T.YAVUZ

M.S. Thesis

2015

AN OPTIMIZATION-BASED DECISION SUPPORT SYSTEM FOR DORMITORY PLACEMENT AT IŞIK UNIVERSITY

TONGUÇ YAVUZ

IŞIK UNIVERSITY

2015

AN OPTIMIZATION-BASED DECISION SUPPORT SYSTEM FOR DORMITORY PLACEMENT AT IŞIK UNIVERSITY

TONGUÇ YAVUZ

B.S., Industrial Engineering (IE), 2013

M.S., Industrial Engineering - Operations Research (IEOR), 2015

Submitted to the Graduate School of Işık University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Industrial Engineering - Operations Research

IŞIK UNIVERSITY

2015

IŞIK UNIVERSITY

GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

AN OPTIMIZATION-BASED DECISION SUPPORT SYSTEM FOR DORMITORY PLACEMENT
AT IŞIK UNIVERSITY

TONGUÇ YAVUZ

APPROVED BY:

Assist. Prof. Dr. S. Tankut ATAN ______________________

(Thesis Supervisor)

Industrial Engineering- Işık University

Assist. Prof. Dr. Mehmet ÖNAL ______________________

Industrial Engineering- Işık University

Assoc. Prof. Dr. Taner ESKİL ______________________

Computer Engineering- Işık University

APPROVAL DATE: 16 / 12 / 2015

AN OPTIMIZATION-BASED DECISION SUPPORT SYSTEM FOR DORMITORY PLACEMENT AT IŞIK UNIVERSITY

ABSTRACT

Placing students to dormitories is a yearly, routine activity at universities. Until recently, this process has been accomplished manually at Işık University. The manual process had several disadvantages: it was slow, it was difficult to update assignments when last minute changes were needed, and most importantly considering all students' requests and finding an assignment with as few complaints as possible was difficult. Furthermore, with increasing number of students, the manual process was expected to become more and more difficult to cope with.

Due to above reasons, the whole dormitory application process has been streamlined and the manual process has been replaced by an Excel-based decision support system which uses mathematical programming, and Gale and Shapley algorithm. During the design process, regular meetings were conducted with the Housing department and other stakeholders such as the Student Council to decide about student placement criteria and their importance. In collaboration with the university's IT department a web-based application form has also been created to collect the data necessary for the system. The system has successfully been used in 2014 and 2015.

KARAR DESTEK SİSTEMİ İLE IŞIK ÜNİVERSİTESİ YURT YERLEŞTİRME ENİYİLEMESİ

ÖZET

Yurtlara öğrenci yerleştirme, kampüs üniversitelerinin yıllık rutin etkinliklerinden olup, aynı zamanda karşılaşılan en önemli problemlerden biridir. İstanbul şehir merkezinden 60 km uzakta, Şile'de yer alan FMV Işık Üniversitesi de bir kampüs üniversitesi olup, gerek İstanbul içinde yaşayan gerekse de İstanbul dışından gelen öğrenciler için 9 farklı çeşitte yurt imkanı sağlamaktadır. Tüm bu yurtlara öğrenci yerleştirme işlemi şimdiye dek elle yapılıyordu fakat bunun da çesitli zorlukları vardı. Öncelikle yerleştirme çok uzun sürüyor ve son dakika değişiklikleri gerektiğinde bu güncellemeleri yapmak oldukça zor oluyordu. Bunun yanında, tüm ögrencilerin isteklerini mümkün olduğunca karşılamak ve süreci en az şikayetle tamamlamak da cok zordu. Son olarak da öğrenci sayısının artan fakülte ve bölümler ile yıldan yıla artmasıyla, elle öğrenci yerleştirme süreci başa çıkılamaz bir hal aldı.

Biz de çalışmamızda, elle yerleştirme işlemini, matematiksel modelleme ve Gale Shapley algoritması kullanan Excel tabanlı bir karar destek sistemi ile değiştirdik. Sistemi tasarlama sürecinde, yerleştirme kriterlerini belirlemek ve önem sıralarını ayarlamak amaçlı, Yurtlar Müdürlüğü ile düzenli görüşmeler yaptik. Sonuç olarak da, tüm yurt yerleştirme sistemi baştan düzenlendi. Sistem için gerekli verileri toplamak amacıyla da, Bilgi İşlem Daire Başkanlığı'nın da katkısıyla internet tabanlı bir başvuru formu oluşturuldu. 2014-2015 ve 2015-2016 akademik yıllarında kullanılan sistemimiz sonucu alınan raporlar, geliştirdiğimiz projenin üniversite kural ve politikalarına uygun bir şekilde yerleştirme yapıldığını ve öğrenci memnuniyetinin geçmiş yıllara göre arttığını göstermiştir.

To my family,

Acknowledgements

There are many people who helped to make my years at the graduate school most valuable. First and foremost, I would like to express my profound gratitude to my project supervisor, Assist. Prof. S. Tankut Atan, for involving me to his study, his great concern, invaluable support, patience and continuous guidance to prepare this project. Having the opportunity to work with him over the last year was intellectually rewarding and fulfilling.

Additionally, I am extremely grateful to the foundation, Işık University and especially Industrial Engineering Department Staff, who have always been helpful during my whole master and bachelor education. I also would like to thank to Secretary General department, specifically Aziz Genç, Secretary General for giving me to study this subject. Other department in Işık University that I would like to express my profound gratitude to is Dormitory Department, specifically the Dormitory Department Head, Selcen Gray and Assistant Manager, Murat Çelik. Çınar Tur Shuttle Service and Department of Information Processing Center were ones of the most heplful departments in the university for collecting data in my thesis. I would also like to thank to my graduate student colleagues who helped me all through the years full of class work and exams.

Also, I am extremely grateful to my family who has always trusted me and has great role on my personal development against to difficulties of life. I thank my parents Halil Mansur Yavuz and Seval Yavuz for their patience and endless encouragement. There were many things to learn from them, and there will always be.

TABLE OF CONTENTS

LIST OF GRAPHS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

At the beginning of each academic year, dormitories face a hard decision problem, namely allocation of students to rooms. The allocation directly affects students' happiness because they spend a significant amount of time in their rooms. Moreover, the rooms do not come for free; as renters, students want to have their wishes fulfilled as much as possible. Thus, university housing departments try to fulfill students' wishes and at the same time to fill the available rooms as much as possible.

The literature on underlying Operations Research topics such as assignment problem, matching problem, etc. is extensive. However, there are not many articles reporting real cases about dorm allocation. This thesis describes the automation process of the dorm allocation problem at Işık University. The process involved development of an optimization-based decision support system along with a development of a web-based application form.

When the demand for a certain type of a dorm room does not exceed the number of such rooms available, all students will get a room that satisfies their wishes. Unfortunately, this is impossible in the real world. In many cases, dormitories' capacity is smaller than the demand, or students' preferences are too complicated to fulfill. For that reason, dormitory allocation needs a solution method. Işık University also faced the same situation because the allocation process was manual and cumbersome.

Işık University is one of the first private universities in Turkey. It was founded in 1996, and today has reached a size of five faculties with 34 undergraduate programs, 2 vocational schools with 20 associate degree programs, 2 graduate institutions, a school of foreign languages, and a center for continuing education. University owns two campuses in Şile and Maslak. The student population is about 6,000, and more than half of them study in Şile campus. Maslak campus is located in the city center, but Şile campus is 60 km from the center where staying in dormitories is a preferred option even for students from İstanbul. The demand for dormitories has been steadily increasing over years.

1.1 Işık University Dormitories

Işık University's main campus is located in Şile. Additionally, dormitories are placed in this campus. Due to reasons such as long distance from the city center, the desire to live with friends in a campus atmosphere, to be able to focus more on the courses, to have less stress and clean air and to be close to nature in a forested area and the Black Sea, many students prefer to stay in dormitories. Graph 1.1 shows the increasing demand for dormitories over years.

Additionally, this increase in demand has different reasons such as increasing number of faculties and departments. We can summarize the meaningful reasons for the students to stay in dormitories and the reasons of the increasing demand as:

- About half of the students' home address is not in İstanbul, so they choose to stay on campus.
- Part of students' home address is İstanbul but they come from the other districts that have serious traffic problem; Şile campus is about 60 km to city center. Thus,

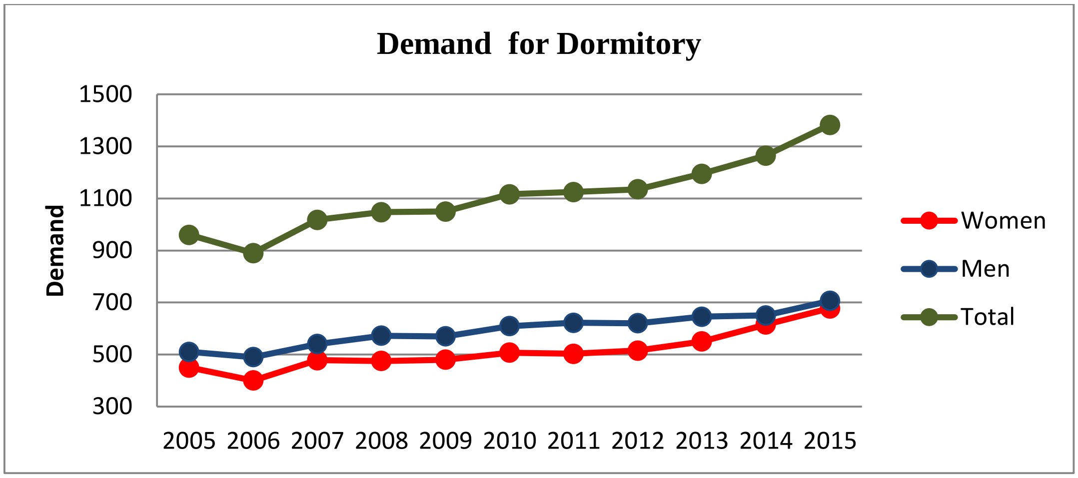
the students spend more than 1.5 hours on average to reach the campus. For this reason, even the students who live in İstanbul still prefer to stay in the dormitories rather than to commute to İstanbul.

- The dormitories provide good resources like library, sport center, dining hall etc. So staying in dormitory is very popular and a good choice for students. They can stay with their friends, go to dinner together, and they do not have to deal with housework such as cooking or cleaning. In addition to these, the students do not have to worry about electric, water, or heating bill. They have more free time to study, to rest, to spend time with their friends or to allocate time for their hobbies.

- Işık University dormitories provide a less stressful life. It is surrounded by the Black Sea and Şile Forest. So, it is the best place for clean air. Many other people prefer to come to Şile to relax in their weekend time or holidays.

- The university also plans to open new faculties and departments in the future, so the demand for dormitories is predicted to increase more with the increasing number of students.

At the moment, Işık University has three dormitory types: Orange, Red and Blue. They have some differences such as capacity, location and style of rooms. But commonly, all dormitories provide TV rooms, studying rooms, fridge in all rooms, free distribution of water on each floor, vending machines, laundries with some washing machines and drying machines, and kitchen with oven, kettle, microwave, a tiny dishwasher with some cooking utensils such as saucepans.



Graph 1.1 Demand for dormitory

The main difference in room types is the room capacities that change from two to four. Except Red A-B blocks, all rooms have private bathrooms. Other types of rooms have minor differences. The capacity and price differences are shown on Table 1.1.

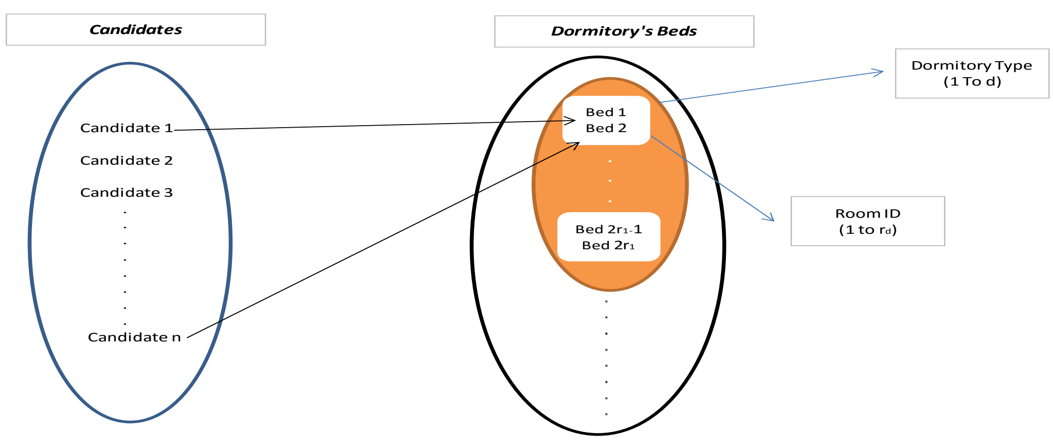
Table 1.1 Dormitory types, capacities and prices

Dorm Type	Block	Capacity	Private Bathrom	Price(2015)
Red	A-B	2	-	6,250 TL
Red	A-B	3	-	4,100 TL
Red	A-B	4	-	3,850 TL
Red	C-D	2	+	6,500 TL
Red	C-D	3	+	5,800 TL
Orange	All	2	+	8,000 TL
Orange	Floor	2	+	6,900 TL
Blue	All	2	+	5,500 TL
Blue	All	3	+	4,000 TL

Each dormitory type has different number of floors. On each floor, there are four rooms on average. Red A-B dormitories have separate bathrooms on the floor, but each room has their own bathroom as well. But, other dormitories have their own private bathroom inside the rooms. Additionally, the Blue dormitories have sea view, and the Orange dormitories have their own kitchen in the rooms. Students specify their wishes based on these differences. Some of them want to stay in bigger rooms and in a crowded environment with their friends and choose less expensive rooms. Some of them want their private bathroom, but some of them accept a bathroom outside. Likewise, some students want to have a sea view, and some of them want to cook in their rooms.

1.2 Problem Definition

The Dorm Allocation Problem is not only Işık University's problem. It is faced by the most universities. Basically, we have a matching problem with two sets: beds and students (Graph 1.2). As mentioned before, generally dorm rooms have some differences. According to these diversities, students' choices show varieties, so we need to find a good matching.



Graph 1.2 Sets and subsets

For example, let’s look at a bed which has some properties about gender (which gender can stay), dormitory, room, room capacity and position. These properties show us the bed differences. University Housing Department creates a code for each bed to identify. For instance, 1MOA11 means that bed’s index is 1, only male student can stay in this bed, it belongs to orange dorm type, it is in A block, its position is 1, and it is inside room 1. Işık University asks students’ preferences such as dorm type, room type and specific wishes. For instance, first male candidate in 2014 wanted to stay in Blue dorm-A block, with four beds room. Additionally, he preferred to share the same room with candidate 5, 6 and 87.

Until 2013-14 academic year, the capacity of dormitories was more than the number of applications. That means the university could offer a room to all candidates. Still, it was very hard to satisfy all students’ wishes with this manual process. And, some of the wishes were impossible to meet. For example, candidate 9 wanted to stay with candidate 10, and she preferred to stay in Orange. But candidate 10 did not choose Orange. In addition to this, the Housing Department could not fulfill some demands despite they were possible to satisfy.

In the past years, after allocations were announced by the Housing Department, many students objected to the results, and the department tried to reallocate them. But, it was harder to allocate again after announcing the results because the academic semester starts almost right after the announcement date, so students need to move in. Thus, it is really important to get the allocations right on the first time.

In 2014, university population reached 5,000 students. Therefore, students’ demand for dormitory has increased, and the university needed changes to make the placement process better and to satisfy the demands of all students for accommodation. This matching procedure can be analyzed from two different perspectives: from the university and from the candidate. Let us examine the process as viewed by the university first.

1.2.1 The University

The university has certain preferences for choosing dormitory students related to their gender, age, home address, etc.

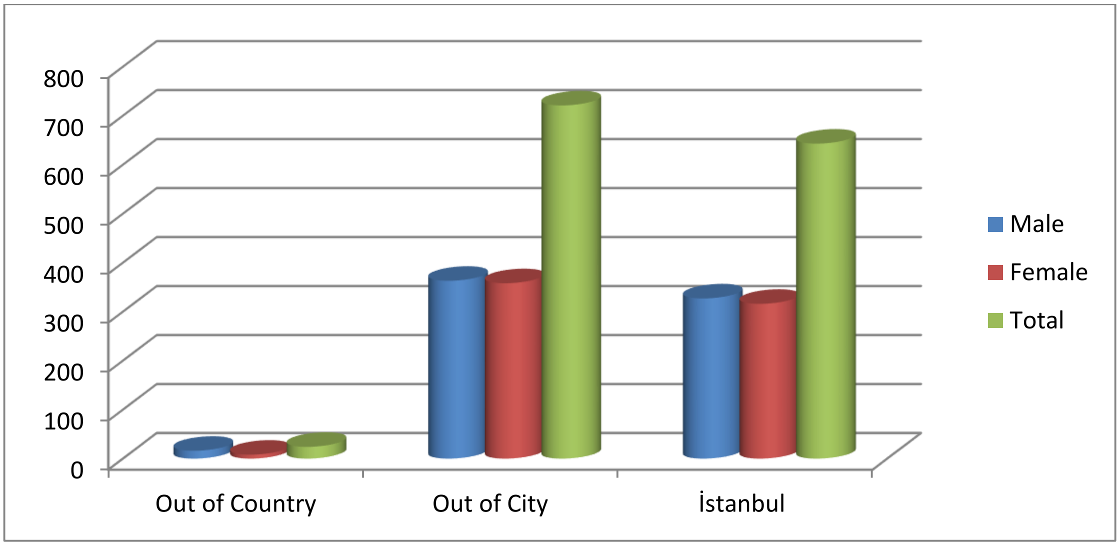
<u>Gender:</u> The University wants to give precedence to female candidates. It means that if a male and female candidate have the same conditions (class, entrance year, home address etc.), the female candidate will be placed. Normally, the dorm blocks are separated by gender. In male blocks, female students cannot live. Similarly, in female blocks, male students cannot live. But, if needed, some male blocks can be reclassified as female blocks.

<u>Home Address:</u> The most important placement criterion for the Housing Department is students' home addresses. The priority is finding rooms for students who are not from İstanbul. As it is seen on Graph 1.3, more than half of candidates' home addresses are out of İstanbul. The university is aware of the fact that these students encounter problems for the orientation to a big city and can feel alienated with potentially serious effects on their psychology. University years are years that a person might never want to forget, but these years can also be years that the same person does not want to remember at all.

After placing these students, the department starts to place the students whose home address is furthest from the campus first.

<u>Class:</u> Another important criterion about dormitory placement is giving precedence to new students and senior students. The department believes that staying in dormitory helps new students in adapting to university life. They can be more social and have more chance to meet new people and find new friends. In addition to the new students, the department also thinks that senior students must have priority to stay on the campus.

<u>Academic Success:</u> According to university policies, students are encouraged to follow a double major or a minor if their academic success permits. In support for their academic ambitions, the university gives precedence to such students in placement.



Graph1.3 Number of students according to their hometown

1.2.2 The Candidate

Students also have preferences regarding room types and roommates.

Roommate: Nearly 65% of candidates want to share the same room with a definite friend or friends. It is the most important wish for most of the students. Normally, students have two ways of specifying their wishes: he/she stays without choosing their roommate but in the desirable dormitory, or with a chosen roommate but in an undesirable dormitory type. Most of the candidates prefer choosing their roommate. Even when students do not have a specific friend in mind to stay with, they can still have a preference for age such as freshman, or a non-smoking roommate.

Room Type: As we mentioned before, there are nine different types of rooms. They have lots of differences in terms of prices, capacities and architecture. The most important factor in students' decisions is price. Generally, candidates prefer to stay in cheaper dorms, but few of them prefer to stay in more expensive rooms because of the rooms' features such as private bathroom or kitchen.

Other Wishes: The Housing Department faces some other preferences such as a specific room, or neighbor. The department tries to satisfy those wishes also.

Let us summarize all information above. Our goal is to allocate students to rooms and beds by taking university policies and students' wishes into consideration. There is a tradeoff between these two perspectives. For instance, a student may deserve to stay in a dorm according to his/her home address, but he/she chooses to share the same room with a student who lives in İstanbul close to the campus. Figure1.1 summarizes the university's and students' point of views and what is important to them.

In summary, the dorm allocation is a multidimensional and complex problem where we need to consider students' preferences and university's policies. We also desire to reach solutions within an acceptable time.

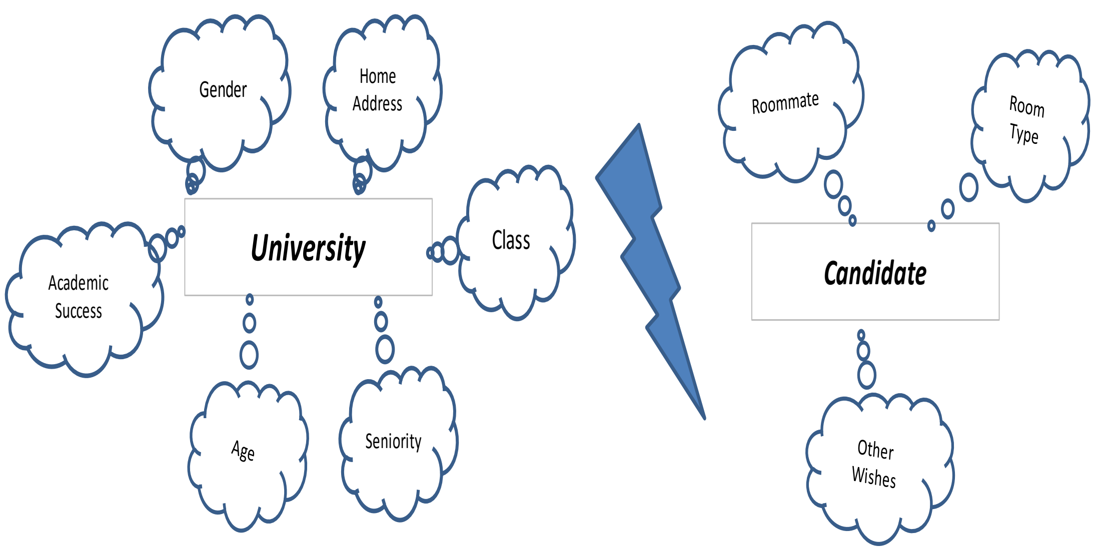


Figure 1.1 Summary of university and candidate preferences

Our goals going into the project for solving this complex problem were as follows:

- Maximization of bed utilization,
- Fulfilling students' demand for room types,
- Fulfilling students' demand for roommates,

Next, a literature review is provided in Chapter 2. Chapter 3 gives the description of the manual process which was replaced with the decision support system and process that was developed as part of this thesis. Relevant data is discussed in Chapter 4 followed by a mathematical model in Chapter 5. Chapter 6 describes a heuristic approach for solving the problem. The description of the implemented decision support system precedes the final section on implementation and numerical results.

CHAPTER 2

LITERATURE REVIEW

In this chapter, we first show the differences and similarities between our thesis problem and other existing studies, and then we summarize some articles that we used in our study.

Dormitory allocation is a Bipartite Matching Problem. There are two sets to be matched: students and beds. In addition to this, student sets also have matching within them, so we need to focus on two assignment problems at the same time. Our problem shows similarities to several subjects such as: Knapsack Problem, Assignment Problem, Stable Marriage Problem, Stable Roommates' Problem, Hospital/Residents Problem (College Admissions Problem) and House Allocation Problem. However none of these problems exactly fit the description of the dormitory allocation problem we are studying.

The knapsack problem is the maximization of the values of items in the knapsack without exceeding the capacity of the knapsack. The similarity of this problem with our problem is that we also have limited capacity in our dormitories, and we try to choose students according to the dormitory placement criterion. Each student has a weight according to these criteria, and if we think of our problem as Knapsack Problem, we see that our problem tries to maximize the summation of these weights. In addition to this, our problem seems easier than the Knapsack Problem in one way; the item sizes can be different in Knapsack Problem, the area that a student covers in dormitories (single bed) is the same for each student. If we were thinking according to the university policy, this approximation would be useful for our problem. However, other important point in our problem is the wishes of the students. So, in our problem, it is possible to say that we have more than one knapsack. Since the placement to the dormitories (knapsacks) also depends on the wishes of the students, the solution algorithms developed for Knapsack Problem is not suitable for our problem. We have to find the optimal solution for both the university side and the students side.

In the Stable Marriage Problem, there are two types of sets, and the goal is to find perfect matching between these sets as we want in our problem. Gale and Shapley algorithm (1962) finds the optimal solution to this problem in polynomial time [1]. In our problem, we try to find solutions according to three criteria: 1) university regulations, 2) dorm preferences of students, 3) roommate preferences of students. If we ignore the third criterion, our problem turns into College Admissions Problem which is a special kind of Stable Marriage Problem [2]. If we think dormitory types as a kind of set in our problem, other set is candidate students for dormitories. The first set can be matched with the second set as much as capacity allowes. As opposed to the Stable Marriage Problem, in the Dormitory Placement Problem same-sex preferences (student to student) also exist on top of preferences for the other sex (beds).

Stable Roommates Problem searches for matching people with roommates. The difference of this problem with Stable Marriage Problem is that there is only one set in this kind of problem. The matchings happen inside of this set. A solution was found in polynomial time in 1985 with Irving Algorithm [3]. This algorithm still does not fit our problem perfectly because the students make their roommate preferences according to their preferred room types (maximum of three ranked preference). The roommate preferences are symmetric with the rate of 95% which means that if student A chooses student B, student B also chooses student A. So, perfect matching is easier now without a special algorithm.

Other algorithms in literature such as Minimum or Maximum-Weight Matching [4], Auction Algorithm [5], Hungarian Algorithm or Kuhn-Munkres Algorithm [6] also do not present solutions for our problem because they try to do the assignment or matching according to one side only. The problem we studied has an additional dimension and three criteria are tried to be fulfilled at the same time (matching students with beds while observing roommate preferences). Problems studied in literature do not consider all three aspects of the problem.

Table 2.1 Used articles in this study

#	Date	Authors	Problem Type	Solution
1	1962	Gale, D. and Shapley, L.S.	Stability of Marriage/College Admissions Problem	Gale and Shapley algorithm
2	1969	Crandall, R. H.	Student Housing	Mathematical modelling
3	1981	Dubins, L.E. and Freedman, D.A.	College Admissions Problem	Gale and Shapley algorithm
4	1985	R.W. Irving,	Stable Room-mates problem	Irving algorithm
5	1989	R.W. Irving,	Strongly Stable Matchings	Gale and Shapley algorithm
6	1989	Gusfield, D. and Irving, R.W.	Stable Matching Problem	Gale and Shapley algorithm
7	1990	Wright, M. B.	Linear Assignment Problem	Hungarian algorithm
8	1992	Bertsekas, D. P.	Network Flow Problem	Auction algorithms
9	1993	Ahuja, R.K., Magnanti, T.L. and Orlin, J.B	Assignment Problem	LP, Heuristic approach
10	1999	Abdulkadiroglu, A. and Sönmez, T	House Allocation	Lottery Mechanism
11	2006	Ergin, H. and Sönmez, T.	College Admissions Problem	Student-optimal stable mechanism, the Boston Mec.

12	1999	Balinski, M. and Sönmez, T.	College Admissions Problem	Gale and Shapley algorithm
13	2003	Ehuchi, A., Fujishige, S. and Tamura, A.	Stable Matching Problem	Gale and Shapley algorithm
14	2006	Dean, B.C., Goemans, M.X. and Immorlica, N.	Stable Matching Problem	Heuristic approach
15	2008	Perach, N., Polak, J. and Rothblum, U.G	Dormitory placement	Deferred acceptance algorithm
16	2010	McDermid, E.J. and Manlove, D.F.	Stable Matching Problem	Gale and Shapley algorithm
17	2011	Biro, P. and Klijn, F	Stable Matching Problem	Heuristic approach
18	2014	Biro, P., Manlove, D.F. and McBride, I	Hospital/Residents Problem	Integer Programming
19	2014	Firat, M., Hurkens, C.A.J. and Laugier, A.	Stable assignments	Gale and Shapley algorithm
20	2014	Duan, R. and Pettie, S.	Maximum cardinality	Maximum weight matching algorithm

Article that guided this study can be found in Table 2.1. The classical assignment problem is a closely related topic to our study. Ahuja, Magnanti and Orlin [7] define the matching problem as a wish to find the best way to pair objects or people to achieve some desired goal which also fits the Dormitory Placement Problem.

House allocations are a type of assignment problem. Abdülkadiroğlu and Sönmez [8] study the house allocation with existing tenants in their study. Similar to our problem, there is a set of houses which have to be allocated to a group of agents. In this problem, they propose Pareto efficient by motivated the idea of that existing tenants do not want to leave their house before moving to a new house, but they need to give up.

Another example for assignment problem is Ergin and Sönmez's work [2] which deals with assigning children to public schools and is currently used in Boston. Many school districts in the U.S. use a student assignment mechanism which is referred as Boston mechanism. The Boston mechanism is the most widely used student assignment mechanism in real-life applications of school choice problems. A student assignment mechanism is a systematic mechanism that selects a matching for each school choice problem. In Boston mechanism, a student loses his priority at a school if his parents do not rank it as their first choice. The key point is matching with fixed priorities and given preferences.

Balinski and Sönmez [9] introduce a new class of matching problems that models centralized college admissions via standardized tests in Turkey. The central authority makes the placements of students into colleges. This class of problems is closely related to the celebrated College Admissions Problem (Gale and Shapley) [1]. The difference is that in the student placement problem the only agents are the students because their examination scores and preferences for colleges are the only determining factors for assignments [9]. On the other hand, in the College Admissions Problem not only the students are the determining factor, but also the colleges are agents. Students express their preferences for colleges and colleges for students. Despite this difference, both problems are similar to each other. Balinski and Sönmez propose Gale and Shapley student optimal mechanism exploiting the relationship between them.

Fırat et al. [10] analyzes stability in multi-skill workforce assignments of technicians and jobs. Like the others, they also extend the notion of blocking pairs as stated in the marriage model of Gale and Shapley to the multi-skill workforce assignment. They propose an integer programming (IP) model to construct optimal stable assignments.

The Hospitals/Residents (HR) Problem is another real-life application of assignment problem [11]. The Hospital/Residents Problem with couples (HRC) is a generalization of the classical HR problem, and it models the case where couples submit joint preference lists over pairs of hospitals. Like many other assignment problems, this problem involves two sets as residents and hospitals, but it is a many-to-one allocation problem. Like McDermid and Manlove [11], Biro et al. [12] studied HRC. They presented NP-completeness results for the problem and an IP model for HRC. In addition to this, they described an empirical study of an IP model for HRC. Their model was used to allocate junior doctors to hospitals in Scotland.

Dean et al. also studied HRC before [13]. Differently from others, their version permit a hospital's capacity to be exceeded by the assignment of a couple. They formulate the problem in terms of assigning residents with integral sizes to hospitals with capacities. They provide a polynomial tome integral variant of the Gale and Shapley algorithm that finds a stable matching in which each hospital is congested by at most the processing time of the largest resident.

Matching with couples [14] is also an important problem related to our study, use a heuristic to solve their problems. They focus on presence of couples that each members looks for a job in the same market. It is different from most of the studies because here the agents' preferences contain complementarities such as the presence of married couples.

In the classical Bipartite Matching Problem objects are separated into two groups, and the main objective is to find best pairs in different groups [7]. Also we can divide these problems into two versions: the cardinality problem which tries to find solution by using maximum arcs and the weight problem which is known as assignment problem in Operation Research literature. Here, each arc is weighted, and the target is to reach solution that contains the largest overall weight.

Stable Marriage Problem is a version of Bipartite Matching Problem which can be stated as follows: A certain community consists of n men and n women, two sets. Each person is ranked in accordance to his or her preferences for a spouse [15]. A matching is a set of n disjoint pairs of men and women, and is called stable if there is no pair whose members prefer each other to their partners in the matching [16].

Gale and Shapley proved $O(n^2)$ solution time in 1962 in their own study of Stable Marriage Problem [1]. According to Gale and Shapley algorithm in the paper, each man proposes to his favorite girl, and then the girls who receive one proposal accept these proposals. But if more than one proposal is received, she only can keep the proposal which comes from her favorite from among of proposals. In the second stage, those men who were rejected propose to their second choices. Again, the woman accepts the proposal, only if she did not accept a proposal before. The process keeps going until all people are paired and is reached a stable matching.

But what is the meaning of stable matching? The definition of unstable helps us to understand the stable matching. Again, Gale and Shapley original paper gives example about college admissions which have two sets that are colleges and students. We can think of students like men and colleges like women. So, let's focus on the paper's definition and example; "An assignment of applicants to colleges will be called ***unstable*** if there are two applicants α and β who are assigned to colleges A and B, respectively, although β prefers A to B and A prefers β to α". For this example, If we match A and β, both A and β would consider the change as an improvement. The first requirement on an assignment is that it does not exhibit instability. And the paper gives us another definition: "***A stable assignment*** is called **optimal** if every applicant is at least as well off under it as under any other stable assignment".

Gale and Shapley algorithm helped in designing a heuristic solution to our problem. According to Gusfield and Irving's book [17], all possible executions of the Gale and Shapley algorithm with men as proposers lead to a stable matching. That implies that if each man is independently paired with his best stable partner, then the result is a stable matching. If the roles of the sexes in the algorithm are changed, the results will be optimal for women.

Dubins and Freedman's study [18] is another resource for explaining Gale and Shapley algorithm for assigning students to universities that gives each student the best university available in a stable system of assignments. In this study, each student rank orders all the universities and each university rank orders all students. The goal is to pair the students and universities in a stable way.

In the only implementation for dormitory allocation, Perach et al.'s studied assignment of students to dormitories at the Technion-Israel Institute of Technology [19]. They use two criteria in considering applicants. The first criterion is used to make decisions about the privilege of getting dormitories. The second one is used for the actual assignment of the eligible students to specific dormitories. The priority of a student is determined by academic seniority and academic excellence. They described a modification of Gale and Shapley algorithm that produces a matching satisfying the conditions and desirable properties. But this study also ignores roommate preferences.

CHAPTER 3

MANUAL APPROACH

Işık University has given accommodation service in dormitories since it moved to Şile campus. When we look at the dormitories chronologically, we see that Orange, Red A-B, Red C-D and Blue dormitories came into service respectively. First inhabitants of the campus were preparation school students. After them, except the Fine Arts Faculty, other faculties moved to Şile campus one by one.

In first years, the dormitory capacity and demands were very low. The main policy of the university was to provide accommodation to the students who want to take advantage of this service because Şile campus is far away from the city center and most of students come from other cities. To provide this service, the university expanded the opportunities of accommodation until reaching the physical borders of the campus.

When the university just moved to Şile campus, the opportunities for accommodation were. There were only Orange dormitories. Orange dormitories are designed for mostly for two students. There were rooms for four people with a lower price, but these rooms were limited. Additionally, the demand was lower than the capacity, so the dormitory office just paid attention to demands for roommates for placement. For the students who have not roommate preferences, the dormitory office made the placement according to their entrance years and departments.

Since the rooms housed only two students and the demand was lower than the capacity, almost all students were happy. The dormitory office had a simple approach about this placement process. In the first step, the students were separated into two groups according to their room preferences. Then the students who had roommate preference were placed to free rooms. The rest of students were placed according to their entrance years and departments. The process is depicted in Figure 3.1.

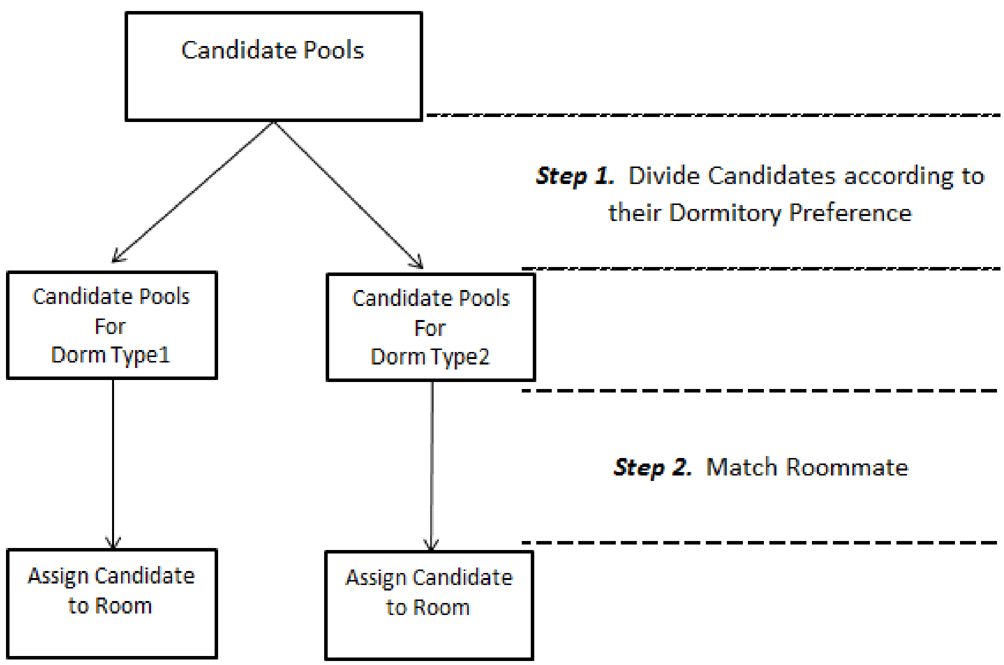


Figure 3.1 Steps for manual placement until 2006

In later years, when other faculties moved to the campus and number of students increased, the dormitories could not meet the demand. To meet this demand, the university built new dormitories. During this process, the university increased dormitory types, so room designs and prices came into play. The first dormitories after Orange dormitories were Red A and Red B. These were designed for three or four people to meet the high demand.

These room types were designed with shared bathrooms on each floor. Because these rooms were for more than two people and had shared bathroom, the price was lower than Orange type dormitories.

One year later, all faculties (except Fine Arts) moved to the campus, and even new dorms could not meet the demand. Some students did not want to stay in orange dormitories because the price was high. Additionally they wanted to have a private bathroom. To meet these wishes, university built a new type dormitory Red C-D. These were also for three or four people, but they had private bathroom, so the price was lower than Orange Rooms, but higher than Red A-B.

Increasing demand and dormitory types made the placement process harder. In first years, the demand was low and the applications were received in paper. It was not difficult to organize and arrange these documents. But when the demand increased, the number of application forms increased, and just to organize the data was taking at least one week with three people. Still, the dormitory office continued to apply the same manual process.

The satisfaction of students was decreasing very fast. The reason for this dissatisfaction was seen as having inadequate dormitory types, and as a solution the university decided to build Blue Dormitories. These dormitories were designed for two people and three people. The Blue Dormitories for two people were built for meeting high demand for Orange Dormitories, and three people room were built for meeting the high demand for Red Dormitories.

With Blue dormitories, the dormitory capacity reached 1250 people. But the number of students who want to accommodate in dormitories continued to increase with the rise of faculties and departments. For a solution, the dormitory office decided to add extra beds to the existing rooms, and the dormitory capacity reached to 1300.

The goal of the university was to meet all the demands of students after moving to Şile campus. But many students could not be placed in their desired rooms types, and complaints about this situation started to reach to the university's managing body.

Additionally, even with capacity increase, some students could not be placed to any rooms, and this situation pushed the university to change the placement process. In the past, the target was to replace all students to the dormitories, but the new idea was to place students of top priority. To meet these privileged students' roommate preference, room preference became the most important criterion.

The new process can be split into three stages: Organizing data, to decide the students who will be placed to dormitories and to assign the students to the rooms (Figure 3.2 and 3.3). Until 2012, the applications were made with paper forms. The candidate students were separated according to their dormitory preferences, and were matched with their roommates. After 2012, the applications were taken by an on-line application. It was a much easier process than hard copy applications, but it still had many difficulties. The main difficulties were open-ended questions. Another difficulty was unmatched-roommate problem.

After placement data was organized and controlled, students who have priorities were decided. The main criterion in this step was residence address and class of the student. The students who have residence address farthest away from Şile had priority in placement. Additionally, new registered students and senior students had priority. But the final decision was made by the person who makes the placement. In some cases, the students who live near Şile who asked for a roommate living outside of İstanbul had more advantage than the students who are far away from Şile and not wanted any roommate.

After collecting the applications, the forms were checked according to placement rules and violating applications were removed. If number of applications was lower than the dormitory capacity, the next step was the placement process. Otherwise, the students were separated according to their resident address and classes, and excess students were rejected starting from the end of the list. The actual placement process was the most difficult part because there were no hard rules for giving priorities to students in the Dormitory Regulations of the university. Because of this reason the person who makes the placement had to make subjective decisions in some cases such as the decision of new registered student who lives near Şile and the student who is in Preparation School and lives near Şile. These hard questions increased placement time, and additionally because of subjectivity the placement caused dissatisfaction among students.

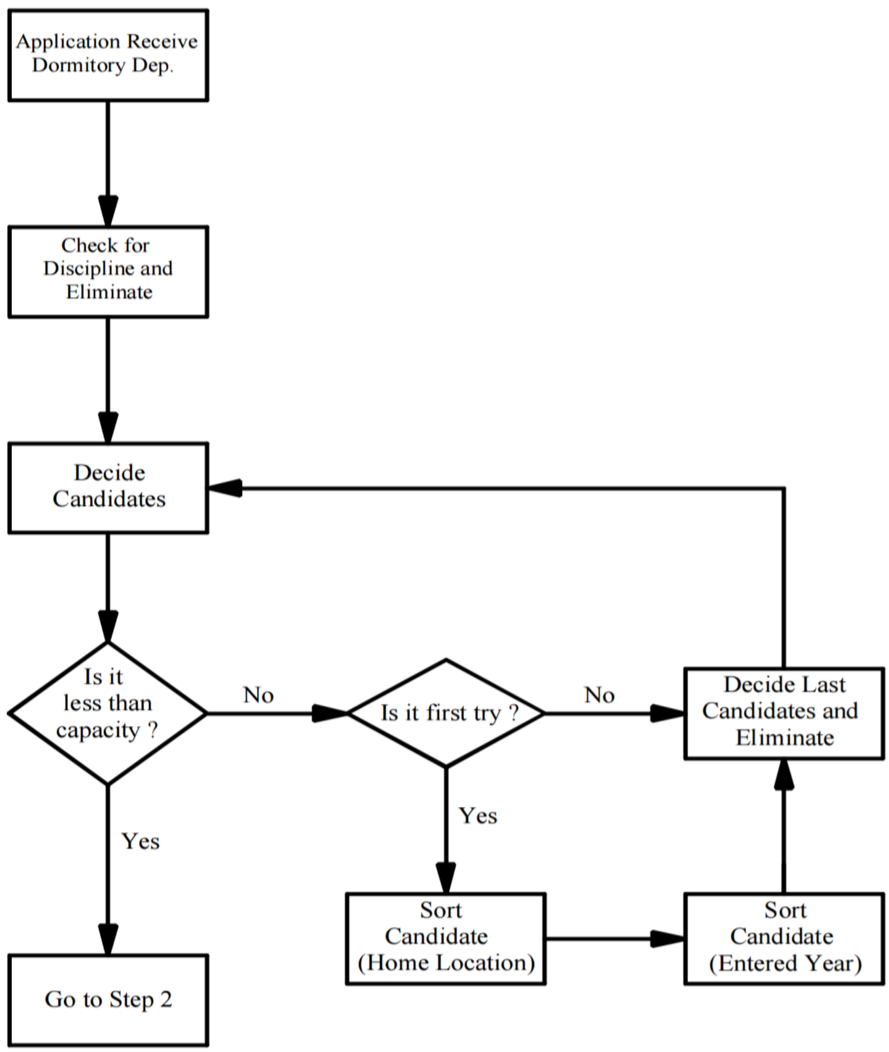


Figure 3.2 Steps for manual placement for dormitory type after 2006

In the second step, the Housing Department assigned the candidates who have roommate preferences first. After this match, if there is enough capacity for these students' preferred room type, they were assigned to this room. For example, if student-A and student-B want to stay in dormitory type Red-D together, their wishes were respected. If they cannot be placed to their desired rooms, they were placed to their second choice. If they cannot be placed to any preferred rooms, they were placed to another room with a lower price. We can say that when matching candidates with roommate preferences, they assumed them to be one candidate student occupying two beds.

With increasing dissatisfaction among students about placement decisions, the university bought some online software in 2012 so that students could choose their own rooms. The Housing Department only checked whether the student met their financial obligation and were eligible to stay in dormitories. However, it was first come first serve system, it could lead many dissatisfaction so well. Or it was plausible that student that student from nearly address could be quickly fill the dormitories. The university still decided to go ahead with this system, on the day of selection, the system did not work and it was turned off and placement was done manually again. After this mishap, the university decided to implement an in-house solution.

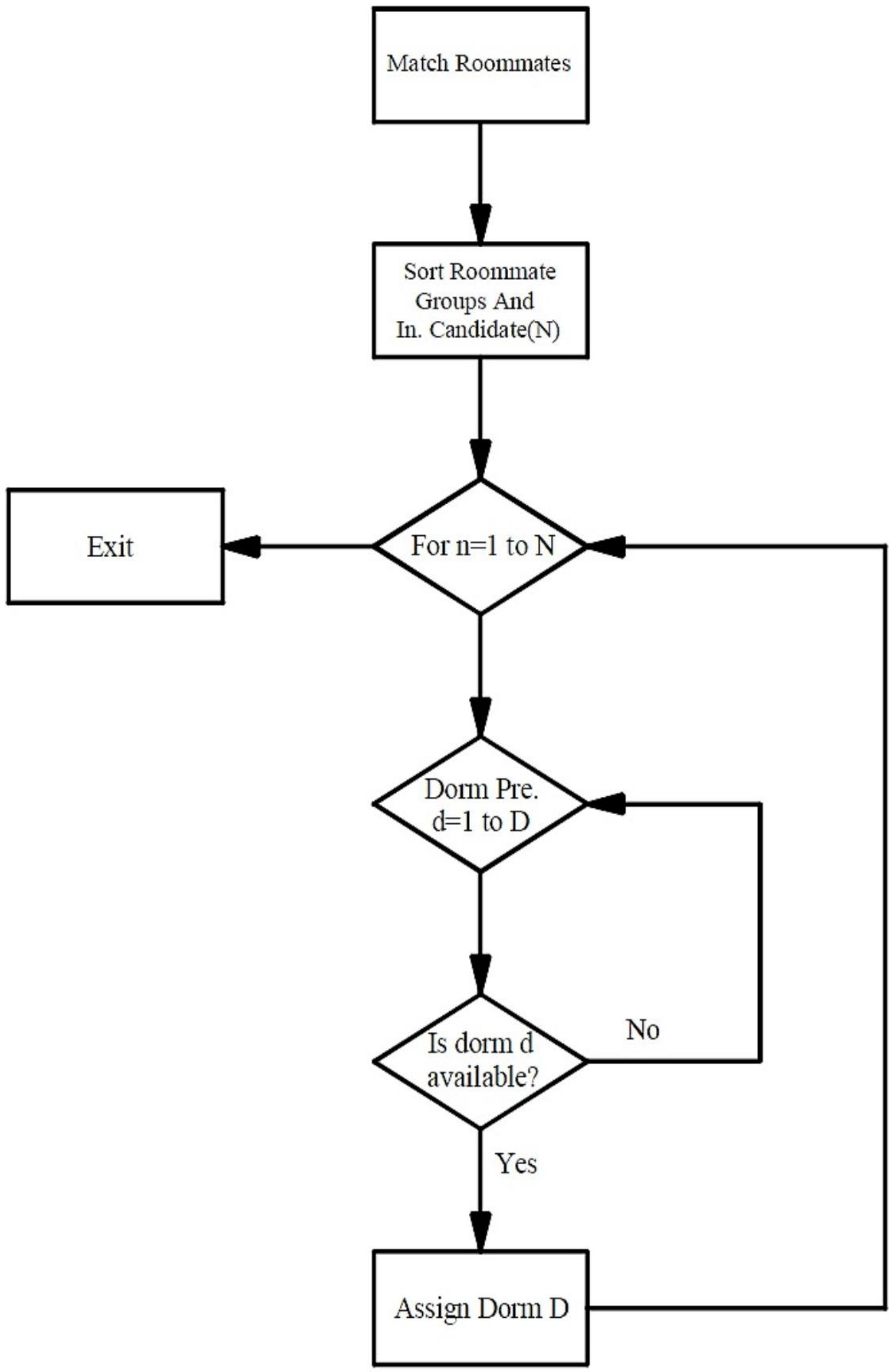


Figure 3.3 Steps for manual room assignment after 2006

CHAPTER 4

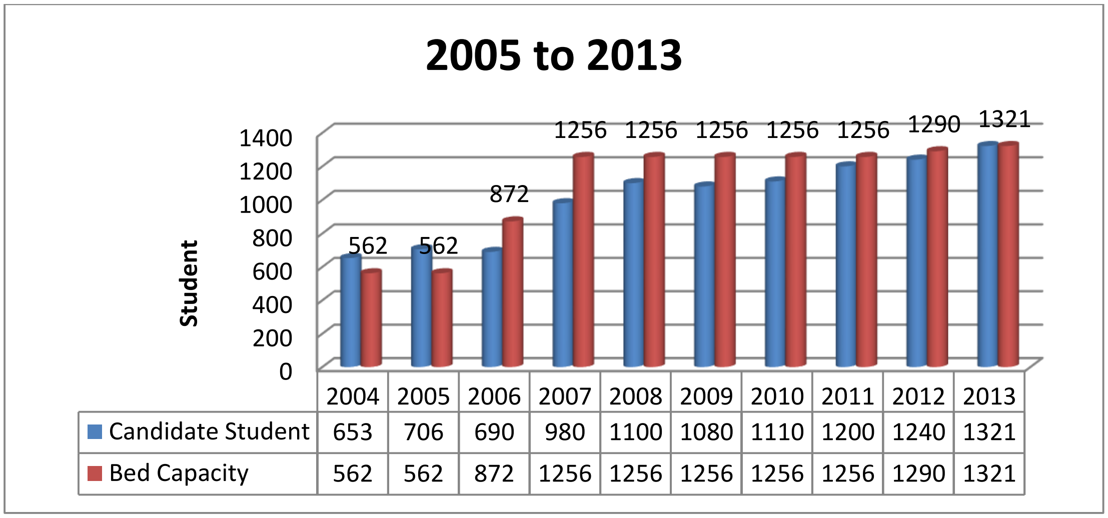
DATA ANALYSIS

Graph 4.1 shows the number of applicants and dormitory capacities over the years. The university also increased capacity by putting extra-beds into rooms since constructing new building is very expensive. Currently, all dormitories have reached their maximum capacity. Additionally, the university has no plans for building new dormitories. As a result, approximately 130 students could not be placed in the 2014-2015 academic year. Also, in the 2015-2016 academic year, more than 170 demands could not met.

Next, a more detailed analysis of 2014-15 and 2015-16 data is provided. Older data collected manually were not suitable to conduct this analysis.

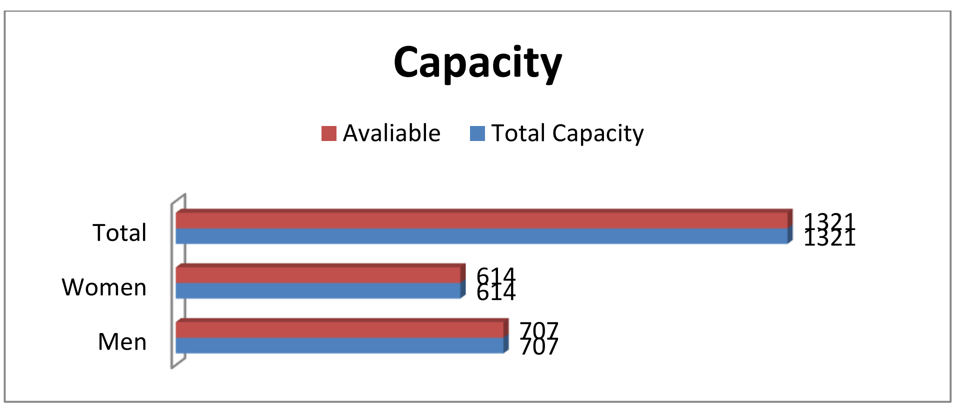
4.1 2014-2015 Applications

The application data for the 2014-2015 academic year is very important because 2014 is the first year for testing the developed system.



Graph 4.1 Total number of candidate students and dormitory capacities

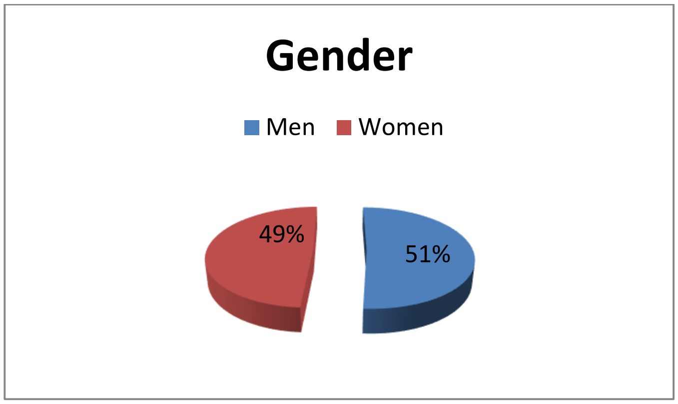
According to the Işık University Dormitory Regulations, male and women cannot stay in the same dormitory blocks. For this reason, the male blocks and female blocks are separated. Dormitory type capacities according to gender are shown in the Graph 4.2 that also shows available and total capacity. Some years the university protects some rooms for many reasons like conference, but after 2013 the policy is left.



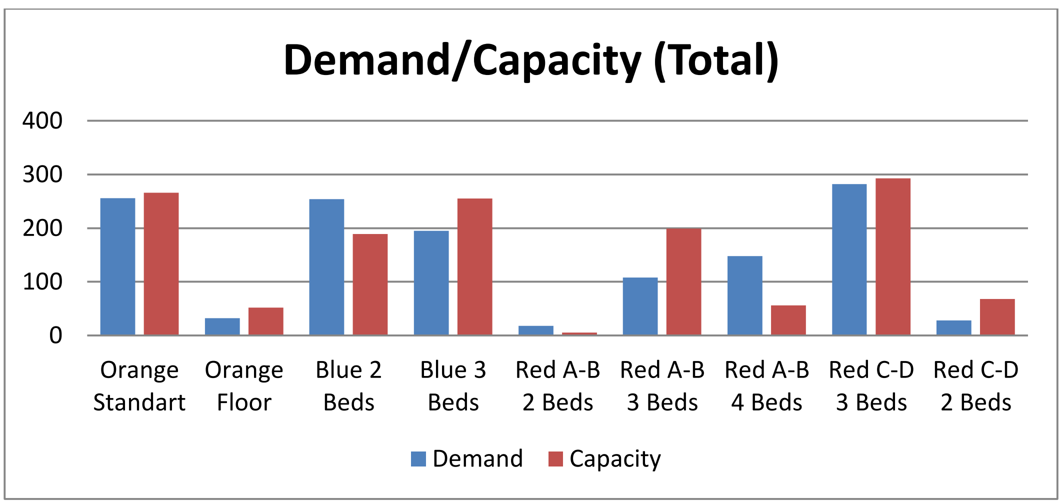
Graph 4.2 Dormitory type capacities according to gender

Almost 50% of the students were female in 2014-2015 (Graph 4.3). In previous years, male students had a higher percentage. Since women are given priority in placement, some new blocks had to be categorized as women only in 2014-2015, which was decided after running some numerical tests with the new system.

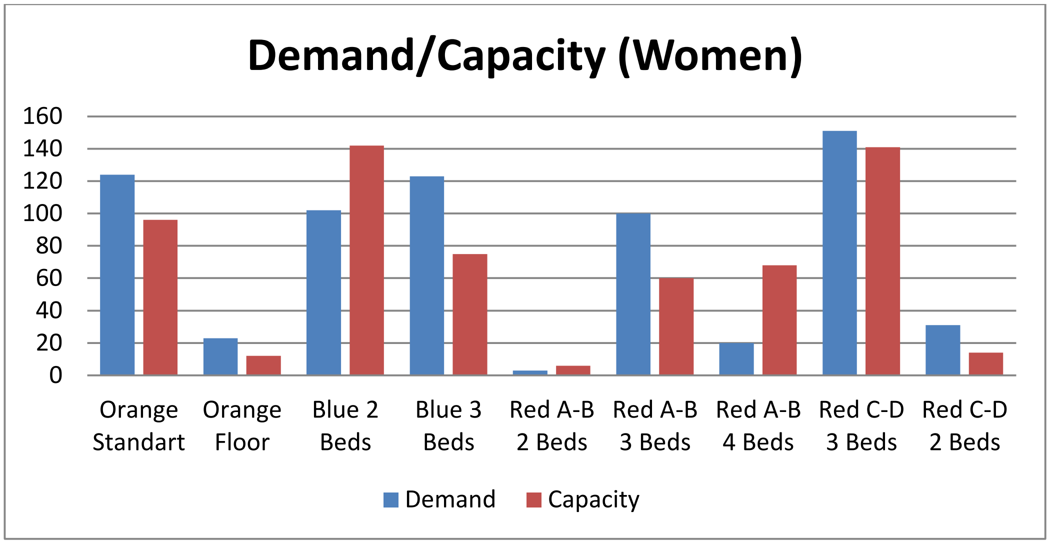
Graph 4.4, 4.5 and 4.6 show demands and capacities by dormitory type and gender.



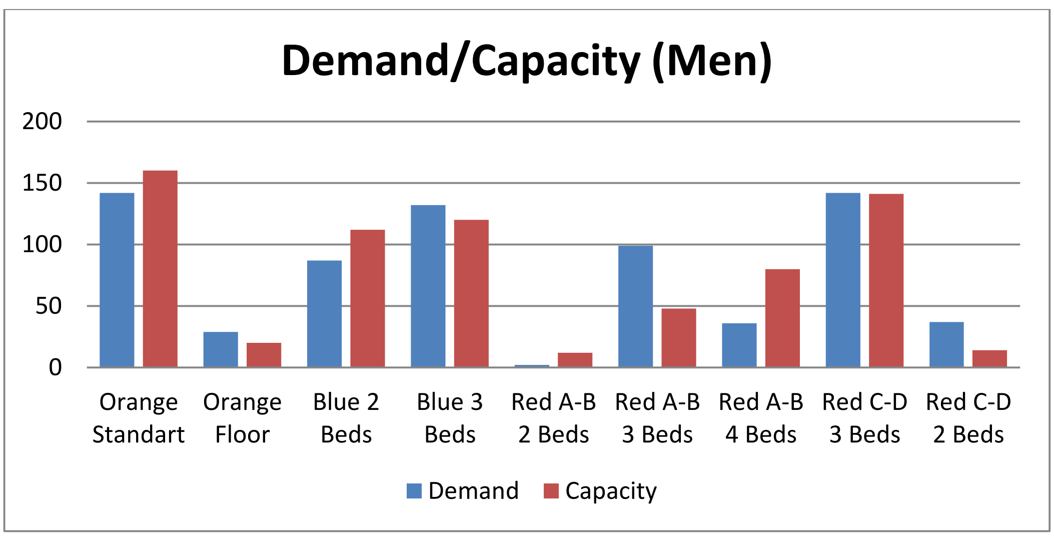
Graph 4.3 Percentage of candidate students according to gender



Graph 4.4 Demand and capacity for all students



Graph 4.5 Demand and capacity for women

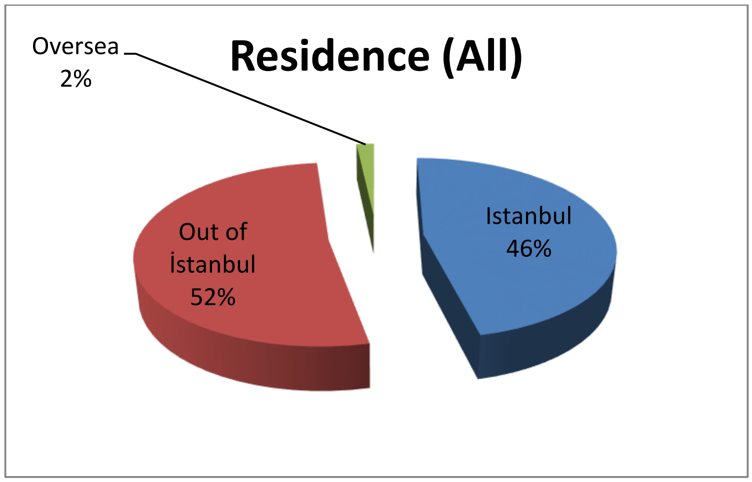


Graph 4.6 Demand and capacity for men

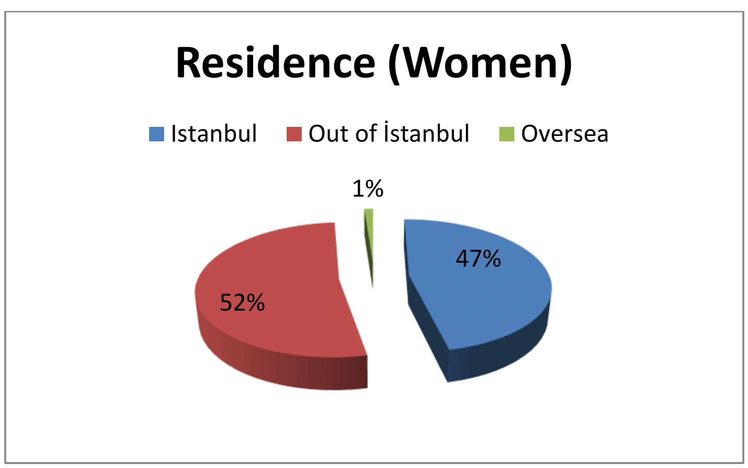
When we examine these graphs in detail, it is easily observed that the total demand for two people in Blue dormitories is more than the capacity of these rooms, but the opposite holds for the rooms with three people in Blue dormitories. Still, the university preferred to keep all of the Blue three rooms to have additional capacity.

Moreover, as a change from the previous year, one Orange block was devoted as a female block (Orange C).

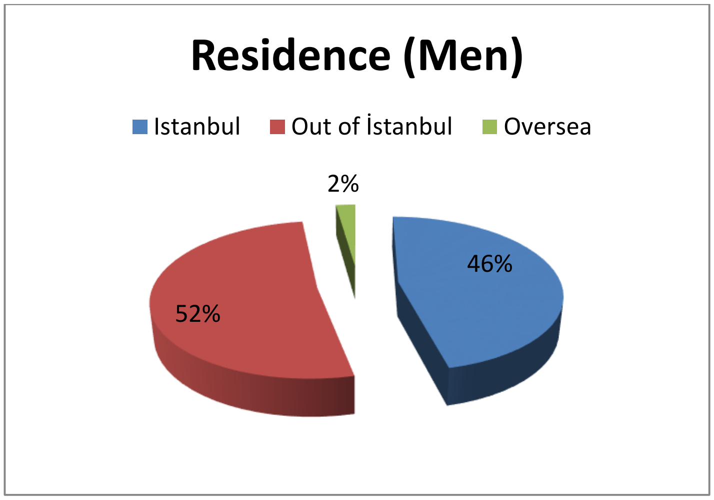
Graph 4.7, 4.8 and 4.9 show that applicants from İstanbul constituted almost 50% of the applicants.



Graph 4.7 Residence percentage for all students



Graph 4.8 Residence percentage for women



Graph 4.9 Residence percentage for men

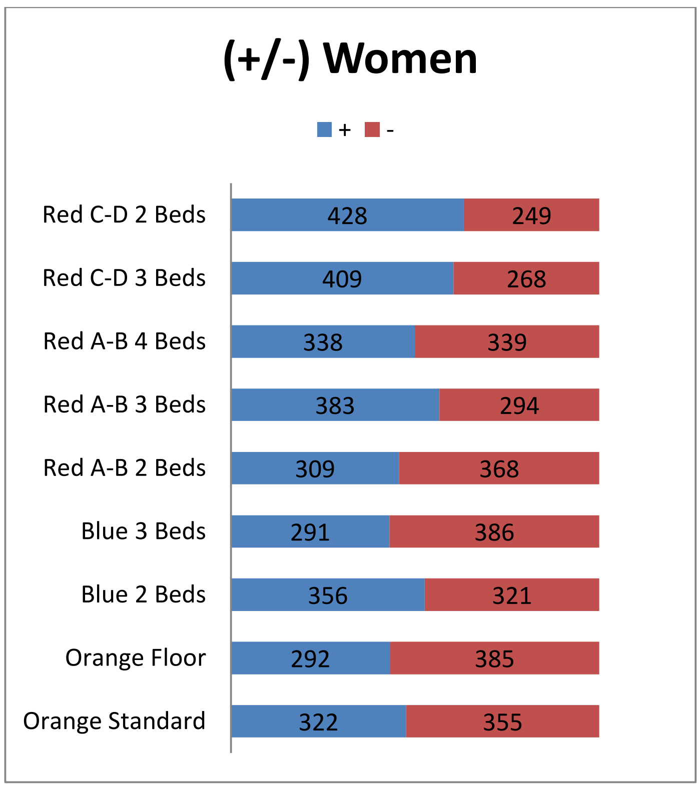
In 2014-2015, the students were also asked for the first time in which rooms they do not want to stay. Graph 4.10 and 4.11 give the details of the responses, where a (+) sign shows their preferred room type, and a (-) sign means that the students do not want to stay in that room type under any conditions. Data analysis helped us to understand the existing conditions and affected our next decisions on the following steps.

4.2 2015-2016 Applications

The applications for 2015-2016 academic year were completely different than 2014-2015 academic year. Until the 10^{th} of August 2015 that is the due date of applications, 1121 students applied for dormitories. We thought the reasons of this decline as:

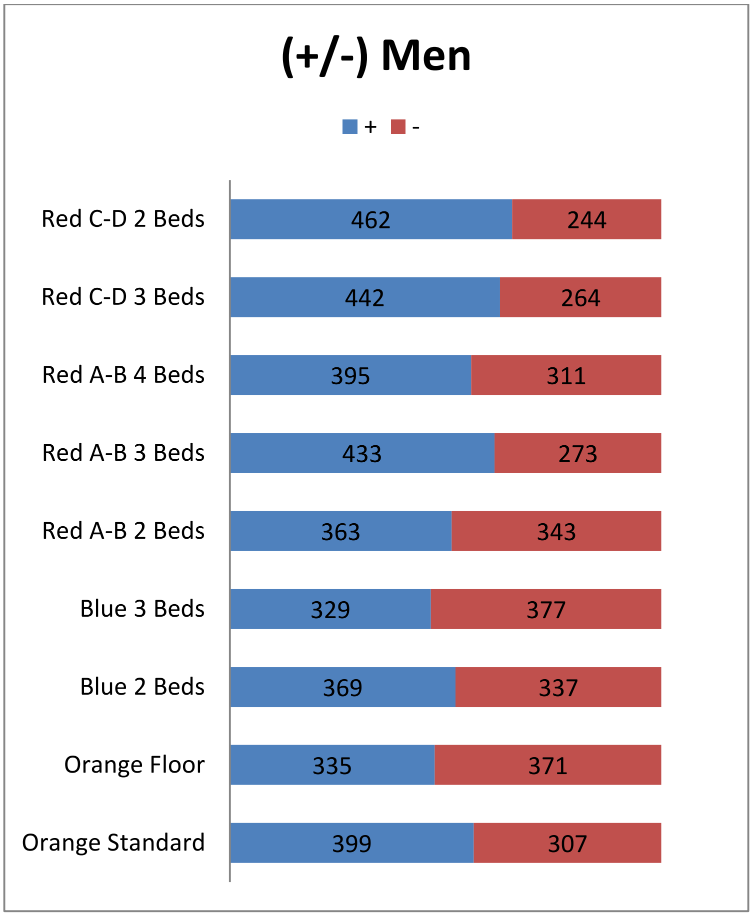
- Earlier due date compared to other years,
- Inadequate announcement about the due date of dormitories by the Student Council
- New housing near the university

But new requests for dormitories arrived to the Dormitory Office after the 20th of August that is the date of the announcements for the application results. The general justification about their late applications was that they did not know the true due date of applications. After that the dormitory office decided to extend the due date until the 28th of August, so the number of applications increased to 1453 that was higher than the previous year. For two weeks after the opening date of the university, the requests for dormitories continued to come in. Because we made the dormitory placements twice in that year, we had two data sets. The capacities for 2015-2016 academic year were the same with 2014-2015 academic year.



Graph 4.10 Preferences for women

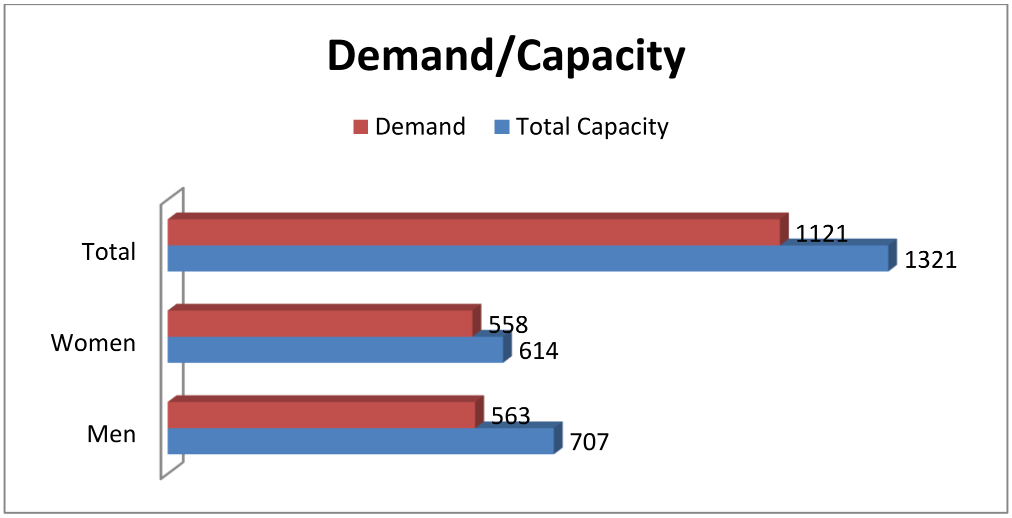
.



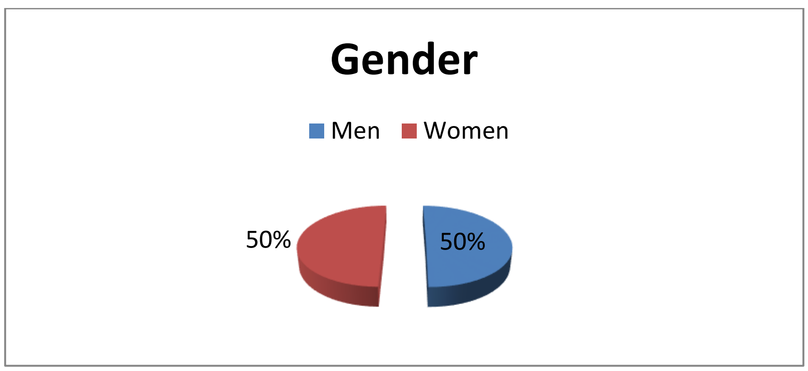
Graph 4.11 Preferences for men

Graph 4.12 shows the first requests in normal application time and total capacity for 2015-2016 academic-year.

Despite the first applications showed decrease compared to previous years, still the application rates according to the gender were very close (Graph 4.13).

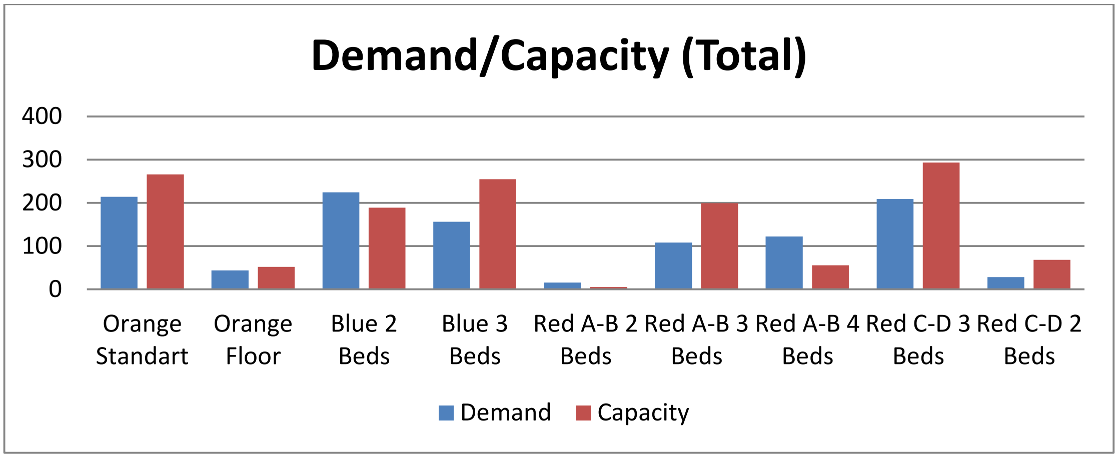


Graph 4.12 First requests and total capacity for 2015-2016 applications



Graph 4.13 Gender percentages for 2015-2016 applications

Although the initial number of applications were less than other years, the demand for some dormitory types were more than the capacity of these room types (Graph 4.14). Still, the decrease in demand was distributed equally according to the dormitory types, so the proportion of application numbers for dormitory types to the capacities did not change much compared to previous years.



Graph 4.14 Demand and capacity for all students

4.3 Allocation Criteria

In the past, the student applications were evaluated subjectively for eligibility depending on the opinions of the Housing personnel.

Our first target was to determine eligibility rules clearly and to share these rules with the students. For transpareny, we arranged different meetings with the university management, student council and administrative units. Then we clarified the placement criteria.

Firstly, a survey was prepared that contained the clauses in dormitory regulations. We asked the students, university management and administrative staff to fill this survey and sort these clauses according to priorities. This survey was an important starting point to determine the priorities. Generally, all participants agreed that should have priority students with distant home addresses, senior students and prep students.

Other priorities were based on the following criteria.

4.3.1 ÖSYM Achievement Grant

Students who have reached to a success level on the student selection examination which is applied in whole country have the right to stay in dormitories free. This criterion is our first certain criterion.

4.3.2 Nonadult Students

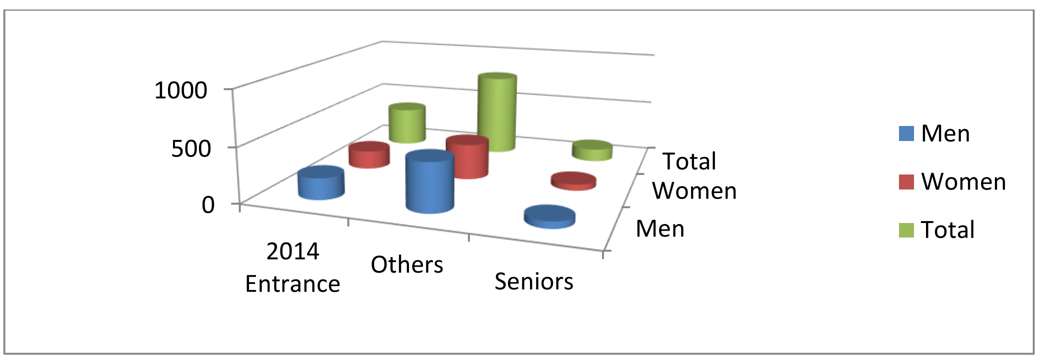
Other important rule of university is that students who are not adult (student under 18) automatically get to stay in dormitories.

4.3.3 Normal Period of Study

Students who exceed the normal length of study, that is 2 years for preparation school and 5 years for faculty, can only stay in the dormitories if there is free capacity.

4.3.4 Entrance Year and Seniority

First year and senior students were given priority over other applicants. Moreover first year students were preferred over senior students (Graph 4.15).



Graph 4.15 Applications according to entrance year and seniority

4.3.5 Home Addresses

We can cluster FMV Işık University students in three groups: students who come from abroad, local students who come from outside of İstanbul and students who live in İstanbul. Students from outside of İstanbul were obviously given priority. You can see the location of FMV Işık University in Figure 4.1.

Figure 4.1 Location of Işık University in Turkey

After meeting with Çınar Tur, the İstanbul region was also divided into eight sub-regions according to their proximity to the campus shown on Table 4.1.

4.3.6 Other Criteria

Other student characteristics (GPA, double major, minor, membership for student council and club member) were not used as a criterion in 2014-2015 and 2015-2016 as people had very opposite views about these.

Table 4.1 Districts in subregions

3. Group	ARNAVUTKOY	BEYLIKDUZU	BUYUKCEKMECE	CATALCA	ESENYURT	SILIVRI	ADALAR
4. Group	AVCILAR	BAGCILAR	BAHCELIEVLER	BAKIRKOY	BASAKSEHIR	BAYRAMPASA	ESENLER
	GAZIOSMANPASA	GUNGOREN	KUCUKCEKMECE	SULTANGAZI	ZEYTINBURNU		
5.Group	FATIH	PENDIK	TUZLA				
6.Group	EYUP	SARIYER					
7.Group	BESIKTAS	BEYOGLU	KAGITHANE	SISLI	KARTAL	MALTEPE	
8.Group	ATASEHIR	BEYKOZ	KADIKOY	USKUDAR			
9.Group	CEKMEKOY	SULTANBEYLI	SANCAKTEPE	UMRANIYE			
10.Group	SILE						

4.4 New Application Form

Dormitory applications were made online since 2009. However, due to open-ended questions and students false responses the data had no integrity. Together with IT department, a new application form was designed. The national identity number was used as a unique key to identify the students and data that already exists about students (GPA, gender, etc.) in other university databases were automatically filled in the new form.

In the new form, students can specify three dormitory type preferences. If they cannot be placed in their three preferred types, they are placed to another dormitory type if they accept to stay. The students can also specify where they do not want to stay. This allows for distributing the students better decreasing the risk of the student moving off-campus. Online application form was coded in ASP.NET using C#. Figure 4.3 shows an initial design for the application form in Excel.

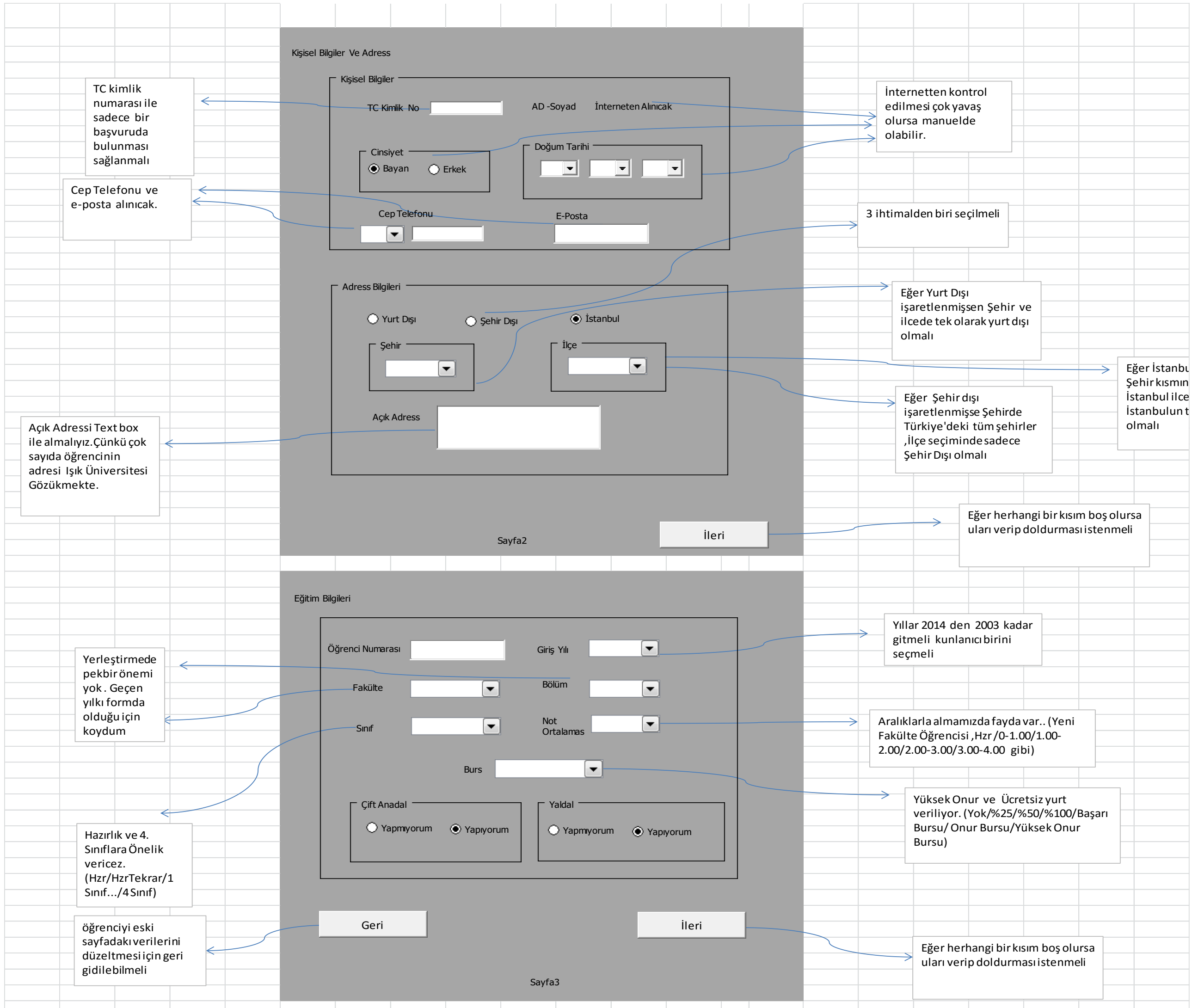


Figure 4.2 Sample design of application form

CHAPTER 5

MATHEMATICAL MODEL

The goal of the dormitory placement is to satisfy students' wishes as much as possible while observing university rules and regulations. To achieve this, the first approach we tried was a 3-stage process (Figure 5.1 and 5.2). First, students who were not eligible to stay in dormitories were eliminated. For example, students with disciplinary penalties can not stay in dormitories. Then each student's dorm is decided. In the last step, roommate preferences are considered and students are placed in specific rooms.

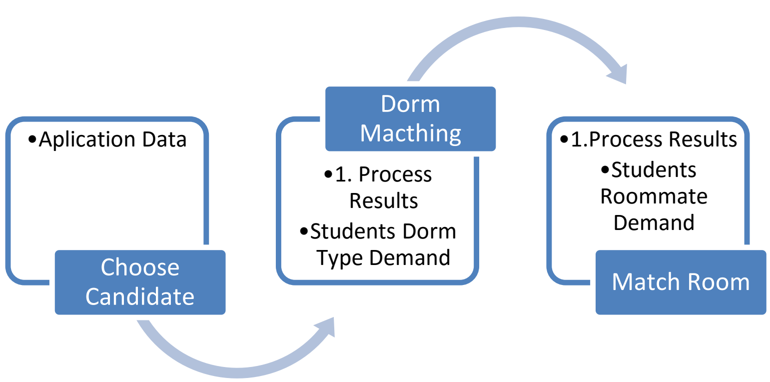


Figure 5.1 Placement process steps

Unfortunately, we had some problems with this approach. Although, we have met the requests of the university management, the satisfaction of the students was really low. Especially, in roommate preferences, the dissatisfaction of students reached almost 70% because for many students, roommate was more important than the room types.

Thus, we tried another approach where we combined all stages into a single stage by using different weights in the objective function for students' dorm type, and roommate preference. But with this approach, we have met with a new problem: How could we create these weights?

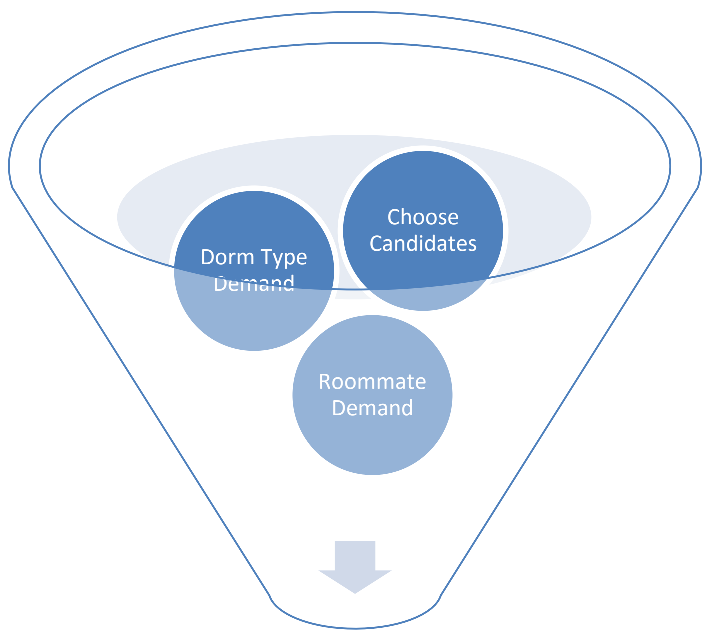


Figure 5.2 Components of placement process

5.1 Weight Decision

There are three basic weights which are necessary in our mathematical modelling. The models maximize sum of these weights. These are weights for basic placement of the students, dormitory type and roommate preference. If we compare the importance of these weights, we can say that the most important weight is to place the students at first according to the university rule and regulations because our first target is to choose the students who are eligible to stay in dormitories. From students' perspective, there are two weights which are dormitory type and roommate preference. After meetings with the Student Council and students, we observed that meeting roommate preferences of the students is more important than placing the student into a desired room with a random student. So, we set the weight for roommate part is much more than the weight for placement part as it is seen below.

$$\text{Placement Weight} \gg \text{Roommate Weight} \gg \text{Dorm Type Weight}$$

5.1.1 Placement Weight

Placement Weight was decided according to university administrators' preferences. The weights for students who are new, who are younger than 18 and who have physical disabilities were given bigger weights. We created a parameter for these three criteria. The parameter is a big number if the student carries these features. Otherwise the parameter is set to 1.

For other students who do not have any special conditions, points were set according to the distance from the campus. The student whose home country is farthest away from the campus took a 100. As explained in Chapter 4, there are a total of 10 regions (Table 4.1).

Newly registered students were given 1.2 points, and all others received a 1. Students who exceeded their maximum number of allowed bachelor years (7 years) were given 0.0001 points, whereas seniors were given 1.1 points, and others received a 1. Students who received disciplinary penalties had a coefficient of -1000.

Values for placement weights are shown in Table 5.1. By multiplying all weights for each student, we found the overall placement points of the students.

Table 5.1 Placement weights

		Weights
Age	<18	10^5
	≥18	10
Region	R1	10^2
	R2	50
	R3	25
	R4	20
	R5	15
	R6	10
	R7	5
	R8	3.5
	R9	2
	R10	1
Seniority	Freshman	1.2
	Senior	1.1
	>7 years	10^{-3}
	Others	1

5.1.2 Dorm Type and Roommate Weight

After determining the placement points for each student, it was time to determine the weights for the dorm type and roommate weight. For determining this, we used the weights for the placement process. The main reason is to increase happiness of the students who have the right to stay in dormitories at first. We tried to keep these weights smaller than the weights for the placement process, because placement process is our first priority.

The students have three options for dormitory types. Additionally, they are asked the students which dormitory type they do not want to stay in any condition also. We gave the largest weight for their first preference, smaller weight for the second preference, smaller weight for the third preference, and smaller weights for the rooms which they did not enter as preference. To determine the weights, we multiplied the point by $3x10^{-4}$ for the first preference, $2x10^{-4}$ for the second preference and 10^{-4} for the third preference. 10^{-5} was the weight for other rooms except preferred and non-preferred rooms.

For the room type where the students do not want to stay in any condition, we thought two approaches. First approach was to make change in parameters, and the second one was to make changes in the model. It seemed us to make changes in parameters easier for us. As we told before, our objective function is to maximize the points after placement process to meet the demand. So, we thought that we can give minus coefficient for the room type the students do not want to stay in, and we can make this point in absolute value for the placement point for this student. Because of the minus sign, the model does never want to assign this room to the student because it will decrease the objective function. So, we multiplied the dorm types which are not preferred by (-) 1. But, we avoided positive coefficients by multiplying two (-) values by using a proper code.

Additionally, for the roommate preferences, we arranged weights larger than the weights of room types but smaller than the weights for the placement part. So, we determined the coefficient as $2x10^{-3}$ and multiplied the placement point of the student by $2x10^{-3}$ (Table 5.2).

After determining all weights, we calculated the placement point of each student (Figure 5.3). An example for a student is shown in Table 5.3.

Table 5.2 Weights for dormitory preferences and roommate

		Weights
Dormitory Type	1st Preference	3×10^{-4}
	2nd Preference	2×10^{-4}
	3rd Preference	10^{-4}
	Out of Preference	-2
	Others	10^{-5}
	Roommate	2×10^{-3}

Table 5.3 Example weight calculation

		Age	Region	Seniority	Placement Weight (W)	Dormitory Type 1	Dormitory Type 2	Dormitory Type 3	Dormitory Type 4	Roommate
Student	Data	17	R4	Freshman	---	1st Preference	Out of Prefecence	Others	2nd Preference	----
	Weight	10^5	20	1.2	24×10^5	$W \times 3 \times 10^{-4} = 720$	$W \times (-2) = -48 \times 10^5$	$W \times 10^{-5} = 24$	$W \times 2 \times 10^{-4} = 4.8$	$W \times 2 \times 10^{-3} = 4800$

5.2 Model

Our first approach was to set up a model for assigning both dorm-types and roommates. In the model, most constraints are expressed as soft constraints. The only hard constraints is allocating each student to a single room and not exceeding a room’s bed capacity.

Sets

S = set of candidate students

D = set of dormitory types

R = set of rooms

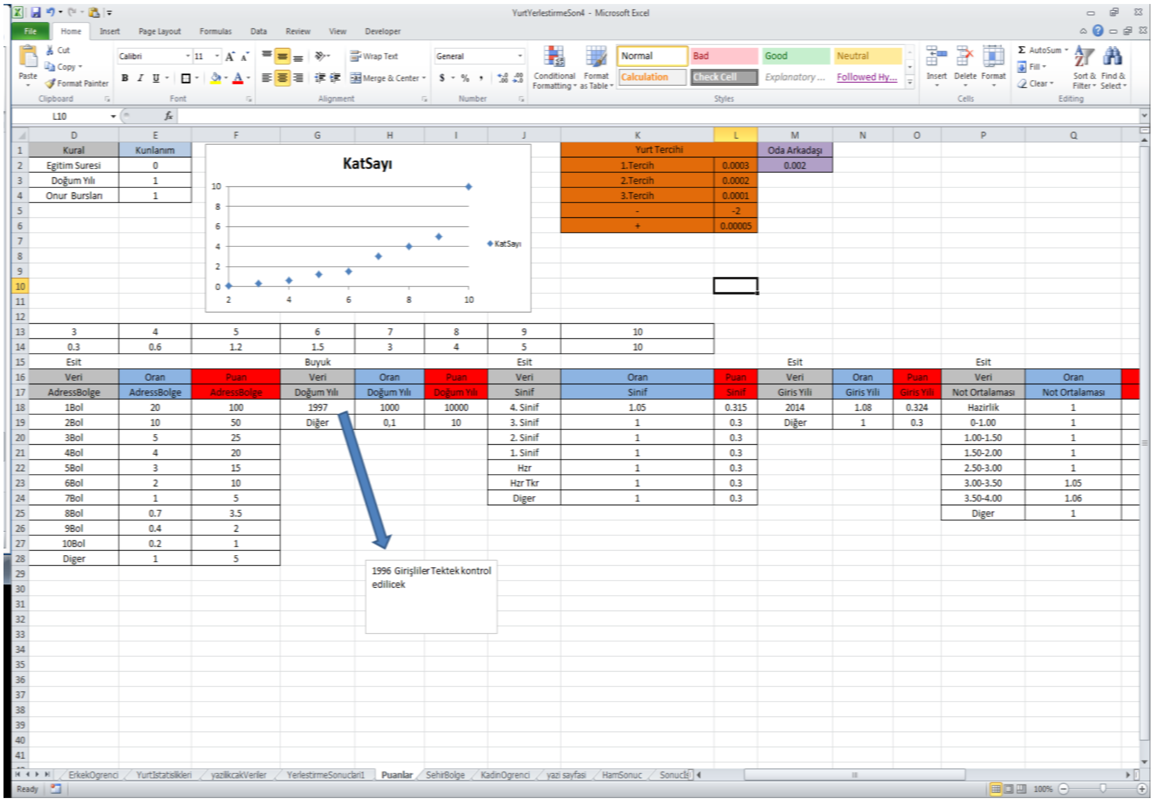


Figure 5.3 Placement point calculation

Parameters

<u>Weight Parameters</u>

w_s= weight used when candidate student **s** is placed in any dormitory.

$dw_{s,d}$ = weight used when candidate student **s** is placed in dormitory type **d**.

rw_s= weight used when candidate student **s** is placed with his/her preferred roommate(s).

<u>Other Parameters</u>

$r_{s,ss}$ = 1 if student **s** prefers student **ss** as a roommate, 0 otherwise.

$cap_{d,r}$ = capacity of dormitory type **d,** room **r**.

lim_d = maximum permitted number of students for dormitory type **d**.

Decision variables

$\boldsymbol{X_{s,d,r}}$ = 1 if candidate student **s** is placed to **d**-type dormitory, room **r** 1, 0 otherwise.

$\boldsymbol{Y_{s,d}}$ = 1 if candidate student **s** placed **d**-type dormitory, 0 otherwise.

$\boldsymbol{RM_{s,d,r}}$ = total number of desired roommates for candidate **s** for dormitory type **d** room **r**.

The model is as follows:

$$max\ z = \sum_{s\in S}\sum_{d\in D}\sum_{r\in R} w_s X_{s,d,r} + \sum_{s\in S}\sum_{d\in D} dw_{s,d} Y_{s,d} + \sum_{s\in S}\sum_{d\in D}\sum_{r\in R} rw_s RM_{s,d,r} \qquad (1)$$

s.t.

$$\sum_{d\in D}\sum_{r\in R} X_{s,d,r} \leq 1 \quad , \quad \forall s \in S \qquad (2)$$

$$\sum_{s\in S} X_{s,d,r} \leq cap_{d,r} \quad , \quad \forall d \in D,\ \forall r \in R \qquad (3)$$

$$\sum_{s\in S}\sum_{r\in R} X_{s,d,r} \leq lim_d \quad , \quad \forall d \in D \qquad (4)$$

$$\sum_{r\in R} X_{s,d,r} = Y_{s,d} \quad , \quad \forall s \in S, \forall d \in D \qquad (5)$$

$$\sum_{ss\in S(ss\neq s)} r_{s,ss} X_{ss,d,r} \geq RM_{s,d,r} \quad , \quad \forall s \in S, \forall d \in D, \forall r \in R \qquad (6)$$

$$(cap_{d,r} - 1) X_{s,d,r} \geq RM_{s,d,r} \quad , \quad \forall s \in S, \forall d \in D, \forall r \in R \qquad (7)$$

$$X_{s,d,r} \in \{0,1\} \quad , \quad \forall s \in S, \forall d \in D, \forall r \in R \qquad (8)$$

$$RM_{s,d,r} \geq 0 \quad , \quad \forall s \in S, \forall d \in D, \forall r \in R \qquad (9)$$

$$Y_{s,d} \geq 0 \quad , \quad \forall s \in S, \forall d \in D \qquad (10)$$

In our model, we have three sets: students (S), dormitory types (D) and rooms (R).

The parameters are split into two parts as weight parameters and capacity parameters. One of the weight parameters is crude placement parameter **(w_s)** that is calculated for each student when they are placed to any room type. Other weight parameter is **($dw_{s,d}$)** dormitory type weight parameter. When student s is not placed to his/her preferred dormitory type-d, the point is set to a negative value. Our last weight parameter is preferred roommate weight parameter **(rw_s)**. This point is gained when the student s is matched with his/her preferred roommate.

We have two parameters about capacity. First of them is **$cap_{d,r}$** which shows the capacity for dormitory type d and room r. This parameter is used to ensure that dormitory rooms can only take students as many as their capacities. Other capacity parameter is **lim_d** which shows the maximum student number that can be placed to dormitories.

The decision variables specify which room is assigned to which student. **$X_{s,d,r}$** determines whether student s is assigned to room r in dormitory d or not. It is activated if a candidate student is placed in a specific room. Other question that needed to be answered was which dormitory type the student is placed into. To answer this question, we defined **$Y_{s,d}$**. Additionally, the last decision variable that is for answering the number of students who stay in the same room with student s was defined as **$RM_{s,d,r}$**.

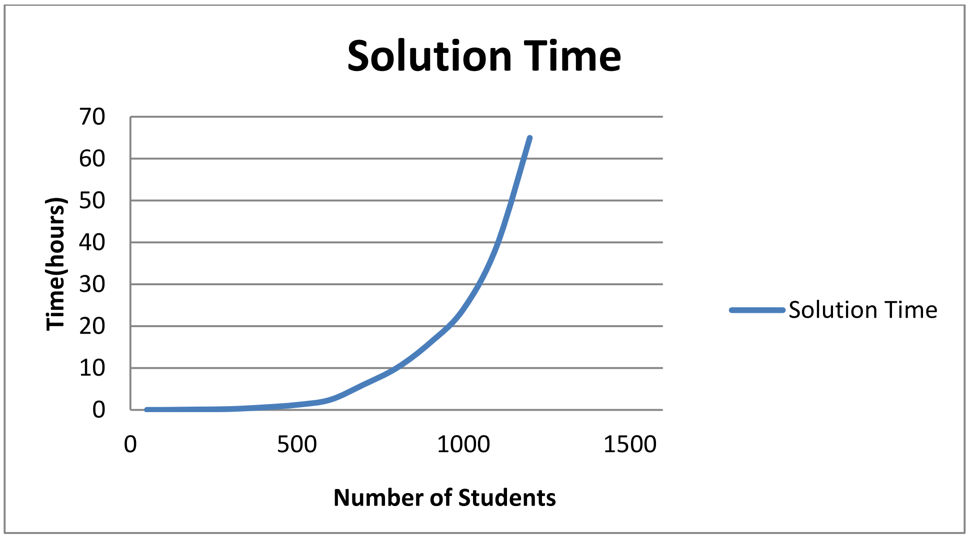
As we have mentioned before, we have three targets in our study: 1) to find students who are eligible to stay in dormitories, 2) to fulfill students' wishes for placement as much as possible and 3) to make their roommate preferences true as much as possible. Because of these different targets, the objective function (1) also has three parts. First is to maximize points gained only from the placement, second is to maximize points gained according to the dormitory type, and the last one is to maximize points gained from roommate preferences. Our objective function is constructed by summing these three objectives and maximizes the overall sum. As mentioned in Section 6.1, the weights were already defined according to the importance of our objectives, so there is no need to make any addition on the summation of these three parts to declare their importance.

In our model, there are three hard constraints. Constraint (2) ensures that all students must be assigned to only one room. Constraint (3) guarantees that dormitory rooms can only take as many students as their capacities. Constraint (4) limits the maximum number of students that can be assigned to the rooms. This constraint also ensures that there can be free rooms for some special cases.

When the sum of the placement point is calculated for each student and each room, if the result is 1, Constraint (5) ensures that the student s is placed to dormitory type-d. Otherwise, it means that the student s is not assigned to this dormitory type-d. Because of Constraint (2), the result is automatically binary, so $\mathbf{Y_{s,d}}$ does not need to be defined as binary variable in the model (Constraint 10). This feature of our model shortens the running time. Additionally, it is known that the lost point when the student is assigned to the un-preferred room is absolutely more than the gained point from the preferred placement. Because our model is a maximization problem, the objective function value will decrease when the student is assigned to the un-preferred room, so the model never tries to assign this student to this un-preferred room type. Constraint (6) limits the total number of desired roommates, and Constraint (7) limits the number of desired roommates in a room to the room's capacity including student s. Both constraints limit $\mathbf{RM_{s,d,r}}$ together, so if one of these constraints do not occur, $\mathbf{RM_{s,d,r}}$ is automatically set to 0.

$\boldsymbol{RM_{s,d,r}}$ and $\boldsymbol{Y_{s,d}}$ are nonnegative integer variables whereas $\boldsymbol{X_{s,d,r}}$ are binary variables.

After we set up our model, for small-large problems, we did not encounter with any problem, and we found the solution. On the other hand, on the constant capacity, when we increase the number of students, it was more difficult to find the result. When the number of students reached 800, finding a solution took almost a day (Graph 5.1). In addition to this, for 2014-2015 academic-year, the number of students reached 1300 for dormitory application, so it was nearly impossible to find a solution in a reasonable time. In addition to this, we have met with a new problem; running out of memory. To fix this problem, we changed our computer. The old computer had 4 GB RAM and i5 processor, but our new computer had 8 GB RAM and i7 processor. Still, we could not solve our problem fast and efficiently. So, we decided to develop new solution methods.



Graph 5.1 Change in solution time change according to number of students

5.3 Decomposite Model

As the composite model's solution times were not acceptable for the real-life problem, a second approach was developed. The decision process was split into two stages where the dormitory-type decisions were followed by the roommate and room decisions (Figure 5.4).

Figure 5.4 Steps of decomposite model

5.3.1 Dormitory Type Model

In the first part of our de-composite model, we decide the students who are eligible to stay in dormitories and match these students with dormitory types without considering the room and rommate preferences.

So, we do not need the set for room types in this model. This provides a meaningful decrease in the number of decision variables and constraints. As it is seen in Table 5.4, there is a huge difference between the number of decision variables and constraint for both models for 1400 students, 9 dormitory types and 80 rooms. So, the optimum solution time decreased to seconds as desired. Table 5.4 also shows the size of the Room Assignment Model which will be discussed in 5.3.2.

Table 5.4 The difference between the number of DVs and constraints for two models

	Composite Model	**Dormitory Type Model**	**Room Assignment Model**
DV	2,028,600	25,200	3,200
Constraint	2,030,720	26,618	9,600

Sets

$\boldsymbol{S}$ = set of candidate students

$\boldsymbol{D}$ = set of dormitory types

Parameters

<u>Weight Parameters</u>

$\boldsymbol{w_s}$= weight used when candidate student **s** is placed in any dormitory.

$\boldsymbol{dw_{s,d}}$ = weight used when candidate student **s** is placed in dormitory type **d**.

$\boldsymbol{rw_s}$= weight used when candidate student **s** is placed with his/her preferred roommate(s).

<u>Other Parameters</u>

$\boldsymbol{r_{s,ss}}$ = 1 if student **s** prefers student **ss** as a roommate, 0 otherwise.

$\mathbf{caproom_d}$ = capacities the rooms in d dormitory type

$\mathbf{capdorm_d}$ = total capacity in dormitory type d.

$\boldsymbol{lim_d}$ = maximum permitted number of students for dormitory type **d**.

Decision variables

$\boldsymbol{X_{s,d}}$ = 1 if candidate student **s** is placed **d**-type dormitory 1, 0 otherwise.

$\boldsymbol{RM_{s,d}}$ = total number of desired roommates for candidate **s** for dormitory type **d**.

The model is as follows:

$$max\ z = \sum_{s\in S}\sum_{d\in D}(w_s + dw_{s,d})X_{s,d} + \sum_{s\in S}\sum_{d\in D} rw_s RM_{s,d} \qquad (1)$$

s.t.

$$\sum_{d \in D} X_{s,d} \leq 1 \quad , \quad \forall s \in S \tag{2}$$

$$\sum_{s \in S} X_{s,d} \leq lim_d \quad , \quad \forall d \in D \tag{3}$$

$$\sum_{ss \in S (ss \neq s)} r_{s,ss} X_{ss,d} \geq RM_{s,d} \quad , \quad \forall s \in S, \forall d \in D \tag{4}$$

$$(cap_{d,r} - 1) X_{s,d} \geq RM_{s,d} \quad , \quad \forall s \in S, \forall d \in D \tag{5}$$

$$X_{s,d} \in \{0,1\} \quad , \quad \forall s \in S, \forall d \in D \tag{6}$$

$$RM_{s,d} \geq 0 \quad , \quad \forall s \in S, \forall d \in D \tag{7}$$

We identified two new parameters as **$caproom_d$** and **$capdorm_d$** instead of **$cap_{d,r}$**. **$caproom_d$** is the parameter that shows the capacities of the rooms in d dormitory type, and **$capdorm_d$** is the parameter that shows the total capacity in dormitory type d.

We removed r set from **$X_{s,d,r}$** and **$RM_{s,d,r}$** that are the main decision variables, and show that the room type that the student is assigned to and occured roommate preference, respectively. And we removed the decision variable that shows the room type that the student is placed to completely.

The new objective function (1) is also constituted of three main weight parts like in our composite model, but the difference is that we do not need different summation notation for placement points and room type preference anymore because we can show the placement condition and the placed dormitory type with only one decision variable.

Like in the composite model, all students can be assigned to only one room (Constraint 2). In addition to this, Constraint (3) ensures that dormitory rooms can only take students as many as their capacities. More than dormitory capacities cannot be assigned to the students.

Because we ignore the room factor in this model, we do not need a constraint to decide the dormitory type for the student. So, we can move to the roommate preference that is the second wish of students. Constraint (4) limits the total number of desired roommates, and Constraint (5) limits the number of desired roommates in a room to the room's capacity including student s. Both constraints limit $\mathbf{RM}_{\mathbf{s,d}}$ together, so if one of these constraints do not occur, $\mathbf{RM}_{\mathbf{s,d}}$ is automatically set to 0.

$\boldsymbol{RM_{s,d}}$ are nonnegative integer variables whereas $\boldsymbol{X_{s,d}}$ are binary variables.

When we solve this part of the model in the same computer that we used in composite model, we saw the optimal solution time decreased to seconds. Only 8 students showed differences when we solved two models with 1400 students. It was an acceptable solution.

5.3.2 Room Assignment Model

After decided the dormitory types for students, we had a new problem for assigning of students to the room types. We used the output of dormitory type model as input in room assignment model to assign the students into the rooms. Because we produced solution according to our three main targets in the first model, we only had to use these results correctly in this model.

Sets

$\boldsymbol{S}$ = set of candidate students

$\boldsymbol{R}$ = set of rooms

Parameters

$r_{s,ss}$ = 1 if student **s** prefers student **ss** as a roommate, 0 otherwise.

cap = capacity of dorms.

Decision variables

$X_{s,r}$ = 1 if candidate student **s** is placed **d**-type dormitory room **r** 1, 0 otherwise.

$RM_{s,r}$ = total number of desired roommates for candidate **s** for dormitory type **d** room **r**.

The model is as follows;

$$max\ z = \sum_{s \in S} \sum_{r \in R} RM_{s,r} \quad (1)$$

s.t.

$$\sum_{r \in R} X_{s,r} = 1 \quad , \quad \forall s \in S \quad (2)$$

$$\sum_{s \in S} X_{s,r} \leq cap \quad , \quad \forall r \in R \quad (3)$$

$$\sum_{ss \in S (ss \neq s)} r_{s,ss} X_{ss,r} \geq RM_{s,r} \quad , \quad \forall s \in S, \forall r \in R \quad (4)$$

$$(cap - 1) X_{s,r} \geq RM_{s,r} \quad , \quad \forall s \in S, \forall r \in R \quad (5)$$

$$X_{s,r} \in \{0,1\} \quad , \quad \forall s \in S, \forall r \in R \quad (6)$$

$$RM_{s,r} \geq 0 \quad , \quad \forall s \in S, \forall r \in R \quad (7)$$

There are changes in the sets now. We do not need the set of ***d*** (dormitory type) anymore because we will get the solutions for each dormitory types separately in this model. We defined a new set as $\boldsymbol{r}$ (rooms) to show the rooms in dormitory types that we will make the placement. And the last new set is $\boldsymbol{s}$ (students) that represents the students who will be assigned to the rooms.

The parameters that we will use again in this model are $\boldsymbol{r_{s,ss}}$ that shows the demands of the students for roommate preferences and $\boldsymbol{cap}$ that shows the capacities of these rooms. Because the capacity of rooms are the same for each dormitory type, it is a constant value for this model.

We have two decision variables in this model. The first thing that we have to decide is that $\boldsymbol{X_{s,r}}$ that decides the room that students are assigned to. It is a binary decision variable because the candidate student is assigned to the room or not. The second decision variable is $\boldsymbol{RM_{s,r}}$ that shows similarity to the decision variable that we used in the first model and decides the roommate for the student $\boldsymbol{s}$. This decision variable takes integer values.

The target is to meet the roommate preferences as much as possible. We do not have to give weights to the students as in the composite model and the first model of the decomposite model. Hence, the objective function (1) maximizes the number of satisfied roommate preferences.

Constraint (2) ensures that a student is assigned to only one room. Constraint (3) ensures that we do not exceed the capacity of a room. Constraints (4) and (5) again show the roommate condition as in the composite model and the first model of the de-composite model. Constraint (4) limits the total number of desired roommates, and Constraint (5) limits the number of desired roommates in a room to the room's capacity including student s. $\boldsymbol{RM_{s,r}}$ are nonnegative integer variables whereas $\boldsymbol{X_{s,r}}$ are binary variables.

With our current number of dormitory capacities and rooms, this model gives the optimal solution in an acceptable time. The longest solution time is observed in Orange dormitory types that has 45 rooms and the capacity of each room is 2. Still, the solution time is under three minutes when we solve this model on a computer with 5 GB RAM and i5 processor. Even if we increase the data 5 times of the old data, the solution time does not exced 5 minutes.

CHAPTER 6

HEURISTIC APPROACH

As we said before, we solved the existing placement by separating mathematical programming model into two parts. Even in this case, we observed that the processing time in the placement part to the rooms for some dorm types increased too much, so we encountered with a new question that we can do the placement with more dorm types, more capacity and more demand in a reasonable time, or not. After experiments, we saw that the processing time increases exponentially with the increase of dimension of the mathematical model. Thus, we thought that we should find a solution in polynomial time for the next years.

In the previous chapter, the dormitory type allocation was handled via linear integer model. However, the same problem can also be solved via a Gale and Shapley [1] algorithm. Gale and Shapley algorithm finds a stable solution to the Stable Marriage Problem. It works as follows (Figure 6.1).

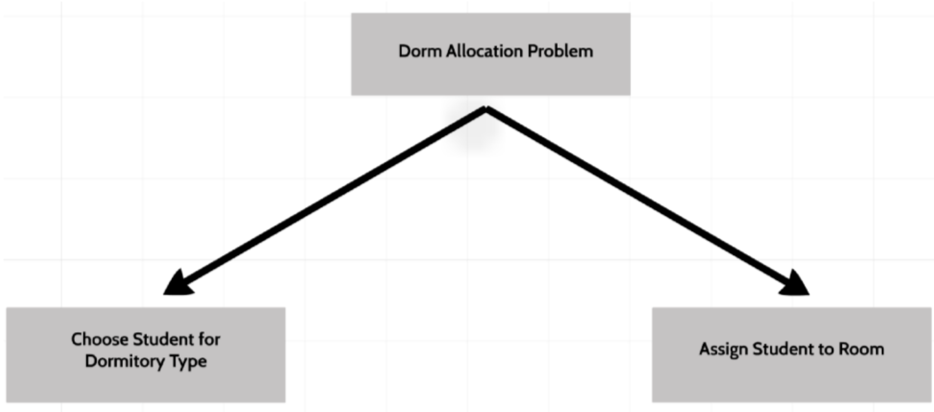


Figure 6.1 Dorm allocation problem

Gale and Shapley Algorithm is an algorithm which tries to find optimum matching between a group of women and men. Each man and woman has ideas about opposite sex, and there is a sorting according to these ideas in the algorithm. For instance, the first choice of man-A is to marry woman-B, his second choice is woman-C and his third and last choice might be woman-D. The algorithm seeks the best matching to make all these people happy (optimum result). A stable assignment is called optimal if every applicant is at least as well off under it as under any other stable assignment. Let us explain this situation with an example. Table 6.1 shows us the preference lists of two women and men.

Table 6.1 Example preference lists of two women and men

	Men's Preferences	
Men	1	2
A	C	D
B	D	C

	Women's Preferences	
Women	1	2
C	A	B
D	A	B

We have A-C and B-D matchings. The changes on this matching cannot lead to happiness for anyone. For example, if we match A with D, while the happiness of D is increasing (because D matches with her first preference), the happiness of A is decreasing (because he matches with his second preference rather than his first preference). If the same situation occurs in other matchings, it means that there is stable matching.

In our problem, the opposite sets correspond to dormitories and students. However, we needed to make some changes to adapt the Gale and Shapley Algorithm to our problem. For example, in the Gale and Shapley Algorithm, there is one to one correspondence, but we have more than one student who stays in dormitories, so we made the algorithm suitable for this. It means that we ensured that number of students and dormitory capacities could be matched in our new algorithm.

Additionally, it is accepted that the dimensions of two sets in the Gale and Shapley Algorithm are equal, and there are preferences for the opposite sex. But these two assumptions do not exist in our problem, so we created new solutions for these cases.

The changes in the algorithm gave us the answer of the question that who should stay in the dormitories, but we should assign these students to the rooms also. So, in the second step of our Heuristic, we focused on this sub-problem, and we developed two different solution strategies for the solution of this problem.

6.1 Dormitory Allocation

We mentioned before that we made many changes to the Gale and Shapley Algorithm to choose the students who will stay in dormitories. Additionally, it was important to create input sets to run this algorithm. There are three basic input sets to use in the algorithm: students' preference list, dormitories' preference list and dormitories' capacities. Let us tell how we created these input sets.

We asked the students which dormitory they want to stay in and which dormitory they do not want to stay in on the application form. We used the weights which we calculated according to given answers to these questions on the mathematical modelling part. Additionally, we created the students' preference lists according to these weights. Differently from the mathematical modelling, we did not add the dormitory type which they do not want to stay in to their preference lists, so they never were placed to these dormitory types because Gale and Shapley Algorithm works with the idea that a set offers to the other set, and the other set accepts this offer. If the students do not offer the dormitory types which they do not want stay in, the possibilities for matching are removed. Additionally, we gave equal weights to the dormitory types which they did not reject to stay in and added them to their preference lists randomly. You can see the sample preference list for a student group in Table 6.2. It can be easily observed that some students just prefer 1 type of room, but others prefer all room types. The basic reason is that some students just use 1 of their 3 choices in the application form and added the dorms where they do not want to stay in, but other students accepted to stay in all kind of dormitories.

To create out other input which is dormitories preference list, we decided to use the students' placement points because these points are calculated by using important criteria such as the students' home addresses, classes or ages. These criteria are important for the university management when choosing the students who will stay in dormitories. Students with high points are preferred by the university to stay in dorms. It was enough to use only one list for all dormitories because dormitory placement criteria do not change according to dormitory types. You can see a sample dormitory preference list for a student group in Table 6.3.

Since dormitory capacities are fixed, we did not have to make any changes to them.

Table 6.2 Dormitory weights for sample students

Student	Dormitory Weight								
	Dorm 1	Dorm 2	Dorm 3	Dorm 4	Dorm 5	Dorm 6	Dorm 7	Dorm 8	Dorm 9
Std 1	0.0135	-90	-90	-90	-90	-90	-90	-90	-90
Std 2	0.0054	-36	-36	-36	-36	-36	-36	-36	-36
Std 3	-90	0.0135	0.009	-90	-90	-90	-90	-90	-90
Std 4	0.00009	0.00009	0.00018	0.00036	-3.6	-3.6	-3.6	0.00054	0.00009
Std 5	-6.3	-6.3	0.000315	0.0001575	0.0001575	0.0001575	0.0001575	0.00063	0.000945
Std 6	-90	-90	0.00225	0.00225	-90	0.0135	0.009	0.0045	-90
Std 7	-6.3	-6.3	-6.3	-6.3	0.0001575	0.000945	-6.3	-6.3	-6.3
Std 8	0.0054	-36	-36	-36	-36	-36	-36	-36	-36
Std 9	0.000945	-6.3	-6.3	-6.3	-6.3	-6.3	-6.3	-6.3	-6.3
Std 10	-90	-90	-90	-90	-90	0.0135	-90	-90	-90

Student	Student Choices								
	Dorm 1	Dorm 2	Dorm 3	Dorm 4	Dorm 5	Dorm 6	Dorm 7	Dorm 8	Dorm 9
Std 1	1	-	-	-	-	-	-	-	-
Std 2	1	-	-	-	-	-	-	-	-
Std 3	2	3	-	-	-	-	-	-	-
Std 4	8	4	3	1	2	9	-	-	-
Std 5	9	8	3	4	5	6	7	-	-
Std 6	6	7	8	3	4	-	-	-	-
Std 7	6	5	-	-	-	-	-	-	-
Std 8	1	-	-	-	-	-	-	-	-
Std 9	1	-	-	-	-	-	-	-	-
Std 10	6	-	-	-	-	-	-	-	-

Table 6.3 Sort candidates

ID	Name	Weight
1	Std 1	45
2	Std 2	18
3	Std 3	44
4	Std 4	1.8
5	Std 5	3.15
6	Std 6	45
7	Std 7	3.15
8	Std 8	16
9	Std 9	3
10	Std 10	40

Sort Candidates

Dorm Pref.	Std ID
1 Choose	1
2 Choose	3
3 Choose	6
4 Choose	10
5 Choose	2
6 Choose	8
7 Choose	5
8 Choose	7
9 Choose	9
10 Choose	4

Before telling how our heuristic works, we want to give some information about how Gale and Shapley algorithm works. Firstly, one set is created as women set, and other set is created as men set. The elements of set of men propose their first preferences on their lists. If the woman did not take any proposal before, she accepts him temporarily until she receives the proposal from a better man (better means that on higher position in woman's preference list), so if she receives a proposal from this better man, she rejects the first man, and accepts the second man. The rejected man proposes his other woman in his preference list. It continues until all men are matched.

We must not forget that men are honorable in this algorithm. It means that men do not propose the women that they proposed before again. At the result, we see an optimum matching.

We can encounter with two results after Gale and Shapley Algorithm. Depending on who proposes first, optimum solutions change. Hence, two different solutions are obtained when students or dormitories are the first proposes. On the other hand, we observed that there is no difference between the optimum matching according to the students and the optimum matching according to the dormitories. The reason for this is that dormitories use the same preference list with the students, so the decision-maker is the student set in our problem.

We can easily see how the process works in Figure 6.2.

Now, we can explain how our algorithm works.

Definitions

"Free Capacity": Is there any free capacity in all dormitories for students?
"Free Dormitory": Is there any free capacity in the dormitory type which is proposed by STD (i)?
"All Match": Are all students matched with dormitories?
"All Propose": Do all non-matching students propose to their last preference?

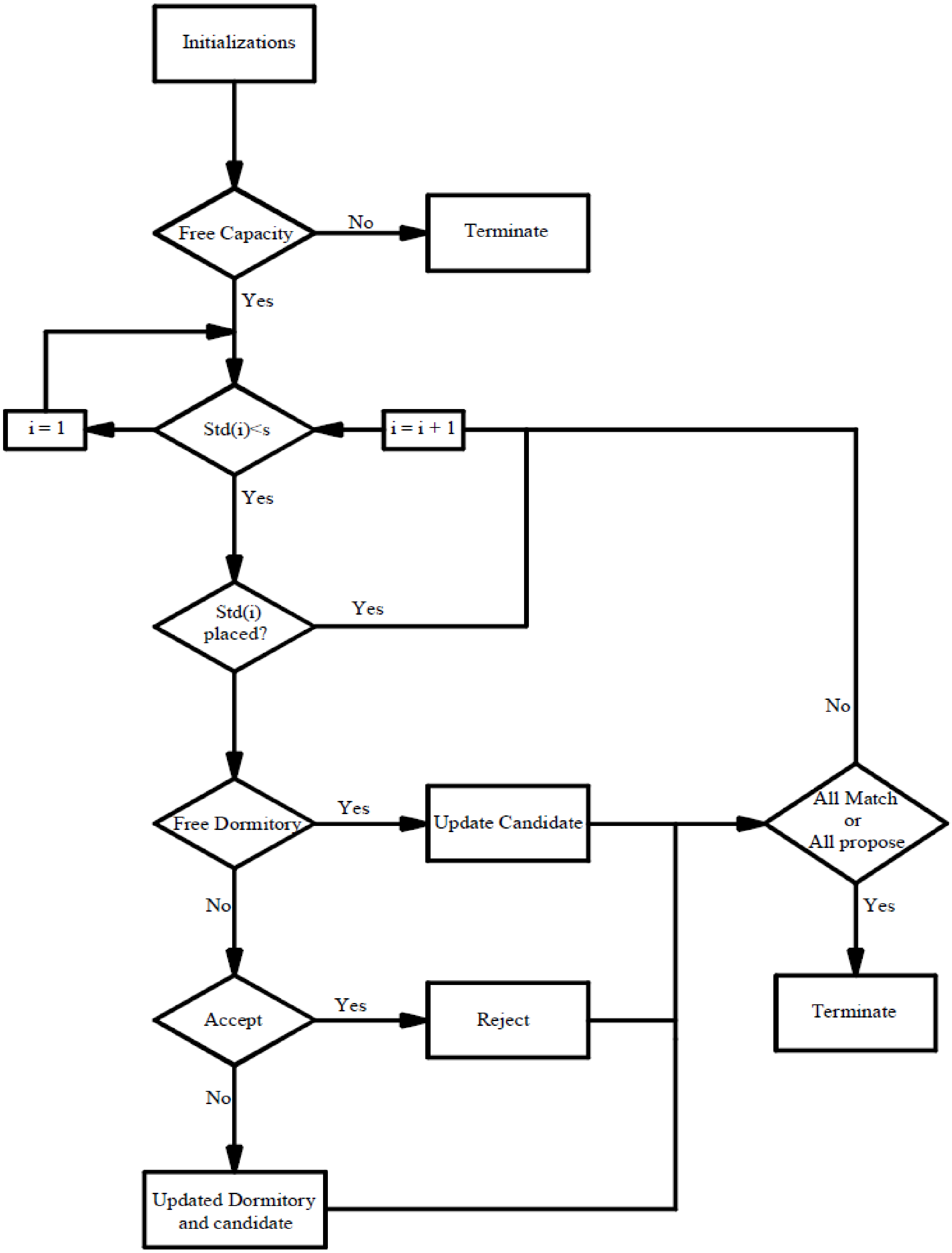


Figure 6.2 Placement process for heuristic

"Accept": Is the STD (i) on a higher position on the dormitory preferences list than the last placed student?
"Updated Candidate": STD (i) is added to the placement list of dormitory which is proposed by STD (i) and the last student on the dormitory preference list is updated.
"Reject": STD (i)'s proposal is assigned as rejection.
"Updated Dormitory and Candidate": The last student is in the dormitory's placement list is removed from the list and his/her proposal is assigned as rejection. Than STD (i) is added to the placement list. Finally the last student is updated on the dormitory preference list.

"All Match" and "All Propose" are the control statements that will decide that the process will continue or not. The possibility for realization is pretty low on practical implementations because as it is seen from last years, the main problem is that the demand for dormitories is more than the capacities of dormitories, so our algorithm can terminate with "All Propose". Let us say that we have n number of students, m kind of dormitories and assume that each dorm's capacity is 1 and all students accept to stay in all kinds of dormitories. Even in this worst case, the complexity of the problem is O(n.m). For example, we have 3000 students and 9 dormitory types. The first student's point is 1, and 3000. Students' point is 3000, and we assume that all dormitories' capacity is 1. Students' dormitory preferences are the same. The first preference of students for dormitory type is dorm-1, and the last preference is dorm-9. The first student proposes to the first dorm, and because this dorm is free now, he/she is accepted to this room. Then the second student comes for the first dormitory again. Because he/she has higher points than the first student, the first student is removed from dormitory type-1, and the second student is added to this room. This situation continues until the last. Student is placed for the dormitory type-1. Then the first student proposes to his/her second choice, and the same situation occurs for the second room type. All process is terminated after 9 dormitory types are full with the last 9 students of 3000. So the process generally occurs for 9*3000 times. On the other hand, we know that dormitory capacities are more than 1, so our algorithm gives us the result in a reasonable time.

6.2 Room Allocation

After deciding which students deserve to stay in dormitories, the next step is to assign these students to the rooms. Our first target in this step is to assign the students with their roommate preferences if both deserve to stay in dormitories. We tried two approaches to reach this target which are a simple heuristic and the second part on the decomposed model.

We start our simple heuristic with two basic assumptions. First, we assume that all students who make application are rational people. It means that we assume that students know how many students can stay in their room preference, so they choose their roommates according to this number. The maximum roommate demand for a student cannot exceed dormitory room capacity minus one. For example, a student who wants to stay in an orange room can choose only one student as his/her roommate. Next, we assumed that the roommate preferences are symmetric for the students. For example, if student-A chooses student-B as his/her roommate, student-B also chooses student-A as his/her roommate symmetrically.

After these assumptions, we assigned the students with a simple mechanism. Firstly, the students are assigned to free dormitory rooms with their roommates. Then, the rest of the students are assigned to free beds in dormitory rooms.

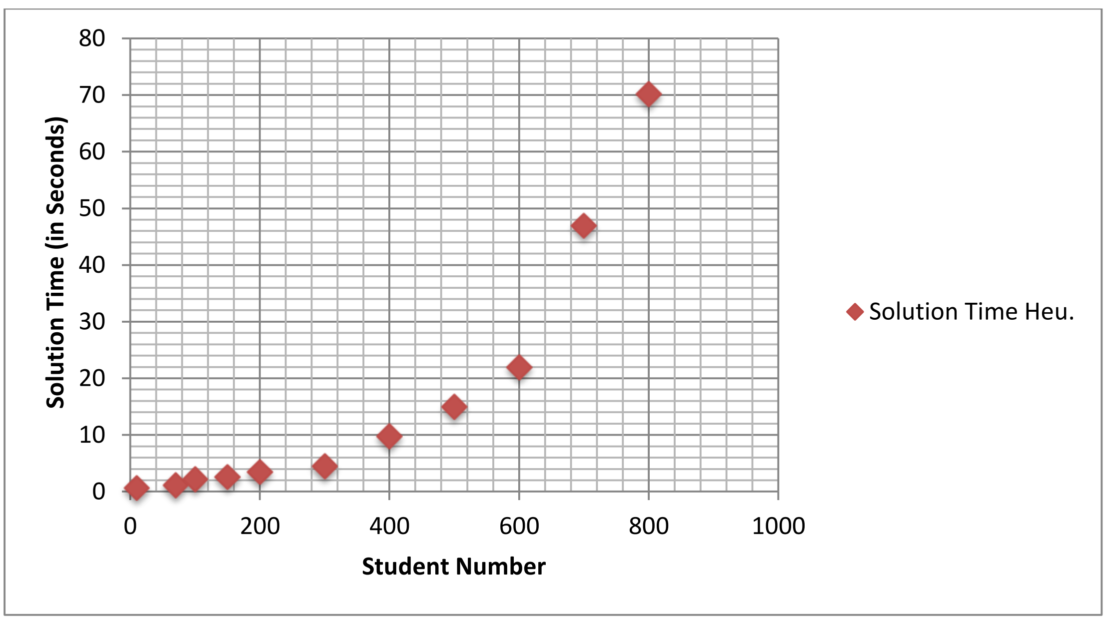
The second approach we used is the second model of our decomposed model. This model is used for assigning the students whose dormitory rooms are definite. We here already decided which students will stay in dormitories such as in the decomposed model, so we could easily use this model.

If we compare these two approaches, the processing times for both are really fast. The only problem here is we did not pay attention some factors which we cared about in decomposed model such as class differences of roommates, but as we said before, our basic target is to assign students with their roommates when they both deserved to stay in the same dormitory room.

6.3 Analysis of Heuristic

The heuristic which we developed is a solution for our lack of time problem. Even on the experiments with 1000 students, the processing times are seen as reasonable.

You can see the experiment results for 9 dormitory types and different group of students on the following Graph 6.1.



Graph 6.1 Experiment results for 9 dormitory types

We can say that the base period is taken by our heuristic mostly, but if we compare it with the mathematical approach, we can tell that this processing time is really fast with regard to time.

We can say that the priority of this heuristic is to choose the students according to the placement criteria of the university. Additionally, the roommate preference is irruptively more important according to this model approach. When we look at the results, we can see that the accruement rate for meeting roommate preferences dropped very much. But when we look at the results placement results for room preferences, it is observed that the placement rate for the students who have higher points is really high than the placement rate for the students who have lower points.

CHAPTER 7

DECISION SUPPORT SYSTEM

To meet the expectations of the users and to design a user-friendly interface, we arranged meetings with the dormitory office personnel and included them in the design process of the system. The system was developed according to their wishes and was designed with ease-in-use in mind.

We had many goals in the DSS design. Presenting efficient reports about student applications and dormitory capacities, helping the user to make better decisions with these reports, making the overall process easier, creating useful result reports after dormitory placements and presenting the outcome of applications on a form which the dormitory office wanted to see.

We separated our design into three parts: inputs, solving and results. Before mentioning these parts one by one, we want to talk about the Dashboard of the DSS.

The user sees the Dashboard when the DSS is started (Figure 7.1). The user can make any changes that he/she wants on this screen. He/she can see the results what he/she want to desire to see and create the results.

Figure 7.1 Dashboard of the DSS

DSS provides the user much functionality to change the parameters in the system. For these changes DSS uses approximately seventy Excel sheets. Some sheets are sheets for the user to use and some of them are used internally by the system, and the user cannot take any action on these sheets. We avoided the user to get lost in the system and made the system as user-friendly as possible.

Additionally, the user can change the data by using some Excel Sheets. We gave limited rights to the user to change data on these sheets. By using Data Verification Function, we wanted to minimize the problems which occur by the user when entering input data.

The first part on the main page is reserved for inputs. We gave three rights to the user to change on this part and created efficient reports about the input data.

The first input is information about students. The user can pull the applications from the on-line application system and can make some changes on the applications. The reason to let these changes is that many amended requests come from the students after the applications are completed. Some reasons for changes are mistakes in the addresses for example. But the main amended request is to change the room types or roommates. We developed two approaches for this problem. First was to ignore the wishes to change after the application process is completed, and the second one was to take these changes into consideration.

To put the first approach into practice would be contrary to the main policy of the university because it would decrease satisfaction level of the students visibly. But to distinguish amended requests between the applications during the on-line application process is really difficult, so we limited the fields for the user to change. Additionally, as it is seen on Figure 7.2, we minimized the errors generated by the user by letting the user to pull the data from the list-box such as living city, county or age of the student.

After taking input data, we added other important component of our system which is the dormitory part. It was important to make changes on dormitory types according to the years because dormitory office had rights to make changes on the dormitory types. For example, they returned blue dorms for 2 people into for 4 people by adding bunk beds in 2013-2014 academic year. Additionally, it was possible to assign some rooms for research assistants or administrative personnel in some cases. So, the dormitory office needed to close some rooms for these people.

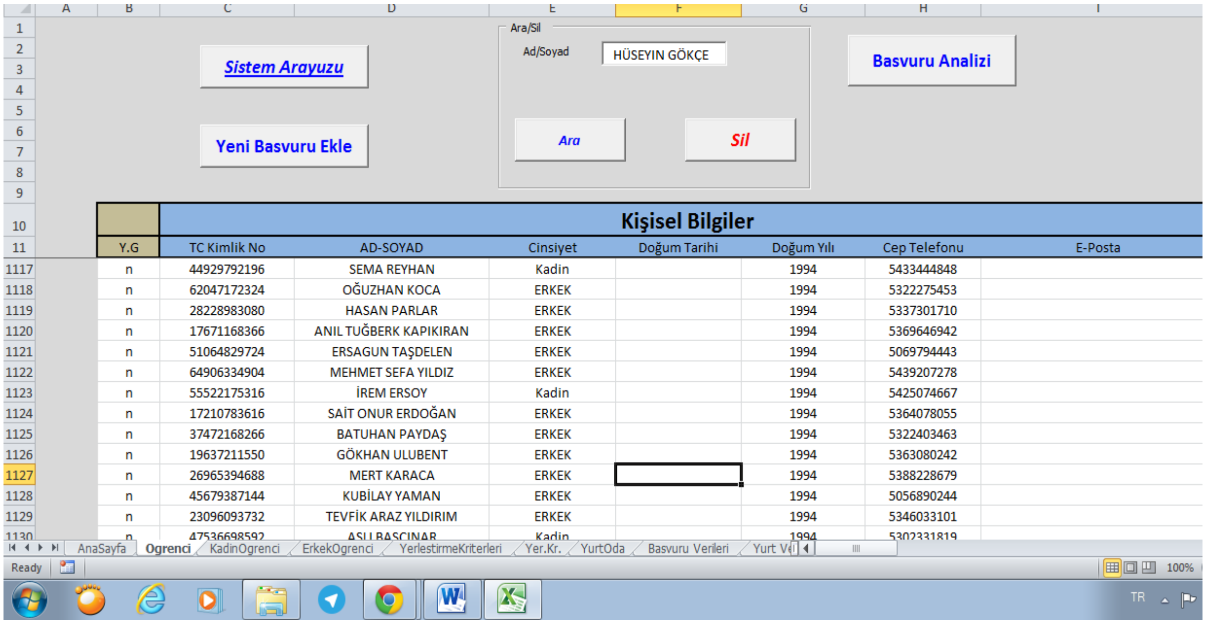
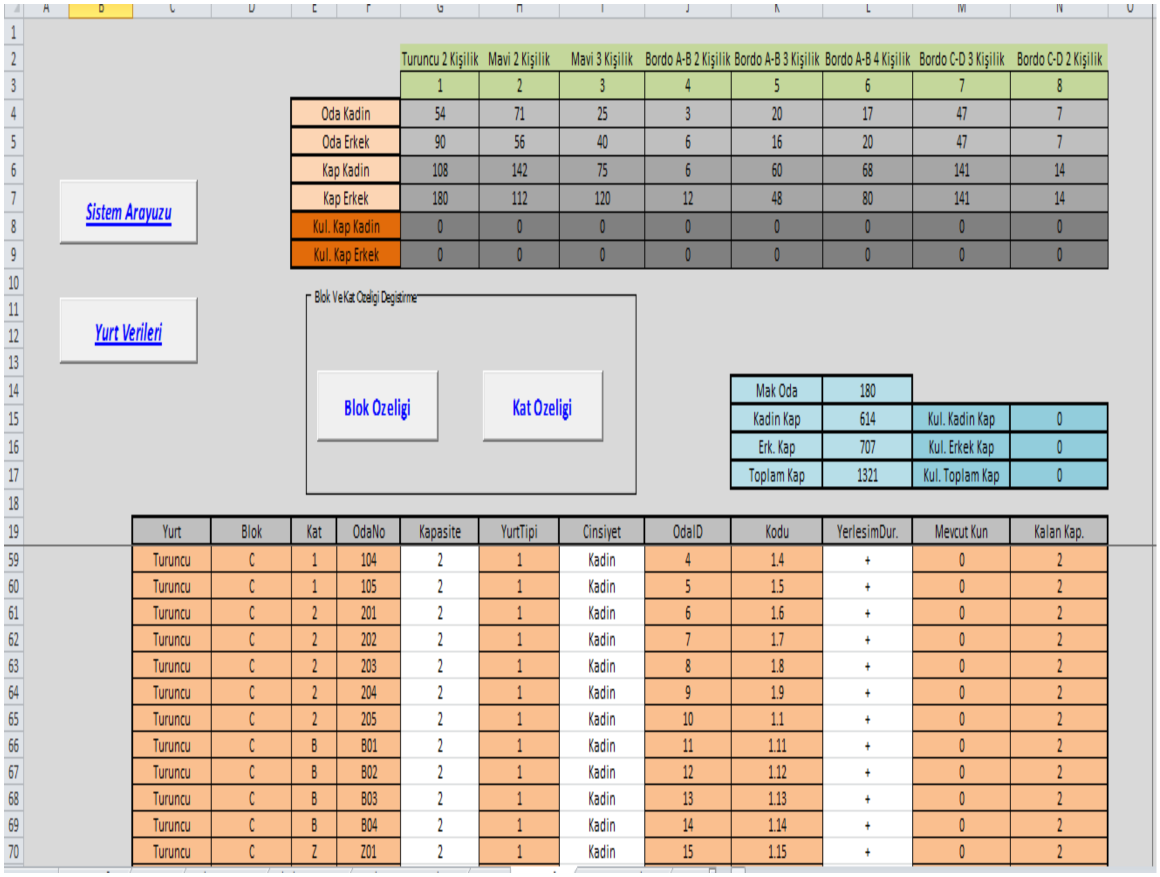


Figure 7.2 Pulling candidate data

For all these reasons, we prepared a sheet for the user to change the dormitory types. As same as in students sheet, we tried to make the change process easiest as much as possible for the user by using the functionality in Excel. In addition to this, we added a table which shows the effects of the changes which the user did on the dormitory capacities.

Another provided convenience for the user was the chance to make these changes on the Dashboard rather than going to these special Excel sheets like when deleting some old data about the students or closing some room types for the special students. For minimizing the errors of the user on the input, the user can also take back-ups of the datasets (Figure 7.3).

We created a dynamic system for the university to be used for many years. The user can enter new data and add new dormitory types when needed.



	Turuncu 2 Kişilik	Mavi 2 Kişilik	Mavi 3 Kişilik	Bordo A-B 2 Kişilik	Bordo A-B 3 Kişilik	Bordo A-B 4 Kişilik	Bordo C-D 3 Kişilik	Bordo C-D 2 Kişilik
	1	2	3	4	5	6	7	8
Oda Kadin	54	71	25	3	20	17	47	7
Oda Erkek	90	56	40	6	16	20	47	7
Kap Kadin	108	142	75	6	60	68	141	14
Kap Erkek	180	112	120	12	48	80	141	14
Kul. Kap Kadin	0	0	0	0	0	0	0	0
Kul. Kap Erkek	0	0	0	0	0	0	0	0

Mak Oda	180		
Kadin Kap	614	Kul. Kadin Kap	0
Erk. Kap	707	Kul. Erkek Kap	0
Toplam Kap	1321	Kul. Toplam Kap	0

Yurt	Blok	Kat	OdaNo	Kapasite	YurtTipi	Cinsiyet	OdaID	Kodu	YerlesimDur.	Mevcut Kun	Kalan Kap.
Turuncu	C	1	104	2	1	Kadin	4	1.4	+	0	2
Turuncu	C	1	105	2	1	Kadin	5	1.5	+	0	2
Turuncu	C	2	201	2	1	Kadin	6	1.6	+	0	2
Turuncu	C	2	202	2	1	Kadin	7	1.7	+	0	2
Turuncu	C	2	203	2	1	Kadin	8	1.8	+	0	2
Turuncu	C	2	204	2	1	Kadin	9	1.9	+	0	2
Turuncu	C	2	205	2	1	Kadin	10	1.1	+	0	2
Turuncu	C	B	B01	2	1	Kadin	11	1.11	+	0	2
Turuncu	C	B	B02	2	1	Kadin	12	1.12	+	0	2
Turuncu	C	B	B03	2	1	Kadin	13	1.13	+	0	2
Turuncu	C	B	B04	2	1	Kadin	14	1.14	+	0	2
Turuncu	C	Z	Z01	2	1	Kadin	15	1.15	+	0	2

Figure 7.3 Room list

The most dynamic part in our project was specification of placement criteria. Many criteria have their default values specified. New values can be specified by the user. Additionally, the user can add new criteria (Figure 7.4).

Moreover, the user can change the order of importance for these criteria. Additionally, we let the user to see a summary of current criterion as it is seen on Figure 7.5.

Proper data input is very important for running the system efficiently. To help the user to make decisions we prepared detailed reports. We can separate these reports into two parts: application reports and dormitory reports.

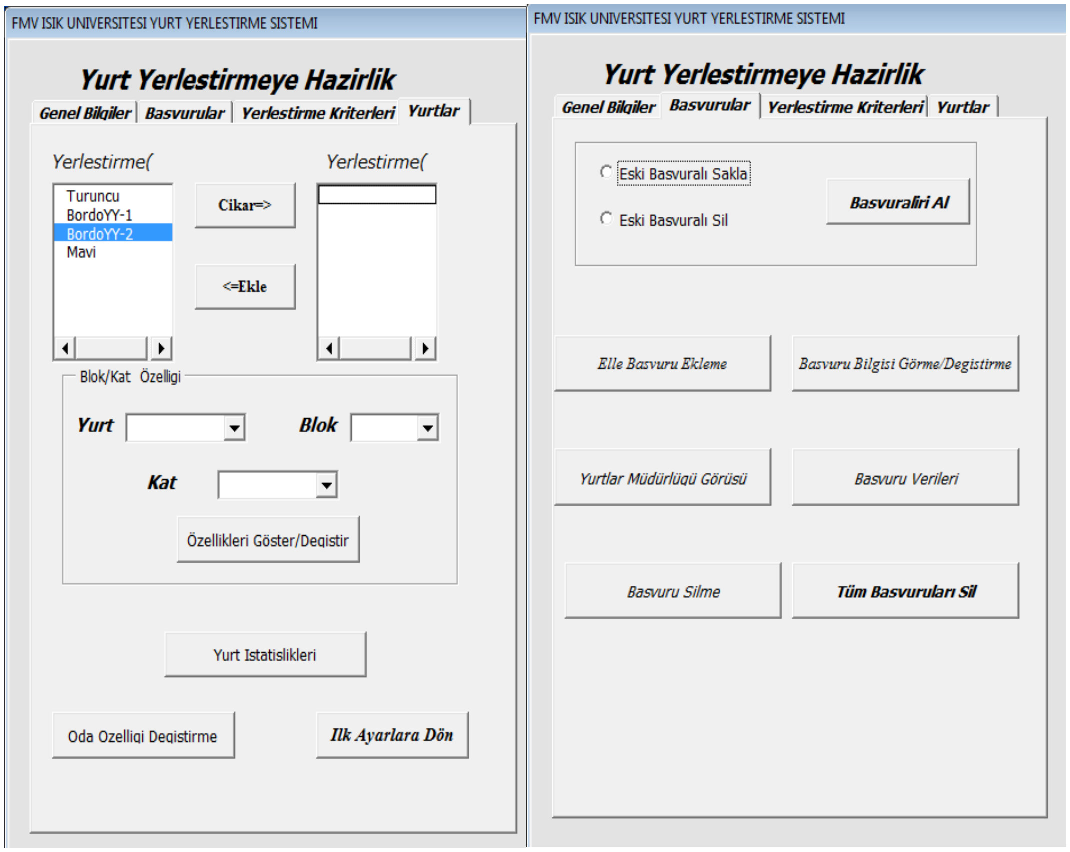


Figure 7.4 Placement system

We separated the reports for the dormitories into three parts (Figure 7.6 and 7.7). First part was general reporting for all dormitories. In this part, we gave comments without making any gender gap. As we know, we separated the dormitory types according to the genders on the placement. But, in this report we wanted to show a general table to the dormitory office and university management. This report gives answers for the following questions. What are dormitory capacities? What percentage of dormitories is reserved for women? What percentage of dormitories is useable?

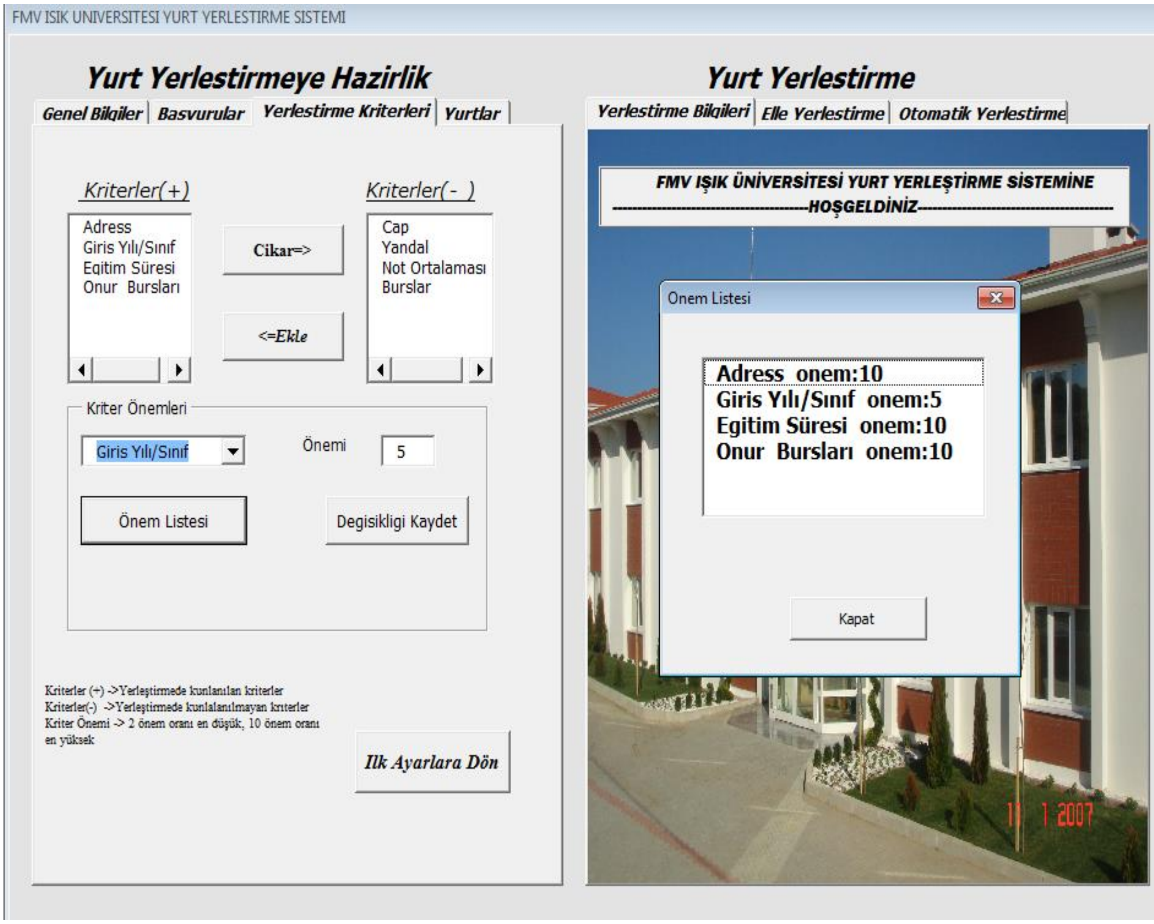


Figure 7.5 Interface for placement system

The main target for the reports by separating according to female and male students was to see the distribution of demands and capacities in gender. For example, after these reports we saw that demand of women for Orange dormitories is much more than the capacity of Orange dormitories for women. On the other hand, demand of male students for Orange dormitories is less than the capacity of Orange dormitories for male students. So, we changed a gender of one block of Orange dormitories. We increased satisfaction of students without making any optimization process.

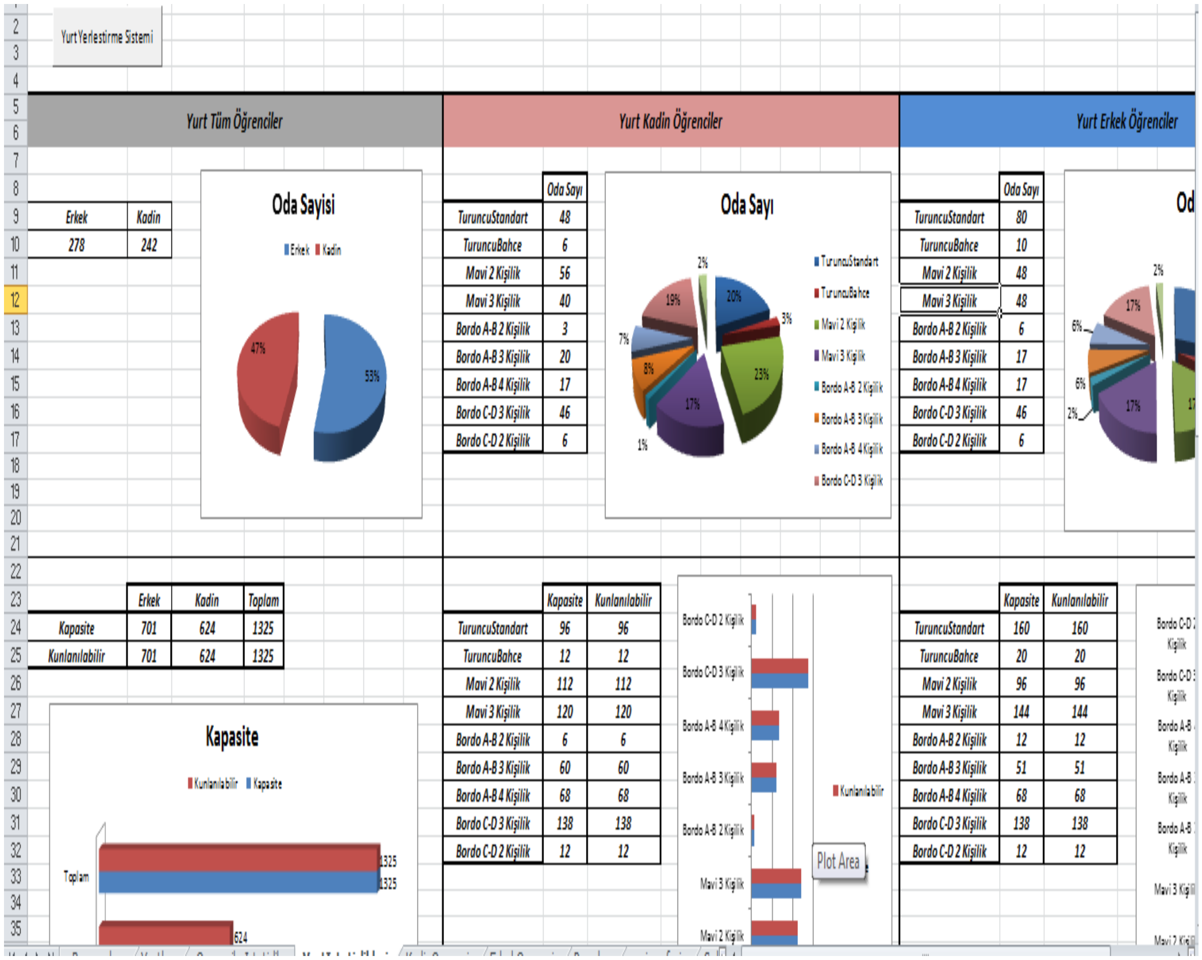


Figure 7.6 Reports 1

This example shows us the importance of making right comments on taken inputs and how this right comment increases the success of the system. So, we saw that the comments on taken input are as much as important of the accuracy of the taken input. In second reports part, we presented reports according to the students. For example, we reported some placement criterion such as home addresses, classes or GPAs of students. There reports proved us these taken data and criterion are enough to place the students. We could need some extra criterion to add if we saw that the given criterion are not enough to distinguish the students on the placement.

These reports also show us demand of students for the dormitory types. Additionally, we showed the second and third room preferences of the students in this second report part. So, we saw the most preferable room types. This representation helps the university management for deciding new investments about dormitories for the following years.

Additionally, we reported the dormitory types where the students do not want to stay. The dormitory office saw that in which dormitories to stay make the students unhappy. Still, we saw that none of the dormitory types makes the students very unhappy.

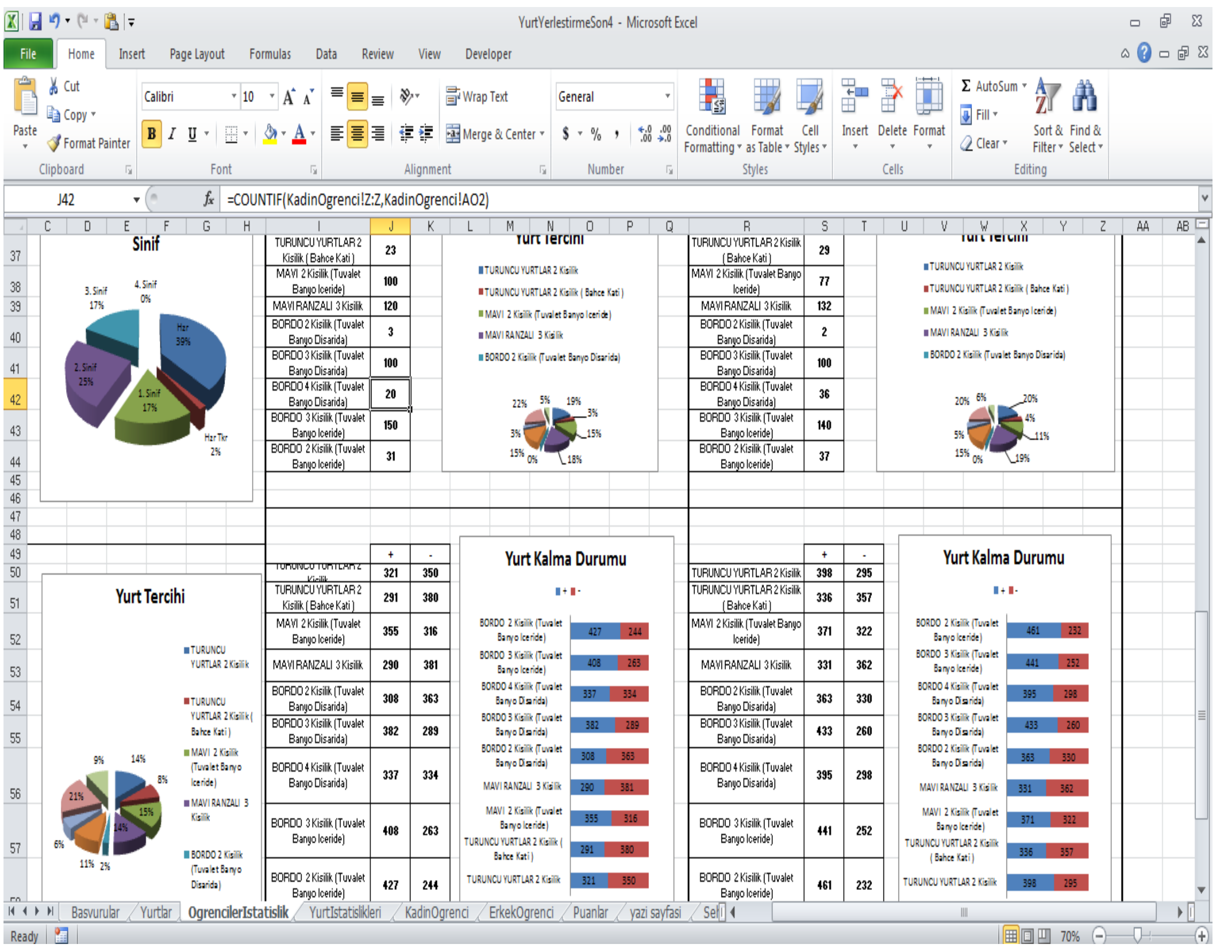


Figure 7.7 Reports 2

The second part of the system is dormitory placement process. For this process, we gave the user two options: Manual application and automation.

Manual application process was demanded by dormitory office because they wanted to place some students in special conditions directly without these students going through the automatic placement process (Figure 7.8). For example, if students have physical disability, these students should be placed into special rooms. Again, we prepared a user-friendly part for this manual part. The user could see the information of this student easily such as room type or roommate wish. The user assigned this student into a room manually, but other rooms were assigned automatically. Of course, if there was roommate preference, first this roommate was assigned to the other bed manually. This manual process was also useful for assigning the rest of the students to free beds after automated placement.

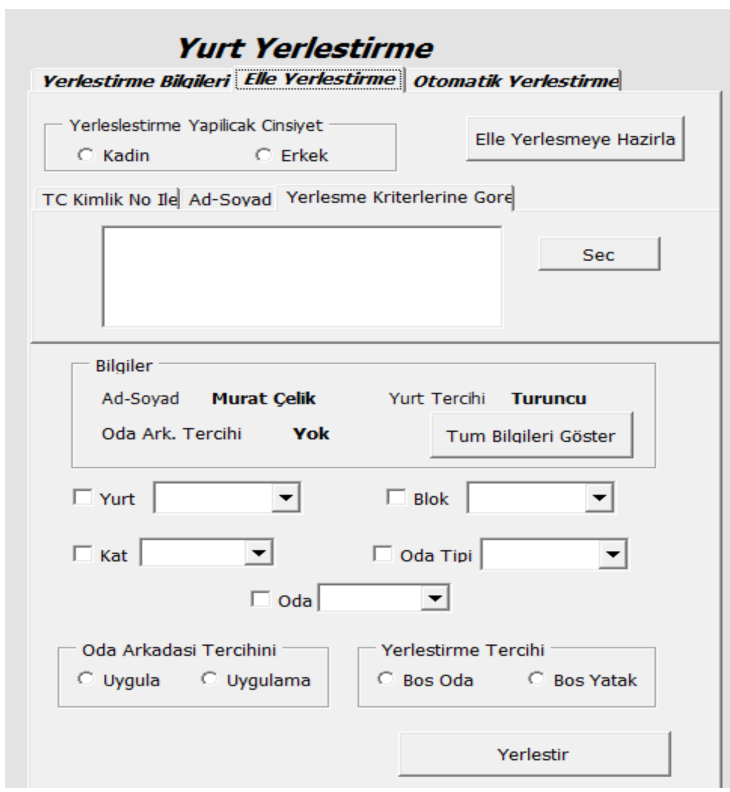


Figure 7.8 Manual placement tool

Again, we created a user-friendly system for the automation part. To click on a button was enough to run the whole system and run the GAMS in the background. We made the placement optimization program in two parts as male and women. We separated into two groups because to place female and male students were two different processes. According to the university policy, they could not stay in same rooms and blocks, so dormitory blocks were already assigned for female and male students before running the system. Additionally, application sets were different also. So, we thought this process as two different assignment problems. This approach also helped to shorten the running time for the solution. One of the best benefits of this separating system was that if we do not like the results for one group, we could fix it without running other group.

We wanted the user to make some decisions on the automation part for the optimization. First decision was to determine the order of importance for room type and roommate preferences on the placement. For 2014-2015 academic year the decision was in favor of the roommate preferences. We prepared a system to ask the user this data because this decision might change from year to year according to the university policy.

The user can also give the maximum running time of the model. Furthermore, the user can specify a gap to the upper bound as a stopping criterion.

Of course, we gave default answers to all these three questions to the system at the beginning. The default answer of roommate question was roommate, default answer of running time was 2 hours, and the default answer of gap question was 0.0 %.

For different runs, we used our default answers to see the results, so we took longer runs than the user enters as running time. To warn the user about this situation, we gave the user a warning when he/she enters a running time.

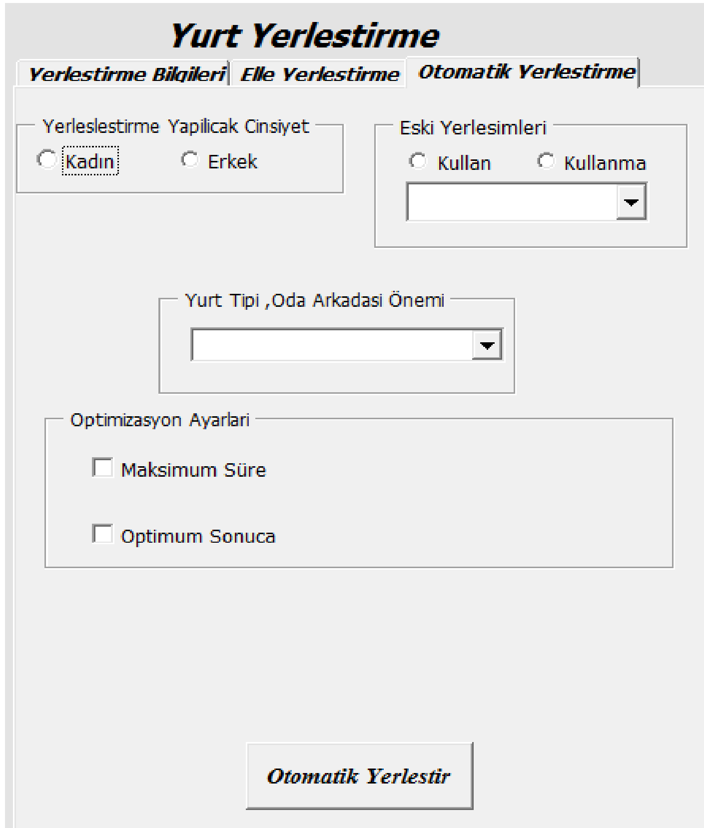


Figure 7.9 DSS solution

The last part of DSS is to view the results with reports (Figure 7.9). First, we presented the general results as the dormitory office wishes (Figure 7.10). Results were presented in a format which FMV Işık University was used to see.

Results of our models and heuristic approach depend on weights. Because of this reason, the user sometimes needs to take more than one run. For these kinds of situations, we gave the user opportunity to save the results in the system (Figure 7.11). Additionally, he/she could choose the report type for each result.

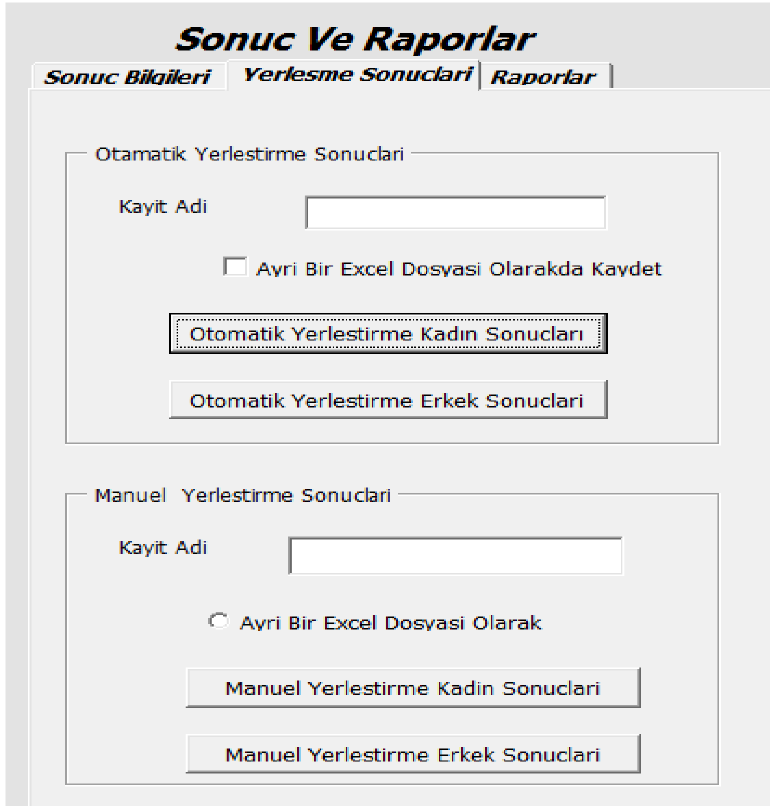


Figure 7.10 Solution interface

We created a system for the user to use the system manually in special times such as during summer schools. In summer schools, the demand for dormitories is not high, so we provided functionality to place students manually, and pull the results from the system. Additionally, the user can save his/her work.

The students who were not placed before with the old placement system were sorted according to the placement criterion with our system. Even if they are not placed again, we would be able to see their sorting. When the students gave up their rights for the placement, we could easily replace these students by looking their sorting.

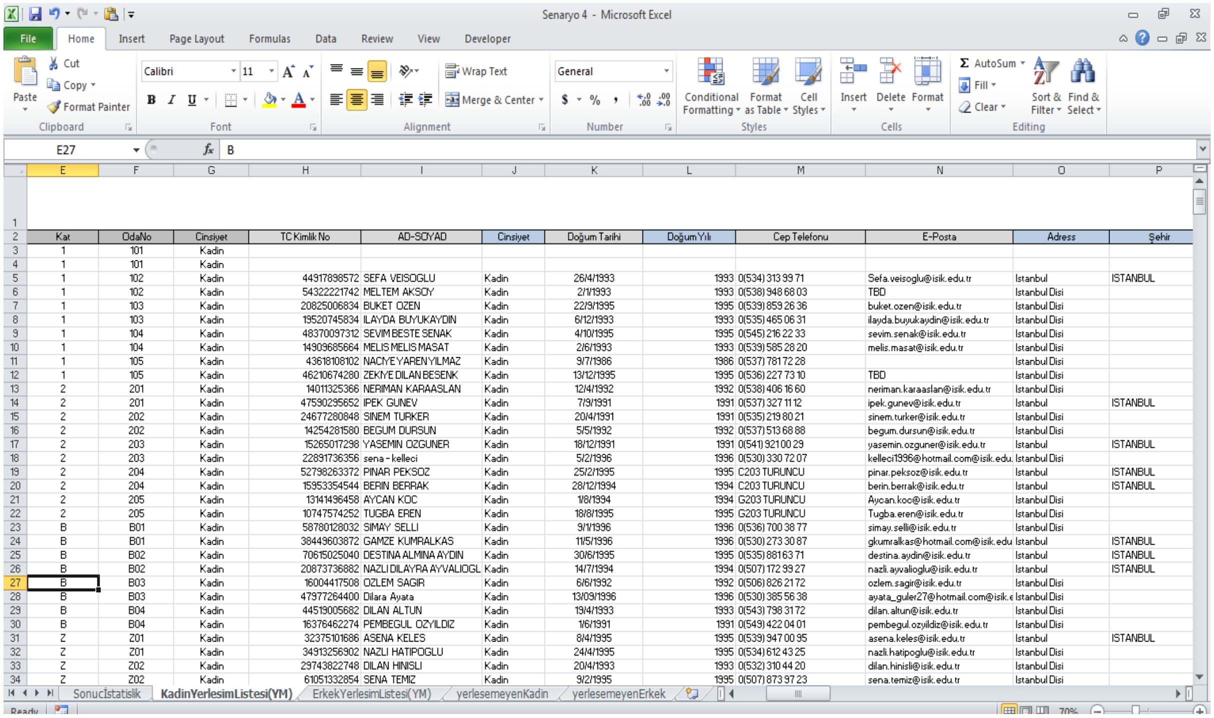


Figure 7.11 Placement results

Like the data input part, we arranged again meetings with the dormitory office, university management and student council for the results part. We prepared reports according to their wishes. After examining the results, we saw that so many answers of some questions like which students were placed to their which preferences, or which targets we have reached at the end of the placement process?

After placement process and give the results, the university management and the dormitory office gave us really positive feedback about our system. Our user-friendly interface and various reports avoided data loss, wrong data input and prejudice about our system.

CHAPTER 8

SCENARIOS AND RESULTS

In this section, we provide results for four different scenarios which result in different room distributions.

In all scenarios, students from Dicstrict-8 (Kadıköy) are not placed. Additionally, the placement rate and general happiness about room preference and roommate preference are really high in all four scenarios. We used 2014-15 academic year application data for these scenarios.

Scenario 1: Existing dormitory allocations are preserved. **No beds** are reserved for future placement or other usages.

Scenario 2: Gender of Orange-D is made female rather than male. **No beds** are reserved for future placement or other usages.

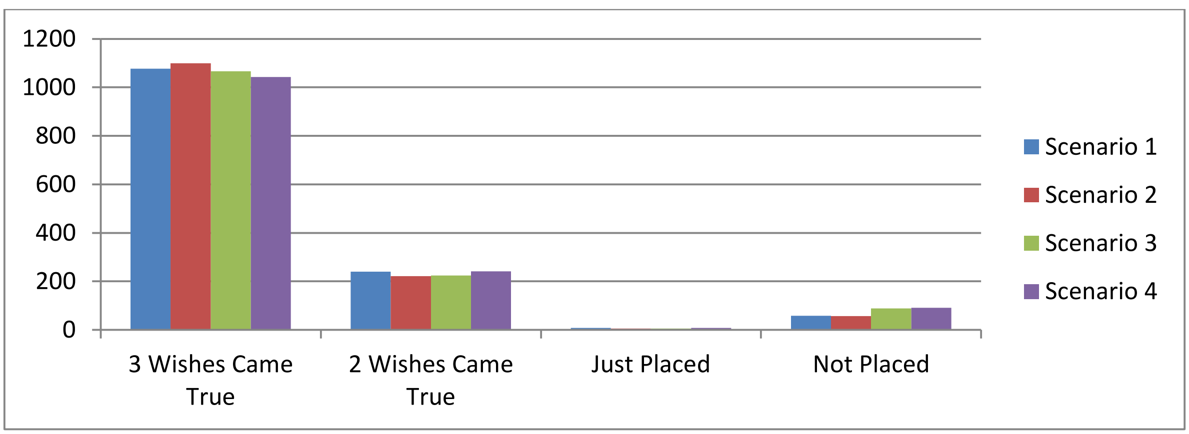
Scenario 3: Existing dormitory allocations are preserved. **One room is reserved** from each room type for future placement or other usages.

Scenario 4: Gender of Orange-D is made female rather than male. **One room is reserved** from each room type for future placement or other usages.

8.1 Results

8.1.1 General Happiness

General happiness is measured by the number of wishes of students which came true. Each student has 3 demands at most; to be placed, room preference and roommate preference. Of course, it is the best case if the student is placed according to his/her room preference with his/her preferred roommate. Graph 8.1 and Table 8.1 show us the general happiness results for each scenario.



Graph 8.1 General happiness results

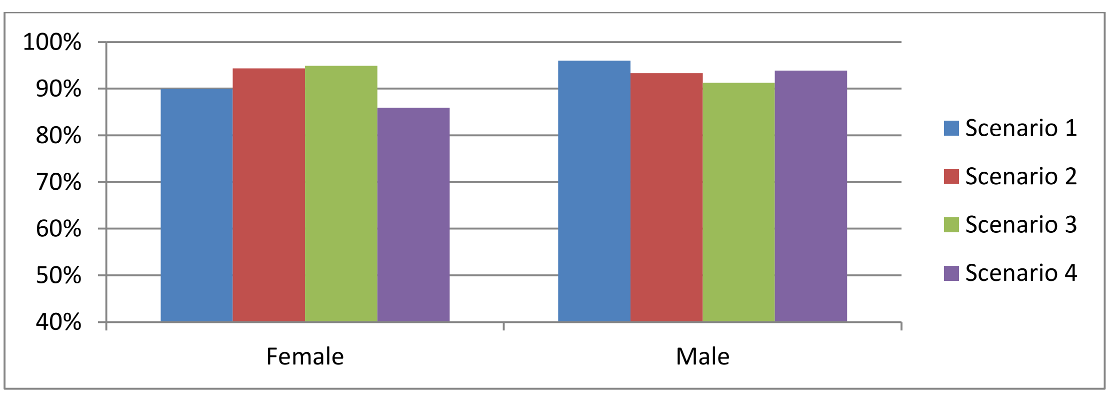
Table 8.1 General happiness table

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
3 wishes came true	1077	1099	1066	1043
2 wishes came true	240	221	224	241
Just Placed	8	6	5	8
Not Placed	58	57	88	91

The placement rate for male students is higher than female students in Scenario 1 and Scenario 3. But in the other two scenarios, this rate is higher for female students rather than male students. The main reason is that changing Orange D Dormitory. The gender of Orange-D is changed from male to female. Additionally, in Scenario 3 and Scenario 4, the accruement rate for wishes decreased because of decreased quota.

8.1.2 Roommate

The placement rate according to roommate preferences is shown on the following chart and table for four different scenarios (Graph 8.2 and Table 8.2).



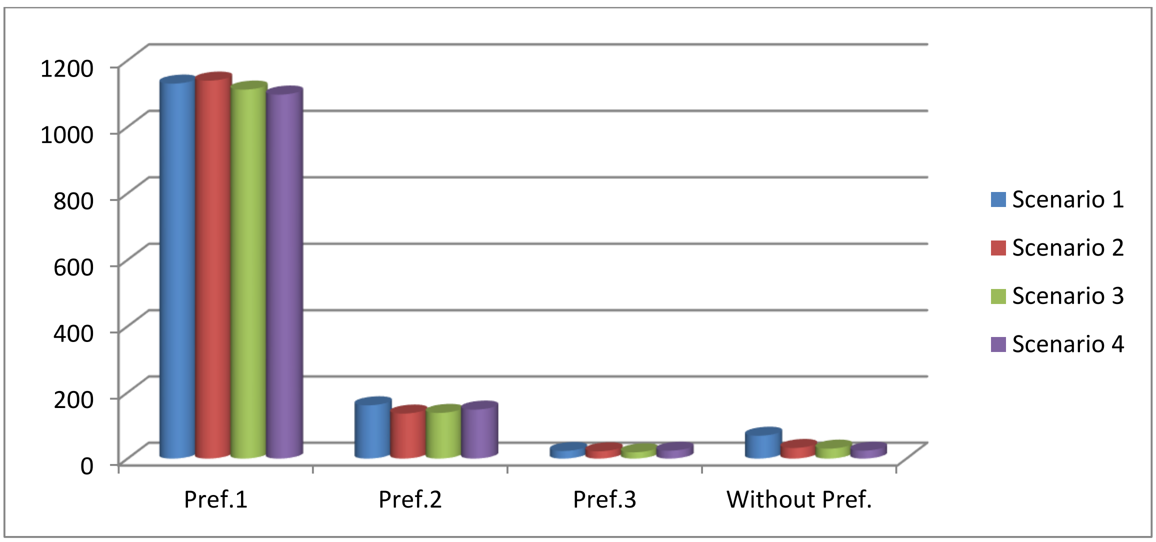
Graph 8.2 Placement rate according to roommate preferences

Table 8.2 Placement rate according to roommate preferences

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Female	**90%**	**94%**	**95%**	**86%**
Male	**96%**	**93%**	**91%**	**94%**

8.1.3 Placement Preference

The placement rate is shown on the following chart and table for four different scenarios (Graph 8.3 and Table 8.3).



Graph 8.3 Placement rate for four different scenarios

8.1.4 Class Preference

The wishes of seniors and students who entered in 2014 came true more than other students as expected also.

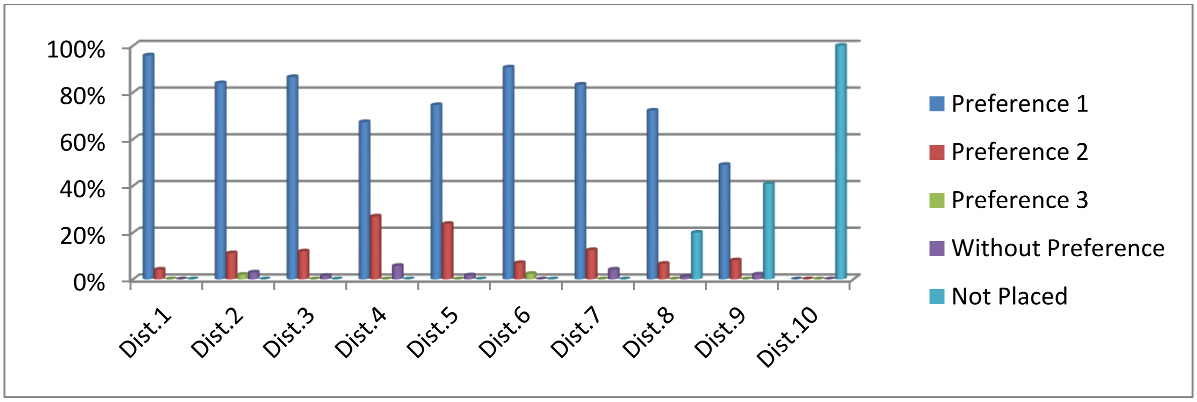
Table 8.3 Placement rate for four different scenarios

Preference	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Preference 1	1130	1139	1112	1097
Preference 2	161	136	138	148
Preference 3	23	22	19	24
Without Preference	69	32	30	24

8.2 Scenarios

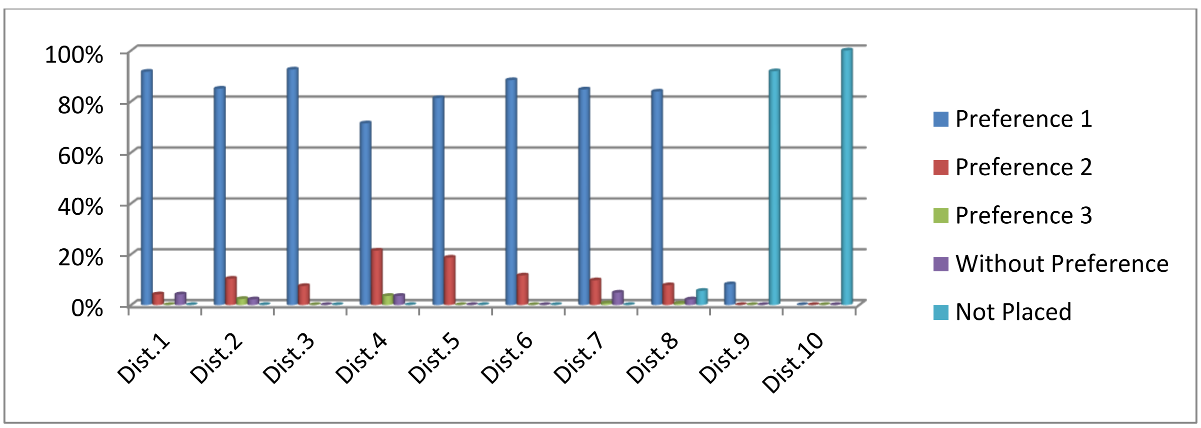
Following graphs show us number of students who are placed to their dormitory preferences according to their home addresses for all scenarios (Graph 8.4, 8.5, 8.6 and 8.7). We expect to see a decrease on the placement rate for the preferences of students when district number increases. However, 2014-15 placement results showed different solutions than our expectations for some district types. The main reason is dormitory type requests of students who live in this region. For example, because the first preferences of students who live in District-6 are for the dormitory types that have been requested less, the placement rate for this district is higher than some of previous districts. On the other hand, because the demanded dormitories for District-4 are preferred before by the students who live in previous districts, their placement rate is lower.

8.2.1 Scenario 1



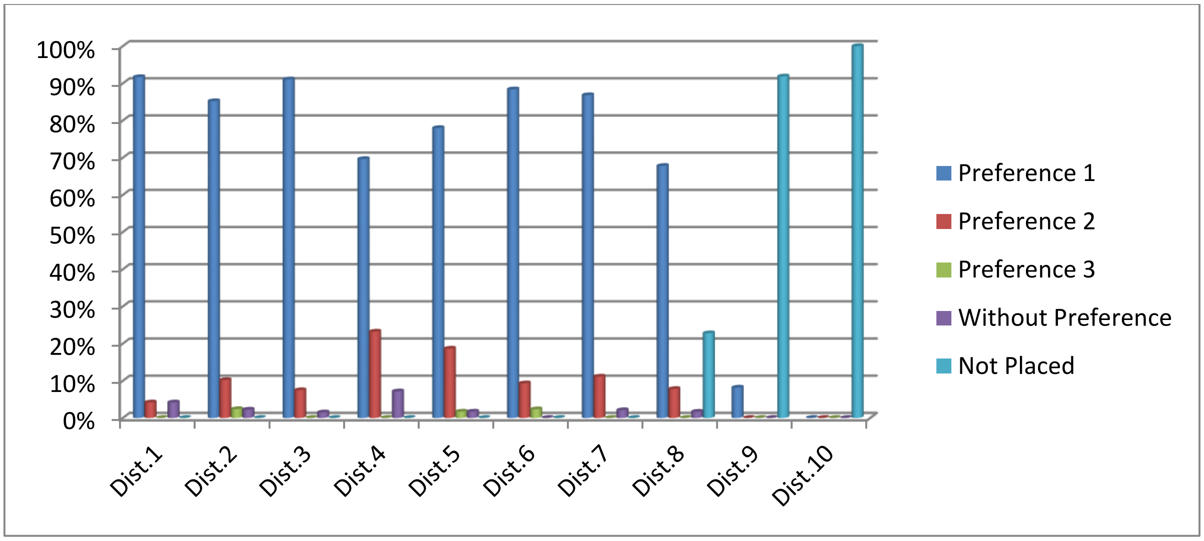
Graph 8.4 Preferences for Scenario 1

8.2.2 Scenario 2



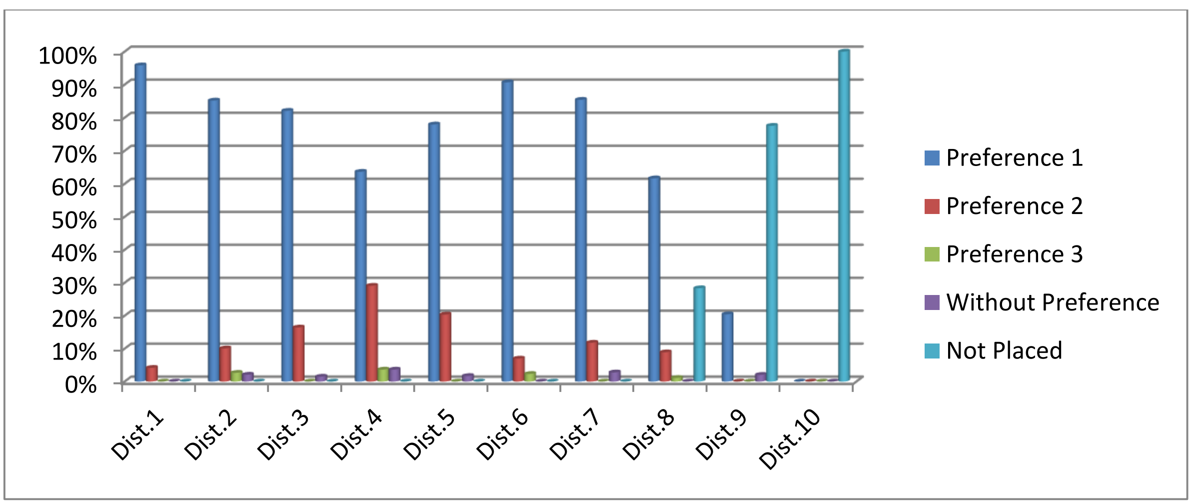
Graph 8.5 Preferences for Scenario 2

8.2.3 Scenario 3



Graph 8.6 Preferences for Scenario 3

8.2.4 Scenario 4



Graph 8.7 Preferences for Scenario 4

CHAPTER 9

CONCLUSIONS

9.1 Summary of the Research

We solved dormitory placement problem in FMV Işık University Şile campus and designed a process from taking the applications to the announcement of the results and getting the feedback.

Our DSS has many reports for the user to make the application and decision process easier. We improved our program according to the requirements of the Dormitory Department. One of the most important things in design phase was to make it as dynamic as possible for possible last minute changes. So, the dormitory office could change the system easily when required.

Experiments with developed mathematical models and heuristic methods we showed that our program works efficiently even if the problem size increases to five times of its current dimension. In addition to this, we decreased the solution time of the problem to hours from weeks with the new data input and decision support system. So the time the dormitory office had to spend for placement decreased significantly.

The DSS was used in 2014-2015 and 2015-2016 academic years. Problems found during these runs were fixed for the next academic year.

9.2 Opportunities for Future Work

The main problem in our study is the combination of Stable Marriage Problem and Stable Roommate Problem. While both problems have been researched extensively, there is no study for the combination of these two problems. Because our problem has small scale, we could solve our problem by mathematical modelling, but for large scale problems, it would be more difficult to solve it in acceptable solution time. Furthermore, in our problem, the difference between the importance of the preference of dormitory type and preference of the roommate is big. Because of that we could use current algorithms separately. On the other hand, if the difference were smaller, applying algorithms separately would keep us away from the real solution. Because of these reasons, it could be useful to study this possibility.

REFERENCES

[1] Gale, D. and Shapley, L.S. (1962). College Admissions and the Stability of Marriage. *The American Mathematical Monthly*, 69.1, 9-15.

[2] Ergin, H. and Sonmez, T. (2006). Games of School Choice under the Boston Mechanism. *Journal of Public Economics*, 90, 215-237.

[3] Irwing, R. W. (1985). An efficient algorithm for the "stable roommates" problem. *Journal of Algorithms*, 6.4, 577-595.

[4] Duan, R. and Pettie, S. (2014). Linear-time approximation for maximum weight matching. *Journal of the ACM*, 61.1.
DOI:http://dx.doi.org/10.1145/2529989

[5] Bertsekas, D. P. (1992). Auction algorithms for network flow problems: A tutorial introduction. *Computational Optimization and Applications*, 1.1, 7-66.

[6] Wright, M. B. (1990). Speeding up the Hungarian algorithm. *Computers and Operations Research*, 17.1, 95-96.

[7] Ahuja, R.K., Magnanti, T.L. and Orlin, J.B. (1993). *Network Flows: Theory, Algorithms and Applications*, (1st ed.) Michigan: Prentice Hall.

[8] Abdulkadiroglu, A. and Sonmez, T. (1999). House Allocation with Existing Tenants. *Journal of Economic Theory*, 88, 233-260.
Available. Retrieved March, 2015, from
http://www.idealibrary.com/

[9] Balinski, M. and Sonmez, T. (1999). A Tale of Two Mechanisms: Student Placement. *Journal of Economic Theory*, 84, 73-94.
Available. Retrieved June, 2015, from
http://www.idealibrary.com/

[10] Firat, M., Hurkens, C.A.J. and Laugier, A. (2014). Stable Multi-still Workforce Assignments. *Annals of Operations Research*, 213, 95-114.

[11] McDermid, E.J. and Manlove, D.F. (2010). Keeping Partners Together: Algorithmic Results for the Hospitals / Residents Problem with Couples. *Journal of Combinatorial Optimization*, 19, 279-303.

[12] Biro, P., Manlove, D.F. and McBride, I. (2014). The Hospitals / Residents Problem with Couples: Complexity and Integer Programming Models. *Experimental Algorithms*, 8504, 10-21.
Available. Retrieved May, 2015, from
http://arxiv.org/abs/1308.4534

[13] Dean, B.C., Goemans, M.X. and Immorlica, N. (2006). The Unsplittable Stable Marriage Problem. *Proceedings of IFIP TC: 4th IFIP (International Federation for Information Processing) International Conference on Theoretical Computer Science*, 209, 65–75.

[14] Biro, P. and Klijn, F. (2011). Matching with Couples: a Multidisciplinary Survey. Institute of Economics, Hungarian Academy of Sciences, Budapest.

[15] Irving, R.W. (1994). Stable Marriage and Indifference. *Discrete Applied Mathematics*, 48, 261-272.

[16] Ehuchi, A., Fujishige, S. and Tamura, A. (2003). A Generalized Gale-Shapley Algorithm for a Discrete-Concave Stable-Marriage Model. *Algorithms and Computation, 14th International Symposium*, Kyoto, Japan.
DOI: 10.1007/978-3-540-24587-2_51

[17] Gusfield, D. and Irving, R.W. (1989). *The Stable Marriage Problem: Structure and Algorithms.* The MIT Press, Cambridge, MA.

[18] Dubins, L.E. and Freedman, D.A. (1981). Machiavelli and the Gale-Shapley Algorithm. *The American Mathematical Monthly*, 88.7, 485-494.
Available. Retrieved September, 2015, from
http://www.jstor.org/stable/2321753

[19] Perach, N., Polak, J. and Rothblum, U.G. (2008). A Stable Matching Model with an Entrance Criterion Applied to the Assignment of Students to Dormitories at the Technion. *International Journal of Game Theory*, 36, 519-535.
DOI 10.1007/s00182-007-0083-4

[20] Crandall, R. H. (1969). A Constrained Choice Model for Student Housing. *Management Science*, 16.2, 112-120.
Available. Retrieved March, 2014, from
http://www.jstor.org/stable/2628494

Curriculum Vitae

Tonguç Yavuz was born on the 20th of April, 1988, in Yalova. He received his B.S. degree in Industrial Engineering from Işık University in 2013. He completed his M.S. degree in Industrial Engineering and Operations Research at Işık University in 2015. During his graduate studies, he served as a teaching assistant for "Operations Research Modelling Applications", "Scheduling" and "Operations Research III" courses. He also successfully designed and implemented decision support systems for Üsküdar University and Young Talent Reconnaissance Project for the FunTheMental Incorporated Company using Microsoft Office Excel VBA. The first project was about generating university course timetables, and the second project aimed to understand who among young children is talented in which sports branch by considering their physical, psychological and nourishment features. His research interests include operations research, decision support systems, scheduling, mathematical modeling and network models.