A Project Report

On

**Kura-Kani**

Submitted in partial fulfillment of the requirement of

Project-VI (BCA379CO)

Of

Bachelor of Computer Application

**Submitted To:**

****

Purbanchal University

Biratnagar, Nepal

**Submitted By:**

Bikalpa Sangat (362526)

Grishma Shrestha (362528)

Kilika Shrestha (362529)

**KANTIPUR CITY COLLEGE**

Putalisadak, Kathmandu

A Project Report

On

**Kura-Kani**

Submitted in partial fulfillment of the requirement of

Project- VI (BCA379CO)

Of

Bachelor of Computer Application

****

**Submitted To:**

Purbanchal University

Biratnagar, Nepal

**Submitted By:**

Bikalpa Sangat (362526)

Grishma Shrestha (362528)

Kilika Shrestha (362529)

**Project Supervisor**

Mr. Rubim Shrestha

Ass. Lecturer

**KANTIPUR CITY COLLEGE**

Putalisadak, Kathmandu

**TOPIC APPROVAL SHEET**

It is hereby informed that the topic selected by Bikalpa Sangat (362526), Grishma Shrestha (362528), Kilika Shrestha (362529) of Bachelors of Computer Application VI semester 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 and supervisor for the above-mentioned students.

Topic Approved: Kura-Kani

**--------------------------------**

**Rubim Shrestha**

**Assistant Professor**

Project Supervisor

**CERTIFICATE FROM SUPERVISOR**

This is to certify that the project entitled “Kura-Kani” submitted by Bikalpa Sangat (362526), Grishma Shrestha (362528) and Kilika Shrestha (362529) to the Department of Information Technology, School of Science and Technology at Kantipur City College, Kathmandu, Nepal towards the requirement for Project-VI (BCA) of is an original work carried out by them under my supervision and guidance.

Signature:

--------------------

# ACKNOWLEDGEMENTS

The project members would like to express the deepest appreciations to our project supervisor, Mr. Rubim Shrestha, who has the attitude and the substances of genius: he continually and convincingly conveyed a spirit of adventure in regard of research, and an excitement in regard to guiding to study and prepared this project. Without his guidance’s and persistence help this project would not have been possible.

The project members would like to thank our Network Programming lecturer Ramesh Parajuli whose work demonstrated to the project members about the meaning and importance of socket programming in making a real time system. In addition, the project members also thank instructor Saroj Panday of Kantipur City College who introduced the project members to Web Development and whose enthusiasm for the “underlying characteristics of software development” had lasting effect.

The project members thank the Purbanchal University for permission to include format of the project.

The project member would like to express sincere gratitude to the selection committee for considering correct subject for the project.

# ABSTRACT

The project entitled “**Kura-Kani**” is a chat application, for users to instantly communicate with each other. This project created a web based application for users to instantaneously communicate with one another from any place at real time. The most important part of building a good chat application is to focus on the real time data flow on web. This project spends a good amount of research to find the most appropriate technology for delivering the data flow, which is REST API for account management, Web Socket from socket.io library for text chat and react-notify as real time push notification. We have also used varieties of cloud computing services for different purposes as well from hosting images of users in Cloudinary to cloud database as well. The cloud based database and single threaded backend system written in Nodejs is capable of handling thousands of users simultaneously. With a proper loaded balancing the system can be scaled up for millions of users.

Contents

[ACKNOWLEDGEMENTS 5](#_Toc113091205)

[ABSTRACT 6](#_Toc113091206)

[LIST OF FIGURES 9](#_Toc113091207)

[LIST OF TABLE 10](#_Toc113091208)

[Chapter 1: INTRODUCTION 1](#_Toc113091209)

[1.1 Introduction 1](#_Toc113091210)

[1.2 Problem Statement 1](#_Toc113091211)

[1.3 Objectives 2](#_Toc113091212)

[1.4 Features 2](#_Toc113091213)

[1.5 Limitations 2](#_Toc113091214)

[1.6 Assignment of the role and responsibilities 3](#_Toc113091215)

[Chapter: 2 Existing System Overview 4](#_Toc113091216)

[2.1 Viber 4](#_Toc113091217)

[2.1.1 Introduction 4](#_Toc113091218)

[2.1.2 Pros. 4](#_Toc113091219)

[2.1.2 Cons. 4](#_Toc113091220)

[2.2 WhatsApp 4](#_Toc113091221)

[2.2.1 Introduction 4](#_Toc113091222)

[2.1.2 Pros. 5](#_Toc113091223)

[2.1.2 Cons. 5](#_Toc113091224)

[CHAPTER 3: SYSTEM ANALYSIS 6](#_Toc113091225)

[3.1 System Development Model 6](#_Toc113091226)

[3.1.1 Communication Phase 7](#_Toc113091227)

[3.1.2 Planning Phase 7](#_Toc113091228)

[3.1.3 Modeling Phase 7](#_Toc113091229)

[3.1.4 Construction Phase 7](#_Toc113091230)

[3.1.5 Deployment Phase 8](#_Toc113091231)

[3.2 Requirement Specification 8](#_Toc113091232)

[3.2.1 Functional Requirement 8](#_Toc113091233)

[3.2.2 Non-functional Requirement 8](#_Toc113091234)

[3.3 Feasibility Study 8](#_Toc113091235)

[3.3.1 Technical Feasibility 8](#_Toc113091236)

[3.3.2 Economic Feasibility 9](#_Toc113091237)

[3.3.3 Resources and Time Feasibility 9](#_Toc113091238)

[3.3.4 Schedule Feasibility 9](#_Toc113091239)

[3.3 Gantt Chart 9](#_Toc113091240)

[Chapter 4: System Design 11](#_Toc113091241)

[4.1 System Architecture 11](#_Toc113091242)

[4.1.1 Block Diagram 11](#_Toc113091243)

[4.1.2 Context Diagram 12](#_Toc113091244)

[4.1.3 Use case Diagram 14](#_Toc113091245)

[4.1.4 Database Design 15](#_Toc113091246)

[Chapter 5: System Development and Implementation 16](#_Toc113091247)

[5.1 Programming platform 16](#_Toc113091248)

[5.2 Operating Environment 16](#_Toc113091249)

[5.2.1 Software Requirements: 16](#_Toc113091250)

[5.2.2 Hardware Specification: 16](#_Toc113091251)

[5.3 Testing and debugging 16](#_Toc113091252)

[5.4 Implementation and result analysis 17](#_Toc113091253)

[Chapter 6: Conclusion and Future Enhancement 18](#_Toc113091254)

[6.1 Conclusion 18](#_Toc113091255)

[6.2 Limitations 18](#_Toc113091256)

[6.3 Future Enhancement 18](#_Toc113091257)

[REFERENCES 19](#_Toc113091258)

# LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| **S.N** | **Name of figures** | **Page no.** |
| 1 | Fig1 : Spiral Model | 6 |
| 2 | Fig 2: Gantt Chart | 10 |
| 3 | Fig 3: Block Diagram of System | 11 |
| 4 | Fig: Context Level (DFD-0) Diagram of System. | 12, 13 |

# LIST OF TABLE

|  |  |  |
| --- | --- | --- |
| **S.N** | **Name of tables** | **Page no.** |
| 1 | Assignment Of the Role and Responsibilities | 3 |
| 2 | Gantt Chart | 9 |
| 3 | Data Dictionary | 15 |
|  |  |  |

# Chapter 1: INTRODUCTION

## Introduction

Teleconferencing or Chatting, is a method of using technology to bring people and ideas together despite of the geographical barriers. The technology has been available for years but the acceptance it was quite recent. Our project is an example of a real time chat system.

Kura-Kani is an online chat application build using React18 on the Front end and ExpressJs in the backend. It is a real-time chat system that uses web sockets from socket.io package. The main purpose of the software is to provide users with an instant messaging tool that has the ability to handle thousands of users simultaneously when needed and can be easily done.

## Problem Statement

Chat rooms have become a popular way to support a forum for n-way conversation or discussion among a set of people with interest in a common topic. Chat applications range from simple, text-based ones to entire virtual worlds with exotic graphics. In this project we were required to implement a simple text-based chat client/server application.

Email, newsgroup and messaging applications provide means for communication among people but these are one-way mechanisms and they do not provide an easy way to carry on a real-time conversation or discussion with people involved. Chat room extends the one-way messaging concept to accommodate multi-way communication among a set of people.

## Objectives

The objectives of this project are given below:

* To provide user with instant messaging tool that has the ability to handle a lot of users simultaneously in real time.
* Learn to work with socket.io for real time communication.
* Define and implement a “chat” protocol for interaction between a chat server and a chat client

## 1.4 Features

The features of the projects are as follows:

* Simple Login and registration
* Cloud backup and recovery
* Data security
* Real time messaging
* Push Notification
* Group Chat

## 1.5 Limitations

* No audio or video calling features as for now.
* No android/ iOS app.
* Cannot send files (images, videos, documents).

## 1.6 Assignment of the role and responsibilities

|  |  |  |
| --- | --- | --- |
| **Group Member** | **Symbol no.** | **Responsibilities** |
| sBikalpa Sangat |  | Overall contribution on frontend socket implementation, Backend and database development |
| Kilika Shrestha |  | Overall contribution on–Boilerplate and backend API development |
| Grishma Shrestha | 362528 | Overall contribution on Frontend Development documentations, Testing –black box testing and presentation. |

# Chapter: 2 Existing System Overview

## 2.1 Viber

### 2.1.1 Introduction

Viber, or Rakuten Viber, is a cross-platform voice over IP and instant messaging software application owned by Japanese multinational company Rakuten, provided as freeware for the Google Android, iOS, and Microsoft Windows, Apple MacOS and Linux platforms. It Provides Unlimited free voice and video calls and text messages to other Viber users. Group text messages. Cheap calling to landline and mobile numbers.

### 2.1.2 Pros.

* + Free VOIP Calls
  + Sending Audio and video messages
  + Sending images and files

### 2.1.3 Cons.

* + Chat cannot done from only mobile app or desktop based app. Users cannot chat from their website

## 2.2 WhatsApp

### 2.2.1 Introduction

**WhatsApp Messenger**, or simply **WhatsApp**, is an internationally available freeware, cross-platform centralized instant messaging (IM) and voice-over-IP (VoIP) service owned by American company Meta Platforms (formerly Facebook). It allows users to send text and voice messages, make voice and video calls, and share images, documents, user locations, and other content. WhatsApp's client application runs on mobile devices, and can be accessed from computers. The service requires a cellular mobile telephone number to sign up. In January 2018, WhatsApp released a standalone business app called WhatsApp Business which can communicate with the standard WhatsApp client.

### 2.1.2 Pros.

* + Free audio and video call
  + Sending Audio messages, video messages, images and files.

### Cons.

* Chat history can be backed up only to google drive.

# CHAPTER 3: SYSTEM ANALYSIS

## 3.1 System Development Model

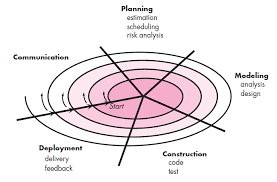


Fig 1: Spiral Model

For this project we have used spiral model of software development because the spiral model was primarily designed to help a project to adopt to changes in requirements quickly. So, the main aim of the Spiral Model is to facilitate quick project completion. To accomplish this task flexibility is required. Flexibility is achieved by fitting the process the project to the project, removing activities that may not be essential for a specific project. Also, anything that is wastage of time and effort is avoided. Since there were three of us working in the project simultaneously, there needed to be proper communication between the team members as well. Using spiral methods results in each step of the project contributing to the final outcome by providing a result that can be functionally used at every stage of completion. System project management helps in finding the source of the problem quickly through frequent testing and feedback. That’s why we decided to use spiral method to develop this project.

### 3.1.1 Communication Phase

Under this phase, all kinds of information have been gathered to successfully make the project of the desired specifications. It includes:

* Informal Meetings
* Personal Discussions

### 3.1.2 Planning Phase

The planning activity encompasses as set of management and technical practices that enable the software team to define a road map as it travels toward its strategic goal and tactical objectives. It includes the issues like:

* Clarification of the reasons for developing the system.
* Identification of the functionalities to be built.
* Proper scheduling of the system.

### 3.1.3 Modeling Phase

System modeling is an important element of the system engineering process. Under this phase, models are developed to gain a better understanding of the actual entity to be built.

### 3.1.4 Construction Phase

The construction activity encompasses a set of coding and testing tasks that lead to operational software that is ready for delivery to the customer or end-user.

* Coding
* Testing

### 3.1.5 Deployment Phase

This phase encompasses following activities:

* Delivery of Software, Support For User and Feedback

## 3.2 Requirement Specification

### 3.2.1 Functional Requirement

* This system must be compatible with mobile devices which supports different web browser.
* This system should allow the user send and receive real time messages and notifications.

### 3.2.2 Non-functional Requirement

* Smooth and proper layout to make it simple yet attractive for the end user.
* The system should provide robust authentication.
* The system performance should not be heavy on browser.
* The system should offer a cloud backup of chats

## 3.3 Feasibility Study

Various factors that dictate whether the project can be performed successfully. It also determines whether those proposed requirements and solutions are feasible in software or system.

### 3.3.1 Technical Feasibility

Kura-Kani is a real time web based chat application. The main technologies associated with this project are HTML5, CSS3, React, node with ExpressJs, SocketIo and mongoDB. Most of these tools are freely available and technical skills required are manageable. While in development phase website will be hosted locally and tested and later it will be hosted in a free and paid web hosting service gradually.

### 3.3.2 Economic Feasibility

Being a web-based application, Kura-Kani will have high hosting cost as the number of users grow. Since the system consist of many multimedia contents and data transfer, bandwidth required for the operation will be high. This system will follow freeware software standards. Bugs fixes and maintaining cost will have high cost too.

### 3.3.3 Resources and Time Feasibility

Resources required for Concept Two project includes: programming device (laptop), hosting space (some of which are freely available), programing tools (freely available) and programing individuals (Bishal and Aayush) and testers (Grishma). Time required for this project is about the actual time that has taken to design, code and test the system.

### 3.3.4 Schedule Feasibility

Schedule feasibility is defined as the probability of a project to be completed within its scheduled time limits, by a planned due date. If a project has a high probability to be completed on time, then its schedule feasibility is appraised as high. In many cases, a project will be unsuccessful if it takes longer than it was established some external environmental condition may change. Hence, a project can lose its benefits.

## 3.3 Gantt Chart

(From Jan 2022 – March 2022)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Weeks  Process | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| Idea presentation |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| Background study |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| Requirement analysis and gathering |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| Coding and testing |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| Documentation |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

Fig 2: Gantt chart

# Chapter 4: System Design

## 4.1 System Architecture

System architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system.

A system architecture can consist of system components and the sub-systems developed, that will work together to implement the overall system. There have been efforts to formalize language to describe system architecture. We have implemented context level diagram, data flow diagram, Use Case diagram, database schema in our project.

### 4.1.1 Block Diagram

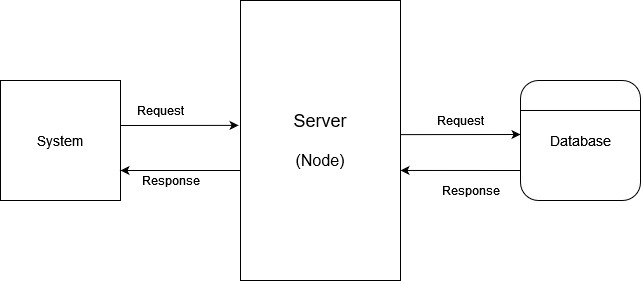


Fig 3: Block Diagram of System

### 4.1.2 Context Diagram

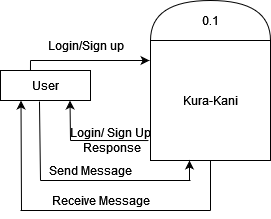


Fig: Context Level (DFD-0) Diagram of System.

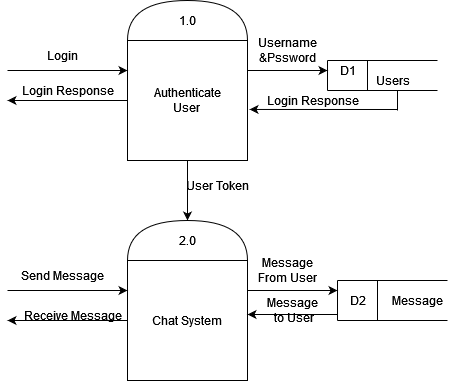


Fig: DFD Level-1 Diagram

### 4.1.3 Use case Diagram

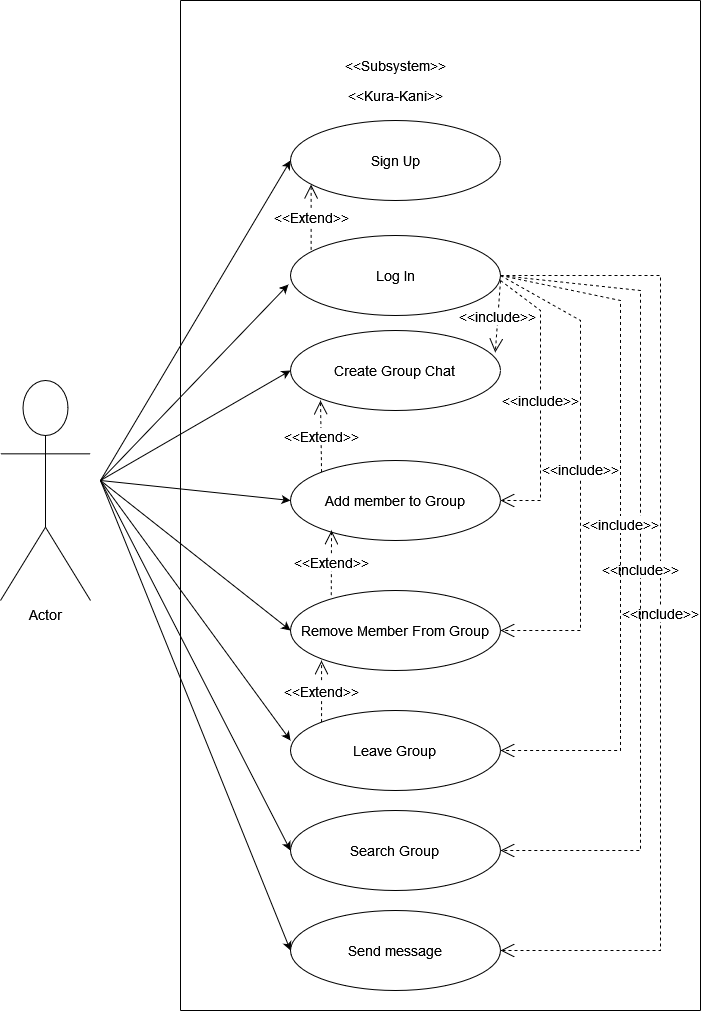


Fig: Use Case Diagram of System

### 4.1.4 Database Design

#### **4.1.4.1 Data Dictionary**

**Chat**

|  |  |
| --- | --- |
| **Field Name** | **Type** |
| Chat Name  Is Group Chat  Users  Latest Message  Group Admin  Timestamp | String  Boolean  Array of Objects  String  String  Date |

**Message**

|  |  |
| --- | --- |
| **Field Name** | **Type** |
| Sender  Content  Chat  Timestamp | String  String  String  Date |

**User**

|  |  |
| --- | --- |
| **Field Name** | **Type** |
| Name  Email  Password  Profile  Timestamp | String  String  String  String  Date |

# Chapter 5: System Development and Implementation

The one of the most important target of this project is the delivery information instantly.

Thus picking the right technologies in communicating has become the main focus of this project.

## 5.1 Programming platform

Every system has their own platform to run in the computer systems. Programming platform defines the rules and requirements of the system to work according to their design. We have used many combinations of languages. HTML 5, CSS 3, and React are used in client side programming and Node Js, Express Js are used as Server Side Programming with the combination of MongoDb database.

## 5.2 Operating Environment

### 5.2.1 Software Requirements:

* Web browsers like Google Chrome, Internet Explorer.

### 5.2.2 Hardware Specification:

* RAM: 512GB
* Processor: Intel core i3 3rd gen

## 5.3 Testing and debugging

Testing helps to verifying correctness of the system. Debugging also can be done in case of error found in the system. Basically, testing can be done some important module of the system design. Some important modules are: requirement analysis, interface design, algorithm design, implementation, and integration with other modules. Debugging is a technique of solving program related error, testing and code correction. Testing done at the level of debugging is different than other stage. Final module testing focused on correctness of the code to run the system, whereas testing during debugging is perform to locating errors.

## 5.4 Implementation and result analysis

This section focuses on the implementation of the function and the test for defects and necessary properties such as performance and reliability. These tests involve the execution of the application with test data to ensure that all requirement

# Chapter 6: Conclusion and Future Enhancement

## 6.1 Conclusion

There is always an opportunity for improvement in every product, and we attempted to adjust the design accordingly, while still keeping our constraints in mind. During the course of developing this application, we faced a slew of issues and learned how to solve them through study. With the end product, we think our idea was evident and well-presented.

## 6.2 Limitations

* No audio or video calling feature as of now
* No oath Authentication feature
* Cannot tag users in message

## 6.3 Future Enhancement

In future we may be extended to included features such as:

* Build android/iOS app
* Add audio/video calling feature using webRtc.
* Add feature to send files, images and short videos

# REFERENCES

* *Socket.io + ReactJS Tutorial | Learn Socket.io for Beginners*. (n.d.). Www.youtube.com. Retrieved August 26, 2022, from https://www.youtube.com/watch?v=djMy4QsPWiI
* *Host websites for FREE on HEROKU | 2020*. (n.d.). Www.youtube.com. Retrieved August 26, 2022, from https://www.youtube.com/watch?v=wIH2qBLwonE
* *React Router V6 Routing*. (2021, November 6). Stack Overflow. https://stackoverflow.com/questions/69868011/react-router-v6-routing
* *Advanced MongoDB Atlas Search Queries*. (n.d.). Www.youtube.com. Retrieved August 26, 2022, from https://www.youtube.com/watch?v=1TQ7f\_\_ISgE
* *Create an Elegant Login and Register Form on React*. (2019, May 25). YouTube. https://www.youtube.com/watch?v=juUaJpMd2LE