ADAM

ADAPtive Microlearning using virtual flashcards in blended learning

Diogo Filipe Vieitos Pais

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisor: Prof. Hugo Miguel Aleixo Albuquerque Nicolau

Examination Committee

Chairperson: Prof. Miguel Nuno Dias Alves Pupo Correia
Supervisor: Prof. Hugo Miguel Aleixo Albuquerque Nicolau
Member of the Committee: Prof. João Manuel Brisson Lopes

October 2017
Acknowledgements

Quero agradecer em primeiro lugar ao meu orientador, o Prof. Dr. Hugo Nicolau, que não só aceitou prontamente o tema de tese que lhe propus como ainda me pôs nas mãos os primeiros papers, um dos quais foi extremamente útil para o arranque e desenvolvimento da minha tese. Para além disto, sempre me deu excelente feedback, excelentes sugestões e abriu-me todas as portas para que eu conseguisse chegar onde cheguei. Um muito grande obrigado por tudo isto!

Em segundo lugar, gostaria de agradecer a todos os professores da cadeira de Produção de Conteúdos Multimédia, que me permitiram testar a minha aplicação no mundo real com utilizadores reais e que se mostraram dispostos a partilhar comigo as suas opiniões e sugestões em relação à minha tese. Muito obrigado!

Quero ainda agradecer a todos os estudantes da cadeira de Produção de Conteúdos Multimédia do 2º semestre de 2016/2017 por terem utilizado a aplicação e terem participado nas entrevistas que realizei. Sem vocês isto não teria sido possível, e por isso vos agradeço.

Formalismos à parte, estou extremamente feliz por chegar ao fim desta longa, longa etapa. Custou, e custou muito, aqui chegar. E para ser sincero, ainda custa um pouco a acreditar, até! No entanto, se aqui cheguei foi graças aos meus amigos e à minha família.

Posso não ter feito muita coisa nestes 23 anos de existência, mas se há algo que me orgulho de ter feito são as minhas amizades. Quero agradecer ao Vasco Silva, ao Tiago Castanheira, ao Pedro Silva e ao Hugo Farinha por me terem acompanhado desde os 15 anos nesta longa jornada. Que estes longos 8 anos sejam apenas o começo e que pareçam segundos comparados com tudo aquilo que nos espera daqui para a frente.

Agradeco ainda aos meus fiéis companheiros de curso André Domingues e Francisco Gonçalves, com os quais passei muitas, muitas, demasiadas horas, noites e madrugadas a trabalhar e a festejar. Posso estar enganado, mas acho que nenhum dos três teria conseguido este feito se um dos outros faltasse. Que continuemos a bater recordes vida fora! Se sobrevivemos a isto, sobrevivemos a tudo!

Podia estar aqui uma eternidade a tentar lembrar-me dos nomes de todas as pessoas que me ajudaram, mas se há nomes que nunca terei de me esforçar por lembrar são os nomes Anabela Pais e Jorge Pais. Sempre me deram toda a liberdade e apoio financeiro que alguma vez precisei, sem que nada lhes tivesse de pedir uma única vez. É quase inútil agradecer-lhes num pedaço de papel como este, mas ainda assim o faço. Obrigado por me darem uma vida.
Abstract

University classes require students to learn large amounts of information in order to succeed. Even though most or all of this information is lectured in physical classes, such classes target the body of students as a whole. Therefore, some students may not be able to keep up with them and may miss information due to distractions. Factors like lack of motivation and time constraints may also prevent students from keeping their own regular study schedule. This dissertation describes an adaptive mobile microlearning application based on virtual flashcards and its integration in a university class, as well as the results obtained from the usage data collected and interviews conducted with teachers and students of this class. Both think this approach is very promising and effective, and it seems to benefit both parties. First results suggest that grades increased when ADAM was introduced, even though students only used it near evaluation dates.

**Keywords:** adaptive, mobile, microlearning, application, flashcards, blended, learning, university
Resumo

As cadeiras universitárias requerem que os alunos aprendam grandes quantidades de informação para terem sucesso. Ainda que a maioria ou totalidade desta informação seja dada em aulas físicas, estas aulas têm como alvo o corpo de estudantes como um todo. Alguns estudantes podem, por isso, não conseguir acompanhá-las e deixar escapar informação devido a distrações. Factores como falta de motivação e de tempo podem ainda impedir os estudantes de efectuar um estudo regular. Esta dissertação descreve uma aplicação móvel de micro-aprendizagem adaptável baseada em cartões de memória virtuais e a sua integração numa cadeira universitária, bem como os resultados obtidos a partir dos dados de utilização recolhidos e de entrevistas realizadas com professores e estudantes desta cadeira. Ambos consideram esta abordagem muito promissora e eficaz, e parece beneficiar ambas as partes. Os primeiros resultados sugerem que as notas subiram com a introdução do ADAM, ainda que os estudantes só o utilizassem perto das datas das suas avaliações.

Palavras-chave: aplicação, móvel, micro-aprendizagem, adaptável, cartões de memória, aprendizagem, híbrida, universidade
Table of Contents

ACKNOWLEDGEMENTS ......................................................................................................................... II

ABSTRACT .............................................................................................................................................. III

RESUMO ................................................................................................................................................ IV

TABLE OF CONTENTS .......................................................................................................................... V

LIST OF FIGURES ............................................................................................................................... VIII

LIST OF TABLES .................................................................................................................................... X

LIST OF ACRONYMS ............................................................................................................................. XI

1. INTRODUCTION ............................................................................................................................... 12
   1.1. Problem ...................................................................................................................................... 13
   1.2. Approach ................................................................................................................................. 14
   1.3. Objectives .............................................................................................................................. 15

2. RELATED WORK ............................................................................................................................. 16
   2.1. Learning and cognitive sciences .......................................................................................... 16
       2.1.1. Testing effect .................................................................................................................. 16
       2.1.2. Spacing effect ................................................................................................................. 17
       2.1.3. Forgetting curve ............................................................................................................ 18
   2.2. Mobile and Microlearning theory ..................................................................................... 19
   2.3. Blended learning ................................................................................................................... 20
   2.4. Microlearning applications ............................................................................................... 21
       2.4.1. MemReflex .................................................................................................................... 21
       2.4.2. MicroMandarin ............................................................................................................. 22
       2.4.3. Tip Tap Tones ............................................................................................................. 23
       2.4.4. Gassler et al. ................................................................................................................. 24
       2.4.5. Chen et al. ................................................................................................................... 26
       2.4.6. µLearn ........................................................................................................................... 27
       2.4.7. LingoSnacks ................................................................................................................ 29
   2.5. Discussion ............................................................................................................................. 31
       2.5.1. Study approach .............................................................................................................. 32
       2.5.2. Contextual learning ...................................................................................................... 32
       2.5.3. Mobile learning ............................................................................................................ 32
       2.5.4. Time constraints ............................................................................................................ 32
APPENDIXES

1. APPENDIX A – STUDENT QUESTIONNAIRE .................................................. 89
2. APPENDIX B – TEACHER QUESTIONNAIRE ............................................. 89
3. Appendix C – Usability Tests Script .................................................................................................................. 90
4. Appendix D – Usability Tests Resources ......................................................................................................... 95
List of figures

Figure 1 – Forgetting curve resulting from adaptive spaced repetition learning [8] ........................................ 18
Figure 2 – MemReflex’s adaptive flashcards [8] ................................................................................................ 21
Figure 3 – MicroMandarin’s contextual microlearning [13] ........................................................................ 22
Figure 4 – Tip Tap Tones example [14] ........................................................................................................... 23
Figure 5 – Gassler et al.’s learning screensaver [15] ....................................................................................... 24
Figure 6 – Chen et al.’s adaptive mobile microlearning application [16] .............................................................. 26
Figure 7 – µLearn mobile microlearning application [17] .............................................................................. 27
Figure 8 – LingoSnacks mobile microlearning application [18] .................................................................... 29
Figure 9 – The architecture of our solution, ADAM ...................................................................................... 37
Figure 10 – In the Leitner system, correctly answered cards are advanced to the next, less frequent box, while incorrectly answered cards return to the first box [33] ......................................................... 39
Figure 11 – ADAM’s starting screen .................................................................................................................. 44
Figure 12 – “Issue report” page in ADAM ........................................................................................................ 45
Figure 13 – “My courses” page in ADAM ........................................................................................................ 46
Figure 14 – Mobile version of the “My courses” page in ADAM. (a) Left panel collapsed. (b) Left panel expanded ........................................................................................................................................ 47
Figure 15 – “Create course” page in ADAM .................................................................................................... 48
Figure 16 – “Edit course” page in ADAM .......................................................................................................... 48
Figure 17 – “View course” page in ADAM ......................................................................................................... 49
Figure 18 – “Manage teachers” page in ADAM ............................................................................................... 50
Figure 19 – “Create semester” page in ADAM ............................................................................................... 51
Figure 20 – “Edit semester” page in ADAM .................................................................................................... 51
Figure 21 – “Clone semester” page in ADAM ................................................................................................. 52
Figure 22 – “View semester” page in ADAM as seen by an enrolled student ................................................. 53
Figure 23 – “View semester” page in ADAM as seen by a student not enrolled ............................................. 53
Figure 24 – “View semester” page in ADAM as seen by a student not enrolled when the semester is closed ........................................................................................................................................ 54
Figure 25 – “Semester leaderboard” page in ADAM ....................................................................................... 54
Figure 26 – “View semester” page in ADAM as seen by a course administrator ............................................ 55
Figure 27 – “Create deck” page in ADAM ....................................................................................................... 56
Figure 28 – Example of created deck in the semester’s deck list ....................................................................... 57
Figure 29 – “View deck” page in ADAM ........................................................................................................... 58
Figure 30 – “Deck leaderboard” page in ADAM ............................................................................................. 59
Figure 31 – Deck deletion confirmation in “Edit deck” page in ADAM ............................................................ 59
Figure 32 – “View deck” page in ADAM as seen by a teacher ......................................................................... 60
Figure 33 – “Deck stats” page in ADAM .......................................................................................................... 61
List of tables

Table 1 – Feature comparison between reviewed applications.................................................................31
Table 2 – Examples of inter-study intervals for the first 10 boxes in the Leitner system calculated with ADAM’s adaptive spaced repetition algorithm using three different modifier values [33].................................41
Table 3 – Multimedia Content Production mini-test grades. Gray cells represent mini-tests where ADAM was available for the students of this class................................................................................................................................70
Table 4 – Multimedia Content Production mini-test grades of the 36 most active ADAM users. Gray cells represent mini-tests where ADAM was available for the students of this class...........................................71
Table 5 – Multimedia Content Production mini-test grades of the 37 least active ADAM users. Gray cells represent mini-tests where ADAM was available for the students of this class.................................................71
Table 6 – Task execution times gathered in the usability tests, in the format “mm:ss”. ..................................80
Table 7 – Single ease question answers gathered in the usability tests....................................................81
List of acronyms

ADAM – ADAptive Microlearning using virtual flashcards in blended learning
API – Application Programming Interface
CSS – Cascading Style Sheets
HTML5 – HyperText Markup Language 5
IQR – Interquartile Range
ISI – Inter-study Interval(s)
REST – Representational State Transfer
UI – User Interface
1. Introduction

We live in an era where information is easily accessible anywhere and at any time. As a consequence, we are also bombarded with information during almost every single minute that we are awake. This results in our brains getting used to look at information and ignore it even if it is potentially helpful [1]. Most of the times this is not a problem since we can access that same information later whenever we need it. However, there are contexts where it is convenient or even crucial to save this information in the long term (e.g. foreign language learning, school classes).

In the context of college education, memorization is an essential requirement for the success of students. Students are evaluated periodically on the materials studied in class and they need to have these materials memorized in order to correctly fulfill the tasks needed to pass those evaluations. Even when these evaluations allow students to access materials, it is still beneficial for the student to know them at least partially as it will avoid wasting evaluation time searching for the answer to every question. University classes are specially demanding when it comes to memorization because usually there are less evaluations per class and each evaluation contains a very large amount of materials that need to be studied. Moreover, the time available to study the materials tends to be somewhat shorter than it should, due to projects, weekly assignments and evaluation date collisions, which suggests that students should take advantage of every fragment of free time they have during the semester to keep class materials up to date in their mind.

Nowadays we can see some attempts to implement a blended learning environment in the context of university classes [2, 3, 4, 5, 6]. The goal is to keep the same physical classroom environment and combine it with a set of virtual tools that each student can use to learn and complete evaluated tasks at their own pace [2, 3, 4, 5, 7]. This approach tries to overcome the issue that arises when there is a number of fixed evaluation dates and students have to study a big chunk of information for each evaluation, which many may not be able to due to time constraints related to other university classes or even personal matters. With blended learning, each student can perform the tasks in the order they choose and when it better fits their schedule. This should result in a more sparse and continued study instead of just the usual bursts resulting from regular classes and evaluation methods, which brings benefits in the long term retention of the class materials [8, 9, 10, 11].
1.1. Problem

The issue with memorization is that it is time consuming and involves repetition, which makes it boring and unappealing [12]. Typically, when trying to memorize something, people tend to dedicate relatively long periods of time where they try to cram as much information as they can, hoping that it will stay there for a while. However, studies have shown that shorter and spaced study sessions are much more effective at storing information in the long term [8, 9, 10, 11]. But due to the repetitive and boring nature of these study sessions and also the lack of time to dedicate to them, it is hard to keep a study schedule that will let us achieve our goals.

In theory, physical classes like the ones we have at a university should proportionate the optimal continued and spaced study sessions that would enable students to learn effectively. However, because classes are always oriented towards collective learning rather than individual learning, not every student will be able to take full advantage of them. There are many variables at play, like class noise and the teachers' way of expressing themselves, and ultimately everyone has their own personal way of learning things and an optimal pace at which they can learn. Plus, in a physical class, a single distraction can make you miss something of big importance. If, instead, every student had, besides the physical classes, a personalized way to learn where they could study without missing content and they could do this at their own pace whenever they felt comfortable (i.e. motivated, focused and not tired), it should be expected that they would learn more and would overall feel more motivated towards classes as they would always be up to date with the materials. The current reality is that, even though students should have a continuous learning process and review class materials on their own time after each class to make sure they were always up to date, this just does not happen because of lack of time and motivation.
1.2. Approach

Microlearning applications have been extensively used to try to overcome memorization issues [8, 13, 14, 15, 16, 17, 18, 19, 12, 20] [21, 22, 23]. The core idea is that study sessions are spread throughout the day and have a short duration. This way people can take advantage of fragments of free time to study instead of having to schedule and dedicate a larger amount of time each day. This brings plenty of advantages, some related to user motivation and time efficiency [15, 19, 12, 18]. Microlearning is also versatile as it can be used with different learning methods like quizzes, flashcards or any sort of minigame.

Flashcards are one of the simplest ways to help people learn facts. They are easy to create and require very little interaction in order to be effective. However, traditional flashcards are hard to use because they require students to physically carry them around whenever they want to study and advanced study strategies are extremely hard to put in practice, especially as the number of flashcards grows. It is also easy to drop a few flashcards and mess the whole structure up. Plus, like every physical object, flashcards can be lost and damaged.

Thankfully, technological advancements have enabled the use of virtual flashcards in any sort of electronic device, overcoming all of these disadvantages [24]. One thing that is important to note is that we can now use software to manage virtual flashcards for us, allowing for advanced and effective study strategies. It is possible to keep track of the ideal time to study each flashcard using spaced repetition algorithms and even to exploit the capabilities of the device we are using to enable context aware learning.

This dissertation presents a microlearning application that uses virtual flashcards to improve students' learning performance in a blended learning environment. We will use an adaptive spaced repetition algorithm that will adjust the inter-study interval of each flashcard based on the overall memorization performance of each user, which is expected to accelerate the learning process by trying to achieve the optimal study interval for each user.

So far there does not seem to have been made a real significant effort to try to introduce a microlearning application into serious learning environments like universities, which motivates this dissertation.

The choice of using virtual flashcards instead of other learning methods was made based on its adequacy to microlearning scenarios and mobile contexts. The most basic version of virtual flashcards requires only a binary input from the user (right or wrong) while still being able to provide a rich variety of media to support learning (e.g. text, audio, images, gifs, videos, usage examples, mnemonics). This makes it a great choice when you take into account that many of the study sessions may take place in crowded and noisy areas. Another relevant factor is that there are already studies that combine virtual flashcards and microlearning and these can therefore be used as a starting point for this dissertation [8, 13, 17].
1.3. Objectives

There are four main objectives in this dissertation.

**Develop an adaptive mobile microlearning application and introduce it in a university class.**

The first main objective is to develop a mobile microlearning application that uses virtual flashcards and an adaptive spaced repetition algorithm that adjusts the inter-study interval of each flashcard based on the overall memorization performance of each user to be introduced in a university class.

**Understand whether there is a positive learning effect in using microlearning in a blended learning environment.**

The second main objective is to determine if the students that frequently used the application consistently had better results in the evaluations than the others who did not use the application or who used it less frequently.

**Determine if students adopted a more continued study strategy.**

The third main objective is to determine if the students felt motivated enough by the application to keep a regular study schedule throughout the semester, instead of quitting after a while or just using it close to the evaluation dates.

**Develop the application in such a way that it can be used in any domain.**

The fourth main objective is to make sure that the developed application can be used to learn materials from any domain, not being limited to learning the materials of a specific university class.
2. Related Work

In this chapter, we will analyze related work in the areas of learning and cognitive sciences, mobile and microlearning theory, and blended learning. We will also review a few relevant microlearning applications and then discuss their features in the Discussion chapter.

2.1. Learning and cognitive sciences

When using flashcards to study we are actually performing a cued recall. When we combine cued recall with a spaced repetition strategy, we are taking advantage of the testing and spacing effects. These effects have been studied in order to better understand how our memory works and to create a model that explains how memories are forgotten and how the strength of a memory is increased.

2.1.1. Testing effect

It has been observed in studies that study sessions that involve the subject being tested and then given the correct response increase the memory strength significantly more than study sessions where the information is presented right away and the subject is given some time to look at it. This effect also seems to not be tied to a specific domain, having been observed in a variety of contexts and domains like foreign vocabulary acquisition, word lists memorization, face-name associations and general facts [8, 25, 26, 12].

One important thing to note about the testing effect is that it happens even if there is no outward response given by the test subject. What this means is that the mental retrieval itself is enough for this effect to manifest [26]. This is great because it makes it possible to study in public places, where it is undesirable to produce any sort of vocal response, and it also means that a specific type of entity that the subject needs to communicate with in a specific way is not required, giving us a lot of freedom when implementing a strategy that takes advantage of this effect. The subject can simply look at a cue, try to recall the target mentally, check if the response obtained matches the target and finally provide feedback to any sort of entity in any sort of way that fits the target scenario. This could either be making a gesture to a camera, producing a sound to a microphone, pressing a button on a keyboard or touching a touch screen with the tip of a finger.

It has also been observed that by studying using cued recall tests such as flashcards, where we have tests of the form cue→target, the retention in the opposite direction (target→cue) is also enhanced. This even seems to enhance the free recall of cues and targets. This fact is quite beneficial too for the student because it means that by studying a single flashcard it is possible to get all these additional benefits with no additional effort, making it an efficient study method [25].
2.1.2. Spacing effect

The spacing effect can be observed when we compare the retention resulting from multiple shorter presentations of a chunk of information and the retention resulting from a single longer presentation of that same chunk of information. The multiple shorter presentations result in a superior retention than the alternative, which means that keeping a regular and non-intensive study schedule results in a more time efficient study than intensive study sessions closer to the evaluation dates [27, 11, 9, 10].

The way our memory works is that when you study a piece of information you will remember that piece of information for a certain time, usually referred to as the retention interval. The retention interval is usually at first quite short and further presentations of that piece of information will increase the retention interval further and further. However, as the retention interval increases, multiple presentations close to one another start to show diminishing returns and the inter-study interval should therefore be increased as well. This means that as the strength of a memory increases, the frequency of study of the pertaining piece of information can be decreased, freeing up time to learn other pieces of information that the student is less familiar with [8, 27].

It has been suggested that this effect could occur because the spacing of the presentations gives our mind some time to forget the weaker parts of a memory, which become more apparent during further presentations and allow us to better correct meanings, clarify ambiguities, correct misconceptions, and identify areas of weakness that need special attention [11, 9].

There is also another theory that says that when we study a piece of information we inherently absorb information from the context where we are studying and that information becomes attached to the information we are trying to memorize. By further spacing the presentations we increase the likelihood of bigger variations in the context we are at and more unique information from that context becomes attached to that memory as well. This attached information can then act like a sort of alternative entry point for that memory, making it easier to access the memory, in the same sort of way that a mnemonic helps us remember things [9].
2.1.3. Forgetting curve

The way our mind forgets things can be described through a forgetting curve [8, 28]. By looking at forgetting curves it is possible to determine that our minds forget things according to an inverse exponential curve. When there is a spaced presentation of a piece of information the strength of the pertaining memory increases and that memory's forgetting curve becomes less steep, which means that the process of forgetting for that memory becomes more gradual and takes longer, as can be seen in Figure 1.

This once again shows us that as the strength of a memory increases the inter-study interval should also be increased in order to compensate for the diminished benefits of further closer presentations. Presentations should only occur when the recall chance of that memory decreases to a certain value in order to obtain the benefits of the spacing effect discussed in the previous chapter. The recall chance can, thankfully, be predicted using the plotted forgetting curve and it is therefore possible to determine the approximate optimal time to schedule the next presentation.

Figure 1 – Forgetting curve resulting from adaptive spaced repetition learning [8].
2.2. Mobile and Microlearning theory

Current widespread learning systems tend to impose time and organizational constraints that do not fit the way people live their lives in modern days [8, 16]. With interactive devices and distractions absolutely everywhere it is hard for people to find time during the day to solely devote to learning. Not only this, but it also requires a big deal of self-motivation to stick to a learning schedule when you know that there is a near infinite amount of more fun activities you could be doing instead. So, researchers are now realizing that instead of focusing on ways to convince people to dedicate a certain amount of time to learning and keeping them motivated, like turning learning activities in game-like activities, a smarter and easier approach might be to try to help people take advantage of the really small time periods where they are not focusing on anything else. Even though it is hard to dedicate a longer period of time to studying, like half an hour or even an hour, there is a huge number of small pauses throughout the day that we could exploit and turn into learning sessions. Actually, these could even be called microlearning sessions since their duration can be measured in merely a few minutes or even seconds. The really short duration, however, does not mean that we will not be able to learn a decent amount because, due to the high frequency of these small pauses, they can quickly add up to a decent chunk of your daily time [29, 12, 30, 31, 20, 23, 15].

Nowadays, people carry mobile smart devices like smartphones, tablets and laptops everywhere they go. It is actually hard to meet someone who does not carry one of these. This means that microlearning activities can be easily introduced into people’s daily lives since they will not need to purchase or otherwise acquire any new equipment to get started. Installing an app or accessing a website is all that is needed to instantly start learning on these devices. In every small period of time where people are not focusing on anything, like when waiting for a bus, riding on the metro or even waiting for the food to cook, people can just pull out their smartphones and perform learning activities.

This is basically the foundation of mobile microlearning: taking advantage of every small period of free time throughout the day to learn, no matter where you are [29, 12, 30, 31, 20, 23, 15].

It is possible to tell already that learning apps are moving towards the concept of mobile microlearning but are not quite there yet. Many people are starting to adopt these apps because they realize the benefits they can bring them but quickly stop using them as the initial motivation goes away. The most obvious reason for this to happen is that these apps still focus more on long term retention and not enough on short term retention. This leads to poor initial results and a very slow start. Because of this, people quickly realize that these apps are not giving them the rewards they were expecting and either give up immediately or try something else. Some researchers tried to develop applications that focus more on the short term retention and test subjects gave positive feedback as they felt that, even though they were learning many new contents at a fast pace, if they failed to remember a content, the app would just quickly reinforce the training of that content so they could quickly get it back [8].
An interesting fact about mobile microlearning is that, despite its apparent advantages, apps that try to implement this concept are still just targeting these individual users that try them out for themselves. There has not been a real significant effort to try to put it in practice into serious learning environments like schools. Doing so would allow to improve or even develop new and better algorithms and strategies by testing them on students who could really benefit from using these apps.

2.3. Blended learning

The introduction of a learning app like the one described in this dissertation in a classroom environment like the ones found in a university would constitute a form of what is known as blended learning (or hybrid learning).

Blended learning is the result of mixing a physical learning environment with a set of virtual tools or even a virtual environment meant to assist the learning of the contents covered in the physical environment [2, 3, 4, 5, 7].

The benefits that blended learning brings to the table is that it allows for students to attend the same physical classes they are already used to, where they are thought class contents by the teacher, with whom they can interact at any time to clear up misconceptions, misunderstandings and ambiguities, while also allowing them to progress at their own pace on their free time using the virtual tools provided. These tools can be virtually anything: apps to assist the learning of the contents via games, quizzes or flashcards, a forum where the students can communicate more conveniently both with each other and the teachers at times where they would not be able to, or even a platform where students can submit projects to be evaluated as they complete a set of given tasks in the order and at the pace they desire and better fit their personal skills and schedule. This basically provides a more personalized way for students to learn, study and be evaluated that could not be achieved using just the traditional methods that are used today [2, 3, 4, 5, 7].

Teachers responsible for blended courses reported that the students were able to produce higher quality papers and projects, achieved better results in exams and understood the course materials to the point of being able to have more meaningful discussions on those materials. It was also reported that both the teachers and the students felt a higher level of satisfaction due to the increased student engagement in learning, which resulted from the enhanced flexibility of teaching and the enhanced interaction with the students [2, 3, 4, 5, 6].

The aim of this dissertation is to create a blended learning environment by introducing a mobile microlearning application that allows the teacher to create flashcard decks that the students can use to study whenever and wherever they want. The application will also adjust the inter-study intervals for each student, providing an extra bit of personalized study. The students in this class will have weekly quizzes that are part of their final grade and it is expected that this application will help them study continuously throughout the semester for these quizzes in order to achieve higher grades.
2.4. Microlearning applications

In this chapter, we will be reviewing some microlearning applications. The goal here is to see what has already been done, how and why, in order to try to make better choices regarding the application that is going to be developed for this dissertation.

2.4.1. MemReflex

Edge et al. developed the MemReflex application to try to solve problems often found in flashcard systems [8]. The first problem pointed out in their paper is that flashcard systems typically rely on intense initial memorization with subsequent tests delayed up to days later, not exploiting the short, sparse and mobile opportunities for microlearning throughout the day. They also think that these systems do not support learners who need the motivation that comes from successful study sessions. To solve these problems, the MemReflex application provides a system of adaptive flashcards that gives fast-feedback by retesting new items in quick succession and dynamically schedules future tests according to a model of the learner's memory. After development, they evaluated the application across three user studies, showing that it is effective for both audio and text modalities even while walking and distracted, and that it enhanced the learner's accuracy, confidence and perceptions of control and success. The algorithm used in MemReflex is based on the exponential intervals of the Pimsleur language system because of the uniform emphasis they place on short term-overlearning and long-term retention [32]. They also decided to incorporate Leitner-style binary feedback from the user after a cue has been presented to indicate whether the target was recalled correctly or not [33]. They defend this by saying that such binary feedback is more suitable for lightweight mobile input than the more numerous choices and the text input used by other applications. In the MemReflex application, the inter-study intervals are adaptively manipulated according to the learner performance, attempting to converge on a goal recall rate at all stages of retention for any given item. Finally, they try to identify opportune moments to introduce new items when there are no other items due
for review. The strategy they decided to use is to only introduce new items when there are no items with a recall chance lower than 90% according to their model of the learner's memory.

2.4.2. MicroMandarin

Figure 3 – MicroMandarin's contextual microlearning [13].

MicroMandarin is a contextual microlearning application developed by Edge et al. that leverages the location-based service Foursquare to automatically provide contextually relevant study content to help in second-language acquisition [13]. The authors performed a four weeks study with 23 users where they compared the results on Mandarin Chinese learning of this application's contextual system to a system based on word frequency. According to their study's results, study sessions with the contextual version lasted half as long but occurred in twice as many places as sessions with the frequency version, suggesting a complementary relationship between the two approaches. The rationale behind this application is that the microlearning of language in its context of use would encourage the use of that language in meaningful interactions with native speakers, providing immediately valuable learning material. By using the location-based social networking service Foursquare, the application can find the venue closest to the user’s mobile GPS coordinates and automatically suggest the study of vocabulary related with that context. This application uses flashcards to proportionate its microlearning aspect. MicroMandarin's contextual microlearning is based on three findings of Cognitive Psychology: encoding specificity, spaced repetition and situated cognition. Encoding specificity means that recall is best when the contexts of learning and retrieval share perceptual cues, hence learning vocabulary in its context of use should help us to recall that vocabulary when and where you need to use it. Spaced repetition means that learning is best when content is presented over time rather than in quick succession, which means that, by learning in short bursts in real-world locations, repetitions will naturally be spaced over both time and the places where we need it the most. Situated cognition supports the notion that knowledge cannot be fully abstracted away from the activities, contexts and cultures in which it is developed, so language can be learned implicitly when using it context as well as through explicit study. The MicroMandarin application uses an extension of the Leitner
algorithm adapted to multiple contexts [33]. This algorithm uses binary feedback self-reported by the user to determine if the word on the back of the flashcard was correctly recalled. For each flashcard, there is a correct count that is incremented on correct recall or reset to zero on failure. The front and the back of the flashcard are alternated as the correct count increases and the flashcard is considered learned when the count reaches four. For each context, there is a set of 10 unlearned flashcards and flashcards with a correct count of N are shown every Nth session so that better known cards are spaced further apart. After every 10 flashcards, the user is presented with quantitative feedback about progress made in that microlearning session. Whenever one of those 10 flashcards is learned, it is replaced with the next available flashcard for the current context.

2.4.3. Tip Tap Tones

Tip Tap Tones is a mobile microtraining game developed by Edge et al. with the goal of breaking down the challenge of mastering a foreign sound system into minute-long episodes of "microtraining" delivered through mobile gaming [14]. It also attempts to overcome the issue that arise when the learner's brain must be retrained to identify sounds not present in his or her native language. This issue being that it requires regular practice but many learners struggle to find the time and motivation. Tip Tap Tones has the purpose of helping learners acquire the tonal sound system of Mandarin Chinese and the authors performed a three weeks study with 12 users to evaluate its effectiveness. This study showed that an average of 71 minutes of gameplay improved tone identification by around 25%, even if the sounds tested had not been used to train tone perception. It also showed that mobile microtraining is an efficient, effective and enjoyable way to master the sounds of Mandarin Chinese, with applications to other languages and domains. Tip Tap Tones drew inspiration from the rhythmic tapping of Tap Tap Revolution and the steadily increasing challenge of Tetris, both of which use time pressure and repeated actions to facilitate the kind of concentrated attention that leads to skill development. This game employs a "press your luck" game mechanic: accurate responses are required to complete a level, but more points are awarded for fast level completion. The user can select different Mandarin sound playback speeds and must advance through 3
levels, each one getting harder by increasing the number of possible answers, by correctly answering a certain number of tones in each level. Learning is facilitated by replaying sounds using the repeat button and by feedback, with a correct response followed by a green flash of the screen and an incorrect response followed by a red flash and “punishing” tactile vibration. Tip Tap Tones incorporates adaptation by varying the game experience according to performance. Learners who consistently give successive incorrect answers will remain in the screens of Level 1, which provide fewer options and thus support initial tone acquisition. Learners who reach Level 2 must demonstrate an additional awareness of the differences between syllables and their spellings in pinyin. Completing Level 3 requires even finer discrimination of sounds while maintaining both accuracy and speed. The decision to limit the duration of each game to a short time period was jointly motivated by the mobile context of use and the natural structure of the minimal-difference identification task. In early experimentation, a game duration of 1 minute was found to provide novices with sufficient time to complete at least one game screen successfully (unlike a 30-second version), while allowing sustained concentration from more advanced learners without noticeable fatigue (unlike a 2-minute version).

2.4.4. Gassler et al.

Gassler et al. seem to be one of the first researchers to have explored the concept of a microlearning application in the paper "Integrated Micro Learning - An outline of the basic method and first results" [15].

Figure 5 – Gassler et al.’s learning screensaver [15].
They started by realizing that, even though one of the most important requirements for successful learning experiences is learning activity on a regular basis, learners often got stuck while using e-learning systems because they could not motivate themselves to use those systems for learning. In contrast, the same learners made use of their computers, cell phones and similar tools without much effort. Aiming at these motivational factors, they developed a method for technology enhanced learning. Their approach supports repetitive learning through embedding the learning process into the daily routine by making use of communication devices. Through this method they expected a new learning space to emerge and become available for lifelong learning. The concept of “Integrated Micro Learning” (IML) introduced in this paper had the goals of supporting individual learning (and not institutional learning as had been done to date) and solving an essential problem of Corporate Learning which is the absence of the employee from work during the learning process. With the method presented, e-learning is integrated in everyday life (even at work) and thereby represents a technologic support for lifelong learning. From the view of the learner the method solves typical motivational problems and enables a quick, interactive way of learning with high probability for success integrated into everyday life. The first prototypical realization of IML was carried out on the PC using the screensaver as the contact point. They developed a special screensaver to act as user interface for the interactive learning program that has to be performed to unlock the computer. For the first prototype, phrases in German and English, French, Spanish, Italian and Polish were compiled as learning content to establish a simple test surrounding to refresh language skills. Their solution used a study method similar to the first two applications discussed in this chapter: it showed a sentence in a foreign language, the user has to figure out the translation, press a button to see the solution and then provide a binary feedback to tell the system if he figured out the correct translation or not. Sentences evaluated as right were not repeated as often as wrongly translated ones. The algorithm used ran accordingly to common flashcard index systems. After the evaluation, the screensaver is closed automatically and the PC is accessible in the habitual way. In this version, the user could also cancel the evaluation at any time by pressing a button. The first results of this first prototype revealed a surprisingly high acceptance rate of the created learning time frames: the user group accepted the learning offer voluntarily in about 75% of the cases, which means there was an abort quota of only 25%. Another interesting experimental value was the frequency of learning: during the last weeks of the study, an average of 15 learning units per learner and day were performed. Assuming 200 workdays per year even with this minimal version one gets 3000 learning activities per user and year. Many users also requested to be able to learn longer than the system suggests voluntarily, which also showed how well the system was accepted. Compared to former experiences with e-learning systems, IML seemed to cause a completely different effect. The quota of acceptance was as high as otherwise the dropout rates, which often represent more than 75% of the users. In contrast to the frequently reported e-learning frustration, the initial users wished to be able to increase the suggested workload easily “on demand”.

25
Chen et al. made a study in which they presented a personalized mobile English vocabulary learning system based on Item Response Theory and learning memory cycle [16]. This system recommends appropriate English vocabulary for the user to learn according to their individual vocabulary ability and memory cycle. The goal of this study was to develop a modern assisted-learning tool that supported effective English learning. The authors consider the development of this kind of tool to be a critical issue in the English language education field, due to the high popularity of learning English in non-English speaking countries, which requires the memorization and practice of a large number of vocabulary words and numerous grammatical structures. Many previous studies had attempted to improve the efficiency and effectiveness of English vocabulary learning and, with the accelerated growth in wireless and mobile technologies, mobile learning using mobile devices had gradually become considered effective because it inherits all the advantages of e-learning and overcomes limitations of learning time and space that limit other web-based learning systems. The proposed learning system was successfully implemented on personal digital assistant (PDA) for personalized English vocabulary learning. This system tunes the review cycle of each individual learner based on that learner's vocabulary ability, the difficulty parameters of learned vocabulary words and the test outcomes for learned vocabulary. The way this system tests the learner's knowledge on vocabulary is by using multiple choice, where the user is presented with a cue word in their native language and 4 possible and similar answers and they must try to pick the correct translation. The vocabulary difficulty is measured based on its use frequency, length and pronunciation. The review process used is a modification of Ho's scheme for English vocabulary learning based on individual memory retention ability,
because each person has different retention abilities, even when learners are learning the same material, hence review cycles should be tailored to each person. Ho's scheme extends the memory cycle for individual learners progressively while the learner can give correct responses for tested vocabulary words. In turn, if the learner fails to give correct answers, the memory cycle will be progressively shortened until the learner can answer correctly. The modification made to Ho's scheme was meant to allow the adjustment of the review process based on learner's vocabulary ability, the difficulty of vocabulary words and test results. Finally, the memory cycle adjustment is made based on the Fibonacci sequence. The system was used by 15 university students during a five-week period in order to evaluate its learning effectiveness. The evaluation consisted of three evaluation procedures: a pre-test, a post-test and a questionnaire. The first two procedures were meant to evaluate the learning performance of the learners and the questionnaire was meant to get the learner's feedback on the system. The experimental results indicated that the proposed system could indeed promote the learning performances and the interest of learners due to its effective and flexible learning method. Most learners believed that the review strategy was very helpful when learning English vocabulary and that this system proportioned a seamless ubiquitous learning environment for English learning without constraints of time or place.

2.4.6. µLearn

Figure 7 – µLearn mobile microlearning application [17].
Brandt et al. developed the µLearn mobile application, which addresses the issues of fast creating and seamless consuming of flashcards for ubiquitous microlearning, to try to overcome function constraints that hinder context-aware and frictionless microlearning on many widespread modern flashcard mobile microlearning applications [17]. The authors start their paper by stating that nowadays people use multiple personal computing devices like desktop computers, laptops, smartphones and tablets. These devices are used in different situations, for different reasons and in different periods. They think that the combination of all kind of devices is a solid basis for improving the learning process of every user. The Web has become the main source for informal life-long learning, with platforms like Wikipedia, blogs, Q&A sites and online dictionaries being used to fill in the gaps in personal knowledge. The Web platforms, however, often contain too much information in relation to the users’ needs. Scraping and saving the relevant parts of a website helps the user by collecting only the information needed to understand a certain topic. With this in mind, µLearn allows the user to collect pieces of information from web pages related to a topic. A browser extension developed alongside the mobile application enables the user to semi-automatically collage and reuse this content, which is then fed to the mobile application as digital flashcards. In this browser extension, named Multi-Lookup, the user can specify sources for web search and join them into groups for specific search interests. The goal with Multi-Lookup is to avoid the time consumption and mental effort that comes from the “research missions” performed when a user searches for and visits multiple web pages in order to satisfy complex and highly-personalized search needs, usually within fields with multiple heterogeneous web sources. For example, searching for meaning of some phrase in German for non-native speaker involves looking up in multiple translation, dictionary, thesaurus and example web sources. According to Donato et al., these “research missions” account for about 25% of the search volume on Yahoo! search. After identifying several result pages as potential answers to the search query using Multi-Lookup, the user tries to identify a comprehensible answer. The user selects the parts of these different web pages that seem the most relevant in the current context and these are appended to the learning object. This tool greatly reduces the time needed to collect chunks of web pages since it reduces the selection operation to a single click on the highlighted element. The scraped content is then filtered, cleaned and automatically synchronized with the microlearning mobile application. It is still possible to create the learning objects on the mobile application without requiring the use of the Multi-Lookup browser extension. The learning objects in µLearn can be a question or a fact and, when studying, the user can, at any time, rate the current learning object on a scale based on the difficulty felt. There is also an option to flag the learning object as “needing enhancements” when the user feels that there is not enough information being provided. This application uses the “Super Memo 2” spaced repetition algorithm to determine the inter-study interval. The two-way synchronization between the browser extension and the mobile application is made via a web server also developed alongside these two. The application was evaluated to determine whether it is easy to understand and intuitive to use. The focus of the user evaluation was the usability of the application and not the learning effect. A questionnaire consisting of questions regarding the design and the usability of the application and some information about the user was handed to six test users, both experienced and non-
Every user had a different device, so the application was tested on six different devices with also different OS versions. The questionnaire was collected after a week, and showed that most of the users used the application 10 to 20 times in the given time frame, all users understood the application and even the meaning of most icons was clear. The results also showed that the design and the structure of the app are intuitive and that the test users had no problems using the application. The conclusions taken by the authors of this paper state that in future versions of this application there should be multiple additional spaced repetition strategies that the user could select from, learning notifications to inform the user of what he should learn on each day and the ability to share learning statistics.

2.4.7. LingoSnacks

![Figure 8 – LingoSnacks mobile microlearning application [18].](image)

LingoSnacks is a game-based mobile microlearning application developed by Erradi et al. to provide an interactive learning assist of Arabic vocabulary [18]. LingoSnacks allows short sessions of microlearning during any brief opportunities that learners have throughout the day and also addresses the problem of dry content and the boring traditional way of learning Arabic because it was designed and implemented as a solution to provide novel and enjoyable ways to learn Arabic using multi-touch interactive games. Additionally, the platform facilitates authoring, sharing and exchange of learning content. This application assists the creation of contents using online sources in order to further enrich them. It is possible to add images and audio to the created contents. LingoSnacks allows users to create and share learning contents organized into packages that are pushed to a web server. After sharing packages, creators can maintain them in order to keep them up-to-date for everyone studying them. Learners can explore available learning packages, download desired ones and engage in a range of interactive learning activities and games based on the content of the selected package. Learners can also rate downloaded packages. The application also
keeps track of the learner’s performance. Instead of flashcards, LingoSnacks offers a variety of game-like activities to practice spelling and vocabulary acquisition. LingoSnacks key nonfunctional requirements is to provide a system that is not only easy and intuitive to use but also offers an entertaining and interactive user experience to foster active learning and encourage exploration. After the implementation, the system was evaluated by 12 young users and 10 adults, 7 of which were Arabic teachers. The adults were asked to prepare some learning content. The results of the evaluation showed that the application is effective on the enhancement of acquired vocabulary and that learners were able to better recognize and recall the learned vocabulary after using the application. The application was also successful in keeping the learners motivated thanks to the game-like activities that attract the learners with multimedia content that can enjoy. Most of teachers that tested the application were satisfied with how easy it was to prepare new contents.
2.5. Discussion

In this chapter, we will go over the main features of the reviewed applications, pointing out some advantages and disadvantages and taking them into account as we discuss what can be brought into our application.

Table 1 – Feature comparison between reviewed applications.

<table>
<thead>
<tr>
<th>Application</th>
<th>Study approach</th>
<th>Contextual learning</th>
<th>Mobile learning</th>
<th>Time constraints</th>
<th>Spacing strategy</th>
<th>Content type</th>
<th>Assessment of content difficulty</th>
<th>ISI adjustment strategy</th>
<th>Content creation assist</th>
<th>Server connection</th>
<th>Domain specific</th>
<th>Adaptive to individual learning ability</th>
</tr>
</thead>
<tbody>
<tr>
<td>MemReflex [8]</td>
<td>flashcards with binary feedback</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>time based</td>
<td>text and audio</td>
<td>yes, based on incorrect responses</td>
<td>exponential intervals based on Pimsleur algorithm [32]</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>MicroMandarin [13]</td>
<td>flashcards with binary feedback</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>session based</td>
<td>text only</td>
<td>no</td>
<td>extension of Leitner algorithm for multiple contexts [33]</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Tip Tap Tones [14]</td>
<td>game with multiple possible answers</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>unspecified</td>
<td>text and audio</td>
<td>no</td>
<td>unspecified</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>yes, through increasing level difficulty</td>
</tr>
<tr>
<td>Gassler et al. [15]</td>
<td>flashcard-like with binary feedback</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>unspecified</td>
<td>text only</td>
<td>no</td>
<td>unspecified</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Chen et al. [16]</td>
<td>flashcard-like with multiple possible answers</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>time based</td>
<td>text only</td>
<td>yes, based on vocabulary characteristics and learner’s performance</td>
<td>modified Ho’s scheme with Fibonacci increments and decrements</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>μLearn [17]</td>
<td>flashcards with feedback based on a perceived-difficulty scale</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>time based</td>
<td>text, images and HTML excerpts</td>
<td>yes, based on user feedback</td>
<td>Super Memo 2 algorithm</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>LingoSnacks [18]</td>
<td>game-like activities</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>unspecified</td>
<td>text, images and audio</td>
<td>no</td>
<td>unspecified</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>no, but it is a planned addition</td>
</tr>
</tbody>
</table>
2.5.1. Study approach

Starting out with the study approach, it is possible to see some variety between the applications reviewed. The vast majority of these applications uses flashcards and the approach with flashcards and binary feedback seems to be the most frequent one. In the application developed in this dissertation we will be taking on this approach because it allows for easier and faster content creation, while also enables the application to be used for a probably much wider range of domains than the game-based alternatives. Besides this, binary feedback allows the application to be used easily in environments with frequent distractions where the learner is only paying partial attention to the application, increasing the range of possible scenarios where study sessions can take place.

2.5.2. Contextual learning

Contextual learning was only explored in one of the applications reviewed, MicroMandarin [13]. And, even though contextual learning seems like an interesting and promising direction to take, it probably does not make sense for many or even most domains. In particular, it does not make much sense in the context of this dissertation. Therefore, contextual learning will not be a part of the application developed here. However, it could be interesting as future work to explore how contextual learning could be used in a university scenario where you could take into account the position of the student inside the university to try to predict which materials could be more useful to be studied right away. Another addition could be to analyze the student's schedule and focus on materials that are related to classes that the student will be having soon.

2.5.3. Mobile learning

Mobile learning is practically a must when it comes to microlearning, as can be observed on Table 1. Only one of the applications reviewed does not focus on mobile learning. We want study sessions to be available everywhere and at all times, so mobile learning is vital in our application.

2.5.4. Time constraints

Only one of the reviewed applications uses time constraints: Tip Tap Tones. This application leverages the game levels to adapt the difficulty to the learner's abilities, because higher levels are harder and if the learner cannot get past the first levels he will not have to deal with that increased difficulty to which he is not ready for, practicing and improving his skills at his current level until he can progress further. By using time constraints, it is possible to limit the duration of sessions even when the learner cannot reach the higher levels, preventing frustration but still allowing for a decent study amount. Another factor in play here is that this application tries to promote quick responses by rewarding the learner with more points. If the learner did not have this hard time limit, even apt learners could easily get worn down and maybe even frustrated by having to perform quick responses for an extended duration. In the context of flashcards rather than games, it could make sense to have a time limit set on the study sessions in order to give the learner
more control over the duration of the study sessions, especially if the learner could set that duration himself. However, it seems more logical to have study sessions with an unlimited duration and just let the learner stop whenever he wants to. This way there are less interruptions for longer study sessions, which can take up all of the free time that the learner has available in that moment, while also allowing for very short study sessions and not making the learner feel that he studied too little because he did not study for the complete duration of the session, which could possibly be a little discouraging. Another alternative would be to, instead of limiting session duration, time how long the learner takes to study a certain number of flashcards and maybe even take each flashcard study duration into account to calculate the estimated difficulty of that flashcard. Even though this seems to make sense, it does not really fit into the concept of microlearning, where the learner can be distracted at any time for an unpredictable amount of time, which would render the measured times useless in any case as we have no way of knowing if the learner was ever distracted during the study session or if he just took longer to recall.

2.5.5. Spacing strategy

From the spacing strategies used by these applications, the more logical one to use seems to be the time based one. The session based strategy just seems like a lower resolution and less accurate version of the time-based strategy. We know the user needs to study a certain flashcard in two sessions but we cannot know if those two sessions will be minutes, days or even weeks apart. The user could be studying the flashcards either too soon or too late, which would not give good results in any case and just be a waste of time. The best approach is to determine the exact optimal time for the user to study the flashcard and only notify and allow the user to study it once that time is reached. This way the user can focus on studying other possibly new or more difficult flashcards in the meanwhile but still make sure that he does not forget that flashcard by making him study at the ideal time to refresh his memory of it.

2.5.6. Content type

Surprisingly, quite a few of these applications allow for text only content to be created and studied. It seems rather obvious that the materials should be as rich as possible as long as it is not overdone and just makes the materials confusing or distracting. In many contexts and domains, text, audio, images, animated gifs and even videos seem to be a great resource to have when studying. Audio can be great to learn word pronunciation while images, animated gifs and videos can be very useful to learn certain gestures and movements. All of these can also be used to create mental associations between memories to the point where hearing a word or seeing a gesture being performed reminds the learner of its meaning. At the same time, these can also be helpful when studying in situations that involve movement, where the learner can still hear the cues by wearing earphones and glance at the image on the screen, allowing him to recall the target without having to read the cue, which requires a little more focus and effort. In our application, an effort will be made to allow this kind of resources to be embedded in the flashcards in order to enjoy these benefits when possible.
2.5.7. Assessment of content difficulty

The assessment of content difficulty can be useful to better adjust the inter-study interval of each flashcard so that difficult cards are reviewed more often than easier cards, which can probably be memorized faster and require less maintenance. In fact, some of the applications reviewed take this factor into account. In our application, a few strategies could be used: the estimated difficulty for each card could be set by the deck owner (e.g. teacher) or estimated using the average performance of all learners, which would probably yield more accurate values. An additional alternative would be a combination of the two, where the difficulty set by the deck owner would provide a starting point that would progressively be adjusted using the learners’ performance. This would very likely lead to even better results but, for the sake of simplicity, this dissertation will probably not be using any sort of difficulty assessment, focusing solely on individual performance in order to adjust the inter-study intervals of all cards.

2.5.8. ISI adjustment strategy

Every application seems to use a different algorithm to perform ISI adjustments. Based on the findings on learning and cognitive sciences that we collected on the Related Work chapter, the exponential intervals based on the Pimsleur algorithm seem to be a better fit for our application [32]. They account for the forgetting curve and help maintain a certain recall chance at all times. If the learner fails to recall the back of a flashcard, that flashcard's progress will be reset, allowing for the learner to catch up on its content.

2.5.9. Content creation assist

The implementation of a content creation assist is undoubtedly a great addition to a mobile microlearning application, as could be concluded from the work of Brandt et al. and Erradi et al. [17, 18]. However, such an assist would require a large amount of additional effort and goes beyond the scope of this dissertation. Therefore, our application will likely not possess such assist.

2.5.10. Server connection

Nearly all of the reviewed applications feature a server connection in order to synchronize learning contents and performance statistics. This is a feature that makes complete sense to have in the context of this dissertation, as it allows the students to be updated with new learning contents at any time and allows the teacher to follow and assess the learning performance of the students in real time as well. This server connection can also be used so that the students can provide feedback on the learning contents (e.g. notify the teacher that there are errors or ambiguities, lack of information and maybe even ask questions about the content that the teacher can then answer).
2.5.11. Domain specific

From all of the reviewed applications, only µLearn was developed in a way that enables it to be used to learn contents from any domain [17]. One of the development goals of our application is that it should be domain-free so that it can be used to learn contents from any class of any study branch and even contents that are not related to university classes in any way.

2.5.12. Adaptive to individual learning ability

Nearly half of the applications reviewed include a way to personalize the learning process based on the estimated individual learning ability of each learner. Our solution aims to solve the problem of lack of personalized learning in the context of university classes, so it makes sense to include such a feature. The individual learning ability can be estimated by analyzing the percentage of correct responses that each user produces.
3. ADAM

In this chapter, we will describe our solution, ADAM, going over our approach, the architecture behind it, the adaptive spaced repetition algorithm used to calculate the inter-study intervals, and the web application that teachers and students interact with, presenting its features and UI.

3.1. Approach

Our solution was to develop a mobile microlearning application and introduce it in a university class in order to create a blended learning environment, in which students could use the application to learn on their own and at their own pace on their free time outside physical classes.

The materials covered in the class were converted to flashcard decks that were made available to the students, who were then given the option to use the application to study these decks whenever they wanted. However, students could still choose not to use the application and instead study using the lecture slides provided by the teachers or by any other means they saw fit.

The application was developed for the web so that it could be used in any platform, mobile or not. This was to allow any student to access the application in a smartphone, tablet, laptop or desktop computer, widening the range of opportunities to perform learning activities.

The chosen study approach was to use flashcards with binary feedback combined with a time based inter-study interval adjustment strategy based on the Leitner system and the exponential intervals of the Pimsleur algorithm [33, 32]. These flashcards can contain text, images and animated gifs, and more types of content can easily be added in the future in order to enrich the learning experience and give more flexibility in the creation of contents.

The application was developed with the intent of being domain-free, meaning that it can be used to learn contents from any domain. Therefore, it is not limited to the contents of the university class where it was introduced, and can be used in other classes and even to learn numerous other subjects like foreign vocabulary and trivia.
3.2. Architecture

ADAM has a very simple architecture. We have a Node.js instance running in a cloud web server that serves the web application to the users as well as expose a REST API that performs all the back-end operations. This Node.js instance is connected to a MongoDB instance also running in a cloud server, with which it communicates whenever an operation needs to be performed.

We decided to use Node.js because it allows for fast development and produces clean and readable code, which can be easily understood, maintained and modified by people who might pick up the project in the future. When deployed, Node.js also provides good performance and scalability, allowing us to keep response times low even as the number of users on the platform increase. Additionally, Node.js is being increasingly adopted by web developers in current times and there are many open source libraries available on npm to perform all sort of tasks, which save us development time and allow us to focus more on developing the project itself.

MongoDB is a NoSQL database that stores documents in JSON format, the same format that Node.js uses internally. This allows for easy integration between the two, which also saves time and makes the code simpler. We chose MongoDB instead of similar alternatives because it is the most popular one.

The back-end Node.js code is built on top of the Express framework and uses many open-source and publicly available npm libraries to perform all its operations. Most notably, it uses Mongoose to define the database schemas and handle all the database operations (create, find, edit, delete), and Joi to perform the validation of the REST API endpoints’ parameters.
The web application served is the same for every connecting device and was developed in such a way that is practical and comfortable to use whether that device is a computer, a tablet or a smartphone. Further details and specifics about this web application will be discussed further in this chapter.

Besides serving the web application and performing the database operations needed, the web server also includes the adaptive spaced repetition algorithm used to calculate the inter-study intervals, which will also be discussed further in this chapter.
3.3. Adaptive spaced repetition algorithm

![Diagram of the Leitner system](image)

Figure 10 – In the Leitner system, correctly answered cards are advanced to the next, less frequent box, while incorrectly answered cards return to the first box [33].

The adaptive spaced repetition algorithm used to calculate the inter-study interval in ADAM is inspired by the algorithm developed for the MemReflex application by Edge et al. [8], and may even be considered a somewhat simplified version of that algorithm, although the equation used is not the same.

Both algorithms are meant to be used with flashcards with binary feedback, where the student is supposed to look at the front of a flashcard and try to remember what is on its back. If the student fails to do so, he sends a negative feedback to the application, and if he succeeds, he sends a positive feedback instead.

The equations of both algorithms output the interval of time in seconds that should pass before the student needs to study that flashcard again. The algorithms are based on the Leitner system (Figure 10), where each flashcard is put into the first box (or bucket) after the first study and moves to the next box on positive feedback or to the first box again on negative feedback [33]. In this system, boxes are numbered and flashcards on boxes with higher number are studied less frequently. The inter-study interval of each flashcard, therefore, depends mainly on the box the flashcard is in.

In MemReflex and in ADAM, the base value of each box’s inter-study interval is calculated with Pimsleur’s algorithm [32], using the equation:

$$ t = 5^n $$

In this equation, $t$ is the inter-study interval in seconds, and $n$ is the box number which ranges from 1 to infinity.

This equation, however, does not consider the individual performance of the students. Some students memorize things faster and others memorize things slower than the average. Therefore, both MemReflex and ADAM modified this equation in different ways to adapt the inter-study intervals for each individual student. They chose to adjust the exponent in MemReflex and we chose to adjust the base in ADAM.

Another difference from the two algorithms is that MemReflex keeps an interval modifier associated with each box for each student, where ADAM simplifies this and only keeps a single interval modifier for each student, which is shared among all boxes.
The way in which ADAM makes use of this modifier is very simple. Every student starts out with a modifier with a value of zero and the modified equation used to calculate the inter-study interval is the following:

\[ t = (5 + \frac{m}{1000})^n \]

In this equation, \( t \) is the inter-study interval in seconds, \( m \) is the student’s individual modifier, and \( n \) is the box number which ranges from 1 to infinity.

The modifier is divided by 1000 in the new equation to reduce the impact of changing it to acceptable levels. If it was divided by a smaller number, very tiny changes to the modifier would yield massive time differences, especially in boxes with a high \( n \) value, due to the nature of exponentiation.

When developing the algorithm, we established a target recall rate of 90%, the same used in MemReflex. For each student, we keep track of the number of positive feedbacks as well as the total number of feedbacks received. When a student sends feedback after studying a flashcard, we adjust those values accordingly and calculate the proportion between the two. This proportion is that student’s recall rate. If the recall rate is equal to or bigger than 90% and the feedback was positive, we increase the modifier by one. If the recall rate is lower than 90% and the feedback was negative, we decrease the modifier by one. In all the other cases, we do not touch the student’s modifier at all.

The logic behind these modifier adjustments is that, if a student has a recall rate equal to or higher than 90%, it means that the user is doing better than our target recall rate. Therefore, if he sends a positive feedback we should increase his modifier to space out his inter-study intervals further. This way he can spend more time learning new flashcards and less time reviewing flashcards that he probably recalls very well still. If he sends a negative feedback, he’s still over our target recall chance and therefore should not be penalized. If a student has a recall rate lower than 90% and sends a positive feedback it means that the student is moving towards the target recall chance and we should not increase nor decrease his modifier, as it seems to be at a good value for this student. If, however, this same student sends a negative feedback, we should decrease his modifier because the inter-study intervals seem to still be a bit higher than they should and his performance is suffering.

With these adjustments, the student should theoretically come closer to and stabilize near his optimal modifier value over time, which will allow him to recall 90% of the studied flashcards at any given time if he studies whenever the algorithm tells him to. It does not matter if the modifier has a very low value (negative even) or a very high value. The purpose of the modifier is simply to adapt the study frequency for each user and make sure that they are always at or near the target recall chance of 90%.
Table 2 – Examples of inter-study intervals for the first 10 boxes in the Leitner system calculated with ADAM’s adaptive spaced repetition algorithm using three different modifier values [33].

<table>
<thead>
<tr>
<th>Box</th>
<th>Modifier = -200</th>
<th>Modifier = 0</th>
<th>Modifier = 200</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4.8s</td>
<td>5s</td>
<td>5.2s</td>
</tr>
<tr>
<td>2</td>
<td>23s</td>
<td>25s</td>
<td>27s</td>
</tr>
<tr>
<td>3</td>
<td>1m 50.6s</td>
<td>2m 5s</td>
<td>2m 20.6s</td>
</tr>
<tr>
<td>4</td>
<td>8m 50.8s</td>
<td>10m 25s</td>
<td>12m 11.2s</td>
</tr>
<tr>
<td>5</td>
<td>42m 28s</td>
<td>52m 5s</td>
<td>1h 3m 22s</td>
</tr>
<tr>
<td>6</td>
<td>3h 23m 50.6s</td>
<td>4h 20m 25s</td>
<td>5h 29m 30.6s</td>
</tr>
<tr>
<td>7</td>
<td>16h 18m 26.8s</td>
<td>21h 42m 5s</td>
<td>1d 4h 33m 27.2s</td>
</tr>
<tr>
<td>8</td>
<td>3d 6h 16m 32.8s</td>
<td>4d 12h 30m 25s</td>
<td>6d 4h 29m 57.3s</td>
</tr>
<tr>
<td>9</td>
<td>15d 15h 43m 25.5s</td>
<td>22d 14h 32m 5s</td>
<td>32d 4h 11m 45.9s</td>
</tr>
<tr>
<td>10</td>
<td>75d 3h 28m 26.2s</td>
<td>113d 40m 25s</td>
<td>167d 7h 25m 10.6s</td>
</tr>
</tbody>
</table>

In Table 2 we can see examples of inter-study intervals for three students with three distinct modifier values. The modifier value of -200 belongs to a student that has a little trouble memorizing the flashcards with the base intervals from the Pimsleur algorithm [32]. The modifier has, therefore, decreased to a negative value in order to help him achieve the target 90% recall chance. As a consequence, he will have to study the same flashcards more frequently. If, however, over time he gets better at memorizing the flashcards, the modifier will slowly increase once again so that he will not have to study as frequently and can spend more time learning new flashcards instead, as is the case of the student with the modifier value of 200.

As we can see by looking the Table 2, different modifier values will have little impact at first but in more advanced boxes the differences in inter-study interval become much more visible. Not all students might be able to recall 90% of the flashcards for 6 days just by studying them 7 times. The algorithm, therefore, adapts to them and makes them study the flashcards in 3 days instead. The goal is to never let the students forget too much by constantly telling them to study when they really have to, and only when they really have to.
Edge et al. also rewrote the equation of the human forgetting curve model to represent memory strength as a fixed scaling of the inter-study interval [8, 28]. With this equation, we can calculate a student's estimated recall chance of a card at any moment. The equation is the following:

\[ E(\text{recall}) = 0.9^{\frac{t_e}{\tau}} \]

In this equation, \( E(\text{recall}) \) is the student's estimated recall chance of a card, \( t_e \) is the time elapsed since the last review of the card in seconds, and \( \tau \) is the inter-study interval in seconds. The base of the exponentiation is our target recall rate of 90%. Using a different target recall rate would require us to replace that value in this equation.

If the time elapsed is zero seconds, the estimated recall chance is 100%, which makes sense because the student saw the solution zero seconds ago. If the time elapsed is equal to the inter-study interval, the recall chance is exactly 90%, which is our target recall rate goal. If the time elapsed grows higher than the inter-study interval, the recall chance will lower beyond 90% following a reverse exponential curve, which goes according to the forgetting curve (Figure 1).

This estimated recall chance equation is useful because it allows us to show the student his estimated recall chance of a card when he is studying it. However, we chose to only show estimated recall chances equal to or higher than 50%, because we think that while high values motivate the student to try harder to recall the solution, low values might do the opposite and make him quit sooner and not try as hard.
3.4. Application

In this chapter, we will present the ADAM web application, talking a bit about its development and showing the available features.

3.4.1. Development

The ADAM web application was developed using the Vue.js framework and Google's Material Design Lite. It is a single page application, meaning that, as the user navigates the application, he never really leaves the page, resulting in very fast response times and fluid interaction with the application. As soon as the user clicks a button or link, the resulting action takes place instantly, there are never seconds long loading times. This is done by Vue.js via the HTML5 History Web API available in modern browsers and is completely transparent to the developer.

We also took the effort of implementing a Service Worker for the application, which lets us control the way modern browsers handle the cache and the web requests. With this Service Worker, we can cache the entire application in the browser and prevent the browser from requesting it again in future visits until a new version is released, instead serving it from the cache every time. This saves internet traffic (great for mobile users), increases the loading times of the application (from 600ms to 500ms) and frees up our web server from receiving those requests and serving those static files, allowing it to handle the API requests even faster.

Google's Material Design Lite makes sure that the application looks great in every device, providing a simple, clean, minimalistic and modern appearance to the application.

During the development process, we ended up creating two additional projects to help developing ADAM. The first one was maky, a promise-based task runner inspired by Gulp and compatible with Gulp in both directions (maky within Gulp and Gulp within maky) [34]. It helped overcome a few issues we were having with Gulp and simplify the creation of additional build plugins. One of those plugins was used to version the application and the Service Worker so that the clients would be able to tell when a new version had been released in order to download and cache it.

The second project was a flexbox library that allowed us to design great-looking and responsive layouts extremely easily and quickly [35]. The existing alternatives all had quite apparent limitations, such as not letting us choose whether a certain element was a row or a column without writing our own custom CSS, had long class names, hard to write and to remember, and lacked some features. So, we created our own flexbox library with simple and consistent class names and with all the features needed to use the complete flexbox grid without having to write any CSS regarding the layout in the application itself.

ADAM was first developed as a simple application for flashcard decks. However, it was later defined that it should be more similar to the Fénix system used in Instituto Superior Técnico and some other universities. It should contain the concepts of course and semester, so that decks could be better organized and
accessed by students. This also allows for a single instance of the application to be used by many teachers and students for an indefinite period of time. Teachers can create courses and semesters within those courses, every semester students join the corresponding semesters of their courses, and teachers can then close those semesters and create new ones for the new students that would join the course. This also opens up new possibilities like the cloning of semesters and decks, and the collaboration between teachers of a course in creating and editing decks and flashcards.

In this chapter, we will only present the reworked version of ADAM which is a superset of the previous version, with the concepts of course and semester added on top of the concepts of deck and flashcard.

3.4.2. Authentication

ADAM was developed with the goal in mind of introducing it in a university class at Instituto Superior Técnico. This university’s Fénix system allows us to register an application in it and gives us a unique API key that we can use in our application in order to use Fénix as a third-party authenticator. When a user wants to login in ADAM we redirect them to Fénix, where they login with their university credentials, authorize our application and are then redirected back to ADAM. After this, ADAM receives that user’s identity information from Fénix, removing the need to create new accounts within ADAM and storing and validating passwords, as all of that is delegated to Fénix.

![Login Screen](image)

Figure 11 – ADAM’s starting screen.
3.4.3. Reporting issues

ADAM allows any user to report issues, even if they are not logged in. These issues are sent to the server and stored in the database so that the developers can later review them. This feature was implemented because we were afraid that the application might have some hidden bugs and, since we were using the application in a real scenario with real people, we wanted users to be able to report them if they found any so that we could fix them. Luckily, no bugs were ever detected in the application so this feature never saw any real use.

In Figure 12, it is possible to see the “Issue report” page. It is just a basic form where the user can optionally fill out his student ID and name (these are auto-filled by default when the user is logged in but he can still choose to clear the fields) as well as describe the issue.
3.4.4. Navigation

ADAM has a left panel used for navigation within the application. The home page of the application is the “My courses” page. Here, the user can quickly access the courses in which he is assigned as a teacher or where he enrolled in a semester, as well as quickly access the enrolled semesters if that is the case.

The second navigation link takes him to the “All courses” page which is similar to the previously described page but shows every single course that exists within the application. In both of these pages there is a “Create course” button on the top, only visible to users assigned as teachers in the application.

At the bottom of the panel there is the link to the “Issue report” page already discussed and presented in Figure 12.

This panel also includes the first and last name of the user currently logged in and the logout button. Each user also has his own unique silly avatar just to give a more friendly, colorful and personal touch to the application. In mobile devices with smaller screens, the left panel is automatically collapsed to free up screen space and can be expanded by clicking/touching the menu button at the top left corner of the screen, as can be seen in Figure 14.

Figure 13 – “My courses” page in ADAM.
Figure 14 – Mobile version of the “My courses” page in ADAM. (a) Left panel collapsed. (b) Left panel expanded.
3.4.5. Courses

Users assigned as teachers in the application can create courses. In Figure 15 we can see the “Create course” page in ADAM. A course needs a name and, optionally, a description. When a teacher creates a course, he automatically gets assigned as an administrator of that course. Administrators of a course can edit it in order to change its name or description at any time. The “Edit course” page looks exactly the same as the “Create course” page, as can be seen on Figure 16.

Figure 15 – “Create course” page in ADAM.

Figure 16 – “Edit course” page in ADAM.
In Figure 17 we can see how the course page looks like from the point of view of one of the course’s administrators. Other users would not see any of the buttons as they do not have the permissions to perform any of those actions. In this page, we can read the course’s description, if it has one, the list of teachers, with administrators identified as such, and the list of semesters of this course, if there are any. If we click on a semester (in pink) we are sent to that semester’s page. Closed semesters are identified as such and semesters with a password have a padlock icon after the name.

![Figure 17 – “View course” page in ADAM.](image)

When a course administrator clicks on the “Manage teachers” button, he goes to the “Manage teachers” page of that course, where he can see a list of that course’s administrators and teachers, as well as a list of the users of the application, as can be seen in Figure 18. He can then change the role of any of these by clicking the corresponding button in front of the user’s name. He can even change his own role if he wants to, but he will receive an error if he tries to demote himself when there are no other course administrators, as a course needs at least one administrator at all times.

Users assigned as teachers cannot perform as many actions as administrators, like managing other teachers, editing the course, or creating and editing semesters. They are limited to creating, editing and deleting decks inside existing semesters, and creating, editing and deleting flashcards inside of any of the course’s decks. They can also consult the semester and deck leaderboards as well as the deck stats.
Figure 18 – “Manage teachers” page in ADAM.
### 3.4.6. Semesters

Creating and editing semesters in ADAM is similar to creating and editing courses. The pages are identical as well, as can be seen in Figure 19 and Figure 20. Semesters require a name, may have a password and may also be flagged as closed. If a semester is flagged as closed, students can no longer enroll in it, which is useful when reaching the end of a semester to prevent students to join it by accident. The password is also useful to control which students may enroll in the semester. The teacher can give the students the password via email, for instance, and students may then enroll the semester if they wish. Alternatively, a semester may have no password, in which case anyone will be able to enroll in it.

![Figure 19 – “Create semester” page in ADAM.](image1)

![Figure 20 – “Edit semester” page in ADAM.](image2)
Instead of creating a semester from scratch every time, teachers can also clone an existing semester, which will also clone recursively all the decks inside that semester and their respective flashcards. This is especially useful when you have many decks in a semester and do not want to go through the trouble of creating them once again or importing them one by one. In Figure 21, we can see the “Clone semester” page, which is similar to the “Create semester” page except that here we have a list of all the semesters in all the courses in which the user is assigned as a teacher. After filling out the form, the user simply has to click the “Clone” button of one of these semesters and the process of cloning is complete.

[Image of the “Clone semester” page in ADAM]

In Figure 21 we can see how the semester page looks like from the point of view of a student enrolled in that semester. There is a button to view the semester leaderboard and a list of the decks in that semester, if there are any. The student can then click on the “View” button of one of the decks to go to that deck’s page.

Alternatively, in Figure 23, we can see how the semester page looks like when the student is not enrolled. In this case, the semester has a password that the student must type before enrolling. If the semester did not have a password, the password field would not be visible and the student would only need to click the button to enroll.

If a semester is closed, any enrolled student can still access it normally but students that are not enrolled will only see an empty page similar to the one represented on Figure 24.
Figure 22 – “View semester” page in ADAM as seen by an enrolled student.

Figure 23 – “View semester” page in ADAM as seen by a student not enrolled.
Students enrolled in a semester or teachers of that course can view that semester’s leaderboard, where it is possible to see and ordered list of all the students enrolled by score, and the number of correct answers and total answers that each student has, as can be seen in Figure 25. This is useful for students to see how they are doing compared to other students and for teachers to keep track of the students’ performance on that semester.
From a course administrator’s point of view, the semester page has a few extra buttons, as can be seen in Figure 26. The first one is the “Edit semester” button, which sends you to the “Edit semester” page. The other three are for creating, cloning or importing a deck. These tasks will be explained further in this chapter. A user assigned as teacher in this course will not be able to see the “Edit semester” button, as that task is reserved for administrators only.

![Figure 26 – “View semester” page in ADAM as seen by a course administrator.](image)
3.4.7. Decks

Decks are more customizable than courses and semesters. In Figure 27, we can see the “Create deck” page. The only thing required to create a deck is a name. However, it is possible to add a description, an image (via URL) and any number of categories. It is also possible to specify whether we want the flashcards in this deck to be learned in a random order instead of the default sequential order. By default, decks are kept private unless we flag them as public. Private decks are only visible to the course’s teachers and administrators. In the bottom of this page there is a preview area where we can see the final result that will appear on the semester’s list of decks, as can be seen on Figure 28.

![Create deck page in ADAM](image)

Figure 27 – “Create deck” page in ADAM.
In Figure 28, we have the page of the created deck as viewed by student enrolled in the semester. Besides the deck information, it is possible to view the deck leaderboard by clicking the corresponding button, our current goals status, our study stats and the list of cards in this deck.

The deck leaderboard is identical to the semester leaderboard, as can be seen in Figure 30, but only shows the stats related to this deck.

Currently there are three goals implemented. These are simple goals that seek to give the student some feedback about his current study status in this deck as well as motivate them a bit to keep studying. All three goals start with a red dot on the left which turns green when each is accomplished.

The study stats feature a line chart that shows our performance in the last 7 days, where we can compare our number of correct answers and number of total answers of each day. It also tells us how many of the cards we have already learned out of the total number of cards in this deck and the current estimated recall rate percentage for the entire deck.

In the card list, we can click on the eye icon in front of the card’s name to view that individual card or we can click on the “View all cards” button to view them all at once. This is useful if we want to quickly glance at a specific card or if we want to search for a specific term and we do not know in which card it is. At the left of every card’s name there’s a dot that can assume three different colors: red, if the card has not been learned yet, yellow, if the card has been learned but is due for review, and green, if the card has been learned and is still up to date (i.e. this card’s recall chance is still above 90%). Hovering/touching the name of the card also reveals the status in text form, including the time left until the next review on up to date cards.
Figure 29 – “View deck” page in ADAM.
In Figure 32, we can see the page of the same deck from the point of view of a teacher. It is also worth noting that this teacher is not enrolled in this semester, hence the warning right below the deck’s image and the “Go to semester” button being visible.

As can be seen, the teachers have a few extra buttons available: the “Edit deck” button, the “Create card” button, the “View deck stats” button and the “Export deck as JSON button”. The first button takes the teacher to the “Edit deck” page which is identical to the “Create deck” page with the exception that it has a “Delete deck” button at the bottom that, when clicked shows another button that says “Really delete”. If a user presses this last button, the deck is immediately deleted and all its cards are deleted as well. This can be seen on Figure 31.
Figure 32 – “View deck” page in ADAM as seen by a teacher.

Clicking the “View deck stats” button in the deck page will send the teacher to the “Deck stats” page, which can be seen on Figure 33. Here we can see a list of the deck’s cards sorted by the percentage of correct answers out of their total answers. This page is very useful for teachers to find out which cards are hard for students to memorize, which could mean that these cards may need some improvements or that their contents are truly hard and that the teachers may be able to go over them more thoroughly on their classes to help students understand them better.

On the cards list we can also see a few new icons in front of the cards’ names. The arrows allow us to change the order in which the cards are learned and the pencil allows us to edit that card.
The “Export deck as JSON” button shown in Figure 32 will, when clicked, sends the teacher to the “Export deck” page, seen on Figure 34, where he can copy the given JSON in order to store it somewhere or make some other use out of it outside of the application. It is also possible to paste this JSON in the “Import deck” page, seen on Figure 35, to import the deck to any semester that the teacher wants.
Lastly, it is also possible to clone existing decks from any semester inside of a course in which the user is assigned as a teacher or administrator. To do this, the user must go to the page of the semester where he wants the cloned semester to go into and press the “Clone existing deck” button, seen on Figure 26. The process is similar to cloning a semester, as can be seen on Figure 36. This operation allows teachers to import decks from other semesters, even from other courses, at any time, saving the trouble of creating them again from scratch and without the need to export and import the deck manually as a JSON.

![Figure 36 – “Clone deck” page in ADAM.](image)
3.4.8. Flashcards

When we click the “Create card” button seen on Figure 32, we go to the “Create card” page, seen on Figure 37. A card requires only a name to be created, but it needs at least one element on the front and one on the back to actually be useful. ADAM’s flashcards can have an image (via URL), a header and some additional text on each side. Every one of these elements is optional, and it is up to the user to decide which ones are best for each card. It is also possible to align the header and the text using the buttons right below the corresponding fields, which might be useful in some specific cases.

![Create card page in ADAM.](image)

Once again, the “Edit card” page is identical to the “Create card page” but, like in the decks, it includes a “Delete card” button at the bottom with an extra confirmation button, as can be seen in Figure 38.
When a user clicks on the eye icon in front of a card’s name, in the deck’s cards list seen in Figure 29, he is sent to the “View card” page of the corresponding card. In this page, he can visualize both the front and the back of the card at once. This might be useful if he wants to quickly have a glance at a specific card at a given moment.

Alternatively, a user can also press the “View all cards” button seen in Figure 29 to view all the deck’s cards at once. This page is similar to the “View card” page but shows all the cards instead of just one. In Figure 40, we can see partially how this page looks like. This page is also useful, not only because it allows to quickly scroll through all the cards, but also because it allows for the user to search for specific terms in this page in order to find the cards that contain them. For instance, if a user wanted to quickly view all the cards that contained the word “proximity”, he can come to this page and use the search feature present in his browser to quickly jump through all the occurrences of this word, therefore viewing all the cards that contained it.
Figure 40 – “View all cards” page in ADAM.
3.4.9. Study

After a user enrolls in a semester, he can start studying that semester’s decks, taking advantage of ADAM’s adaptive spaced repetition algorithm. To start studying a deck, the user has to go to that deck’s page and click the “Study” button see in Figure 29. Above the button there is also a bit of text letting him know how many cards he has due for review and that it is time for him to study, if that is the case. If he has no cards due for review, this text will instead tell him how much time is left before the next review date, if he has learned any of the cards in the deck already, as can be seen in Figure 41.

In the normal study mode, the user is first presented with the cards that are due for review ordered by their review date, if there are any. Cards with an older review date are presented before cards with a more recent review date, because their review is more urgent.

After studying the cards due for review, the user will be presented with the cards that he has not learned yet, if there are any.

A user can still study if he does not have any cards due for review or new cards to learn, if he wants to, in which case he will enter the random review mode. After a user studies all the cards he has due for review and learns all the cards in the deck, he will automatically enter this mode without interruption. In this mode, he can study for an indefinite period of time and correct answers will not affect the studied cards’ next review date but incorrect answers will reset the card’s progress just like in the normal study mode.

If at any point, while learning new cards or reviewing random cards, a card becomes due for review, that card will be presented next to the user after he finishes studying the current card.

In the normal study mode, correct answers will move the card to the next box in the Leitner system, scheduling the next review of that card based on the new box’s number and the user modifier using ADAM’s adaptive spaced repetition algorithm [33]. Incorrect answers will reset that card’s progress, sending it back to the first box and scheduling its next review date based on that box instead.

When studying, the user will be presented with the front face of the current card (Figure 42). He must look at it and try to recall what is on the back face of that card. After succeeding or giving up, he must then click/touch the card to flip it. After flipping the card, he will be able to view the back face of the card (Figure 43). He must now verify if he was correct, or try to memorize it again if he failed to recall it. If he was correct, he must press the green button at the bottom right corner of the card to mark its answer as correct and proceed to the next card. If he was not correct or failed to recall it altogether, he must press the red button...
in the bottom left corner of the card instead, marking its answer as incorrect and proceeding to next card as well.

In both faces of the card he will also be able to see his current estimated recall chance for the current card near the center bottom, if it happens to be equal to or above 50%, to motivate him to try recalling it harder. In both faces, he will also able to see, at the top left of the card, the status of the current card: if it has not been learned yet (red dot), if is due for review (yellow dot), or if it is up to date (green dot). The “up to date” status only shows up during random review mode and shows how much time is left before that card becomes due for review.

The order in which new cards show up depends on if the deck is set for cards to be learned in random or sequential order. In the first case, new cards will show up in random order, and in the second, the order will be the same that can be observed on the deck’s cards list (Figure 29). This order can be changed by the teacher by clicking the arrow icons in front of the cards’ names (Figure 32).

Figure 42 – “Study” page in ADAM with the front face of the card visible.

Figure 43 – “Study” page in ADAM with the back face of the card visible.
3.4.10. Application administration

The last feature that ADAM possesses is the ability to change the roles of users within the application. These roles are global and independent from the roles inside the individual courses.

By default, every user that enters the application is assigned as a student. However, that role can be changed to teacher or admin. There is a fourth role, superadmin, which cannot be assigned within the app. Superadmins are hardcoded in the server configuration file and cannot be demoted as well.

Both superadmins and admins can change the roles of every user in the application, except if that user is a superadmin. Superadmins and admins can perform any action inside of the app, as if they were administrators of all the courses in the application.

The only difference between teachers and students is that teachers can create courses and students cannot. However, users assigned as students can still be assigned as teachers or even administrators in individual courses and perform all the corresponding actions without any limitations.

In order to change the roles of users inside of the application, superadmins and admins have an extra button on the left panel, next to the logout button, that, when clicked, sends them to the “Admin” page (Figure 44). In this page, we can see a list of superadmins, admins, teachers and students in the application, and change their roles by clicking on the buttons in front of their names.

![Image of the “Admin” page in ADAM.](image-url)
4. Evaluation

In this chapter, we will present and discuss the multiple evaluation steps that we took. These included the analysis of ADAM’s usage data collected during the semester in which it was introduced in a university class to understand how it affected students that used it to study, student interviews to understand what these students thought of it and how it could be improved further, teacher interviews to hear their thoughts on ADAM and understand if they thought it could bring benefits to their classes and what kind of features they felt were necessary to use it more efficiently and effectively, and usability tests, in which 5th year Master’s students performed tasks that a university teacher would perform when using the application in his classes, to understand if is practical for continuous real-world usage.

4.1. Integration in a university class

ADAM was introduced in the second semester of 2016/2017 of the class of Multimedia Content Production of the Master’s degree in Engineering Systems and Computer Engineering at Instituto Superior Técnico. Starting on the 10th lecture, in the week of 27 of March, we started creating a deck of flashcards for each of the lectures of this class with help from the teachers of this class. This went on until the end of that semester, except for two of the last lectures that we were not able to convert into flashcards because the corresponding slides were purely image-based or too complex. The teachers of the class introduced ADAM to the students when it was first introduced, in that same week, during their practical classes. The students then, were free to enter ADAM and use it to study the materials from each lecture for their theoretical evaluations, if they wished.

The theoretical evaluations of this class consisted on 9 mini-tests, where only the 8 best grades out of the 9 counted for part of their final grades. The teachers of this class provided us with the grades of each mini-test for us to analyze along with the usage data we collected in ADAM. The grades of each mini-test vary from 0 to 750. By looking at the grades we quickly realized that some students never showed up to a single evaluation, so we removed them from the dataset. We also realized that, because only the best 8 grades counted, a vast majority of the students gave up on the last mini-test, either scoring a zero or having a very low grade. This makes this mini-test irrelevant, so we also removed it from the dataset. Finally, we also removed all students that had any zeroes in the remaining mini-tests, since we did not want to taint our results with students that were not taking the class seriously. This left us with 73 students remaining out of the 97 enrolled in the class.

The first obvious step was to look at the grades of all the mini-tests and try to differentiate between the first three, where ADAM was not available yet, and the last five, where ADAM was already available.

When ADAM was introduced in the class, the first 3 mini-tests had already been done, so ADAM started being available for students to study only from the 4th mini-test onward.
Table 3 – Multimedia Content Production mini-test grades. Gray cells represent mini-tests where ADAM was available for the students of this class.

<table>
<thead>
<tr>
<th>Mini-test</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Average grade</strong></td>
<td>371,5</td>
<td>415,1</td>
<td>475,3</td>
<td>502,4</td>
<td>503,4</td>
<td>603,4</td>
<td>580,0</td>
<td>473,6</td>
</tr>
<tr>
<td><strong>Median grade</strong></td>
<td>375</td>
<td>450</td>
<td>475</td>
<td>450</td>
<td>450</td>
<td>600</td>
<td>600</td>
<td>450</td>
</tr>
<tr>
<td><strong>IQR of the grades</strong></td>
<td>150</td>
<td>150</td>
<td>250</td>
<td>150</td>
<td>450</td>
<td>200</td>
<td>300</td>
<td>225</td>
</tr>
</tbody>
</table>

In Table 3, we can see that, during the period that ADAM was made available, the average grades were noticeably higher than before, with the exception of mini-test 8, where they fell down again. In mini-test 4, the median grade actually lowered slightly but the interquartile range reduced significantly, and the average went up a bit, suggesting that there were more students moving towards this median and less having lower grades. The interquartile ranges tripled from mini-test 4 to mini-test 5, but then lowered significantly after that, even though the averages and medians were much higher in mini-tests 6 and 7. For the 8th mini-test, we were not able to create one of the decks of its corresponding lectures because it was about “Animation & Synchronization” and its slides were purely image based, with no information that we could extract directly from them. As we can see in the table, the grades in this mini-test fell for the first time to the same range of values that they were at before ADAM was introduced. While this certainly does not prove that ADAM was responsible for the higher grades obtained while it was available, due to existence of many other external factors (e.g. mini-tests 4 through 7 could have been easier than the others), it does, at first sight, suggest that ADAM helped students achieve these higher grades.

Trying to understand this better, we separated the 73 students into two groups. One of the groups had the 36 most active ADAM users and the other had the 37 least active ones. To distinguished between the two groups, we sorted them by the number of cards studied during the semester. We were then surprised to see that the least active member actually got significantly better grades in all mini-tests but the 2nd, as can be observed by comparing Table 4 and Table 5. While this result went completely in the opposite direction than we expected it to, it does not necessarily mean that ADAM did not help students from the first group or even from both. Students from both groups still managed to get better grades while ADAM was available, and even if people from the second group did not take much advantage of the adaptive spaced repetition algorithm, they might have still used ADAM to view the cards manually. And even if that is not the case, it just means that not everyone needs ADAM in order to succeed, but maybe some do. Maybe the most active students were the ones that needed it the most because they were having a harder time getting good grades, and ADAM might have helped them, which is why they continued using it more than the others. This would be a hard question to answer, as it would require us to know more about the academic background of each of the students, and even so, just because they were good or bad in other classes
does not mean they should be equally good or bad in this one. But, at least, they must have had a good reason to decide to invest their time in ADAM.

Table 4 – Multimedia Content Production mini-test grades of the 36 most active ADAM users. Gray cells represent mini-tests where ADAM was available for the students of this class.

<table>
<thead>
<tr>
<th>Mini-test</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Average grade</strong></td>
<td>347,0</td>
<td>437,5</td>
<td>466,7</td>
<td>435,4</td>
<td>441,7</td>
<td>577,1</td>
<td>537,5</td>
<td>464,6</td>
</tr>
<tr>
<td><strong>Median grade</strong></td>
<td>300</td>
<td>450</td>
<td>450</td>
<td>450</td>
<td>450</td>
<td>600</td>
<td>600</td>
<td>450</td>
</tr>
<tr>
<td><strong>IQR of the grades</strong></td>
<td>234,3</td>
<td>300</td>
<td>262,5</td>
<td>187,5</td>
<td>300</td>
<td>231,3</td>
<td>150</td>
<td>243,8</td>
</tr>
</tbody>
</table>

Table 5 – Multimedia Content Production mini-test grades of the 37 least active ADAM users. Gray cells represent mini-tests where ADAM was available for the students of this class.

<table>
<thead>
<tr>
<th>Mini-test</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Average grade</strong></td>
<td>395,3</td>
<td>393,2</td>
<td>483,8</td>
<td>567,6</td>
<td>563,5</td>
<td>629,1</td>
<td>621,3</td>
<td>482,4</td>
</tr>
<tr>
<td><strong>Median grade</strong></td>
<td>450</td>
<td>450</td>
<td>475</td>
<td>600</td>
<td>600</td>
<td>600</td>
<td>600</td>
<td>450</td>
</tr>
<tr>
<td><strong>IQR of the grades</strong></td>
<td>188</td>
<td>150</td>
<td>225</td>
<td>300</td>
<td>300</td>
<td>150</td>
<td>150</td>
<td>225</td>
</tr>
</tbody>
</table>
Next, to understand if the students kept a regular study schedule over the semester, we analyzed the number of times that the application was loaded for each day of the week. As we can observe in Figure 45, ADAM was loaded mostly on Monday and about half as many times on Sunday. The explanation for this is simples: mini-tests were done on Mondays. Students only really accessed the application right before the mini-tests, which also explains why Sunday had so many accesses as well.

![Figure 45 – Visualization of the number of times, for each day of the week, that ADAM was loaded.](image)

In Figure 46, we can see the number of times that students studied a card for each day of the week. Monday and Sunday were the days in which students studied the most, which was predictable after seeing Figure 45, but there were some cards being studied on Saturday and Thursday, which was a bit surprising and nice to see.

![Figure 46 – Visualization of the number of times, for each day of the week, that students studied a card in ADAM.](image)
By comparing Figure 46 and Figure 47, we can conclude that, for the most part, students were much keener on studying using the adaptive spaced algorithm than by manually looking at the cards in a deck. This was also a nice surprise, because we expected students to “cheat” and view the cards manually instead of putting in the effort and using the study mode, especially on the days of the mini-tests. There is still a quite big number of manual card views on Monday, but it is much smaller than the number of cards studied on that same day of the week. This same approximate proportion can also be observed on Sunday.

![Bar chart](image)

**Figure 47 – Visualization of the number of times, for each day of the week, that students viewed a card manually in ADAM.**
4.2. Interviews

In this chapter, we will present our results from the student interviews we held with about 30 students of the previously mentioned class, as well as the results from the interviews we held with two teachers of this same class.

4.2.1. Student interviews

Around the last third of the semester, we interviewed about 30 of the students from the class of Multimedia Content Production of the Master’s degree in Engineering Systems and Computer Engineering at Instituto Superior Técnico from the Taguspark Campus, where ADAM was introduced. We interviewed them in groups of three (or 2, in some cases) and asked them 7 questions that can be found in Appendix A.

All the students interviewed knew ADAM, since it had been presented to them in the practical classes by the teachers at the time it was introduced in the class. However, half of the interviewees admitted only having used it a couple of times to try it out, while the other half said they used it one or two days before the mini-tests and on the day of the mini-tests as well.

The first half said that they did not trust the application enough and were afraid of missing information if they studied through it instead of the class slides, or that they preferred studying through the slides because they were used to it already and felt it was faster. Some also said that they tried it for a mini-test but the questions that they got as evaluation were not related to what they had studied. The problem here is that the flashcards were prepared by the author of this thesis and not by the teachers of the class, therefore we were limited to using the information found on the class slides and nothing more. If the teachers were creating the flashcards based on what they actually wanted to evaluate, this kind of mistrust could be avoided and the students would be much happier when studying through the application.

The second half said they enjoyed using the application because it contained the relevant information found on the class slides without the extra content, so they saved time and effort by not having to sift through the hundreds of slides trying to find that information and risk missing it. They too, thought that the contents of the flashcards were a bit off from the evaluation questions but the same thing would happen if they studied through the slides, since they were the source of the flashcards. They said that felt that using the application in the day of the mini-test and one or two days before was enough for them to grasp the concepts, which is why they only started studying near the mini-tests.

Even the students that only used the application a few times thought that it helped them study and really memorize the concepts. Some of them even told us that they still remembered the concepts from the first deck they studied. However, due to the already mentioned differences from the evaluation questions, many were left a bit unmotivated to continue using it often.

The students that continued to use the application thought that it is efficient and very effective, although they felt that the spaced repetition algorithm was too pushy on reviews, making them review too often when
they were trying to learn new cards fast. This problem could be addressed by lowering the recall rate or by skipping the first one or two boxes of the Leitner system to free students from the early reviews that had very short inter-study intervals so that they could spend more time learning new cards before seeing the same cards again in a single study session [33].

The students said they really liked how simple, fast and good looking the application was and the fact that it worked well on multiple devices. Some commented that it was as simple as they could think of but still worked really well, and that they had adopted similar softwares for other classes after trying it out but they would prefer if they could use ADAM in those classes too.

There were some interesting feature suggestions made as well. One was a “sprint mode” where they could quickly run through all the cards in the deck while ignoring review dates, so that they could, for instance, very quickly review an entire deck before an evaluation or just see all the cards in the deck for the first time before starting to use the regular study mode. A few students suggested some gamification aspects like levels, experience, correct answers streaks, scores and leaderboards. A feature in great demand was also a way to view both faces of all the cards at once, similarly to how they could already view both faces of a single card, which was a feature that we implemented right after the interview and is now present in ADAM. Another great suggestion was a “test mode” where a few flashcards would be randomly selected and presented to the student for him to try to recall, and at the end he would get a grade based on how many he recalled correctly. This feature was not meant as an evaluation means but merely as a way for students to practice for evaluations and see how good of a grade they might get.

In this chapter, we will present the results obtained from interviewing two of the teachers of the university class where ADAM was introduced, as well as the results obtained from usability tests where university students performed the role of teachers in the application.
4.2.2. Teacher interviews

After the student evaluations finished, near the end of the semester, we interviewed two teachers of the university class of Multimedia Content Production of the Master’s degree in Engineering Systems and Computer Engineering at Instituto Superior Técnico, where ADAM was introduced. We asked each the same three questions, which can be found in Appendix B.

To the first question, the first teacher said that he thinks ADAM brings great benefits not only for the Multimedia Content Production but many other classes as well. According to him, there are many instances where students are evaluated but there are not many instances where students get feedback. Students attend a test and get a grade, but there is not an opportunity to iterate and interact in a way that allows them to try multiple times and figure out what they did well and what they did wrong, and learn with that. In this class, they try to implement this kind of strategy, where they allow students to submit a project, get a grade and receive comments, and to try to improve and submit it again, possibly improving their grade. However, he says this really only applies to the practical aspect of the class, and is much more difficult to deal with the theoretical aspect. He says that they adopted a continuous evaluation method with multiple mini-tests instead of few regular tests to give the students more opportunities to get good grades, but this still falls under the same evaluation method where students do a test and get a grade. They still don’t have a way to improve in a subject after the test is done. Therefore, he welcomes ADAM, since it allows students to study the theoretical parts of the class while still getting feedback as they study, allowing them to identify their mistakes and try to correct them. He also thinks the spaced repetition is a great addition that makes ADAM able to fill in an existing need in the class that, so far, they had only been able to fill in the practical aspect and not in the theoretical aspect. He thinks ADAM is definitely able to help the students that put in the effort to use the application.

To the second question, the first teacher said that he sees ADAM as a formative tool and not as a means to evaluate students, which is also how we see it. Therefore, he thinks that, while tracking the detailed progress of the students would seem like a good idea at first, morally he thinks that it might not be such a great idea because, even unconsciously, looking at those stats might make teachers want to favor some students over the others. Alternatively, he thinks the best approach would be to have a way to see which cards students are having more trouble with, sort of like a leaderboard for the cards themselves. He says that teachers many times have a hard time figuring out where students are having trouble, and this would allow them to find out and try to explain those parts better to them, even before the evaluations, so that students can get better grades. We thought this suggestion was pretty good and it convinced us to implement the “Deck stats” page after the interview (Figure 33).

To the third and final question, the first teacher said that the biggest problem that teachers face is not having the time to create more resources for students to use and study from. Especially in this class, where they have many aspects of gamification already implemented, they end up spending most of their time just managing all those aspects, which leaves them with very few spare time to do anything else. He said that,
when he wants to add a new mini-test question to the Moodle platform where students are evaluated, even if he has everything planned out already on his head, it still takes him from 5 to 10 minutes to add that single question. This is due to the platform complexity, which offers many options which he has to select from. He thinks that the flashcards are much simpler than that, which for him is a good thing. So, the thing he values the most is having a really simple way to create flashcards very fast with minimal effort. Ideally, for him, creating a flashcard or even a bunch of flashcards should be as easy as editing a text file. He thinks that the application should offer a good, simple interface but also a way for him to input some text or even upload a text file, which is then converted into flashcards (ADAM already has the feature of importing a deck as a JSON, which could be expanded upon). Additionally, he thinks that the concepts of course and semester are very useful, as well as the ability to carry over the existing decks from one semester to the other (these concepts and functionality are already in ADAM). Another feature he feels that is very important is the ability to export decks from the application so that the application does not become a sink, where they put in time and effort to create flashcards and then can never get them out if they later decide to move to other system or do something else with them (decks can be exported as a JSON in ADAM). He said that it was good that the application was fit for mobile usage since that is now the most used type of devices. At the end of the interview, he said he believed ADAM was a great application that would certainly help this class in the future and hopefully even others.

To the first question, the second teacher said that she believed ADAM can benefit the students of this class thanks to the adaptive strategy used, since it helps achieve the personalized study that this class strives for. Students can reinforce the parts where they are not so good by studying with ADAM thanks to its adaptive spaced repetition algorithm that adapts to each student individually, helping them learn at their own pace.

To the second question, the second teacher said that she thinks that it is very important for the student to be able to know how they are progressing with their study in the application and also to be able to compare their own progress with other students (currently this is possible via the stats and charts available in the deck page and via the leaderboards). By comparing their progress with other students, she believes that students can understand if they are progressing at a regular pace or if they need to invest more time studying. By watching their own progress, they should also be able to understand if they are getting closer to achieving their personal goals (for instance, they might want to get a grade of 18 in the next evaluation). From a teacher point of view, she would like to see where the students are struggling the most so that she can put special effort into helping them overcome their difficulties in those particular areas, either by trying to explain them again in other ways or by creating other resources for them to study or practice with. She also talked about how they usually are able to predict how grades will turn out as the semester goes on and that having additional information about the students’ progress could help them realize in advance if students are going to do poorly in the near future, so that they can try to help them or at least notify them that they need to put in some more effort or they might not do so great in the mini-tests. She said that very often in the practical classes, they brought in concepts lectured in the theoretical classes and made students
work with them in practice to better understand those concepts. With the help of ADAM, she believes that it would be possible to pick the most adequate concepts by looking at the ones that students seem to have a hard time learning, therefore helping them more than if they used other concepts that students are already familiar with or find easy to understand.

To the third and final question, the second teacher said that the ideal feature to help create decks and flashcards would be one that could take advantage of the existing materials, like class slides, and automatically be able to extract the information from them. However, she believed that this would be a huge technical challenge. However, she thought that having a very simple way to create flashcards that would require very little effort would be good enough for it to become practical, like the suggestion the previous teacher made of converting bare text into flashcards automatically.
4.3. Usability tests

After the interviews with the teachers, we reworked the application, adding a few extra features to fill in most of the needs that the teachers brought up. It was now time to evaluate the application from the point of view of the teachers, to see if it was easy to use and practical enough to be continuously used in a university class.

Because this evaluation followed the end of the semester, it would be hard to perform it with actual teachers, so we chose instead to perform the usability tests with a group of 10 university students from a few different areas. These students evaluated the application by performing tasks that a teacher would perform when using the application in his university classes. The students that performed this test are all in their 5th year and completing their Master’s degrees, and are, therefore, used to the way classes are lectured and what type of resources they usually have available to study or practice with. Because of this, we believe we can say that they are as close as anyone can get, without actually being teachers, of understanding the role of a university teacher trying to provide his students with resources and means to study.

The script and the resources used during the usability tests can be found in Appendixes C and D, respectively. The tasks performed by the subjects in these usability tests were developed with the help of a teacher of the university class of Multimedia Content Production of the Master’s degree in Engineering Systems and Computer Engineering at Instituto Superior Técnico.

Before performing the usability tests with the subjects, the author of this thesis, Diogo Pais, performed all the tasks by himself and timed it so that we could then compare the times of an experienced user of the application with those of people who were using it for the first time.

The students performing the tests were given 3 minutes to fully explore the application, being allowed to create, edit and delete anything they wanted, so they could get a feel for it and understand minimally how it worked. Then, we restored the state of the app and started the tests, which consisted in a series of tasks that they were given, once at a time, and should try to perform without any help. During the tests, we timed the execution time of each task and took note of the mistakes made.

Every task was performed by the students with a 100% completion rate.
Table 6 – Task execution times gathered in the usability tests, in the format “mm:ss”.

<table>
<thead>
<tr>
<th>#</th>
<th>Task</th>
<th>Diogo Pais</th>
<th>Subjects’ median</th>
<th>Subjects’ IQR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Create a deck</td>
<td>00:38</td>
<td>01:06</td>
<td>00:13</td>
</tr>
<tr>
<td>2</td>
<td>Create three cards</td>
<td>00:50</td>
<td>02:12</td>
<td>00:21</td>
</tr>
<tr>
<td>3</td>
<td>Add a description to the course</td>
<td>00:16</td>
<td>00:22</td>
<td>00:14</td>
</tr>
<tr>
<td>4</td>
<td>Edit two cards</td>
<td>00:32</td>
<td>01:18</td>
<td>00:20</td>
</tr>
<tr>
<td>5</td>
<td>Edit the deck</td>
<td>00:21</td>
<td>00:28</td>
<td>00:08</td>
</tr>
<tr>
<td>6</td>
<td>Assign a user as course teacher</td>
<td>00:06</td>
<td>00:10</td>
<td>00:08</td>
</tr>
<tr>
<td>7</td>
<td>Change card order</td>
<td>00:12</td>
<td>00:27</td>
<td>00:11</td>
</tr>
<tr>
<td>8</td>
<td>Delete a card</td>
<td>00:04</td>
<td>00:09</td>
<td>00:06</td>
</tr>
<tr>
<td>9</td>
<td>Close the semester</td>
<td>00:11</td>
<td>00:14</td>
<td>00:11</td>
</tr>
<tr>
<td>10</td>
<td>Clone the semester</td>
<td>00:26</td>
<td>00:31</td>
<td>00:12</td>
</tr>
<tr>
<td>11</td>
<td>Consult semester leaderboard</td>
<td>00:06</td>
<td>00:12</td>
<td>00:07</td>
</tr>
<tr>
<td>12</td>
<td>Clone a deck</td>
<td>00:27</td>
<td>00:32</td>
<td>00:06</td>
</tr>
<tr>
<td>13</td>
<td>Consult deck leaderboard</td>
<td>00:05</td>
<td>00:20</td>
<td>00:07</td>
</tr>
<tr>
<td>14</td>
<td>Consult deck stats</td>
<td>00:05</td>
<td>00:25</td>
<td>00:15</td>
</tr>
</tbody>
</table>

As we can see in Table 6, an experienced user is able to perform every task in well under a minute, even tasks that require multiple steps, like creating a course, a semester and a deck, creating three cards, and editing two cards. The subjects of the usability test took a little more time, as was expected, but were also able to finish the tasks in very reasonable times, only taking over a minute in three of the tasks. The task that took the most time was the creation of three cards, which is understandable as this was the most complex task and the most time consuming one. However, they were able to finish it in a little over two minutes, which equates to less than a minute per card. When we compare this time with the 5 to 10 minutes that one of the teachers interviewed previously said that it took him to create a new question in the Moodle platform that they use to evaluate students, it seems like a major improvement.

The interquartile range from the times gathered also seem pretty reasonable, considering that 10 students were evaluated, with the two biggest values being 21 and 20 seconds in the tasks that required creating and editing cards, which are the most complex ones.
At the end of every task, we asked the subjects the single ease question, which consists of asking them to classify from 1 (very easy) to 7 (very hard) how hard they thought the task was. The results can be observed on Table 7. The subjects classified most of the tasks as “very easy”, the hardest ones being the cloning tasks and then the tasks that required multiple steps.

Table 7 – Single ease question answers gathered in the usability tests.

<table>
<thead>
<tr>
<th>#</th>
<th>Task</th>
<th>Median answer</th>
<th>IQR of the answers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Create a deck</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Create three cards</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>Add a description to the course</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>Edit two cards</td>
<td>1,5</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>Edit the deck</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>Assign a user as course teacher</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>Change card order</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>Delete a card</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>Close the semester</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>Clone the semester</td>
<td>1,5</td>
<td>1,75</td>
</tr>
<tr>
<td>11</td>
<td>Consult semester leaderboard</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>Clone a deck</td>
<td>1,5</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>Consult deck leaderboard</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>Consult deck stats</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

The most common errors were the subjects not being able to tell the difference between the header and text fields when creating a card, which is understandable as the main difference is the font size and that is not something that you can predict reliably without trying for the first time, and clicking the wrong “Edit” button when asked to edit something else. For instance, when asked to edit the course description or two of the cards of a deck, some immediately clicked the “Edit semester” button instead and then realized that this was not what they wanted to do, but then managed to carry on the task perfectly the rest of the way.

Other less common errors happened when asked to view the stats of the cards of a deck, when two subjects clicked the “View all cards” button instead, and during the task of cloning a semester or a deck. One of the subjects pressed the “Clone” button before filling out the form with the information of the new semester,
because he expected it to be a two-step process, and another subject tried to clone a deck by going into the semester containing the deck he wanted to clone instead of going into the semester where he wanted to clone the deck into.

After all the tasks were completed, we asked the subjects to classify the 10 questions of the System Usability Scale from 1 (strongly disagree) to 5 (strongly agree) [36]. After scoring their responses, we got a median value of 93.75 with an interquartile range of 9.375, putting our application way above the average score of 68, which means that users thought ADAM had great usability overall.

We also asked them a few open-answer debriefing questions in order to get some feedback from their experience using the application during the usability test. The feedback that we got was mostly towards the interface of the application, where subjects said that there were some improvements that could be made to the navigation, by including some extra links and buttons to allow the user to go back or to go from the semester to that semester’s course, and that there were many buttons in the application that looked identical and were a bit unorganized, suggesting we used different colors, and maybe icons, and placed them in a more logical way. There was also some feedback about the cloning process, where one subject suggested that the process could be split in two steps to make it more intuitive.

Other than that, the subjects thought the application was very simple, complete, customizable, practical, easy to use and fast. One also commented that he liked how the courses, semesters, decks and flashcards were structured hierarchically.

To finish the debriefing, we asked the students what was their area of study and took advantage of the fact that they are from a few different areas to ask if they thought that the application could be used in some of their classes. Three of the students are from Information Systems and Computer Engineering, two are from Biomedical Engineering, three are from Biotechnology (which follows Biology), one is from Physics and one is from Electrical and Computer Engineering. All of them believe that this application could be used in many of their classes, because most, if not all, require the memorization of many concepts and definitions, as well as being able to identify and categorize different things. One of the subjects even said that he believed the application could be used to study every kind of theoretical concept.
5. Conclusions

The teachers of the class of Multimedia Content Production of the Master’s degree in Engineering Systems and Computer Engineering at Instituto Superior Técnico interviewed think there is a real need for a tool to help students study the theoretical concepts of their class. They have already spent many resources setting up a blended learning environment with gamification aspects so that students can have an iterative evaluation process, where they submit a project, receive a grade and feedback, and can then resubmit it after improving it based on the feedback in hopes of improving their grade. However, they were only able to apply this to the practical evaluations of this class. The theoretical evaluations of the class still follow the traditional pattern where students do the test and get a grade, but don’t get a chance to improve that grade. Therefore, they think ADAM would be a great addition to their blended learning environment, allowing students to have an iterative, interactive and personalized way of studying while getting feedback on what they need to improve and how good they are doing, and to compare their progress with their peers to see if they are ahead or falling behind. One of the teachers said he even believed ADAM would be a great addition to many other classes as well, and not just this one. Besides helping students, ADAM could also be used for teachers to follow the progress of their students and identify problems with their materials in order to improve them or explain and explore them further in the theoretical and practical classes, benefiting both sides.

Interviews of students of this class also revealed that students think ADAM helps them learn with less effort, because all they need to know is already ready for them to learn within the application, and that the concepts they learn with it really stick in their memory effectively. However, they revealed that they only studied on the day of the evaluations and one or two days before, not really following a continuous study schedule. Many also revealed that they did not trust the application enough because they were afraid there was not as much information there as was in the class slides, or that they gave up because the flashcards were not sufficiently related with the evaluation questions and that they were wasting their time studying them. Most also thought that the adaptive spaced repetition algorithm was very effective but could be toned down to be less pushy in the early stages of study of a flashcard, allowing for more learning and less reviewing per study session.

Usability tests performed with ten 5th year Master’s students, who performed tasks that a university teacher would perform when using the application in his classes, showed that ADAM is believed to be very simple, complete, customizable, practical, fast and easy to use even for complete beginners. Very few and innocuous mistakes were made during the usability tests, having been immediately corrected by the subjects themselves and not happening again for the rest of the test. They also pointed out a few things that could be improved, like more navigation elements and the better placement of some elements like buttons. The subjects were from a few different courses and all thought that ADAM could definitively be used in plenty of classes of their areas of study, since most, if not all, required memorization of concepts
and definitions, as well as being able to identify and categorize different things. One of the subjects even said that he believed the application could be used to study every kind of theoretical concept.

Looking at the grades of the students of the *Multimedia Content Production* class and the data collected in ADAM, during the time in which they used it to study for the theoretical evaluations of this class, allowed us to observe that grades were consistently higher after ADAM was introduced in the class, having fallen down again to the previous range when we failed to create the deck of flashcards of one of the lectures that was tested. However, further analysis showed that the half of the students that were least active in the application managed to get consistently higher grades than the most active half, which went against our expectations. However, this does not necessarily mean that ADAM did not help at least some students achieve better grades than they would if it was not available. Maybe, without ADAM, the most active students would have gotten even worse grades, and the reason they were so active in ADAM might have been that they needed all the help they could get. The data collected further revealed that students studied mostly on the day of the evaluations and about half as much on the day before, which proves that they did not follow a continued study schedule as would be optimal. An interesting observation is that students were much keener in studying using the adaptive spaced repetition algorithm than by manually viewing the flashcards of a deck, which suggests that they find the study mode a better way of studying.
6. Future work

It is, as far as we know, planned to keep ADAM online and available for the students of Instituto Superior Técnico. We think ADAM turned out to be a very good platform and that it could be further improved and enhanced with new features, allowing other studies to be conducted.

One aspect that we think could be further explored is the adaptive spaced repetition algorithm. We believe there are plenty of things that could be experimented with here, such as finding a way to quickly catch back up with a card when we fail to recall it after a long success streak with it, skipping possibly unnecessary short inter-study intervals. It could also be possible to assign each card an inter-study interval modifier by analyzing its recall rate among all the students, making use of the group performance to predict the difficulty of that card and try to compensate for it. Teachers could also be able to attribute each card an estimated difficulty rating when they create it, immediately assigning them a more appropriate modifier, which could then still be adjusted and refined based on the group performance. This modifier would also be useful for teachers to get feedback on the relative difficulty of the flashcards. As we concluded from the teacher interviews, many times teachers have trouble figuring out where students are actually struggling until it is too late, which typically happens when they are grading the tests.

The application itself could be expanded upon and more features could be added, even going beyond just flashcards. More study modes could be implemented, like the “sprint mode” and “test mode” suggested by students in their interviews, and flashcards could go beyond just binary feedback and be able to alternatively have multiple answers for the student to choose from, or a text box for them to write the answer. Gamification aspects could also be added, which might help achieve a continuous study schedule. For instance, if students achieved certain goals they could unlock new practical tasks in their classes that would give them bonuses if done correctly, similarly to how the Multimedia Content Production skill-tree works.

It is also possible to implement ways to automate the creation of decks and flashcards in different ways, like was suggested by the teachers in their interviews. For instance, by uploading a text file which would be converted into flashcards.

Another aspect that could be explored is figuring out the best time of the day to send some sort of notification to the user when he has reviews due. For instance, by creating a heatmap based on the time of his previous study sessions. This way, it would be possible to figure out when the student is more likely to react positively to the notification and start studying, for instance, when he is waiting for a bus or going home in the subway and not really doing anything else productive or interesting.

We are sure there are many other ways in which people could take advantage of ADAM, most of which we have not even thought of, to perform research in a university blended learning environment. ADAM’s code is simple, flexible, robust and easy to read and understand, so we believe there is much that could be done with it and much that could be added on top of it.
7. References


Appendixes

1. Appendix A – Student questionnaire

1. Conheces a aplicação de flashcards da cadeira de PCM?
2. Usas a aplicação? Quando ou com que frequência? Porquê?
3. Sentes que a aplicação te ajudou a estudar?
4. Achas que o estudo é mais eficiente ou eficaz com a ajuda da aplicação?
5. Ponto fracos: o que achas que pode ser melhorado?
6. Pontos fortes: houve alguma coisa que tenhas gostado especialmente na aplicação?
7. Sugestões?

2. Appendix B – Teacher questionnaire

1. Acha que uma aplicação deste tipo traz benefícios para a cadeira de PCM? Porquê?
2. Que dados seriam relevantes visualizar na aplicação? (por ex. para saber qual o panorama actual dos estudos dos alunos)
3. Que funcionalidades seriam cruciais para uma melhor integração desta aplicação na cadeira de PCM? (por ex. relativamente à criação e gestão de conteúdos)
3. Appendix C – Usability tests script

Explicar o papel do sujeito e o conceito e âmbito da aplicação

Durante este teste de usabilidade vais desempenhar o papel de um professor universitário que está a testar a utilização de uma aplicação de apoio à aprendizagem numa das suas cadeiras para auxiliar a aprendizagem dos alunos fora das aulas. Deverás, portanto, manter um espírito crítico em relação à aplicação para que no final consigas dar algum feedback.

Nesta aplicação é possível criar baralhos de cartas, organizados por curso e semestre, que os alunos poderão estudar em qualquer altura e local. Na página 1 do PDF podes ver um exemplo de uma destas cartas, normalmente conhecidas como flashcards. Estas cartas são muito usadas para ajudar na memorização de vários tipos de informação, sendo mais frequentemente usadas para a aprendizagem de vocabulário de línguas estrangeiras, como podes ver no exemplo. Podem, no entanto, ser também usadas para a memorização de factos, conceitos, definições, etc. As cartas têm duas faces e o estudante deve olhar para apenas uma delas e tentar lembrar-se daquilo que está na outra face. De forma a memorizar, o estudante deve efectuar um estudo repetido destas cartas. Esta aplicação contém um algoritmo que agenda a próxima data ideal de estudo de cada carta com base no conhecimento estimado do estudante e no seu desempenho pessoal.

Explicar o teste

Antes de iniciar o teste, dar-te-ei três minutos para explorares a aplicação à vontade e tentares familiarizar-te um pouco com a sua interface e funcionamento. Seguidamente, daremos início ao teste de usabilidade.

Durante o teste, ir-te-ei dando uma tarefa de cada vez que deverás tentar realizar sem a minha intervenção.
Tarefas

1. **Criar um baralho**

A primeira tarefa é criares um baralho de cartas para o 1º semestre de 2017/2018 da cadeira de PCM. Para tal, terás primeiro de criar a cadeira, ou curso, e o semestre. Na página 2 do PDF tens a informação necessária para a realização desta tarefa.

Numa escala de 1 a 7, sendo 1 “muito fácil” e 7 “muito difícil”, quão difícil foi a realização desta tarefa?

2. **Criar três cartas**

Agora irás criar três cartas dentro do baralho criado na tarefa anterior. Estas cartas deverão ter um aspecto idêntico às cartas representadas nas páginas 3, 4 e 5 do PDF. Nessas páginas estará também toda a informação necessária para a sua criação.

De 1 a 7, quão difícil foi a realização desta tarefa?

3. **Acrescentar uma descrição à cadeira**

Quando criaste a cadeira não lhe adicionaste uma descrição. Sentes que isso poderá ser importante e decides acrescentar-lhe uma descrição. Na página 6 do PDF encontrarás uma descrição que poderás adicionar à cadeira.

De 1 a 7, quão difícil foi a realização desta tarefa?

4. **Editar duas das cartas**

Após a última tarefa apercebes-te que há alguma informação que poderás incluir nas duas últimas cartas do baralho criado anteriormente. A tua tarefa é editá-las para ficarem idênticas às cartas representadas nas páginas 7 e 8 do PDF. Novamente, nestas páginas terás toda a informação necessária.

De 1 a 7, quão difícil foi a realização desta tarefa?

5. **Editar o baralho**

Novamente, apercebes-te que o baralho que criaste não tem descrição nem imagem. Como tal, decides acrescentar-lhe a descrição e imagem que estão na página 9 do PDF. Nessa página tens também o aspecto final que deverás obter.

De 1 a 7, quão difícil foi a realização desta tarefa?
6. Atribuir o cargo de professor a outro utilizador

Outro professor da cadeira, o Rui Ramos, acabou de entrar na aplicação pela primeira vez. Atribui-lhe o cargo de professor (teacher) da cadeira de PCM para que ele também possa criar e editar baralhos para os alunos.

De 1 a 7, quão difícil foi a realização desta tarefa?

7. Alterar a ordem de aprendizagem das cartas

Altera agora a ordem de aprendizagem das cartas do baralho que criaste. Na página 10 tens uma representação da ordem correcta.

De 1 a 7, quão difícil foi a realização desta tarefa?

8. Remover uma carta

Um dos outros professores da cadeira notifica-te que há uma parte da matéria que já não irá ser leccionada neste semestre. Uma das cartas que criaste deixa, portanto, de ser precisa e deverá ser removida do baralho. Remove a carta que contém a definição de “Proximity”.

De 1 a 7, quão difícil foi a realização desta tarefa?

9. Fechar o semestre

Chegou o final do 1º semestre de 2017/2018 e está a aproximar-se o início do 2º semestre. Como tal, devemos fechar este semestre da cadeira de PCM para evitar que novos alunos nele se inscrevam por engano.

De 1 a 7, quão difícil foi a realização desta tarefa?

10. Clonar o semestre

Para o 2º semestre de PCM, em vez de criar um semestre vazio, deverás clonar o 1º semestre, mantendo assim o baralho anteriormente criado. Na página 11 tens a informação relativa ao novo semestre.

De 1 a 7, quão difícil foi a realização desta tarefa?

11. Consultar o leaderboard de um semestre

Um dos outros professores de outra cadeira que leccionas, VI, pediu-te para veres quem foi o aluno com melhor pontuação no 1º semestre de 2017/2018 da cadeira de VI para lhe atribuir uma bonificação. A tua tarefa será identificar este aluno.

De 1 a 7, quão difícil foi a realização desta tarefa?
12. Clonar um baralho para o semestre mais recente

Houve uma alteração dos conteúdos leccionados no 2º semestre e foi acrescentada uma nova porção de matéria a PCM. No entanto, a cadeira de VI já contém um baralho acerca desta matéria. A tua tarefa é clonar esse baralho para o 2º semestre de PCM.

Toda a informação necessária está na página 12 do PDF.

De 1 a 7, quão difícil foi a realização desta tarefa?

13. Consultar o leaderboard de um baralho

Após consultar a pontuação do aluno Miguel Pereira, ficas curioso acerca de qual a pontuação que o mesmo obteve no baralho que clonaste anteriormente. A tua tarefa é consultar qual foi a sua pontuação nesse baralho bem como comparar o seu número de respostas correctas com o número total de respostas. Atenção que deves consultar o baralho original, da cadeira de VI, e não a cópia que fizeste para a cadeira de PCM.

De 1 a 7, quão difícil foi a realização desta tarefa?

14. Consultar as estatísticas de um baralho

Após verificares o baixo número de respostas correctas deste aluno face ao número total de respostas, perguntas-te se haverá cartas neste baralho com as quais os alunos estarão a ter dificuldades. Para poderes melhorar estas cartas terás primeiro de identificar quais as 3 cartas com a pior percentagem de respostas correctas.

De 1 a 7, quão difícil foi a realização desta tarefa?
Debriefing

Perguntas para feedback

1. Sentiste algum desafio durante a realização de alguma das tarefas?
2. O que gostaste mais na aplicação?
3. O que gostaste menos na aplicação?
4. Qual é o teu curso?
5. Achas que esta aplicação poderia ser usada em cadeiras do teu curso? Porquê?
6. Tens alguma sugestão ou comentário em relação à aplicação?

System Usability Scale

Numa escala de 1 a 5, sendo 1 “strongly disagree” e 5 “strongly agree”, classifica cada uma das seguintes frases:

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.
4. Appendix D – Usability tests resources

Nome do curso: PCM

Nome do semestre: 2017/2018 - 1º Semestre

Password do semestre: pcm_2946

O semestre deve estar aberto para que os alunos se possam inscrever.

Nome do baralho: Graphic Design Basics

O baralho deverá ser público e as cartas deverão ser aprendidas sequencialmente (por ordem).
**Página 3**

**Name:** Proximity  

**Front**  
In graphic design, what’s the idea behind proximity?  

**Back**  
Related items should be close to each other.

**Página 4**

**Name:** Grouping related information  

**Front**  
What’s the gain of grouping related information?  

**Back**  
Simplifies the design (less elements) and tells the user how it should be read.
### Name: Alignment

**Front**

In graphic design, what does a good alignment achieve?

**Back**

A good alignment unifies and organizes the elements.

---

**Página 6**

**Descrição do curso**

Aqui poderás rever os conteúdos leccionados nas aulas em qualquer altura para que consigas obter as melhores notas nos mini-testes!
Name: Grouping related information

Front

https://i.imgur.com/yis8BRq.png

What's the gain of grouping related information?

Back

https://i.imgur.com/kCxeFTJ.png

Simplifies the design (less elements) and tells the user how it should be read.
**Name:** Alignment

**Front**

In graphic design, what does a good alignment achieve?

**Back**

A good alignment unifies and organizes the elements.

We don't like messy things, we prefer order!
Descrição do baralho

Neste baralho poderás rever todos os conceitos acerca de design gráfico.

Nome do novo semestre: 2017/2018 - 2º Semestre

Password do novo semestre: pcm_3046

O novo semestre deve estar aberto para que os alunos se possam inscrever.
Nome do outro curso: VI

Nome do semestre: 2017/2018 - 1º Semestre

Nome do baralho: Multimedia Presentations

Nome do baralho clonado: Multimedia Presentations

O baralho deverá ser público.