rhythm CHOIR

Bachelor Thesis

Bachelor Industrial Engineering and Management

A Serious Game for Healthcare Professionals to Show the Capacity Pooling Effect

December 2024

Author

Marc Bovenhuis

Supervisors

Erwin W. Hans (University of Twente)

Florentina Hager (University of Twente)

UNIVERSITY OF TWENTE.

Colophon

This thesis is part of Module 12 of the Bachelor Industrial Engineering and Management.

Title

A Serious Game for Healthcare Professionals to Show the Capacity Pooling Effect

Author

Marc Bovenhuis
S1860410
m.d.bovenhuis@student.utwente.nl

Supervisors

Erwin W. Hans
Florentina Hager

University of Twente

Industrial Engineering and Management
PO Box 217, 7500 AE Enschede
+31(0)534899111
Faculty of Behavioral, Management and Social Sciences (BMS)
Industrial Engineering and Management

Date

11 December 2024

Report

Version 1
Number of pages: 69
Number of appendices: 3

Management Summary

Research Motivation

The increasing demand for care in combination with the limited capacity in healthcare, places increasing pressure on healthcare. This means that decision-makers in healthcare are forced to increase efficiency. CHOIR (Center for Healthcare Operations Improvement and Research), a research group of the University of Twente, in combination with its spin-off consultancy company Rhythm, which guides healthcare providers in improving their capacity management, aims to help these decision-makers. During their work, it became apparent that there are knowledge gaps among decision-makers. This led to a joint CHOIR-Rhythm professional education program for healthcare professionals. This research aims to facilitate this program with a serious game to educate decision-makers about the effects of capacity pooling in healthcare.

Approach

The first part of this research consists of two literature studies to gain the needed knowledge. First, we look into the principles of serious gaming, more specifically, those that apply to serious games for healthcare professionals. The second literature study looks at capacity pooling in healthcare. The literature studies lead to the design for a serious game that shows the effect of pooling interactively. The design is then converted into a serious game in Excel/VBA. We use this tool to develop a game script that guides the users of the game. This game script shows the effects of pooling in specific situations. Lastly, we conduct a classroom experiment to test the tool and the script.

Results

This thesis results in a serious game for healthcare professionals. The serious game teaches the user the effect of pooling queues on waiting time and teaches the user some useful lessons about capacity management in healthcare. The results of the classroom experiment showed that four of the six lessons were already well understood and based on the feedback the remaining lessons were adjusted and clarified.

Recommendations

Due to the necessary adjustments based on the feedback from the classroom experiment, it is recommended to conduct such an experiment again. So that the operation of the adjustments made, and the added elements and scenarios is tested. Finally, adding more specific interactive feedback could help further improve the learning experience.

Preface

Dear reader,

This thesis is the result of the last modules of the bachelor Industrial Engineering and Management at the University of Twente. The thesis was aimed at improving efficiency in healthcare and therefore suited me well, as this not only helps the companies but also the patients. During the long period of working on this thesis, there are several people who contributed to this result and whom I would like to thank for this.

First, I would like to thank my study advisor Cornelis ten Napel who was extremely important in getting started with the thesis. He was able to connect me with the right people and did everything in his power to make this thesis a success.

Secondly, I would like to thank my first supervisor Erwin Hans. He always knew how to stimulate and motivate me in the right way and gave progressive feedback to take the game and the report to a higher level. The boundless patience and personal approach also contributed to the writing of this thesis. In addition, a word of thanks to Florentina Hager for quickly being available as a second supervisor and for thinking along with us.

Finally, I would like to thank my friends, family and those previously mentioned for their patience and support during this process. It took longer than hoped and anticipated, but everyone kept believing and encouraging.

I hope you enjoy reading my thesis.

Kind Regards,

Marc Bovenhuis

December 2024

Table of Contents

Chapter 1 Introduction

This introduction chapter provides a motivation for this research (Section 1.1), a problem description (Section 1.2), the goal of the research (Section 1.3) and the research questions (Section 1.4).

1.1 Research Motivation and Background Information

1.1.1 Research Motivation

There is a growing demand in healthcare, while the supply side lags. The ageing population is causing an increasing demand, which inevitably leads to capacity problems in the healthcare sector (Helder, 2023). This problem does not only occur in the Netherlands, but worldwide (Gulumbe & Usman, 2023).

Healthcare systems worldwide are faced with continuously increasing demand for care, while simultaneously experiencing insufficient capacity and unacceptably long patient waiting times (Fagefors & Lantz, 2021). Waiting times in the Netherlands have increased by 3 percentage points in 2023, in 47,5 per cent of cases the agreed waiting time standard has not been achieved (CBS, 2024). While healthcare staff are already reaching their physical limits, partly due to the Covid-19 pandemic. This results in more burnouts and more healthcare personnel leaving the sector (Bishen, 2023).

It is important to make the best possible use of the available capacity. The Integral Healthcare Agreement (in Dutch “Integraal Zorg Akkoord”) in the Netherlands addresses the importance of increasing the (regional) cooperation in healthcare (van Volksgezondheid & en Sport, 2022). Healthcare usually works in a risk-averse manner, which makes It hard to implement new strategies. Besides that, hospitals are separate businesses with each their own financial targets, which leads to an extra obstacle to cooperation.

The increasing demand in healthcare, the increasing workload on healthcare staff and the increasing shortage of personnel make clear that changes are needed in the healthcare sector.

1.1.2 The CHOIR Research Group

This research is conducted on behalf of the Center of Healthcare Operations Improvement and Research group, from now on referred to as CHOIR. CHOIR is a Dutch research centre within the University of Twente. They help healthcare practitioners face their complex logistical challenges, through research, education and valorisation. One of the problems the CHOIR group is facing is to explain and convince decision-makers about the available proven solutions in operations research.

The CHOIR group, in cooperation with a supplier of consultancy services and software (ORTEC), started a spin-off company called Rhythm B.V. Rhythm makes acquired knowledge in the field of healthcare logistics more widely accessible, develops decision support tools, and guides healthcare providers in improving their capacity management.

The CHOIR research group in collaboration with Rhythm organizes professional courses for healthcare managers and clinicians, about (integral) capacity planning. From this, the need arose for a tool or serious game that illustrates one of the principles of capacity planning: the capacity pooling effect.

1.2 Problem Statement

As we mentioned in Section 1.1.1, changes are needed to maintain the current level of healthcare. In this section, we investigate the current problems of integrating capacity pooling in the healthcare sector in The Netherlands. We make use of a problem cluster to identify the core (red block) and action (green block) problem:

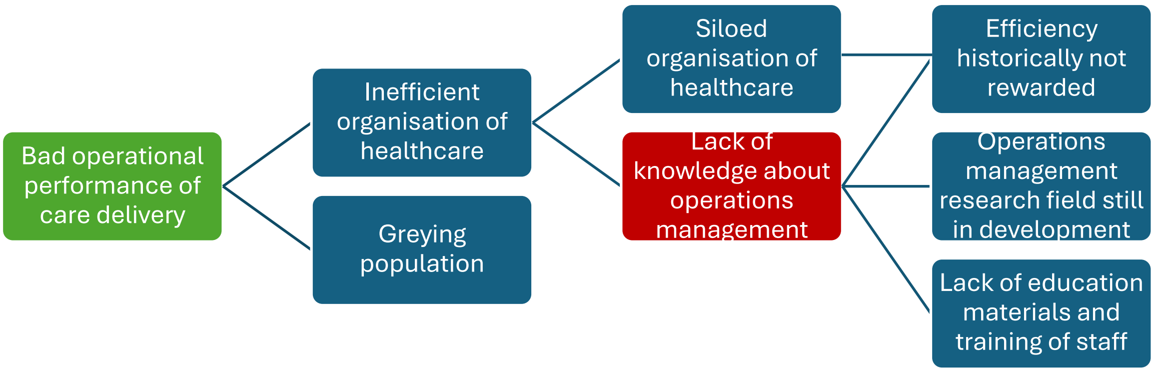


Figure 1: Problem cluster

The problem cluster shows that the core problem is a lack of knowledge about operations management in healthcare. Because efficiency in healthcare has not been rewarded for years, little attention has been paid to operations management in healthcare. With the introduction of the free market in healthcare in the Netherlands, hospitals were forced to be more efficient with their capacity. This was the start of more research into operations management in healthcare. In addition, this research needed to be translated into teaching materials for staff.

During the courses of CHOIR and Rhythm, the need arose for a tool that show the capacity pooling effect. This pooling effect covers a part of the knowledge about operations management. So, when we develop a tool with a game plan that shows decision-makers in healthcare the capacity pooling effect, we can help reduce the core problem.

An action problem is a discrepancy between the norm and the reality, as perceived by the problem owner (Heerkens & Winden, 2017). In this case, we are talking about decision-makers in healthcare who experience bad operational performance of care delivery. This causes, for example, higher waiting lists and workload.

1.3 Research Goal

The purpose of this research is twofold. First, we perform a literature study into serious gaming in the context of healthcare operations and the lessons that can be learnt from the pooling effect. Second, we develop a tool that shows the effect of pooling and create a serious game with this tool. The tool aims to inform both the healthcare professionals and the healthcare managers.

1.4 Research Questions

To accomplish the research goal, we conduct several research questions. By answering the following research questions, we aim to reach the research goal.

1. What serious games for healthcare operations management currently exist in the literature? (Chapter 2)

Chapter 2 dives into the literature and looks for existing serious games for the healthcare sector. The focus is on serious games regarding healthcare operations management. Furthermore, it gives a brief description of a serious game and the benefits and important characteristics of a serious game.

2. What lessons can be learnt from the pooling effect for capacity planning? (Chapter 3)

Chapter 3 gives insight into the basic theory behind the pooling effect. The focus is on the pooling effect for capacity planning. This literature study results in a list of important lessons that can be learnt from this effect.

3. What serious game is suitable to educate healthcare professionals various lessons regarding the pooling effect? (Chapter 4)

Chapter 4 shows the building process of the serious game in Excel/VBA. The lessons from research questions two and three are the start of the process and the game is built around them. The end product is a serious game made in Excel.

4. What is a suitable gameplay protocol for classroom settings? (Chapter 5)

Chapter 5 leads to a gameplay protocol for the serious game. The goal of this protocol is that the lessons of research question three are visually visible and easy to understand.

5. What improvements can be made to the serious game and the gameplay protocol? (Chapter 6)

Chapter 6 consists of 2 experiments with the serious game and the protocol. Based on these experiments, the serious game and the protocol are further improved.

Chapter 2 Serious Games for Healthcare Professionals

This chapter aims to answer the research question about serious gaming through available literature studies. Section 2.1 introduces the reader to the basic principles of learning and serious gaming and forms a definition. Section 2.2 investigates comparable serious games for healthcare professionals and Section 2.3 gives the conclusions.

2.1 Introduction to Serious Gaming

This section starts with a definition of serious gaming. This definition is based on different definitions of serious gaming found in the literature. In addition, an explanation is given as to why a serious game is suitable in this case.

2.1.1 Definition of Serious Games

Serious games are “games that do not have entertainment, enjoyment or fun as their primary purpose” (Michael, D. R., & Chen, 2005). These games contain a double mission, simultaneous achievement of intended effects (serious part) and entertainment (game part) (Caserman et al., 2020). Giessen (2015) states that it is widely known that fear, stress - or also boredom, for that matter - activate the amygdala, whilst knowledge and information connected with positive emotions is absorbed by the hippocampus and then transferred to the cortex for further processing. Thus, learning content should be prepared in a way that activates the hippocampus, not the amygdala. So, it should be transmitted in a somewhat pleasant way that evokes interest and positive emotions.

The literature study of Laamarti et al. (2014) compared several definitions and looked for similarities. The common parts of the definitions and their overlap are shown below:

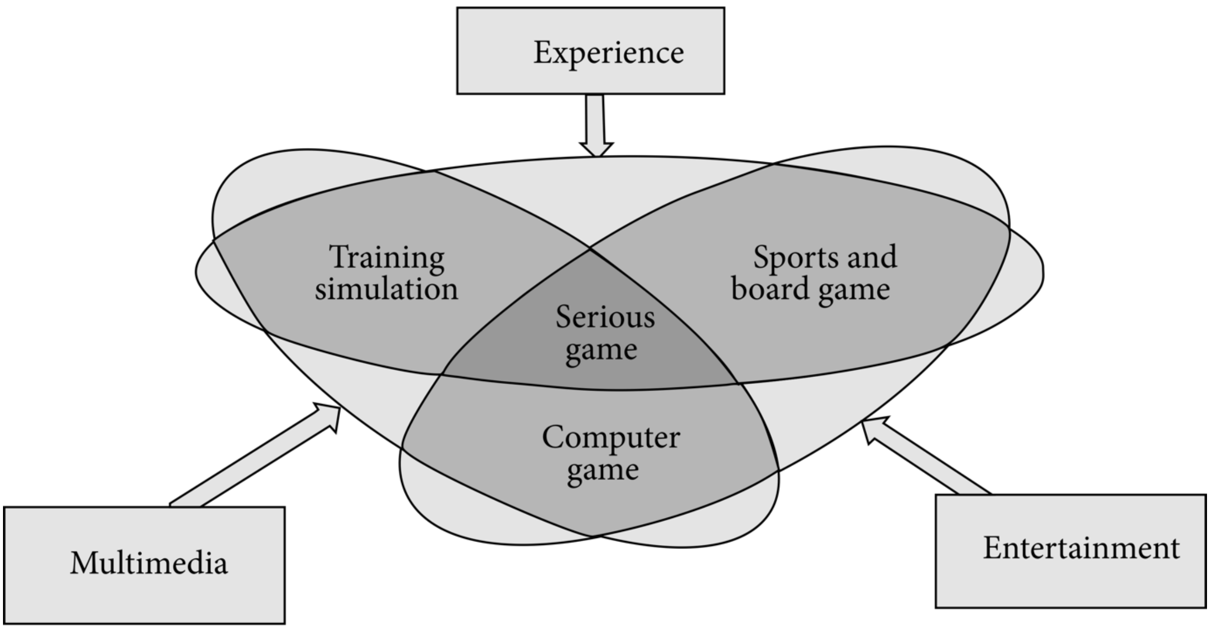


Figure 2: Visualisation of the definition of serious games

Hence, they define serious games as an application with three components: experience, entertainment, and multimedia. Here, multimedia refers to the use of different media, such as text and graphics. Entertainment refers to the "fun" part of games and experience refers to the serious part of serious games. "The "serious" term in serious games comes from their role of conveying some message or input, be it knowledge, skill, or in general some content to the player" (Laamarti et al., 2014). So, we define serious games as an application with three components: experience, entertainment and multimedia.

2.1.2 Basic Principles of Learning

To know the benefits of a serious game, it is important to first cover the basic principles of learning. Cognitive science has identified four main success factors in learning: attention, active engagement, feedback of information, and eventually, consolidation (Dehaene, 2013).

2.1.2.1 Attention

Attention massively modulates brain activity: therefore, the key issue for the purveyor of knowledge – whether parent, teacher or trainer – is to direct attention on the "right level" (Dehaene, 2013). So, the attention should be on the parts you want to convey.

2.1.2.2 Active Engagement

A passive organism does not learn. As Dehaene (2013) describes in his article, active engagement is necessary to learn. To achieve this, it is good to test whether the right knowledge has been acquired during the process.

2.1.2.3 Feedback of Information

The main lesson from feedback of information is that making mistakes is essential for learning. On the other hand, punishing mistakes too harshly is a reason for learning to be worse. This way of learning is also reflected in the article by Swann (1999), who states that encountering problems and making mistakes is essential to arriving at a good solution. He clarifies this with the following formula: . Where P_1 is the initial problem, TS is the trial solution, and this trial solution is subject to error elimination (EE), which leads to the next problem P_2. Swann (1999) states that the change that occurs in ourselves and the world around us while going through this process, is learning.

2.1.2.4 Consolidation

Continuing to repeat and apply what has been learned. This is the best way to remember it for a longer period.

These four factors are reflected in different theories in different ways. Kolb and Fry were one of the first to come up with factors like the above (1974). He designed a reflective cycle that consists of four stages.

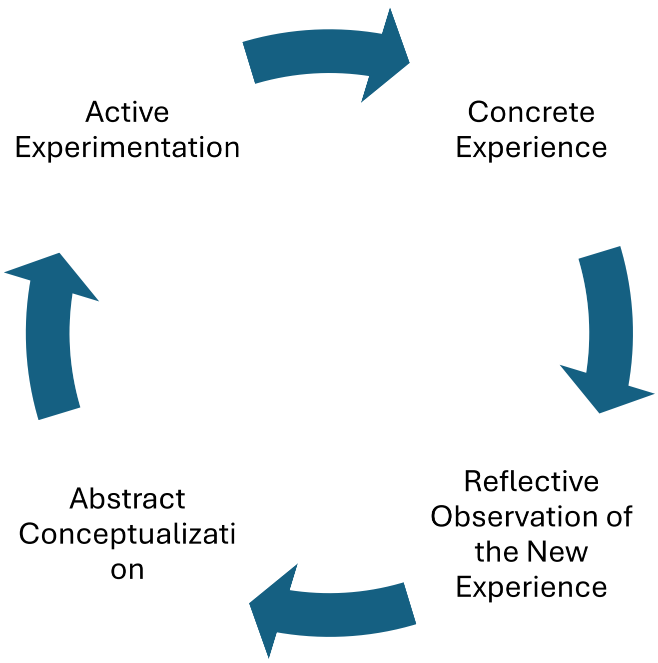


Figure 3: Kolb's Reflective Cycle, based on (Kolb & Fry, 1974)

Just as in Dehaene's theory, Kolb's theory also shows that active learning, processing feedback and repetition are important. So, while making a serious game these four factors should be kept in mind.

2.1.3 Benefits of a Serious Game

After examining the definition of serious games and the basic principles of learning, we look at the benefits of a serious game. Many studies have been conducted on the benefits of serious games and this section highlights the most important conclusions of these studies.

2.1.3.1 Enhanced Learning and Knowledge Retention

The study of Wouters et al. (2013) used meta-analytic techniques to investigate whether serious games are more effective in terms of learning and more motivating than conventional methods. They examined 39 studies and compared the outcomes in the areas of motivation, learning and retention. Several conclusions emerged from this research. They concluded that instruction with serious games yields higher learning gains than conventional instruction. In addition, they proved that serious games were better in terms of knowledge acquisition, cognitive skills and learning retention (Wouters et al., 2013).

A similar literature study that compared the use of a serious game with conventional methods is that of Sitzmann (2011). This study showed that the group that played the serious game had acquired significantly more knowledge and retained this knowledge better compared to a comparison group that used regular methods such as lectures and reading (Sitzmann, 2011). In addition, it emerged that the game worked better when used as a tool rather than when it was used as a standalone game, and a higher entertainment level had no significant added value.

2.1.3.2 Collaboration and Teamwork

The literature study of Connolly et al. (2012) provides the broadest possible investigation into the common benefits of digital games. They investigated 129 papers, of which twelve were specifically about a serious game. The results of this study indicated that the main benefits of digital games were higher motivation and better understanding and acquisition of knowledge. In addition to the benefits described above, serious games also showed benefits in terms of social and soft skills (Connolly et al., 2012). This is because serious games require more collaboration than traditional ways of learning.

This finding was also reflected in the study of Khan et al. (2020). They concluded that serious games are effective in improving cognitive ability and power. The use of serious games in learning was also effective in socio-cultural learning in contexts of cognitive and uplifting effects (Khan et al., 2020).

2.1.4 Suitability Serious Game

Implementing new strategies is difficult because people within the organization must be convinced of the new strategy. "The most dangerous phrase in the language is, 'we've always done it this way.'" (Hopper, 1976). This phrase is often used when decision-makers try to implement a new planning strategy. So, it is important to convince not only the decision-makers but also the main users of the planning strategy, the healthcare professionals.

The core problem of this study is the lack of knowledge of capacity pooling in healthcare. As mentioned in Section 1.1.2, there is a knowledge gap about the capacity pooling effect among decision-makers. A serious game makes the pooling effect visible to the decision-makers and healthcare professionals, which leads to a better understanding and greater success when implemented. Since, the successful implementation of a new planning strategy partly depends on the conviction and experience of the decision-makers and users regarding this strategy (Pelaksanaan et al., 2019).

2.2 Serious Gaming in Healthcare

Serious games and gamification techniques are increasingly being used for a wide range of lessons within healthcare (R. Wang et al., 2016). The literature study of van Zyl-Cillié (2023) aims to determine which classroom games in ORMS (Operation Research and Management Science) could be adapted for teaching ORMS as applied to healthcare. None of the 26 games that were evaluated, applied to healthcare. However, six of the 26 games, seemed to be adaptable to a healthcare application. These games were not specifically made for one specific environment and could be used to explain some principles of ORMS in healthcare. Finally, they conclude that there is a significant research opportunity to develop serious games in ORMS, particularly for application in the healthcare environment (van Zyl-Cillié, 2023).

Serious games in healthcare are currently mainly used to train medical staff. Especially during the COVID-19 period, digital serious games were helpful and needed to train healthcare professionals. Since on-site training was not possible due to restrictions (Montalbano et al., 2022). The serious games used during this period proved to be useful for the nursing staff and nursing students (Calik et al., 2022). Partly because of this, the serious games market is expected to grow 18,74% per year until 2030 (*Serious Games Market Size, Share | Industry Forecast - 2030*, 2022).

2.3 Conclusions

This chapter provides the basics of a serious game and learning principles. We define serious games as an application with three components: experience, entertainment, and multimedia. The combination of these three components ensures better processing of knowledge, as the knowledge is transferred to the cortex via the hippocampus.

The four factors that are important for good knowledge absorption are: attention, active engagement, feedback of information and consolidation. A serious game offers a good opportunity to apply all these factors. A serious game attracts attention because it is a new method and can graphically attract attention. It ensures active engagement because the user must play the game themselves and cannot sit back. You get direct feedback when you do things wrong, so you learn more from it. And you can keep playing and trying, so you remember it longer.

There are few to no serious games available in ORMS specifically aimed at healthcare. The serious games that are available for healthcare so far are mainly intended for the practical training and education of nurses. So, since a serious game is a good way to learn and understand new theories and there is a lack of serious games in our research field, a serious game seems to be the ideal opportunity for our final product.

Chapter 3 The Pooling Effect

This chapter introduces the reader to the effects of pooling and explains in which situations pooling can or cannot be beneficial, done through a study within the available literature. This chapter starts with an introduction to the pooling effect in general (Section 3.1) and pooling in healthcare (Section 3.2). Section 3.3 addresses the question of whether pooling is always beneficial, and Section 3.4 examines whether an intermediate solution is also possible.

3.1 Pooling

Pooling of customer demands, along with pooling of the resources used to fill those demands, may yield operational improvements (Cattani & Schmidt, 2005a). This definition of the pooling principle is most common and is used during this research. During this chapter, the focus is on pooling capacity of healthcare providers.

3.1.1 Queueing Theory

To illustrate the advantages and disadvantages of pooling, we need a notation to show the characteristics of the queueing system. We use Kendall's notation of the queuing theory. Kendall originally proposed describing queueing models using three factors, A/B/C (Kendall, 1953).

- A, The arrival process, the time between arrivals to the queue
- B, Service time distribution, distribution of time of the service of a customer
- C, number of servers

Where the most common distributions for A and B in this notation are Markov and General.

- M, Markov, exponential distribution
- G, General, general distribution

Years later, in 1966 A. M. Lee added 2 more factors and in 1968 Hamdi A Taha added the last factor (Taha, 2007). As a result, the current notation consists of 6 factors: A/B/C/D/E/F.

- D, The number of places in the queue, the capacity of the queue
- E, The calling population, the size of the population from which the customers come
- F, The queue's discipline, the service discipline or priority order

The above notation helps to show the pooling effect in 3.1.3.

3.1.2 Little's Law

To express the effect in figures we need another formula to be able to tell more about the average number of customers in the queue and the waiting times. The most common and easiest formula to do so is Little's law. Little's Law says that, under steady-state conditions, the average number of items in a queuing system equals the average rate at which items arrive multiplied by the average time that an item spends in the system.

- L = average number of items in the queuing system
- W = average waiting time in the system for an item
- λ = average number of items arriving per unit time

The occupation rate per server can be calculated with the following formula:

- c = Number of servers
- μ = Service rate

3.1.3 The Pooling Effect

There are two basic queue structures commonly adopted in service systems: the pooled structure, where waiting customers are organized into a single queue served by a group of servers, and the dedicated structure, where each server has its own queue (Cao et al., 2020). The easiest way to show the benefits of pooling is by using a service system with two parallel servers. Where one system uses two dedicated queues for two servers.

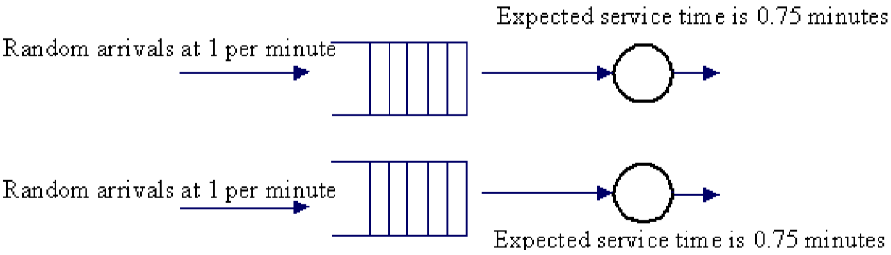


Figure 4: Two single-server systems (Cattani & Schmidt, 2005b)

Figure 4 shows that the arrival rate (λ) is 1 customer per minute and the expected service time (S) is 0,75 minutes. In Kendall's notation represented as M/M/1. This results in a service rate (μ) of 1,33 per minute. All formulas used in this section are derived from chapter eleven of (Slack et al., 2013).

Using Little's Law, the expected waiting time (W) for each single server is.

The expected waiting time is the complete time waiting in the system, for this example it is more useful to look at the waiting time in the queue. The waiting time in the queue can be calculated by the formula: . The expected waiting time in queue is therefore:

This is the waiting time in queue per server with each server having its own queue. Now if we still use 2 servers, but merge the queue, we can see if pooling has any effect in this situation.

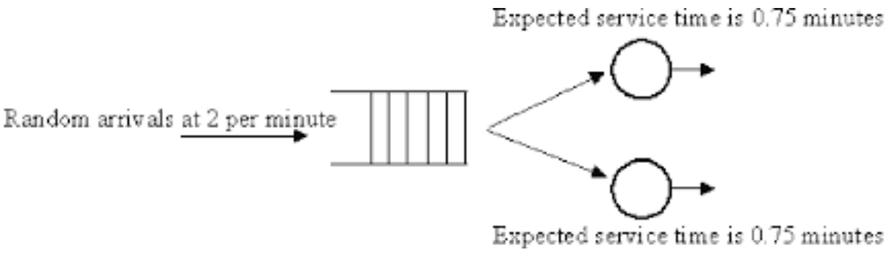


Figure 5: A pooled system (Cattani & Schmidt, 2005b)

The above system is not a M/M/1 but a M/M/2 system, since there is one queue for two servers. This requires a different formula to calculate the waiting time in line.

Where , so

So, in this example, the average waiting time in queue is lowered from 3 minutes to just 0,99 minutes. This change in waiting time has been achieved purely by pooling the queues.

3.2 Introduction to Pooling Effects in Healthcare

Pooling is already a well-known concept in the healthcare sector. The concept of pooling is mainly used by insurance. Insurances use risk pooling to keep costs low and keep care available to as many people as possible. The concept of pooling is therefore not new. However, in this chapter, we look for lessons that can be learned from pooling capacity within healthcare.

Capacity pooling in healthcare can be divided into three types:

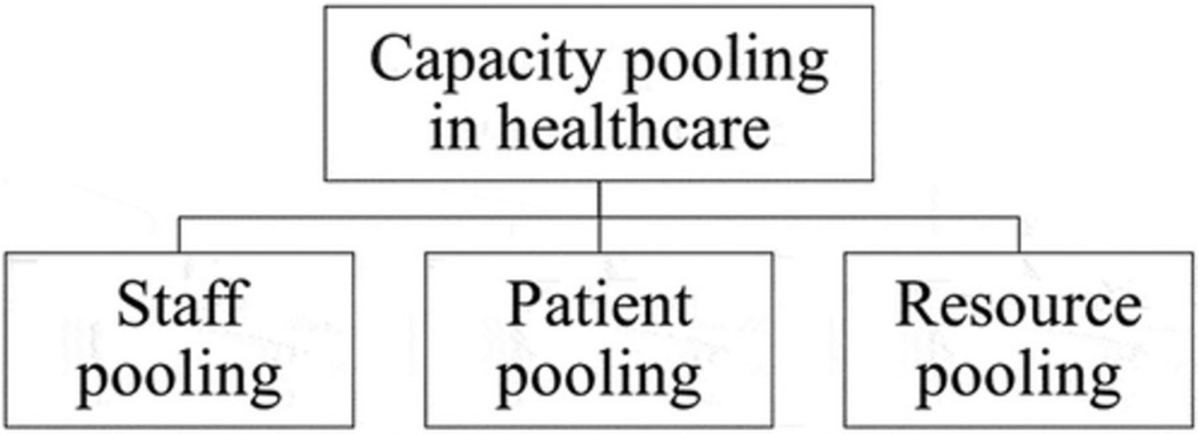


Figure 6: Three types of capacity pooling (Fagefors et al., 2024).

3.2.1 Staff Pooling

Staff pooling refers to allocating staff to where resources are needed. Allocating staff to places in the hospital where demand is high would be a great solution to absorb the fluctuations in demand. However, this is more difficult to achieve in practice. The complicated thing about pooling staff is that there is more and more specialisation in healthcare these days. This specialisation makes it difficult and sometimes even impossible to pool staff.

3.2.2 Patient Pooling

Allocating patients to where resources are available. With this type of pooling, a group of patients is pooled and can be allocated to the first available healthcare professional or bed. This leads to shorter waiting times but can also lead to higher stress levels for healthcare professionals. Since the nurse or doctor may be specialised in other patient groups (Song et al., 2020). This mismatch in specialisation leads, on some occasions, to a longer length of stay for patients.

3.2.3 Resource Pooling

Organising resources (other than staff), for example, hospital beds, to pre-determined units. Pooling hospital beds creates more flexibility in placing patients across the different wards. The tricky thing about this form of pooling is that it can lead to a mismatch between patient type and the nurse's specialisation.

As can be read above, there are many forms of pooling possible within healthcare. However, pooling is more difficult since hospitals traditionally segregate resources into centralized functional departments such as diagnostic departments, ambulatory care centres, and nursing wards. These specializations make pooling more difficult. In recent years, this organisational model has been challenged by the idea that higher quality of care and efficiency in service delivery can be achieved when services are organised around patient groups (Vanberkel et al., 2012). In this way, groups of patients are pooled, but this requires a major change within traditional healthcare.

3.2.4 Psychological Aspects

An often-underestimated factor of waiting for an appointment is the fairness perception of having to wait. “The only thing worse than waiting in line is waiting in the wrong line”. When people feel that they must wait longer than others, they get a feeling of injustice. This causes the perception of waiting time to be longer, even though it may have been less mathematically. In healthcare, this principle is reflected in people waiting together in a waiting room. It is important that not only the shortest waiting time is considered, but in some cases, the empirical side should also be considered (Maister, 1984).

3.3 Pooling Always Beneficial?

The previous sections show that pooling has many advantages. To illustrate the limitations of pooling, we also highlight situations in which pooling is not beneficial. The best-known example of this is pooling in a call centre. The psychological aspect is of secondary importance here, as customers cannot see whether pooling is taking place or not (N. M. Van Dijk & Van Der Sluis, 2008).

N. M. Van Dijk and Van Der Sluis (2008) have a deeper look at a call centre situation where two different types of customers arrive, with each a different arrival time and service time. This variance in arrival time and service time leads to a longer waiting time when the queues of both customer types are pooled. Figure 7 shows that pooling the queues has a negative influence on the average waiting time. In the unpooled system the average waiting time is 4,55 minutes, but when the queues are pooled the average waiting time increases to 6,15 minutes.

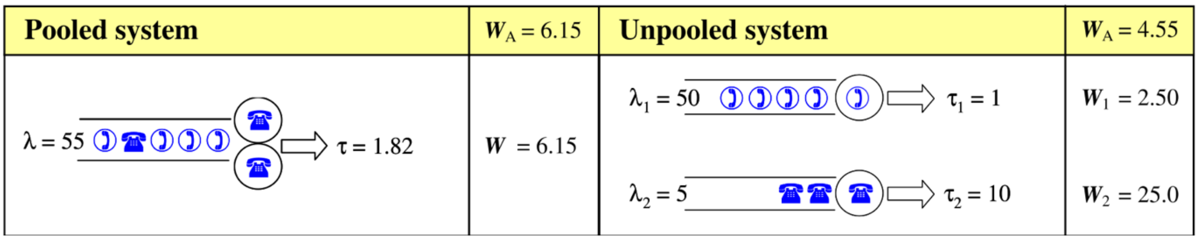


Figure 7: Scenario comparison (N. Van Dijk & Van Der Sluis, 2008)

This is a rather extreme situation, and in most cases, pooling leads to a shorter average waiting time. However, it does show that pooling is not always the solution and that it cannot be applied in every situation without a good calculation or simulation.

Besides the fact that it is not always beneficial in terms of waiting time, it can also be detrimental to the quality of care. Pooling beds within a hospital leads to a longer length of stay and more complications during this period for patients who are placed outside their speciality wards (Song et al., 2019). But it also harms patients who are in the right ward (Lim et al., 2024). These patients had to deal with longer lengths of stay because doctors and nurses had to move between multiple departments. So, it is important to look at all these factors carefully before pooling.

So, this section shows that pooling is not always beneficial. But this only applies to cases with strongly varying service times. In cases with equal service times or slightly different ones, pooling is beneficial.

3.4 Shared Pools

Pooling of queues or capacity can work out well (Section 3.1) or poorly (Section 3.3). In these cases, the focus was on complete pooling. In this section, we look at shared pooling or partial pooling.

Wallace and Whitt (2004) show that a little flexibility (i.e., a little pooling) can possibly "go a long way" in a queueing setting (Cattani & Schmidt, 2005c). As mentioned at the end of Section 3.3, pooling hospital beds leads to a longer length of stay. This is mainly because specialised personnel must go to another department, or the patient is treated by personnel who are not specialised in his health problems. This problem could be tackled by creating a pool of nurses who have broader training and can assist in multiple departments. These so-called float pools are difficult to create as they are expensive, and the staff is scarce. A solution to this is a clustered pool where there are multiple pools of nurses who can work in a few different wards. This solution is not cost-efficient and is therefore often not chosen (Dziuba-Ellis, 2006).

Another solution to this is to use pools, which can be used by multiple hospitals or locations (Fagefors et al., 2020; Morris, 2021). A recent example is Kimberley Morris' thesis, in which she uses a flex pool for nurses. With a small flex pool of two people, a reduction of 25 to 30 per cent in patient movement could be achieved (Morris, 2021). This shows that with a little bit of flexibility and pooling, significant profits can be made.

3.5 Conclusion

This chapter shows that pooling in a homogeneous situation often leads to capacity advantages. However, when applied incorrectly, or in a situation with a lot of variation, it can work counterproductive.

Section 3.2 went into more detail about the 3 different types of capacity pooling in healthcare: staff pooling, patient pooling and resource pooling. This showed that the specification in healthcare makes it difficult to pool. So, for an efficient way of pooling, it must first be examined whether it has an advantage in these specific cases.

Creating a serious game that shows the effect of pooling capacity contributes to the knowledge about pooling. But it can also contribute as a simulation to help determine whether pooling is a good idea.

Chapter 4 Development of the Serious Game

This chapter gives insights into the development of the serious game. The goal is to create an interactive tool, which is part of the serious game that makes the pooling effect visible. Section 4.1 presents the learning objectives of the serious game. Section 4.2 discusses the design approach. Section 4.3 presents the game simulation design and Section 4.4 the layout design. Lastly, Section 4.5 shows a few additional features.

4.1 Learning Objectives of the Serious Game

Following the literature study in Chapter 3 and after consultation with the company, we formulated learning objectives for the player, which form the basis of the serious game. Each goal is briefly described and explained below.

4.1.1 Pooling

The main goal of the serious game is to show the pooling effect. We aim to include three learning objectives:

- In queues where the treatment time of patients does not show a strong variation, pooling ensures a reduced waiting time.
- In queues where the treatment time of patients distinctly differs, pooling may be disadvantageous.
- ‘A little flexibility goes a long way’: not only by pooling complete queues can there be a gain in waiting times. By partially pooling queues, there will already be significant advantages in waiting times.

4.1.2 Occupancy/Utilisation Rate

Utilisation of shared resources has long been an important performance indicator in healthcare. The assumption was that if resources are utilised well, income is high, costs are covered and waiting time is low. Whenever consequently the work pressure of staff was perceived as too high, care providers would strive to increase capacity. Two of the serious game’s learning objectives relate to the occupancy/utilisation rate:

- Increasing the occupancy rate causes an exponential growth of the waiting time and increases workload fluctuation (Stephens & Broome, 2019). Increasing the occupancy rate thus increases resource efficiency at the expense of staff work pressure and patient waiting times. A balance must therefore be found between these.
- A waiting list can be dissolved quickly by making sure there is on average excess capacity, in other words, by setting the utilisation rate not too close to 100%. This can be done by temporarily increasing capacity, or by temporarily reducing new demand. This learning objective also implies that if the waiting list does not structurally increase, there is sufficient capacity.

4.1.3 Variability

The final learning objective is that variability is the cause of exponential growth in waiting time and higher workload fluctuations. It does not matter whether the variability is in demand or supply.

4.2 Design Approach

This section goes into more detail about the chosen method for the serious game. It shows why simulation is the best option and which application is used for this. Finally, it discusses the different scenarios of playing.

4.2.1 Simulation

The best and easiest way to visualize the pooling effect is through simulation. Since we discussed in Chapter 2 that the multimedia and visual aspects of a serious game are important, a serious game based on simulation is inevitable. Simulation is already increasingly used in healthcare and with great success. As described earlier, most simulations focused on medical interventions. Nevertheless, the effect of simulations during the training of healthcare personnel and students is considerable (Marion-Martins & Pinho, 2020).

The most widely used simulation method in healthcare is discrete event simulation, followed closely by Monte Carlo simulation (Wang & Demeulemeester, 2023). 'A Discrete Event System is a system where state changes (events) happen at discrete instances in time, and events take zero time to happen' (Varga, 2005). Our game has more variation, which makes discrete event simulation not the most suitable option. The principle behind Monte Carlo simulation is that the behaviour of a statistic in random samples can be assessed by the empirical process of drawing lots of random samples and observing this behaviour (Mooney, 1997). The strategy for doing this is to create a pseudo-population, which resembles the real world in relevant respects. So, for the serious game, the Monte Carlo simulation is the best option, as it comes closest to reality.

4.2.2 Application

To keep the serious game widely applicable, we make sure that it can be played by many users. An important part of this is that the application in which the simulation is played should cost as little as possible. Partly because the game will only be used for a short period. This leaves one serious option, Excel/VBA. Almost all healthcare institutions have the application and are used to working with it. This makes it easier to play the game. In addition, the past has proven that Excel is a good option for a Monte Carlo simulation (Botchkarev, 2015).

4.2.3 Playing Style

The game is played through four scenarios. The first scenario is a model that is as simple as possible to illustrate the pooling effect. Only arrival and capacity per hour are used and the waiting list is calculated based on this. The user adjusts a parameter per step and this ties in with at least one of the learning objectives from 4.1.

The second scenario contains more variability because now two patient types are used. This comes closer to reality and in this scenario, the user is asked again to adjust parameters step by step and learn the learning objectives from 4.1.

Scenario 3 is similar to Scenario 1 in many ways but with one important difference: the distribution on which both demand and capacity are determined. Scenario 3 does not use a uniform distribution, but a truncated normal distribution. All other calculations are identical.

Scenario 4 is a completely new scenario. This scenario generates a waiting list with two types of patients. The waiting list is then scheduled based on two different strategies. The first strategy does not differentiate between the two types of patients. The second does differentiate between the patients on the waiting list and refers the fast patients to a fixed server.

4.3 Game Simulation Design

As described above, the game consists of four scenarios. Their construction and calculations are described below.

4.3.1 Scenario 1

Scenario 1 is the simplest of the two and uses the uniform distribution.

4.3.1.1 The Base

The basis consists of input boxes where the user can decide the number of servers and hours per server that will be simulated. These servers could serve as, for example, operating rooms, outpatient rooms or nursing wards. The next input box indicates how many hours are simulated per server. With these input values, the number of tables is known (number of servers) and the number of rows per table is known (number of hours per server).

4.3.1.2 Uniform Distribution

A straightforward option is to determine the demand and supply using a uniform distribution. The user must enter four values, the minimum and maximum value of both the demand and the supply (capacity) per hour. A randomizer is used to draw a number between the minimum and maximum value every hour, resulting in the supply and demand values.

Uniform distribution	From	Till
Demand per hour	8	13
Capacity per hour	9	15
Number of servers	3	
Hours per server	40	
Flexible capacity per server	0	
Initial waiting list	0	
Current occupancy	87,5%	

Figure 8: Screenshot input boxes uniform distribution

4.3.1.3 Served, Waiting List and Unused Capacity

With the completed supply and demand parameters together with the initial waiting list, the number of patients served, the unused capacity, and the waiting list can be calculated. Figure 9 shows an example of a table showing the results of these calculations.

Served

The number of patients served is calculated based on the following formulas:

Waiting List

The waiting list for the first hour per server is zero unless the user has set an initial waiting list. In that case, the waiting list in the first hour is equal to the initial waiting list. The waiting list for the subsequent hours is calculated as follows:

Unused Capacity

If more capacity is available than is being used, there will be unused capacity, which is calculated as follows:

Hours	Demand	Capacity	Served	Waiting List	Unused
1	10	15	10	5	5
2	11	12	12	4	0
3	10	9	9	5	0
4	10	12	12	3	0
5	8	14	11	0	3
6	13	15	13	0	2
7	12	12	12	0	0
8	13	14	13	0	1

Figure 9: Screenshot data table scenario 1

4.3.2 Scenario 2

Scenario 1 provides two options to calculate demand and capacity, these options are based on one patient type and do not consider different treatment times. To provide even more options and flexibility, we added a new tab called 'Scenario 2'. This scenario still uses the same base in terms of number of servers and hours. However, supply and demand are calculated differently. The user fills in the minimum and maximum treatment time in minutes of two different types of patients and the ratio between these types. In addition, he determines the maximum capacity available per hour and the average occupation and variability per doctor (see Figure 10).

Treatment time minutes	Van	Tot
Patient type short	10	15
Patient type long	15	25
Capacity minutes		
Max per hour	60	
Average physician occupancy	85	
Variability	5	
Percentage of long-term patients	40	
Number of servers	3	
Hours per server	40	
Flexible capacity per server (min.)	0	
Initial waiting list	0	

Figure 10: Screenshot input boxes Scenario 2

The demand is filled with patients in the indicated ratio until the maximum capacity is exceeded. As soon as the capacity is exceeded, the last patient causing the excess is removed. This means that the demand always remains below the available capacity. This can be compared to filling an agenda. So, if a doctor had a 100% occupancy rate, there would be no waiting lists.

It is not realistic for a doctor to have a 100% occupancy rate. Doctors are regularly called away for emergencies and some patients do not show up. The game user enters the average occupation of the doctor and the variability. The game randomly draws an occupation rate that falls within that range, and this determines the capacity. The number of patients served, the waiting list, and the unused capacity are calculated in the same way as Scenario 1.

To keep track of both patient types, two additional columns appear. These columns show the number of short-term patients and long-term patients. These are counted while executing the code. Figure 11 shows an example of this table.

Hours	Demand	Capacity	Served	Waiting List	Unused	#Short	#Long
1	51	48	48	3	0	2	2
2	47	53	50	0	3	2	2
3	48	52	48	0	4	2	2
4	58	51	51	7	0	2	3
5	54	52	52	9	0	4	1
6	39	53	48	0	5	1	2
7	57	53	53	4	0	1	3
8	59	54	54	9	0	3	2

Figure 11: Screenshot data table Scenario 2

4.3.3 Scenario 3

Scenario 3 uses the same sheets as Scenario 1 and uses the calculations from 4.3.1. However, there is one major difference and that is the distribution on which demand and supply are based. Previous scenarios use a uniform distribution, scenario 3 uses a truncated normal distribution. The normal distribution often comes closer to the real situation but will not be understood by all users. Hence, the possibility for both a uniform distribution and a normal distribution. Once Scenario 3 is selected, the input values for demand and capacity change. It now asks for a mean and standard deviation, see Figure 12.

Normal distribution	Average	Standard deviation	Scenario 3
Demand per hour	10	3	
Capacity per hour	12	2	
Number of servers	3		
Hours per server	40		
Flexible capacity per server	0		
Initial waiting list	0		
Current occupancy	83,3%		

Figure 12: Screenshot input boxes Scenario 3

The determination of the demand and capacity is done using a function in VBA. This function uses the Norm.Inv formula in Excel, where the mean and standard deviation come from the input values from Figure 12. The probability is determined by VBA by taking a random value between 0 and 1 every time the function is used. However, this function could return values below zero. It is not possible to have a value for demand or capacity that is less than zero, so when a value below zero comes out of the function, the function is called again until it returns a value greater than or equal to zero. The final output of the function is rounded to a whole number because in this scenario we assume a number of patients and this cannot be a decimal number. Figure 13 shows the code of this function.

```
Function DrawNormalRounded(mu As Double, sigma As Double) As Double

DrawNormalRounded = Round(WorksheetFunction.Norm_Inv(Rnd(), mu, sigma), 0)
Do While DrawNormalRounded < 0
   DrawNormalRounded = Round(WorksheetFunction.Norm_Inv(Rnd(), mu, sigma), 0)
Loop

End Function
```

Figure 13: VBA code of function truncated normal distribution

4.3.4 Scenario 4

Scenario 4 uses a different approach. In this scenario, the user fills in multiple values, which generates a waiting list. Figure 14 shows a screenshot of all the input values, the meaning of these values is explained below the figure.

Treatment time minutes	From	To	Scenario 4
Patient type short	5	10	
Patient type long	40	45	
Capacity minutes			
Max per hour	60		
Working hours per day	8		Advice B8:
Percentage of long-term patients	27	%	27
Server occupancy rate	90	%	
Number of servers	3		
Number on waiting list	5000		

Figure 14: Screenshot input boxes Scenario 4

Patient types short and long

The entry values for short-term and long-term patients determine the treatment time. In this case, the treatment time of the short-term patients is a random number between 5 and 10. This treatment time is determined by a uniform distribution.

Capacity

The value for 'Max per hour' indicates how many minutes per hour may be scheduled. Together with the number of hours per day and the number of servers, they determine the maximum capacity per day.

Waiting list

The percentage of long-term patients determines what percentage of the waiting list consists of patients with type long. Next to the input field for this percentage is a recommendation for this percentage. This recommendation is calculated based on the entered values with the formula described below. This formula aims to ensure that a third of the capacity is used by short-term patients.

$$UP$$

The value for number on waiting list determines the total number of patients on the waiting list and the server occupancy rate will be discussed later.

Schedule per day

After the waiting list has been generated, it is checked how many patients fit per day. The waiting list is checked per patient and fills day 1 until it reaches the maximum capacity per day. At that moment, 4 more places on the waiting list are checked to see if those patients would still fit in day 1, if this is the case they are still assigned to day 1. The patients who do not fit are assigned to day 2 and day 2 is then filled again until the maximum capacity is reached again. This process repeats itself until all patients on the waiting list have been assigned a day. This designated day is considered the arrival day.

Schedule per server

The patients scheduled on day 1 are distributed over the three servers based on two strategies, a pooled situation and a split situation.

Pooled situation

The pooled situation does not distinguish between servers or patients. It fills the servers based on the waiting list and starts by filling Server 1. The servers are filled until the capacity per server is full, the formula for this is below the text. The code checks for each patient which day of arrival he has and then tries to place him in Server 1 for that day. If this does not fit, the patient is placed in Server 2. If this also does not fit, the patient is placed in Server 3 and if that also does not fit, he tries the same with the next day starting with Server 1 and so on. Each patient is therefore first tried to be placed in Server 1 on the day of arrival.

The difference between the actual day of treatment and the day of arrival is the number of days the patient was treated later. The number of days the patient was treated late is noted in a separate column. At the end of the day, this column shows the sum of the total number of days that appointments of that day were postponed. The column next to it shows how many patients who were scheduled for that day were not helped that day. The sum of the postponed days divided by the number of postponed patients on that day gives the average number of days that must be waited per postponed patient.

Split situation

The split situation does make a distinction, Server 1 only treats short-term patients and the long-term patients are divided between Servers 2 and 3. Each short-term patient is first tried to be placed in Server 1 on the day of arrival, if this is full, they are placed in Server 1 of the next day and if this is also full, in Server 1 of the day after that.

The long-term patients are placed in the same way as in the pooled situation. However, there are only 2 servers available and only the long-term patients are placed. For each long-term patient, an attempt is made to place him in Server 2 on the day of arrival, if this does not fit, then in Server 3 on the day of arrival and if this does not fit either, then in Server 2 of the next day, and so on. In this situation, the number of days the appointment is postponed and the number of patients postponed are also recorded.

4.3.5 Learning Objectives Scenario 1 to 3

The Key Performance Indicators (KPIs) that are needed to achieve the learning objectives are the waiting list in the standard situation, in the pooled situation and in the situation when there is partial pooling. By comparing the waiting list in all three situations, the effect of pooling and the effect of the utilization rate can be determined. The calculation for the standard waiting list can be found in 4.3.1.3. The calculation for the other two waiting lists is shown below.

Complete Pooling

The first learning objective shows the effect of pooling all servers. The demand and capacity of all servers per hour are added together on a new tab. Based on this new demand and capacity the number of patients served, the unused capacity and the waiting list are recalculated in the way as described in 4.3.1.3.

Partial Pooling

The next part of the learning objective focuses on partial pooling. The user chooses a certain amount of capacity that is available to pool. In the variant with one patient type, this concerns the number of patients of the available capacity that may be pooled. In the variant with two different patient types, a portion of the time (expressed in minutes) of the available capacity may be used for pooling. This available flexible capacity can be used by other servers.

This flexible capacity is tracked in two new columns, capacity pool and served pool. First, the available flexible capacity per server is calculated. All flexible capacity is collected in a so-called capacity pool. This is done using the following formula, where flexible capacity equals the value entered by the user.

The served patients out of the capacity pool per hour is calculated by the following formula.

After using the above formulas, the unused pool capacity can be calculated by subtracting the served pool from the capacity pool.

4.3.6 Learning Objectives Scenario 4

The Key Performance Indicators to make the learning objectives of Scenario 4 visible are the average number of days the treatment is postponed per day and the average over the whole. With these KPIs, we can show that pooling is not beneficial in every situation. In a scenario where treatment times vary widely and short-term patients use approximately one-third of the capacity, pooling is likely not beneficial. In this scenario, assigning 1 server to short-term patients results in a lower average waiting time.

4.4 Layout Design

This section describes the design of the workbook and explains the navigation within the game.

4.4.1 Design of the Base

The use of colour plays an important role in the basis of the game. The colour makes it clear which scenario and which distribution is used. By using the same colours in the borders of the text boxes, not only the text but also the colour acts as a signal. The colour of the scenario is reflected in the buttons, the borders and the input fields.

In addition to the use of colour, different tabs are used. These tabs provide a better distinction between the different scenarios. To make it as easy as possible for the user, navigation is done by buttons. The navigation block contains three buttons that form the basis of the game. These buttons are: Calculate, Effect Pooling, and Home (see Figure 15).

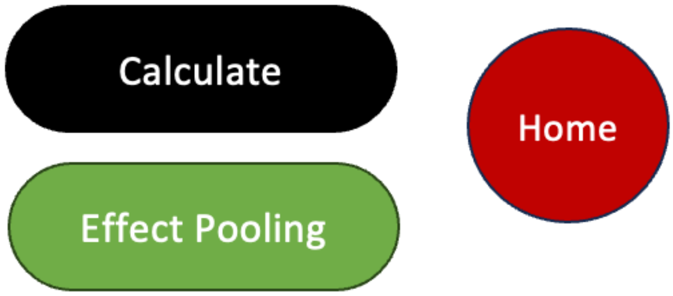
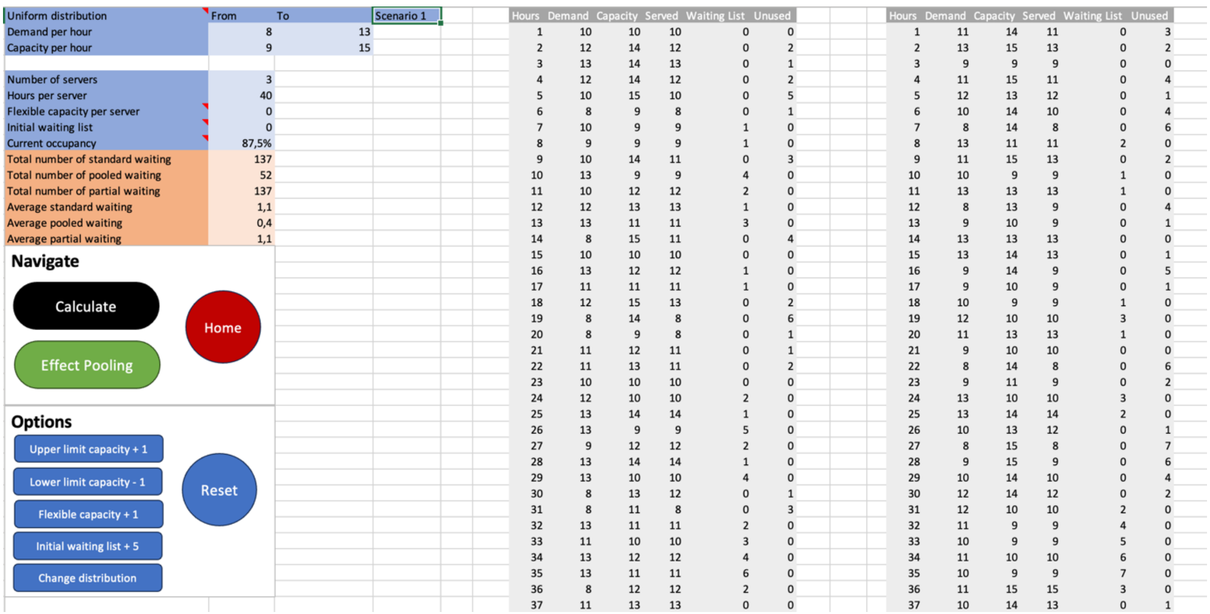


Figure 15: Screenshot of the navigation buttons

The calculate button is used to have the game recalculate the results after the user made changes to the input fields. To maintain the speed of the file, there is no automatic recalculation after changes to input fields, but only after pressing the button.

The button effect pooling jumps to the pooling tab, where a visual representation of the tables is displayed. Below that are the tables with the data on which the graphs are based. The pooling tab has a navigation block with only two buttons: back and home. The back button takes the user back to the previous tab, where you can perform new calculations. The red home button takes the user back to the home screen, where the learning objective can be read again, or a different scenario can be selected.

Figure 16 shows a screenshot of the Scenario 1 tab where all input values are filled in and changed and where the calculation per server takes place. The tabs of the other scenarios are similar with the main difference being the colour scheme and other input fields.



Uniform distribution	From	To	Scenario 1
Demand per hour	8	13	
Capacity per hour	9	15	
Number of servers	3		
Hours per server	40		
Flexible capacity per server	0		
Initial waiting list	0		
Current occupancy	87,5%		
Total number of standard waiting	137		
Total number of pooled waiting	52		
Total number of partial waiting	137		
Average standard waiting	1,1		
Average pooled waiting	0,4		
Average partial waiting	1,1		

Hours	Demand	Capacity	Served	Waiting List	Unused
1	10	10	10	0	0
2	12	14	12	0	2
3	13	14	13	0	1
4	12	14	12	0	2
5	10	15	10	0	5
6	8	9	8	0	1
7	10	9	9	1	0
8	9	9	9	1	0
9	10	14	11	0	3
10	13	9	9	4	0
11	10	12	12	2	0
12	12	13	13	1	0
13	13	11	11	3	0
14	8	15	11	0	4
15	10	10	10	0	0
16	13	12	12	1	0
17	11	11	11	1	0
18	12	15	13	0	2
19	8	14	8	0	6
20	8	9	8	0	1
21	11	12	11	0	1
22	11	13	11	0	2
23	10	10	10	0	0
24	12	10	10	2	0
25	13	14	14	1	0
26	13	9	9	5	0
27	9	12	12	2	0
28	13	14	14	1	0
29	13	10	10	4	0
30	8	13	12	0	1
31	8	11	8	0	3
32	13	11	11	2	0
33	11	10	10	3	0
34	13	12	12	4	0
35	13	11	11	6	0
36	8	12	12	2	0
37	11	13	13	0	0

Hours	Demand	Capacity	Served	Waiting List	Unused
1	11	14	11	0	3
2	13	15	13	0	2
3	9	9	9	0	0
4	11	15	11	0	4
5	12	13	12	0	1
6	10	14	10	0	4
7	8	14	8	0	6
8	13	11	11	2	0
9	11	15	13	0	2
10	10	9	9	1	0
11	13	13	13	1	0
12	8	13	9	0	4
13	9	10	9	0	1
14	13	13	13	0	0
15	13	14	13	0	1
16	9	14	9	0	5
17	9	10	9	0	1
18	10	9	9	1	0
19	12	10	10	3	0
20	11	13	13	1	0
21	9	10	10	0	0
22	8	14	8	0	6
23	9	11	9	0	2
24	13	10	10	3	0
25	13	14	14	2	0
26	10	13	12	0	1
27	8	15	8	0	7
28	9	15	9	0	6
29	10	14	10	0	4
30	12	14	12	0	2
31	12	10	10	2	0
32	11	9	9	4	0
33	10	9	9	5	0
34	11	10	10	6	0
35	10	9	9	7	0
36	11	15	15	3	0
37	10	14	13	0	1

Figure 16: Screenshot of the sheet for calculating Scenario 1

4.4.2 Visualisation of the Learning Objectives

This section covers the visual part of the learning objectives of Chapter 4, so the learning objectives are learned faster and easier. Chapter 2 showed that the visual part has a big effect on the gathering of knowledge. Figure 17 shows an example of a tab with the graphs, the tabs with the graphs of the other scenarios look very similar.

Figure 17: Screenshot of a tab with the graphs

Graphs Scenarios 1 to 3

The main objective is to make the user understand the effect of pooling. As described in Chapter 2, attention is one of the four factors for a successful learning experience. For this reason, the largest and most prominent graph shows the difference between pooling, partial pooling and no pooling (see Figure 18).

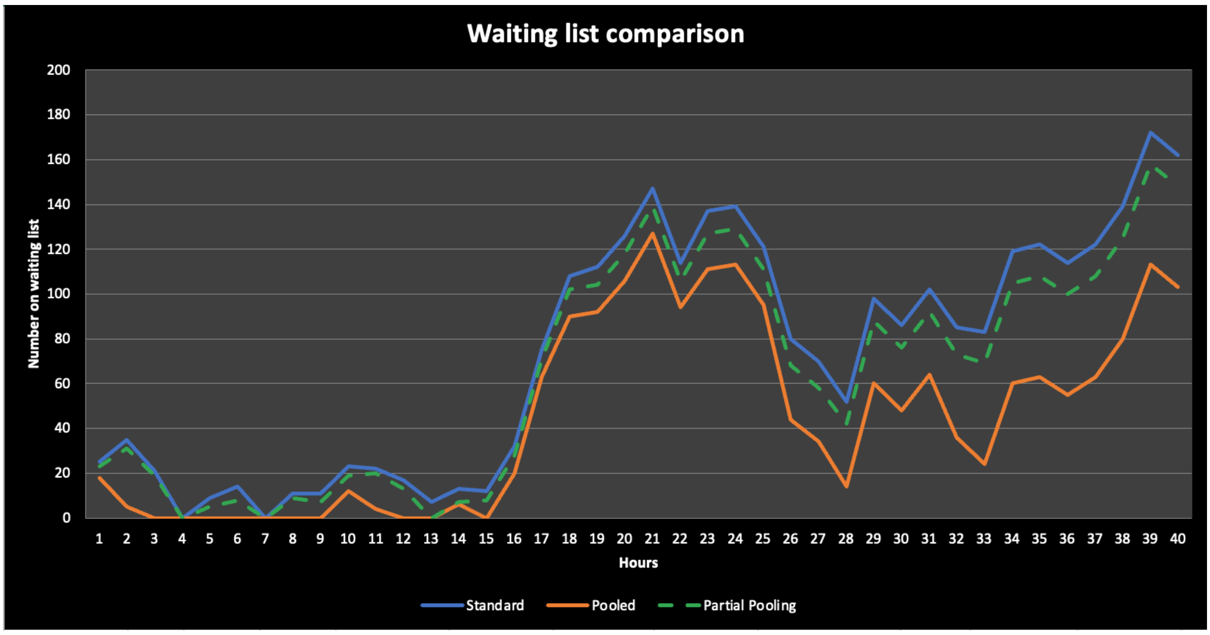


Figure 18: Screenshot graph waiting list comparison

This line graph compares the waiting list of the three pooling situations. In the case of no flexible capacity, the line of partial pooling will be equal to the line of no pooling. That is why this line is dotted so that it can be seen when it overlaps with one of the other lines. During the game, many situations are compared. To better compare different situations, there are buttons to display the lines of the previous calculation in the graph.

In addition to the line graph, two bar graphs are shown that show the total number of people waiting and the average number of people waiting per situation. These graphs provide a quick, simple representation of the effect of pooling. Figure 19 shows an example of both graphs.

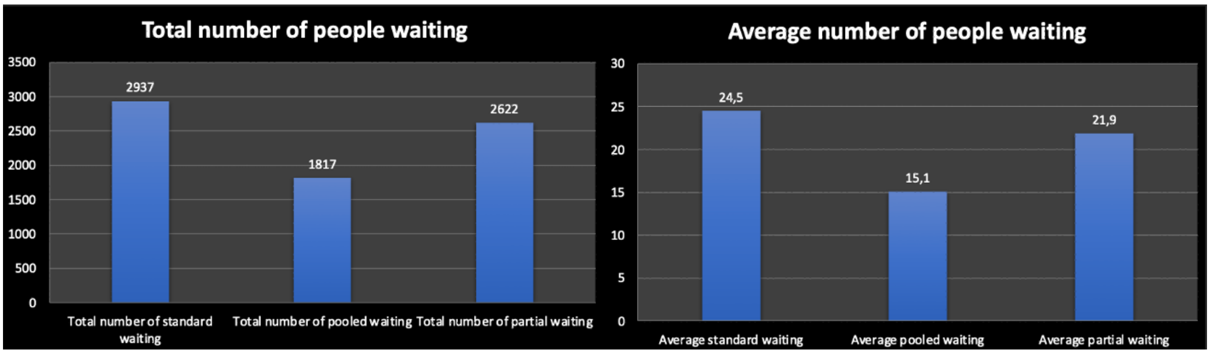


Figure 19: Screenshot bar graphs

Graphs Scenario 4

The line graph of Figure 20 shows the average number of days a treatment is postponed per day. The number on the line therefore indicates per day what the average number of postponed days was per postponed patient in both situations.

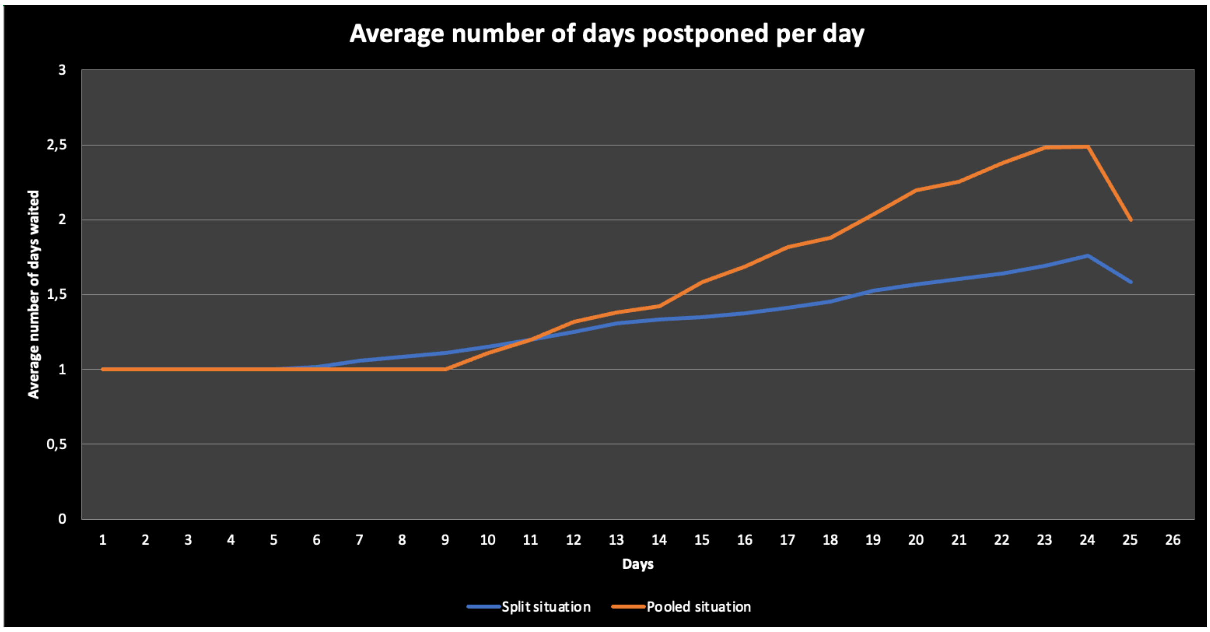


Figure 20: Graph comparing average days postponed per day

The bar chart of Figure 21 shows the average number of days the treatment of the postponed patients was delayed. This graph gives a quick overview of the two situations.

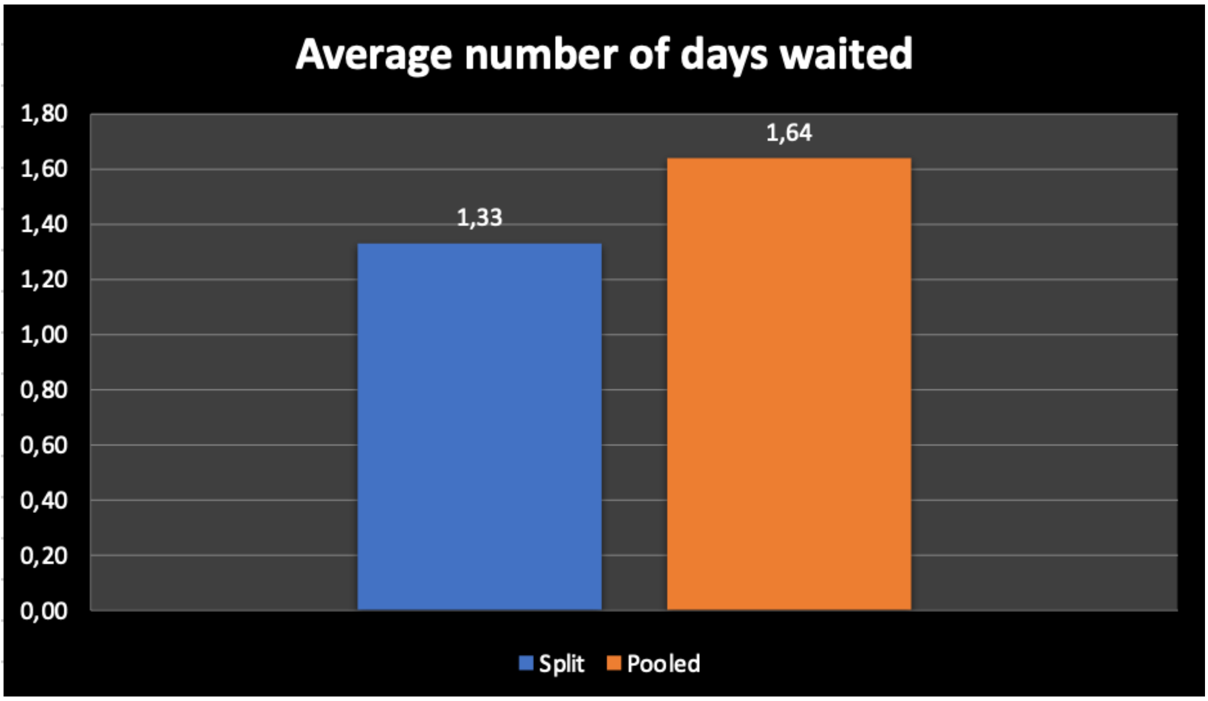


Figure 21: Bar chart average number of days waited per situation

Figure 22 shows per situation how many patients saw their treatment postponed. In this graph, a distinction is made in the bar of the split situation between the number of short-term and long-term patients. This distinction is made because this situation also makes this distinction during planning and these values can therefore be relevant.

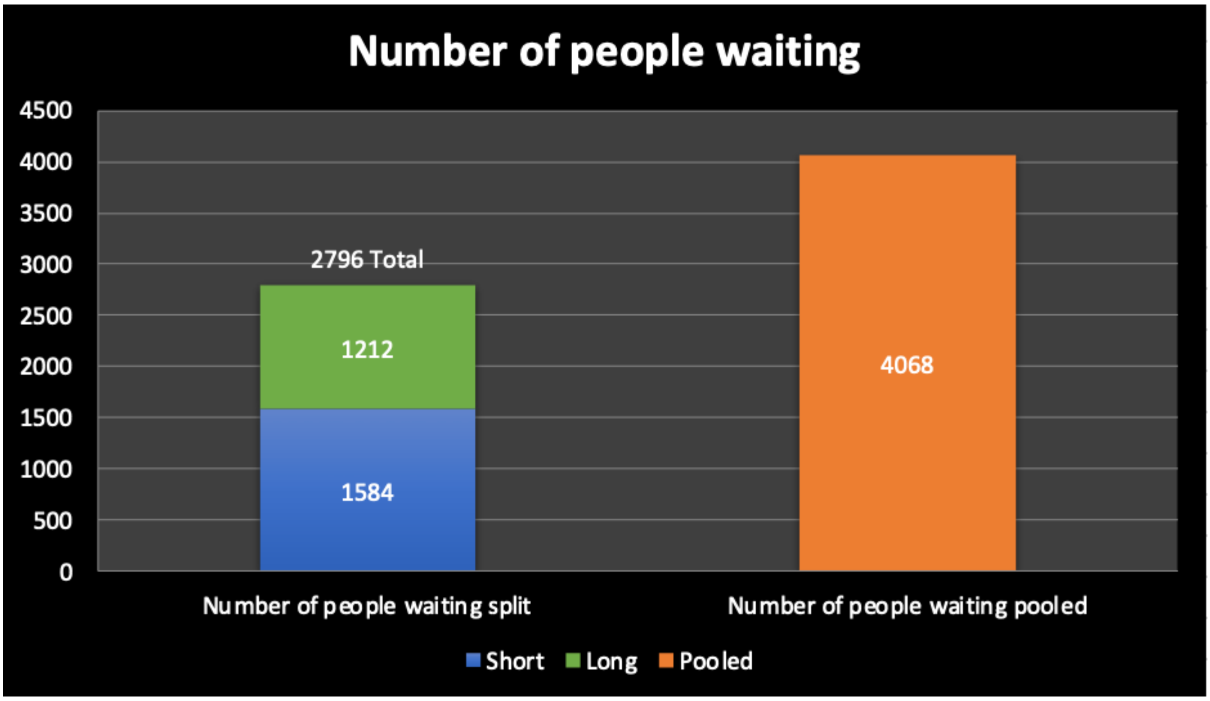


Figure 22: Bar chart total number of people waiting per situation

4.5 Additional Features

For users who would like to dig deeper, on the sheets with the graphs of scenarios 1 to 3 there are three buttons that lead to the raw data used for the graphs. Each button leads to a table with the data for the situation in question. This contains the results of the calculations of the different kinds of pooling per hour. For example, this can be interesting to see at what time the initial waiting list has been cleared. This is harder to see in the graph and can be seen more clearly in the table.

Another feature is that the entire workbook can be translated on the home screen. When a different language is selected, all tabs and graphs are automatically translated. It is built in such a way that it is easy to extend to other languages.

Finally, VBA ensures that the screen is automatically zoomed correctly when the user opens the file. This ensures that the user does not miss any parts of the file and thus sees the full explanations and learning objectives.

Chapter 5 Gameplay Protocol

This chapter presents the gameplay protocol. Section 5.1 gives a brief general introduction, Section 5.2 contains information about the home screen and general information, and Section 5.3 gives the instructions that should ensure learning the learning objectives.

5.1 Introduction

The gameplay protocol aims to provide short, clear instructions from which users can learn the learning objectives for themselves. The protocol consists of four parts, following the four scenarios that were presented in Chapter 4:

- Scenario 1:
  Scenario 1 is a scenario where demand and capacity are determined by the uniform distribution. This concerns the number of patients or beds per hour.
- Scenario 2:
  Scenario 2 uses two patient types with each having a different treatment time. The demand is filled based on the maximum capacity filled in by the user. The actual capacity is determined by the occupancy rate and variability, which are also entered by the user. Both supply and demand use a uniform distribution.
- Scenario 3:
  Scenario 3 is a variation of Scenario 1, the only difference being that it uses a normal distribution.
- Scenario 4:
  Scenario 4 is a new scenario that shows that pooling is not always beneficial. For this, a waiting list is used with short-term and long-term patients who are scheduled in two different ways.

Each scenario allows adjustments to the number of servers and simulated hours, as well as specific options.

In Chapter 2 we stated that serious games are applications with three components: experience, entertainment and multimedia. This chapter focuses on the text part of multimedia, the graphics part is already explained in Chapter 4. Besides that, it also covers the experience part as this section covers the serious part of the serious game, the teaching of the learning objectives. Both Chapter 4 and Chapter 5 contribute to the final component, the entertainment part.

Chapter 2 also showed that there are four important factors for a successful serious game: attention, active engagement, feedback of information, and consolidation. We aim to include these four factors in the gameplay protocol. We do this by using bright colours (attention), letting them make the requested changes themselves (active engagement), visual results that clarify the outcomes (feedback of information), and repeating the same lessons in a slightly different form (consolidation).

The gameplay protocol is shown on the home screen of the Excel workbook and consists of multiple text boxes and buttons, the contents of which are discussed in Sections 5.2 and 5.3.

5.2 Design of the Home Screen

The home screen of the workbook is where the gameplay protocol is located so it cannot be missed. The gameplay protocol consists of six text boxes. Two contain general information which is discussed in this section, and four concern the game scenarios which are discussed in Section 5.3. Figure 23 shows a screenshot of the home screen of the serious game.

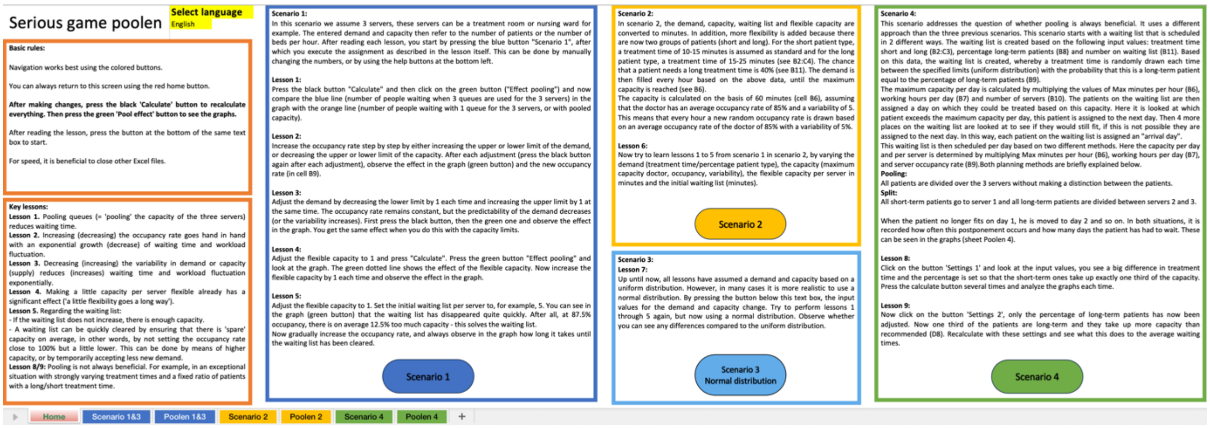


Figure 23: Screenshot of home screen serious game

5.2.1 General Information

The first text box contains information on how to play the game, the basic rules. These basic rules are important throughout the entire game.

Basic rules:

Navigation works best using the colored buttons.

You can always return to this screen using the red home button.

After making changes, press the black 'Calculate' button to recalculate everything. Then press the green 'Pool effect' button to see the graphs.

After reading the lesson, press the button at the bottom of the same text box to start.

For speed, it is beneficial to close other Excel files.

Figure 24: Screenshot text box basic rules

The first two sentences provide the rules for smooth navigation. For a smooth game flow, good navigation is important. By pointing the user to the use of the buttons, this is made easier. The bold sentence contains the most important rule, users must press the calculate button again after each adjustment. As discussed in Chapter 4, this does not happen automatically because the calculation takes some time and will slow down the game too much. The penultimate sentence is about the controls on the home screen itself. Under each text box with instructions is a button that goes directly to the right scenario. The last line is a general comment to keep the speed as high as possible. Closing other Excel files and applications promotes speed as more computing power is available.

5.2.2 Key Lessons

The next text box contains all the key lessons that can be learned from the game. This text box can help when the lesson is not immediately clear after the instructions and therefore serves as a summary and resource if the user does not understand. When the tool is used for education, the textbox can be removed to push learners to find out the lesson themselves. Figure 25 shows the text box and the instructions from which these key lessons should emerge are discussed in Section 5.3.

Key lessons:

Lesson 1. Pooling queues (= 'pooling' the capacity of the three servers) reduces waiting time.

Lesson 2. Increasing (decreasing) the occupancy rate goes hand in hand with an exponential growth (decrease) of waiting time and workload fluctuation.

Lesson 3. Decreasing (increasing) the variability in demand or capacity (supply) reduces (increases) waiting time and workload fluctuation exponentially.

Lesson 4. Making a little capacity per server flexible already has a significant effect ('a little flexibility goes a long way').

Lesson 5. Regarding the waiting list:

- If the waiting list does not increase, there is enough capacity.
- A waiting list can be quickly cleared by ensuring that there is 'spare' capacity on average, in other words, by not setting the occupancy rate close to 100% but a little lower. This can be done by means of higher capacity, or by temporarily accepting less new demand.

Lesson 8/9: Pooling is not always beneficial. For example, in an exceptional situation with strongly varying treatment times and a fixed ratio of patients with a long/short treatment time.

Figure 25: Screenshot of text box key lessons

5.3 The Instructions

This section discusses the gameplay protocol instructions per scenario and discusses the lessons that can be learned from them.

5.3.1 Scenario 1

The third text box contains instructions for playing Scenario 1 (see Figure 26). There is a short introduction that describes an example of a situation, and it once again shows the most important basic rules. Next, instructions for lessons 1 through 5 are given, these instructions ensures that the user can draw the lessons of Figure 25 from the graphs. Below Figure 26 we show per lesson how the lesson becomes clear after following the instructions. An extended version of the explanation of the lessons can be seen in Appendix B, that version shows the effect of the lessons using the graphs.

Scenario 1:
In this scenario we assume 3 servers, these servers can be a treatment room or nursing ward for example. The entered demand and capacity then refer to the number of patients or the number of beds per hour. After reading each lesson, you start by pressing the blue button "Scenario 1", after which you execute the assignment as described in the lesson itself. This can be done by manually changing the numbers, or by using the help buttons at the bottom left.

Lesson 1:
Press the black button "Calculate" and then click on the green button ("Effect pooling") and now compare the blue line (number of people waiting when 3 queues are used for the 3 servers) in the graph with the orange line (number of people waiting with 1 queue for the 3 servers, or with pooled capacity).

Lesson 2:
Increase the occupancy rate step by step by either increasing the upper or lower limit of the demand, or decreasing the upper or lower limit of the capacity. After each adjustment (press the black button again after each adjustment), observe the effect in the graph (green button) and the new occupancy rate (in cell B9).

Lesson 3:
Adjust the demand by decreasing the lower limit by 1 each time and increasing the upper limit by 1 at the same time. The occupancy rate remains constant, but the predictability of the demand decreases (or the variability increases). First press the black button, then the green one and observe the effect in the graph. You get the same effect when you do this with the capacity limits.

Lesson 4:
Adjust the flexible capacity to 1 and press "Calculate". Press the green button "Effect pooling" and look at the graph. The green dotted line shows the effect of the flexible capacity. Now increase the flexible capacity by 1 each time and observe the effect in the graph.

Lesson 5:
Adjust the flexible capacity to 1. Set the initial waiting list per server to, for example, 5. You can see in the graph (green button) that the waiting list has disappeared quite quickly. After all, at 87.5% occupancy, there is on average 12.5% too much capacity - this solves the waiting list.
Now gradually increase the occupancy rate, and always observe in the graph how long it takes until the waiting list has been cleared.

Scenario 1

Figure 26: Screenshot text box Scenario 1

Lesson 1

Lesson 1 shows the difference between complete pooling and no pooling. No further adjustments are required for this, the game is automatically set up correctly so that this effect is visible. After following the steps, you see a graph where the blue line (Standard) is clearly higher than the orange line (Pooling). In addition, you see a clearly lower number of people waiting in the bar charts in the case of pooling. This shows that pooling queues reduces the waiting time.

Lesson 2

Lesson 2 shows the effect of the occupancy rate on the waiting list. By adjusting the occupancy rate step by step and looking at the effect on the waiting list per step. Although the uniform distribution can lead to larger differences per simulation, you see that a higher occupancy rate leads to an exponential growth in waiting time. The graph shows higher waiting lists and more workload fluctuations.

Lesson 3

Lesson 3 shows the effect of variability on the waiting list. Increasing variability step by step while keeping occupancy the same causes an exponential increase in the waiting list and more fluctuations in the workload. This effect is visible in the graph, you mainly see a more erratic line with more peaks and troughs. In addition, the number of people waiting increases. Due to the fluctuations caused by the uniform distribution, it can sometimes be difficult to observe this at once. Therefore, it is important to keep repeating the steps to make the effect more visible.

Lesson 4

Lesson 4 shows that making capacity a little bit flexible already has a significant effect on the waiting list. By increasing the flexible capacity by one per step, the dotted line (partial pooling) in the graph shifts more and more towards the complete pooling situation. When the flexible capacity is zero the dotted line is equal to the standard situation and when the flexible capacity is large enough it will be equal to the complete pooling situation.

Lesson 5

Lesson 5 shows the effect of the occupancy rate on the waiting list and any initial waiting list. The effect of the occupancy rate on the waiting list has already been discussed in lesson 2, however, an initial waiting list is now added, and flexible capacity is available. The graph now starts with a peak, the initial waiting list, the higher the occupancy rate the longer it takes for this peak to be resolved.

5.3.2 Scenario 2

The fourth text box on the home screen contains all the information and instructions regarding Scenario 2. This text box starts with an extensive explanation of what Scenario 2 entails and how demand and capacity are now calculated. Now that all data is displayed in minutes, the user is confronted with larger numbers. In addition, this variant contains an extra layer by processing two different patient types in the demand. This combination provides new stimuli and, in combination with repeating lessons 1 to 5 of Scenario 1, an improved learning experience. The repetition and the reactivating of the brain with new elements contribute to learning the objectives.

Scenario 2:
In scenario 2, the demand, capacity, waiting list and flexible capacity are converted to minutes. In addition, more flexibility is added because there are now two groups of patients (short and long). For the short patient type, a treatment time of 10-15 minutes is assumed as standard and for the long patient type, a treatment time of 15-25 minutes (see B2:C4). The chance that a patient needs a long treatment time is 40% (see B11). The demand is then filled every hour based on the above data, until the maximum capacity is reached (see B6).
The capacity is calculated on the basis of 60 minutes (cell B6), assuming that the doctor has an average occupancy rate of 85% and a variability of 5. This means that every hour a new random occupancy rate is drawn based on an average occupancy rate of the doctor of 85% with a variability of 5%.

Lesson 6:
Now try to learn lessons 1 to 5 from scenario 1 in scenario 2, by varying the demand (treatment time/percentage patient type), the capacity (maximum capacity doctor, occupancy, variability), the flexible capacity per server in minutes and the initial waiting list (minutes).

Scenario 2

Figure 27: Screenshot text box Scenario 2

5.3.3 Scenario 3 Normal Distribution (truncated)

The fifth text box contains lessons and instructions for Scenario 3. Where Scenarios 1 and 2 used the uniform distribution, this lesson focuses on the normal distribution. Lessons 1 through 5 of Scenario 1 are used again because repetition is important during the learning process. The user is asked whether, even though the same lessons are applied, there are also differences to be observed due to the different distribution. The biggest difference that can be observed is that the normal distribution more often shows values around the average, but also gives a chance of extreme outliers. With the uniform distribution, this is not possible because every value remains between the given values. Despite these differences, the outcomes of the lessons will still be much the same.

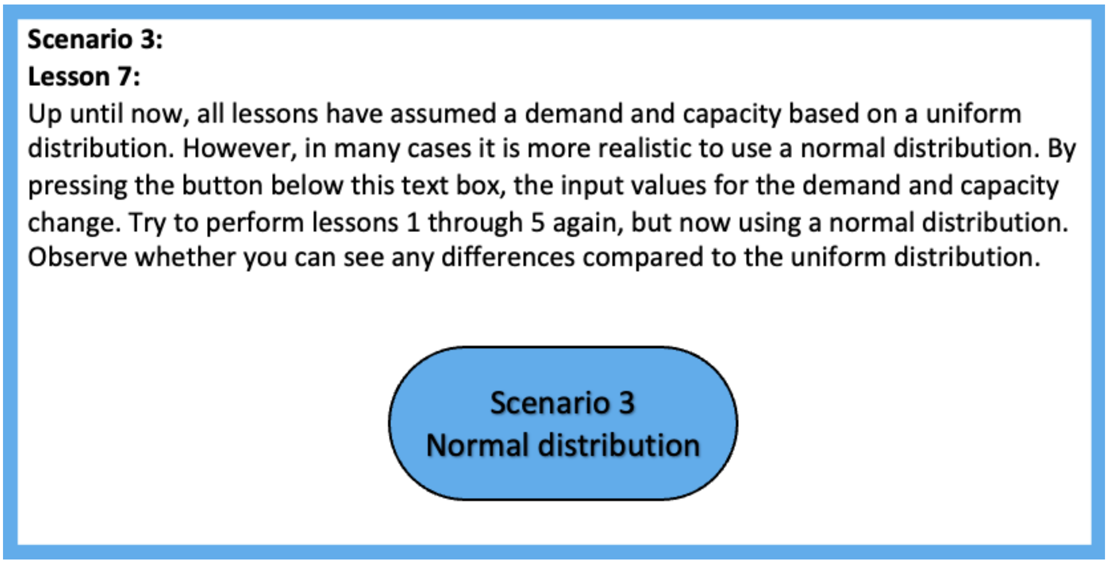


Figure 28: Screenshot text box Scenario 3 normal distribution

5.3.4 Scenario 4

The last text box contains the latest instructions and lessons. The text box begins with an extensive explanation of how this situation works as it is quite different from the other scenarios. This explains how the different planning strategies work and which calculations are used. This explains how the different planning strategies work and which calculations are used. After this explanation, 2 lessons follow, in which it becomes clear that pooling is not beneficial in all cases. These lessons are described in Figure 29, this figure shows the entire text box of Scenario 4. Appendix C provides a more detailed explanation of the lessons including screenshots of the graphs.

Scenario 4:

This scenario addresses the question of whether pooling is always beneficial. It uses a different approach than the three previous scenarios. This scenario starts with a waiting list that is scheduled in 2 different ways. The waiting list is created based on the following input values: treatment time short and long (B2:C3), percentage long-term patients (B8) and number on waiting list (B11). Based on this data, the waiting list is created, whereby a treatment time is randomly drawn each time between the specified limits (uniform distribution) with the probability that this is a long-term patient equal to the percentage of long-term patients (B9).

The maximum capacity per day is calculated by multiplying the values of Max minutes per hour (B6), working hours per day (B7) and number of servers (B10). The patients on the waiting list are then assigned a day on which they could be treated based on this capacity. Here it is looked at which patient exceeds the maximum capacity per day, this patient is assigned to the next day. Then 4 more places on the waiting list are looked at to see if they would still fit, if this is not possible they are assigned to the next day. In this way, each patient on the waiting list is assigned an "arrival day".

This waiting list is then scheduled per day based on two different methods. Here the capacity per day and per server is determined by multiplying Max minutes per hour (B6), working hours per day (B7), and server occupancy rate (B9).Both planning methods are briefly explained below.

Pooling:

All patients are divided over the 3 servers without making a distinction between the patients.

Split:

All short-term patients go to server 1 and all long-term patients are divided between servers 2 and 3.

When the patient no longer fits on day 1, he is moved to day 2 and so on. In both situations, it is recorded how often this postponement occurs and how many days the patient has had to wait. These can be seen in the graphs (sheet Poolen 4).

Lesson 8:

Click on the button 'Settings 1' and look at the input values, you see a big difference in treatment time and the percentage is set so that the short-term ones take up exactly one third of the capacity. Press the calculate button several times and analyze the graphs each time.

Lesson 9:

Now click on the button 'Settings 2', only the percentage of long-term patients has now been adjusted. Now one third of the patients are long-term and they take up more capacity than recommended (D8). Recalculate with these settings and see what this does to the average waiting times.

Scenario 4

Figure 29: Screenshot text box Scenario 4

Lesson 8

Lesson 8 asks the user to press a button with the parameters set in such a way that pooling is often not beneficial. This can be clearly seen by comparing the average number of days waited for both situations; Figure 21 shows an example of this. We ran the simulations with these values 200 times and in 183 out of 200 cases pooling was not more advantageous. In other words, in 91.5% of the cases pooling with these settings is not beneficial. To make sure the user sees this too, the user is asked to repeat the simulation multiple times with the same input values.

Lesson 9

Lesson 9 makes it clear that the input values for which pooling is not more advantageous come very precisely. By slightly increasing the percentage of long-term patients it becomes clear that pooling works better than the split variant. From the combination of lessons 8 and 9, we can learn that pooling is not always more beneficial, but that this is only the case in exceptional cases. This makes it clear that it is important to first investigate whether pooling is actually beneficial.

5.4 Conclusion

In this section, we discuss the extent to which the processing of the theory has been successful in the gameplay protocol. As discussed in Section 5.1, we considered a definition containing three components and four factors during the writing of the gameplay protocol.

All three components of the definition are covered in the game. The multimedia component is reflected in the use of both graphics and text. The experience component is reflected in the lessons that lead to the achievement of the learning objectives. And the final entertainment component is reflected in the fact that the user can make adjustments and "have fun" experimenting with the numbers. Additionally, the four factors were applied as described in Section 5.1. Namely, by using bright changing colours (attention), allowing the user to make adjustments (active engagement), graphs that show the results (feedback of information), and repeating lessons in changing scenarios (consolidation).

Chapter 6 Experiences and Feedback

This chapter gives an overview of different moments of feedback. Section 6.1 is about the very first feedback done by a teaching assistant. Section 6.2 is about a classroom experiment that was done to see how effective the lessons are. Finally, in Section 6.3 a short conclusion follows based on both types of feedback.

6.1 First Feedback

The first form of feedback was done by a teaching assistant (an undergraduate student of Industrial Engineering and Management). This feedback is intended to further improve the game and the protocol, so that it functions properly and that the learning objectives are achieved. The feedback itself can be found in Appendix A.

6.1.1 Most Important Changes

This section lists the changes made based on the feedback. This is followed by an overview of additional adjustments that were made while waiting for the first feedback.

- Adding axis titles to charts.
- Removing redundant and unclear data.
- Splitting the scenarios into different tabs to clarify the distinction and facilitate comparison.
- Give the graphs a more prominent place and make them more approachable.

In addition to these adjustments based on feedback, some adjustments have been made to simplify playing or add more functionality:

- Adding more buttons to simplify navigation within the document.
- Making previous calculations visible in the graph by a button.
- Automatic zoom when opening the document, so that all data is visible when opened and the chance of users missing something is reduced.

6.2 Classroom Experiment

A classroom experiment was conducted to investigate how students experience working with the serious game and to see how well the lessons were understood. To see if the learning objectives are clear without being named, the text box with the learning objectives has been removed. To ensure that the assignment was taken seriously, a bonus was awarded to those who performed satisfactorily. Because both experiments, the classroom experiment and the feedback from the teaching assistant, were too close together in time. In both cases, the same version of the game was used. This resulted in an overlap in feedback but also provided sufficient new insights and problems that users encountered.

6.2.1 Experiences of the Users

Part of the assignment was to give feedback on the game. Some of the given feedback overlapped with that of the teaching assistant and we therefore only briefly touch on this. Two things that often came back were the lack of overview. Due to the lack of clear axis titles and no clear distinction between the scenarios and tabs. This also came up in 6.1 and was therefore already covered.

The classroom experiment also revealed several new user insights. It was not entirely clear to some users why the graphs with the same input values could differ so much. To make this extra clear, it is explicitly stated that the absolute numbers in the uniform distribution can fluctuate considerably. In addition, not everyone understood which values may be adjusted. We will clarify this by stating this explicitly in the text.

6.2.2 Overview of Points Achieved per Lesson

To get a good overview of the educational value of the game, we have included the results of the classroom experiment in Table 1. The table counts the number of times a lesson is understood, but also looks at where it went wrong when this was not understood. This could be because the assignment was not clearly formulated or because the lesson was too difficult and was not understood without help.

	Lesson 1	**Lesson 2**	**Lesson 3**	**Lesson 4**	**Lesson 5**	**Lesson 6**
Lesson understood	15	9	11	12	8	9
Assignment not clear	2	2	2	2	7	5
Assignment clear, but lesson not understood	1	7	5	4	3	4

Table 1 Overview of classroom experiment results

The table shows clear differences between the different lessons. It is noticeable that lesson 1 was understood well in 15 out of 18 cases, but lesson 5 was only understood in 8 out of 15 cases. The main conclusions based on the table are listed below:

- The main learning objective of this thesis, pooling, is best understood. In 83.3% of the cases. Also, the part about partial pooling was often understood, specifically in two-thirds of the cases.
- The assignment was clear with four of the six lessons, but this was not the case with lessons 5 and 6.
  For this reason, the text of both lessons has been adapted, which should make the assignment clearer and the lessons easier to understand.
- Lessons 2 and 3 were the most difficult for the students. The assignment was clear, but the learning objectives were not taken from the graphs.

It was decided not to change anything in the text, in the normal situation the text box with the learning objectives is visible. The combination of the instructions with these learning objectives should be sufficient in our opinion to understand these lessons as well.

6.3 Conclusions

There was a lot of overlap between the general feedback from Sections 6.1 and 6.2. Much of this feedback was about improving clarity. Therefore, we made several changes to both the text and, for example, axis titles in graphs. In terms of lessons, the most confusion was with lessons 5 and 6, so the text for both lessons has been adjusted. Although some of the students were unable to extract the learning objectives from lessons 2 and 3, we did not adjust the text. The assignment was clear, and we believe that when the text box with learning objectives is visible, the learning objective can be extracted from the graphs.

Overall, the serious game has shown to be a good way to convey the effect of pooling queues. The choice for a serious game has worked out well and after some minor adjustments, we believe that this game can achieve its goal of making the pooling effect visible and conveying it to healthcare professionals. In addition, the other learning objectives will contribute to knowledge in the field of ORMS, which makes this game more widely applicable.

Chapter 7 Conclusion and Recommendations

Section 7.1 presents the conclusions and Section 7.2 discusses possible improvements to the serious game and provides recommendations for further research.

7.1 Conclusions

This section discusses the results and conclusions of the research questions of Chapter 1.

1. What serious games for healthcare operations management currently exist in the literature? (Chapter 2)

Chapter 2 shows that there is still little available in the field of serious games in ORMS. The serious games found related to healthcare were focused on improving care. While it became clear that serious games are an excellent means to transfer knowledge in this field. With this thesis in combination with the developed serious game, we jump into this gap.

2. What lessons can be learnt from the pooling effect for capacity planning? (Chapter 3)

Chapter 3 shows what the positive effect of pooling queues can be on the waiting time, but also highlights an example where pooling has a negative effect. The lesson is that pooling queues is usually beneficial, but in exceptional cases with high variability it is not. Hence this serious game that visualises both situations.

3. What serious game is suitable to educate healthcare professionals various lessons regarding the pooling effect? (Chapter 4)

Chapter 4 shows the structure of the serious game and the associated calculations. Excel is used as a basis and the learning objectives are learned by going through different scenarios. The theory from Chapters 2 and 3 is used during the development because this increases the chance that users will learn the learning objectives. The main goal of the game is to demonstrate the effect of pooling queues, but it also has learning objectives that identify standard pitfalls in capacity management in healthcare.

4. What is a suitable gameplay protocol for classroom settings? (Chapter 5)

Chapter 5 provides the structure of the gameplay protocol, which is located on the home screen of the Excel file. The gameplay protocol consists of two parts. The first part consists of a general part with some basic rules and the most important learning objectives. The second part consists of text boxes that provide instructions per situation that lead to the learning objectives from the first part.

5. What improvements can be made to the serious game and the gameplay protocol? (Chapter 6)

The feedback from the teaching assistant and the classroom experiment gave a lot of overlap in feedback. The first version of the game contained too many ambiguities that could lead to confusion. Based on this feedback, the graphs have been adjusted, with the titles and axes now more clearly indicated and the graphs given a more prominent position. Lesson 5 of the gameplay protocol was not well understood by many people, after which the text was adjusted. Finally, Chapter 6 showed that the serious game succeeded in its primary goal. As many as 83.3% of the groups understood the effect of pooling after playing the serious game.

7.2 Recommendations

During the development of the serious game and the study of the literature, additional ideas for both extensions and improvements to the serious game emerged. Since these are beyond the scope of this thesis, we will highlight them below.

7.2.1 The Serious Game

The latest version of the serious game has not yet been tested. It is advisable to have it tested again by a group of students to see if the lessons are understood even better now. In addition, the game was expanded quite late in the process with a new scenario that has not yet been tested at all. Hence, the advice is to conduct another feedback experiment with it to be able to implement any possible improvements.

To make the serious game even more successful, more interactive feedback during the game can be considered. This contributes to a better learning experience and can also further increase the success rate of the lessons learned.

In this case, Excel was chosen as the tool for the serious game however, Excel is quite limited in visual effects. For more visual options the game could be converted into another tool. Making the game more attractive with visual effects has a positive effect on the learning experience.

The goal of this thesis was that healthcare professionals would understand the effect of pooling. This has only been tested on students who did a healthcare-related study, but not yet on healthcare professionals. It would therefore be good to test the game on, for example, healthcare managers and practitioners. This allows both managers and practitioners to see the effect of pooling.

7.2.2 Serious games in the field of ORMS

Although this game fills a part of the gap of few to no serious games in the field of ORMS, there is still room for additional serious games that can illuminate other aspects within this field of research. An example of this is the development of a game that shows the effect of flex pools on nursing departments. These flex pools contain nurses who can be deployed in multiple departments and can therefore better cope with peaks in demand. In this thesis, it was repeatedly shown that these pools have a positive effect when used properly, but when not applied properly they have a negative effect. A serious game can help to see in which situations these flex pools can be beneficial.

Bibliography

Bishen, S. (2023). *World Health Day: 8 trends shaping global healthcare | World Economic Forum*. https://www.weforum.org/agenda/2023/04/world-health-day-healthcare-trends/

Botchkarev, A. (2015). *Assessing Excel VBA Suitability for Monte Carlo Simulation*. https://arxiv.org/abs/1503.08376v1

Calik, A., Cakmak, B., Kapucu, S., & Inkaya, B. (2022). The effectiveness of serious games designed for infection prevention and promotion of safe behaviors of senior nursing students during the COVID-19 pandemic. *American Journal of Infection Control*, *50*(12), 1360–1367. https://doi.org/10.1016/J.AJIC.2022.02.025

Cao, P., He, S., Huang, J., & Liu, Y. (2020). To Pool or Not to Pool: Queueing Design for Large-Scale Service Systems. *Https://Doi.Org/10.1287/Opre.2019.1976*, *69*(6), 1866–1885. https://doi.org/10.1287/OPRE.2019.1976

Caserman, P., Hoffmann, K., Müller, P., Schaub, M., Straßburg, K., Wiemeyer, J., Bruder, R., & Göbel, S. (2020). Quality Criteria for Serious Games: Serious Part, Game Part, and Balance. *JMIR Serious Games 2020;8(3):E19037 Https://Games.Jmir.Org/2020/3/E19037*, *8*(3), e19037. https://doi.org/10.2196/19037

Cattani, K., & Schmidt, G. M. (2005a). The Pooling Principle. *INFORMS Transactions on Education*, *5*(2), 17–24. https://doi.org/10.1287/ITED.5.2.17

Cattani, K., & Schmidt, G. M. (2005b). The Pooling Principle. *INFORMS Transactions on Education*, *5*(2), 17–24. https://doi.org/10.1287/ited.5.2.17

Cattani, K., & Schmidt, G. M. (2005c). The Pooling Principle. *INFORMS Transactions on Education*, *5*(2), 17–24. https://doi.org/10.1287/ITED.5.2.17

CBS. (2024). *SDG 3 Goede gezondheid en welzijn | CBS*. https://www.cbs.nl/nl-nl/visualisaties/monitor-brede-welvaart-en-de-sustainable-development-goals/sdg-s/sdg-3-goede-gezondheid-en-welzijn

Connolly, T. M., Boyle, E. A., Macarthur, E., Hainey, T., & Boyle, J. M. (2012). A systematic literature review of empirical evidence on computer games and serious games. *Computers & Education*, *59*, 661–686. https://doi.org/10.1016/j.compedu.2012.03.004

Dehaene, S. (2013). *Did neuroscience find the secrets of learning?* http://www.paristechreview.com/2013/11/07/neuroscience-secret-learning/?media=print

Dziuba-Ellis, J. (2006). Float pools and resource teams: a review of the literature. *Journal of Nursing Care Quality*, *21*(4), 352–359. https://doi.org/10.1097/00001786-200610000-00013

Fagefors, C., & Lantz, B. (2021). Application of Portfolio Theory to Healthcare Capacity Management. *International Journal of Environmental Research and Public Health*, *18*(2), 1–9. https://doi.org/10.3390/IJERPH18020659

Fagefors, C., Lantz, B., & Rosén, P. (2020). Creating Short-Term Volume Flexibility in Healthcare Capacity Management. *International Journal of Environmental Research and Public Health 2020, Vol. 17, Page 8514*, *17*(22), 8514. https://doi.org/10.3390/IJERPH17228514

Fagefors, C., Lantz, B., Rosén, P., & Siljemyr, L. (2024). Staff pooling in healthcare systems – results from a mixed-methods study. *Health Systems*, *13*(1), 31–47. https://doi.org/10.1080/20476965.2022.2108729

Giessen, H. W. (2015). Serious Games Effects: An Overview. *Procedia - Social and Behavioral Sciences*, *174*, 2240–2244. https://doi.org/10.1016/j.sbspro.2015.01.881

Gulumbe, B. H., & Usman, N. I. (2023). Projected shortfall of 10 million healthcare workers by 2030: implications for low-and middle-income countries and the way forward. *Healthcare in Low-Resource Settings*, *11*(1). https://doi.org/10.4081/HLS.2023.11359

Heerkens, H., & Winden, A. van. (2017). *Solving Managerial Problems Systematically*. Noordhoff Uitgevers. https://research.utwente.nl/en/publications/solving-managerial-problems-systematically

Helder, C. (2023). *Arbeidsmarktprognose zorg en welzijn*. https://www.prognosemodelzw.nl/documenten/brieven/2023/03/21/nieuwe-arbeidsmarktprognose-zorg-en-welzijn

Hopper, G. (1976). *Computerworld*. Vol. 10, No. 4. https://books.google.com.au/books?id=3u9H-xL4sZAC&vq=%22most+dangerous%22&source=gbs_navlinks_s

Kendall, D. G. (1953). Stochastic Processes Occurring in the Theory of Queues and their Analysis by the Method of the Imbedded Markov Chain. *Https://Doi.Org/10.1214/Aoms/1177728975*, *24*(3), 338–354. https://doi.org/10.1214/AOMS/1177728975

Khan, A. Q., Khan, S., & Safaev, U. (2020). *Serious Games and Gamification: A Systematic Literature Review*. https://doi.org/10.20944/PREPRINTS202011.0280.V1

Kolb, D. A., & Fry, R. E. (1974). Toward an applied theory of experiential learning. In *C. Cooper, Theories of group processes* (pp. 33–57). https://www.researchgate.net/profile/David-Kolb-2/publication/238759143_Toward_an_Applied_Theory_of_Experiential_Learning/links/57d2add208ae0c0081e25093/Toward-an-Applied-Theory-of-Experiential-Learning.pdf

Laamarti, F., Eid, M., & El Saddik, A. (2014). An overview of serious games. *International Journal of Computer Games Technology*, *2014*. https://doi.org/10.1155/2014/358152

Lim, J. M., Song, H., & Yang, J. J. (2024). The Spillover Effects of Capacity Pooling in Hospitals. *Https://Doi.Org/10.1287/Mnsc.2022.02202*. https://doi.org/10.1287/MNSC.2022.02202

Maister, D. H. (1984). The Psychology of Waiting Lines. *Harvard Business Review Press (China Case Studies)*. https://davidmaister.com/articles/the-psychology-of-waiting-lines/

Marion-Martins, A. D., & Pinho, D. L. M. (2020). Interprofessional simulation effects for healthcare students: A systematic review and meta-analysis. *Nurse Education Today*, *94*, 104568. https://doi.org/10.1016/J.NEDT.2020.104568

Michael, D. R., & Chen, S. L. (2005). *Serious games: Games that educate, train, and inform. Muska & Lipman/Premier-Trade*. 287.

Montalbano, L., Gallo, L., Ferrante, G., Malizia, V., Cilluffo, G., Fasola, S., Alesi, M., & La Grutta, S. (2022). Serious Games: A new Approach to Foster Information and Practices About Covid-19? *Frontiers in Robotics and AI*, *9*. https://doi.org/10.3389/FROBT.2022.830950

Mooney, C. Z. (1997). *Monte Carlo simulation, Series 07-116*. 103. https://books.google.com/books/about/Monte_Carlo_Simulation.html?hl=nl&id=xQRgh4z_5acC

Morris, K. J. C. (2021). *Reducing Neonatal Intensive Care Transports by Introducing a Regional Nurse Flex Pool*. https://essay.utwente.nl/88967/1/Morris_MA_BMS.pdf

Pelaksanaan, P., Strategis, R., Pelayanan, P., Di, K., Sakit, R., Daerah, U., Dharyanti, F., Jati, S. P., & Pramukarso, D. T. (2019). Implementation process of strategic planning at district general hospital. *Jurnal Administrasi Kesehatan Indonesia*, *7*(2). https://doi.org/10.20473/jaki.v7i2.2019.123-131

Serious Games Market Size, Share | Industry Forecast - 2030. (2022). https://www.alliedmarketresearch.com/serious-games-market

Sitzmann, T. (2011). A meta-analytic examination of the instructional effectiveness of computer-based simulation games. *Personnel Psychology*, *64*(2), 489–528. https://doi.org/10.1111/J.1744-6570.2011.01190.X

Slack, Nigel., Brandon-Jones, Alistair., & Johnston, Robert. (2013). *Operations management*. 733. https://books.google.com/books/about/Operations_Management_7th_editon_eBook_P.html?hl=nl&id=MipTBwAAQBAJ

Song, H., Tucker, A., Graue, R., Moravick, S., & Yang, J. (2019). Capacity Pooling in Hospitals: The Hidden Consequences of Off-Service Placement. *SSRN Electronic Journal*. https://doi.org/10.2139/SSRN.3186726

Song, H., Tucker, A. L., Graue, R., Moravick, S., & Yang, J. J. (2020). Capacity pooling in hospitals: The hidden consequences of off-service placement. *Management Science*, *66*(9), 3825–3842. https://doi.org/10.1287/mnsc.2019.3395

Stephens, A. S., & Broome, R. A. (2019). Impact of emergency department occupancy on waiting times, rates of admission and representation, and length of stay when hospitalised: A data linkage study. *Emergency Medicine Australasia : EMA*, *31*(4), 555–561. https://doi.org/10.1111/1742-6723.13204

Swann, J. (1999). What happens when learning takes place? *Interchange*, *30*(3), 257–282. https://doi.org/10.1023/A:1007652708139/METRICS

Taha, H. A. (2007). *Operations Research An Introduction* (Eighth Edition). Pearson Education, Inc. https://dl.icdst.org/pdfs/files3/7e932ab65f9aa3de7122b4cea3587377.pdf

Van Dijk, N. M., & Van Der Sluis, E. (2008a). To pool or not to pool in call centers. *Production and Operations Management*, *17*(3), 296–305. https://doi.org/10.3401/POMS.1080.0029

Van Dijk, N. M., & Van Der Sluis, E. (2008b). *To pool or not to pool in call centers*. *17*, 296–305. https://doi.org/10.3401/poms.1080.0029

Van Dijk, N., & Van Der Sluis, E. (2008). Pooling is not the answer. *European Journal of Operational Research*, *197*, 415–421. https://doi.org/10.1016/j.ejor.2008.06.014

van Volksgezondheid, M., & en Sport, W. (2022). *Integraal Zorg Akkoord - Samen werken aan gezonde zorg*.

van Zyl-Cillié, M. (2023). Game on! Serious games in operations research and management sciences applied to healthcare: a systemised literature review. *South African Journal of Industrial Engineering*, *34*(3), 56–67. https://doi.org/10.7166/34-3-2949

Vanberkel, P. T., Boucherie, R. J., Hans, E. W., Hurink, J. L., & Litvak, N. (2012). Efficiency evaluation for pooling resources in health care. *OR Spectrum*, *34*(2), 371–390. https://doi.org/10.1007/S00291-010-0228-X/METRICS

Varga, A. (2005). *OMNeT++ Discrete Event Simulation System User Manual*. https://doc.omnetpp.org/omnetpp/UserGuide.pdf

Wallace, R. B., & Whitt, W. (2004). *Resource pooling and staffing in call centers with skill-based routing*.

Wang, L., & Demeulemeester, E. (2023). Simulation optimization in healthcare resource planning: A literature review. *IISE Transactions*, *55*(10), 985–1007. https://doi.org/10.1080/24725854.2022.2147606

Wang, R., DeMaria, S., Goldberg, A., & Katz, D. (2016). A systematic review of serious games in training: Health care professionals. *Simulation in Healthcare*, *11*(1), 41–51. https://doi.org/10.1097/SIH.0000000000000118

Wouters, P., van Nimwegen, C., van Oostendorp, H., & van Der Spek, E. D. (2013). A meta-analysis of the cognitive and motivational effects of serious games. *Journal of Educational Psychology*, *105*(2), 249–265. https://doi.org/10.1037/A0031311

Appendix A

Feedback – Serious game poolen – Leon de Greef (in Dutch)

Sheet “Home”
- Uitleg is duidelijk en goed te volgen
- Bij het derde punt van “Basisregels” zou een afbeelding van de knoppen een toevoeging zijn om nog duidelijker te maken welke knoppen bedoeld worden, ondanks dat voor dit kleine bestand het al relatief gemakkelijk is om ze te vinden.

Sheet “Scenario 1” en “Scenario 2”
- De berekeningen lijken correct te werken en de stappen van berekenen en poolen voelen intuïtief
- De verschillende kleuren maken duidelijk dat dit een ander scenario is dan 1.
- De knoppen voor het toevoegen van variabelen en veranderen van de verdeling zijn een goede toevoeging

Sheet “Poolen”
- As-titels bij de grafieken zouden bijdragen aan de leesbaarheid.
- Er staat nog veel data wat als onduidelijk en overbodig gezien zou kunnen worden (R90:V129)
- Twee losse sheets voor poolen voor scenario 1 en 2 zou praktischer zijn om data terug te zoeken.
- Voor les 7 zou een extra sheet met grafieken het gemakkelijker maken om het verschil te zien tussen de normale en uniforme verdeling.
- De grafieken zouden prominenter in beeld mogen zijn, hierdoor is beter te zien wat het resultaat is, en daardoor is het verschil tussen de lijnen poolen en gedeeltelijk poolen beter te zien.

Appendix B

The input values per graph are in the caption. The other input values are the same in all cases, 3 servers and 40 hours per server are simulated.

Lesson 1

Both graphs show that the waiting list is significantly lower in the case of pooling.

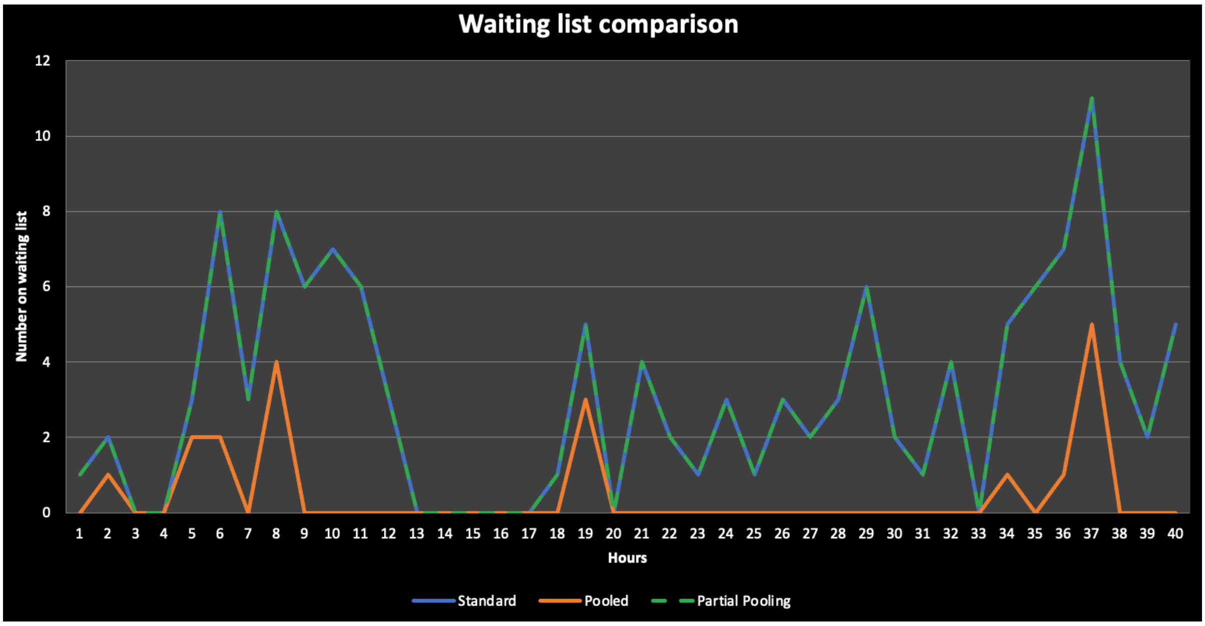


Figure 30: Demand 8-13, Capacity 9-15, Occupancy 87,5%

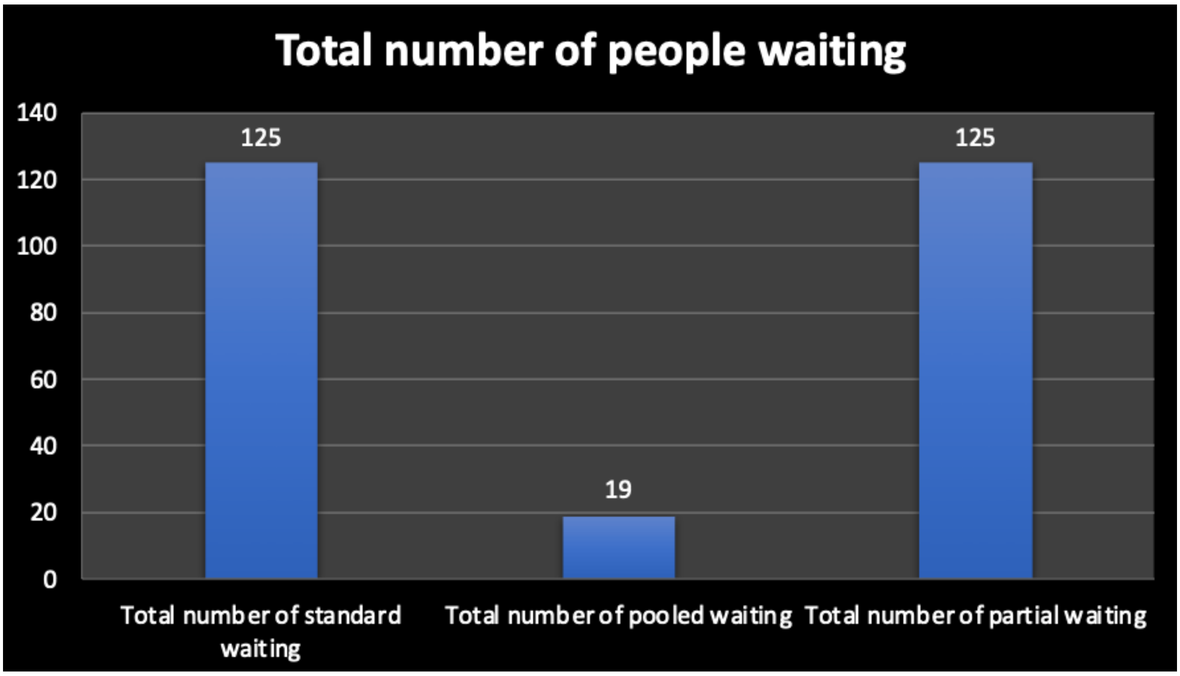


Figure 31: Demand 8-13, Capacity 9-15, Occupancy 87,5%

Lesson 2

In the graphs below, the occupancy rate is 91,3% and in the graphs of lesson 1, the occupancy rate is 87,5%. Lesson 2 therefore shows that increasing the occupancy rate causes an increase in the waiting list and more fluctuations in the lines. These fluctuations show that there are more workload fluctuations.

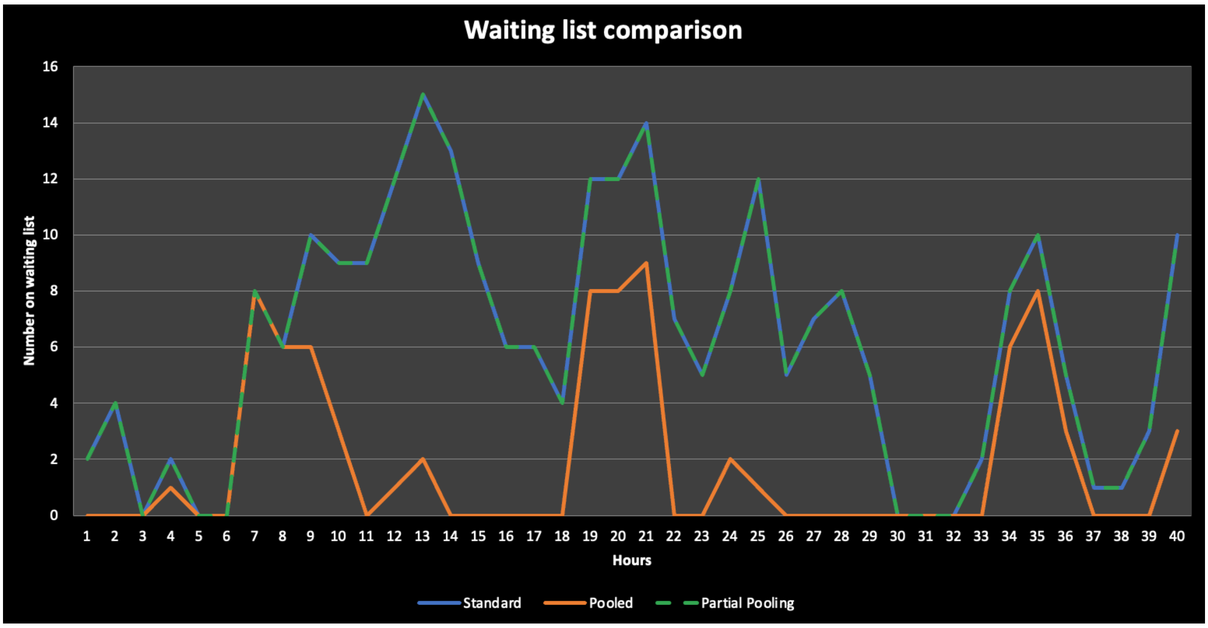


Figure 32: Demand 8-13, Capacity 8-15, Occupancy 91,3%

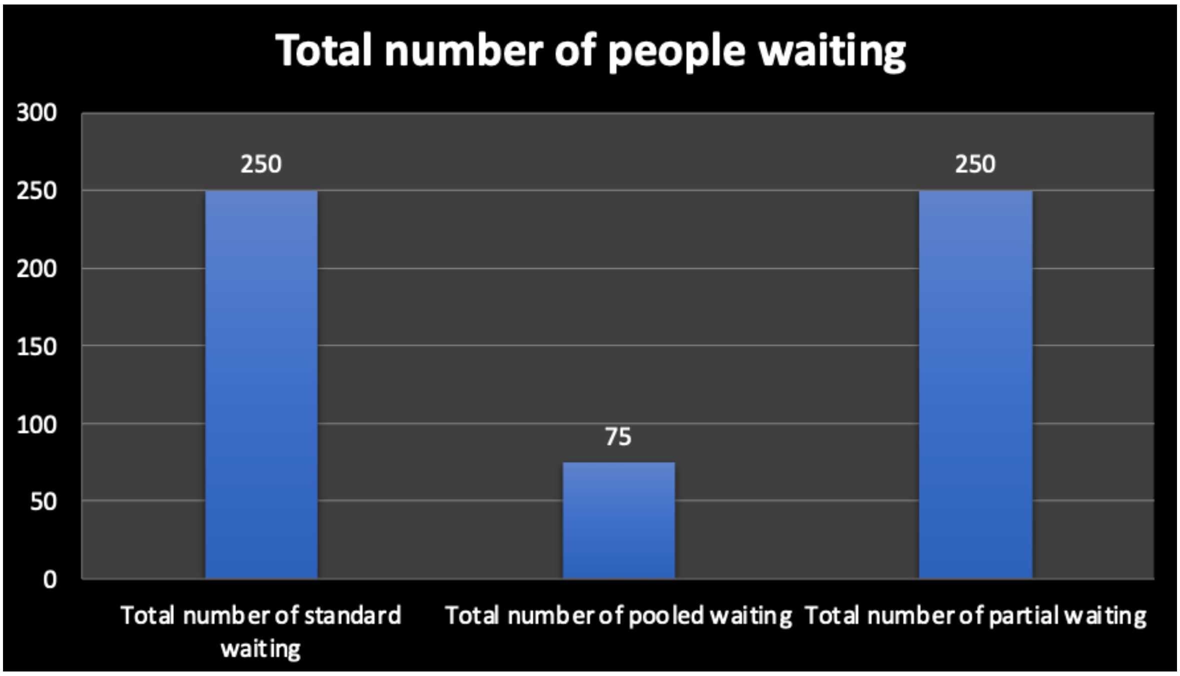


Figure 33: Demand 8-13, Capacity 8-15, Occupancy 91,3%

Lesson 3

The first graph is the same as in lesson 1, in the second graph the occupancy rate is the same, but more variability is added. This is done by lowering the lower limit of the capacity by 2 and increasing the upper limit of the capacity by 2. This change makes the line more erratic, which means more fluctuations in workload. In addition, the line is higher for a longer period of time, which means that the waiting list has increased.

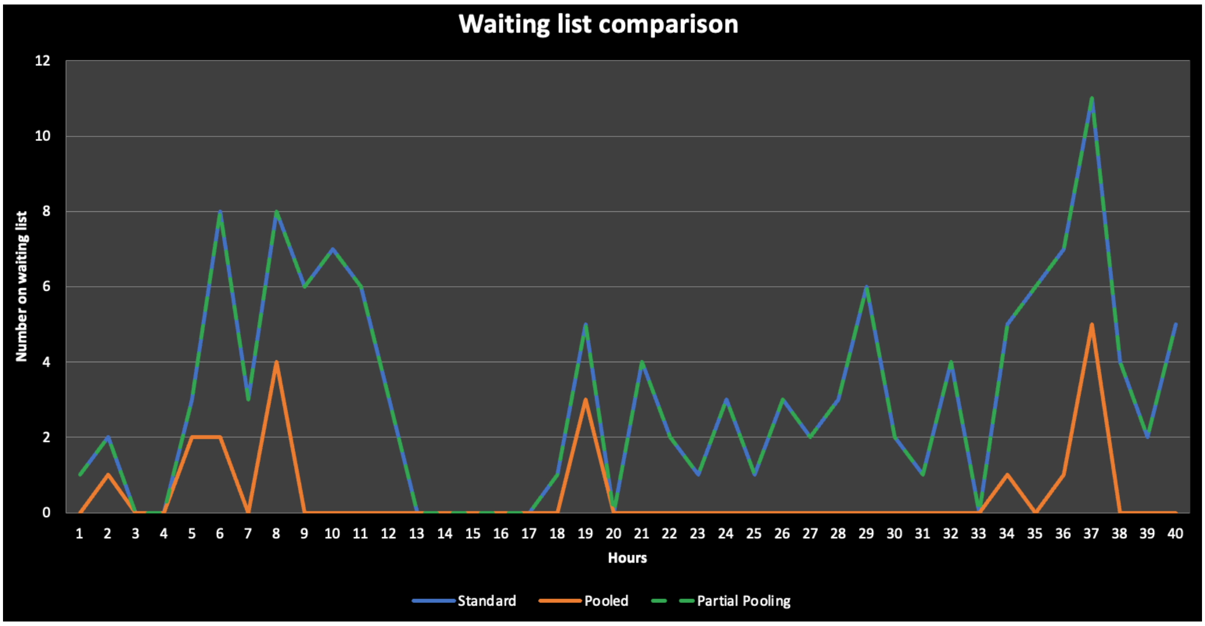


Figure 34: Demand 8-13, Capacity 9-15, Occupancy 87,5%

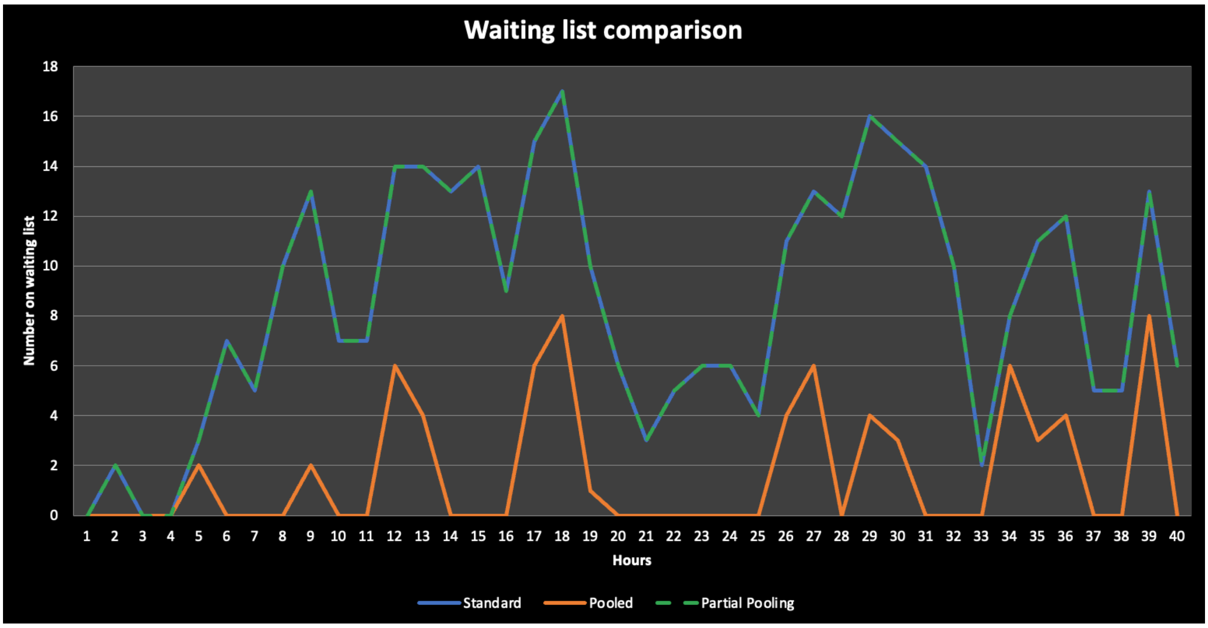


Figure 35: Demand 8-13, Capacity 7-17, Occupancy 87,5%

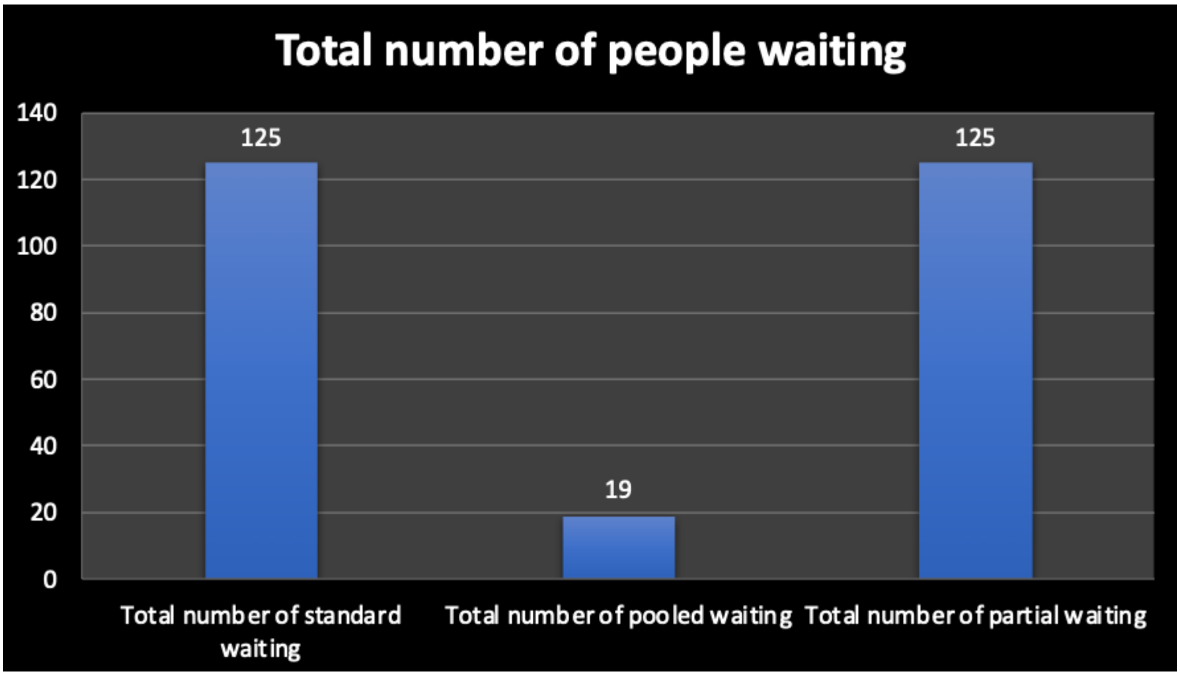


Figure 36: Demand 8-13, Capacity 9-15, Occupancy 87,5%

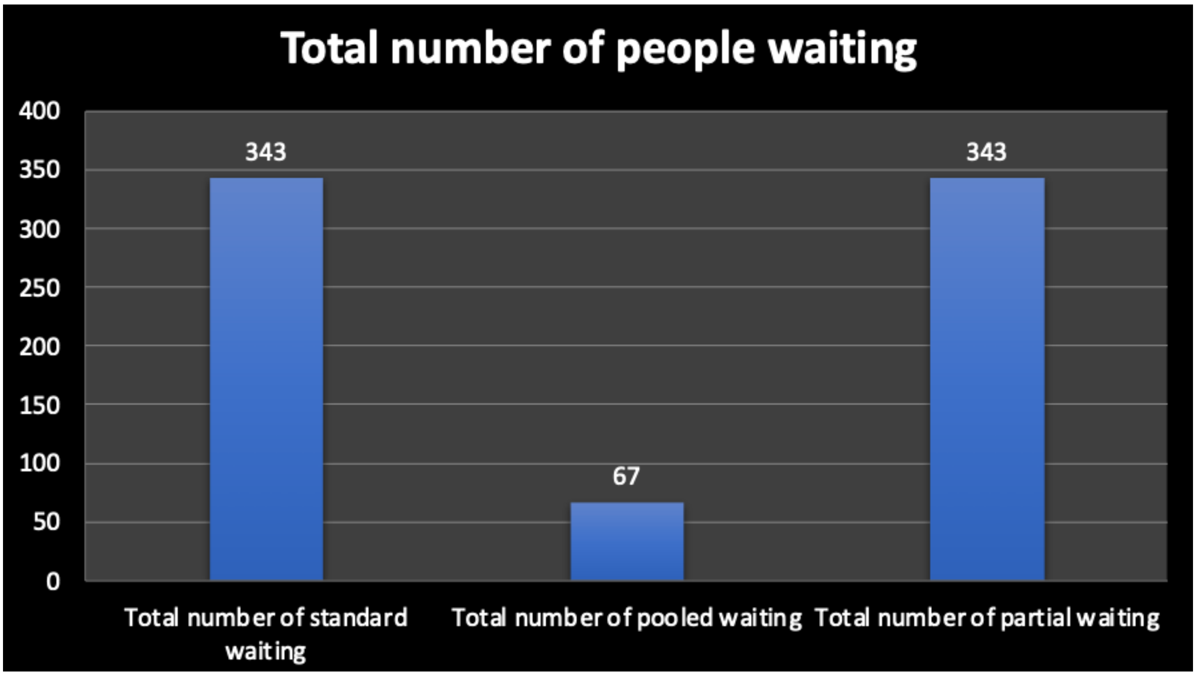


Figure 37: Demand 8-13, Capacity 7-17, Occupancy 87,5%

Lesson 4

In both the line graph and the bar chart you can see that with a flexible pool of 1 per server, the waiting list is already a lot lower. It also shows that a small amount of flexible capacity can have a big impact, 'a little flexibility goes a long way'.

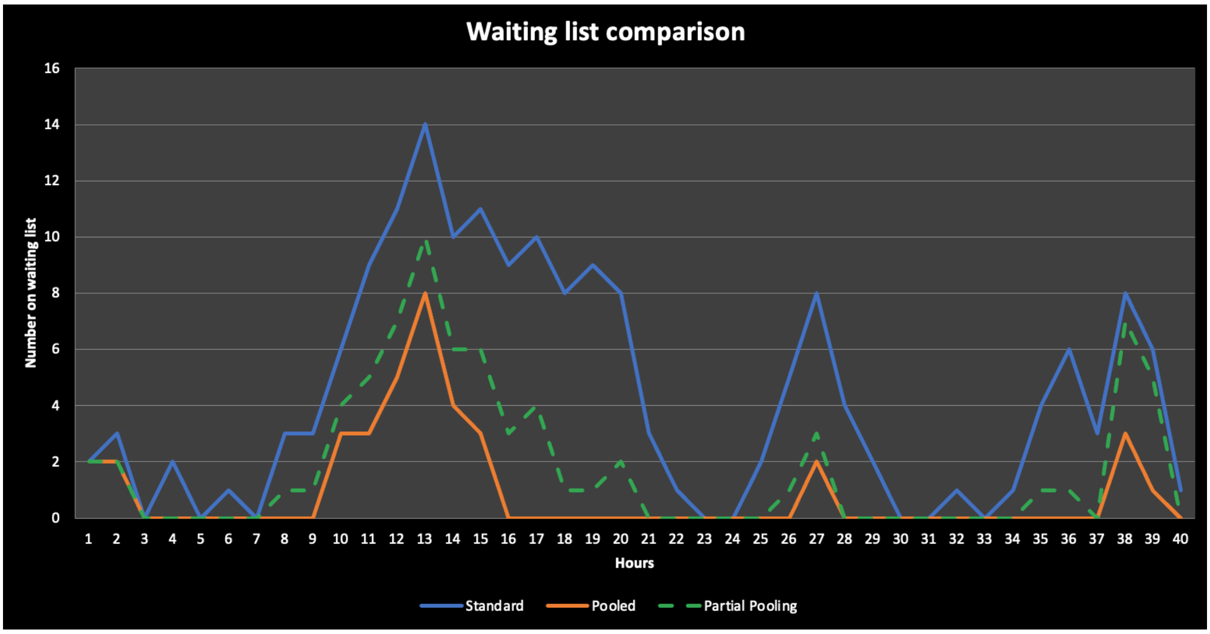


Figure 38: Demand 8-13, Capacity 9-15, Occupancy 87,5%, Flexible Capacity 1

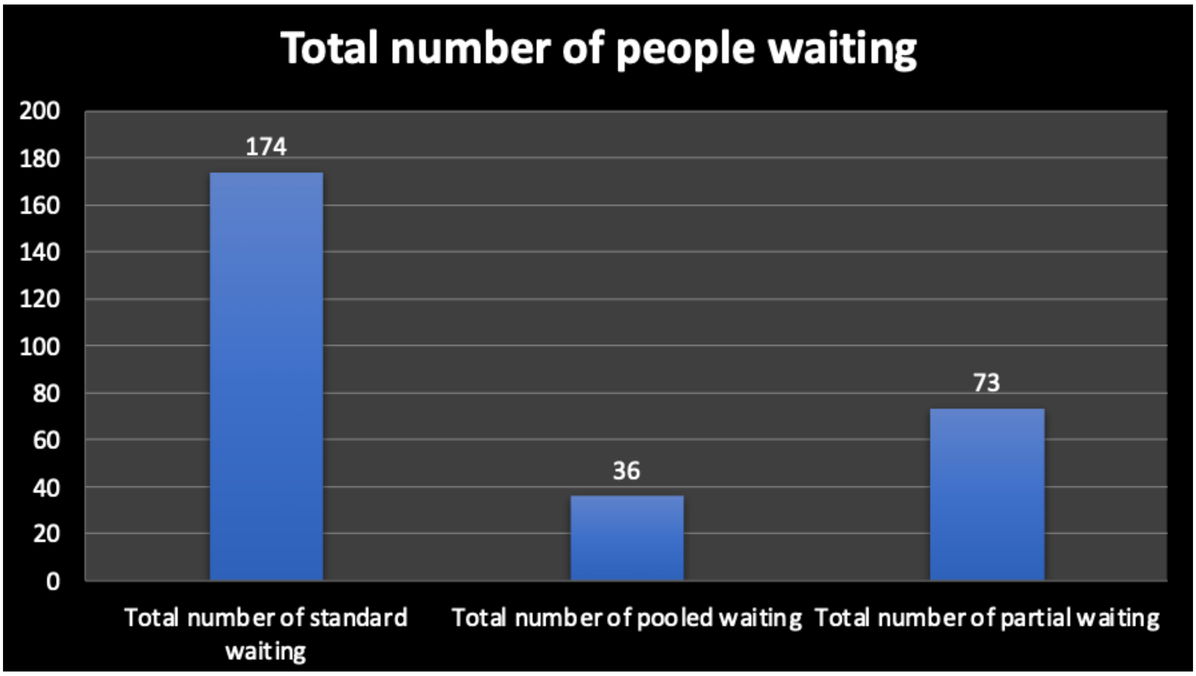


Figure 39: Demand 8-13, Capacity 9-15, Occupancy 87,5%, Flexible Capacity 1

Lesson 5

This graph shows that when you start with an initial waiting list of 5 per server, it takes 5 days to clear this in the standard situation. While in the pooled situation it only takes 3 days. You can also see that because you have an occupancy rate that is below 100%, you have room to clear the waiting list in a short time.

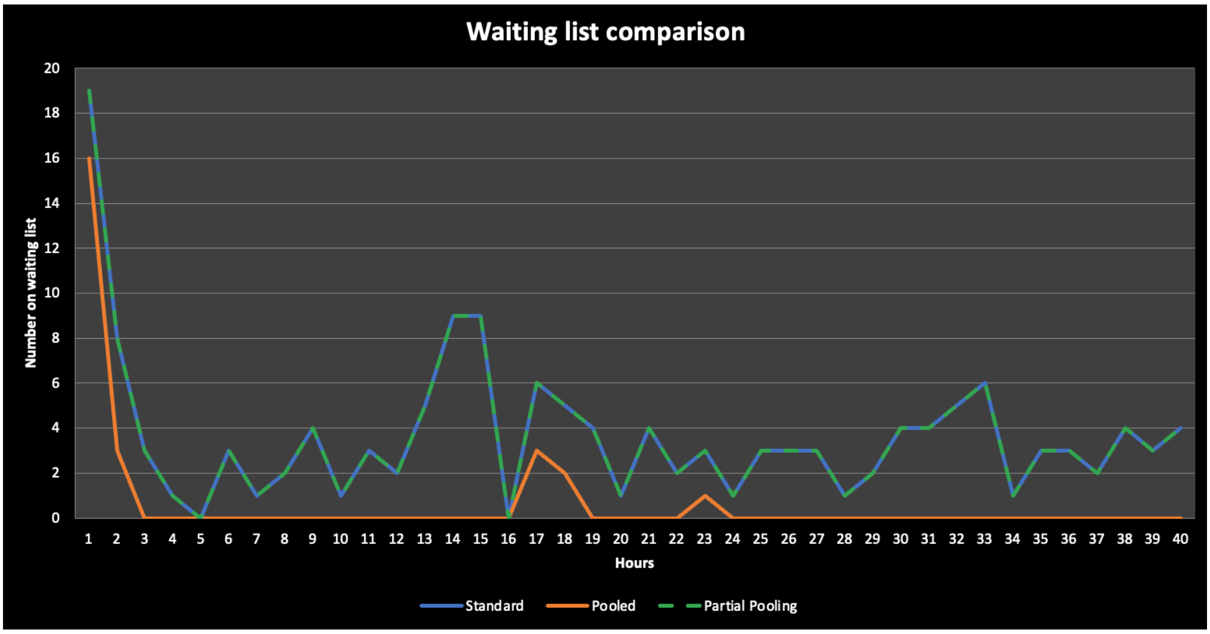


Figure 40: Demand 8-13, Capacity 9-15, Occupancy 87,5%, Initial Waiting List 5

Appendix C

Lesson 8

After performing the steps in lesson 8, it is clear from both the line graph and the bar chart that the average number of days waiting is lower in the split situation. In the line graph, you can see that the blue line is below the orange line most days. This means that on most days the split variant was better than the pooled. In the bar charts you can clearly see that the average number of days waited over the total is also clearly lower in the split situation.

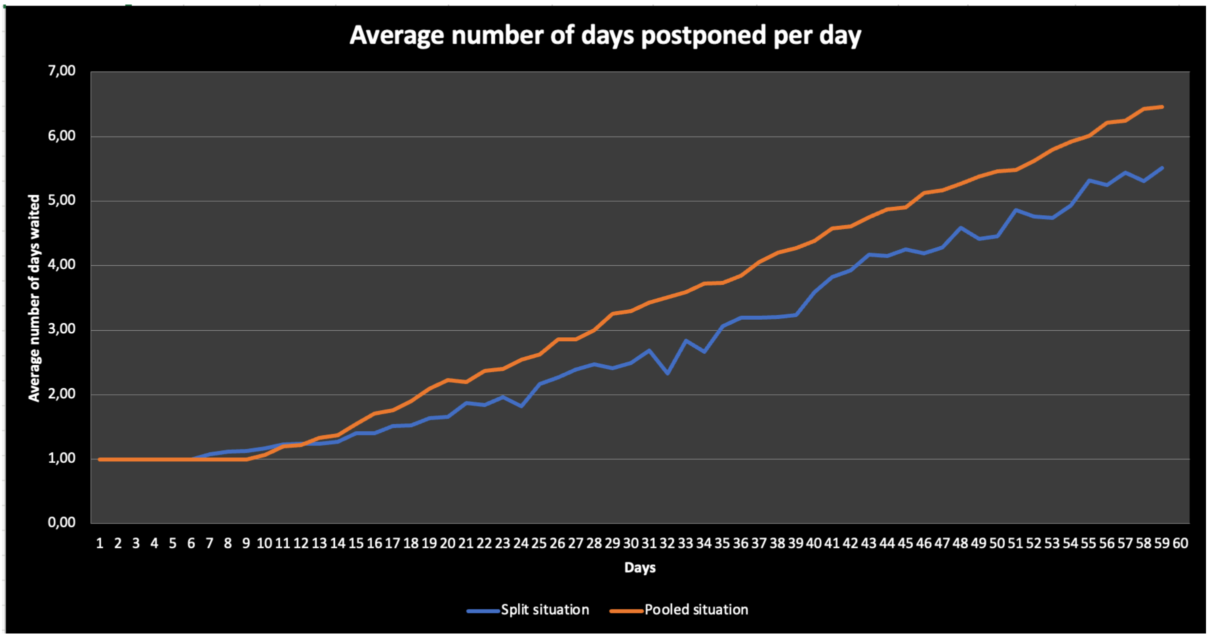


Figure 41: Line graph first simulation

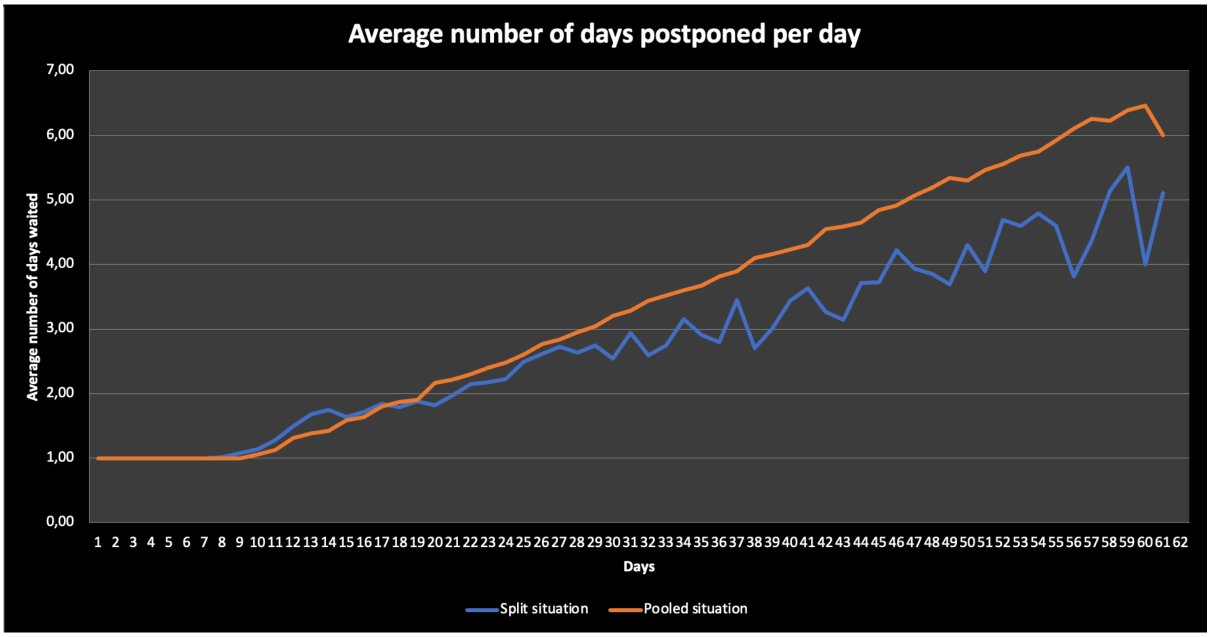


Figure 42: Line graph second simulation

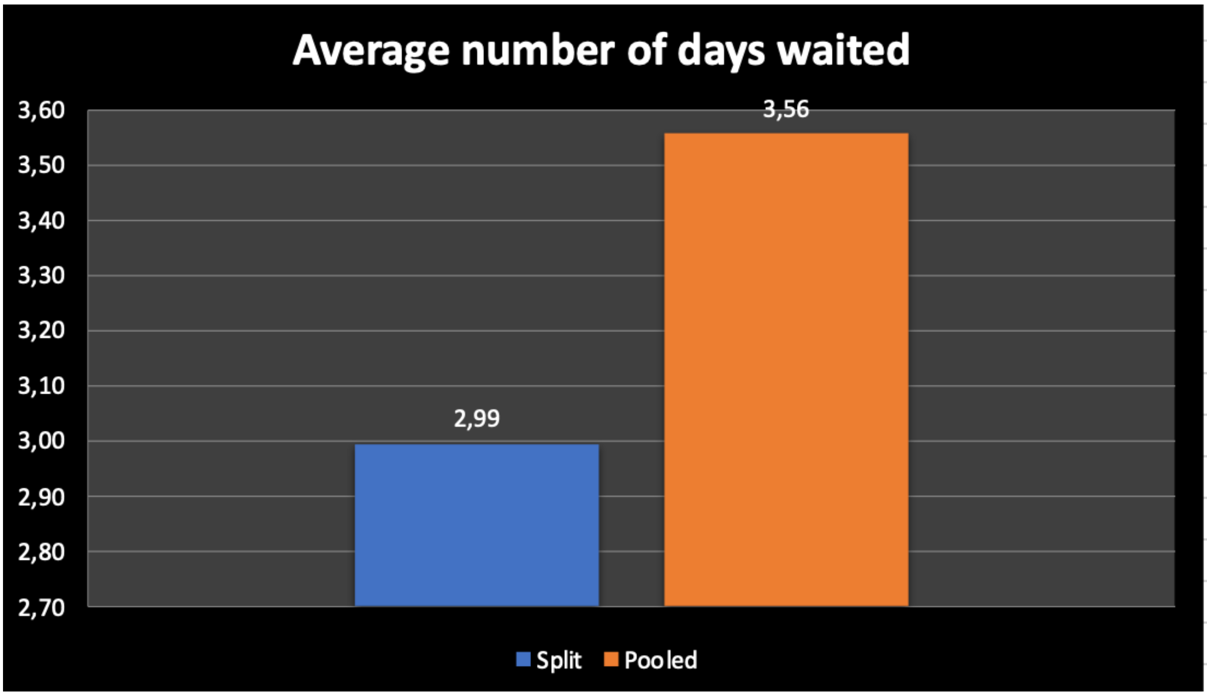


Figure 43: Bar chart first simulation

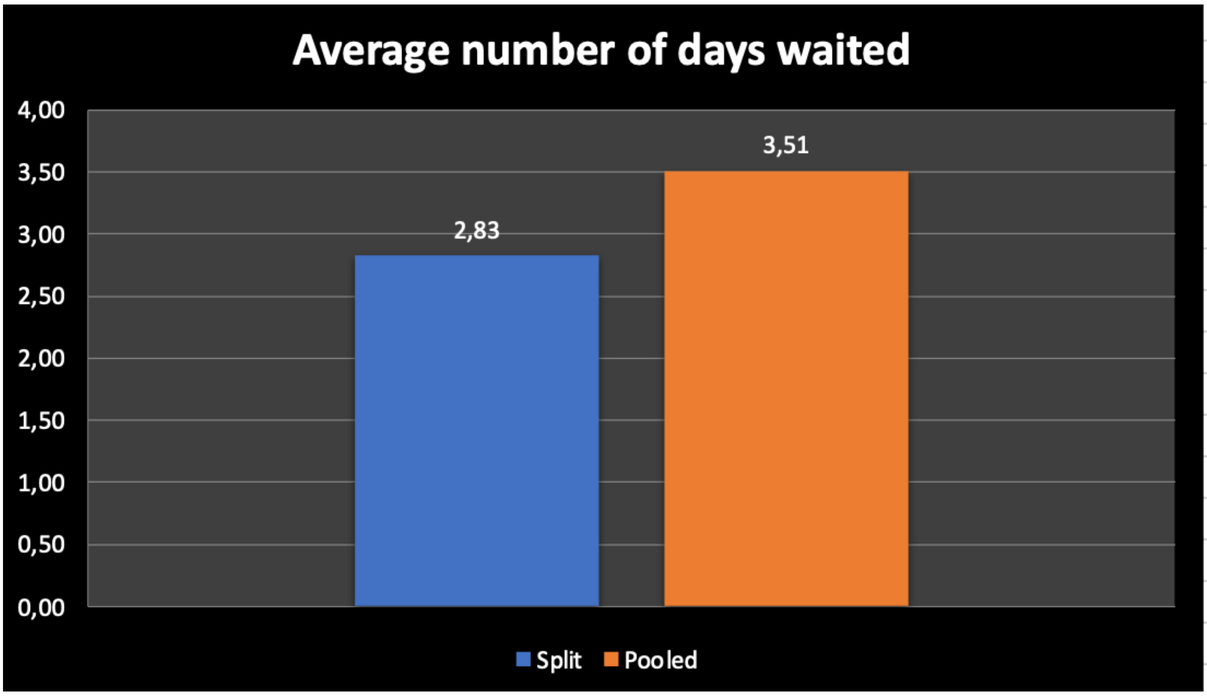


Figure 44: Bar chart second simulation

Lesson 9

Lesson 9 shows that slightly increasing the percentage of long-term patients (increased from 27 to 33) has a major impact and that pooling is now more effective than split scheduling. In both line graphs you can see that the blue line is much higher than the orange one, so almost every day patients wait longer in the split situation. In the bar graphs, you can see that the average in the split variant is much higher than in the pooled variant. So overall you see that there are situations where pooling is not more beneficial, but these are quite unique situations.

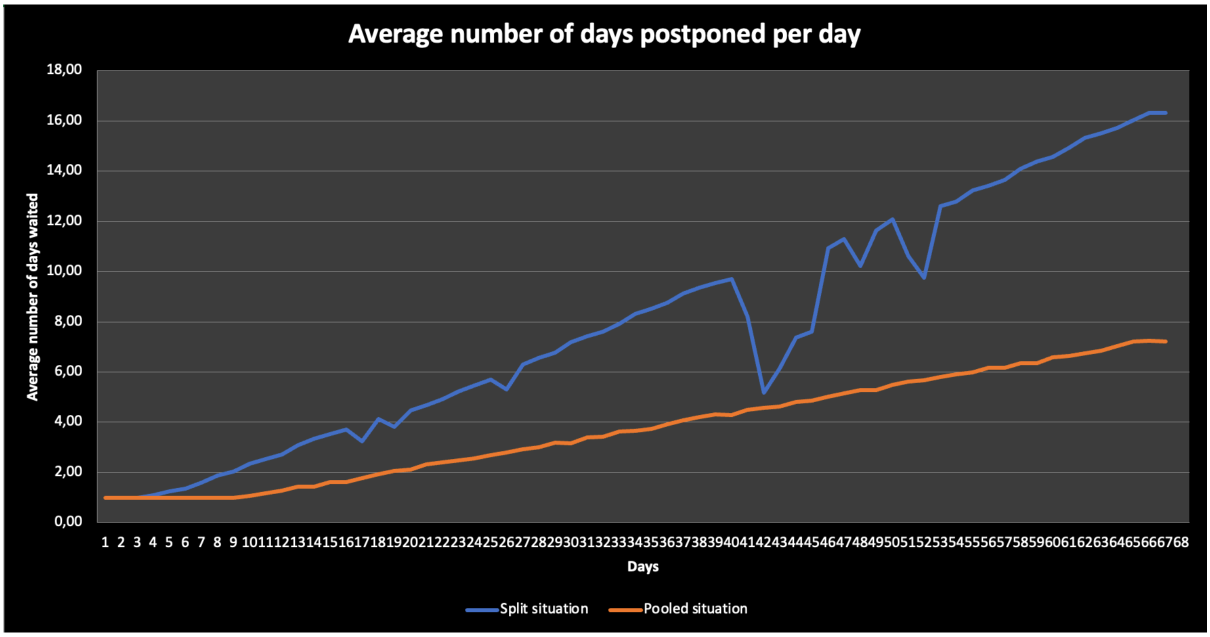


Figure 45: Line graph first simulation

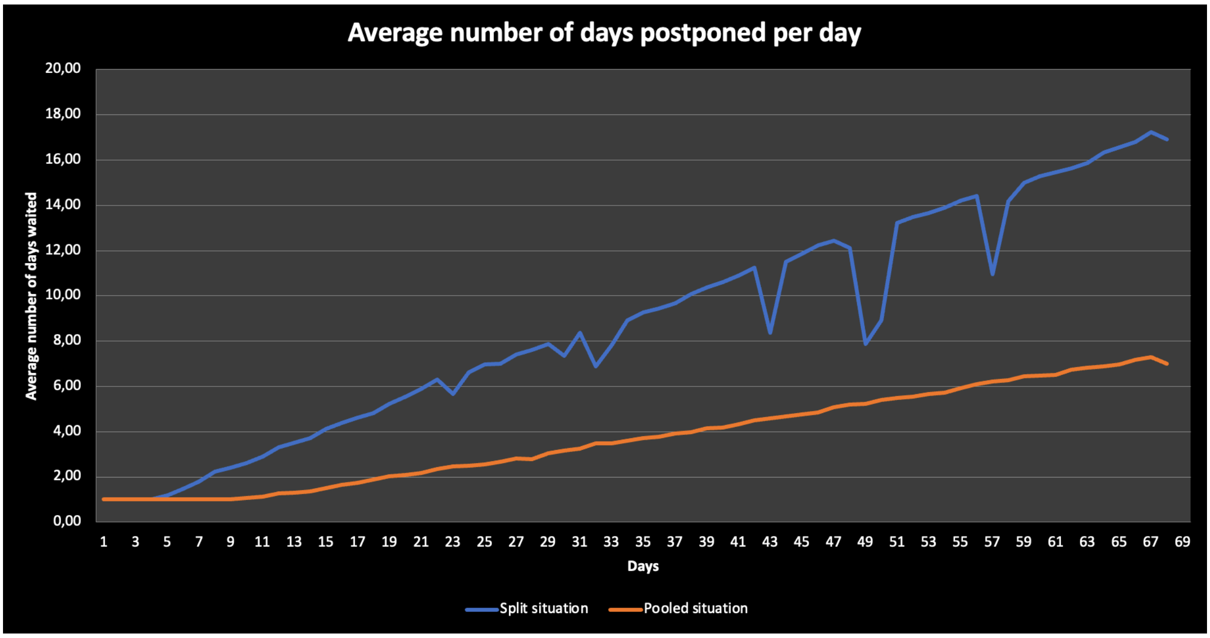


Figure 46: Line graph second simulation

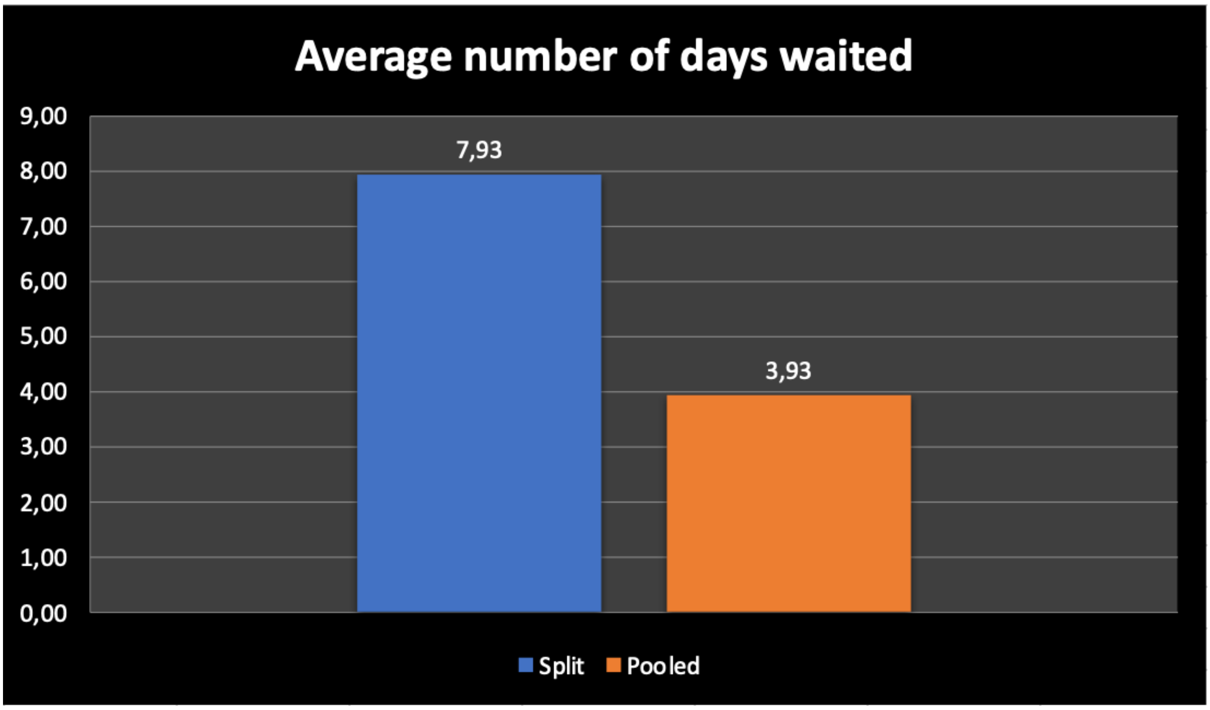


Figure 47: Bar chart first simulation

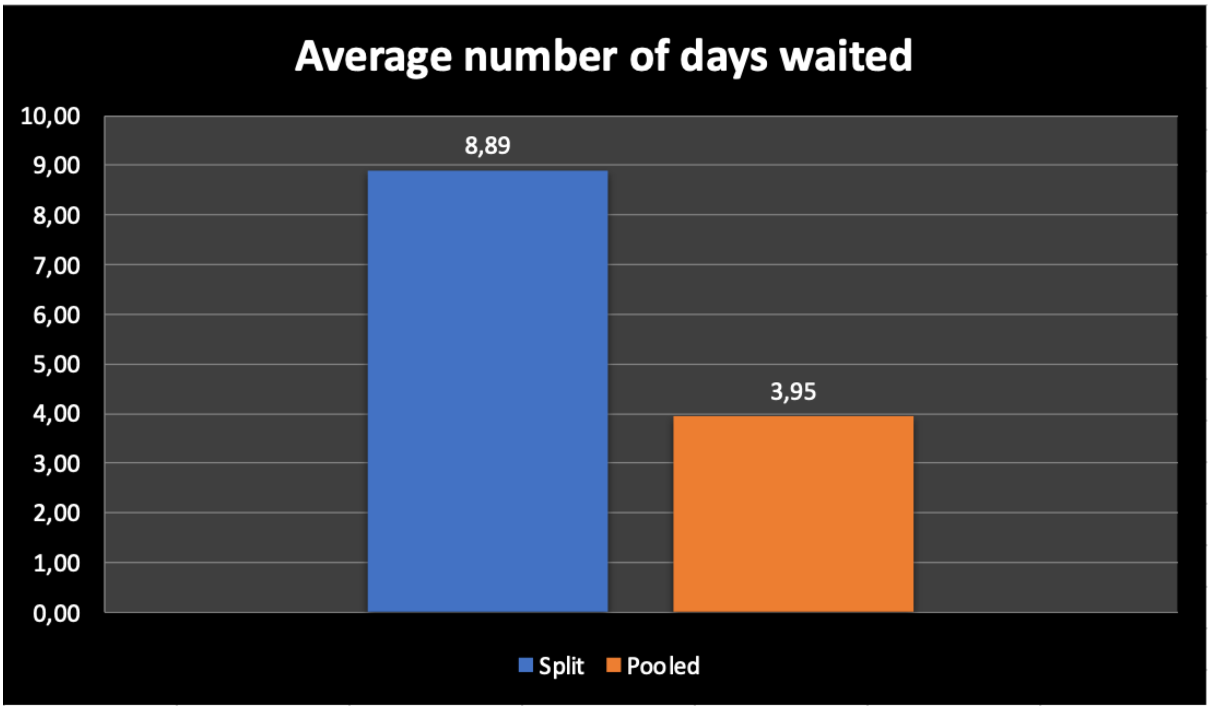


Figure 48: Bar chart second simulation