used for the interpretation, otherwise the non-parametric Mann-Whitney U test is used.

Interviews

Since there is only one group interview to be analysed it is not deemed necessary to transcribe and encode the interview data (Baarda, de Goede, & Teunissen, 2009). Instead, only the relevant quotes are extracted and transcribed together with the time codes and added to the corresponding results within the next chapter.

Results

User participation

The Participation statistics appendix on page 142 shows statistics on how many days the different used the system. In table 5 it is shown that in total 63 students made an account within the system. From these students 44 used the system on at least one day for longer than 15 minutes, of which the average student participated on 4.68 days. The usage of the system is also depicted in figures 43 and 44. Finally, figures 45, 46 and 47 display on which days users have been actively using the system, where day 0 is the day the system was introduced within the presentation and day 21 the final day before the exam. Interesting to note here is that there are two subsequent strong increases in finished users around 6 days after the system was introduced, but that there is also a very strong increase the day before the students' exam. This is also reflected within a highly increased activity within the first week, and on the day before the exam. There is even some noticable activity after the exam took place, possibly of students already having invested some time into the system before the exam, but still finishing up in order to be rewarded with the icecream voucher.

This resulted in a total number of 25 finished users, of which 13 users within the flashcard condition and 12 users within the flashmap condition. Strange enough, both in the flashcard and the flashmap condition there is one user which did use the software for 6 days, but then did not partake in the posttest. They could be confused by the posttest (since it looks exactly the same as the pretest), and then decided to not fill it in. Finally, in figure 44 it can be seen that there are 4 students which used the system for only 5 days, which meant that they just missed out on the reward. For one student this is explained by the fact that she went somewhere else just before finishing, and thereby forgetting about it. General participation can be explained by students either not being motivated to continue, or by them forgetting about it. One student for example indicated that he almost forgot about the system for one day and barely made it by reviewing the flashcards in bed.

Within the interviews, it was gathered that students were mainly interested in participation, because they had problems preparing the previous exams from the same course and that they believed the system to help them being better prepared. The icecream voucher was only a secondary motivator, but a motivator nonetheless. One student indicated that the system might however be more successful within younger students, since the older students often already have their own learning systems and prefer to stick to these systems.

A sample of 23 divided over two conditions is too small for making any generalisations, and therefore any results stemming from this experiments are only indicatory and should be further investigated before they can be used. Since the only students usable for the rest of the result section are those finishing the posttest, the other students will be omitted from consideration.

Participant descriptives

The participant descriptives are included in the Descriptive statistics appendix on page V, containing distributions of student gender and age.

Gender As can be seen in figure 8, 15 out of the 23 total participants are male and 8 are female, where within the flashcards condition there is a 7 to 5 ratio and within the flashmap a 8 to 3 ratio. This is probably just coincidental due to the small sample size.

Age All students have an age within the range of 15 to 17 with an average age of 15.75 and a modus of 16, which is to be expected from VWO4 students. There is also no considerable age difference among the conditions, indicated in table 10 and figure 51.

Learning gain

The pre- and posttest statistics are displayed in the Pretest and posttest statistics appendix on page V, divided in the inter-rater reliablity statistics and the result scores on the test. As can be seen in table 11, Cohen's Kappa is 0.800 — with 1 response option indicated as met by the teacher and not met by the researcher and 3 response options indicated as vice versa —, which is generally seen as good (Baarda et al., 2009). However, it is relatively low in comparison to the proportional agreement of 0.944. This can be attributed to relatively few response options being indicated as met in general and the Cohen's Kappa having a base assumption of purely chance-based agreement.

The results are separately described for the knowledge questions and comprehension scores on the test, since they measure different variables. The different scores described and compared are the pretest scores, posttest scores, total scores, and the absolute (abs_learn_gain) and relative learning gains (rel_learn_gain). Additionally, they are reported as classical test theory scores (ctt), item response theory person abilities (irt), and person abilities from item response theory using fixed item difficulties from the combined pretest scores (fixed irt). Per category, the sample size, minimum, maximum, and mean values are displayed as descriptive values; the skew, kurtosis, and t and p values from the scipy normaltest are displayed as values for describing the distribution of the results; and α describes the reliability of the test (either Cronbach's alpha for the ctt results or the EAP value for the irt results). These results are described for the flashcard condition, the flashmap condition, and the combined sample of both conditions. The included graphs display histograms depicting the test matrices. Finally, the pre- and posttest scores are compared with each other by means of the non-parametric Mann-Whitney U test and the parametric Welchs' t-test in order to verify that users scored significantly higher on the posttest than on the pretest, and the learning gains between conditions are compared in order to answer research question Ia.

Remarkable are the low scores on the tests, with on average only 1.3 points on the knowledge questions (0.43 on the pretest and 2.17 on the posttest) and only 1.33 points on the comprehension questions (0.33 on the pretest and 2.33 on the posttest). This would indicate that the tests are exceptionally difficult, even though the questions are directly derived from the textbook itself and the users drilling these questions over the course of 6 days. Especially the low posttest knowledge question scores from the flashcard group are striking, since these questions were literally rehearsed during the experiment.

For both the knowledge questions as for the comprehension questions, the ctt reliability for the combined pre- and posttest score for the combined flashcard and flashmap users is around .7 — mainly because the omission process of unreliable items —, whereas the fixed irt reliability

	α	μ_{fc}	μ_{fm}	p
Knowledge				
abs-ctt	.721	1.25	2.27	.394
rel-ctt	.721	0.04	0.05	.464
irt	.671	-2.67	3.17	.000
Comprehension				
abs-ctt	.714	2.00	0.91	.218
rel-ctt	.714	0.07	0.04	.245
irt	.606	-1.28	-.97	.688

Table 2: Compact view of the results relevant for answering research question Ia

is around .6 (see table 14 and table 23). According to DeVellis (2003), this means that the results obtained from classical testing are acceptable, whereas the results obtained from the item response theory are questionnable at best. Additionally, both the score outcomes, the figures, and the pre- and posttest comparisons in tables 15, 16, 17, and tables 24, 25, 26 indicate an average positive learning gain from the classical test theory, but a negative gain from the item response theory. Therefore, the conclusion will be based on the results from the classical test theory only.

Table 2 summarises the results related to learning gains. In the rows, the absolute and the relative classical test scores and the absolute item response theory person ability scores with fixed item difficulties are included for both the knowledge and comprehension questions. The item response theory results are only included for reference, and from this only the absolute learning gains are taken into consideration, since the person abilities are already estimated relative to the item difficulties. the columns include the reliability, the p-value of the normality test, the flashcard and flashmap mean score, and the p-values for in this case the Mann-Whitnney U test, since none of the results seem to be normaly distributed.

The flashmap users seem to have a higher learning gain than the flashcard users on the knowledge questions, and that looking at only the ctt results they seem to have a lower gain on the comprehension questions. None of the Mann-Whitney U test p-values for the ctt results seem to be significant however, so no conclusions can be drawn yet. This is highly likely due to the low response rate, and more significant results might be found when using a larger sample, especially since the difference in mean values are relatively high in comparison to the variance in scores.

System use

In order to verify whether the users of the different conditions spent the same amount of effort and time on the system, answering research question Ib, the Instance statistics appendix on page 166 provides different statistics on the rehearsed instances. This entails the number of reviewed instances, the number of responses, the exponent from the instance.get_exponent() function (indicating how often the instance was retrieved correctly since the last incorrect retrieval), the percentage of correct retrievals in comparison to the total amount of retrievals, and finally the amount of time the users spent on the system. For every category, the descriptives (sample, minimum, maximum, mean, and variance), distribution (skew, kurtosis and normality test t- and p-values), and cronbach's alpha are displayed, both separate for the flashcard and flashmap condition and for the combined sample of users. It should be noted that in most cases the cronbach's alpha is not a sufficient measure for determining the reliability of the test, since there

	α	μ_{fc}	μ_{fm}	p
Reviewed instances				
abs	0.979	72.83	131.45	0.001
rel	0.979	0.78	0.66	0.188
Responses				
mean	0.938	7.61	5.61	0.024
Exponents				
mean	0.893	6.67	6.38	0.625
Correct retrievals				
mean	0.978	0.86	0.89	0.000
Time spent				
abs	0.924	12374.41	14121.58	0.000
mean	0.924	169.77	117.30	0.000

Table 3: Compact view of the results relevant for answering research question Ib

is a natural decrease in most statistics over the course of the instances, since the latter instances are repeated less often than the earlier statistics. The ratio of correct retrievals might be the only exception here, however one would still expect a higher ratio of correct retrievals in earlier instances than in later instances.

Furthermore, the absolute score, the relative score and the mean score are displayed for each condition, where the relative score is the absolute score divided by the total amount of either flashcards or edges. In the combined condition, the relative score is omitted, since the relative score is only useful for comparing the conditions.

Finally, the flashcard and flashmap conditions are again compared using the Mann-Whitney U test and Welch's t-test. Table 3 displays all results in a more compact manner.

Number of reviewed instances Within these statistics, the mean statistics are left out, since they only make sense when reviewing them over the entire range of available instances. Furthermore, the flashcards cannot simply be compared one on one with the edges, since some of the flashcards encompass multiple edges instead of merely representing one edge. This is most notably the case when multiple sibling edges are presented as one instance to the user, which are thereby also represented by only one flashcard. Therefore, the combined conditions table only shows the relative score, since this is the only score comparable across both conditions. This is also noticable in the large difference in absolute mean values of the flashcard and flashmap condition (72.83 vs 131.45) and the low p-value on the Mann-Whitney U test (0.001). When however purely looking at the relative score, the average user reviewed 70% of the available cards, where the flashcard users reviewed 78% of the cards and the flashmap users 66%. This difference is not yet significant ($p = .188$ on the Mann-Whiney U test). As can be seen in figure 69 on page 167, this difference is mainly due to both groups having reviewed about equal numbers of instances except for 2 users in the flashmap condition having reviewed less than 50% of the edges.

Number of responses These first statistics depict the total number of reviews of instances per user. On average, a user has 641.87 responses, where flashcard users have 561.33 and flashmap users have 729.73 responses. Both the Mann-Whitney U test as the Welch's t-test provide a relatively high p-value (.813 and .810) when looking at the difference in relative score, indicating no (significant) difference between the number of responses within the flashcard users and the flashmap users when adjusting for the different numbers of flashcards and edges.

Exponents These statistics are useful as an indicator of learning progress of the user, since the exponents are used for rescheduling the instance where a higher exponent indicates a longer time interval until the next review. The mean exponent for a reviewed instance was 6.53, where it was 6.67 for flashcard users and 6.38 for flashmap users. This difference is not significant ($p = 0.625$).

Correct retrievals The ratio of correct and total retrievals is also included, since this provides more insight in the effectiveness of the scheduling algorithm and the presentation form. For each reviewed instance, the ratio of correct and total retrievals is 0.87, where 0.86 for flashcard and 0.89 for flashmap users. This results in a small although significant ($p < 0.001$) difference in favour of the flashmap users. In the interviews however some students stated to not have noticed the instructions teaching how to mark retrievals to be either correct or incorrect, resulting in a 100% correct retrieval rate within the database (see also figure 79 on page 175).

Time spent The most efficient variable for determining the efficiency of the system is the amount of time spent by the user on the system. In total, the average user spent 13210 seconds (3 hours and 40 minutes) on the system, where the average flashcard user spent 12374 seconds (3 hours and 26 minutes) and the average flashmap user spent 14122 seconds (3 hours and 55 minutes). Per instance, this was on average 144.7 seconds, where 169.8 seconds for flashcard users and 117.3 seconds for flashmap users. Both these differences are significant ($p < 0.001$).

Interviews During the interviews it was learned that the students thought that some of the instances were reviewed too often, sometimes leading to frustration. Some of the students offered the suggestion to add a third validation for a longer scheduling interval, much like Anki already does. Others suggested an option to learn flashcards per section of the instructional material seperately instead of reviewing all the flashcards, so that they could focus on the lesser known or more important sections. Finally, one flashcard user stated that it was not necessary to read the instructional material at all for using the system, and that he learned the flashcards without reading. Other flashmap students however indicated that the book was necessary to comprehend the relations between the concepts.

User perceptions about the software

Both the question of how useful (research question IIa) as of how easy to use (research question IIb) the system is perceived to be is investigated quantitatively as qualitatively by means of the questionnaire, and open questions and the interviews. The quantitative results for both perceived usefulness as perceived ease of use are displayed in table 4.

Perceived usefulness

Questionnaire The perceived usefulness part of the questionnaire has a relatively low reliability ($\alpha = 0.651$), which makes drawing definite conclusions questionable according to DeVellis (2003). The average perceived usefulness score was 7.61 on a scale of -24 to 24, which indicates a slight

	α	μ_{fc}	μ_{fm}	p
Usefulness	0.651	6.50	8.82	0.245
Ease of use	0.829	6.58	8.27	0.482

Table 4: Compact view of the results relevant for answering research question II

positive score. The average flashcard user rated it as 6.50, and the average flashmap user as 8.82. These differences are however not significant ($p = 0.245$).

Open questions Within the open question asking what the user perceived to be good about the system, students mainly mentioned why the system helped them preparing for the exam. The responses could be categorised in 3 subcategories:

- the structure provided by the questions
- the distribution of learning over a period of time
- the repetition of questions

The first category contains statements about the questions hinting at the important aspects of the text ("It indicates the things that are important") and about the system providing a certain overview ("It was effective for learning the sequence of the history"). Statements such as "One learns a small portion every day instead of everything on one day" were placed in the second category. Finally, the third category contained statements such as "The repetition, which made the subject matter really sink in" or "The incorrect items were repeated so that these answers were learned better". However, the open question about what could be improved also contained certain comments about the repetition, mainly that certain questions were sometimes repeated too often. One comment also indicated that the questions were rather superficial. Next to these statements, some general comments were made such as "It was fun to do" or "It was convenient".

Interviews Within the interview, both flashcard and flashmap indicated that they thought of the tool itself as generally useful for learning textual material, aligning with the questionaire result. Respondents indicated the instances to be easier to comprehend than the textbook itself, and that they performed better on the final test than they normally did. When asked whether they would use a flashcard system again for another subject, they stated that they would definitely do this for other courses based on long texts such as history, but also for subjects like chemistry. One respondent even already used the flashcard system for music history, using the traditional system where every wrong card is rescheduled for today and all other cards are scheduled for the next day. They were also interested in hearing about alternative flashcard systems such as Anki. The flashmap users also stated that they liked the overview of the relations among concepts the map provided, however that this could also be a drawback since the user does not have to draw any connection for himself. The flashcards also did indicate that finding the relations between the concepts in the flashcards was not very difficult.

There were however some problems with the specific implementation. As already stated before, users thought that the instances were reviewed too often, resulting in a too high emphasis on the earlier instances. Furthermore, they did state that the system did not prepare them enough for the final test, mentioning the concept of "petrarkism" as an example. This concept was however present in both the flashcards as in the flashmap, but might have had a too low exposure. Finally, they stated that making the network or flashcards could be beneficiary, because then the relations or questions are better understood instead of only rehearsed.

Perceived ease of use

Questionnaire The perceived ease of use part of the questionnaire had a sufficiently high reliability ($\alpha = 0.829$), and is therefore reliable to be used for drawing conclusions. The average score for the perceived ease of use section was 7.39, which translates again to a slightly positive score. The difference between the flashcard and flashmap users is relatively small (6.58 for flashcard users and 8.27 for flashmap users), which is not significant ($p = 0.485$).

Open questions Most of the statements about what could be improved related to the ease of use of the system. Some flashmap users stated that certain flashmaps were too large and therefore unreadable, whereas others perceived zooming in and out on the flashmap to be inconveniant. One of the flashcard users indicated that the questions were sometimes unclearly formulated. Some of the users generally stated the instructions should be more clear, or that the indicators could be improved (e.g. “It was unclear how much one should learn on a day”). Finally, certain students perceived the system to be a bit boring because of the repetitive nature of drill and practice.

Interviews Within the interviews, the students stated that the system was generally easy to use, again aligning with the questionnaire results. One user however stated that the mobile version of the flashmap system was somewhat less user-friendly, since one had to zoom in and out in order to be able to read the larger concept maps on a smaller screen, and that users had to scroll down to show the answer to the instances. Some of the users did not even know that the flashmap was interactable. For flashcards this was not an issue. Users did agree generally that the instruction panel was quite unclear, either because the instructions were rather sparse, or because the users did not notice the panel or the text changing within the panel. This for example also resulted in users not indicating whether answers were retrieved correctly. They suggested a better introduction at the beginning of the use of the system. Furthermore, the learning progress overview was indicated to be either not very clear in the case of the flashcard system, or not very insightful in case of the flashmap system. Additionally, students mentioned the inconsisency in the graph layout, and that it was not always clear when they were finished for today with regards to the reward. Finally, forgetting to daily use the system was an issue for users, but also the fact that they had to use it daily was conceived as rather rigid.

Conclusion and discussion

Part IV

Recommendations

Part V

Backmatter

Epilogue

References

Ajzen, I. (1991). The theory of planned behavior. *Organizational Behavior and Human Decision Processes*, *50*(2), 179–211. doi: 10.1016/0749-5978(91)90020-T

Anderson, J. (1974). Retrieval of propositional information from long-term memory. *Cognitive Psychology*, *6*(4), 451–474. doi: 10.1016/0010-0285(74)90021-8

Anderson, J. (1982). Acquisition of cognitive skill. *Psychological Review*, *89*(4), 369–406. doi: 10.1037/0033-295X.89.4.369

Anderson, J. (2015). *Cognitive psychology and its implications* (8th ed.). New York, NY: Worth publishers.

Anderson, R. (2001). The power law as an emergent property. *Memory & Cognition*, *7*, 1061–1068. doi: 10.3758/BF03195767

Atkinson, R., & Shiffrin, R. (1968). The psychology of learning and motivation: Advances in research and theory. In K. Spence (Ed.), (pp. 89–195). New York: Academic Press.

Ausubel, D. (1968). *Educational psychology: A cognitive view.* New York, NY: Holt, Rinehart & Winston.

Baarda, D., de Goede, M., & Teunissen, J. (2009). *Basisboek kwalitatief onderzoek: handleiding voor het opzetten en uitvoeren van kwalitatief onderzoek.* Groningen/Houten: Noordhoff Uitgevers.

Barclay, J., Bransford, J. D., Franks, J. J., McCarrell, N. S., & Nitsch, K. (1974). Comprehension and semantic flexibility. *Journal of Verbal Learning and Verbal Behavior*, *13*(4), 471 - 481. Retrieved from `http://www.sciencedirect.com/science/article/pii/S0022537174800241` doi: http://dx.doi.org/10.1016/S0022-5371(74)80024-1

Barnes, C. (1979). Memory deficits associated with senescence: A neurophysiological and behavioral study in the rat. *Journal of Comparative Physiology*, *43*, 74–104.

Blakemore, S., Burnett, S., & Dahl, R. (2010). The role of puberty in the developing adolescent brain. *Human Brain Mapping*, *31*, 926–933. doi: 10.1002/hbm.21052

Blankenship, J., & Dansereau, D. (2000). The effect of animated node-link displays on information recall. *The Journal of Experimental Education*, *68*(4), 293–308. doi: 10.1080/00220970009600640

Bliss, T., & Collingridge, G. (1993). A synaptic model of memory: long-term potentiation in the hippocampus. *Nature*, *361*, 31–39. doi: 10.1038/361031a0

Bloom, B., Engelhart, M., Furst, E., Hill, W., & Krathwohl, D. (1956). *Taxonomy of educational objectives: The classification of educational goals. handbook i: Cognitive domain.* New York, NY: David McKay Company.

Burdo, J., & O'Dwyer, L. (2015). The effectiveness of concept mapping and retrieval practice as learning strategies in an undergraduate physiology course. *Advances in Physiology Education*, *39*, 335–340. doi: 10.1152/advan.00041.2015

Burgess, S., & Murray, A. (2014). Use of traditional and smartphone app flashcards in an introductory psychology class. *Journal of Instructional Pedagogies*, *13*, 1–7.

Cañas, A., & Novak, J. (2012). *Freedom vs restriction of content and structure during concept mapping: possibilities and limitations for construction and assessment.*

Chien, C. (2015). Analysis the effectiveness of three online vocabulary flashcard websites on l2 learners' level of lexical knowledge. *English Language Teaching*, *8*(5), 111–121. doi: 10.5539/elt.v8n5p111

Chung, G., O'Neil Jr., H., & Herl, H. (1999). The use of computer-based collaborative knowledge mapping to measure team processes and team outcomes. *Computers in Human Behaviour*, *15*(3–4), 463–493. doi: 10.1016/S0747-5632(99)00032-1

Cohen, J. (1968). Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological Bulletin*, *70*(4), 213–220. doi: 10.1037/h0026256

Craik, F. I., & Lockhart, R. S. (1972). Levels of processing: A framework for memory research. *Journal of Verbal Learning & Verbal Behavior*, *11*(6), 671–684. doi: 10.1016/S0022-5371(72)80001-X

D'Agostino, R., & Pearson, E. (1973). Tests for departure from normality. *Biometrika*, *60*, 613–622. doi: 10.2307/2335012

Davis, F. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, *13*(3), 319–340. doi: 10.2307/249008

Davis, F., Bagozzi, R., & Warshaw, P. (1989). User acceptance of computer technology: A comparison of two theoretical models. *Management Science*, *35*(8), 982–1003. doi: 10.1287/mnsc.35.8.982

DeVellis, R. (2003). *Scale development: Theory and applications* (2nd ed.). London: Sage publications Ltd.

Dirksen, J. (2007). Leerlingen, literatuur en literatuuronderwijs. *Forum of arena: opvattingen over literatuuronderwijs. Een stand van zaken in 2007*. Retrieved from `http://taalunieversum.org/inhoud/von-cahier-1/leerlingen-literatuur-en-literatuuronderwijs`

Edge, D., Fitchett, S., Whitney, M., & Landay, J. (2012). *Memreflex: Adaptive flashcards for mobile microlearning.* doi: 10.1145/2371574.2371641

Eppler, M. (2006). A comparison between concept maps, mind maps, conceptual diagrams, and visual metaphors as complementary tools for knowledge construction and sharing. *Information Visualization*, *5*, 202–210. doi: 10.1057/palgrave.ivs.9500131

Ericsson, K., & Kintsch, W. (1995). Long-term working memory. *Psychological Review*, *102*(2), 211–245. doi: 10.1037/0033-295X.102.2.211

Giedd, J. N., Blumenthal, J., Jeffries, N. O., Castellanos, F. X., Liu, H., Zijdenbos, A., ... Rapoport, J. L. (1999). Brain development during childhood and adolescence: a longitudinal mri study. *Nature neuroscience*, *2*(10), 861–863. doi: 10.1038/13158

Golding, J., Wasarhaley, N., & Fletcher, B. (2012). The use of flashcards in an introduction to psychology class. *Teaching of Psychology*, *39*(3), 199–202. doi: 10.1177/0098628312450436

Grever, M., Pelzer, B., & Haydn, T. (2011). High school students' views on history. *Journal of Curriculum Studies*, *43*(2), 207–229. doi: 10.1080/00220272.2010.542832

Hulstijn, J. (2001). *Intentional and incidental second-language vocabulary learning: A reappraisal of elaboration, rehearsal and automaticity.* Cambridge University Press.

Hwang, G., Wu, P., & Ke, H. (2011). An interactive concept map approach to supporting mobile learning activities for natural science courses. *Computers and Education*, *57*, 2272–2280. doi: 10.1016/j.compedu.2011.06.011

Joseph, L., Eveleigh, E., Konrad, M., Neef, N., & Volpe, R. (2012). Comparison of the efficiency of two flashcard drill methods on children's reading performance. *Journal of Applied School Psychology*, *28*(4), 317–337. doi: 10.1080/15377903.2012.669742

Karpicke, J. (2012). Retrieval-based learning: Active retrieval promotes meaningful learning. *Current Directions in Psychological Science*, *21*(3), 157–163. doi: 10.1177/0963721412443552

Karpicke, J., & Bauernschmidt, A. (2011). Spaced retrieval: Absolute spacing enhances learning regardless of relative spacing. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *37*(5), 1250–1257. doi: 10.1037/a0023436

Karpicke, J., & Blunt, J. (2011). Retrieval practice produces more learning than elaborative studying with concept mapping. *Science*, *331*, 772–775. doi: 10.1126/science.1199327

Keller, J. (2000). *How to integrate learner motivation planning into lesson planning: The arcs model approach.* Retrieved from http://apps.fischlerschool.nova.edu/toolbox/instructionalproducts/itde8005/weeklys/2000-Keller-ARCSLessonPlanning.pdf

Kintsch, W., Welsch, D., Schmalhofer, F., & Zimny, S. (1990). Sentence memory: A theoretical analysis. *Journal of Memory and Language*, *29*, 133–159.

Kornell, N., & Bjork, R. (2008). Optimising self-regulated study: The benefits - and costs - of dropping flashcards. *Memory*, *16*(2), 125–136.

Krathwohl, D. (2002). A revision of bloom's taxonomy: an overview. *Theory into practice*, *41*(4), 212–264.

Kwaliteitsonderzoek in het kader van het onderwijsverslag 2016, het stedelijk lyceum - locatie kottenpark, havo, vwo (Tech. Rep.). (2015). Inspectie van het Onderwijs. Retrieved from https://zoekscholen.onderwijsinspectie.nl/pdf/arrangement.pdf?pdfId=A0000333800

Leerlingen, deelnemers en studenten; onderwijssoort, woonregio (Tech. Rep.). (2016). CBS. Retrieved from http://statline.cbs.nl/Statweb/publication/?DM=SLNL&PA=71450ned&D1=0&D2=a&D3=8-9&D4=0,2&D5=a&D6=251&D7=l&HDR=T,G4,G6,G5&STB=G3,G1,G2&CHARTTYPE=1&VW=T ([Data set])

Logan, J., Castel, A., Haber, S., & Viehman, E. (2012). Metacognition and the spacing effect: the role of repetition, feedback, and instruction on judgments of learning for massed and spaced rehearsal. *Metacognition Learning*, *7*, 175–195. doi: 10.1007/s11409-012-9090-3

Macquarrie, L., Tucker, J., Burns, M., & Hartman, B. (2002). Comparison of retention rates using traditional, drill sandwich, and incremental rehearsal flash card methods. *School psychology review*, *31*(4), 584–595.

Mann, H. B., & Whitney, D. R. (1947, 03). On a test of whether one of two random variables is stochastically larger than the other. *The Annals of Mathematical Statistics*, *18*(1), 50–60. Retrieved from http://dx.doi.org/10.1214/aoms/1177730491 doi: 10.1214/aoms/1177730491

Matsaridis, G. (2013, April). *Neuroscience for beginners v2.0: Lesson 2 - the materialistic mind - your brain's ingredients.* Retrieved from http://neuroscientist.weebly.com/blog/lesson-2-the-materialistic-mind-your-brains-ingredients (Retrieved September 9, 2016)

Mayer, R. (2008). *Learning and instruction* (2nd ed.). Upper Saddle River, NJ: Pearson Education.

McCullough, C. (1955). Flash cards - the opiate of the reading program? *Elementary English*, *32*(6), 39–381.

Moore, J., North, C., Johri, A., & Paretti, M. (2015). Effectiveness of adaptive concept maps for promoting conceptual understanding: Findings from a design-based case study of a learner-centered tool. *Advances in Engineering Education*, *4*(4), 1–35.

Murre, J., & Chessa, A. (2011). Power laws from individual differences in learning and forgetting: mathematical analyses. *Psychonomic bulletin & review*, *18*(3), 592–597. doi: 10.3758/s13423-011-0076-y

Nakata, T. (2011). Computer-assisted second language vocabulary learning in a paired-associate paradigm: a critical investigation of flashcard software. *Computer Assisted Language Learning*, *24*(1), 17–38. doi: 10.1080/09588221.2010.520675

Nelson, D. L. (1979). Remembering pictures and words: Appearance, significance, and name. In

L. S. Cermak & F. I. M. Craik (Eds.), *Levels of processing in human memory* (p. 45-76). Hillsdale, NJ: Lawrence Erlbaum Associates.

Nesbit, J., & Adesope, O. (2006). Learning with concept and knowledge maps: A meta-analysis. *Review of Educational Research*, *76*(3), 413–448. doi: 10.3102/00346543076003413

Newell, A., & Rosenbloom, P. (1981). Mechanisms of skill acquisition and the law of practice. In (Vol. 1, pp. 1–55).

Nieveen, N., Folmer, E., & Vielgen, S. (2012). *Evaluation matchboard.* Enschede: SLO.

Novak, D., & Cañas, A. (2008). *The theory underlying concept maps and how to construct and use them* (Tech. Rep.). Florida: Institute for Human and Machine Cognition.

(OMG), O. M. G. (2015). *Omg unified modeling language reference manual* (2.5 ed.). OMG Document Number formal/2015-03-01 (`http://www.omg.org/spec/UML/2.5/PDF`).

Pavlik, I., & Anderson, J. (2005). Practice and forgetting effects on vocabulary memory: an activation-based model of the spacing effect. *Cognitive Science*, *29*(4), 559–586. doi: 10.1207/s15516709cog0000_14

Plomp, T., Feteris, A., & Pieters, J. (1992). *Ontwerpen van onderwijs en trainingen* (W. Toic, Ed.). Utrecht: LEMMA.

Powell, K. (2006). Neurodevelopment: How does the teenage brain work? *Nature*, *442*, 865–867. doi: 10.1038/442865a

Raymond, C., & Redman, S. (2006). Spatial segregation of neuronal calcium signals encodes different forms of ltp in rat hippocampus. *Journal of Physiology*, *570*, 97–111. doi: 10.1113/jphysiol.2005.098947

Religie; naar regio; 2000/2002 of 2003 (Tech. Rep.). (2004). CBS. Retrieved from `http://statline.cbs.nl/Statweb/publication/?DM=SLNL&PA=70794NED&D1=a&D2=0,8,28&D3=a&VW=T` ([Data set])

Religieuze betrokkenheid; kerkelijke gezindte; regio (Tech. Rep.). (2015). CBS. Retrieved from `http://statline.cbs.nl/Statweb/publication/?DM=SLNL&PA=83288NED&D1=a&D2=0%2c2%2c8&D3=l&HDR=T&STB=G1%2cG2&CHARTTYPE=2&VW=T` ([Data set])

Robitzsch, A., Kiefer, T., & Wu, M. (2017). Tam: Test analysis modules [Computer software manual]. Retrieved from `https://CRAN.R-project.org/package=TAM` (R package version 2.1-43)

Rohrer, D., Taylor, K., Pashler, H., Wixted, J., & Cepeda, N. (2005). The effect of overlearning on long-term retention. *Applied Cognitive Psychology*, *19*, 361–374. doi: 10.1002/acp.1266

Ruiz-Primo, M., & Shavelson, R. (1996). Problems and issues in the use of concept maps in science assessment. *Journal of Research in Science Teaching*, *33*(6), 569–600.

Schoonenboom, J. (2014). *Points of addition and points of integration: Why one point of interface is not enough.* Philadelphia.

Siegel, L. L., & Kahana, M. J. (2014). A retrieved context account of spacing and repetition effects in free recall. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *40*(3), 755–764.

Singh, I., & Moono, K. (2015). The effect of using concept maps on student achievement in selected topics in chemistry at tertiary level. *Journal of Education and Practice*, *6*(15), 106–116.

Slings, H. (2007). Het waarom en hoe van historisch literatuuronderwijs. *Forum of arena: opvattingen over literatuuronderwijs. Een stand van zaken in 2007*. Retrieved from `http://taalunieversum.org/inhoud/von-cahier-1/het-waarom-en-hoe-van-historisch-literatuuronderwijs`

Smith, P., & Ragan, T. (2005). *Instructional design* (3rd ed.). Westford, MA: John Wiley & Sons, Inc.

Squire, L. (1987). *Memory and brain.* New York, NY: Oxford university press.

Subramaniam, K., & Esprivalo Harrell, P. (2015). An analysis of prospective teachers' knowledge for constructing concept maps. *Educational research*, *57*(3), 217–236.

Tzeng, J. (2010). Designs of concept maps and their impacts on readers' perfomance in memory and reasoning while reading. *JOurnal of Research in Reading*, *33*(2), 128–147. doi: 10.1111/j.1467-9817.2009.01404.x

van Berkel, H., & Bax, A. (2006). *Toetsen in het hoger onderwijs* (2nd revised ed.). Houten: Bohn Stafleu van Loghum.

van der Meulen, G., & Kraaijeveld, R. (2010). *Laagland. literatuur nederlands voor de tweede fase. informatieboek vwo* (2nd ed.). Amersfoort: ThiemeMeulenhoff.

Verkoeijen, P., & Delaney, P. (2008). Rote rehearsal and spacing effects in the free recall of pure and mixed lists. *Journal of Memory and Language*, *58*, 35–47. doi: 10.1016/j.jml.2007.07.006

von Glaserfield, E. (2001). Radical constructivism and teaching. *Prospects*, *31*(2), 161–173. doi: 10.1007/BF03220058

Wahlheim, C., Maddox, G., & Jacoby, L. (2014). The role of reminding in the effects of spaced repetitions on cued recall: Sufficient but not necessary. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *40*(1), 94–105. doi: 10.1037/a0034055

Welch, B. L. (1947). The generalization of student's problem when several different population varlances are involved. *Biometrika*, *34*(1-2), 28. doi: 10.1093/biomet/34.1-2.28

White, A. (2003). What happened? alcohol, memory blackouts and the brain. *Alcohol research and health*, *27*(2), 186–196. doi: 10.3390/ijerph6112783

Wickelgren, W. (1974). Single-trace fragility theory of memory dynamics. *Memory & Cognition*, *2*, 775–780. doi: 10.3758/BF03198154

Willse, J. (2014). Ctt: Classical test theory functions [Computer software manual]. Retrieved from `https://CRAN.R-project.org/package=CTT` (R package version 2.1)

Wixted, J., & Carpenter, S. (2007). The wickelgren power law and the ebbinghaus savings function. *Psychological Science*, *18*(2), 133–134. doi: 10.1111/j.1467-9280.2007.01862.x

Wixted, J., & Ebessen, E. (1991). On the form of forgetting. *Psychological Science*, *2*(6), 409–415. doi: 10.1111/j.1467-9280.1991.tb00175.x

Xue, G., Mei, L., Chen, C., Lu, Z.-L., Poldrack, R., & Dong, Q. (2011). Spaced learning enhances subsequent recognition memory by reducing neural repetition suppression. *Journal of Cognitive Neuroscience*, *23*(7), 1624–1633.

Zhang, P., & von Dran, G. (2000). Satisfiers and dissatisfiers: A two factor theory for website design and evaluation. *Journal of the American society for information science*, *51*(14), 1253–1268. doi: 10.1002/1097-4571(2000)9999:9999⟨::AID-ASI1039⟩3.0.CO;2-O

Zirkle, D., & Ellis, A. (2010). Effects of spaced repteition on long-term map knowledge recall. *Journal of Geography*, *109*(5), 201–206. doi: 10.1080/00221341.2010.504780

Appendices

Test literature history 16th and 17th century

This is not the test used as pre- and posttest within the research, but a test provided to the previous generation of students provided by the teacher.

1. Provide a Dutch word for the term ‘renaissance’. Furthermore, explain the central idea of the renaissansistic body of thoughts.
2. Indicate whether the following statements are ‘true’ or ‘false’:
   (a) The renaissance originated in the Northern and Central Italian republican citystates.
   (b) The literature from the renaissance is only a revival of classical genres.
   (c) The eventual goal of renaissance writers is imitatio.
   (d) An amount of great playwriters from the renaissance is literarily schooled within a chamber of rhetoric [*rederijkerskamer*].
3. What is the essence of humanism?
4. Read the citation below: [...]
   (a) What is the title of the book from which this citation originates and who wrote this book?
   (b) What was the goal of writing this book and why is the book still attractive to read?
5. (a) What was the reason for writing the Dutch Authorised Version of the bible [*Statenbijbel*]?
   (b) Some of the expressions we are still using come from the Dutch Authorised Version of the bible. Why was this bible, generally stated, so important for language in that time?
6. Except of imitatio writers used two other methods. Enlist the three methods in the right order and provide a description for each.
7. (a) Which poet is the great example for those who write lovepoetry in this era?
   (b) Explain what platonic love is and what this has to do with lovepoetry from question 7a and with the adjoining picture [see figure 22].
8. Which combination of terms best displays the central ideas from renaissance literature?
   (a) Learning and pleasure

   (b) Antiquity and church
   (c) Love and antiquity
   (d) Church and pleasure

9. Provide for each of the genres of theater (tragedy, comedy, and farce [*klucht*]) a name of a matching writer and the title of a matching play.

10. Provide two differences between a tragedy and a farce.

11. Quickly after Willem-Alexander became king of the Netherlands, he visited different provinces together with Máxima. The province of Drenthe gave a small book on the occasion of this visit, containing among else the following poem: [...]
    (a) This poem is a sonnet. Enlist the characteristics of a sonnet regarding the form and content.
    (b) Explain how the content-related characteristic is included in the poem above.
    (c) Who was our most important sonnet writer in the 17th century?

12. (a) What is the goal of *emblematiek*? Include the term 'analogical thinking' in your answer.
    (b) Of which three parts does an emblem exist? Use the original terms/names.

13. View and read the emblem below [see figure 23] and conduct the following assignments:
    (a) Explain in your own words which analogy is made in the emblem and which lesson the writer wants to teach the reader.
    (b) With which word from the original emblem does the analogy start?

14. In the children series 'Dappere Dodo', 75 episodes were broadcasted on the Dutch TV between 1955 and 1964. The programm revolved around *Dappere Dodo*, who together with his friends Kees, Uncle Harrie, the captain, Grandfather Buiswater and Mrs Vulpen sailed around the world and experienced all kinds of adventure. 'Dodo' is in this series an appropriate name for the main person. Provide a good explanation for this.

15. The shipsjournal of Bontekoe went up in flames during a shipboard fire. Why did he write the journal again after the sea journey?

16. What does the Meertens Institute? It occupies itself with:
    (a) the study and documentation of Dutch language varation and folk culture
    (b) research into and documentation of European language and culture
    (c) collecting and documenting songs, specific from the period of the 16th and 17th century
    (d) research into dialects and socilects in European context.

Figure 22: The figure accompanying question 7b

Figure 23: The figure accompanying question 13. This figure was accompanied by the following text: “Soo lang de Roe wanckt. Veel mensche zijn deughdelijck, soo langh zy onder het kruys en verdruckinghe leven: maer als de Roede van den eers is, soo worden zy luy in den dienste Goods. Ghelijuck enen Drijf-tol, die niet meer gheslaghen of ghegispt en wort, die valt haest in onmacht ende blijft ligghen. Uit: Roemer Visscher, Sinnepoppen.” In the original test a modern Dutch translation was also provided.

Flashmap server Documentation

Release 1.0

M.C. van den Enk

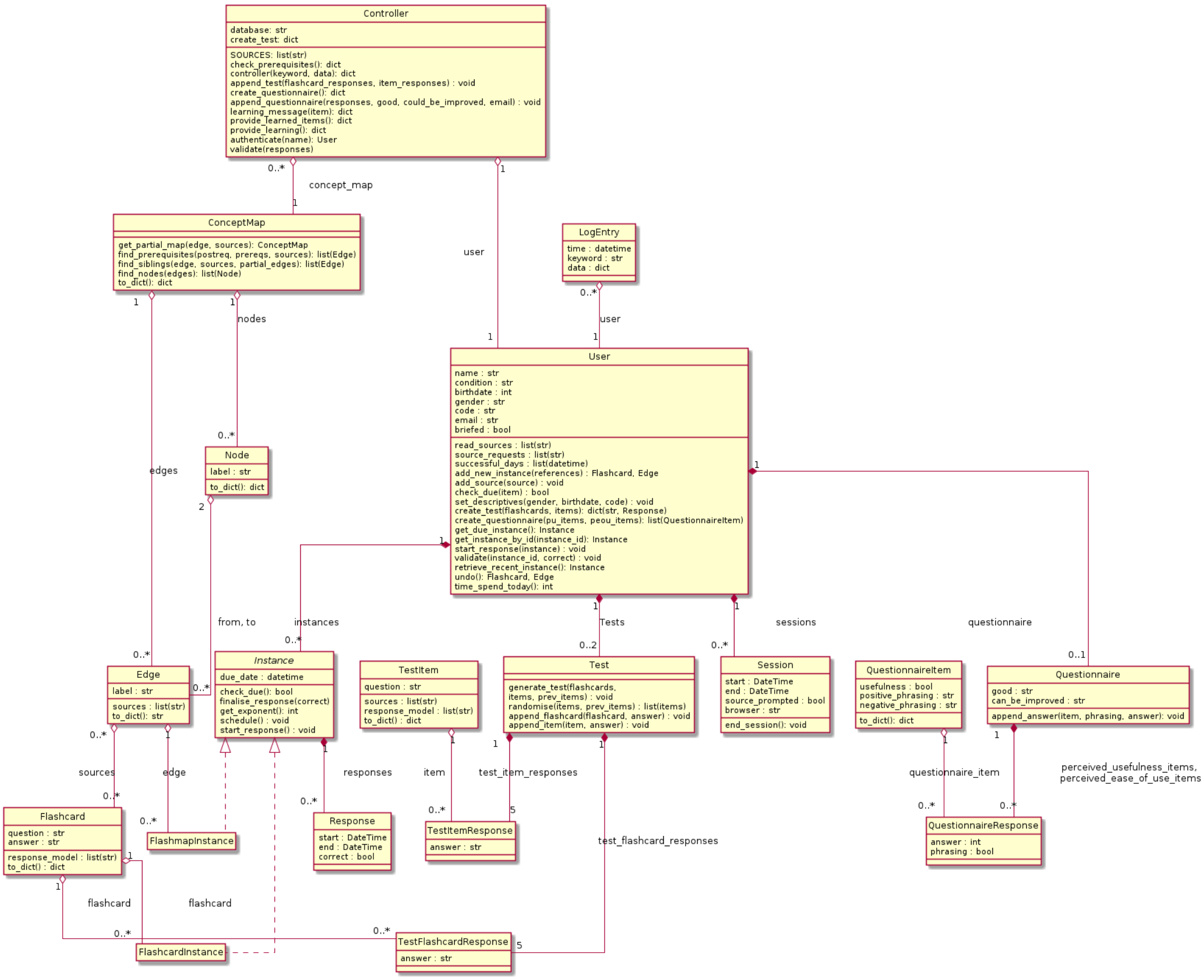
May 27, 2017

CONTENTS

CHAPTER

ONE

CLASS DIAGRAM



CHAPTER

TWO

MODULES:

2.1 concept_map module

class `concept_map.`**`ConceptMap`**(**args*, ***values*)

Bases: `mongoengine.document.Document`

A class representing a concept map

Variables

- **`nodes`** – a list of nodes (by default all existing node documents)
- **`edges`** – a list of edges (by default all existing edge documents)

`find_nodes`(*edges*)

Returns the from and to self.nodes given a list of self.edges

Parameters **`self.edges`** (*`list (Edge)`*) – The list of self.edges for which to find the self.nodes

Returns The list of nodes referred to in the edges

Return type list(*Node*)

`find_prerequisites`(*postreq*, *prereqs*, *sources*)

Return a list of parent self.edges given a certain edge from a list of self.edges, filtered by a list of sources

Parameters

- **`postreq`** (`Edge`) – The edge which is currently investigated for parent self.edges
- **`prereqs`** (*`list (Edge)`*) – A list of already found parent self.edges (starts usually empty, necessary for recursion)
- **`sources`** (*`list (string)`*) – A list of the currently read sources, self.edges which have a source not included in this list will not be included in the resulting list

Returns A list of self.edges which are prerequisites from edge

Return type list(*edge*)

`find_siblings`(*edge*, *sources*, *partial_edges*)

Return a list of self.edges which are siblings of the given edge and have the same label

Parameters

- **`edge`** (`Edge`) – The edge investigated for siblings
- **`sources`** (*`list (string)`*) – The sources to filter on when looking for siblings

- **partial_edges** (*list (*Edge*)*) – A list of self.edges for exclusion when looking for siblings

Returns A list of edges which are siblings of edge and have the same label

Return type list(*edge*)

get_partial_map (*edge*, *sources*)
Returns a concept map containing only the parent and sibling self.edges together with the referred self.nodes

Parameters

- **edge** (Edge) – The input edge
- **sources** (*list (string)*) – The list of sources to filter on

Returns A concept map containing parent and sibling self.edges of edge together with the referred self.nodes

Return type *ConceptMap*

to_dict ()
Returns a dictionary representation of this object

The representation is compatible for use with vis.js, with 'self.nodes' entries containing an 'id' and 'label', and 'self.edges' entries containing an 'id', 'label', 'from', 'to', and an additional 'source' entry

Result The dictionary representation, compatible with visjs

Return type dict

2.2 controller module

class controller.**Controller** (*database*)
Bases: object

This is the class from which the program is controlled. It can be used together with the handler module in order to communicate with an external client over a websocket

Variables

- **database** – The mongodb to connect to
- ***concept_map*** – The concept map object containing references to nodes and edges
- **SOURCES** – All of the sources referenced to in the edges of the concept map
- ***user*** – The active user

append_questionnaire (*responses*, *good*, *can_be_improved*, *email*)
A method for appending a questionnairy to the user given responses

Parameters

- **responses** (*list (dict)*) – A list of dict objects containing the id of a QuestionnaireItem (key = 'id'), the phrasing (key = 'phrasing') and an answer (key = 'answer')
- **good** (*string*) – A description of what was good about the software according to the user
- **can_be_improved** (*string*) – A description of what can be improved about the software according to the user

append_test (*flashcard_responses*, *item_responses*)
A method for appending a test to the user given flashcard and item responses

Parameters

- **flashcard_responses** (*dict*) – A list of dict objects containing the id of an `Flashcard` (key = 'id') and an answer (key = 'answer')
- **item_responses** (*dict*) – A list of dict objects containing a `TestItem` (key = 'item') and an answer (key = 'answer')

authenticate (*name*)
A function to either set self.user to an existing `user.User` or to a new User based on the given name

Parameters name (*str*) – The self.username

check_prerequisites ()
Checks whether the self.user still has to fill in forms and returns the appropriate message

Returns A dict containing the appropriate keyword and data for this self.user

Return type dict

controller (*keyword*, *data*)
Pass data to the function corresponding to the provided keyword for the provided user

Parameters

- **keyword** (*str*) – the keyword for which function to use
- **data** (*dict (str, str or dict)*) – the data necessary for executing the function

Returns Contains the keyword and data to send over a websocket to a client

Return type dict(str, str or dict)

learning_message (*item*)
Generates a learning message for the provided instance

Parameters instance (`Instance`) – The instance which has to be rehearsed

Returns The message with keyword "LEARNING RESPONSE" and data containing the partial concept map or flashcard dict representation

Return type dict

provide_learned_items ()
Provides an overview of all learning

Returns A partial concept map containing all instances for this self.user or a message containing progress information

Return type dict

provide_learning ()
Provides a dict containing relevant information for learning

Provides a dict containing the keyword "NO_MORE_INSTANCES", "READ_SOURCE-REQUEST", or "LEARNING-RESPONSE" and relevant data (the source string for "READ_SOURCE-REQUEST" or either the output of `ConceptMap.to_dict()` with an added 'learning' entry or the output of `Flashcard.to_dict()` for "LEARNING-RESPONSE" with an added condition entry)

Returns A dict containing 'keyword' and the relevant 'data' described above

Return type dict

validate(*responses*)
Adds responses to certain instances

Parameters **responses** (*list (dict)*) – A list of responses containing an instance id and a boolean correctness value

2.3 edge module

class edge.**Edge**(**args*, ***values*)
Bases: `mongoengine.document.Document`

A class representing an edge from a concept map

Variables

- **from_node** – The parent node of the edge
- **to_node** – The child node of the edge
- **label** – A label describing the relation between from_node and to_node
- **sources** – The source where this edge is described (e.g. paragraph 13.2 from Laagland)

to_dict()
Returns a dictionary representation of this object

It contains an 'id', 'label', 'from', 'to', and 'sources' entry

Returns The dictionary representation of this object, compatible with visjs

Return type dict

2.4 flashcard module

class flashcard.**Flashcard**(**args*, ***values*)
Bases: `mongoengine.document.Document`

A class representing a flashcard

Variables

- **question** – The question on the front side of the flashcard
- **answer** – The answer on the back side of the flashcard
- **sources** – The edges where this flashcard is based on
- **response_model** – A list consisting of parts of valid responses to the question (for the test matrix)

to_dict()
Returns a dictionary representation of this object

It contains an 'id', 'question', 'answer', and 'sources' entry

Returns The dictionary representation of this object

Return type dict

2.5 flashcard_instance module

class `flashcard_instance.`**`FlashcardInstance`**(**args*, ***kwargs*)

Bases: *`instance.Instance`*

A class for storing responses from the flashmap system

Variables **`reference`** – The flashcard to which this instance refers

2.6 flashmap_instance module

class `flashmap_instance.`**`FlashmapInstance`**(**args*, ***kwargs*)

Bases: *`instance.Instance`*, `mongoengine.document.EmbeddedDocument`

A class for storing responses from the flashmap system

Variables **`reference`** – The edge from the concept map to which this instance refers to

2.7 instance module

class `instance.`**`Instance`**(**args*, ***kwargs*)

Bases: `mongoengine.document.EmbeddedDocument`

A class describing a general flash instance, which can either be a FlashmapInstance or a FlashcardInstance

Variables

- **`responses`** – A list of responses provided to this instance (an empty list by default)
- **`reference`** – A reference to either an edge in a concept map or a flashcard (defined within the subclass)
- **`due_date`** – The date this instance is due for repetition

`check_due`()

Checks whether this instance is due for repetition

Returns Whether the due datetime is earlier than the current datetime

Return type bool

`finalise_response`(*correct*)

Sets the correctness value for the final response and sets the end date to now

Parameters **`correct`** (*`bool`*) – Whether the response was correct

`get_exponent`()

Determines the exponent for the rescheduling of this instance

Returns The amount of times this instance was answered correctly since the previous incorrect answer

Return type int

`schedule`()

Reschedules this instance for review based on the previous responses

`start_response`()

Adds a new response to this instance

2.8 log_entry module

class `log_entry.LogEntry`(**args*, ***values*)

Bases: `mongoengine.document.Document`

An object representing a incoming or outgoing network message

Variables

- `user` – The user which was involved with this network message
- `keyword` – The network keyword
- `data` – The dictionary containing the necessary data
- `time` – The time that this message was received or transmitted

2.9 node module

class `node.Node`(**args*, ***values*)

Bases: `mongoengine.document.Document`

A class for representing nodes in the concept map

Variables `label` – The label appearing within the node

`to_dict`()

Returns a dictionary representation of this object

It contains an 'id' and 'label' entry

Returns The dictionary representation of this object, compatible with visjs

Return type dict

2.10 questionnaire module

class `questionnaire.Questionnaire`(**args*, ***kwargs*)

Bases: `mongoengine.document.EmbeddedDocument`

A class representing a stored questionnaire for a user

Variables

- `perceived_usefulness_items` – Responses to the perceived usefulness items from TAM
- `perceived_ease_of_use_items` – Responses to the perceived ease of use item from TAM
- `good` – A description of what was good about the software according to the user
- `can_be_improved` – A description of what could be improved according to the user

`append_answer`(*item*, *phrasing*, *answer*)

Appends an answer to an item within the questionnaire

Parameters

- `item` (`QuestionnaireItem`) – The item to which the answer refers

- **phrasing** (*bool*) – Whether the item is positively (True) phrased or negatively (False)
- **answer** (*string*) – The answer to be appended

generate_questionnaire (*pu_items*, *peou_items*)
A method to set the questionnaire items based on two sets of items

Parameters

- **pu_items** (*list (QuestionnaireItem)*) – The perceived usefulness items of TAM
- **pu_items** – The perceived ease of use items of TAM

2.11 questionnaire_item module

class questionnaire_item.**QuestionnaireItem** (**args*, ***values*)
Bases: mongoengine.document.Document

A class representing a single item on the questionnaire

Variables

- **usefulness** – Defines whether the item is part of the perceived usefulness items (True) or of the perceived ease of use items (False)
- **positive_phrasing** – The version of this item which is positively phrased
- **negative_phrasing** – The version of this item which is negatively phrased

to_dict (*phrasing*)
A method for generating a dictionary representation of this object

Parameters **phrasing** (*bool*) – Whether the positive or negative question is required

Returns The representation containing an id field, a phrasing field and a question field

Return type dict

2.12 questionnaire_response module

class questionnaire_response.**QuestionnaireResponse** (**args*, ***kwargs*)
Bases: mongoengine.document.EmbeddedDocument

A class for storing singular responses to questionnaire items

Variables

- ***questionnaire_item*** – The questionnaire item to which this answer refers
- **answer** – The value of the likert-scale rating the user gave to this item (ranges from -2 to 2)
- **phrasing** – Whether this answer refers to the positively (True) or the negatively (False) phrased version of the questionnaire_item

2.13 response module

class response.**Response**(**args*, ***kwargs*)
Bases: mongoengine.document.EmbeddedDocument

A class representing a singular response to an Instance.

Variables

- **start** – The moment the parent Instance was sent to the client
- **end** – The moment the answer from the client was received
- **correct** – Whether the answer to the Instance was correct (True) or incorrect (False)

2.14 session module

class session.**Session**(**args*, ***kwargs*)
Bases: mongoengine.document.EmbeddedDocument

A class representing a session the user was logged in

Variables

- **start** – The time that the user logged in
- **end** – The time that the user logged out
- **source_prompted** – Whether the user was asked to have read a certain source from SOURCES
- **browser** – The type of browser used to log in

end_session()
Closes this session

2.15 test module

class test.**Test**(**args*, ***kwargs*)
Bases: mongoengine.document.EmbeddedDocument

A class representing a pre- or posttest the user filled in

Variables

- **test_flashcard_responses** – A list of responses to the flashcard questions on the test
- **test_item_responses** – A list of responses to the item questions on the test

append_flashcard(*flashcard*, *answer*)
Adds a flashcard response to this test

Parameters

- **flashcard** (Flashcard) – The flashcard this item refers to
- **answer** (*string*) – The answer to the flashcard provided by the user

append_item (*item*, *answer*)
Adds an item response to this test

Parameters

- **item** – The test item this item refers to
- **answer** (*string*) – The answer to the flashcard provided by the user

generate_test (*flashcards*, *items*, *prev_flashcards=[]*, *prev_items=[]*)
A method for creating test items for this test based on a set of given flashcards and items, using randomise()

Parameters

- **flashcards** (*list (Flashcard)*) – The flashcards to be used for the test
- **items** (*list (Item)*) – The items to be used for the test
- **prev_flashcards** (*list (Flashcard)*) – The list of flashcards to be excluded from this test
- **prev_items** (*list (TestItem)*) – The list of items to be excluded from this test

randomise (*items*, *prev_items*)
A method for taking five random items in a random order from the provided list of items without the items in the previous items

Parameters

- **items** (*list (Flashcard) or list (TestItem)*) – The complete list of items
- **prev_items** (*list (Flashcard) or list (TestItem)*) – The list of items to be excluded from the result

Result A sample of five items from items not included in prev_items

Return type list(FlashcardResponse) or list(*TestItemResponse*)

2.16 test_flashcard_response module

class test_flashcard_response.**TestFlashcardResponse** (**args*, ***kwargs*)
Bases: mongoengine.document.EmbeddedDocument

An answer for a flashcard item within a pre- or posttest

Variables

- **answer** – The answer provided by the user
- **reference** – The flashcard to which this response refers to
- **scores** – The list of correct response elements in the answer

2.17 test_item module

class test_item.**TestItem** (**args*, ***values*)
Bases: mongoengine.document.Document

A class representing an item from a pre- or posttest

Variables

- **question** – The question for this item
- **sources** – A list of sources relevant to this question
- **response_model** – A list of the parts of a valid answer used for the test matrix

to_dict()
A method for generating a dictionary representation of this object

Returns The representation containing an id field and a question field

Return type dict

2.18 test_item_response module

class test_item_response.**TestItemResponse**(**args*, ***values*)
Bases: mongoengine.document.Document

A class representing singular answers to test items

Variables

- **answer** – The answer to item provided by the user
- **reference** – The specific item this response refers to
- **scores** – The list of correct response elements in the answer

2.19 user module

class user.**User**(**args*, ***values*)
Bases: mongoengine.document.Document

A class representing a user

Variables

- **name** – The username
- **type** – string
- **condition** – The condition of the user (“FLASHMAP” or “FLASHCARD”)
- **birthdate** – The birthdate of the user
- **read_sources** – A list of read sources by the user
- **gender** – The gender of the user (can be either ‘male’, ‘female’, or ‘other’)
- **code** – The code from the user’s informed consent form
- **tests** – The pre- and posttest
- ***questionnaire*** – The questionnaire
- **instances** – A list of instances storing the flashmap/flashcard data for the user
- **sessions** – A list of past sessions for this user
- **email** – The email address for this user
- **source_requests** – The days that the user was prompted a source request
- **successful_days** – The days that the user successfuly completed a session

- **debriefed** – Whether the user already got the briefing after the experiment

add_new_instance (*references*)

Adds a new `Instance` to this user

Parameters **reference** (*list (Flashcard or Edge)*) – A set of flashcards or edges for which to add a new instance

Returns The reference for which a new instance was added

Return type *Flashcard* or *Edge*

add_source (*source*)

Adds a read source to self

Parameters **source** (*string*) – The source to be added

check_due (*item*)

Checks whether the provided item is due for review

Parameters **item** (*Edge or Flashcard*) – The item to which the checked instance refers to

Returns Whether the provided item is due for review

Return type bool

create_questionnaire (*pu_items*, *peou_items*)

A method for creating a new questionnaire

Parameters

- **pu_items** – A list of perceived usefulness items
- **peou_items** – A list of perceived ease of use items

Returns A randomised list of questionnaire items

Return type list(*QuestionnaireItem*)

create_test (*flashcards*, *items*)

A method for creating a new test with unique questions

Parameters

- **flashcards** (*list (Flashcard)*) – A list of flashcards from the database
- **items** (*list (TestItem)*) – A list of items from the database

Returns A dict containing a list of FlashcardResponses and TestItemResponses

Return type dict(string, *Flashcard* or *TestItem*)

get_due_instance ()

Returns the instance with the oldest due date

Returns Either the instance with the lowest due date or a None object

Return type *Instance*

get_instance_by_id (*instance_id*)

Retrieves an instance based on a provided instance id

Parameters **instance_id** (*ObjectId*) – The id of the instance to be requested

Returns The instance or None if no instance with instance_id exists

Return type *Instance*

retrieve_recent_instance()
Retrieves the instance most recently answered by the user

Returns The instance with the latest response.end being the most recent of all instances

Return type *instance*

set_descriptives(*birthdate*, *gender*, *code*)
A method for setting the descriptives of the user

Parameters

- **birthdate** (`DateTime`) – The provided birthdate of the user
- **gender** (`string`) – The gender of the user (can be either 'male', 'female', or 'other')
- **code** (`string`) – The code from the informed consent form

time_spend_today()
A method for calculating the amount of seconds the user has spend on practicing flashcards

Returns The amount of seconds between every start and end of all responses of all instances of today

Return type int

undo()
Removes the response last submitted by the user, reschedules the respective instance, and returns the referred flashcard or edge

Returns The flashcard or edge referred to by the instance with the latest response

Return type *Flashcard* or *Edge*

validate(*instance_id*, *correct*)
Finalises a `Response` within an existing `Instance`

Parameters

- **instance_id** – The id of the instance which the response refers to
- **correct** (`boolean`) – Whether the response provided by the user was correct or not

PYTHON MODULE INDEX

INDEX

S

T

U

V

Source files of the client

index.html

../software/client/index.html

```
<!DOCTYPE html>
<html>
    <head>
        <title> Flashmaps </title>
        <link rel="stylesheet" type="text/css" href="vis/dist/vis.css" />
        <link rel="stylesheet" type="text/css" href="style.css" />
        <script src="vis/dist/vis.js"></script>
        <script src="client.js"></script>
    </head>
    <body>
        <div id="wrapper">
            <header>
                <nav style="visibility:hidden">
                    <a href="#" onclick="help()">Help</a>
                    <a href="#" onclick="view_learned()">Show learned items</a>
                    <a href="#" onclick="learn()">Start learning</a>
                    <a href="#" onclick="logout()">Log out</a>
                    <a href="mailto:mvdenk@gmail.com">mvdenk@gmail.com</a>
                </nav>
                <div id="instructions"> <p>Helaas werkt de applicatie niet in deze browser, je hebt een modernere browser nodig.</p> </div>
            </header>
            <div id="mycontainer" style="visibility:hidden">
                <h3> Login </h3>
                Username:
                <input id="username" type="text" name="username"
                    onkeydown="if (event.keyCode == 13) authenticate()" />
                <a href="#" onClick="authenticate()">Log in</a>
            </div>
            <footer>
                <div id="panel"> </div>
            </footer>
        </div>
    </body>
</html>
```

style.css

../software/client/style.css

```
* {
    margin:0;
    padding:0;
    font-family: Sans-serif;
    color: #554600;
}

html, body {
    min-height: 100%;
    background: #FFF0AA;
}

#wrapper {
    padding-top: 120px;
    padding-bottom: 90px;
    position: absolute;
    top: 0;
    bottom: 0;
    left: 0;
    right:0;
}

#mycontainer {
    background: rgba(255, 255, 255, 0.3);
    text-align: center;
    min-height: 100%;
    margin: auto;
    margin-top: 10px;
    margin-bottom: 10px;
    width: 80%;
    padding: 10px;
    border-radius: 5px;
}

#mycontainer p {
    max-width: 30em;
    margin: auto;
    margin-top: 1em;
    text-align: left;
}

hr {
    max-width: 30em;
    margin: auto;
    margin-top: 1em;
}

header {
    text-align: center;
    margin-top: -100px;
    height: 100px;
}

a {
    display: inline-block;
    background: #D4C26A;
    margin: .6em;
    padding: .4em;
    text-decoration: none;
    color: #554600;
    border-radius: .2em;
```

```
}

a:hover {
    background: #AA9739;
}

h3 {
    margin: auto;
    margin-top: 1em;
    margin-bottom: .3em;
    width: 50%;
}

footer {
    text-align: center;
    margin-bottom: -50px;
    height: 50px;
}

table {
    text-align: left;
    margin: auto;
}

table tr td {
    margin: .3em;
}

input[type=text], textarea {
    border: 2px solid;
    color: #806D15;
    background: #D4C26A;
    outline: none;
    padding: .4em;
}

input[type=text]:focus, textarea:focus {
    background: #FFF0AA;
}

#instructions p {
    margin: 5px;
    width: 80%;
    margin: auto;
}

#instructions a {
    display: inline;
    background: 0;
    font-weight: bold;
    margin: 0;
    padding: 0;
}

#instructions a:hover {
    color: #805C15;
```

client.js

../software/client/client.js

```
var uname = "";
var cont = "mycontainer";
var ws = new WebSocket("ws://128.199.49.170:5678");
var network
var nodes
var edges
var map
var options = {
    nodes: {
        shape: 'box',
        color: {
            border: "#554600",
            background: "#D4C26A",
            highlight: {
                border: "#554600",
                background: "#AA9739"
            }
        },
        font: {
            color: "#554600"
        }
    },
    edges: {
        arrows: {
            to: {enabled: true}
        },
        color: {
            color: "#554600",
            highlight: "#554600"
        },
        font: {
            color: "#554600"
        }
    },
    interaction: {
        selectable: true,
        dragNodes: true
    }
//    physics : {barnesHut: {avoidOverlap: 1}}
};
var show_undo = false;
var question = "";
var answer = "";
var fc_id = "";
var logged_in = false;

ws.onopen = function (event) {
    document.getElementById("instructions").innerHTML = "<p>Je kunt hier inloggen door een al bestaande gebruikersnaam in te vullen, of een nieuw account aanmaken door een zelbedachte, nieuwe gebruikersnaam in te vullen. Als dit niet lukt, stuur dan een email naar <a href='mailto:mvdenk@gmail.com'>mvdenk@gmail.com</a>.</p>";
    document.getElementById(cont).style.visibility = "visible";
    document.getElementById("username").focus();
}

ws.onclose = function(event) {
    if (logged_in) {
        if (!alert("Connection lost.")) window.location.reload();
    }
```

```
}

function authenticate() {
    uname = document.getElementById("username").value;
    console.log(uname);
    var msg = {
        keyword: "AUTHENTICATE-REQUEST",
        data: {
            name: uname,
            browser: navigator.platform
        }
    }
    ws.send(JSON.stringify(msg));
}

ws.onmessage = function (event) {
    var msg = JSON.parse(event.data);
    console.log(msg);
    if (uname != "questionnaire") logged_in = true;

    switch(msg.keyword) {
        case "ACTIVE_SESSIONS":
            logged_in = false;
            break;
        case "MAP-RESPONSE":
            show_map(msg.data);
            break;
        case "AUTHENTICATE-RESPONSE":
            show_menu();
            help();
            break;
        case "LEARNED_ITEMS-RESPONSE":
            if (msg.condition == "FLASHMAP") {
                coloured_map = colourise_progress(msg.data);
                show_map(coloured_map);
            }
            else if (msg.condition == "FLASHCARD") {
                show_flashcard_progress(msg.data);
            }
            break;
        case "LEARN-RESPONSE":
            if (msg.condition == "FLASHMAP") {
                show_map(flashmap(msg.data, msg.time_up, msg.successful_days));
                break;
            }
            else if (msg.condition == "FLASHCARD") {
                show_card(msg.data, msg.time_up, msg.successful_days);
                break;
            }
        case "NO_MORE_INSTANCES":
            done_learning(msg.data, msg.successful_days);
            break;
        case "READ_SOURCE-REQUEST":
            prompt_source_request(msg.data);
            break;
        case "DESCRIPTIVES-REQUEST":
            ask_descriptives();
            break;
        case "TEST-REQUEST":
            test(msg.data);
            break;
        case "QUESTIONNAIRE-REQUEST":
```

```
                questionnaire(msg.data);
                break;
            case "DEBRIEFING-REQUEST":
                debriefing();
                break;
        }
}

function ask_descriptives() {
    document.getElementById("panel").innerHTML = "";
    document.getElementById("instructions").innerHTML = "Voer hier je algemene gegevens in. Ben je je code kwijt? Stuur dan even een email met je gebruikersnaam en echte naam naar mvdenk@gmail.com";
    document.getElementById(cont).innerHTML = " \
        <h3> Wat is je geslacht? </h3> \
        <table> \
        <tr> \
        <td> <input type='radio' name='gender' value='male' checked/> </td><td> Mannelijk </td> \
        </tr><tr> \
        <td> <input type='radio' name='gender' value='female' /> </td><td> Vrouwelijk </td> \
        </tr><tr> \
        <td> <input type='radio' name='gender' value='other' /> </td><td> Anders </td> \
        </tr> \
        </table> \
        <h3> Wat is je geboortedatum? </h3> \
        <input type='text' name='birthdate' id='birthdate' /> <br /> (dd-mm-yyyy) \
        <h3> Wat is de code vermeld op de toezeggingsverklaring? </h3> \
        <input type='text' name='code' id='code' /> <br /> \
        <a href='#' onClick='send_descriptives()'>Verstuur</a> \
        <div id='invalid' />";
}

function send_descriptives() {
    msg = {keyword: "DESCRIPTIVES-RESPONSE", data: {gender: "male", birthdate: 0}};
    var genderbuttons = document.getElementsByName('gender');
    for (i = 0; i < genderbuttons.length; i++) {
        if (genderbuttons[i].checked) msg.data.gender = genderbuttons[i].value;
    }
    msg.data.code = document.getElementById('code').value;
    datestr = document.getElementById('birthdate').value;
    var parts = datestr.split("-");
    if (parts.length != 3
            || parseInt(parts[2], 10) < 1900 || parseInt(parts[2], 10) > new Date().getFullYear()
            || parseInt(parts[1], 10) < 1    || parseInt(parts[1], 10) > 12
            || parseInt(parts[0], 10) < 1    || parseInt(parts[0], 10) > 31) {
        document.getElementById('invalid').innerHTML = "INVALID DATE"
        return
    }
    msg.data.birthdate = new Date(parseInt(parts[2], 10), parseInt(parts[1], 10) - 1, parseInt(parts[0], 10));
    if (msg.data.birthdate > new Date()) document.getElementById(cont).innerHTML += "INVALID DATE";
    else ws.send(JSON.stringify(msg));
}

function test(data) {
```

```
    document.getElementById("panel").innerHTML = "";
    document.getElementById("instructions").innerHTML = "<p> Probeer de onderstaande toets zo goed mogelijk in te vullen. Je mag vragen overslaan als je de antwoorden niet weet. Als dit de eerste toets is en je hebt de papieren versie al gemaakt kun je de toets overslaan door hem leeg te versturen. </p>"
    document.getElementById(cont).innerHTML = ""
    for (i = 0; i < data.flashcards.length; i++) {
        document.getElementById(cont).innerHTML += " \
            <h3>" + data.flashcards[i].question + "</h3> \
            <textarea rows='4' cols='50' class='test' name='flashcard' id='flashcard" + data.flashcards[i].id + "' />";
    }
    for (i = 0; i < data.items.length; i++) {
        document.getElementById(cont).innerHTML += " \
            <h3>" + data.items[i].question + "</h3> \
            <textarea rows='4' cols='50' class='test' name='item' id='item" + data.items[i].id + "' />";
    }
    document.getElementById(cont).innerHTML += "<br /><a href='#' onClick='send_test_results()'>Verstuur</a>";
}

function send_test_results() {
    msg = {keyword: "TEST-RESPONSE", data: {flashcard_responses : [], item_responses : []}}
    var flashcards = document.getElementsByName('flashcard');
    for (i = 0; i < flashcards.length; i++) {
        msg.data.flashcard_responses.push(
            {id : flashcards[i].id.slice(9), answer : flashcards[i].value});
    }
    var items = document.getElementsByName('item');
    for (i = 0; i < items.length; i++) {
        msg.data.item_responses.push(
            {id : items[i].id.slice(4), answer : items[i].value});
    }
    ws.send(JSON.stringify(msg));
}

function questionnaire(data) {
    document.getElementById("panel").innerHTML = "";
    document.getElementById("instructions").innerHTML = "<p>Hieronder staan stellingen waarbij je aan kunt geven of je het er mee eens of oneens bent. Dit is voor mij om te kunnen bepalen of je het flashcard systeem nuttig vond en makkelijk te gebruiken.</p>";
    container = document.getElementById(cont);
    container.innerHTML = "";
    container_text = "";
    var form = "";
    for (i = 0; i < data.questionnaire.length; i++) {
        if (data.questionnaire[i].phrasing == "positive") form = "+";
        else form = "-";
        item_text = " \
            <h3>" + data.questionnaire[i].question + "</h3> \
            <table style='text-align:center;'> \
                <tr> \
                    <td>Zeer mee oneens</td><td>Mee oneens</td><td>Noch mee eens,<br />noch mee oneens</td><td>Mee eens</td><td>Zeer mee eens</td> \
                </tr><tr>";
        for (j=-2; j <= 2; j++) {
```

```
            item_text += "<td><input type='radio' class='item' name='"+form+data.questionnaire[i].id+"' value='"+j+"' /></td>";
        }
        item_text += "</tr></table>";
        container_text += item_text;
    }
    container_text += " \
        <h3>Wat vond je goed aan het flashcard systeem?</h3> \
        <textarea rows='4' cols='50' class='questionnaire' name='goed' id='goed'></textarea> \
        <h3>Wat zijn eventuele verbeteringen die gemaakt zouden kunnen worden?</h3> \
        <textarea rows='4' cols='50' class='questionnaire' name='kan_beter' id='kan_beter'></textarea>";
    container_text += "<br /><p>Als je bereid bent om later ge&iuml;nterviewd te worden over het flashcard systeem, vul dan hieronder je emailadres in.</p> \
                    <input type='text' id='email'/>";
    container_text += "<br /><a href='#' onClick='send_questionnaire_results()'>Verstuur</a>";
    container.innerHTML = container_text;
}

function send_questionnaire_results() {
    msg = {keyword: "QUESTIONNAIRE-RESPONSE", data: {responses : [], goed: "", kan_beter: "", email: ""}}
    var q_item = document.getElementsByClassName('item');
    for (i = 0; i < q_item.length && q_item[i].checked; i++) {
        var phrasing = 'negative'
        if (useful[i].name.charAt(0) == '+') var phrasing = 'positive'
        msg.data.responses.push({id: useful[i].name.slice(1), answer: useful[i].value, phrasing: phrasing});
    }
    msg.data.good = document.getElementById("goed").value;
    msg.data.can_be_improved = document.getElementById("kan_beter").value;
    msg.data.email = document.getElementById("email").value;
    console.log(JSON.stringify(msg));
    ws.send(JSON.stringify(msg));
}

function show_map(map) {
    // provide the data in the vis format

    nodes = new vis.DataSet(map.nodes);
    edges = new vis.DataSet(map.edges);

    var graph = {
        nodes: nodes,
        edges: edges
    };

    var container = document.getElementById(cont);
    container.innerHTML = "";
    container.style= "height:100%";

    // initialize your network!
    network = new vis.Network(container, graph, options);

    network.on('click', function(properties) {
        for (i=0; i < map.edges.length; i++) {
            if ('correct' in map.edges[i] && properties.nodes[0] == map.edges[i].to) {
```

```
                map.edges[i].correct = !map.edges[i].correct;
                if (map.edges[i].correct) {
                    edges.update([{ id: map.edges[i].id, color: {color: "#0F640F", highlight: "#0F640F"}, font: {color: "#0F640F"}}])
                    nodes.update([{ id : map.edges[i].to, color: {border: "#0F640F", background: "#55AA55", highlight: {border: "#0F640F", background: "#55AA55"}}, font: {color: "#0F640F"}}]);
                }
                else {
                    edges.update([{ id: map.edges[i].id, color: {color: "#550000", highlight: "#550000"}, font: {color: "#550000"}}])
                    nodes.update([{ id : map.edges[i].to, color: {border: "#550000", background: "#AA3939", highlight: {border: "#550000", background: "#AA3939"}}, font: {color: "#550000"}}]);
                }
            }
        }
    });
}

function show_menu() {
    document.getElementById("instructions").innerHTML = "";
    document.getElementById(cont).innerHTML = "";
    document.getElementsByTagName("nav")[0].style.visibility = "visible";
}

function colourise_progress(data) {
    document.getElementById("panel").innerHTML = "";
    return data
}

function show_flashcard_progress(data) {
    document.getElementById("panel").innerHTML = "";
    document.getElementById(cont).innerHTML = " \
        <p> Klaar om nu geleerd te worden: " + data.due +" </p> \
        <p> Nog niet gezien: " + data.not_seen + " </p> \
        <p> Nieuw: " + data.new + " </p> \
        <p> Lerende: " + data.learning + " </p> \
        <p> Geleerd: " + data.learned + " </p>"
}

function show_card(data, time_up, successful_days) {
    document.getElementById("instructions").innerHTML = "<p> Probeer de onderstaande vraag te beantwoorden </p>";
    if (time_up) {
        if (successful_days < 6) {
            document.getElementById("instructions").innerHTML = "<p style='color: red;'> Je hebt vandaag 15 minuten geleerd, nog "+ (6 - successful_days).toString() +" dagen te gaan. </p>";
        }
        if (successful_days == 6) {
            document.getElementById("instructions").innerHTML = "<p style='color: red;'> Je hebt vandaag 15 minuten geleerd, kom morgen terug voor de laatste kennistoets en de enquete. </p>";
        }
    }
    question = data.question;
    answer = data.answer;
    fc_id = data.id;
    document.getElementById(cont).innerHTML = question;
```

```
    if (show_undo) document.getElementById("panel").innerHTML = "<a href='#' onclick='undo()'> Undo </a> <a href='#' onclick='show_answer_fc()'> Toon antwoord </a>";
    else document.getElementById("panel").innerHTML = "<a href='#' onclick='show_answer_fc()'> Toon antwoord </a>";
}

function show_answer_fc() {
    document.getElementById("instructions").innerHTML = "<p> Geef aan of het door jou bedachte antwoord correct of incorrect was </p>";
    document.getElementById(cont).innerHTML += "<br><br>" + answer;
    document.getElementById("panel").innerHTML = "<a href='#' onclick='validate_fc(false)'> Incorrect </a><a href='#' onclick='validate_fc(true)'> Correct </a";
}

function flashmap(data, time_up, successful_days) {
    document.getElementById("instructions").innerHTML = "<p> Probeer te bedenken wat er in de oranje lege velden moet komen te staan. </p>";
    if (time_up) {
        if (successful_days < 6) {
            document.getElementById("instructions").innerHTML = "<p style='color:red;'> Je hebt vandaag 15 minuten geleerd, nog "+ (6 - successful_days).toString() +" dagen te gaan. </p>";
        }
        if (successful_days == 6) {
            document.getElementById("instructions").innerHTML = "<p style='color:red;'> Je hebt vandaag 15 minuten geleerd, kom morgen terug voor de laatste kennistoets en de enquete. </p>";
        }
    }
    question = data.question;
    map = data;
    for (i = 0; i < map.edges.length; i++) {
        if (map.edges[i].learning) {
            map.edges[i].color = "orange";
            for (j = 0; j < map.nodes.length; j++) {
                if (map.edges[i].to == map.nodes[j].id) {
                    map.nodes[j].color = {background : "orange"};
                    map.nodes[j].true_label = map.nodes[j].label;
                    map.nodes[j].label = "______";
                }
            }
        }
    }
    if (show_undo) document.getElementById("panel").innerHTML = "<a href='#' onclick='undo()'> Undo </a> <a href='#' onclick='show_answer_fm()'> Toon antwoord </a>";
    else document.getElementById("panel").innerHTML = "<a href='#' onclick='show_answer_fm()'> Toon antwoord </a>";
    return map;
}

function show_answer_fm() {
    document.getElementById("instructions").innerHTML = "<p> Geef aan of jouw antwoord goed of fout was. Als je op de velden klikt, veranderen ze van kleur, waarbij groen een goed antwoord is en rood een fout antwoord. </p>";
    var index
    for (i = 0; i < map.edges.length; i++) {
        if (map.edges[i].learning) {
            for (j=0;j < map.nodes.length; j++) {
```

```
                if (map.edges[i].to == map.nodes[j].id) index = j;
            }
            edges.update([{ id: map.edges[i].id, color: {color: "#0F640F", highlight: {color: "#0F640F"}}, font: {color: "#0F640F"}}])
            nodes.update([{id: map.edges[i].to, color: {border: "#0F640F", background : "#55AA55"}}]);
            nodes.update([{id: map.edges[i].to, label: map.nodes[index].true_label }]);
            map.edges[i].correct = true;
        }
    }
    document.getElementById("panel").innerHTML = "<a href='#' onclick='validate_fm()'> Volgende </a>";
}

function view_learned() {
    var msg = {keyword: "LEARNED_ITEMS-REQUEST", data: {}};
    ws.send(JSON.stringify(msg));
}

function undo() {
    var msg = {keyword: "UNDO", data: {}};
    ws.send(JSON.stringify(msg));
    show_undo = false;
}

function validate_fc(correct) {
    var msg = {keyword: "VALIDATE", data: {responses: [{}]}};
    msg.data.responses[0].id = fc_id;
    msg.data.responses[0].correct = correct;
    ws.send(JSON.stringify(msg));
    show_undo = true;
}

function validate_fm() {
    var msg = {keyword: "VALIDATE", data: {}};
    var responses = [];
    for (i = 0; i < map.edges.length; i++) {
        if (map.edges[i].learning) responses.push({id: map.edges[i].id, correct: map.edges[i].correct});
    }
    msg.data.responses = responses;
    ws.send(JSON.stringify(msg));
    show_undo = true;
}

function learn() {
    var msg = {
        keyword: "LEARN-REQUEST",
        data: {}
    }
    ws.send(JSON.stringify(msg));
}

function done_learning(data, successful_days) {
    if (data.source != "") {
        if (successful_days < 6) document.getElementById(cont).innerHTML = "<p>Er zijn geen flashcards meer behorende tot de paragrafen die je gelezen hebt. Je bent daarmee klaar voor vandaag, en je hebt nog "+ (6 - successful_days).toString() +" dagen te gaan. Als je paragraaf " + data.source + " gelezen kun je verder met de volgende flashcards.</p><a href='#' onclick='confirm_source(\"" + data.source + "\")'> Verder
```

```
            </a>";
        else if (successful_days == 6) document.getElementById(cont).innerHTML = "<p>Er zijn geen flashcards meer behorende tot de paragrafen die je gelezen hebt. Je bent daarmee klaar voor vandaag, kom morgen terug voor de laatste kennistoets en de enquete. Als je paragraaf " + data.source + " gelezen kun je verder met de volgende flashcards.</p><a href='#' onclick='confirm_source(\"" + data.source + "\")'> Verder </a>";
        else document.getElementById(cont).innerHTTML = document.getElementById(cont).innerHTML = "<p>Er zijn geen flashcards meer behorende tot de paragrafen die je gelezen hebt. Je bent daarmee klaar voor vandaag. Als je paragraaf " + data.source + " gelezen kun je verder met de volgende flashcards.</p><a href='#' onclick='confirm_source(\"" + data.source + "\")'> Verder </a>";
    }
    else {
        if (successful_days < 6) document.getElementById(cont).innerHTML ="<p>Er zijn geen flashcards meer voor nu, en daarmee ben je klaar voor vandaag. Je hebt nog "+ (6 - successful_days).toString() +" dagen te gaan.</p>";
        else if (successful_days == 6) document.getElementById(cont).innerHTML ="<p>Er zijn geen flashcards meer voor nu, en daarmee ben je klaar voor vandaag. Kom morgen terug voor de laatste kennistoets en de enquete.</p>";
        else document.getElementById(cont).innerHTML ="<p>Er zijn geen flashcards meer voor nu, en daarmee ben je klaar voor vandaag.</p>";
    }
    document.getElementById("panel").innerHTML = "";
}

function prompt_source_request(data) {
    document.getElementById(cont).innerHTML = "Heb je paragraaf " + data.source + " al gelezen? Zo nee, lees deze dan nu.";
    document.getElementById("panel").innerHTML = "<a href='#' onclick='confirm_source(\"" + data.source + "\")'> Gelezen </a>";
}

function confirm_source(source_) {
    var msg = {keyword: "READ_SOURCE_RESPONSE", data: { source : source_ }};
    ws.send(JSON.stringify(msg));
}

function help() {
    document.getElementById("instructions").innerHTML = "";
    document.getElementById("panel").innerHTML = "";
    document.getElementById(cont).innerHTML = "<p>Dankjewel voor het meedoen aan het experiment. Hier kun je iedere dag met de flashcards oefenen om je zo goed voor te kunnen bereiden op de toets over Nederlandse literatuur uit de 17de eeuw.</p><p>Het flashcard systeem is het meest effectief als je iedere dag tijd eraan besteed. Bovendien krijg je de waardebon alleen als je iedere dag het systeem gebruikt voor 15 minuten, of totdat de flashcards voor die dag op zijn. Op het moment dat je op een bepaalde dag klaar bent krijg je vanzelf een popup die aangeeft dat je klaar bent voor vandaag.</p>";
}

function debriefing() {
    document.getElementById("instructions").innerHTML = "";
    document.getElementById("panel").innerHTML = "";
    document.getElementById(cont).innerHTML = "<p>Hartelijk bedankt voor het meedoen aan het onderzoek, en gefeliciteerd met de waardebon. Je zult deze binnenkort van je leraar ontvangen als je de toezeggingsverklaring hebt
```

```
        ingeleverd. Verder staat het je vrij om gebruik te blijven maken van het
        flashcard systeem, het is goed om de kennis die je geleerd hebt vers te
        houden tot de toets. De resultaten van dit onderzoek kun je op verzoek ter
         inzage bij mij aanvragen. Als je net je email adres hebt ingevuld krijg
        je binnenkort een mail om een datum in te plannen voor het interview.
        Verder wens ik je nog veel succes voor dit vak. Als je nog vragen hebt kun
         je me altijd nog een email sturen (mvdenk@gmail.com).</p><a href='#'
        onclick='acc_debriefing()'>Gelezen</a>";
}

function acc_debriefing() {
    ws.send(JSON.stringify({keyword: "DEBRIEFING-RESPONSE", data: {}}));
}

function logout() {
    logged_in = false;
    ws.close();
    location.reload();
}
```

Screenshots of the client

Learning process

Figure 24: The user interface when prompting the user whether he has read paragraph 13.1

Figure 25: The user interface when prompting a flashcard

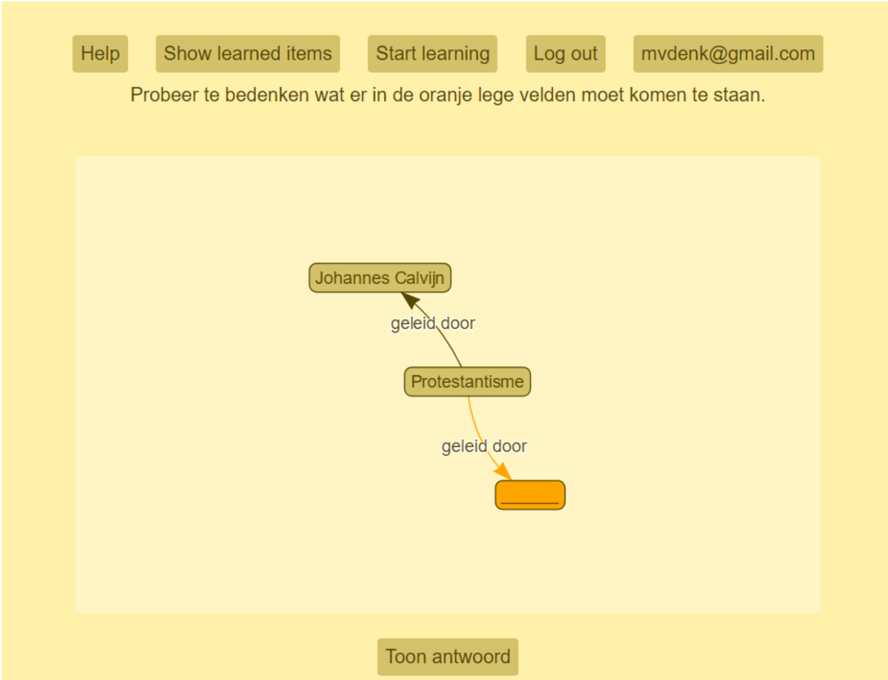


Figure 26: The user interface when prompting a flashmap

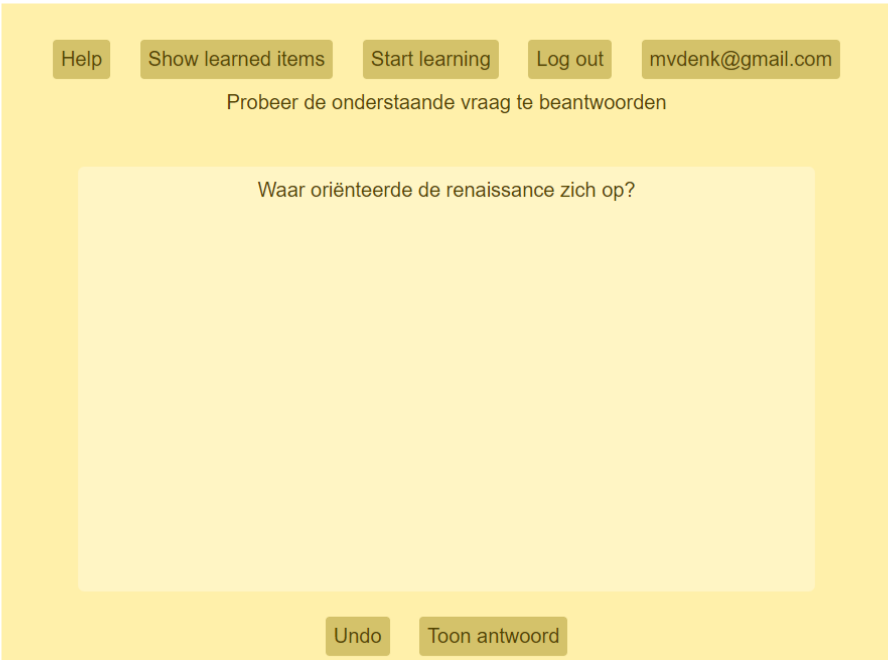


Figure 27: The user interface when prompting a flashcard with an undo option

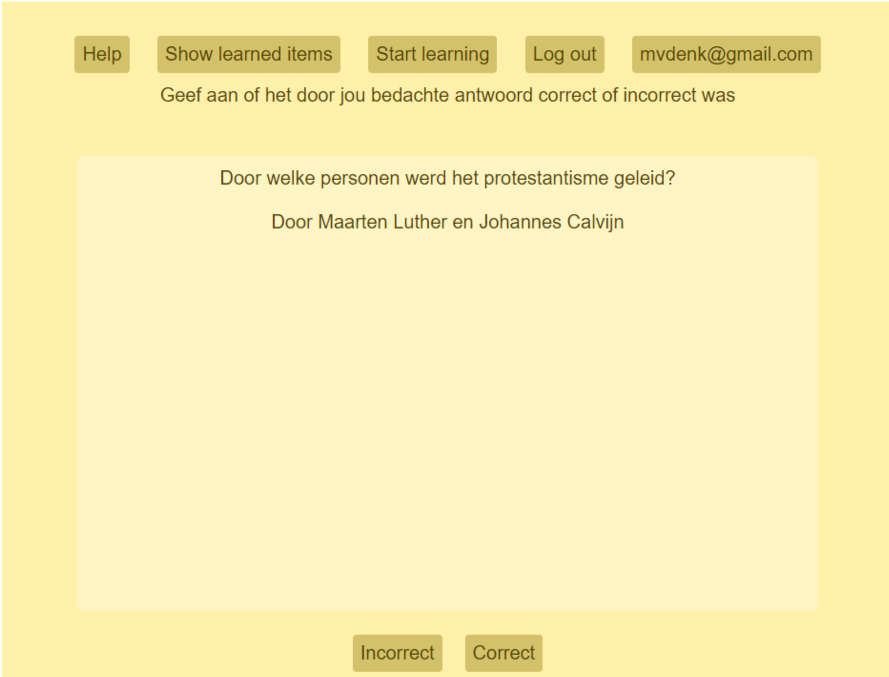


Figure 28: The user interface when showing the answer to a flashcard

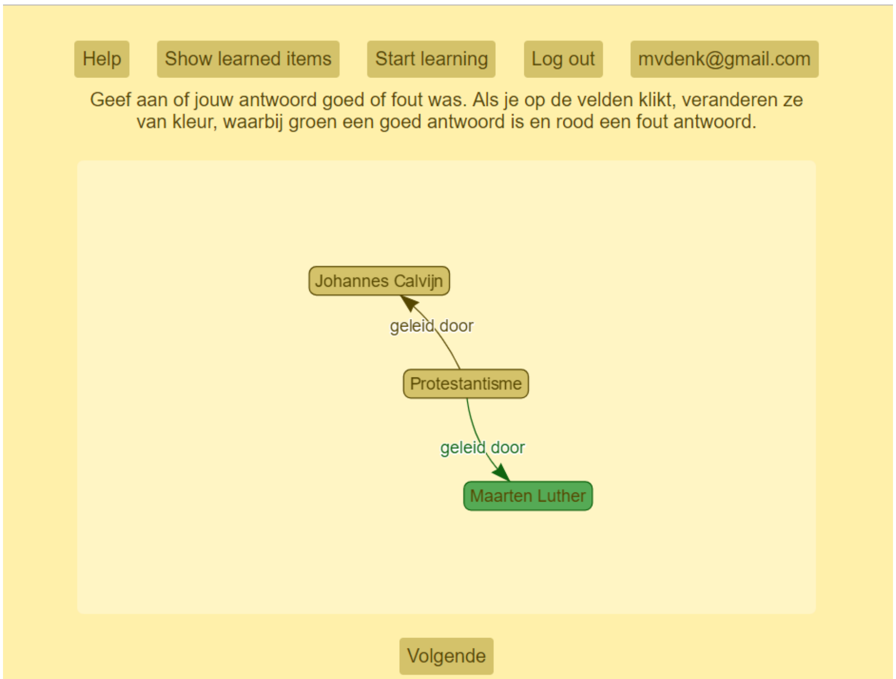


Figure 29: The user interface when showing the answer to a flashmap, here indicated as correct by the user

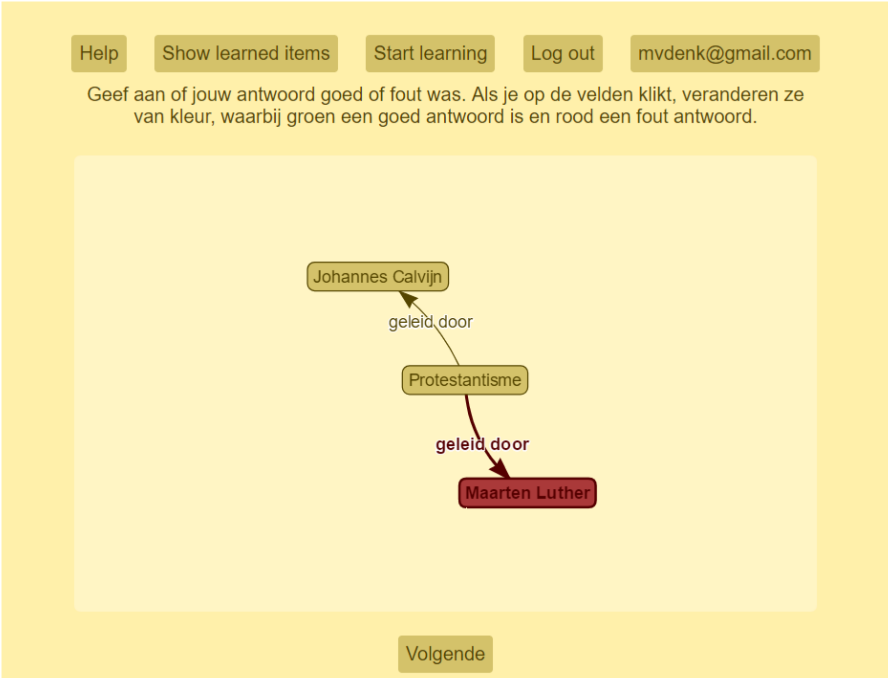


Figure 30: The user interface when showing the answer to a flashmap, here indicated as incorrect by the user

Figure 31: The user interface when showing that there are no new instances left to learn

Other views

Figure 32: The login screen

Figure 33: The descriptives screen

Figure 34: The top of the test screen

Figure 35: The bottom of the test screen

Figure 36: The top of the questionnaire screen

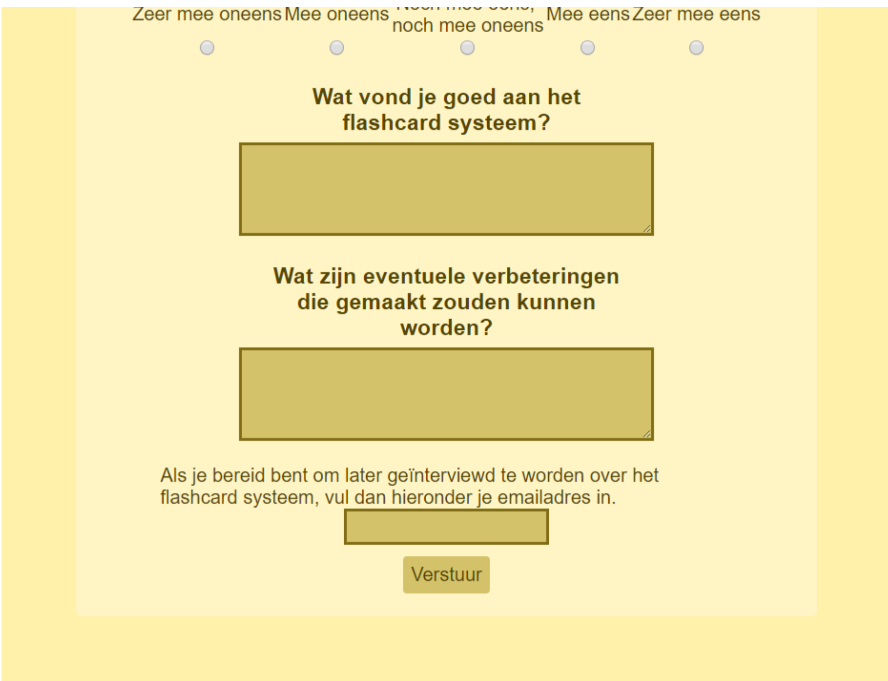


Figure 37: The bottom of the questionnaire screen

Figure 38: The debriefing screen

Figure 39: The help screen

Figure 40: The user interface when showing the learning progress to a flashcard user

Figure 41: The user interface when showing the learning progress to a flashcard user after reviewing some of the flashcards

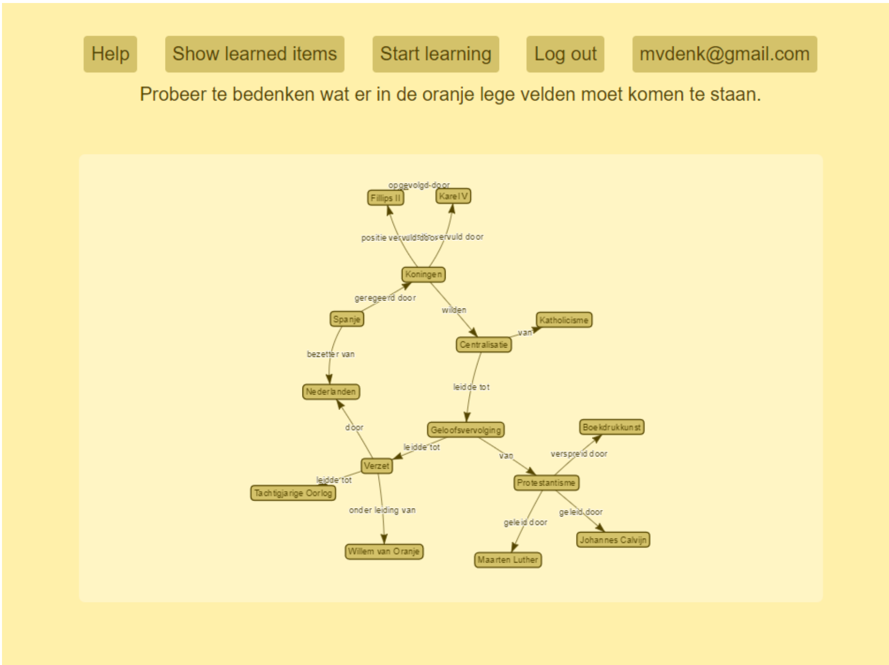


Figure 42: The user interface when showing the learning progress to a flashmap user

Informed consent form

Letter

Betreft: onderwijskundig onderzoek

Geachte ouder/verzorger,

Als masterstudent Educational Science and Technology aan de Universiteit Twente voer ik vanaf 17 tot en met 25 mei een onderzoek uit binnen de VWO 4 klassen van Het Stedelijk Lyceum. Dit onderzoek loopt binnen het vak Nederlands en is volledig in samenwerking met de docenten.

Aangezien het welzijn en vrijwillige deelneming van de deelnemer zeer belangijk is is het binnen de universiteit verplicht om voor een onderzoek toestemming te krijgen van de deelnemer, en als deze jonger is dan 18 jaar ook van een ouder of verzorger. Bijgaande deze brief vind u uitgebreide informatie over het onderzoek en een toestemmingsverklaring, dat door u samen met uw (pleeg-)kind ingevuld kan worden. Wanneer u of uw (pleeg-)kind geen toestemming geeft zal laatstgenoemde niet deelnemen aan het onderzoek zonder dat daar negatieve consequenties aan verbonden zijn. Bovendien is deze toestemming volledig vrijblijvend en kan de deelnemer zich op ieder moment terugtrekken.

Mogen er nog vragen zijn over het onderzoek kunt u mij altijd benaderen via [email address]. Bij voorbaat dank voor uw medewerking.

Met vriendelijke groeten,

Micha van den Enk

Informatie over het onderzoek

Beschrijving van het onderzoek

Veel leerlingen ervaren moeilijkheden bij het leren van teksten voor een toets, bijvoorbeeld omdat ze niet weten hoe ze het aan moeten pakken of dat ze niet vroeg genoeg beginnen. Dit is begrijpelijk, aangezien er weinig aandacht besteed wordt op school hoe dit aangepakt kan worden. Echter, er zijn een aantal hulpmiddelen beschikbaar die hier geschikt voor zijn, waaronder het flashcard systeem.

Dit systeem bestaat uit een aantal kaarten, de flashcards, waarop aan de ene kant een vraag en aan de andere kant een antwoord geformuleerd staat. Vervolgens besteedt de leerling iedere dag een bepaalde tijd aan het beantwoorden van deze vragen en dit vervolgens te controleren. Als het antwoord fout was wordt de kaart dezelfde dag nog herhaald, en als het goed was schuift de kaart door naar de volgende dag.

Voor dit onderzoek is er een nieuw digitaal systeem ontwikkeld gebaseerd op het flashcard systeem, dat een aantal extra mogelijkheden introduceert waardoor het gebruik van flashcards efficiënter en betekenisvoller wordt. Het doel van het onderzoek is het evalueren van dit nieuwe systeem in de context van het leren over 17de eeuwse Nederlandse literatuur. In het experiment worden twee verschillende versies getoetst, echter omwille het belang van het onderzoek zal hier verder niet over worden uitgewijd.

Voordelen voor de participant

Flashcards zijn al frequent getoetst in de wetenschap en blijken veel bij te dragen aan de leer-effectiviteit van leerlingen, zowel op de korte als de lange termijn. Bovendien kan een leerling op deze manier het werk verspreiden over de week en daarmee de werkdruk bij het naderen van het toetsmoment verlagen. Bovendien krijgt de leerling een waardebon van 5 euro voor Van der Poel IJs op het moment dat deze volledig aan het experiment meedoet (zie de voorwaarden in de procedure).

De gegevens die verzameld worden gedurende het experiment worden volledig anoniem opgeslagen, en er wordt geen cijfer aan het experiment gebonden.

De deelnemer kan er los voor kiezen om zich beschikbaar te stellen voor een interview, waarbij er gevraagd zal worden naar hoe deze het systeem gebruikt heeft. De informatie verkregen uit dit interview zal opnieuw geanonimiseerd worden en alleen via het onderzoeksrapport gecommuniceerd naar derden. Bovendien kan de leerling inzicht krijgen in zijn eigen resultaten op het moment dat deze beschikbaar zijn.

Zie ommezijde

Procedure

1. In de les van 17 mei zal er een korte introductie van het systeem plaatsvinden
2. De leerling kan vervolgens online inloggen in het systeem
3. Hier maakt deze een voorkennistoets
4. Vervolgens besteedt hij 7 dagen lang iedere dag 15 minuten aan het systeem
5. Aan het einde maakt de leerling nog een kennistoets om te kunnen bepalen wat er geleerd is in de tussenliggende periode
6. Ook vult deze kort een enquete om zijn ervaringen met het systeem te meten
7. Als de leerling minstens op 6 van de 7 dagen 15 minuten heeft besteed aan het systeem en de toetsen heeft ingevuld, krijgt deze de waardebon
8. Ten slotte vindt er eventueel een interview plaats

Na het einde van het onderzoek is er nog een week tijd voor de leerling om zichzelf voor te bereiden op de toets. Gedurende deze tijd kan de applicatie nog steeds gebruikt worden.

Als de student de waardebon wil ontvangen dient hij de code opgenomen in de toestemmingsverklaring in te vullen in de webapplicatie. Hierdoor kan de onderzoeker de identiteit van de deelnemer achteraf achterhalen, zonder dat identiteitsgegevens opgeslagen hoeven te worden in de gegevensbank. Als de student anoniem wenst te blijven kan hij dit veld in de webapplicatie leeglaten.

Toestemmingsverklaring

Ik verklaar op een voor mij duidelijke wijze te zijn ingelicht over de aard, methode, doel en [indien aanwezig] de risico's en belasting van het onderzoek. Ik weet dat de gegevens en resultaten van het onderzoek alleen anoniem en vertrouwelijk aan derden bekend gemaakt zullen worden. Mijn vragen zijn naar tevredenheid beantwoord.

Ik stem geheel vrijwillig in met deelname aan dit onderzoek. Ik behoud me daarbij het recht voor om op elk moment zonder opgaaf van redenen mijn deelname aan dit onderzoek te beëindigen.

Naam deelnemer: ....

Handtekening deelnemer: ....

Naam ouder of voogd: ....

Handtekening ouder of voogd: ....

Datum: ....

Code: XXXX

Participation statistics

Descriptives

Table 5: Including users with no active days

	N	min	max	mean	var	skew	kurt	normal-t	normal-p
Flashcards	33	0	7	3.27	7.58	0.02	-1.75	56.992	0.0000
Flashmap	30	0	11	3.27	10.41	0.39	-1.04	3.868	0.1446
General	63	0	11	3.27	8.78	0.24	-1.26	22.212	0.0000

Table 6: Excluding users with no active days

	N	min	max	mean	var	skew	kurt	normal-t	normal-p
Flashcards	25	1	7	4.32	5.39	-0.48	-1.47	11.919	0.0026
Flashmap	19	1	11	5.16	6.47	-0.12	0.11	0.624	0.7321
General	44	1	11	4.68	5.90	-0.24	-0.50	0.823	0.6628

Comparisons among conditions

Mann-Whitney-U k	**Mann-Whitney-U p**	**Welch's t-test k**	**Welch's t-test p**
0.008	0.9936	0.008	0.9937

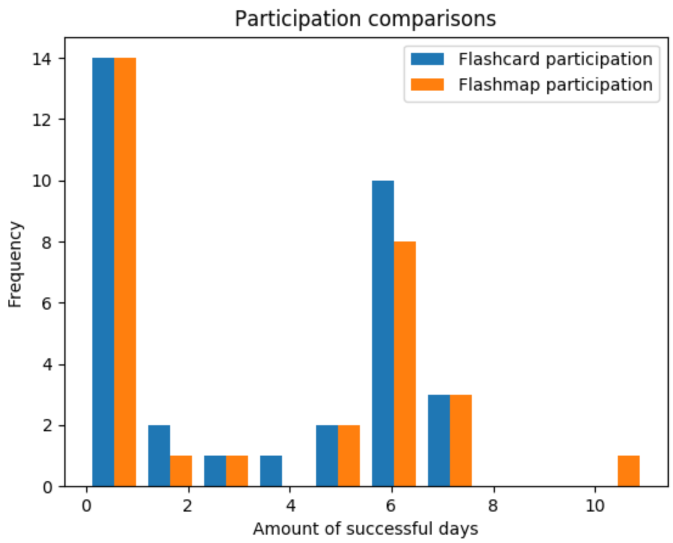


Figure 43: A histogram comparing the amount of active days between flashcard and flashmap users

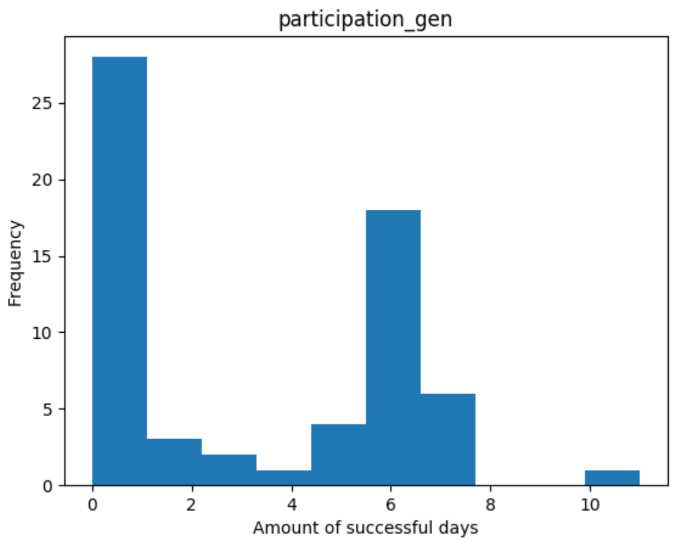


Figure 44: A histogram depicting the amount of active days of all participants

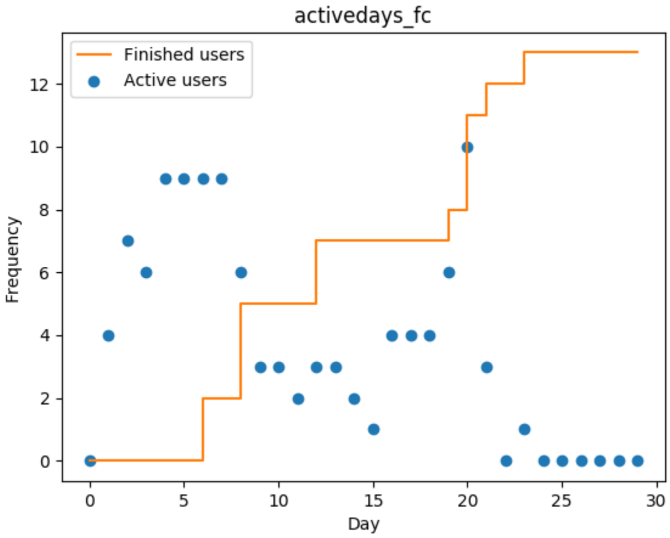


Figure 45: A scatter diagram depicting how many flashcard users were active during which day of the experiment, combined with a step diagram depicting how many flashcard users finished the experiment that day (more than or equal to 6 successful days)

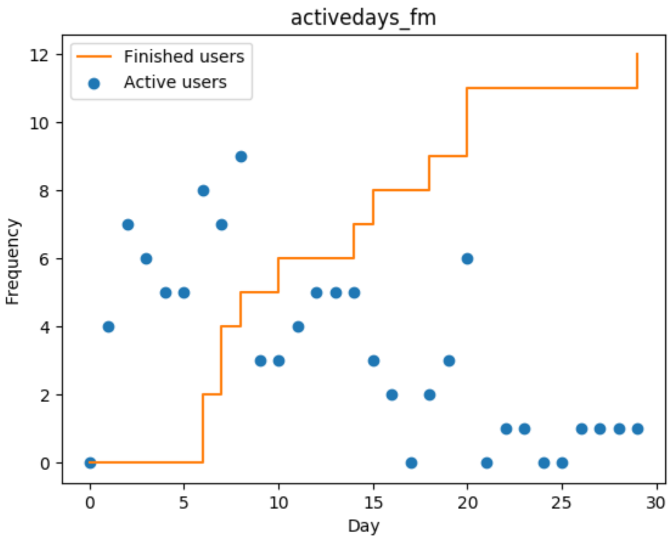


Figure 46: A scatter diagram depicting how many flashmap users were active during which day of the experiment, combined with a step diagram depicting how many flashmap users finished the experiment that day (more than or equal to 6 successful days)

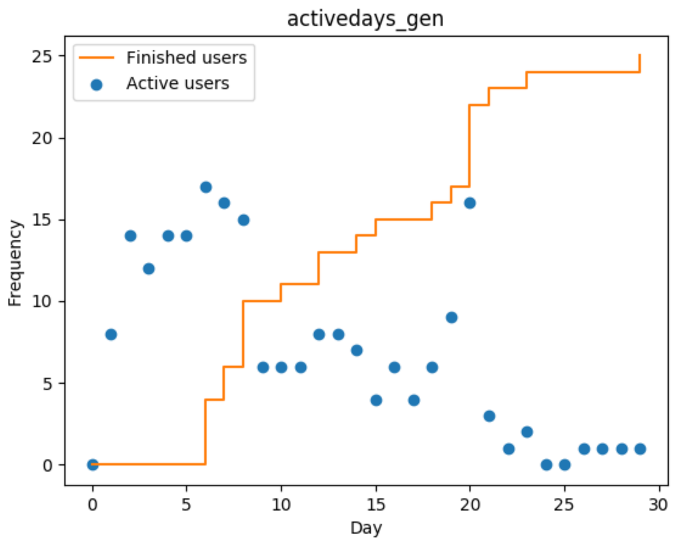


Figure 47: A scatter diagram depicting how many users in total were active during which day of the experiment, combined with a step diagram depicting how many users finished the experiment that day (more than or equal to 6 successful days)

Descriptive statistics

Gender

	Total	Male	Female	Other
Flashcards	12	7	5	0
Flashmap	11	8	3	0
Total	23	15	8	0

Age

	N	min	max	mean	var	skew	kurt	normal-t	normal-p
Flashcards	12	15	17	15.75	0.39	0.15	-0.52	0.096	0.9530
Flashmap	11	15	17	15.73	0.42	0.25	-0.62	0.211	0.8997
General	23	15	17	15.74	0.38	0.20	-0.57	0.307	0.8576

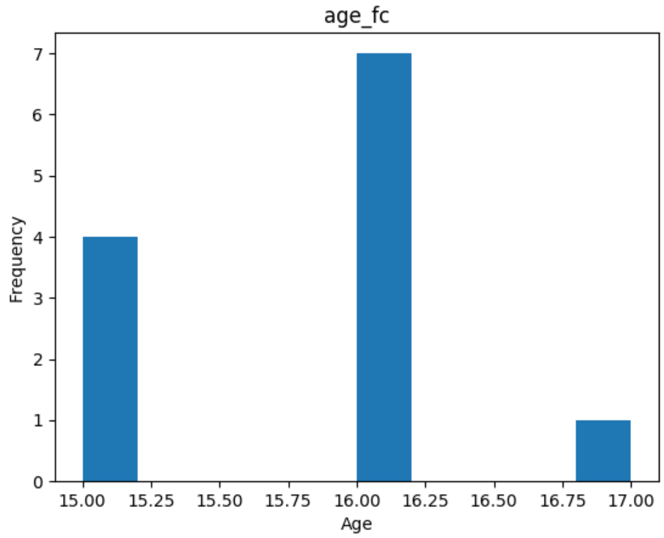


Figure 48: A histogram of the age of the flashcard users finishing the experiment

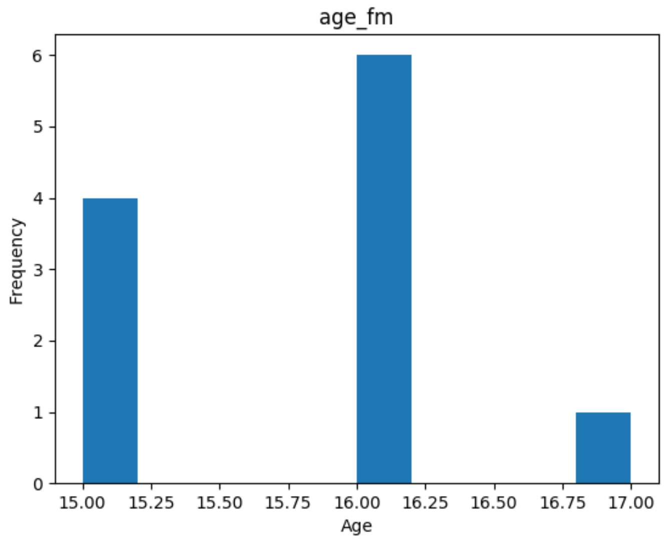


Figure 49: A histogram of the age of the flashmap users finishing the experiment

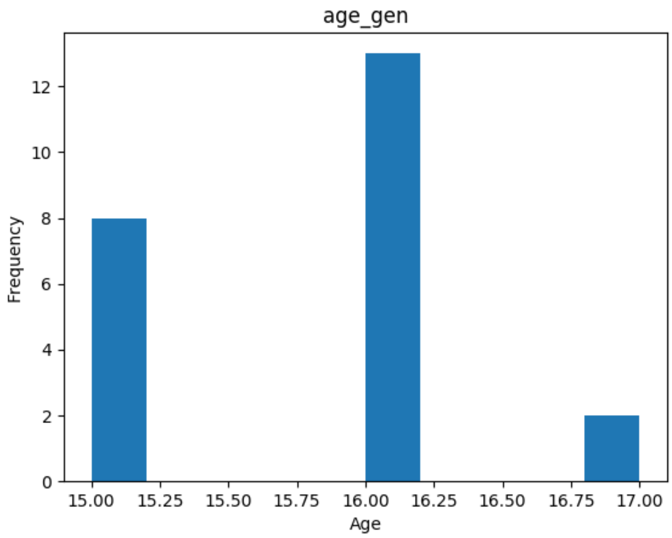


Figure 50: A histogram of the age of all users finishing the experiment

Comparisons of ages among conditions

Age

Mann-Whitney-U k	Mann-Whitney-U p	Welch's t-test k	Welch's t-test p
0.086	0.9323	0.086	0.9325

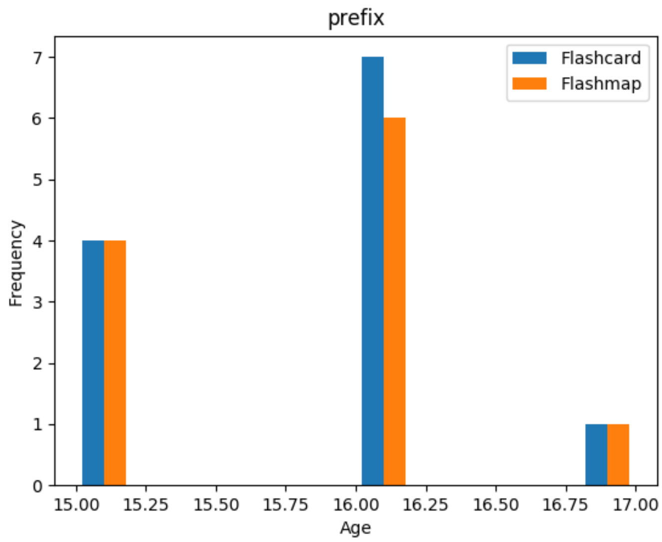


Figure 51: Age comparisons of flashcard and flashmap users

Pretest and posttest statistics

Inter-rater reliability

Amount of rated flashcards	10
Amount of rated items	12
Granted by both	10
Only granted by the teacher	1
Only granted by the researcher	3
Granted by none	58
Total amount of possible responses	72
Proportionate agreement	0.9444
Cohen's kappa	0.8003

Table 11: Statistics for calculating the inter-rater reliability

Descriptives of the knowledge questions

Table 12: Flashcard condition

	N	min	max	mean	var	skew	kurt	norm-t	norm-p	α
ctt:total	24	0	6	1.29	4.13	1.34	0.40	8.732	0.0127	0.6958
ctt:pretest	12	0	3	0.67	1.15	1.16	-0.19	4.546	0.1030	0.4290
ctt:posttest	12	0	6	1.92	6.63	0.70	-1.29	3.371	0.1854	0.7261
ctt:abs_learn_gain	12	-3	6	1.25	8.39	0.47	-0.84	0.900	0.6378	0.4290
ctt:rel_learn_gain	12	0	0	0.04	0.00	0.43	-0.86	0.810	0.6671	0.4290
irt:total	24	-1	4	-0.03	3.05	1.05	0.04	5.537	0.0627	0.4556
irt:pretest	12	0	0	0.00	0.08	0.62	0.07	2.059	0.3573	0.0687
irt:posttest	12	-2	3	-0.01	2.70	0.61	0.63	3.146	0.2074	0.3769
irt:abs_learn_gain	12	-2	3	-0.01	2.87	0.91	0.61	4.553	0.1026	0.0687
irt:rel_learn_gain	12	0	0	0.02	0.00	0.90	0.59	4.487	0.1061	0.0687
fixed irt:total	24	-4	3	-0.83	4.14	0.58	-0.36	1.812	0.4042	0.5294
fixed irt:pretest	12	1	3	2.60	0.17	0.43	0.39	2.010	0.3661	0.1088
fixed irt:posttest	12	-2	3	-0.07	2.71	0.61	0.63	3.188	0.2031	0.3774
fixed irt:abs_learn_gain	12	-4	1	-2.67	2.91	1.01	0.64	5.199	0.0743	0.1088
fixed irt:rel_learn_gain	12	0	0	-0.03	0.00	1.01	0.62	5.132	0.0769	0.1088

Table 13: Flashmap condition

	N	min	max	mean	var	skew	kurt	norm-t	norm-p	α
ctt:total	22	0	7	1.32	4.89	1.48	0.77	10.348	0.0057	0.7424
ctt:pretest	11	0	1	0.18	0.16	1.65	0.72	9.711	0.0078	-0.1132
ctt:posttest	11	0	7	2.45	7.27	0.45	-1.31	2.304	0.3160	0.6841
ctt:abs_learn_gain	11	-1	7	2.27	7.42	0.45	-1.14	1.448	0.4848	-0.1132
ctt:rel_learn_gain	11	0	0	0.05	0.00	0.44	-1.15	1.490	0.4747	-0.1132
irt:total	22	-2	4	-0.03	3.02	0.62	0.46	3.124	0.2097	0.3942
irt:pretest	11	0	0	-0.00	0.00	1.65	0.72	9.711	0.0078	0.0000
irt:posttest	11	-1	1	-0.00	0.76	-0.00	2.50	6.534	0.0381	0.1362
irt:abs_learn_gain	11	-1	1	-0.00	0.76	-0.00	2.50	6.534	0.0381	0.0000
irt:rel_learn_gain	11	0	0	0.02	0.00	0.00	2.50	7.592	0.0225	0.0000
fixed irt:total	22	-4	3	-0.80	3.94	0.17	0.04	0.556	0.7575	0.4530

fixed irt:pretest	11	0	0	0.11	0.00	0.00	-3.00	1.057	0.5894	0.0000
fixed irt:posttest	11	2	4	3.27	0.15	-0.02	2.44	6.403	0.0407	0.1020
fixed irt:abs_learn_gain	11	2	4	3.17	0.15	-0.02	2.44	6.403	0.0407	0.0000
fixed irt:rel_learn_gain	11	0	0	0.07	0.00	-0.02	2.44	6.403	0.0407	0.0000

Table 14: Combined conditions

	N	min	max	mean	var	skew	kurt	norm-t	norm-p	α
ctt:total	46	0	7	1.30	4.39	1.42	0.64	14.471	0.0007	0.7112
ctt:pretest	23	0	3	0.43	0.71	1.83	2.28	17.317	0.0002	0.3937
ctt:posttest	23	0	7	2.17	6.70	0.58	-1.30	6.839	0.0327	0.6851
ctt:abs_learn_gain	23	-3	7	1.74	7.84	0.40	-0.93	2.023	0.3637	0.3937
ctt:rel_learn_gain	23	0	0	0.03	0.00	0.39	-0.93	1.977	0.3722	0.3937
irt:total	46	-3	5	-0.10	4.92	0.98	-0.23	7.303	0.0259	0.5856
irt:pretest	23	-2	2	-0.00	1.22	0.97	2.19	9.614	0.0082	0.2141
irt:posttest	23	-3	3	-0.02	3.68	0.87	-0.19	3.757	0.1528	0.4740
irt:abs_learn_gain	23	-4	4	-0.02	5.31	0.42	-0.29	0.967	0.6166	0.2141
irt:rel_learn_gain	23	0	0	0.01	0.00	0.38	-0.27	0.833	0.6592	0.2141
fixed irt:total	46	-5	3	-1.85	5.68	0.80	-0.29	5.224	0.0734	0.6710
fixed irt:pretest	23	-2	2	-0.01	1.22	0.98	2.19	9.677	0.0079	0.2142
fixed irt:posttest	23	-4	3	-0.88	5.74	0.31	-0.53	0.564	0.7541	0.5859
fixed irt:abs_learn_gain	23	-7	4	-0.88	8.23	-0.21	0.11	0.742	0.6900	0.2142
fixed irt:rel_learn_gain	23	0	0	0.00	0.00	-0.26	0.19	1.015	0.6019	0.2142

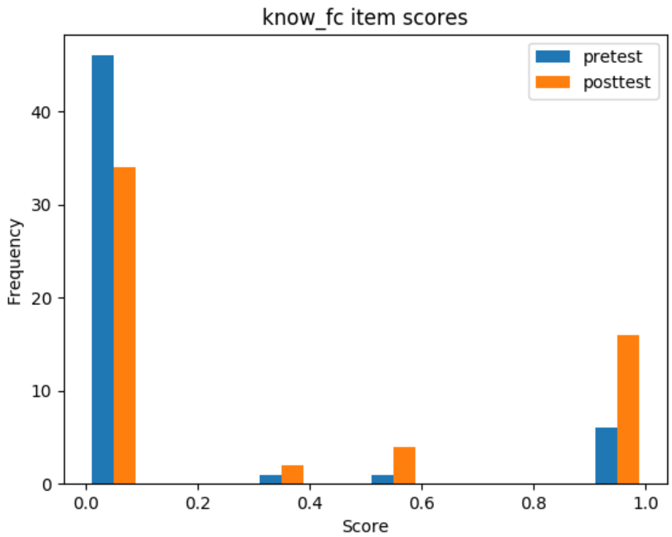


Figure 52: A histogram depicting the scores on the knowledge section of the pre- and posttest per item by flashcard users

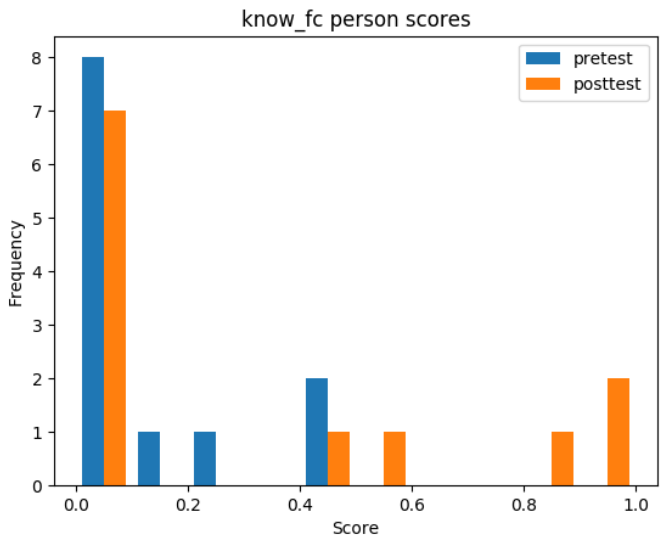


Figure 53: A histogram depicting the scores on the knowledge section of the pre- and posttest per flashcard user

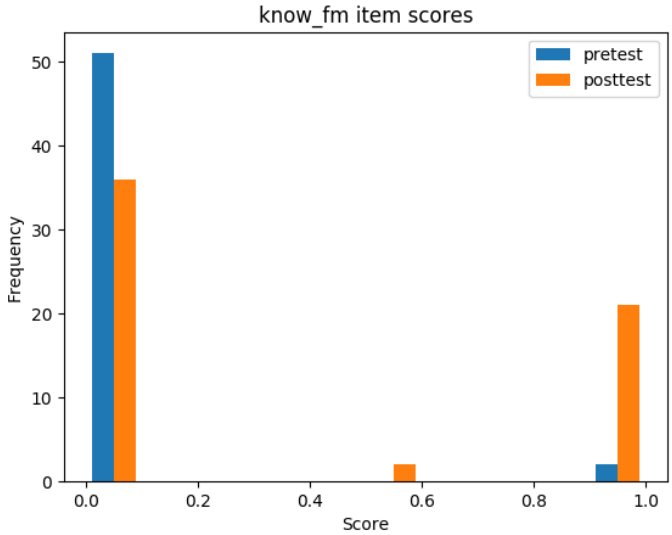


Figure 54: A histogram depicting the scores on the knowledge section of the pre- and posttest per item by flashmap users

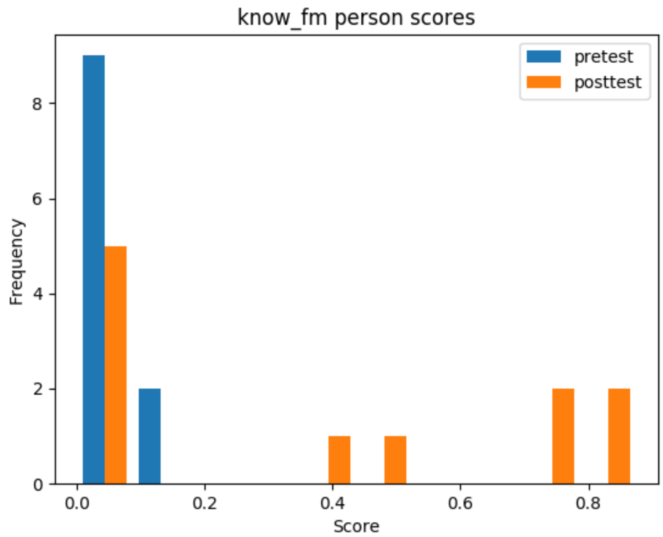


Figure 55: A histogram depicting the scores on the knowledge section of the pre- and posttest per flashmap user

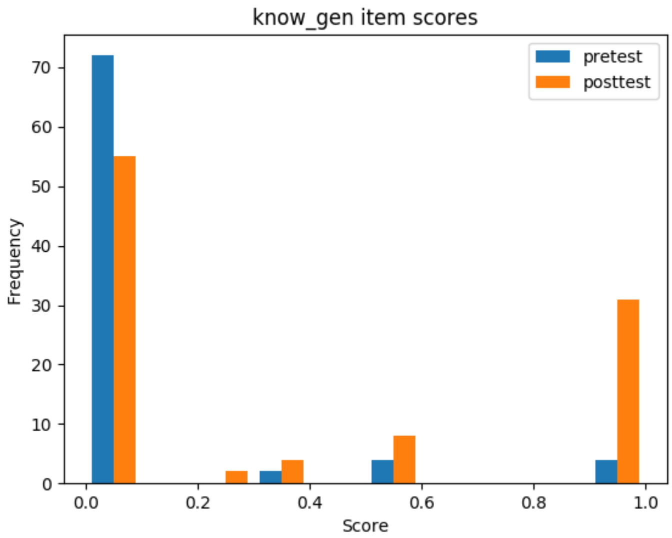


Figure 56: A histogram depicting the scores on the knowledge section of the pre- and posttest per item

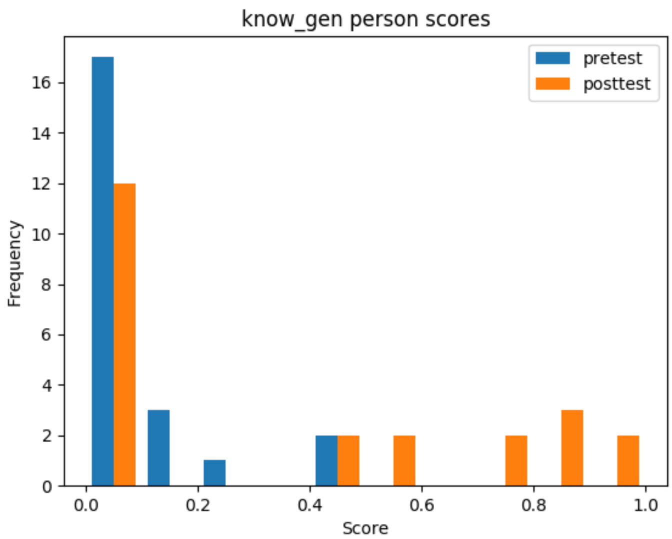


Figure 57: A histogram depicting the scores on the knowledge section of the pre- and posttest per user

Comparisons of the knowledge questions

Pre- and posttest comparisons

Table 15: Flashcard condition

	MW k	MW p	t-test k	t-test p
ctt	-1.552	0.1348	-1.552	0.1418
irt	0.016	0.9872	0.016	0.9873
fixed irt	5.454	0.0000	5.454	0.0001

Table 16: Flashmap condition

	MW k	MW p	t-test k	t-test p
ctt	-2.764	0.0120	-2.764	0.0192
irt	-0.000	1.0000	-0.000	1.0000
fixed irt	-27.206	0.0000	-27.206	0.0000

Table 17: Combined conditions

	MW k	MW p	t-test k	t-test p
ctt	-3.065	0.0037	-3.065	0.0049
irt	0.051	0.9597	0.051	0.9598
fixed irt	1.591	0.1187	1.591	0.1217

Learning gain comparisons between conditions

Table 18: Classical test theory

	MW k	MW p	t-test k	t-test p
total	-0.042	0.9664	-0.042	0.9665
pretest	1.407	0.1739	1.456	0.1669
posttest	-0.489	0.6297	-0.488	0.6305
abs_learn_gain	-0.870	0.3940	-0.873	0.3927

rel_learn_gain	-0.747	0.4635	-0.751	0.4611

Table 19: Item response theory

	MW k	**MW p**	**t-test k**	**t-test p**
total	-0.001	0.9989	-0.001	0.9989
pretest	0.000	1.0000	0.000	1.0000
posttest	-0.014	0.9889	-0.014	0.9887
abs_learn_gain	-0.014	0.9892	-0.014	0.9889
rel_learn_gain	0.072	0.9436	0.074	0.9423

Table 20: Item response theory with fixed item difficulties

	MW k	**MW p**	**t-test k**	**t-test p**
total	-0.050	0.9602	-0.050	0.9602
pretest	20.261	0.0000	21.204	0.0000
posttest	-6.549	0.0000	-6.821	0.0000
abs_learn_gain	-11.067	0.0000	-11.531	0.0000
rel_learn_gain	-10.401	0.0000	-10.845	0.0000

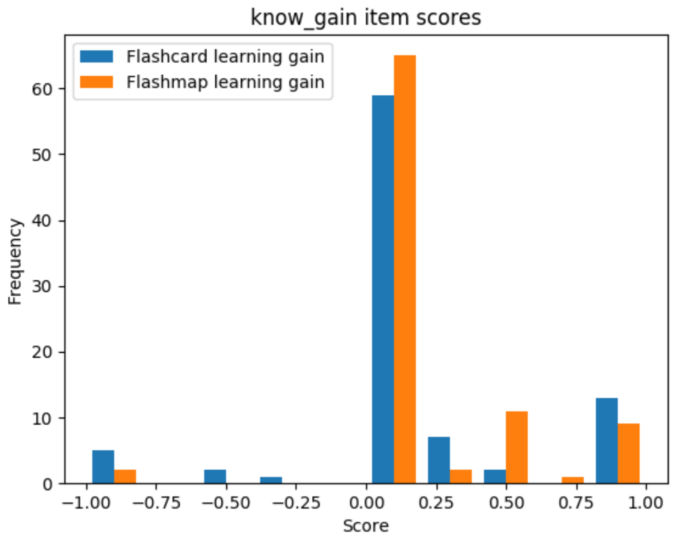


Figure 58: A comparison of figure 52 and 54, with the pretest score subtracted from the posttest score

Figure 59: A comparison of figure 53 and 55, with the pretest score subtracted from the posttest score

Descriptives of the comprehension questions

Table 21: Flashcard condition

	N	min	max	mean	var	skew	kurt	norm-t	norm-p	α
ctt:total	24	0	6	1.33	4.23	1.19	-0.05	6.646	0.0361	0.7215
ctt:pretest	12	0	4	0.33	1.33	3.02	7.09	33.648	0.0000	0.7670
ctt:posttest	12	0	6	2.33	5.33	0.41	-1.25	2.077	0.3540	0.6450
ctt:abs_learn_gain	12	0	6	2.00	5.45	0.72	-0.96	2.091	0.3516	0.6450
ctt:rel_learn_gain	12	0	0	0.07	0.00	0.72	-0.95	2.097	0.3504	0.6450
irt:total	24	-2	4	0.05	4.71	0.80	-0.83	4.030	0.1333	0.6583
irt:pretest	12	-1	4	-0.09	2.62	2.47	5.19	26.077	0.0000	0.3406
irt:posttest	12	-2	2	0.01	3.51	0.00	-1.31	1.869	0.3929	0.7510
irt:abs_learn_gain	12	-3	3	0.10	5.54	-0.07	-1.19	1.211	0.5459	0.3406
irt:rel_learn_gain	12	0	0	0.02	0.00	-0.15	-1.11	0.886	0.6420	0.3406
fixed irt:total	24	-3	3	-1.15	3.83	0.90	-0.52	4.058	0.1315	0.6673
fixed irt:pretest	12	0	5	0.42	2.58	2.72	6.04	29.551	0.0000	0.3207
fixed irt:posttest	12	-3	1	-0.87	3.16	0.03	-1.30	1.835	0.3994	0.7480
fixed irt:abs_learn_gain	12	-5	2	-1.28	5.01	-0.15	-1.04	0.652	0.7218	0.3207
fixed irt:rel_learn_gain	12	0	0	-0.01	0.00	-0.29	-0.81	0.411	0.8142	0.3207

Table 22: Flashmap condition

	N	min	max	mean	var	skew	kurt	norm-t	norm-p	α
ctt:total	22	0	8	1.27	3.92	2.06	4.13	22.828	0.0000	0.7202
ctt:pretest	11	0	4	0.82	1.56	1.64	1.82	12.332	0.0021	0.5351
ctt:posttest	11	0	8	1.73	6.22	1.58	1.62	11.397	0.0034	0.7566
ctt:abs_learn_gain	11	-1	4	0.91	2.89	1.04	-0.35	3.526	0.1715	0.5351
ctt:rel_learn_gain	11	0	0	0.04	0.00	1.09	-0.23	3.961	0.1380	0.5351
irt:total	22	-1	3	0.01	2.83	0.65	-1.02	3.743	0.1539	0.6260
irt:pretest	11	-1	3	-0.00	2.98	0.91	-0.34	2.786	0.2483	0.5317
irt:posttest	11	-2	2	0.01	3.41	0.19	-1.62	4.801	0.0907	0.6901
irt:abs_learn_gain	11	-1	2	0.01	1.71	1.00	-0.12	3.564	0.1683	0.5317
irt:rel_learn_gain	11	0	0	0.02	0.00	0.99	-0.11	3.521	0.1720	0.5317
fixed irt:total	22	-1	3	0.05	2.92	0.66	-1.01	3.727	0.1551	0.6277

fixed irt:pretest	11	0	4	1.54	2.10	0.93	-0.32	2.893	0.2354	0.4888
fixed irt:posttest	11	-1	3	0.57	3.81	0.15	-1.64	5.133	0.0768	0.6957
fixed irt:abs_learn_gain	11	-2	1	-0.97	1.88	1.08	0.01	4.284	0.1174	0.4888
fixed irt:rel_learn_gain	11	0	0	0.00	0.00	1.08	0.01	4.277	0.1178	0.4888

Table 23: Combined conditions

	N	min	max	mean	var	skew	kurt	norm-t	norm-p	α
ctt:total	46	0	8	1.30	3.99	1.58	1.77	19.890	0.0000	0.7140
ctt:pretest	23	0	4	0.57	1.44	2.19	3.53	23.159	0.0000	0.6350
ctt:posttest	23	0	8	2.04	5.59	0.98	-0.02	4.808	0.0903	0.6951
ctt:abs_learn_gain	23	-1	6	1.48	4.35	0.97	-0.34	4.419	0.1097	0.6350
ctt:rel_learn_gain	23	0	0	0.05	0.00	0.95	-0.42	4.277	0.1178	0.6350
irt:total	46	-1	3	0.00	2.33	0.81	-0.90	8.324	0.0156	0.6015
irt:pretest	23	-1	4	0.01	2.56	1.65	1.53	13.943	0.0009	0.4629
irt:posttest	23	-1	2	0.00	2.04	0.19	-1.52	11.285	0.0035	0.6781
irt:abs_learn_gain	23	-3	3	-0.01	2.86	0.32	-0.85	1.311	0.5192	0.4629
irt:rel_learn_gain	23	0	0	0.02	0.00	0.24	-0.77	0.786	0.6749	0.4629
fixed irt:total	46	-2	2	-0.62	2.30	0.82	-0.86	8.005	0.0183	0.6058
fixed irt:pretest	23	-1	4	0.04	2.52	1.66	1.54	13.979	0.0009	0.4618
fixed irt:posttest	23	-1	2	-0.23	1.97	0.17	-1.56	12.676	0.0018	0.6732
fixed irt:abs_learn_gain	23	-3	2	-0.26	2.85	0.27	-0.87	1.246	0.5362	0.4618
fixed irt:rel_learn_gain	23	0	0	0.02	0.00	0.18	-0.76	0.640	0.7262	0.4618

Comparisons of the comprehension questions

Pre- and posttest comparisons

Table 24: Flashcard condition

	MW k	**MW p**	**t-test k**	**t-test p**
ctt	-2.683	0.0136	-2.683	0.0162
irt	-0.146	0.8852	-0.146	0.8852
fixed irt	1.856	0.0768	1.856	0.0770

Table 25: Flashmap condition

	MW k	**MW p**	**t-test k**	**t-test p**
ctt	-1.081	0.2926	-1.081	0.2971
irt	-0.018	0.9854	-0.018	0.9854
fixed irt	1.318	0.2024	1.318	0.2036

Table 26: Combined conditions

	MW k	**MW p**	**t-test k**	**t-test p**
ctt	-2.674	0.0105	-2.674	0.0116
irt	0.023	0.9818	0.023	0.9818
fixed irt	0.595	0.5549	0.595	0.5549

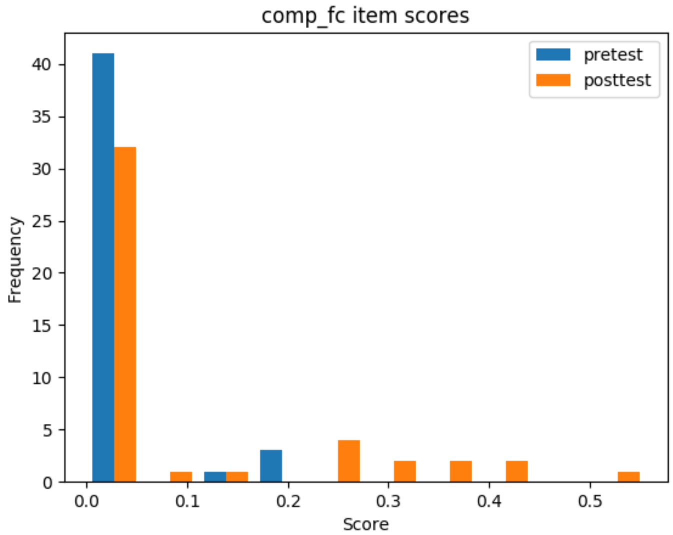


Figure 60: A histogram depicting the scores on the comprehension section of the pre- and posttest per item by flashcard users

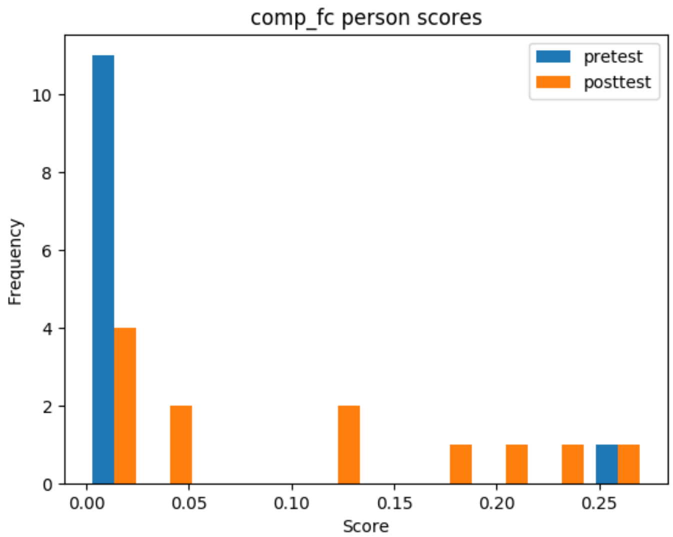


Figure 61: A histogram depicting the scores on the comprehension section of the pre- and posttest per flashcard user

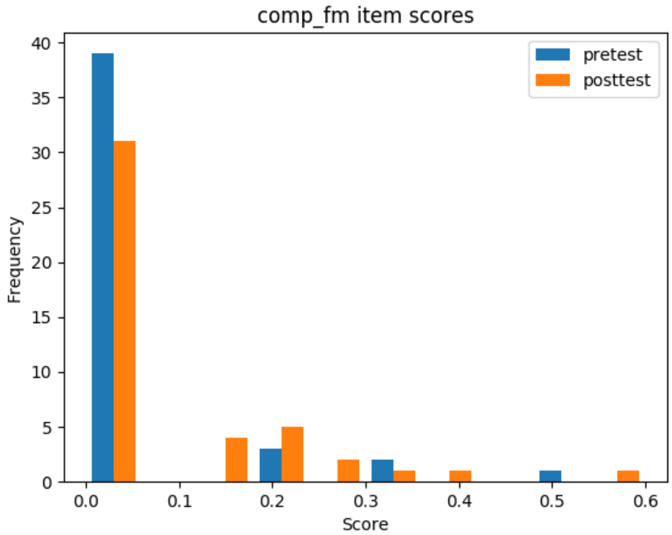


Figure 62: A histogram depicting the scores on the comprehension section of the pre- and posttest per item by flashmap users

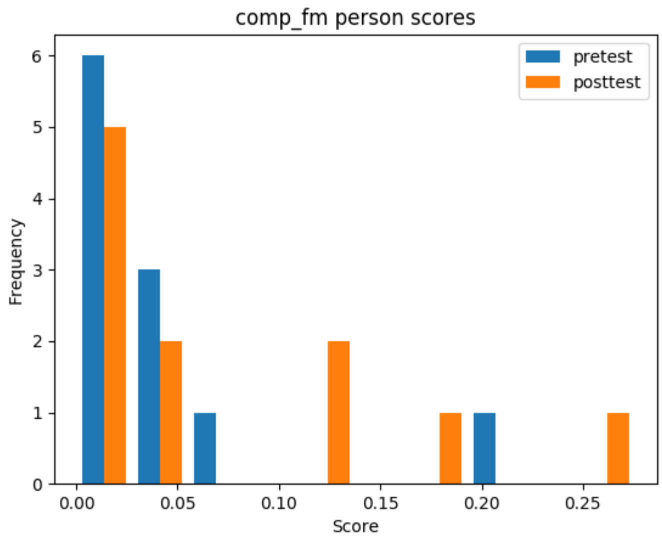


Figure 63: A histogram depicting the scores on the comprehension section of the pre- and posttest per flashmap user

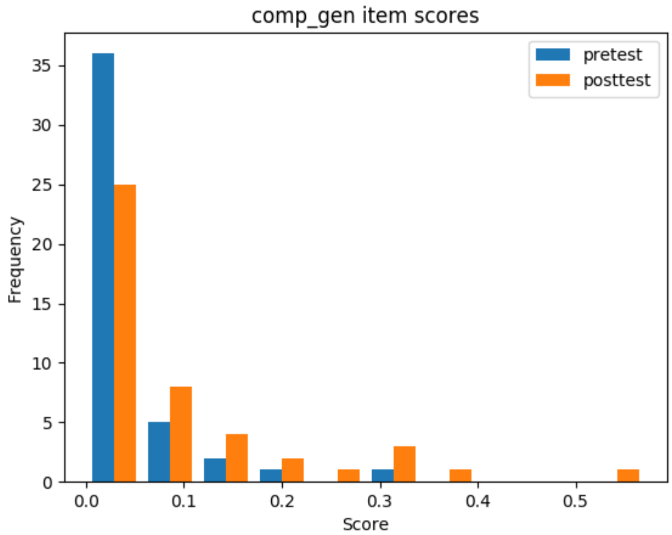


Figure 64: A histogram depicting the scores on the comprehension section of the pre- and posttest per item by users

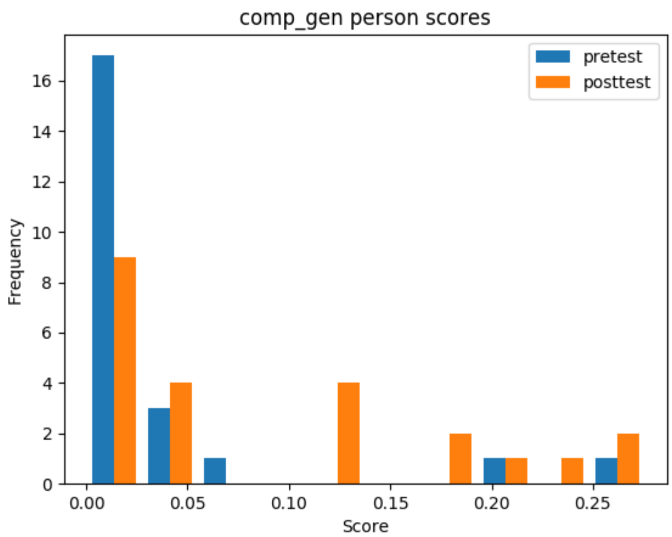


Figure 65: A histogram depicting the scores on the comprehension section of the pre- and posttest per user

Learning gain comparisons between conditions

Table 27: Classical test theory

	MW k	MW p	t-test k	t-test p
total	0.102	0.9195	0.102	0.9194
pretest	-0.967	0.3446	-0.963	0.3466
posttest	0.605	0.5515	0.603	0.5531
abs_learn_gain	1.270	0.2179	1.288	0.2124
rel_learn_gain	1.197	0.2448	1.211	0.2399

Table 28: Item response theory

	MW k	MW p	t-test k	t-test p
total	0.071	0.9436	0.072	0.9430
pretest	-0.132	0.8961	-0.132	0.8965
posttest	-0.002	0.9982	-0.002	0.9982
abs_learn_gain	0.112	0.9116	0.115	0.9097
rel_learn_gain	0.064	0.9496	0.066	0.9484

Table 29: Item response theory with fixed item difficulties

	MW k	MW p	t-test k	t-test p
total	-2.188	0.0340	-2.202	0.0330
pretest	-1.748	0.0950	-1.756	0.0936
posttest	-1.849	0.0785	-1.842	0.0802
abs_learn_gain	-0.407	0.6882	-0.415	0.6826
rel_learn_gain	-0.455	0.6537	-0.465	0.6476

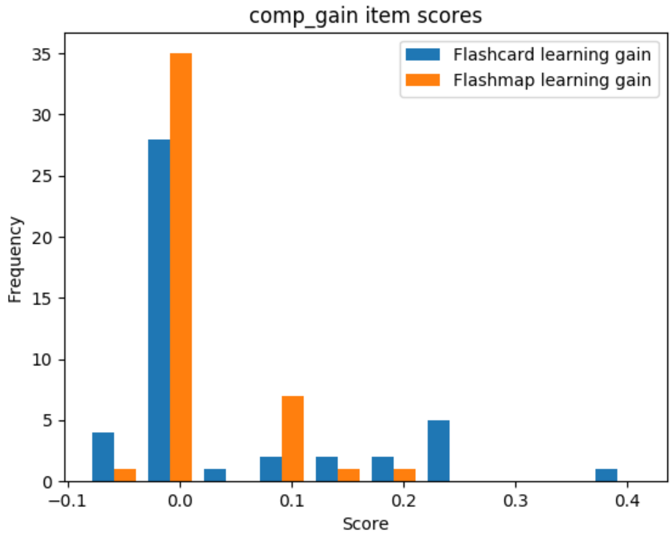


Figure 66: A comparison of figure 60 and 62, with the pretest score subtracted from the posttest score

Figure 67: A comparison of figure 61 and 63, with the pretest score subtracted from the posttest score

Instance statistics

Descriptives on the number of reviewed instances

Table 30: Flashcard condition

	sample	min	max	mean	variance	skew	kurtosis	normal-t	normal-p	α
abs	12	46	93	72.83	344.33	-0.34	-1.43	3.332	0.1890	0.9790
rel	12	0.49	1	0.78	0.04	-0.34	-1.43	3.332	0.1890	0.9790

Table 31: Flashmap condition

	sample	min	max	mean	variance	skew	kurtosis	normal-t	normal-p	α
abs	11	54	199	131.45	2363.47	0.07	-1.18	0.961	0.6186	0.9924
rel	11	0.27	1	0.66	0.06	0.07	-1.18	0.961	0.6186	0.9924

Table 32: Combined conditions

	sample	min	max	mean	variance	skew	kurtosis	normal-t	normal-p	α
rel	23	0.27	1	0.70	0.04	-0.00	-0.86	0.817	0.6647	0.9896

Comparisons of reviewed instances

	Mann-Whitney-U k	**Mann-Whitney-U p**	**Welch's t-test k**	**Welch's t-test p**
abs	-3.886	0.0009	-3.756	0.0025
rel	1.362	0.1875	1.350	0.1924

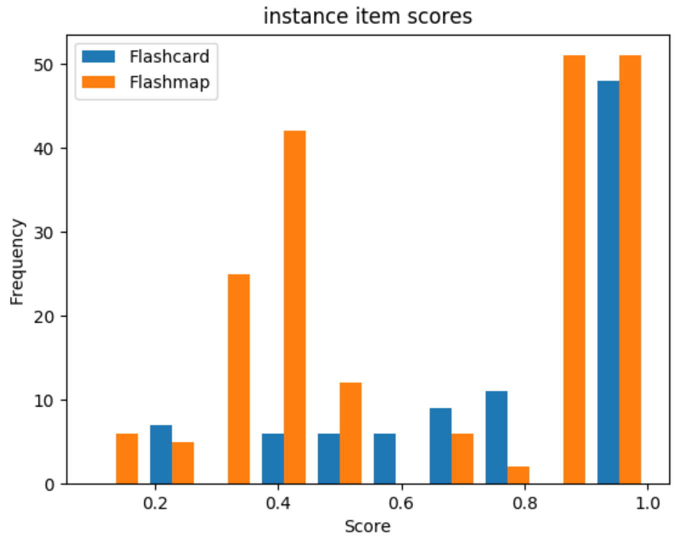


Figure 68: A histogram depicting the number of users covering an instance separate for the groups of flashcard and flashmap users

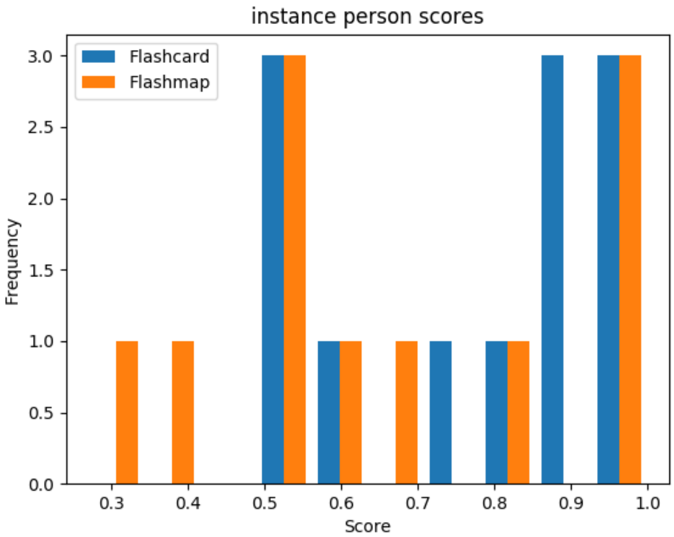


Figure 69: A histogram depicting the number of instances covered by a user separate for the groups of flashcard and flashmap users

Descriptives of the number of responses

Table 34: Flashcard condition

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
abs	12	298	1268	561.33	70295.70	1.69	2.26	13.701	0.0011	0.9378
rel	12	3	13	6.04	8.13	1.69	2.26	13.701	0.0011	0.9378
mean	12	5	14	7.61	5.84	2.33	4.62	23.813	0.0000	0.9378

Table 35: Flashmap condition

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
abs	11	344	1555	729.73	126333.42	1.13	0.55	5.693	0.0581	0.9832
rel	11	1	7	3.67	3.19	1.13	0.55	5.693	0.0581	0.9832
mean	11	3	7	5.61	1.75	0.03	-0.80	0.063	0.9690	0.9832

Table 36: Combined conditions

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
abs	23	298	1555	641.87	99969.48	1.41	1.42	11.547	0.0031	0.9579
mean	23	3	14	6.65	4.72	2.17	6.57	28.261	0.0000	0.9572

Comparisons of the number of responses

	MW k	**MW p**	**t-test k**	**t-test p**
abs	-1.295	0.2092	-1.279	0.2169
rel	2.361	0.0280	2.409	0.0265
mean	2.429	0.0242	2.489	0.0233

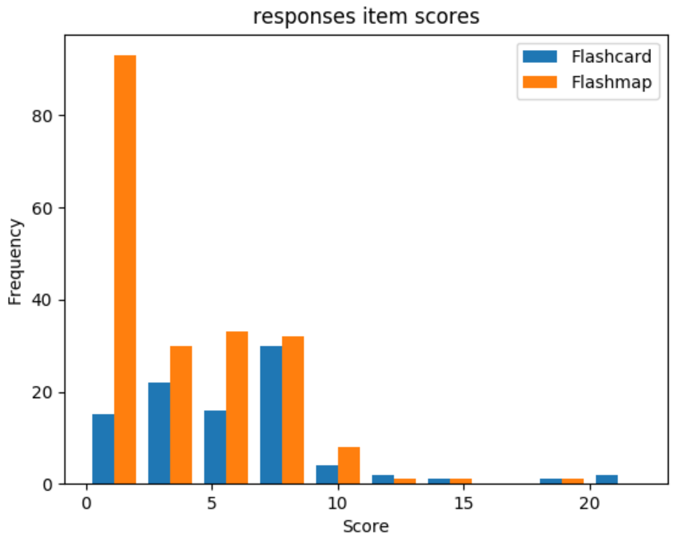


Figure 70: A histogram depicting the number of responses per instance separate for the groups of flashcard and flashmap users

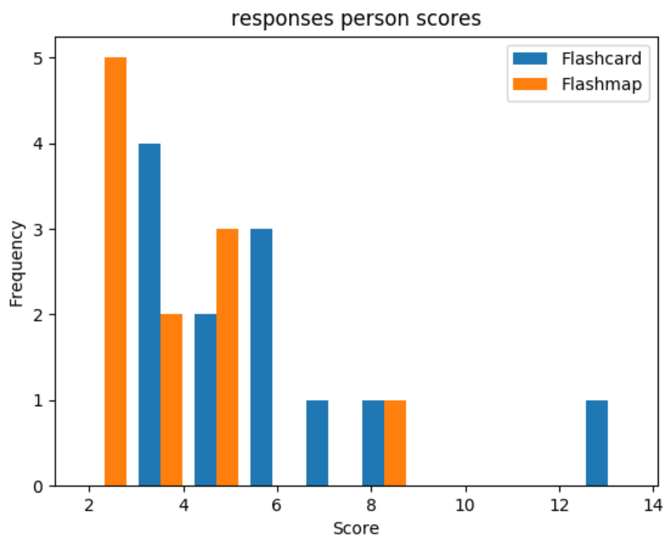


Figure 71: A histogram depicting the number of responses per user separate for the groups of flashcard and flashmap users

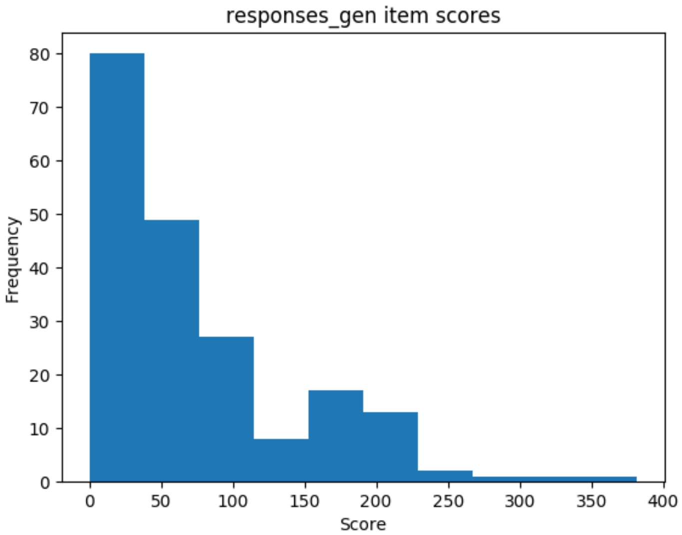


Figure 72: A histogram depicting the number of responses per instance for the combined group of flashcard and flashmap users

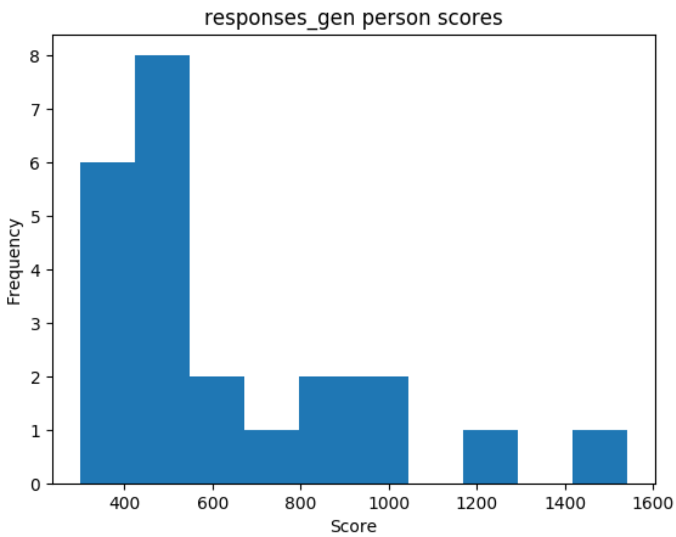


Figure 73: A histogram depicting the number of responses per user for the combined group of flashcard and flashmap users

Descriptives of the exponents of instances

Table 38: Flashcard condition

	sample	min	max	mean	variance	skew	kurtosis	normal-t	normal-p	α
abs	12	218	966	495.42	41792.81	0.86	0.42	3.894	0.1427	0.8933
rel	12	2	10	5.33	4.83	0.86	0.42	3.894	0.1427	0.8933
mean	12	4	11	6.67	2.66	1.83	3.42	17.448	0.0002	0.8933

Table 39: Flashmap condition

	sample	min	max	mean	variance	skew	kurtosis	normal-t	normal-p	α
abs	11	330	1523	842.27	132229.22	0.67	-0.65	1.467	0.4803	0.9800
rel	11	1	7	4.21	3.31	0.67	-0.65	1.467	0.4803	0.9800
mean	11	5	8	6.38	1.01	1.14	0.11	4.835	0.0891	0.9800

Table 40: Combined conditions

	sample	min	max	mean	variance	skew	kurtosis	normal-t	normal-p	α
abs	23	218	1523	661.30	112385.58	1.09	0.62	6.912	0.0316	0.9632
mean	23	4	11	6.53	1.81	1.95	4.68	23.191	0.0000	0.9632

Comparisons of the exponents

	Mann-Whitney-U k	**Mann-Whitney-U p**	**Welch's t-test k**	**Welch's t-test p**
abs	-2.853	0.0095	-2.786	0.0136
rel	1.319	0.2013	1.330	0.1978
mean	0.497	0.6246	0.507	0.6182

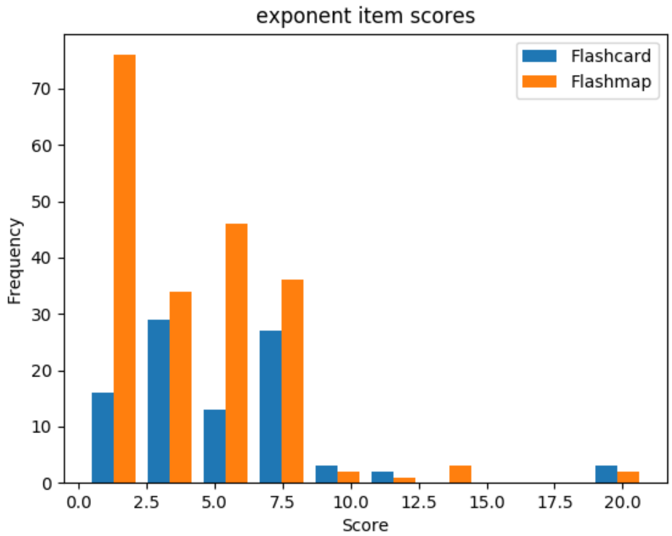


Figure 74: A histogram depicting the exponents per instance separate for the flashcard and flashmap users

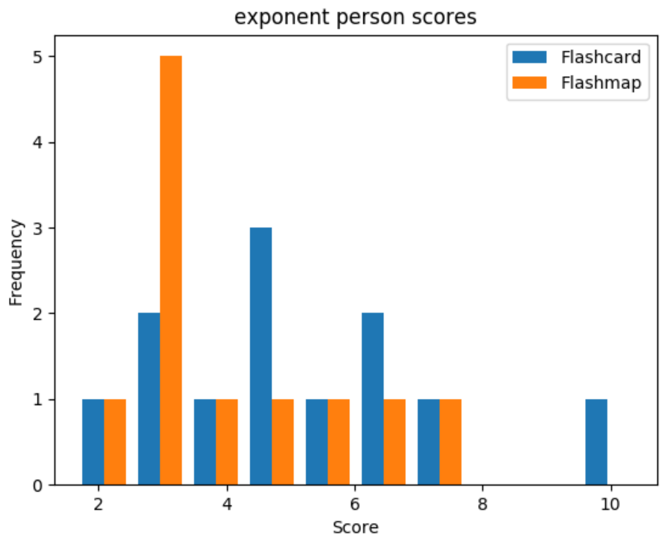


Figure 75: A histogram depicting the exponents per user separate for the flashcard and flashmap users

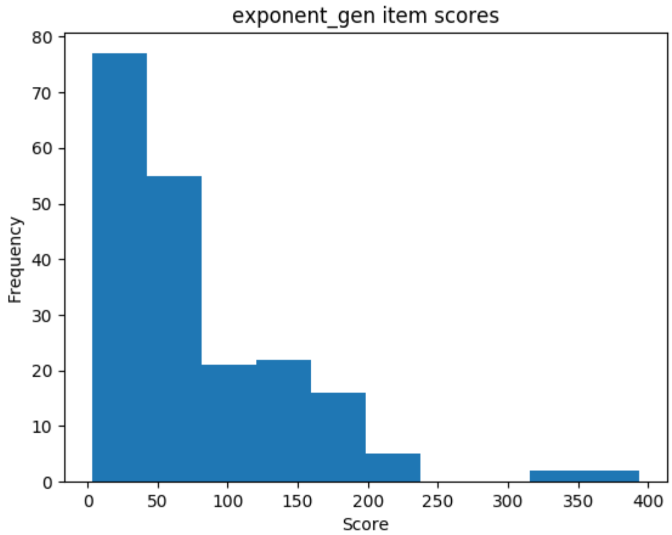


Figure 76: A histogram depicting the exponents per instance for the combined group of flashcard and flashmap users

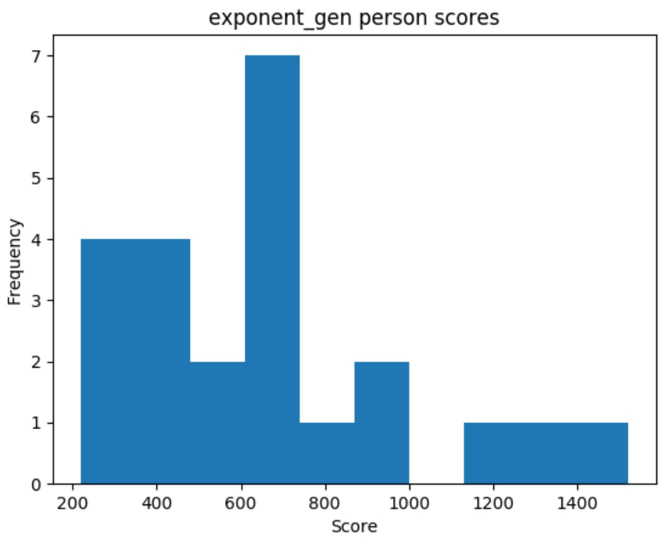


Figure 77: A histogram depicting the exponents per user for the combined group of flashcard and flashmap users

Descriptives of percentage of responses marked as correct

Table 42: Flashcard condition

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
abs	12	35	86	62.33	267.67	-0.30	-1.07	0.993	0.6086	0.9780
rel	12	0.38	0.93	0.67	0.03	-0.30	-1.07	0.993	0.6086	0.9780
mean	12	0	0	0.86	0.00	-0.19	-0.82	0.242	0.8859	0.9780

Table 43: Flashmap condition

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
abs	11	46	185	117.55	2006.59	0.05	-1.06	0.548	0.7605	0.9928
rel	11	0.23	0.93	0.59	0.05	0.05	-1.06	0.548	0.7605	0.9928
mean	11	0	1	0.89	0.00	0.22	-1.19	1.178	0.5549	0.9928

Table 44: Combined conditions

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
abs	23	35	185	88.74	1841.51	0.93	-0.13	4.256	0.1191	0.9910
mean	23	0	1	0.87	0.00	0.07	-0.67	0.272	0.8728	0.9910

Comparisons of the percentage of responses marked as correct

	MW k	**MW p**	**t-test k**	**t-test p**
abs	-16.597	0.0000	-15.857	0.0000
rel	-16.421	0.0000	-15.689	0.0000
mean	-12.448	0.0000	-11.895	0.0000

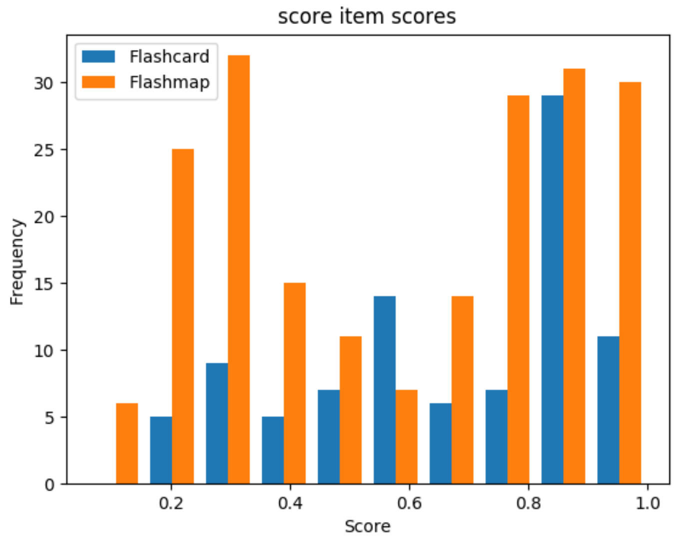


Figure 78: A histogram depicting percentages of correct answers per instance separate for the flashcard and flashmap users

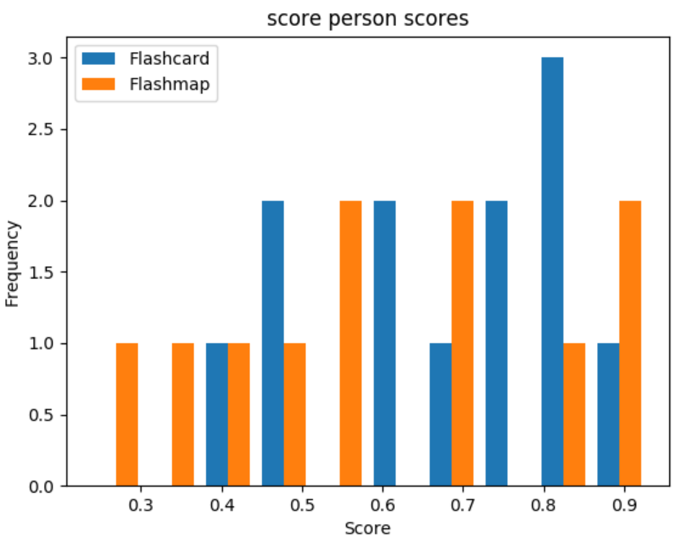


Figure 79: A histogram depicting percentages of correct answers per user separate for the flashcard and flashmap users

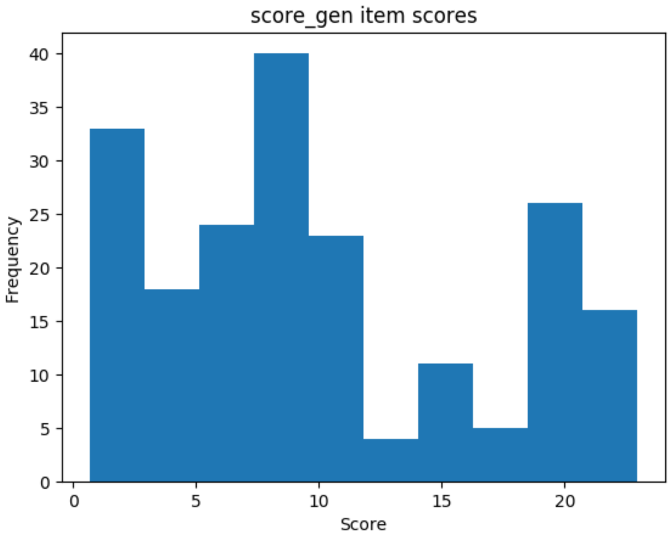


Figure 80: A histogram depicting percentages of correct answers per instance for the combined group of flashcard and flashmap users

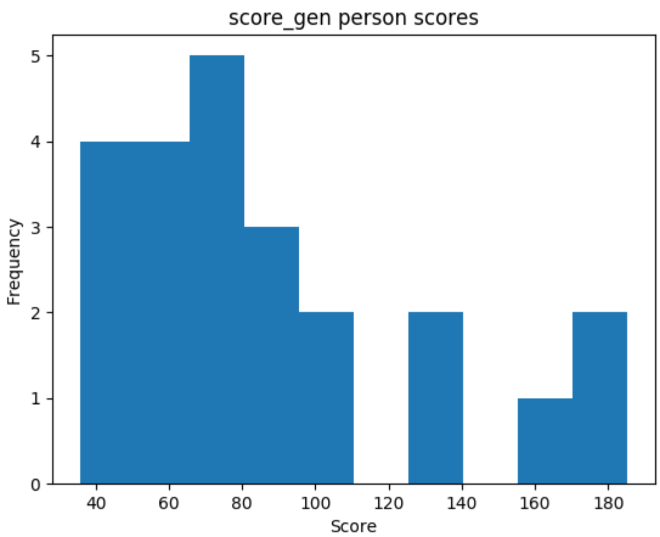


Figure 81: A histogram depicting percentages of correct answers per user for the combined group of flashcard and flashmap users

Descriptives of the amount of time spent on the application

Table 46: Flashcard condition

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
abs	12	1697	19721	12374.41	26140529.78	-0.61	-0.33	1.445	0.4855	0.9239
rel	12	18	212	133.06	3022.38	-0.61	-0.33	1.445	0.4855	0.9239
mean	12	35	249	169.77	3881.30	-1.00	0.11	4.064	0.1311	0.9239

Table 47: Flashmap condition

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
abs	11	2612	26869	14121.58	51451085.83	0.65	-0.07	1.907	0.3854	0.9501
rel	11	13	135	70.96	1299.24	0.65	-0.07	1.907	0.3854	0.9501
mean	11	19	224	117.30	3303.44	0.24	-0.34	0.380	0.8271	0.9501

Table 48: Combined conditions

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
abs	23	1697	26869	13210.01	37253452.58	0.43	0.55	2.350	0.3088	0.9281
mean	23	19	249	144.67	4160.26	-0.29	-0.93	1.630	0.4427	0.9281

Comparisons of the amount of time spent on the application

	MW k	MW p	t-test k	t-test p
abs	7.522	0.0000	7.869	0.0000
rel	7.787	0.0000	8.148	0.0000
mean	8.720	0.0000	9.126	0.0000

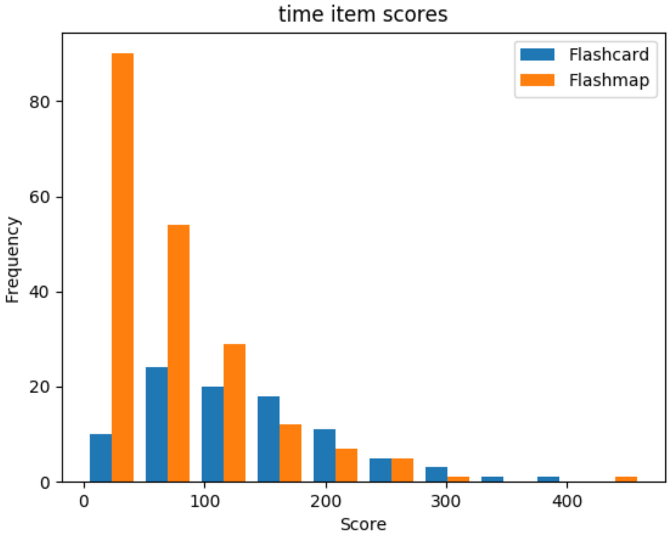


Figure 82: A histogram depicting the time spent in seconds per instance separate for the flashcard and flashmap users

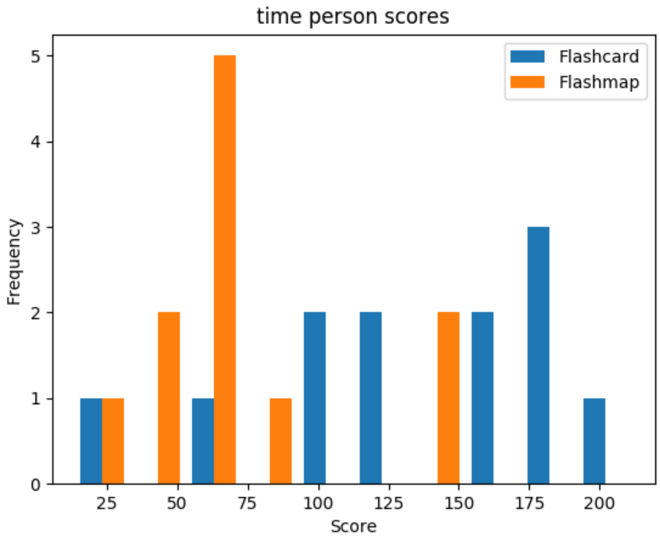


Figure 83: A histogram depicting the time spent in seconds per user separate for the flashcard and flashmap users

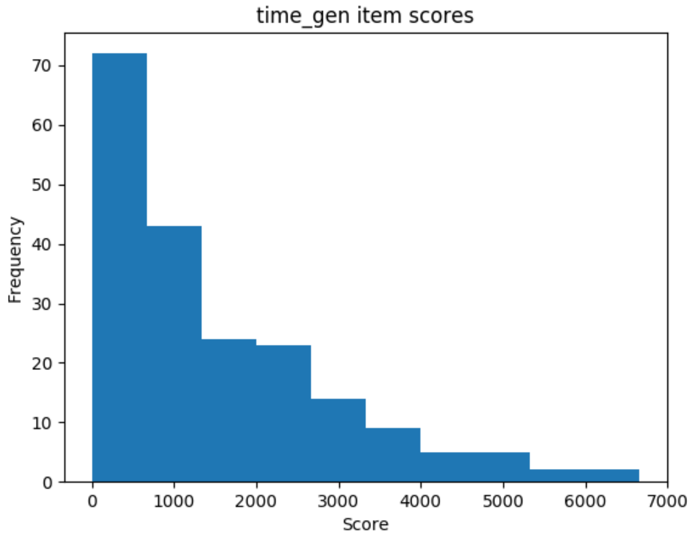


Figure 84: A histogram depicting the time spent in seconds per instance for the combined group of flashcard and flashmap users

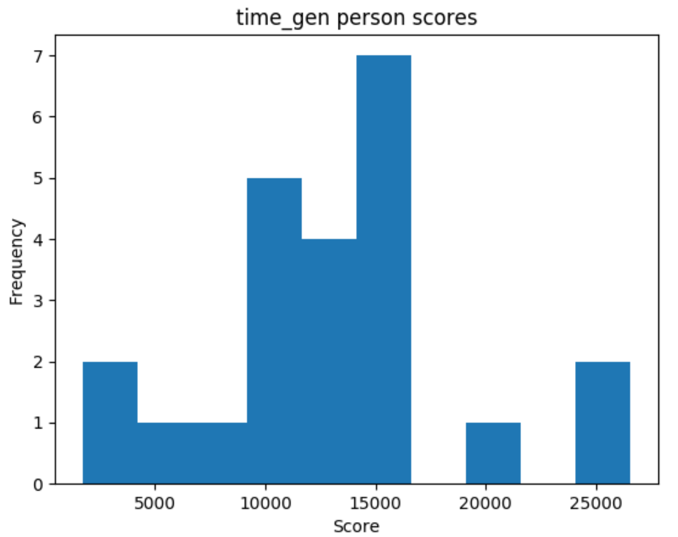


Figure 85: A histogram depicting the time spent in seconds per user for the combined group of flashcard and flashmap users

Questionnaire statistics

Descriptives of Perceived Usefulness questions

Table 50: Flashcard condition

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
ctt	12	-4	14	6.50	27.36	-0.57	-0.52	1.144	0.5643	0.6432
irt	12	-5	1	-0.16	4.40	-1.89	3.15	17.284	0.0002	0.6263

Table 51: Flashmap condition

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
ctt	11	0	13	8.82	15.16	-1.05	0.32	4.698	0.0955	0.6777
irt	11	-3	1	-0.15	1.89	-1.41	1.44	9.670	0.0079	0.5298

Table 52: Combined conditions

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
ctt	23	-4	14	7.61	21.98	-0.86	-0.02	3.864	0.1448	0.6509

Comparisons of the Perceived Usefulness questions

	MW k	**MW p**	**t-test k**	**t-test p**
ctt	-1.196	0.2449	-1.212	0.2395

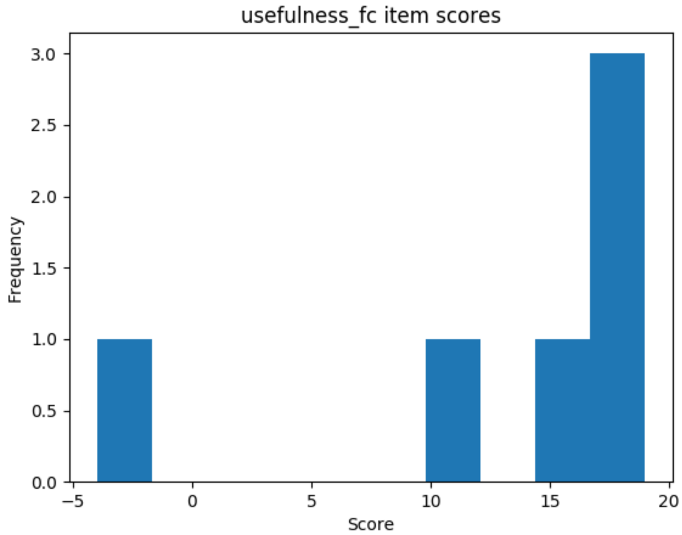


Figure 86: A histogram depicting the scores per Perceived Usefulness item by flashcard users

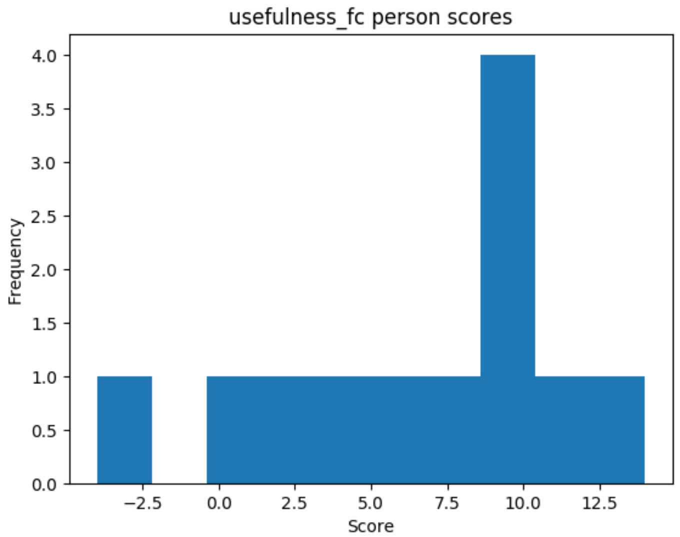


Figure 87: A histogram depicting the scores on the Perceived Usefulness items per flashcard user

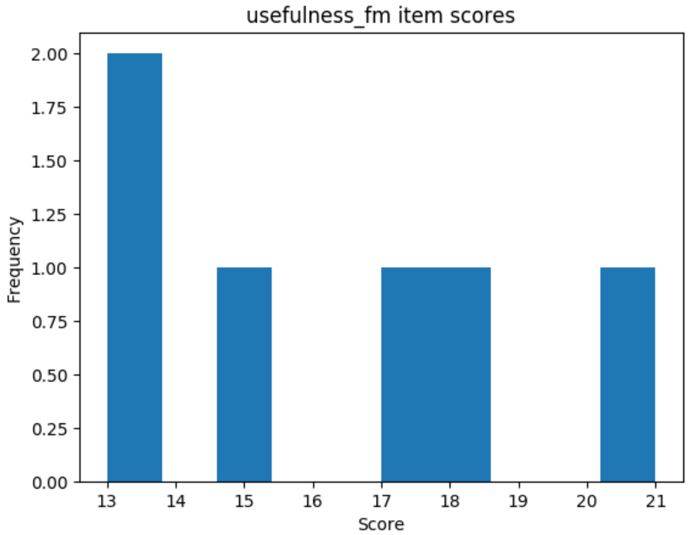


Figure 88: A histogram depicting the scores per Perceived Usefulness item by flashmap users

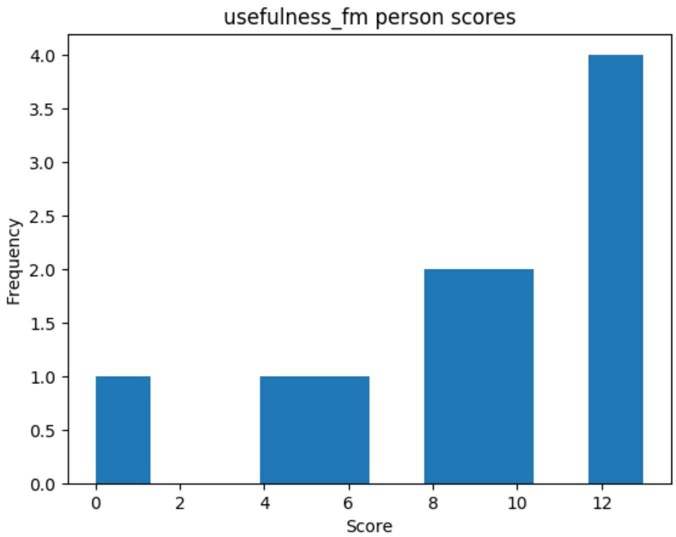


Figure 89: A histogram depicting the scores on the Perceived Usefulness items per flashmap user

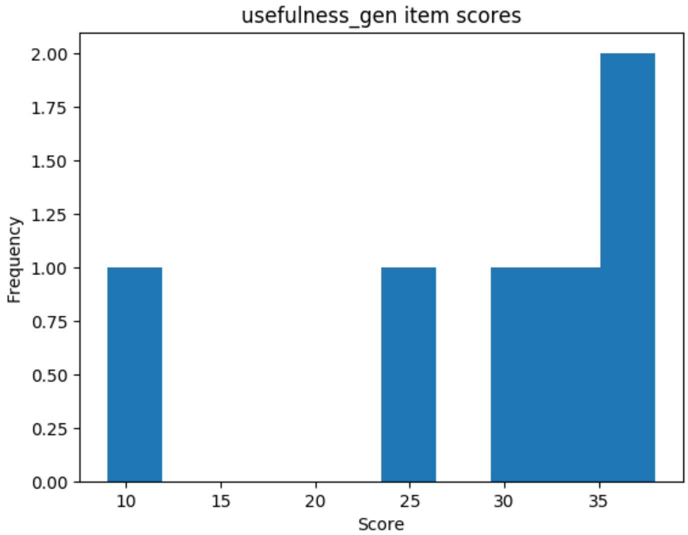


Figure 90: A histogram depicting the scores per Perceived Usefulness item

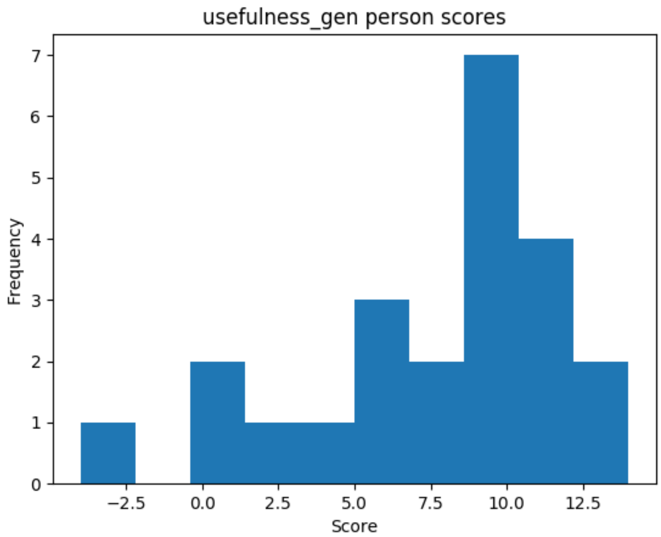


Figure 91: A histogram depicting the scores on the Perceived Usefulness items per user

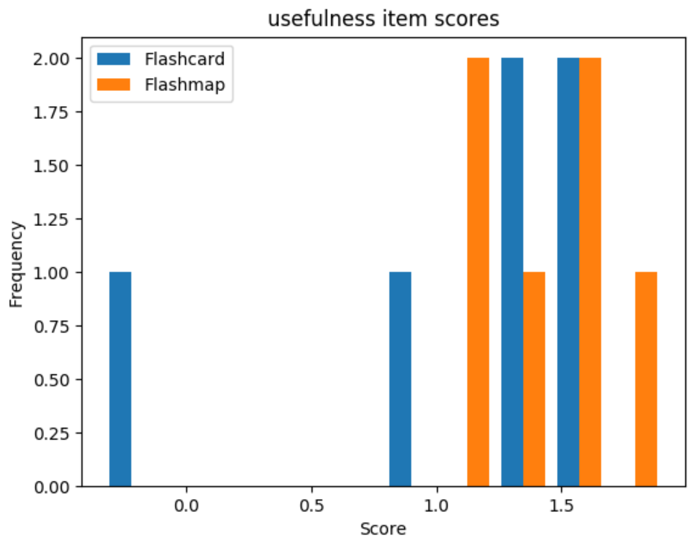


Figure 92: A comparison of figure 86 and 88

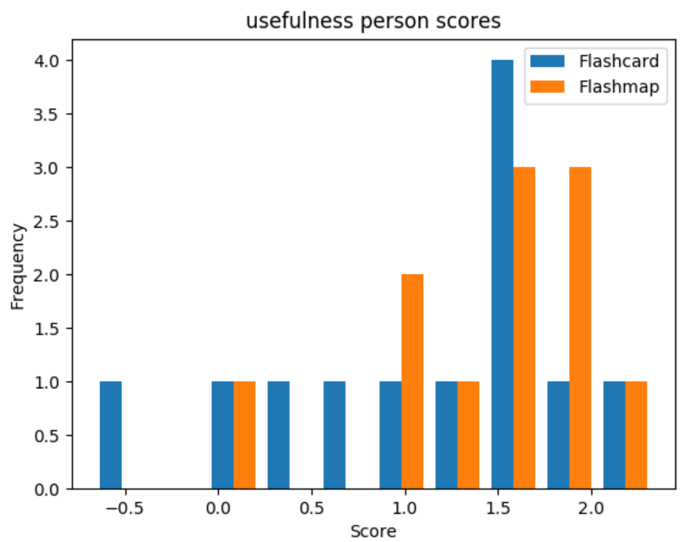


Figure 93: A comparison of figure 87 and 89

Descriptives of Perceived Ease of Use questions

Table 54: Flashcard condition

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
ctt	12	-4	17	6.58	38.08	-0.26	-0.62	0.232	0.8904	0.8794
irt	12	0	4	0.91	1.87	1.08	0.52	5.358	0.0686	0.2295

Table 55: Flashmap condition

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
ctt	11	0	19	8.27	26.22	0.50	0.12	1.725	0.4220	0.7689
irt	11	-2	2	0.22	1.87	-0.20	1.01	3.041	0.2186	0.2538

Table 56: Combined conditions

	N	min	max	mean	variance	skew	kurt	norm-t	norm-p	α
ctt	23	-4	19	7.39	31.70	-0.08	-0.10	0.239	0.8876	0.8285

Comparisons of the Perceived Usefulness questions

	MW k	**MW p**	**t-test k**	**t-test p**
ctt	-0.711	0.4851	-0.717	0.4816

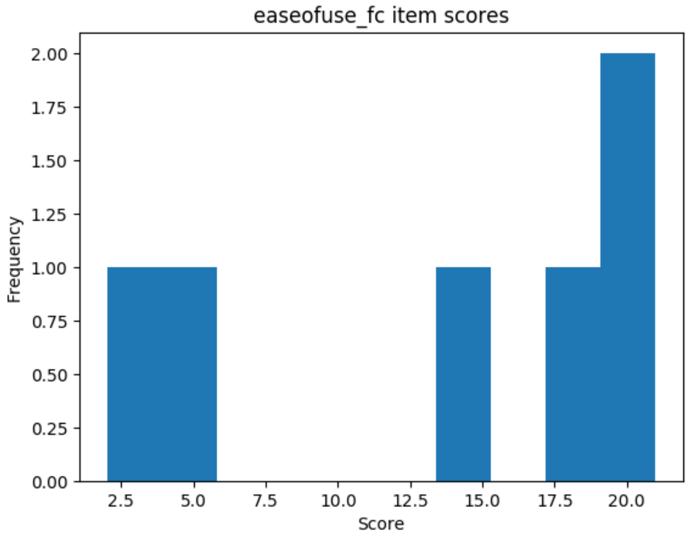


Figure 94: A histogram depicting the scores per Perceived Ease of use item by flashcard users

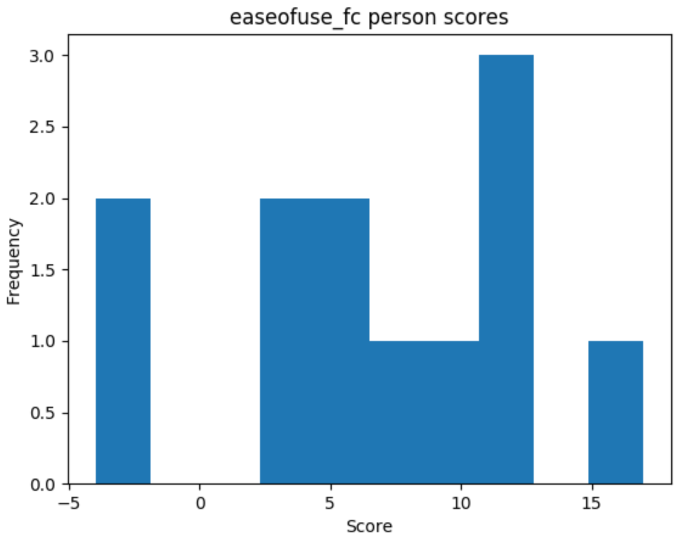


Figure 95: A histogram depicting the scores on the Perceived Ease of use items per flashcard user

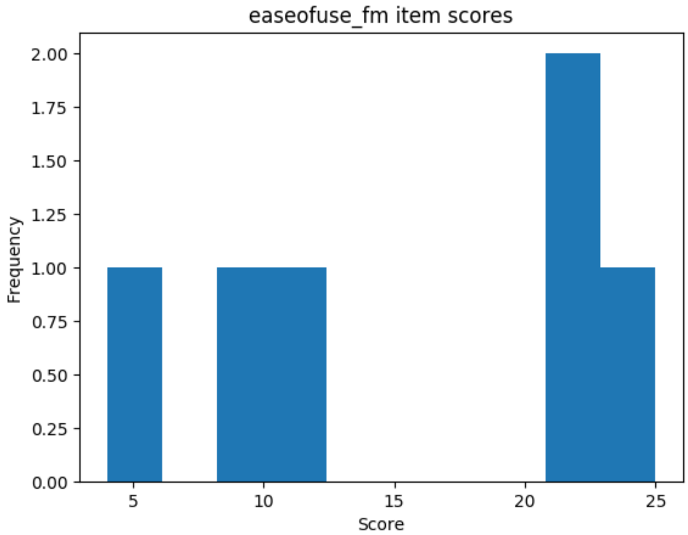


Figure 96: A histogram depicting the scores per Perceived Ease of use item by flashmap users

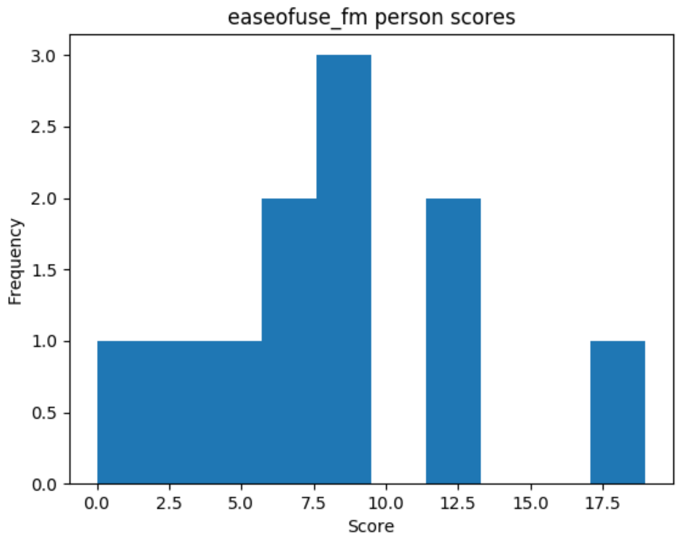


Figure 97: A histogram depicting the scores on the Perceived Ease of use items per flashmap user

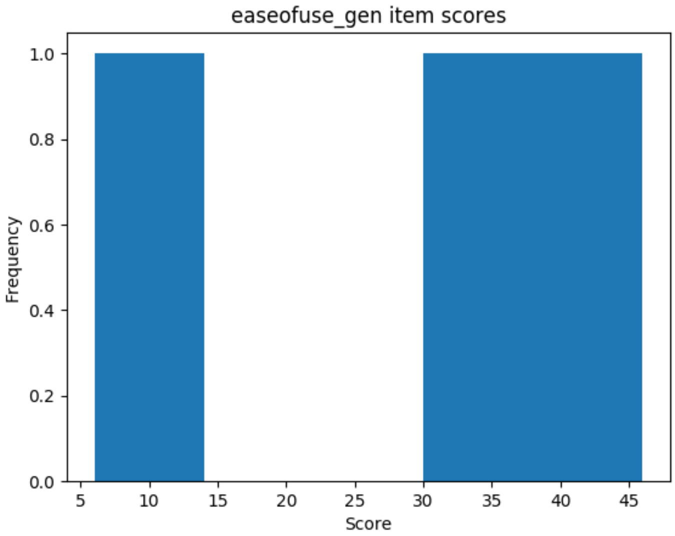


Figure 98: A histogram depicting the scores per Perceived Ease of use item

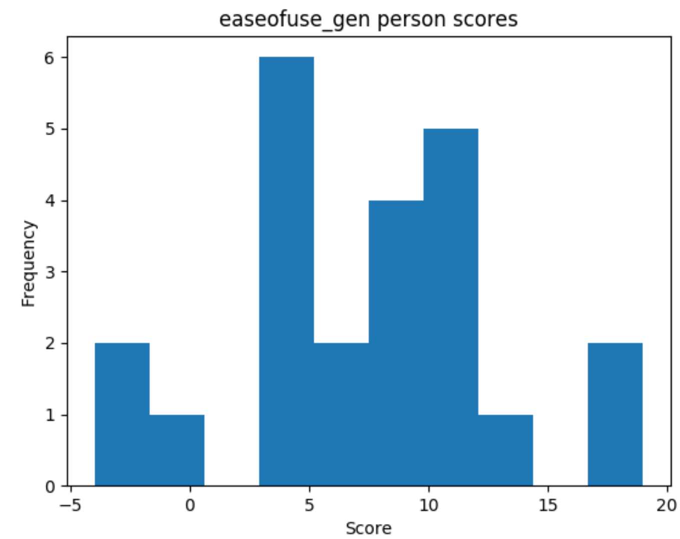


Figure 99: A histogram depicting the scores on the Perceived Ease of use items per user

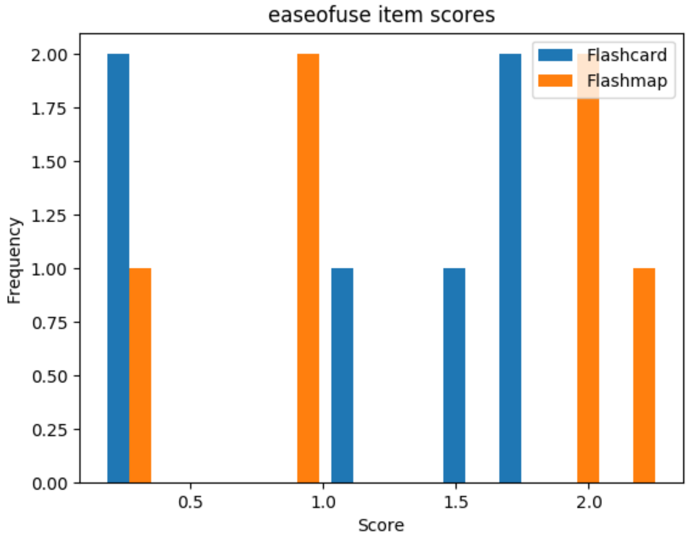


Figure 100: A comparison of figure 94 and 96

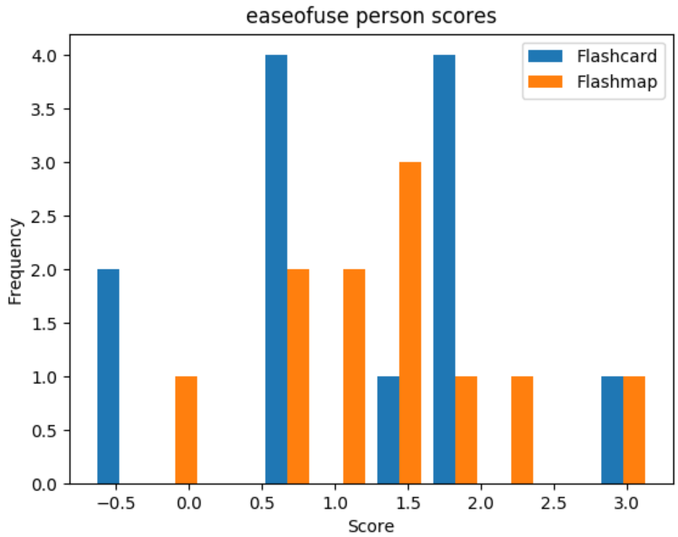


Figure 101: A comparison of figure 95 and 97