HARARE INSTITUTE OF TECHNOLOGY



DEPARTMENT OF INFORMATION TECHNOLOGY

BTECH (HONS) COMPUTER SCIENCE

[HIT ACCOMMODATION FINDER]

BY

[KEITH CHAREDZERA H190232Q]

[LIONNEL TSURO H190415Z]

[RODNEY MUPANDUKI H190197F]

[TINOTENDA MUKWAZHE H190086F]

HIT 200 SUPERVISOR: [MR C MAFIRABADZA]

**This project report was submitted to Harare Institute of technology in partial fulfilment of the bachelor of technology (Hons) degree in Computer Science**

Table of Contents

[**DECLARATION** 6](#_Toc65231822)

[**DEDICATION** 6](#_Toc65231823)

[ACKNOWLEDGEMENTS 6](#_Toc65231824)

[ABSTRACT 7](#_Toc65231825)

[CHAPTER 1 INTRODUCTION 7](#_Toc65231826)

[1.1 Introduction 7](#_Toc65231827)

[1.2 Motivation 7](#_Toc65231828)

[1.3 Problem Statement 8](#_Toc65231829)

[1.4 Related Work 8](#_Toc65231830)

[1.5 Hypothesis 8](#_Toc65231831)

[1.6 Technical Objectives 8](#_Toc65231832)

[1.7 Expected Results 9](#_Toc65231833)

[1.8 Ethics Consideration 9](#_Toc65231834)

[1.9 Conclusion 9](#_Toc65231835)

[CHAPTER 2 REQUIREMENTS ANALYSIS 9](#_Toc65231836)

[2.1 Introduction 9](#_Toc65231837)

[2.1.1 Evaluate Alternatives 9](#_Toc65231838)

[2.1.2 Outsource 10](#_Toc65231839)

[2.1.3 Improvement 10](#_Toc65231840)

[2.1.4 Development 10](#_Toc65231841)

[2.2 User Requirements 10](#_Toc65231842)

[2.2.1 Collection Phase 10](#_Toc65231843)

[2.2.2 Technical Feasibility 10](#_Toc65231844)

[2.2.3 Hardware 11](#_Toc65231845)

[2.2.4 Software 11](#_Toc65231846)

[2.2.5 Technical Expertise 12](#_Toc65231847)

[2.3 Economic Feasibility 12](#_Toc65231848)

[2.3.1 Cost Benefit Analysis 12](#_Toc65231849)

[2.3.2 Tangible Benefits 12](#_Toc65231850)

[2.3.3 Intangible 13](#_Toc65231851)

[2.4 Operational Feasibility 13](#_Toc65231852)

[2.4.1 Schedule Feasibility 13](#_Toc65231853)

[2.5 Work plan 13](#_Toc65231854)

[2.5.1 Work schedule 13](#_Toc65231855)

[2.5.2 Gantt chart 13](#_Toc65231856)

[2.6 Conclusion 14](#_Toc65231857)

[CHAPTER 3 SYSTEM ANALYSIS 14](#_Toc65231858)

[3.1 Introduction 14](#_Toc65231859)

[3.2. Description of current system 14](#_Toc65231860)

[3.3 Analysis of existing system 15](#_Toc65231861)

[3.3.1 Context diagram of the existing system 15](#_Toc65231862)

[3.3.2 Weaknesses of current system 16](#_Toc65231863)

[3.4 Description of the Proposed Solution 16](#_Toc65231864)

[3.4.1 Analysis of the proposed system- Context diagram, DFDS 17](#_Toc65231865)

[3.5 Requirements Analysis 18](#_Toc65231866)

[3.5.1 Functional Requirements (use case diagram) 18](#_Toc65231867)

[3.5.2 Non-functional requirements (outline constraints) 20](#_Toc65231868)

[3.6 System Models 21](#_Toc65231869)

[3.6.1 UML-Activity Diagram 21](#_Toc65231870)

[3.6.2 UML- Class Diagram 21](#_Toc65231871)

[3.6.3 UML-Sequence Diagram 21](#_Toc65231872)

[CHAPTER 4: SYSTEM DESIGN 21](#_Toc65231873)

[4.1 Introduction 21](#_Toc65231874)

[4.2 System Design 21](#_Toc65231875)

[4.2.1 How will the system work? 21](#_Toc65231876)

[4.3 Solution Architecture – architectural diagram of the proposed solution 21](#_Toc65231877)

[4.4 Database Modelling 21](#_Toc65231878)

[4.4.1 E-R Diagram 21](#_Toc65231879)

[4.4.2 Data Dictionary 21](#_Toc65231880)

[4.4.3 Database Schema 21](#_Toc65231881)

[4.5 Algorithm Design 21](#_Toc65231882)

[4.6 Interface Design 21](#_Toc65231883)

[4.7 Security Design 21](#_Toc65231884)

[4.8 Conclusion 21](#_Toc65231885)

[CHAPTER 5 IMPLEMENTATION AND TESTING 21](#_Toc65231886)

[5.1 Introduction 21](#_Toc65231887)

[5.2 Coding Strategy 21](#_Toc65231888)

[5.3 Coding Review 21](#_Toc65231889)

[5.4 Types of Testing and Results 21](#_Toc65231890)

[5.4.1 Functional Testing 21](#_Toc65231891)

[5.4.2 Non-Functional Testing 21](#_Toc65231892)

[5.5 Test Cases 21](#_Toc65231893)

[5.6 Levels of Testing and Results 21](#_Toc65231894)

[5.6.1 Unit Testing 21](#_Toc65231895)

[5.6.2 Integration testing 21](#_Toc65231896)

[5.6.4 System Testing 22](#_Toc65231897)

[5.7 Installation 22](#_Toc65231904)

[5.7.1 User training 22](#_Toc65231905)

[5.7.2 System conversion 22](#_Toc65231906)

[5.7.3 File conversion 22](#_Toc65231907)

[5.7.4 System changeover strategy 22](#_Toc65231908)

[5.8 Conclusion 22](#_Toc65231909)

[CHAPTER 6 CONCLUSION 22](#_Toc65231910)

[6.1 Scope for Future Extension 22](#_Toc65231911)

[6.2 Maintenance 22](#_Toc65231912)

[6.2.1 Interval System Review 22](#_Toc65231913)

[6.2.2 Maintenance Activities 22](#_Toc65231914)

[6.2.3 Disaster Recovery 22](#_Toc65231915)

[6.3 Recommendations 22](#_Toc65231916)

[APPENDIX 22](#_Toc65231917)

[USER MANUAL 22](#_Toc65231918)

**DECLARATION**

This serves to notify that this document is an original copy of the actual thoughts of the people involved. No infringements or plagiarism is involved in the writing of this document We do hereby declare that the information provided is true and has undergone live inspection as well as researches from supervisors.

I \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ do hereby swear that the information provided is my original copy .Date \_\_\_/\_\_\_/\_\_\_\_\_\_\_signed \_\_\_\_\_\_\_\_.

I \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ do hereby swear that the information provided is my original copy .Date \_\_\_/\_\_\_/\_\_\_\_\_\_\_signed \_\_\_\_\_\_\_\_.

I \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ do hereby swear that the information provided is my original copy .Date \_\_\_/\_\_\_/\_\_\_\_\_\_\_signed \_\_\_\_\_\_\_\_.

I \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ do hereby swear that the information provided is my original copy .Date \_\_\_/\_\_\_/\_\_\_\_\_\_\_signed \_\_\_\_\_\_\_\_.

**DEDICATION**

This document is dedicated to the brains behind the HIT Accommodation Finder system.

# ACKNOWLEDGEMENTS

Special thanks goes to Mr. C Mafirabadza for supervising us as well as giving us guidance throughout the project.

# ABSTRACT

At Harare Institute of Technology there are few rooms compared to the number of students, hence there will be a lot of accommodation challenges for the students. Students apply for on-campus accommodation at the Dean’s office only, then the Dean decides whether the students can be offered accommodation. As a result of this manual process, there is a significant measure of strain on the individual running the hostel.

Our HIT Accommodation Finder System project deals with the problems of applying for a hostel, as well as avoiding the problems which can come about in the current manual system. Recognition of the problems with the current system led to the designation of a computerized system that will be more convenient and reliable for every user.

# CHAPTER 1 INTRODUCTION

## Introduction

The HIT Accommodation Finder, is a web-based platform aimed at facilitating HIT students who are eagerly in need of finding on or off-campus accommodation which suits their needs and budget, with the ease of applying from anywhere

## 1.2 Motivation

Motivated by the various challenges, students have to face in securing accommodation places which can be very stressful and frustrating especially those from outside Harare and abroad. Most importantly we were motivated by the greater conveniences that would be brought by the platform, to all the users of the platform that includes the Dean of Students, the students and the off-campus residence owners. All the information they need will be provided at their fingertips. We are also striving to contribute to our Institute’s quality service delivery hence boosting the school’s global recognition and prestige when it comes to software engineering and development. Some of us had to travel a lot of kilometres just to search and secure accommodation

## 1.3 Problem Statement

The existing system is not computerised, hence there will be more work for the person allocating the residency places. There is congestion at the Dean’s office during the applications. Students who stay in towns which are far away from the campus have a hard time applying because application is only done at the Dean’s office. Apparently, some students are given rooms in hostels which they did not apply for, and it can be costly since the prices for accommodation are different. If a student fails to get on-campus accommodation, he/she will be forced to look for off campus accommodation in a short period of time, hence they may end up living in uncomfortable environments. If new students fail to acquire on-campus accommodation they will be stranded because they wouldn’t know any trustworthy places to look for accommodation.

## 1.4 Related Work

We did a literature review so as to understand the background of the system being proposed.

We found the following:

* In Zimbabwe, there is ***The Housing Hub,*** a website that is used to offer online accommodation reservations for off campus accommodation seekers in Zimbabwean universities. It uses jQuery for frontend, Firebase as a database and LiteSpeed a s its Web server. [ https://www.thehousinghub.co.zw/ ]
* ***studentpad*** is a website that is used to find student accommodation in the UK. They serve over 80 higher education institutions and students can look for rooms in private student houses, on campus. The site uses HTML, CSS and React JS for their frontend, Amazon for their database and Nginx for their Webserver. [ <https://www.studentpad.co.uk/> ]
* ***Universityrooms*** is a website that is used to find accommodation for students, and if those rooms are not occupied then anyone can save money and stay in the empty university rooms even if they're not a student. It uses React JS for frontend, ASP.NET for backend and Cloudflare Server as its server. [ <https://www.universityrooms.com/en-GB/> ]
* ***Roomi is*** a website where students can look for accommodation as well as looking for someone to move in with so as to split the rent. It provides a platform where students can chat with each other privately whilst keeping their contact information privately. If someone has a room that they want to rent out, they first post the room, then it is verified by ***Roomi*** and finally, after verification the owner of the place will approve or decline booking requests Payments are made through the ***Roomi*** platform. It uses React JS for its frontend and Amazon Web Service as platform as a Service[ <https://roomiapp.com/> ]
* ***StudentSpareroom*** is a student accommodation finder website for private rented student accommodation. It helps students to find other students to share rooms or flats with. It allows landlords to advertise rooms in existing student flatshares. For frontend they used jQuery and Apache as their web server [ <https://student.spareroom.co.uk/> ]
* We did a study of an online hotel reservation system for ***Kensington Place*** (a hotel in Cape Town, South Africa). The hotel incorporated the online system so as to prevent customers from waiting in long queues after travelling long distances and also to prevent customers from being stranded if they fail to find a place to sleep at night since they can pre-book. This automated system helps the reservation system to be more efficient, fast and accurate. For their frontend they used React JS and jQuery, and MySQL for their database. [ <https://www.kensingtonplace.co.za/> ]

## 1.5 Hypothesis

The hypothesis of this research is:

* Accommodation application system will be more efficient as it will require less overhead
* Student can do their registration easily by the student page and room allotment will be handled by the Dean and he/she will allocate he room according to availability
* Paper work and redundancy will be reduced hence improving productivity and lowering the cost of printing and purchasing registration materials annually.

## 1.6 Technical Objectives

The system is primarily targeted:

* To eliminate the inconvenience of students having to wait in long queues just to get accommodation clearances.
* To offer students with a wide variety of accommodation choices to choose from (On or off-campus).
* To send notifications to successful and unsuccessful applicants on time, so that they can make decisions on time

## 1.7 Expected Results

* A web-based system which will eliminate the paper-based system
* The admin of the system (Dean) will be able to login in the system, view the applications of the students and then send approvals to every approved student.

## 1.8 Ethics Consideration

* The first consideration will be to get approval of the research project and about
* The data will be used for research purposes only and we will ensure confidentiality and that data will not be disclosed.
* Comments from any user will be kept anonymous so that the users will not be victimised

## 1.9 Conclusion

Our proposed system will make the accommodation application and allocation process computerized. The system will solve all the stated problems and fulfil the technical objectives and the expected results.

# CHAPTER 2 REQUIREMENTS ANALYSIS

## 2.1 Introduction

Requirements analysis is the process of gathering facts hence in this chapter we will take a look at the current system, its processes and activities, so as to gain an understanding of the system as well as how our proposed system can improve and to see which advantages it has over its alternatives.

### 2.1.1 Evaluate Alternatives

Sirvoy Booking System is a highly customizable property management system that is designed to suit all accommodation providers. It offers the flexibility to tailor the software to suit the needs of their property. However, there are limits to control over the software and it has the following drawbacks compared to our system:

* Can contain too many unwanted features and functions.
* May have integration issues and not fit the institution work process.
* Can be more expensive over time due to upgrades and support costs.
* May not be supported after time.

### 2.1.2 Outsource

Outsourcing is the process of obtaining services from an outside supplier, hence there is low level flexibility and freedom. Every change must be negotiated with the party hired and it may also include extra costs depending on the complexity.

Organisations usually opt for outsourcing without full understanding, hence leading to poorly specified requirements.

### 2.1.3 Improvement

### 2.1.4 Development

Web based systems only need placement on a server,

## 2.2 User Requirements

For access to the system a user will need:

* A device with internet connection (PC or mobile phone)
* A username
* A password

### 2.2.1 Collection Phase

The fact-finding methods that can be used include:

* Interviews – this refers to the face-to-face communication between two or more people in order to obtain information. The interviewer asks the interviewee some questions.
* Record inspection – it involves scrutinising system documents in order to solicit information
* Questionnaires – They are documents with carefully crafted questions to be answered by the responded by filling on the spaces provided
* Observation – It involves viewing the actual system in operation by the researcher, and the researcher can also take part in operating the system in order to see how the system operates.

### 2.2.2 Technical Feasibility

Technical feasibility determines if the organisation can obtain software, equipment, technology and personnel to develop, install and operate the system effectively. A research is carried out on whether the new technology exists to support the development of the system, as it may happen that after a system is prepared, a new technology may arise and the user may want the system based on that technology.

This system will use HTML, CSS and React for front-end development, Node JS for backend, MongoDB for database and Git for project management.

### 2.2.3 Hardware

To implement the Accommodation Finder System, the Institute will need a domain and a database server. The website will be hosted on the web server and it will be accessible via the internet.

The hardware requirements are:

|  |  |
| --- | --- |
| **Hardware Component** | **Minimum Required** |
| Processor | Intel(R) Core(TM) i3 CPU @ 2.00GHz |
| Memory (RAM) | Minimum 4GB |
| Storage (HDD/SSD) | 20GB |
| Internet connection device | At least 1Mbps internet speed |

***Table 1: Hardware requirements***

### 2.2.4 Software

The following software will be used:

|  |  |
| --- | --- |
| **Component** | **Requirement** |
| Operating System (Development) | Windows 10/Linux/Mac OS |
| Integrated Development Environment | Visual Studio Code |
| Database | MongoDB |
| Testing Environment | Web Browser |
| Document editor (Documentation) | Microsoft Word |
| Version Control | Git/GitHub Desktop |

***Table 2: Software Requirements***

### 2.2.5 Technical Expertise

## 2.3 Economic Feasibility

It is whereby it is determined whether the benefits of the new system will out-weigh the estimated cost involved in developing, purchasing, installing and maintenance of the new system. The cost benefits analysis is important.

As seen in the previous section, there is not any hardware or software that needs to be bought, hence the development costs are minimal.

### 2.3.1 Cost Benefit Analysis

|  |  |
| --- | --- |
| **Item** | **Total cost ($ in USD)** |
| Documentation | 10 |
| Laptop | 350 |
| Internet | 100 |
| Research | 10 |
| **Total** | 470 |

***Table 3: Table of costs***

### 2.3.2 Tangible Benefits

Reduced Costs – The system will save costs of purchasing materials that are associated with printing.

Saved time – Since the system is automated, it will save the Dean some time since he/she will not have to do the process manually.

Reduced Paperwork – The automated system will reduce the paper used in the system.

Reduced Errors – Computerisation will reduce redundancy, and it will promote fairness in selecting successful applicants.

### 2.3.3 Intangible

Increased Productivity – The computerized system will save time; hence the Dean can use that time to do other tasks therefore increasing productivity.

Better User Experience – The application process will be more convenient for the users since the system will be user friendly.

Increased Compliance – The project is aimed at addressing any systems or processes which will address compliance, in this case, automating the process.

Responsive – The system’s data will be updated in real time based on user interactions.

## 2.4 Operational Feasibility

Determines whether the current work practices and procedures are adequate to support the system.

The system will be user friendly and will not require any technical expertise to operate, hence the project is operationally feasible.

### 2.4.1 Schedule Feasibility

The developers are familiar with web development, and the project is moderate in terms of size.

## 2.5 Work plan

### 2.5.1 Work schedule

|  |  |  |  |
| --- | --- | --- | --- |
| TASK | START | FINISH | DUARTION (WEEKS) |
| Project Proposal |  |  |  |
| Documentation |  |  |  |
| Planning Phase |  |  |  |
| Analysis Phase |  |  |  |
| Design Phase |  |  |  |
| Coding |  |  |  |
| Testing |  |  |  |
| Implementation |  |  |  |

### 2.5.2 Gantt chart

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | November | December | January | February | March | April | May | June |
| Project Proposal |  |  |  |  |  |  |  |  |
| Documentation |  |  |  |  |  |  |  |  |
| Planning Phase |  |  |  |  |  |  |  |  |
| Analysis Phase |  |  |  |  |  |  |  |  |
| Design Phase |  |  |  |  |  |  |  |  |
| Coding |  |  |  |  |  |  |  |  |
| Testing |  |  |  |  |  |  |  |  |
| Implementation |  |  |  |  |  |  |  |  |

## 2.6 Conclusion

In this chapter, we

Lporem\

\\ Lorem

\\ Lorem

\\ Lorem

\\ Lorem

# CHAPTER 3 SYSTEM ANALYSIS

## 3.1 Introduction

The purpose of the system analysis and design phases is to refine the project goals into defined functions and operation of the intended application. System requirements are documented using the Unified Modelling Language.

The Unified Modelling Language (UML) is a modelling language that may be used to visualise, specify, construct, and document the artefacts of a software system. It provides a set of notations to create a visual model of the system.

System analysis is a method of problem solving that deals with the breaking down of a system into components parts in order to study how well the individual parts work and interact to accomplish their purpose. It involves the process of enumerating the existing problems, analysing the proposed systems for costs and benefits, analysing the system and user requirements, and considering possible alternative systems.

## 3.2. Description of current system

The existing system is manual based hence there is need for a lot of effort and time. Currently students in need of on campus accommodation have to do the whole process manually.

The process is follows:

* The students first provide their proof of registration to the Dean of Students
* The students must wait in long queues as the Dean decides if each student deserves to get on campus accommodation

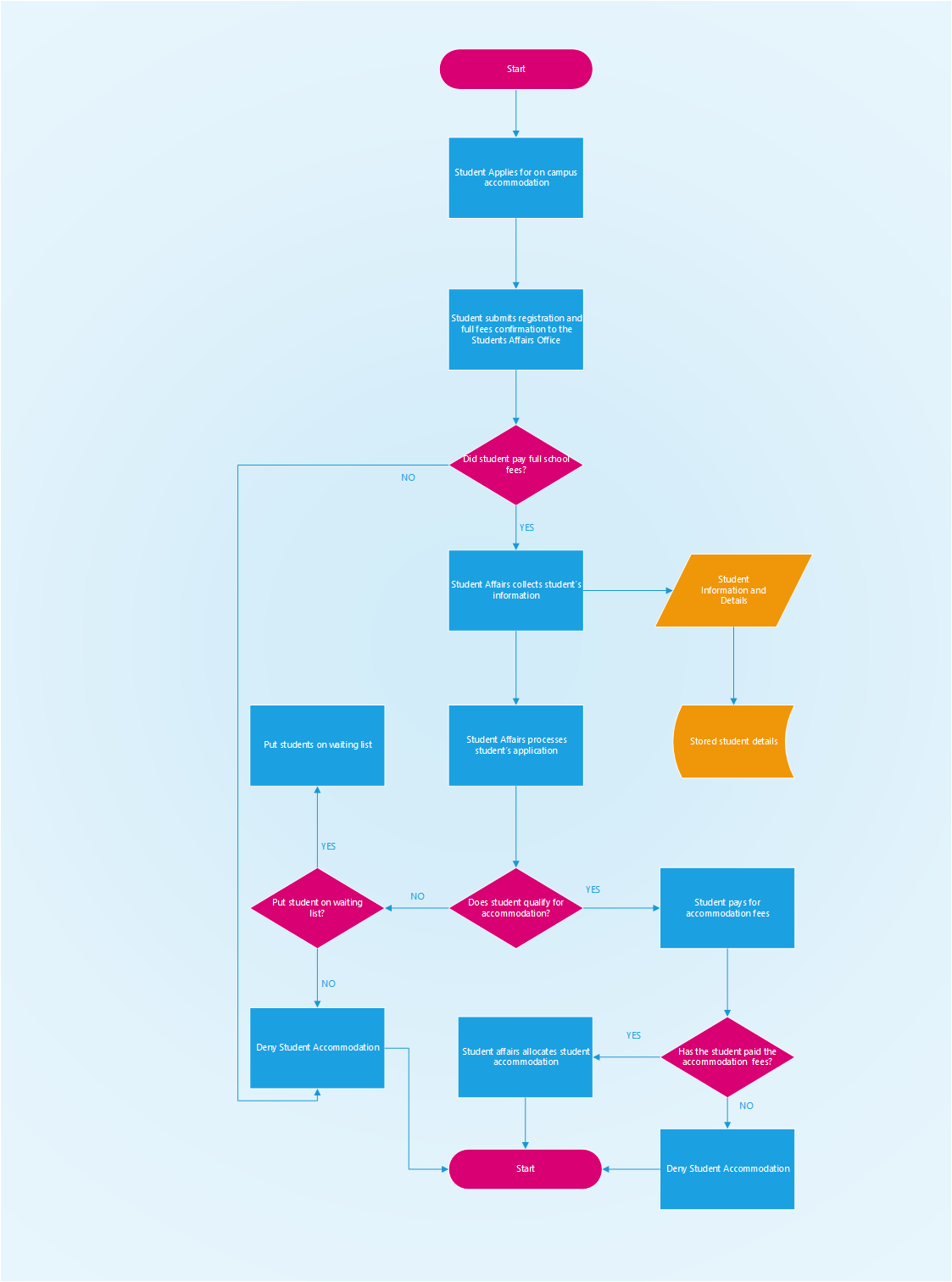
This is unfortunate as some students who stay far away in other cities have to go to campus to apply but are still not guaranteed residence places.

This may also lead to corruption in the allocation process, since there will be high demand.

For off-campus accommodation, students have to search physically for unoccupied boarding houses.

## 3.3 Analysis of existing system

### 3.3.1 Context diagram of the existing system



### 3.3.2 Weaknesses of current system

This phase deals with the weaknesses/problems in the current system.

The problems faced are:

* Data redundancy – Due to a lot of paper work, the same data is recorded repeatedly.
* Time consumption – It takes up a lot of time to write each and every entry in the database.
* Human errors or Favouritism – Data may be inconsistent due to human errors. Allocation may also be unfair due to favouritism.
* More human labour needed to speed up the process
* Data is difficult to edit – manually written data cannot be changed once written. An attempt to rub out the unwanted data will result in the paper being messy
* Data is difficult to handle, maintain or update – The books containing the data may be misplaced, or torn.

## 3.4 Description of the Proposed Solution

To eliminate all the problems faced by the current system we have proposed a HIT Accommodation Finder system which is web based. It will automate all the processes of the system.

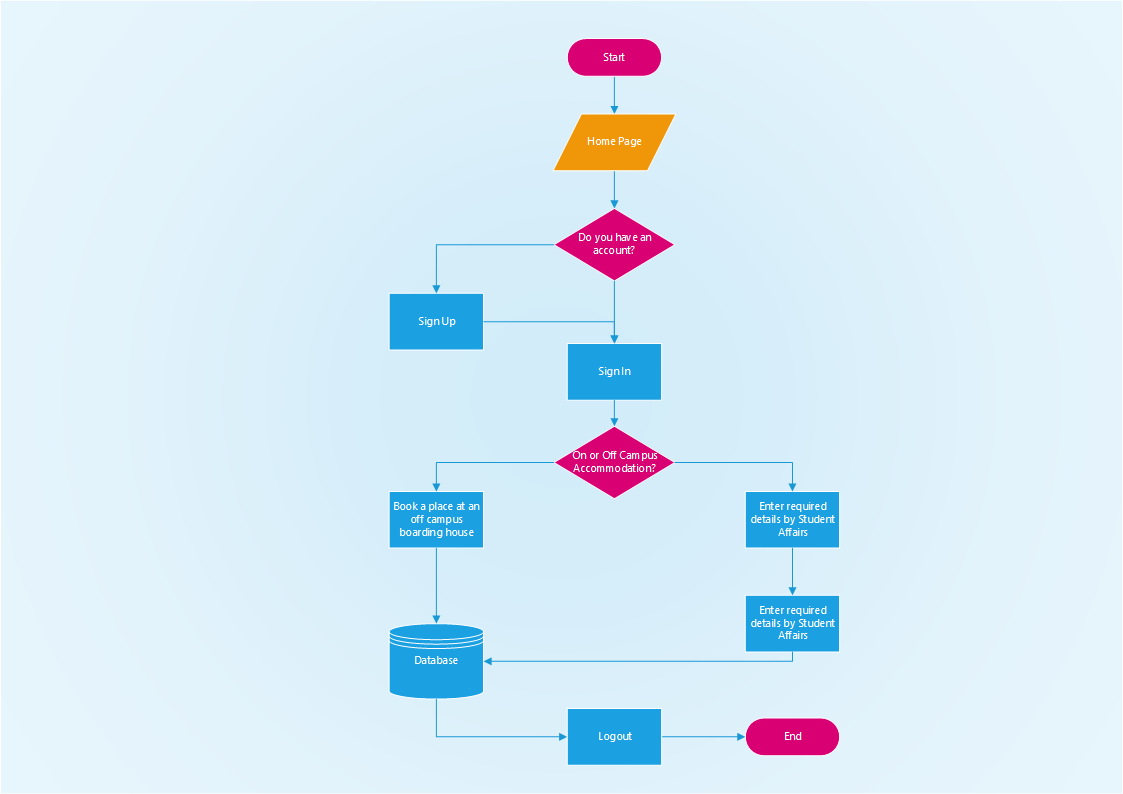
Our proposed system is a website with the following functions:

* Give different access levels of the system depending om the type of user (dean/student)
* Avoiding the duplication of data
* Design a database which will keep every transaction related to each application process

### 3.4.1 Analysis of the proposed system- Context diagram, DFDS

Dean’s side

Students side



## 3.5 Requirements Analysis

### 3.5.1 Functional Requirements (use case diagram)

Functional requirements are specific modules of a system, defined by how they react to user input and produces output. They describe the core functionality of the system. They are the key to the system running and completing its stated objectives. These are stated below

Use Case Models represent the different ways in which a system can be used by the users. They correspond to functional requirements and their purpose is to define a piece of coherent behaviour without revealing the internal structure of the system.

### 3.5.2 Non-functional requirements (outline constraints)

A non-functional requirement is a requirement that specifies a criterion that can be used to judge the operation of a system, rather than specific behaviours.

The non-functional requirements are:

* Security – There should be different access levels to the system and users will require passwords.
* Portability – System should be hosted on a server without causing any problems.
* Performance –This includes requirements such as speed of the system. The system should run smoothly for both the administration and users. Tracking and updating of applications should be done easily. Queries in the database on the server should be performed within a few seconds.
* Maintainability – The system should be easily maintainable without causing problems.
* Availability – The system should always be available and accessible at any time.
* Responsive – The system’s data should update in real time based on user interactions.

## 3.6 System Models

### 3.6.1 UML-Activity Diagram

### 3.6.2 UML- Class Diagram

### 3.6.3 UML-Sequence Diagram

# CHAPTER 4: SYSTEM DESIGN

## 4.1 Introduction

This stage is concerned with the decision on the look, feel and functionality of the system.

## 4.2 System Design

### 4.2.1 How will the system work?

An internet connection is required for any user to access the Accommodation Finder website.

## 4.3 Solution Architecture – architectural diagram of the proposed solution

## 4.4 Database Modelling

### 4.4.1 E-R Diagram

### 4.4.2 Data Dictionary

### 4.4.3 Database Schema

## 4.5 Algorithm Design

## 4.6 Interface Design

## 4.7 Security Design

## 4.8 Conclusion

# CHAPTER 5 IMPLEMENTATION AND TESTING

## 5.1 Introduction

## 5.2 Coding Strategy

## 5.3 Coding Review

## 5.4 Types of Testing and Results

### 5.4.1 Functional Testing

### 5.4.2 Non-Functional Testing

## 5.5 Test Cases

## 5.6 Levels of Testing and Results

### 5.6.1 Unit Testing

### Integration testing

**5.6.3** **Validation testing**

### 5.6.4 System Testing



## Installation

### 5.7.1 User training

### 5.7.2 System conversion

### 5.7.3 File conversion

### 5.7.4 System changeover strategy

## 5.8 Conclusion

# CHAPTER 6 CONCLUSION

## 6.1 Scope for Future Extension

## 6.2 Maintenance

### 6.2.1 Interval System Review

### 6.2.2 Maintenance Activities

### 6.2.3 Disaster Recovery

## 6.3 Recommendations

# APPENDIX

# USER MANUAL

**NB: A Technical paper is to be submitted together with this document.**