**GharBeti**

Submitted in partial fulfillment of the requirement of

**Project – IV** (**BEG479CO**)

Of

Bachelor of Computer Engineering

****

**Submitted to**

Purbanchal University

Biratnagar, Nepal

Submitted By:

Bikash Gupta(382216)

Laxmi Gajurel (382218)

Pratik Subedi(382222)

**KANTIPUR CITY COLLEGE**

Putalisadak, Kathmandu

August 10, 2022

A Project Reports

On

**GharBeti**

Submitted in partial fulfillment of the requirement

Of

**Project–IV (BEG479CO)**

Of

****Bachelor of Computer Engineering

**Submitted to**

Purbanchal University

Biratnagar, Nepal

**Submitted By:**

Bikash Gupta(382216)

Laxmi Gajurel (382218)

Pratik Subedi (382222)

**KANTIPUR CITY COLLEGE**

Putalisadak, Kathmandu

August 10, 2022

# TOPIC APPROVAL SHEET

It is hereby informed that the topic selected by Bikash Gupta, Laxmi Gajurel and Pratik Subedi of Bachelor of computer engineering, Semester VIII for their semester project has been found suitable and as per the credit assigned by Purbanchal University (PU), Biratnagar, Nepal.

The project committee has approved the following topic for the above mention students.

**Topic Assigned: GharBeti**

\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_

Ashim Kc. Raju Kattel Merry Singh

Project Coordinate Principal Supervisor

# CERTIFICATE FROM SUPERVISOR

The Evaluation Committee has approved and recommended to the Department of Science and Technology for acceptance of this project report entitled “Gharbeti” submitted by:

Bikash Gupta(Symbol No:382216)

Laxmi Gajurel (Symbol No.: 382218)

Pratik Subedi (Symbol No: 382222)

Evaluation Committee

Mrs. Merry Singh

Supervisor,

Kantipur City College

(---------------------------------------)

External Examiner

Mr. Raju Kattel

Head of the Department of IT

Kantipur City College

Date: 2079/04/28

**ACKNOWLEDGEMENT**

We would like to express our sincere gratitude and thanks to Kantipur City College and Deputy Head of the department Mr. Subash Rajkarnikar for providing us this opportunity to perform our project.

And, our extreme gratitude goes to our supervisor Mrs. Merry Singh who guided us throughout the project. She has contributed her precious time and energy to complete our project. Without her willing, accommodation, frankness, suggestions and timely clarification, this project could not have been completed in due time.

We would like to thank all other individuals who have contributed directly or indirectly to the success of this project.

# ABSTRACT

Our Project is about “GharBeti” aimed at developing an application of “Property Rental Management System” for finding a place to rent. This system can be used to find the details of the rental place, update the rental detail, produce records of tenant payments and add, edit and remove the tenant and landlord detail, etc. This is one integrated system that contains both the user component (used by landlord, and tenant) and the admin component (used by the administrators for performing admin-level functions such as adding or removing landlord or room to the System, and verifying landlords) This system runs on multiple platforms, offers a GUI interface to its users, and connects to a common database.

# Table of Contents

[TOPIC APPROVAL SHEET i](#_Toc111230993)

[CERTIFICATE FROM SUPERVISOR ii](#_Toc111230994)

[**ACKNOWLEDGEMENT** iii](#_Toc111230995)

[ABSTRACT iv](#_Toc111230996)

[Table of Contents v](#_Toc111230997)

[List of Table viii](#_Toc111230998)

[List of Tables ix](#_Toc111230999)

[ABBREVIATIONS x](#_Toc111231000)

[Chapter 1: Introduction - 1 -](#_Toc111231001)

[1.1. Introduction - 1 -](#_Toc111231002)

[1.2. Problem Statement - 2 -](#_Toc111231003)

[1.3. Objectives - 4 -](#_Toc111231004)

[1.4. Significance of project - 4 -](#_Toc111231005)

[1.5. Project Feature - 4 -](#_Toc111231006)

[1.6. Assignment of role and responsibility - 4 -](#_Toc111231007)

[1.7. Project Documentation - 5 -](#_Toc111231008)

[Chapter 2: Literature Survey - 6 -](#_Toc111231009)

[2.1. Literature Review - 6 -](#_Toc111231010)

[2.2. Existing System - 7 -](#_Toc111231011)

[2.3. Analysis of issue - 8 -](#_Toc111231012)

[2.4. Solution - 8 -](#_Toc111231013)

[Chapter 3: System Analysis - 9 -](#_Toc111231014)

[3.1. System Development model - 9 -](#_Toc111231015)

[3.2. Requirement Specification - 10 -](#_Toc111231016)

[3.2.1. Functional Requirement - 10 -](#_Toc111231017)

[3.2.2. Non-Functional Requirement - 12 -](#_Toc111231018)

[Hardware Interfaces - 12 -](#_Toc111231019)

[Software Interface - 12 -](#_Toc111231020)

[Other Non-Functional Requirements - 12 -](#_Toc111231021)

[2. The system shall be intuitively usable. - 13 -](#_Toc111231022)

[Hardware Requirements - 13 -](#_Toc111231023)

[Software Requirements - 13 -](#_Toc111231024)

[3.3. Feasibility Study - 14 -](#_Toc111231025)

[3.3.1. Operational Feasibility - 14 -](#_Toc111231026)

[3.3.2. Technical Feasibility - 14 -](#_Toc111231027)

[3.3.3. Economic Feasibility - 14 -](#_Toc111231028)

[3.3.4. Legal and Contractual Feasibility - 15 -](#_Toc111231029)

[3.3.5. Scheduled Feasibility - 15 -](#_Toc111231030)

[Chapter 4: Design Specification and Implementation - 16 -](#_Toc111231031)

[4.1. System Architecture - 16 -](#_Toc111231032)

[4.2. Data Flow Diagram (DFD) - 17 -](#_Toc111231033)

[4.2.1. Context Diagram - 17 -](#_Toc111231034)

[4.2.2. DFD Level 1 - 18 -](#_Toc111231035)

[4.3. Use Case Diagram - 19 -](#_Toc111231036)

[3.4. Activity diagram - 20 -](#_Toc111231037)

[3.5. Database Diagram - 21 -](#_Toc111231038)

[4.5.1. E-R Diagram - 22 -](#_Toc111231039)

[4.5.2. Relational Data Structure - 23 -](#_Toc111231040)

[4.5.3. Data Dictionary - 23 -](#_Toc111231041)

[4.6. UI/UX Mechanism - 25 -](#_Toc111231042)

[Chapter 5 Experimental Result and Analysis - 26 -](#_Toc111231043)

[5.1. Programming platform - 26 -](#_Toc111231044)

[5.2. Operating Environment - 26 -](#_Toc111231045)

[5.3. Testing and De bugging - 27 -](#_Toc111231046)

[5.3.1. Testing - 27 -](#_Toc111231047)

[5.3. Experiment and Result Analysis - 28 -](#_Toc111231048)

[5.3.1. Experiment - 28 -](#_Toc111231049)

[5.3.2. Expected Result - 28 -](#_Toc111231050)

[Chapter 6: Conclusion and Future Enhancement - 29 -](#_Toc111231051)

[6.1. Conclusion - 29 -](#_Toc111231052)

[6.2. Limitations - 29 -](#_Toc111231053)

[6.3. Future Enhancement - 29 -](#_Toc111231054)

[Reference - 30 -](#_Toc111231055)

[Appendixes - 30 -](#_Toc111231056)

[Appendix1: Screenshot - 30 -](#_Toc111231057)

[Appendix2: Project Completion Gantt chart - 31 -](#_Toc111231058)

[Appendix3: User Manual - 32 -](#_Toc111231059)

[1.0. General Information - 32 -](#_Toc111231060)

[Appendix4: Revenue Model - 37 -](#_Toc111231061)

# List of Table

[Figure 1: Waterfall model - 10 -](#_Toc111230983)

[Figure 2: System Architecture - 16 -](#_Toc111230984)

[Figure 3: Context Diagram - 17 -](#_Toc111230985)

[Figure 4: Level 1 DFD - 18 -](#_Toc111230986)

[Figure 5: Use Case Diagram - 19 -](#_Toc111230987)

[Figure 6: Activity diagram - 20 -](#_Toc111230988)

[Figure 7: ER Diagram - 22 -](#_Toc111230989)

[Figure 8: Project Completion Gantt Chart - 31 -](#_Toc111230990)

[Figure 9: Revenue Model - 37 -](#_Toc111230991)

[Figure 10: Customer pay for a solution - 38 -](#_Toc111230992)

# List of Tables

[Table 1: Functional Table - 11 -](#_Toc111230973)

[Table 2: Gantt Chart - 15 -](#_Toc111230974)

[Table 3: Relational Data Structure - 23 -](#_Toc111230975)

[Table 4: Admin Data Dictionary - 23 -](#_Toc111230976)

[Table 5: User Master - 24 -](#_Toc111230977)

[Table 6: Property Master - 24 -](#_Toc111230978)

[Table 7: Book masters - 25 -](#_Toc111230979)

[Table 8: Test Case - 27 -](#_Toc111230980)

[Table 9: Revenue model - 40 -](#_Toc111230981)

# ABBREVIATIONS

KCC: Kantipur City Collage

PU: Purbanchal University

DFD: Data flow diagram

OS: Operating System

SQL: Structured Query language

UI: User Interface

UX: User Experience

IDE: Integrated Development Environment.

# Chapter 1: Introduction

## Introduction

As we, all know it is difficult to manage time for daily needs in this fast-moving world. In this case, there is a need for change in a technological field, there's an urgent must embrace and appreciate the power of innovative technology. So there is a vital situation to manage all the essential needs that are a shelter or home for survival. If we are not known to an individual in some cities and want to rent a house, room, apartment as well as shop then it's difficult to hunt out suitable aim time. Hence, there's a requirement to develop a “Gharbeti” App that will simplify the work for the rental managers and tenants so all their work is often done efficiently and effectively. It's also difficult to hunt out the renter on time, for the owner and property managers. This App will provide the whole knowledge about houses and apartment which is accessible for Rent. It will make it easy to hunt out the position of Houses, the need for rooms, and other facts by the renter. Using this app the Landlords even have the provision to post or update their property details whenever they want.

* GharBeti App is a fully static and effective app. It'll provide the knowledge to the tenants about the homes which are accessible for Rent. They can easily search for their needs using keywords like property type, location, etc.
* On the other side, the Landlord has the facility to post or modify their property details with admin approval.
* It can be helpful to easily upload the position, phone number, Expected rent, and No. of rooms, Facilities, and other information by the

**Modules used in this project**

This project has two major modules that are:

**Landlord:**

* Registration by the landlord: First, the owner of the house should register their land with their name, location, contact No, expected rent, No. of rooms, Facilities, and other information.
* Login by Landlord: the owner uses the app by login into the app with their login credentials.
* Add Tenants: New Tenant is added by the landlord by entering Tenant’s personal details with a verification ID.
* Update Availability: The owner can moderate or update no. Of rooms, areas,
* Generate the rent invoice: Monthly bill is generated for the renters.
* Payment status: View payment status and easily detects which tenant has paid, not paid, and has balance.

**Tenants:**

Registration by the tenants: First, the tenant should register themselves by their names and private details like id, contact,

Login by the tenants: The Tenant uses the app by login into the app with their login credentials.

Search property location: they will search for feasible lands and placement in their budget.

Handling Payment: Payment details can be viewed, the status of payment, and submit

## Problem Statement

In Nepal, there are few online rental management systems for those who want to rent and get a lot of trouble finding a place for rent even if the place is available.

There is no properly allocated place and the system is not easily available according to their user interest. The rental management system is almost done through the manual system.

The administrative system does not have the facility to make a rental Management system online, most time the work is done through an illegal intermediate person without awareness of the administrative, and this makes it more complex and more costly to find room for the customer. This leads customers to face more trouble, cost, dishonesty, and time wastage. The problem found in the current system:

* The complexity of finding the property is not easy and more tedious.
* Need extra money to find the property.
* The user cannot get information about home when they need it.
* There is too time-consuming to find the property.
* An emergency repair is required when something in the rental unit has broken and the health or safety of the tenant is in danger or the building or property is at risk until repairs can make.

## Objectives

* To search/post properties (room, Flat, apartment).
* To digitize the monthly rental report.

## Significance of project

* It helps the owner to find the tenants easily.
* This app verifies both owner and the tenants before they contact each other by using the document and personal contact, which verifies them as real people.
* It helps the tenants to find the room easily without searching from one home to another.
* The tenant and owner are easily able to add, update, modify and delete their property details.
* We are able to contact the owner without the agent.

## Project Feature

Some of the useful and important features of “GharBeti” are-

* It has the ability to add & remove different rental details.
* Admin, landlord and tenant panel.
* Landlord adds, updates, and removes personal as well as rent detail.
* Tenant can edit profile and book the available properties.
* Admin is able to manage landlords, tenants, and properties.
* Admin and landlord can view the report

## Assignment of role and responsibility

We assigned the roles and responsibilities equally among all members of the group. There are various phases during the project such as analysis, coding, testing, debugging, documentation, etc. The coding phase itself involves frontend and backend development as well as data entry. As team members, we shared all the work and regularly communicated and helped each other for the successful completion of the project.

|  |  |  |  |
| --- | --- | --- | --- |
| **Role and responsibility** | **Bikash Gupta** | **Laxmi Gajurel** | **Pratik Subedi** |
| Documentation chapter1 |  |  |  |
| Chapter 2 |  |  |  |
| Chapter 3 |  |  |  |
| Chapter 4 |  |  |  |
| Chapter 5 |  |  |  |
| Chapter 6 |  |  |  |
| Sign up |  |  |  |
| Sign in |  |  |  |
| Type of room |  |  |  |
| Book room |  |  |  |
| User profile setting |  |  |  |
| Outlook |  |  |  |

## Project Documentation

Documentation is an important part of any software project. It informs both software developers and users. The documentation for our project was developed throughout the life cycle of the project. The documentation has been

[Chapter 1: Introduction](#_Toc111231001)

[1.1. Introduction](#_Toc111231002)

[1.2. Problem Statement](#_Toc111231003)

[1.3. Objectives](#_Toc111231004)

[1.4. Significance of project](#_Toc111231005)

[1.5. Project Feature](#_Toc111231006)

[1.6. Assignment of role and responsibility](#_Toc111231007)

[1.7. Project Documentation](#_Toc111231008)

[Chapter 2: Literature Survey](#_Toc111231009)

[2.1. Literature Review](#_Toc111231010)

[2.2. Existing System](#_Toc111231011)

[2.3. Analysis of issue](#_Toc111231012)

[2.4. Solution](#_Toc111231013)

[Chapter 3: System Analysis](#_Toc111231014)

[3.1. System Development model](#_Toc111231015)

[3.2. Requirement Specification](#_Toc111231016)

[3.2.1. Functional Requirement](#_Toc111231017)

[3.2.2. Non-Functional Requirement](#_Toc111231018)

[Hardware Interfaces](#_Toc111231019)

[Software Interface](#_Toc111231020)

[Hardware Requirements](#_Toc111231023)

Software Requirements

[3.3. Feasibility Study](#_Toc111231025)

[3.3.1. Operational Feasibility](#_Toc111231026)

[3.3.2. Technical Feasibility](#_Toc111231027)

[3.3.3. Economic Feasibility](#_Toc111231028)

[3.3.4. Legal and Contractual Feasibility](#_Toc111231029)

[3.3.5. Scheduled Feasibility](#_Toc111231030)

[Chapter 4: Design Specification and Implementation](#_Toc111231031)

[4.1. System Architecture](#_Toc111231032)

[4.2. Data Flow Diagram (DFD)](#_Toc111231033)

[4.2.1. Context Diagram](#_Toc111231034)

[4.2.2. DFD Level 1](#_Toc111231035)

[4.3. Use Case Diagram](#_Toc111231036)

[3.4. Activity diagram](#_Toc111231037)

[3.5. Database Diagram](#_Toc111231038)

[4.5.1. E-R Diagram](#_Toc111231039)

[4.5.2. Relational Data Structure](#_Toc111231040)

[4.5.3. Data Dictionary](#_Toc111231041)

[4.6. UI/UX Mechanism](#_Toc111231042)

[Chapter 5 Experimental Result and Analysis](#_Toc111231043)

[5.1. Programming platform](#_Toc111231044)

[5.2. Operating Environment](#_Toc111231045)

[5.3. Testing and De bugging](#_Toc111231046)

[5.3.1. Testing](#_Toc111231047)

[5.3. Experiment and Result Analysis](#_Toc111231048)

[5.3.1. Experiment](#_Toc111231049)

[5.3.2. Expected Result](#_Toc111231050)

[Chapter 6: Conclusion and Future Enhancement](#_Toc111231051)

[6.1. Conclusion](#_Toc111231052)

# Chapter 2: Literature Survey

## 2.1. Literature Review

A literature review is a text written by someone to consider the critical points of current knowledge including substantive findings as well as theoretical and methodological contributions to a particular topic. The main goals are to situate the current study within the body of literature and to provide context for the particular reader

**DalayDai**

It is an app for buying, selling, and renting houses, lands, and everything real estate. It is committed to posting and searching for a house or plot of land for Nepali people effortlessly.

**Feature:** Home, buy, rent, commercial, home loans, easy construction, search relevance properties, search properties according to budget range, type of property, search property according to address, town and property id etc.

**99ACRES**  
Created by the popular property search portal 99acres.com, this app offers the same user experience as the website, on the go. The app, which is free for Android users, allows you to browse properties for rent, along with high-quality pictures, videos, and maps. It is best known for the numerous options it provides, with listings of around 10 lakh properties across the country. It also promises instant contact between landlords and interested tenants through phone calls, texts, or email.

**Feature:** search cities, localities, project etc., shortlisted, activities, profile, home, types of properties, recommended properties, buy and sell properties etc.

**NOBROKER**This app stands for what everyone looking to rent wants to do: cut the broker out of the equation. It lets us find and rent a house without paying any brokerage. Most of us feel that having to pay a hefty brokerage fee is unfair. The No-Broker app resolves this by letting homeowners list their properties easily and then putting them in touch with potential tenants. We can contact the homeowner directly through the app after shortlisting a property.

**Feature:** Rent, Buy, Apartment type, Flat mate

**FLATCHAT**  
This app provides a platform where homeowners and potential tenants can share their location, find contacts in their vicinity and chat with them. It also allows you to sign up to find suitable flat mates to share a rental with. This can make the otherwise difficult process hassle-free since you can find people with the same budget and location preference as you, as well as similar hobbies and habits. So far, the app is only functional in Bengaluru, Mumbai, Pune, and Delhi.

**Feature:** Tenant, seeker, owner, chat, login through Facebook, mobile app

**NESTAWAY**  
This app’s appeal lies in the fact that it offers many fully furnished rental options with standardized amenities, and Nest Away takes responsibility for making sure that these are in working order. Once we shortlist a property, we can schedule a visit through the app, saving the trouble of calling and coordinating with owner or agent. The app stores important documents like rental agreements and receipts for easy access. It also allows us to book prefer.

**Feature:** Tenant, Owner, Room with their price,

## 2.2. Existing System

1. Dalay Dai

2. NoBrokers

3.99Acres

4. Flatchat

At present, the people of our country suffer a lot for want of accommodation

according to their demand. They've to run to and fro for their desired house to

buy/rent. They have to go through every corner of society to get information about the

house/land that is available. Similarly, a landlord/house owner also has to suffer

sometime when he wants to rent/sell his properties. He has to expose the subject

people to people to get his desired client. To solve these types of problems, we're

going to introduce HOUSE RENTAL system.

At present, the people of our country suffer a lot for want of accommodation

according to their demand. They've to run to and fro for their desired house to

buy/rent. They have to go through every corner of society to get information about the

house/land that is available. Similarly, a landlord/house owner also has to suffer

sometime when he wants to rent/sell his properties. He has to expose the subject

people to people to get his desired client. To solve these types of problems, we're

going to introduce HOUSE RENTAL system.

At present, the people of our country suffer a lot for want of accommodation

according to their demand. They've to run to and fro for their desired house to

buy/rent. They have to go through every corner of society to get information about the

house/land that is available. Similarly, a landlord/house owner also has to suffer

sometime when he wants to rent/sell his properties. He has to expose the subject

people to people to get his desired client. To solve these types of problems, we're

going to introduce HOUSE RENTAL system

At present, the people of our country suffer a lot for want of accommodation

according to their demand. They've to run to and fro for their desired house to

buy/rent. They have to go through every corner of society to get information about the

house/land that is available. Similarly, a landlord/house owner also has to suffer

sometime when he wants to rent/sell his properties. He has to expose the subject

people to people to get his desired client. To solve these types of problems, we're

going to introduce HOUSE RENTAL system

At present, the people of our country suffer a lot for want of accommodation

according to their demand. They've to run to and fro for their desired house to

buy/rent. They have to go through every corner of society to get information about the

house/land that is available. Similarly, a landlord/house owner also has to suffer

sometime when he wants to rent/sell his properties. He has to expose the subject

people to people to get his desired client. To solve these types of problems, we're

going to introduce HOUSE RENTAL system

## 2.3. Analysis of issue

. The problem found in the current system:

* The complexity of finding the property is not easy and more tedious.
* Need extra money to find the property.
* The user cannot get information about home when they need it.
* Digitization of monthly rental report is not available.

## 2.4. Solution

“GharBeti” app is the solution for the above issues as:

* Tenant can search for the property easily.
* There is no brokerage charge for finding property.
* User can view the property detail.
* Monthly rental report is digitized.

# Chapter 3: System Analysis

## 3.1. System Development model

The system development methodology is a technique that is used to show how the proposed system will be developed. In this case, the methodology used will be a waterfall model.

**Waterfall Model**

It is comprised of the stages that the developer will use when developing the systems. It is a sequential model hence the name is a waterfall. The developer has to finish one stage before going to the next one. It comprises the feasibility study, analysis phase, design phase, coding phase, testing phase, implementation phase, and finally maintenance phase. It is a simple and easy model to use and understand. With waterfall development-based methodologies, the analysts and users proceed sequentially from one phase to the next. The deliverables from each phase are voluminous and are presented to the project sponsor for approval as the project moves from phase to phase. Once the sponsor approves the phase, it ends and the next phase begins.



Figure 1: Waterfall model

## 3.2. Requirement Specification

Requirement specification involved defining consumer needs and objectives in the context of planned consumer use environments and identifying system characteristics to determine requirements for system functions.

### 3.2.1. Functional Requirement

This is a necessary task, action, or activity that was accomplished. The proposed system is able to:

1. Allow admin to add a property, tenants, and landlord details.
2. Allow the user to delete property, tenants, and landlord details.
3. Allow the admin to search data in the database.
4. Allow admin to edit data in the database.

Table 1: Functional Table

|  |  |
| --- | --- |
| **Actor** | **Function** |
| Landlord | * Advertise the property * Adding information about the property |
| Tenant | * Search the property * Select the property they want * Register to rent the property |
| Admin | * Edit data in the database * Delete data in the database * Update data in the database |

1. **Performance Requirements**

The system should respond within a short period. It depends on the performance of the hardware environment such as RAM and processor.

1. **Process Requirements**

The system should document expectations, targets, and specifications for business processes. They may be collected from multiple groups of stakeholders such as business units, customers, internal customers, users, and subject matter experts.

1. **Input Related Requirements**

The system should set all input bundles required to produce at least a given level of outputs.

1. **Output related Requirements**

The system should predominantly adopt performance-based requirements to define the project scope.

1. **Storage related Requirements**

The system should include any necessary periodic preservation or condition checks.

### 3.2.2. Non-Functional Requirement

The official definition for a non-functional requirement specifies how the system should behave: “A non-functional requirement is a statement of how a system must behave; it is a constraint upon the system behavior.” Non-functional requirements specify all the remaining requirements not covered by the functional requirements. They specify criteria that judge the operation of a system, rather than specific behaviors.

* While user use the system, they feel friendly.
* System is easy to use. no need technical knowledge.
* As per the user’s role, some user are limited to use all feature. So, that they feel safe about their information.

### Hardware Requirements

1. Processor:- Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz
2. Memory:- 4.00 GB RAM and 1TB+128GB Hard drive and SSD
3. Visual Display Unit: window, mac and any type of mobile
4. Input units: mobile screen, keyboard and mouse

### Software Requirements

1. Operating System- Windows, Mac, Android, etc.
2. Microsoft Office PowerPoint- Used during Presentation
3. Microsoft Office- Used during documentation
4. Visual Studio- Used during coding and editing code.
5. Android studio-
6. GitHub-
7. Simulator
8. Draw.io-

## Feasibility Study

The feasibility study is the preliminary study that determines whether a proposed system project is financially, technically, and operationally. The feasibility study is essential to evaluate the cost and benefits of the new system. The alternative analysis is usually included as part of the feasibility study and identifies viable alternatives for the system design and development.

### Operational Feasibility

The system to be developed will provide accurate, active, secure service and decreases the labor of workers and it is not limited to particular groups or body. The system will easily be operational, as it does not affect the existing organizational structure and support the current system. Therefore, the system will be operationally feasible.

### Technical Feasibility

The system is to be developed by using technologically system development technologies such as JavaScript, Flutter, and Strapi without any problems and the group members have enough capability to develop the project. Our focus is to develop a well-organized dynamic website that is technically effective for managing the Online Trade interaction system. Therefore, it can be concluded that the system is technically feasible.

### Economic Feasibility

The system to be developed is economically feasible and the benefit is outweighing the cost. Since this project is already computerized, the existing system and more advanced than the current system and more advance than the current system reducing and changing the labor force to the computerized system. Reduces the cost of the material used.

### Legal and Contractual Feasibility

The system is free from any legal and contractual risks

### Scheduled Feasibility

The project will be completed according to a planned schedule. The planned schedule is displayed in the form of a Gantt chart.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S. N | Tasks | Duration (in Week)-Starts From (13th Apr2022) | | | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| 1 | Concept submission |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Requirement Gathering |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Research and analysis |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | System Design |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Coding |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Testing and debugging |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Documentation |  |  |  |  |  |  |  |  |  |  |  |  |  |

Table 2: Gantt Chart

# Chapter 4: Design Specification and Implementation

## 4.1. System Architecture

A system architecture is the conceptual model that defines the structure, behavior and more views of a system. It is the structure of an IT system. The system architecture is based of the Dart and flutter architecture.



Figure 2: System Architecture

## 4.2. Data Flow Diagram (DFD)

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling the aspects of the operation. They are also a preliminary step used to build an outline of the system that can be built later. DFDs can also be used to simulate data processing (structured design) for the visualization of data processing. A DFD illustrates what sort of information is input to and output from.

### 4.2.1. Context Diagram



Figure 3: Context Diagram

### 4.2.2. DFD Level 1



Figure 4: Level 1 DFD

## 4.3. Use Case Diagram



Figure 5: Use Case Diagram

## Activity diagram

It shows the control flow from one activities to another. Activity diagram is another important diagram to describe dynamic behavior. Activity diagram consists of activities, links, relationships etc. It models all types of flows like parallel, single, concurrent etc. Activity diagram describes the flow control from one activity to another without any messages. These diagrams are used to model high-level view of business requirement.

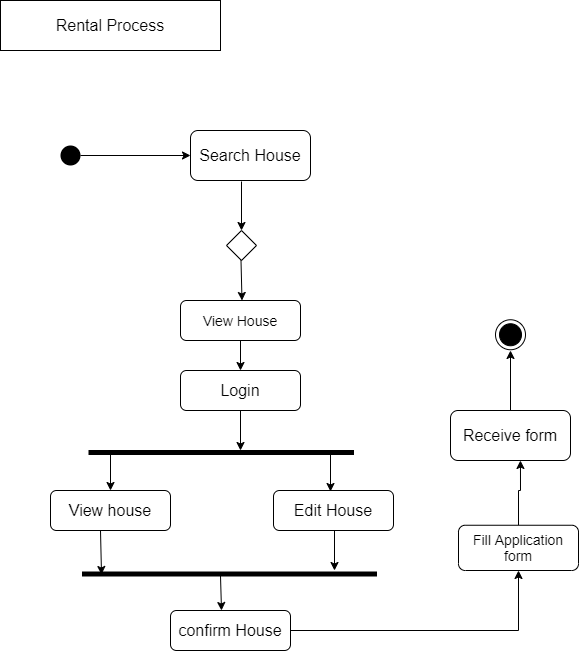


Figure 6: Activity diagram

## Database Diagram

The database stores most of the information required by the system. The database will be maintain by Strapi backend, which uses SQLite3 as a relational database management system (RDBMS) that is based upon the Structured Query Language (SQL).

### 4.5.1. E-R Diagram

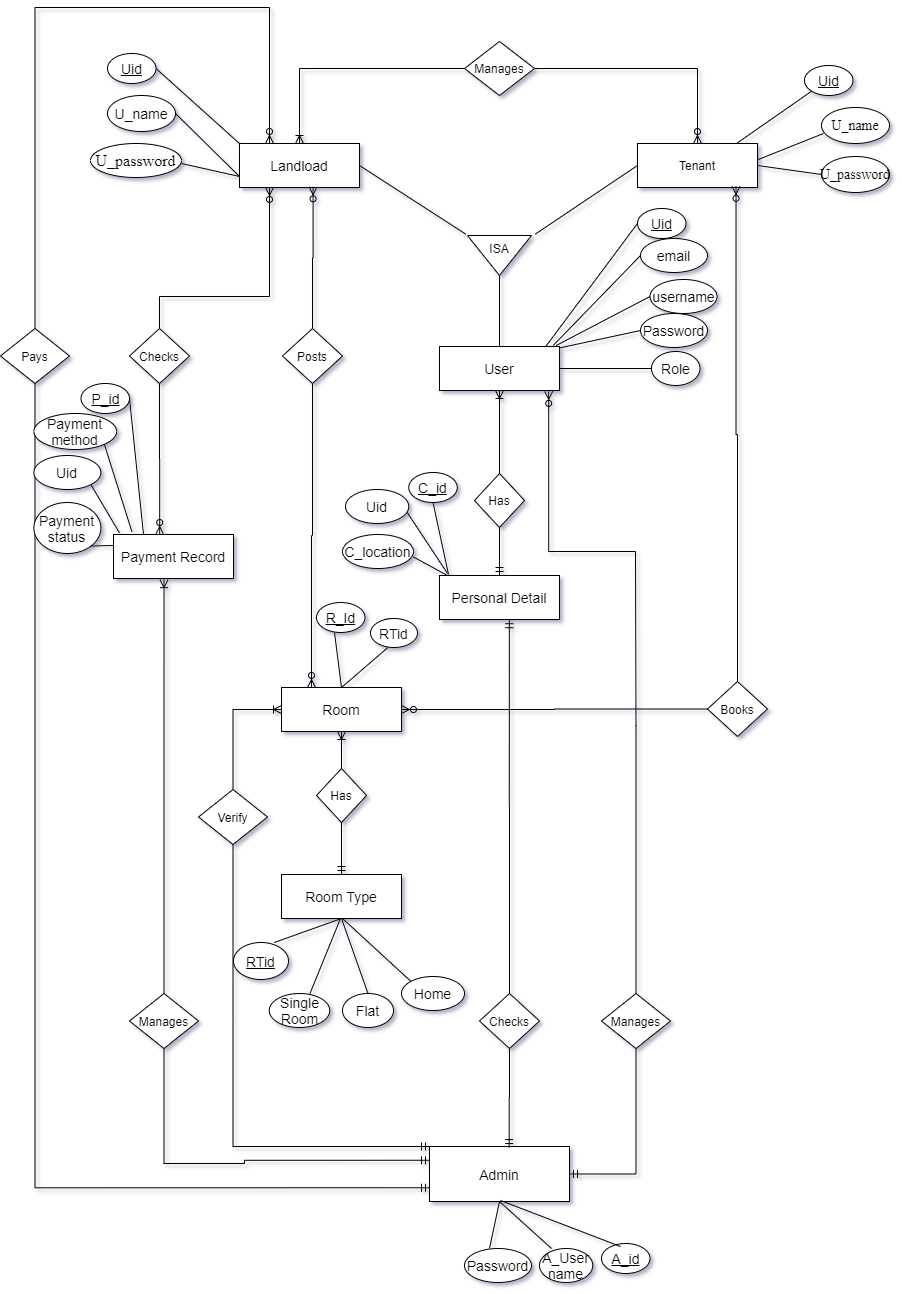


Figure 7: ER Diagram

### 4.5.2. Relational Data Structure

Table 3: Relational Data Structure

|  |  |
| --- | --- |
| **Work** | **Description** |
| Admin | Store information for admin |
| Landlord | Store information for landlord |
| Tenant | Store information for tenant |
| Property | Detail of property |
| Property Type | Detail of each property type |

### 4.5.3. Data Dictionary

The data dictionary is useful in the case of development.

**Table Name:** Admin

**Primary Key:** Admin ID

**Description:** To store the details of admin

Table 4: Admin Data Dictionary

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial No.** | **Field** | **Datatype** | **Description** |
| 1. | Admin ID | Integer | Primary Key |
| 2. | Name | Varchar |  |
| 3. | Email ID | Varchar |  |
| 4. | Contact No. | Integer | Multiple contacts can exist for a single admin |
| 5. | Address | Varchar |  |

**Table Name:** User Master

**Primary Key:** User ID

**Description:** To store the detail of Users

Table 5: User Master

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial No.** | **Field** | **Datatype** | **Description** |
| 1. | User ID | Integer | Primary Key |
| 2. | Name | Varchar |  |
| 3. | Email ID | Varchar |  |
| 4. | Contact No. | Integer | Multiple contacts can exist for a single admin |
| 5. | Address | Varchar |  |

**Table Name:** Property Master

**Primary Key:** Property ID

**Description:** To store the detail of Property

Table 6: Property Master

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial No.** | **Field** | **Datatype** | **Description** |
| 1. | Property ID | Integer | Primary Key |
| 2. | Owner ID | Integer | Foreign key to user master |
| 3. | Location | Varchar |  |
| 4. | Status | Varchar | Available, rented, etc. |
| 5. | Total value | Float |  |

**Table Name:** Book Master

**Primary Key:** Book ID

**Description:** To store the detail of Booking

Table 7: Book masters

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial No.** | **Field** | **Datatype** | **Description** |
| 1. | Book ID | Integer | Primary Key |
| 2. | Property ID | Integer | Foreign key to Property Master. |
| 3. | Owner ID | Integer | Foreign key to Owner Master. |
| 5. | User ID | Integer | Foreign key to User Master. |

## 4.6. UI/UX Mechanism

The development is based on a web application, so the UI is in the form of a web. To view different pages on the window, Linux or Mac there will be a navigation menu that offers drop-down menus. To provide a smooth user interface, we make use of buttons, form, graphics etc.

## Chapter 5 Experimental Result and Analysis

## 5.1. Programming platform

The project is developed based on Application Development techniques. Data is stored in databases. The application uses the following platforms and technologies

* Strapi
* Sqlite3
* Flutter
* Dart

We also used following tools or software for development of the application.

* Microsoft Word & PowerPoint (Documentation & Presentation)
* Visual Studio Code (IDE)
* Mobile app (Running, testing and debugging)
* Draw.io (Drawing & Figures)

## 5.2. Operating Environment

The “GharBeti” is internet based so it will run in any Operating System with internet access through Mobile Application.

* API
* Sqlite3 server

## 5.3. Testing and Debugging

### 5.3.1. Testing

Each component is tested independently, without other system components interfering. This process is concerned with finding errors. It is also concerned with validating that the system meets its functional & non-functional requirements.

Table 8: Test Case

|  |  |  |  |
| --- | --- | --- | --- |
| S.NO | **Test Case** | **Expected Results** | **Actual Results** |
| 1 | Sign in with empty admin name & password | Should display error message | Passed |
| 2 | Sign-in with invalid user name & password | Should display error message | Passed |
| 3 | Sign-in with valid user name & password | Should successfully Sign-in | Passed |
| 4 | Add a rental place with empty field | Should display error message | Passed |
| 5 | Add a rental place with Valid info | Should successfully add rental place | Passed |
| 6 | Landlord add place and personal detail with empty field | Should display error message | Passed |
| 7 | Tenant reserve rental place | Should successfully Reserved | Passed |
| 8 | Tenant reserve rental place with empty field | Should display error message | Passed |
| 9 | In report, landlord provide empty field | Should display error message | Passed |
| 10 | Fill report form by landlord | Should successfully show the report | Passed |

## 5.3. Experiment and Result Analysis

### 5.3.1. Experiment

The coding and testing process was completed successfully and the software behaves as desired. All the objectives of the project were met and we were able to provide all necessary features in the system.

Due to external issues, there was some difference with the time schedule specified in the analysis phase. The project completion Gantt chart is provided as an appendix in appendix 2

### 5.3.2. Expected Result

* Application will be a service, which will connect Renters and Landlords and vice versa.
* It will save the physical hard work and invaluable time to find room/flat/shop.
* It will save Resources for search the rooms/house/flats/shop.
* It will be an Open Source Application which can be freely Install in android smart Phones.
* Application will make easy to find rooms/flats/houses/shops for the tenants and upload the location and other information by landlord/room-master.

# Chapter 6: Conclusion and Future Enhancement

## 6.1. Conclusion

Online Property rental business has emerged with a new possibility compared to the experience where every activity concerning rental business was limited to a physical location only. Even though the physical search for houses has not been totally eradicated; the nature of functions and how these functions are achieved has been reshaped by the power of internet. Nowadays, renters can reserve houses online once the customer is a registered member of the application. The app-based property rental system has offered an advantage to both landlords as well as the tenants efficiently and effectively just with the click of a button.

## 6.2. Limitations

* Lack of insurance module
* No subscription feature.

## 6.3. Future Enhancement

* Add the subscription feature
* Make more secure database
* Make the report digitize

# Reference

[**https://pdfcoffee.com/a-project-on-online-house-rental-management-pdf-free.html**](https://pdfcoffee.com/a-project-on-online-house-rental-management-pdf-free.html)

[**https://www.slideshare.net/FNISHA/apartment-management-system-project**](https://www.slideshare.net/FNISHA/apartment-management-system-project)

## Appendixes

## Appendix1: Screenshot

## Appendix2: Project Completion Gantt chart

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S. N | Tasks | Duration (in Week)-Starts From (20th Aug2022) | | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| 1 | Concept submission |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Resource collection |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Research and analysis |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | System Design |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Coding |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Testing and debugging |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Documentation |  |  |  |  |  |  |  |  |  |  |  |  |  |

Figure 8: Project Completion Gantt Chart

## Appendix3: User Manual

### General Information

General Information section explains in general terms the “GharBeti” application overview and sections of the user manual.

* 1. **Application Overview**
* GharBeti App is fully static and effective app. It'll provide the knowledge to the tenants about the homes which are accessible for Rent. They can easily search for their needs using keywords like property type, location, etc.
* On the other side, the Landlord has the facility to post or modify their property details with admin approval.
* It can be helpful to easily upload the position, phone number, Expected rent, No. of rooms, Facilities and other information
  1. **Organization of the Manual**

The user manual consists of the following four sections

1. **General Information**

This section explains about the hardware and software requirements for accessing “GharBeti” window, Mac and Linux application overview and the sections for the user manual

1. **System Information**

This section explains about the hardware and software

1. **How to download the application**

This section explains the options available to download the GharBeti application on mobile.

1. **Using the application**

This section provides a detailed description of the functionalities of the GharBeti application

1. **System Summary**

System Summary section explain about the hardware and software requirements for accessing GharBeti application and user access levels.

* 1. **Hardware and Software Requirements**

Requires smart phones, iPhone with operating system (OS). To download and use the functionalities of GharBeti app, require an internet connection in any os.

* 1. **User Access Levels**

There are two types of roles in GharBeti application

1. **Guest User**

Guest user is one who access and uses the GharBeti application without registering in the application.

1. **Register User**

Registered user is one who registers himself/herself in the GharBeti application by filling the registration form. Registered user can edit profile, book room etc.

1. **How to download the application**

GharBeti application can be download through google play store.

* 1. **Google Play Store/ AppStore**

Open play store application on mobile by clicking of the play store icon

Click on the google Play text at the top and enter the text “GharBeti in the search bar.

Click on the GharBeti application icon from the search result. Now below page will open which asks to install the application.

Click on the “INSTALL” button to install the application

1. **Guest User –PRIVILEGES**

Following features can be available by the guest user.

* Landlord able to view the report the tenant.
* Landlord able to add, update and remove the rental house detail
* Tenant able to view the detail of rental house
* Tenant and landlord able to view the status of the rental place.
* Can view the notification
* Tenant able to view the price of the room.
* Tenant view the money they remain to pay.
  1. **View Latest Items book(Guest User)**
* As soon as GharBeti application is launched, dashboard will be displayed which shows the main features of the application
* Latest book displayed in the landlord side and book by the tenant.
  1. **View List of available rental place**
* This can be seen in the tenant side, which is sat by the landlord.
* On click of the “list of rental place” we need to select the available room type and price.
  1. **Rental Price**
* Price of the is fixed according to the type of rental place
* Price tag is attached with the rental place detail

1. **Registered Users –PRIVILEGES**

Following feature, can available by register user.

* Add/edit/remove landlord and tenant profile
* Add/edit/remove rental place by landlord
* Able to book room by tenant.
  1. **New User Registration**

Registered user can avail more benefits than a guest user. Hence, it is recommended that  
users register themselves to access the additional services.  
A user can register in GharBeti application using the "Sign In" option available in  
Home Navigation Drawer. If you are a new user, select “Sign Up” to register. Fill all the  
required details and Submit. On successful validation user will be registered and a  
message and notification will be sent.

* 1. **Sign In**

To Sign In in to the application click on the “Sign IN” option from the Navigation Drawer in the Home Page. Enter username and password to login. On successful login, the user is redirected to home page screen.

* 1. **Add or Post the state of rental place**

In this, the property owner updates the status of rental place.

* 1. **View Latest rented Place (Register User)**

Latest rented place can be viewed from the home page.

On click of the interested rented place page will be displayed with the details of the rented place

**How to contact the person who posted the rental place status**

In the rental place, status detail can be edit by the landlord.

Click on the icon to know the detail of rental place and know the rental place available or not.

* 1. **My flat/room/department**

First login into the system as a tenant.

Click on the type of room

Choose the type of room as per the tenant requirement.

**How to modify a rental place**

To modify the rental place detail, click on the edit button highlight to enable the entire field and do the necessary changes

Click on submit/ update button to submit the change.

* 1. **Notification**

Notification can be shown every time when the actor involve to do every activities

* 1. **How to Change Profile**

User can access the change profile feature from the home page.

On click of the “Change Profile”, the form for updating of profile is displayed.

User need to click on the edit button to enable for editing.

After editing click on the “edit” button to reflect the changes.

### Appendix4: Revenue Model

1. **A revenue model is how a business makes money**



Figure 9: Revenue Model

1. **Customer pay for a solution to problem**

****

Figure 10: Customer pay for a solution

1. **Revenue model is an exchange value**

****

**Types of revenue model**

Table 9: Revenue model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Revenue Model** | **Description** | **Pros** | **Cons** | **Example** |
| Transaction revenue | One-time sales of goods or services. | Can have low barrier to trial | Need to continuously generate new sales | American Eagle |
| Project revenue | One-time project | Can have large payouts.  Can have low startup costs. | Scalability issues, May have long sales cycle, May require large upfront investment | Construction Company |
| Recurring revenue | Customer pays for access to a product or service in installment (usually monthly or yearly) | Don’t have to resell customer for each revenue installment | May require large upfront investment, May have long sales cycle. | Dropbox, Salesforce |
| freemium | Service provided to customer. Customer pays for time or expertise | May not require large upfront investment. | Scalability issues. | Law firm, Supercuts |
| Freemium | Offer subset of features for free. Try to convert free customers to paying customers. | Large market of potential customers. Potential secondary revenue source | Hard to convert to paid. Hard to determine paid/unpaid | Dropbox |
| Advertisements | Sell ad space | Large market of potential customers. Potential secondary revenue source. | Need a lot of eyeball to make value. Can detract from user experience. | Google |
| Marketplace, platform | Platform or marketplace to connect buyers to producers. | May not require large upfront investment. Sometimes are winner-take all markets. | Two sided market | Etsy |