**CHAPTER ONE**

**INTRODUCTION**

* 1. **Background to the Study**

Space allocation problem (SAP) is a domain specific combinatorial optimisation problem (COP) that has attracted attention among researchers (Bai 2005; Landa 2003). The problem is similar to scheduling problem such as academic timetabling (Adewumi et. al. 2003 & 2005) with its associated resource efficiency issues. SAP, in a higher institution context, is defined as the allocation of entities (staff, students, laboratories, lecture rooms, etc.) to available rooms in order to satisfy both hard and soft constraints (Landa 2003). It finds relevance in various fields such as storage space allocation on disk and office space allocation. More importantly, the distribution of the available space among staff, students and other demanding entities is a dynamic and continual exercise. In most real life cases with large instances, the use of proven optimization technique becomes inevitable. Hostel space allocation problem (HSAP) arises from the need to allocate available bed spaces distributed around various halls of residence (or hostels) among large number of demanding and eligible categories of students based on a set of given (and sometimes conflicting) constraints. The number of applicants and constraints for each category varies. The objective is to ensure optimal space utilization that satisfies given constraints. The complexity of the problem emanates from the limited number of bed spaces available, increasing demand from students and the conflicting nature of some constraints imposed on the allocation process. The best practice in most Nigerian universities is to use database or spreadsheet applications without recourse to any algorithm that determines an optimal allocation. Burke and Varley (1998a and b) suggested the application of heuristics to SAP. Exact methods have been applied to various (and in most cases smaller) instances of real-life SAP. These include the use of mixed-integer goal programming (Ritzman et al. 1980), linear programming (Benjamin et. al. 1992), and integer goal programming (Giannikos et al., 1995). Like other NP-Hard combinatorial optimization problems (COPs), exact algorithms for SAP have exponential time complexity (Landa 2003). For large instances of SAP and faster results, heuristics and their variants have been applied lately. Bai (2005) investigated the application of different metaheuristics to real-world shelf SAP that arise due to conflict of limited shelf space availability and the large number of products to be displayed. Landa (2003) investigated the application of metaheuristics to real-life instances of office SAP.

* 1. **Statement of the Problem**

The growing number of students in higher institutions all over the world has posed a lot of accommodation problem on the part of students and school management. Students at the beginning of each session waste half of the semester looking for accommodation. The few hostels that exist in the universities are not properly managed. Statistics of the number of rooms required to match the growing number of students are far-fetched. Most often, students pay for hostel fee and end up not getting one due to lack of bed space. Hostel administrators cannot give accurate information of the occupancy of a particular room. These and many more form the statement of the problem that necessitated this research work.

* 1. **Aim and Objectives of the Study**

The aim of this study is to develop an optimal hostel space allocation system using genetic algorithm. The objectives are:

1. Develop a novel model and algorithm for efficiently allocating students to hostels using genetic algorithm.
2. Implement a web based system for dynamic allocation of students to hostels.
3. To design and develop a central database system that would serve as hostel database, which will contain information on all the available rooms in the hostels.
   1. **Significance of the Study**

The importance of this study includes:

1. Facilitate timely allocation of students to hostel rooms.
2. Check the hostel occupancy at any time for proper information management.
3. Enable management to plan on improving hostel living condition.
4. Have first-hand information on the statistics of students in the hostels.
   1. **Scope of the Study**

The research work will cover among other things:

1. Hostel allocation
2. Hostel rooms registration
3. Student reports.
   1. **Limitation of the Study**

Some of the constraints encountered during this project design include the following:

1. Financial Constraints: The design was achieved but not without some financial involvements. One had to pay for the computer time. Also the typing and planning of the work has its own financial involvements.

2. High programming Technique: The programming aspect of this project posed a lot of problematic bugs that took us some days to solve. Problems such database connections using javaScript and mongodb database posed a lot of challenges.

3. Few Literature Sources: The topic though seems to be a common term; it is not a popular topic to surf from the Internet. It had fewer literature sources.

* 1. **Definition of terms**

1. Student: a person formally engaged in learning, especially one enrolled in a school or college.
2. Hostel: A hostel is essentially a form of accommodation that offers reasonably priced, shared accommodation to travellers in either private or dormitory rooms.
3. Decision Support System: Decision support system is a specific type of computerized information system that supports decision-making activities.
4. A management information system (MIS): is a system that provides information needed to manage organizations effectively.
5. Allocation: the action or process of allocating or sharing out something.
6. Optimization: the action of making the best or most effective use of a situation or resource.
7. Combinatorial: of or relating to the arrangement of, operation on, and selection of discrete mathematical elements belonging to finite sets or making up geometric configurations
8. Constraints: a limitation or restriction.
9. Genetic Algorithm: The genetic algorithm is a method for solving both constrained and unconstrained optimization problems that is based on natural selection, the process that drives biological evolution

**CHAPTER TWO**

**LITERATURE REVIEW**

**2.1    Historical Development of Hostels**

Davies et al (2008), defined hostel as an establishment which provides cheap lodging and food for travelers, a place of residence for students. It is also defined as a budget- oriented, shared-room ("dormitory") accommodation that accepts individual travelers or groups for short-term stays, and that provides common areas and communal facilities (Wikipedia Encyclopedia). It further stated that for it to be considered a hostel, the property must provide short-term, shared (dormitory-style) accommodation for individual travelers, though many hostels also provide private rooms.

Hostel has become an opportunity to develop and improve the quality of education in most academic institutions. The desire to provide spaces for students that allow active interaction, comfort and convenience, and opportunities for socialization is foremost in university and college planning (Perkin & Will, 2001). They also assert that all these need to be considered along with the pragmatic spatial needs of furnishings, integration of efficient effective mechanical, electrical, plumbing and safety requirements when planning for hostel buildings.

The first youth hostel was founded by Richard Schirrmann around 1909 (www.hostelscentral.com/hostels-article-6.html). He was a German teacher who organized trips and visits with his students. During one of these excursions, a sudden rainstorm forced his group to seek shelter in an empty school. It was then that Schirrmann had the idea of using schools that were empty during holidays as guest houses for young people that were travelling in groups in the countryside (www.hostelscentral.com/hostels- article-6.html). The concept of student "youth hostel” was born. The movement flourished in Germany. Permanent hostels were established by gifts of hiking and recreation clubs, wealthy patrons and local communities. By 1932, Germany had more than 2000 youth hostels recording more than 4.5 million overnights annually (http://hiusa.org/about\_us/history). In the meantime, Switzerland, Poland, the Netherlands, Norway, Denmark, the British Isles, Ireland, France and Belgium had joined the movement and added another 600 hostels in Europe (http://hiusa.org/about\_us/history). With national hosteling associations spreading across Europe, in 1932 the first international meeting was held in Amsterdam to develop common standards. The International Youth Hostel Federation was formed (http://hiusa.org/about\_us/history). Americans Isabel and Monroe Smith attended the second international meeting in 1933. Shortly afterwards, they opened the first U.S. youth hostel in Northfield, Massachusetts in 1934 (http://hiusa.org/about\_us/history). Pre- war European political currents overshadowed much of the international movement in the late 1930's. Hostels were closed, and even appropriated by governments for military purposes. The operations of many European hosteling associations were suspended. During the war, the growth of the hosteling movement stalled, although parts of the European youth hostel system still continued to operate, as well as a small network of hostels in the US and Canada.

The end of the war brought a time of rebuilding and reflection worldwide. Groups of American youth went to Europe to help rebuild hostels. International youth travel, while still nascent, was embraced by governments as a way to encourage understanding, and avoid future conflict. The International Youth Hostel Federation grew, as the German youth hostel system was re-established and new hostel associations were formed in Africa, Asia, Australia and South America (http://hiusa.org/about\_us/history). In recent times, the concept of hostel has broadened, and now providing cheap but comfortable accommodation and a unique atmosphere that cannot be found in hotels. Worldwide, there about 4000 hostels recording over 33 million overnights in more than 80 countries. (www.hiusa.org/about\_us/history).

**2.2    Student Hostels in Nigerian Tertiary Institutions**

The Nigerian higher educational institution was established with the aim of giving a student a very sound and qualitative education, so as to be able to function effectively in any environment in which they may find themselves, so as to become more productive, self- fulfilling and attain self-actualization (Saint, et al, 2004). This is because in Nigeria, students are the single most important stakeholders in the university/college system. Similarly, student’s accommodation is among the most important facilities that should be provided in a typical Nigeria university campus.

Living in student’s residence on campuses, has been identified as one of the interesting experiences in the life of a university student. This is because it offers them the opportunity to interact amongst their colleagues from faculties other than their own in addition to the unique opportunity for night discussions and social interactions which when put together will help in shaping the social life of the student’s social life, appreciation of their roles and responsibilities in the community and society at large (Esenwa, 2003).  
In the beginning Nigerian universities were established with the intention of providing comfortable hostel accommodation for all students on campus. Up till the early 1970’s there was no problem of student’s accommodation in university campuses (Esenwa, 2003).

**2.3    Information Systems**

An information system consists of three components human, task and application system. In this view, information is defined in terms of the three levels of semiotics.

Data which can be automatically processed by the application system corresponds to the syntax-level. In the context of an individual who interprets the data they become information, which correspond to the semantic-level. Information becomes knowledge when an individual knows and evaluates the information example for a specific. This corresponds to the pragmatic-level. From Wikipedia, In general systems theory, and information system is a system, automated or manual, that comprises people, machines, and or methods organized to collect, process, transmit, and disseminate data that represent user information. According to John Cooper, Jane Sullivan, (2003), failure of the information systems can cause a major loss of service, and so their dependability is a major concern. Current facets of dependability, such as reliability and availability, do not address the needs of critical information systems adequately because they do not include the notion of degraded service as an explicit requirement. Some background material about critical information systems is helpful in understanding the need for a precise notion of survivability and how it differs from other notions of dependability.

According to Zachman J.A., (1987), the subject of information systems architecture is beginning to receive considerable attention. The increased scope of design and levels of complexity of information systems implementations are forcing the use of some logical construct (or architecture) for defining and controlling the interfaces and the integration of all of the components of the system. On the assumption that an understanding of information systems architecture is important to the development of a disciplined approach, the question that naturally arises is "What, in fact, is information systems architecture?" Unfortunately, among the proponents of information systems architecture, there seems to be little consistency in concepts or in specifications of "architecture," to the extent that the words "information systems architecture" are already losing their meaning! Furthermore, it probably is not reasonable to expect reconciliation or commonality of definition to emerge from the professional data processing community itself.

According to Couger John, (1973), the information systems development process is viewed as consisting of analysis, design, and implementation phases, prior to the operation phase. These phases do not ordinarily take place strictly in the order given but rather exist together in a continuing pattern of interaction. The development of information systems then consists of an iterated process of information analysis, system design, and implementation. This "system life cycle," it has been pointed out, applies to other kinds of development effort as well.

**2.4    Genetic Algorithm**

Idea of evolutionary computing was introduced in the 1960s by I. Rechenberg in his work "Evolution strategies". His idea was then developed by other researchers. Genetic Algorithms (GAs) were invented by John Holland and developed by him and his students and colleagues. This lead to Holland's book "Adaption in Natural and Artificial Systems" published in 1975. In 1992 John Koza used genetic algorithm to evolve programs to perform certain tasks. He called his method "genetic programming" (GP). LISP programs were used, because programs in this language can expressed in the form of a "parse tree", which is the object the GA works on.

All living organisms consist of cells. In each cell there is the same set of chromosomes. Chromosomes are strings of DNA and serves as a model for the whole organism. A chromosome consist of genes, blocks of DNA. Each gene encodes a particular protein. Basically can be said, that each gene encodes a trait, for example color of eyes. Possible settings for a trait (e.g. blue, brown) are called alleles. Each gene has its own position in the chromosome. This position is called locus.  
Complete set of genetic material (all chromosomes) is called genome. Particular set of genes in genome is called genotype. The genotype is with later development after birth base for the organism's phenotype, its physical and mental characteristics, such as eye color, intelligence etc.   
During reproduction, first occurs recombination (or crossover). Genes from parent form in some way the whole new chromosome. The new created offspring can then be mutated. Mutation means, that the elements of DNA are a bit changed. This changes are mainly caused by errors in copying genes from parents.

In a genetic algorithm, a population of candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties (its chromosomes or genotype) which can be mutated and altered; traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible. (Whitley, 2015)

The evolution usually starts from a population of randomly generated individuals, and is an iterative process, with the population in each iteration called a generation. In each generation, the fitness of every individual in the population is evaluated; the fitness is usually the value of the objective function in the optimization problem being solved. The more fit individuals are stochastically selected from the current population, and each individual's genome is modified (recombined and possibly randomly mutated) to form a new generation. The new generation of candidate solutions is then used in the next iteration of the algorithm. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population.

A typical genetic algorithm requires:

1. a genetic representation of the solution domain,
2. a fitness function to evaluate the solution domain.

A standard representation of each candidate solution is as an [array of bit](https://en.wikipedia.org/wiki/Bit_array)s (Whitley, 2015). Arrays of other types and structures can be used in essentially the same way. The main property that makes these genetic representations convenient is that their parts are easily aligned due to their fixed size, which facilitates simple crossover operations. Variable length representations may also be used, but crossover implementation is more complex in this case. Tree-like representations are explored in genetic programming and graph-form representations are explored in evolutionary programming; a mix of both linear chromosomes and trees is explored in gene expression programming.

Once the genetic representation and the fitness function are defined, a GA proceeds to initialize a population of solutions and then to improve it through repetitive application of the mutation, crossover, inversion and selection operators.

**2.5    Hostel Space Allocation Problem**

Hostel space is one of the most important resources that influence the choice of higher institutions. Availability and efficient management of hostel space will not only aid concentration and academic performance of students but also assist in maximizing the institution’s throughput in terms of graduation rate. Hostel space allocation and management are therefore critical issues in university administration. Most first generation (in terms of relative date of establishment) universities in developing countries have on-campus residence for students that were built years ago when the average number admission intakes were small compared to the current influx. The upsurge in student population is such that majority of the students reside off-campus with the attendant adverse effects on performance and throughput especially for the brilliant ones. In Nigeria, most of the higher institutions have hostel space enough to accommodate less than twenty percent of the student population (Futminna, 2009). In South Africa, it was recently reported nationally (M&G, 2009) that residence accommodation for students has reached crisis levels with the attendance effect on success and graduation rates. A forum of some tertiary institutions in the country noted that the high rate of student failures and increasing first-year dropout rates are worsened by the lack of student accommodation (M&G, 2009). The recent upsurge coupled with the declining rate of fund availability therefore makes it necessary for university administration to reconsider many issues including space planning, most especially hostel space for students. Meanwhile, given the fact that provision of new and sufficient hostels to meet the current demand in developing countries is a long-term developmental goal; the emphasis should then be on finding an optimal way of utilizing and allocating the available hostels. The allocation of available hostel space can thus be seen as a competitive process with strict guidelines set by various institutions to steer the process. Continual increased in the number of eligible applicants for hostel space with little or no subsequent increase in facility calls for an efficient optimization approach. HSAP can therefore be considered as a decision problem that aim at finding the best possible allocation of scarce resources (bed space within hostels) among many competitive ‘customers’ (eligible students) under given hard and soft constraints. It can be considered as an extension of the knapsack (Pisinger, 1999) or bin parking problems (Federgreun, 1997).

**2.5.1 Basic Terminology**

We provide some basic definitions and notations related to the HSAP.

1. Bed Space: A space created in the hostel to accommodate a single student.

2. Hall (Hostel) (H): A building meant to accommodate a number of students located at various zones within the campus. Each hall has a pre-defined capacity in terms of bed space and consists of set of rooms located at various floors levels and blocks (also referred to as wings). The duo terms of hall and hostel will be used interchangeably in this paper. A general assumption is that halls are located at different part of the campus vicinity.

3. Capacity: the maximum number of students that can be accommodated in a given entity (e.g. hall, floor, room). This is calculated as the total number of bed spaces in that entity.

4. Category (C): A grouping into which eligible applicants are divided for hostel allocation purpose. Each category has varying number of students with varying degree of allocation priority.

5. Allocation (A): The assignment of students in a given category into an allocation entity (hall, block, floor). For example, at the hall level, aij ∈ A refers to the number of students in category i allocated to hall j. At the Floor/block level, it refers to the number of students allocated to the floor/block.

6. Room Capacity (R): Total number of bed spaces in a given room within a given hall.

7. Block Capacity (B): Summation of R for a given block in a given hall.

**2.6    Related Works**

Iraba (2009) developed the Student Residence Management System to facilitate application for accommodation online and to help the staff to manage the different residence activities such as controlling booking, payments and room allocation. The Student Residence Management System was able to notify and confirm all room allocations. Room allocation confirmations were sent by email to students who were given accommodation. The web-based system was able to address problems encountered by users within the current residence administration system. The proposed system found optimal matching for user requirements, and management of residences. System implementation was carried out using PHP, MySQL, PHPMyAdmin and Apache, which are the open source applications. Asmahani (2007) developed the College Student Allocation System. Here are the traditional methods in firstly students must get the college application forms from the college officer. Next students must fill up the forms. The due date for application is fixed. They must attach together their passport size photo, and their parents' monthly salary slips. The HEP officer's faces difficulties to process the application forms. The process of key in the data takes a longer tune. The objectives of this research was to identify all the requirement specification of College Student Allocation System (CSAS), to model the design of College Student Allocation System (CSAS) based on requirement specification and to develop the prototype of the College Student Allocation System adopting web application framework that is Mamboserver. This research project is about developing College Student Allocation System using Wampserver and web application framework in UiTM. This research is an improvement of the college student allocation system to make it more effective in college management the design of College Student Allocation System (CSAS) based on requirement specification. The limitation of this project is focusing on the colleges that are situated nearby the faculties. Anjorin (2005) developed Online Hotel Reservation “case study of Solton international hotel and resort”. The problem encountered in the existing system was that thousands of students file (registration) are processed manually. In a situation where a student registration file is missing and the officer in charge is making a sequential search through the file to get information about the student’s. The problem becomes complex if the information is not in the specified location and this leads to searching through archives when the necessary documents are available, it often takes a long time to get them into the required format. The approach administered to curbing the problem of the existing system (manual system) was to design a website for Solton international hotel and resort, so as to reduce the cost of transportation by prospective guests in the existing system; to provide a communication medium that can serve in any place including the bedroom and to ensure accuracy. Help participants to understand technology’s current potential and impact on their businesses. The system implementation was carried out using PHP, MySQL, and Apache Server. Akanfe (2005) developed an Online Hotel Reservation (case study of Tamarin hotel), the problem in the previous system encountered involves the manual moving of files, in this case, a large number of guests’ data or information can easily be handled. The proposed system was developed so as to give useful, accurate, timely and well-formatted information about guests. The system implementation was carried out using MySQL for the database for the day-to-day activities of booking rooms, PHP, HTML and Apache server was used as the web server. This would enable them to automate their entire business processes, by allowing them to run their businesses dynamically or interactively and rapid decision making. Peter (2012) developed a web-based classroom allocation system. Often, a manual process was used in the allocation of classrooms and keeping track of the classroom. A better way to consistently and effectively allocate classrooms is to use a computer-assisted system that will keep track of all classrooms on campus along with specific details about those rooms that can automatically suggest efficient pairings with the courses offered for a given semester. The efficiency was being judged based on many factors, most importantly being that the size of each room is used effectively. The new system accommodates the most important factors mentioned before. Other factors such as making sure that each course has a room with the required resources for that class and that the rooms should conform to departmental preferences are also considered. Meetings were held weekly with the stakeholders to cover the current progress as well as introducing new requirements. Due to the Service oriented architecture of the system it made for easy segmentation of the system and an easy way to show where progress has been made. The implementation was carried out using MySQL as a backend, Amahi Server to run MySQL and the built in Glassfish web server that comes with NetBeans which were ran on Windows 7 Professional system.

**CHAPTER THREE**

**ANALYSIS AND DESIGN**

**3.1    Analysis of the Existing System**

The existing system of hostel allocation is currently a web based platform. This platform enables the student affairs department to be able to ascertain the number of students available for hostel accommodation and also determine the number of hostels and rooms available. The allocation process is carried out on a first come, first serve basis and is therefore using a brute force approach in its allocation process. The distribution of students to hostels is usually carried out for first and second year students of the universities due to limited hostel facilities available. However, it has been observed that this allocation process is suboptimal. There are cases where some students do not have hostel allocations and squat with those that have or go ahead and take up accommodation off campus. It has also been observed that there are instances where students have been allocated hostels but would prefer a cozier accommodation off campus due to the poor state of facilities within the hostels or due to the personal taste of the students.

**3.1.1    Advantages of the Existing System**

The current system of student hostel allocation do have some advantages and they are:

1. The system does the job of allocating students to hostels based on availability.
2. The system ensures that each room is filled first before allocation to another room is started.
3. The system keeps track of the number of hostels available for both category of students (male and female) and also keeps track of the number of hostels available for both.
4. It allows the student affairs division to have a proper statistics of student allocation which aids in resource distribution to these hostel facilities.

**3.1.2    Disadvantages of the Existing System**

The weakness of the existing system includes:

1. In computing terms, the allocation process is of the order of O(n), which implies that the if we have say 1000 students, it takes us 1000 computational time to fill up the rooms. Each room in an hostel can be modelled as an array. Just as an array has better approaches to reduce the computational time, therefore the allocation process can be done optimally using genetic algorithm as a the focal point.

**3.2    Analysis of the Proposed System**

The proposed system for student hostel allocation utilizes Genetic Algorithm to optimize the allocation process and also reduce the computational time it takes to allocate students to hostels. Allocations of male and female students into hostels are done in a mutually exclusive manner as undergraduate hostels are delineated based on gender. From the dataset available, there are ten on-campus residences with six designated for male and four for females respectively. Usually, residences are built as multi-story structures (with the exception of two hostel) each with varying numbers of floors that are further divided into blocks (otherwise call wings). Rooms are located on each wing per floor with each having one or more beds depending on the number of students it is designed to accommodate. Usually, due to the shortage of space, most rooms are designed to take more than one student and students on each wing have access to common facilities such as toilets and baths. The university, through the office of students’ affairs, sets the criteria that make a student eligible for a bed space and each eligible student is entitled to only one bed space. The eligibility criteria may vary and is manually checked by staff.

1. Final Year Students (Fy): Those in the last year of study

2. Scholars (Sc): Students with cumulative grade point averages that are in the first class range.

3. Foreign Students (Fo): whose nationality and residence is not Nigeria.

4. Health Students (Ht): Physically challenged students.

5. Fresher (Fr): First year and direct-entry students.

6. Sports students (Sp): Male and Female students who participate in sporting activities at the university.

7. Discretionary (Ds): Students considered based on special requests

8. Others (Ot): All other students requiring accommodation (in various years of study).

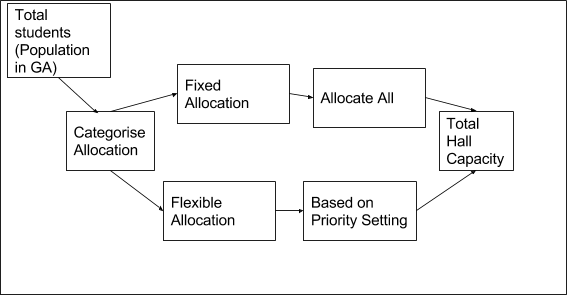
Each category of students has peculiar characteristics and requirements which can be factored in as constraints into the allocation process. For example, disabled (health) students cannot be given allocation on the top floor in any residence since none of the hostels is built with escalators for ease of movement for them. Moreover, since the space available is limited, some of the categories are prioritized based on pre-set administrative and/or other considerations. This serves as a major hard constraint during category allocation. Other administrative considerations that serve as either hard or soft constraints include:

1. First year students must be allocated to a floor that will afford them less distractions (soft).
2. Health students must be accommodated in hostels close to the medical centers and on the lowest floor for easy access (hard);
3. Sport students must be accommodated close to sports facilities due to practice (hard);
4. All First year, Foreign and Health students should be accommodated (hard).

**3.3    Methodology**

The methodology adopted for the development of this system is the object oriented analysis and design methodology. In this system we shall consider several modules that would be plugged together to give the system its full functionality. System design specifies how the system will accomplish the set objectives. This stage comprises design activities that produce system specifications satisfying all the functional requirements. Use case diagrams are adopted to model various activities and scenarios in the proposed system.

**3.4    System Design**

The design of the proposed system is shown in figure 3.1. The students are categorized and for first year students, health students and foreign students are categorized as fixed allocations while others are categorized as flexible allocations. Then for fixed allocations, we allocate all of them while for the flexible allocations, we utilize a priority based allocation to the students. The total number of students are the population in the genetic algorithm system and as the system evolves, mutation sets in and we can compute the fitness function from there.

**Fig 3.1 Design of proposed system**

**3.5    Use Case Diagram**

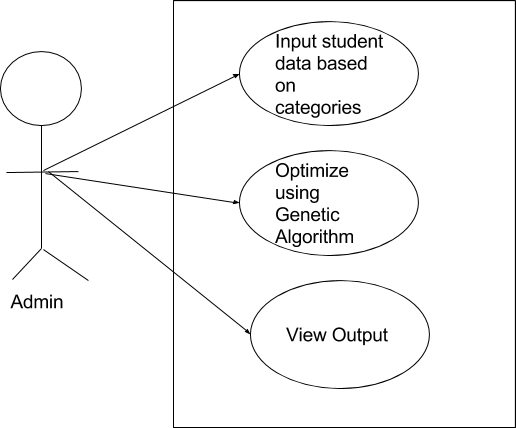
The use case diagram of the proposed system is shown in figure 3.2. It shows the process the user takes to allocate hostels to students. The admin inputs students databased on the categorized earlier mentioned. With the click of a button the admin clicks on optimize using GA and the genetic algorithms performs the function of allocating the students to the hostels by evolving the population of students.

Fig 3.2 Use case Diagram for Student Allocation

**3.6 Software Requirements**

1. Operating system- Windows and mobile operating system is used as the operating system as it is stable and supports more features and is more user friendly.
2. Database MYSQL-MYSQL is used as database as it easy to maintain and retrieve records by simple queries which are in English language which are easy to understand and easy to write.
3. Development tools and Programming language- HTML and is used to write the whole code and develop webpages with cascading style sheet, bootstrap and java script for manipulating the document object model(DOM) and hypertext pre-processor (PHP) for sever side scripting.

**3.6.1 Software tools used**

The whole Project is divided in two parts the front end and the back end.

**FRONT END:** The front end is designed using of HTML, Bootstrap, CSS, Java script

1. **HTML**- HTML or Hyper Text Mark-up Language is the main mark-up language for creating web pages and other information that can be displayed in a web browser.HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like <html>), within the web page content. The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behaviour of HTML web pages.
2. **CSS**- Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a mark-up language. While most often used to style web pages and interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation.CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colours, and fonts. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed.
3. **JAVA SCRIPT**- JavaScript (JS) is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It is also being used in server-side programming, game development and the creation of desktop and mobile applications. JavaScript is a prototype-based scripting language with dynamic typing and has first- class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the self and Scheme programming languages. It is a metaparadigm language, supporting object-oriented, imperative, and functional programming styles.
4. **Bootstrap is a touch-optimized HTML5 UI framework designed to make responsive web sites and apps that are accessible on all screen sizes including smartphone, tablet and desktop devices.**

**BACK END-** The back end is designed using MySQL which is used to design the databases and PHP which is a scripting language for server side.

1. **MYSQL**- MySQL ("My S-Q-L", officially, but also called "My Sequel") is (as of July 2013) the world's second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius daughter, My. The SQL phrase stands for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for- profit firm, the Swedish company MySQLAB, now owned by Oracle Corporation. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL. For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, Drupal and other software. MySQL is also used in many high-profile, large-scale websites, including Wikipedia, Google (though not for searches), Facebook, Twitter, Flickr, and YouTube.
2. **PHP**- PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP is now installed on more than 244 million websites and 2.1 million web servers. Originally created by Rasmus Lerdorf in 1995, the reference implementation of PHP is now produced by The PHP Group. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Pre-processor, a recursive backronym.PHP code is interpreted by a web server with a PHP processor module, which generates the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications. PHP is free software released under the PHP License. PHP can be deployed on most web servers and also as a standalone shell on almost every operating system and platform, free of charge.

**3.7 Hardware Requirements**

A laptop, desktop, tablet or mobile device with at least 1gigabyte RAM and a functioning web browser e.g. Firefox and Chrome.

**CHAPTER FOUR**

**SYSTEM DESIGN AND IMPLEMENTATION**

**4.1 Introduction**

This chapter deals with the system implementation which is the actual development of the program and its documentation. The Hardware and Operating system requirement is also discussed here.

**4.2 Objective of the new system**

The objectives of the new system are:

1. Correctly allocate students to their various hostels.
2. Convenient user interface.
3. Easier end user interpretation.

**4.3 Main Menu (Control Centre)**

The main menu is the landing page of the System after a successful login from the user. From here the user can navigate to other parts of the application either by selecting Profile to check for his status or selecting the allocation list in case he/she wants to check for a friend or asking for help from support.

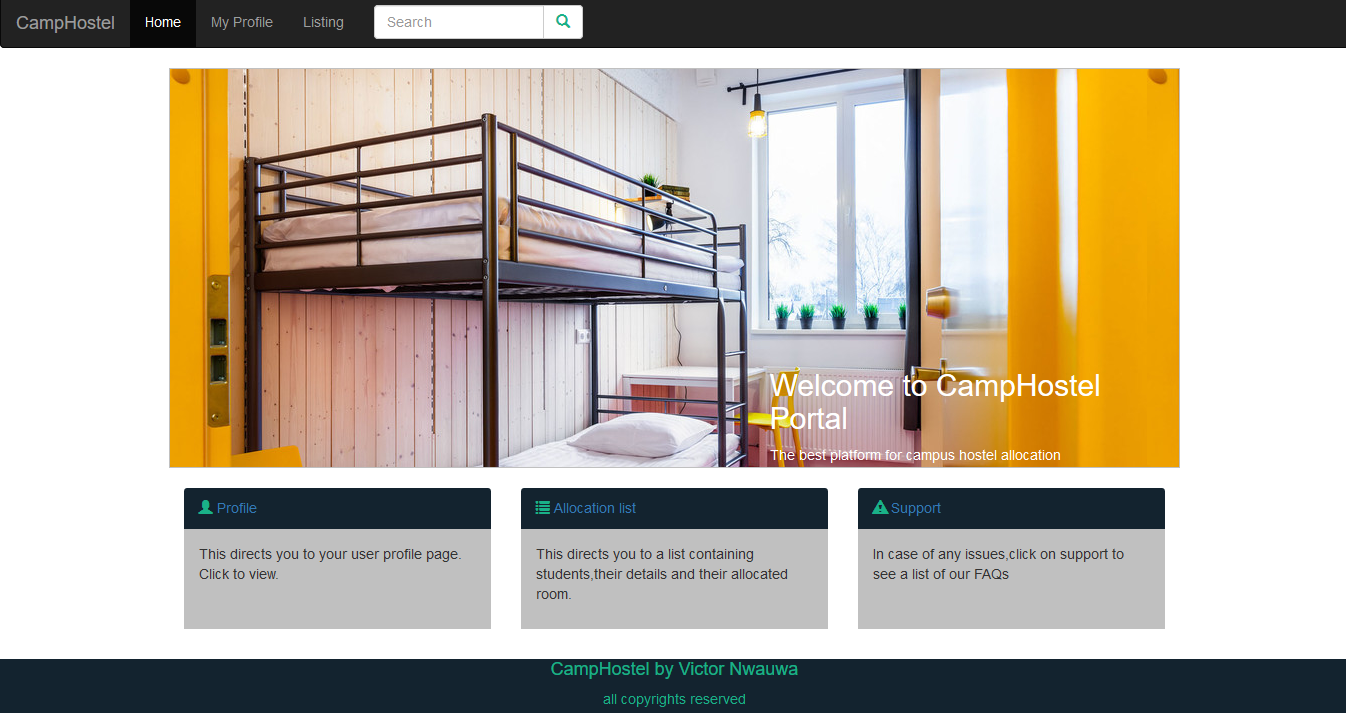


Fig 4.1: Main Menu

**4.4 Database Specification**

The database used in the design of this work is MySQL. A database with name of campHostel was created. It has two tables namely: register and create\_hostel.

The specification for the create\_hostel table is shown in table 4.1. In this table, the attributes used are hostelid, hostelname, capacity and hosteltype, with datatypes of variable character for storing alpha numeric character.

Table 4.1

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Type** | **Constraint** |
| hostelid | Varchar(255) | PRIMARY KEY |
| hostelname | Varchar(255) | NOT NULL |
| capacity | Int(11) | NOT NULL |
| hosteltype | Varchar(255) | NOT NULL |

The specification for the register table is shown in table 4.2. In this table, the attributes used are first name, last name, email, handicap, level, sex, age, matric number, and password with datatypes of variable character for storing alpha numeric character.

Table 4.2

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Type** | **constraint** |
| firstname | Varchar(255) | NOT NULL |
| lastname | Varchar(255) | NOT NULL |
| password | Varchar(255) | NOT NULL |
| id | Int(11) | PRIMARY­\_KEY |
| matnum | Varchar(255) | NOT NULL |
| age | Varchar(255) | NOT NULL |
| level | Int(11) | NOT NULL |
| handicap | Varchar(255) | NOT NULL |
| sex | Varchar(255) | NOT NULL |
| email | Varchar(255) | NOT NULL |

**4.5 Input/Output Screen/Format**

The input screen format shows the screen shot of all the input format in the program. The first input screen is the signup/login page where the user has to either sign up or login.

This is shown in figure 4.2. The second input screen is the register form where the user is required to fill when the user is not registered on the system. This is shown in figure 4.3. The last input screen is the area of the main program that is only visible to the admin where an admin can create, add or edit hostels by populating a table with data being in textboxes upon clicking the create hostel button. This is shown in figure 4.4.

**4.5.1 Login Module**

This is the first point where the user interacts with the program. The user is required to enter a matric number and password and from here if authentication is successful, the home page.

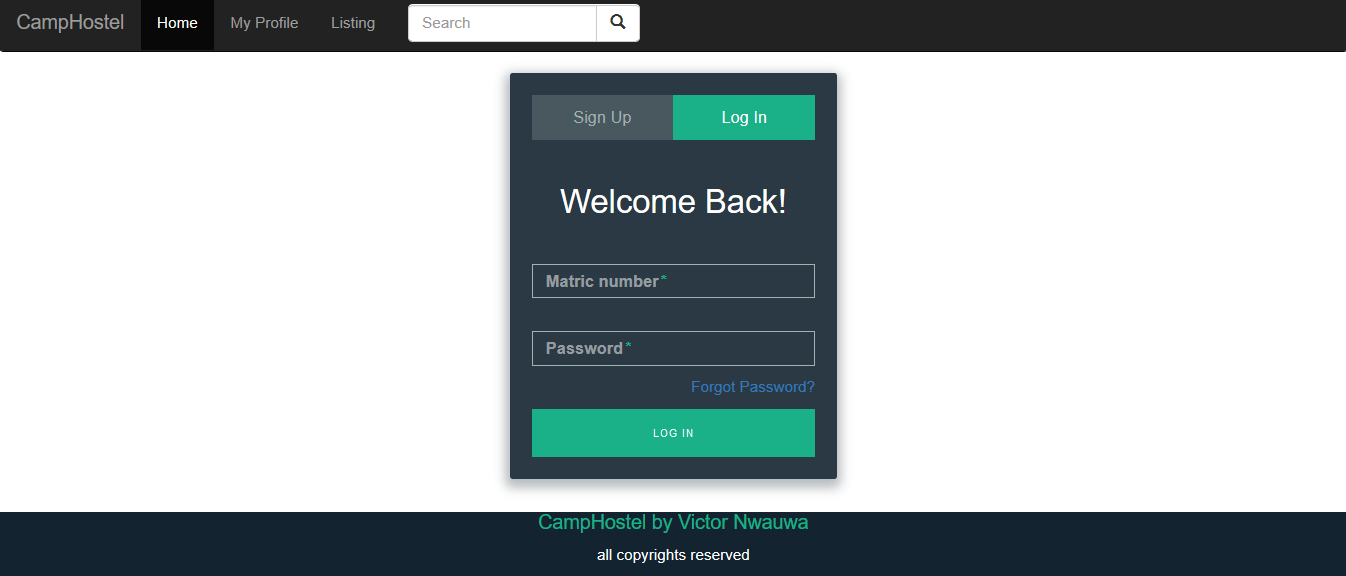


Fig 4.2: Login Module

**4.5.2 Register Module**

This module requires the user registers on the system using the requested fields. Once the fields are filled completely, the user clicks on the Register button and is taken back to the Login module.

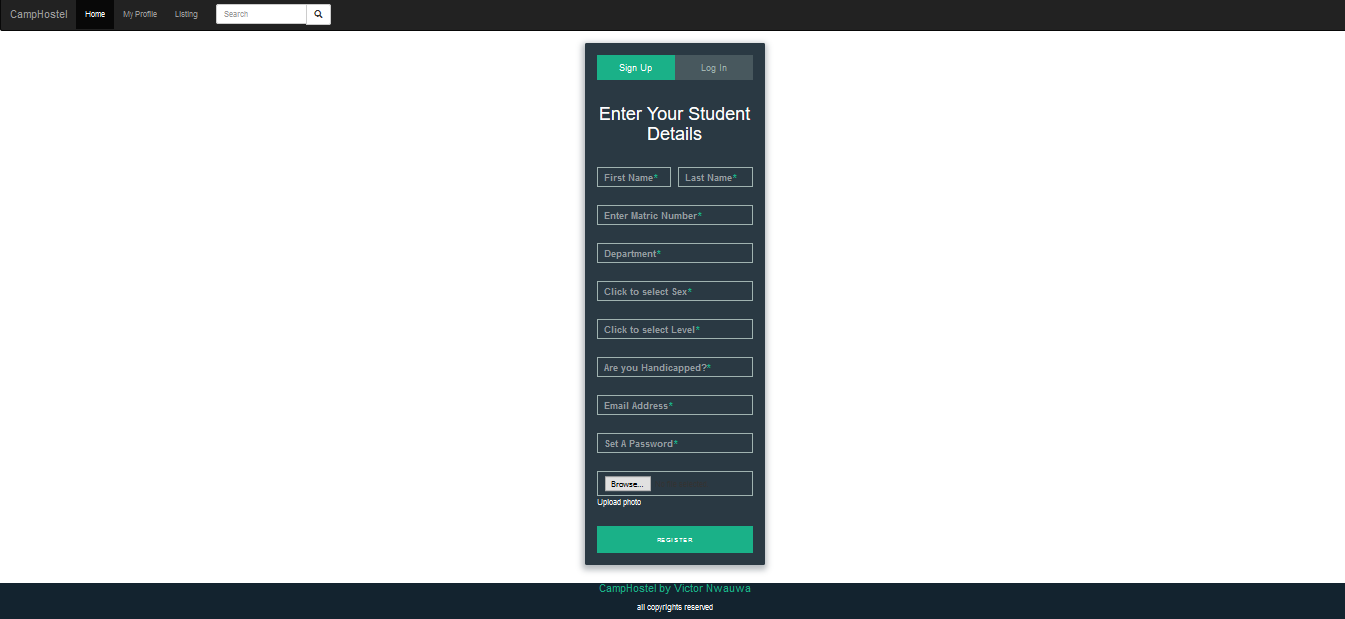
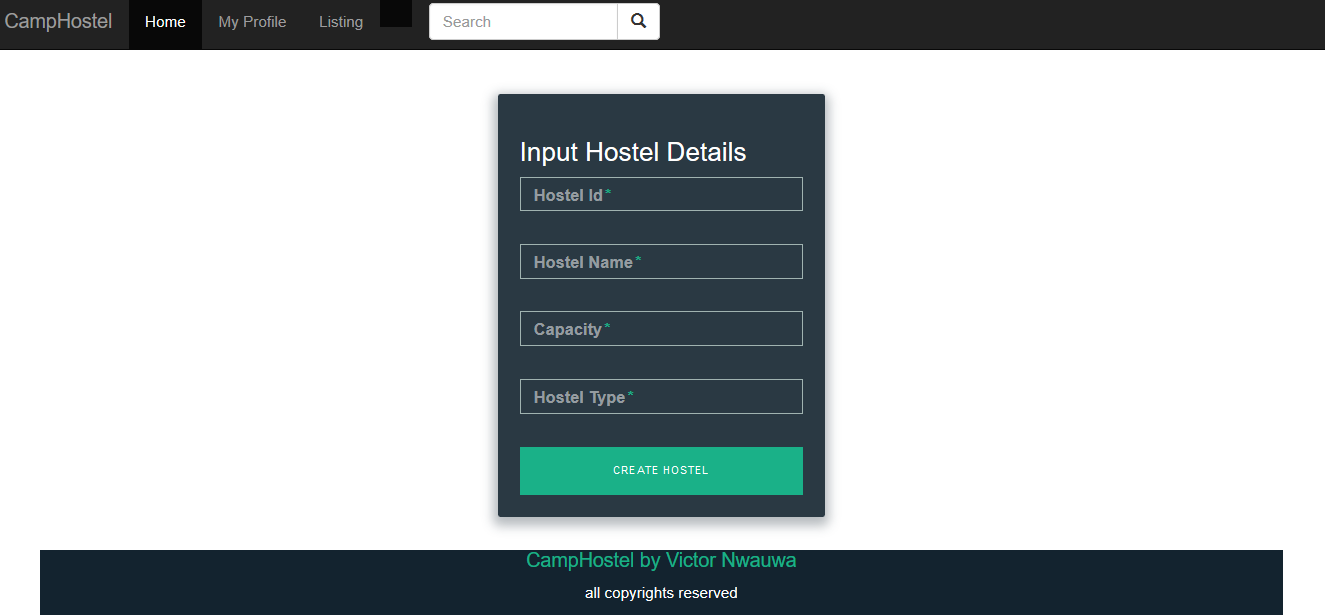


Fig 4.3: Register Module

**4.5.3 Admin Hostel Input Module**

This is a page linked to the admin dashboard where the admin can add or create a hostel by inputting on the required text field. Once the admin clicks on the Create Hostel button, the table being shown on the dashboard will be updated with the relevant data.

Fig 4.4 Hostel Input Module

**4.5.4 Profile Output Module**

The profile output module is the region of the software where the user gets to see his personal details and his/her allocation status. This results will depend on the user entering the correct authentication details.

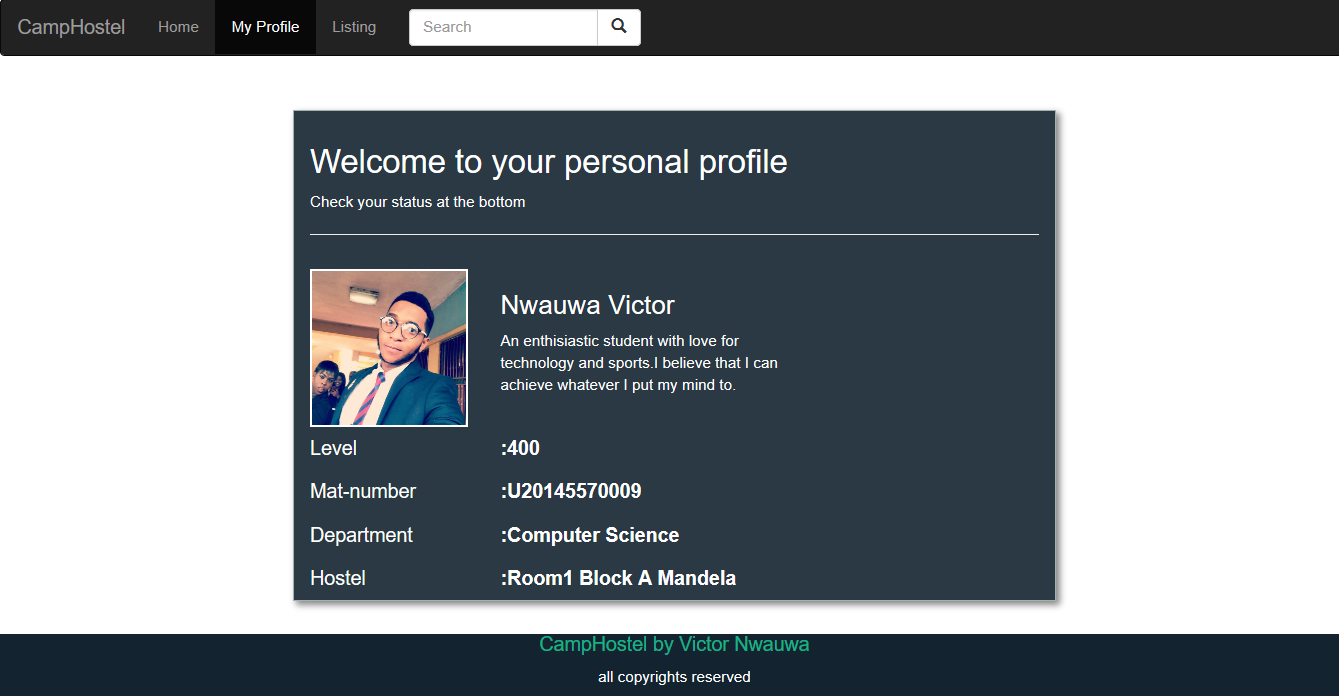
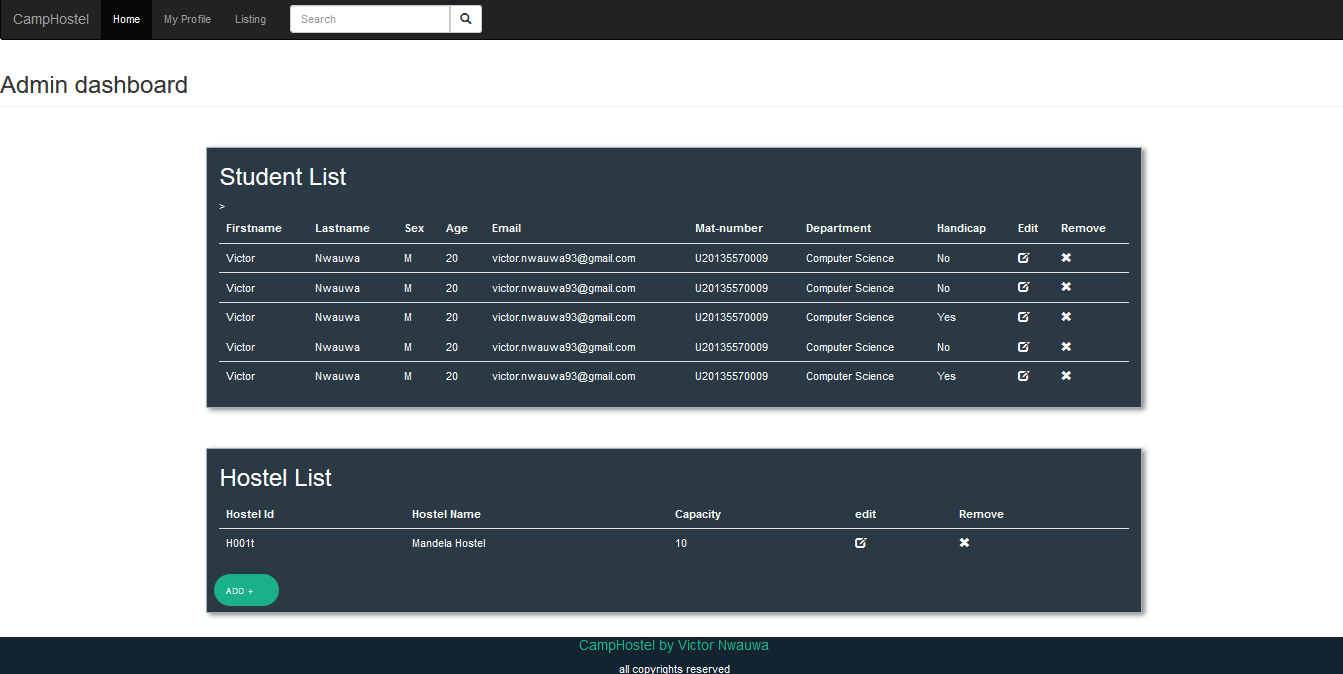


Fig 4.5 Personal profile Area

* + 1. **The Admin Dashboard Output Module**

This module displays a student’s record table and a hostel table. From this table, the admin can add new hostels and monitor for a system malfunction.



**4.6 Hardware and Operating System requirement**

The hardware and Software system needed to run the program include

**4.6.1 Hardware Specifications**

1. Processor Name: Intel Dual Core / AMD
2. Processor Speed: 1.66 GHz
3. RAM: 2 GB
4. Hard Disk Capacity: 80 GB
5. Display Device: 14’ to 19’ Inch Monitor
6. Keyboard Type: PS2 or USB
7. Mouse Type: PS2 or USB

**4.6.2**  **Software Specifications:**

1. Language Used: HTML, PHP, CSS, BOOTSTRAP, JAVASCRIPT, SQL
2. Software Used: MySql,Wamp Server ,PhpStorm
3. Operating System: Windows XP/ Windows 7/ Windows8 / Linux

**4.7 Software testing**

Testing is the process of running a system with the intention of finding errors. Testing enhances the integrity of a system by detecting deviations in design and errors in the system. Testing aims at detecting error-prone areas. This helps in the prevention of errors in a system. Testing also adds value to the product by conforming to the user requirements. The main purpose of testing is to detect errors and error-prone areas in a system. Testing must be thorough and well-planned. A partially tested system is as bad as an untested system. And the price of an untested and under-tested system is high.

The implementation is the final and important phase. It involves user-training, system testing in order to ensure successful running of the proposed system. The user tests the system and changes are made according to their needs. The testing involves the testing of the developed system using various kinds of data. While testing, errors are noted and correctness is the made.

The objectives of testing are:

1. Testing is a process of executing a program with the intent of finding errors.
2. A successful test case is one that uncovers an as yet undiscovered error.

System testing is a stage of implementation, which is aimed at ensuring that the system works accurately and efficiently as per the user need, before the live operation commences. As stated before, testing is vital to the success of a system. System testing makes a logical assumption that if all parts of the system are correct, the goal will be successfully achieved. A series of tests are performed before the system is ready for the user acceptance test.

**4.7.1 Testing Methods**

System testing is the stage of implementation. This is to check whether the system works accurately and efficiently before live operation commences. Testing is vital to the success of the system. The candidate system is subject to a variety of tests: on line response, volume, stress, recovery, security and usability tests. A series of tests are performed for the proposed system is ready for user acceptance testing. The testing Steps are:

1. **Unit Testing;**Unit testing focuses efforts on the smallest unit of software design. This is known as module testing. The modules are tested separately. The test is carried out during programming stage itself. In this step, each module is found to be working satisfactory as regards to the expected output from the module.
2. **Integration Testing**

Data can be lost across an interface. One module can have an adverse effect on another, sub functions, when combined, may not be linked in desired manner in major functions. Integration testing is a systematic approach for constructing the program structure, while at the same time conducting test to uncover errors associated within the interface. The objective is to take unit tested modules and builds program structure. All the modules are combined and tested as a whole.

1. **Validation**

At the culmination of the integration testing, software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of software test begin in validation testing. Validation testing can be defined in many ways, but a simple definition is that the validation succeeds when the software functions in a manner that is expected by the customer. After validation test has been conducted, one of the three possible conditions exists.

a. The function or performance characteristics confirm to specification and are accepted.

b. A deviation from specification is uncovered and a deficiency lists is created.

c. Proposed system under consideration has been tested by using validation test and found to be working satisfactory.

1. **Output Testing**

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in a specific format. The output format on the screen is found to be correct. The format was designed in the system design time according to the user needs. For the hard copy also; the output comes as per the specified requirements by the user. Hence output testing did not result in any correction for the system.

1. **User Acceptance Testing**

User acceptance of a system is the key factor for the success of any system. The system under consideration is tested for the user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes whenever required.

**CHAPTER FIVE**

**SUMMARY, CONCLUSION AND RECOMMENDATIONS**

**5.1 Summary**

In this work, the optimized Hostel Allocation System was developed to ensure that hostels are properly allocated to students who meet the set criteria. It was proposed to replace the existing brute force approach for allocating students to hostels. The pros and cons of both systems were discussed and a suitable Genetic module was developed to aid[ in the allocation process. An application was developed to show the proof of concept and from the result, the system is noted to have performed well.

**5.2 Conclusion**

The Hostel Space Allocation Problem(HSAP) can be solved using Genetic Algorithm that utilizes the rules of genetics and evolution. This system can perform better than the existing approach because of the inherent ability of genetic algorithm to optimization issues. This therefore means that Genetic algorithm can be extended to other areas of Space Allocation Problems (SAP).

**5.3 Recommendation**

In this work, there are other applications of Genetic Algorithm which could give a holistic insight to space allocation. This area includes, use of Genetic algorithm for optimizing Rank selection, use of Genetic Algorithm to optimize course allocation to lecturers in a department etc.