

ABSTRACT

Our project introduces a dynamic web application built with Node.js and Express.js for the backend, EJS for the frontend, and MongoDB for database management. The platform seamlessly integrates Power BI dashboards to provide users with powerful data visualization capabilities. Leveraging the flexibility of MongoDB, unstructured data comprising strings and files is efficiently managed, supporting user registration, request submission, and dashboard creation. The Node.js and Express.js backend ensures smooth data flow and processing, while EJS templates facilitate interactive frontend experiences. By incorporating Power BI dashboards, users gain access to advanced analytics features, enhancing decision-making and insight generation. Future enhancements may include further integration with Power BI, expanded data visualization options, and improved user interfaces for enhanced usability. Overall, our project offers a comprehensive solution for collaborative data analysis, combining the strengths of Node.js, Express.js, MongoDB, EJS, and Power BI to deliver a powerful and user-centric platform.

ACKNOWLEDGEMENT

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along the completion of our project.

We would like to thank **Shri. Narayan Rao R Maanay**, Secretary, BNMEI, Bengaluru for providing the excellent environment and infrastructure in the college.

We would like to sincerely thank **Prof. T J Rama Murthy**, Director, BNMIT, Bengaluru for having extended his constant support and encouragement during the course of this project.

We would like to sincerely thank **Dr. S Y Kulkarni**, Additional Director, BNMIT, Bengaluru for having extended his constant support and encouragement during the course of this project.

We would like to express my gratitude to **Prof. Eishwar N Maanay**, Dean, BNMIT, Bengaluru for his relentless support and encouragement.

We would like to thank **Dr. Krishnamurthy G N**, Principal, BNMIT, Bengaluru for his constant encouragement.

We would like to thank, **Dr. Chayadevi M L**, Professor & Head of the Department of Computer Science and Engineering for the encouragement and motivation she provides.

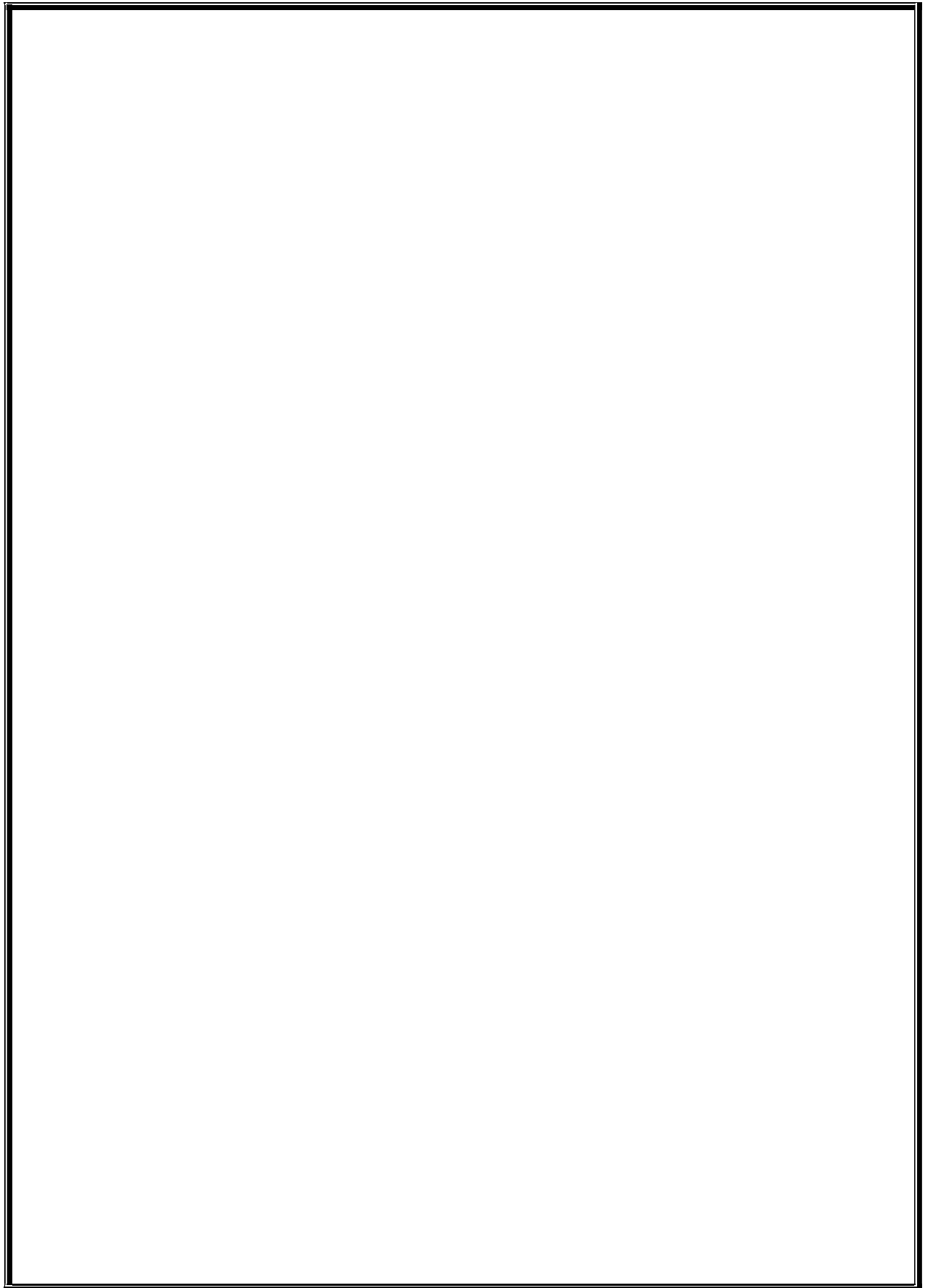
We would also like to thank **Prof. Akshitha Katkeri**, Assistant Professor, Department of Computer Science and Engineering for providing me with her valuable insight and guidance wherever required throughout the course of the project and its successful completion.

Skanda J

1BG21CS127

Srujan Raghavendra S

1BG21CS130



Chapter 1

INTRODUCTION

1.1 Overview

In an era defined by digital connectivity, client-server architectures form the backbone of communication in distributed systems. From online banking to social media interactions, these architectures enable seamless exchanges of data between users and remote servers. However, with this convenience comes the ever-present challenge of safeguarding sensitive information from potential threats. Encryption and decryption emerge as pivotal tools in the arsenal of cybersecurity, offering a shield against unauthorized access and data breaches. Through complex algorithms, encryption scrambles data into an unreadable format, while decryption reverses this process, allowing authorized parties to access the original information securely. Our project embarks on a journey to explore the intricacies of client-server architecture, with a special focus on the encryption and decryption processes.

1.2 Problem Statement

In contemporary distributed systems, ensuring secure data exchange between clients and servers is paramount. However, there is a lack of accessible tools that offer a visual understanding of encryption and decryption processes within a client-server framework. Existing resources often fail to provide an interactive and engaging learning experience, hindering users' comprehension of these crucial security concepts.

1.3 Motivation

In an era marked by increasing digital interconnectedness, the security of data exchanged between clients and servers is paramount. Whether it's personal communications, financial transactions, or sensitive business information, the integrity and confidentiality of this data are constantly at risk from cyber threats. By delving into the intricacies of encryption and decryption within client-server architecture, individuals and organizations can fortify their defences against these threats. The motivation behind this project is to empower individuals and organizations to navigate the digital landscape with confidence and resilience.

1.4 Computer Graphics

Computer graphics and multimedia technologies are becoming widely used in educational applications because they facilitate non-linear, self-learning environments that particularly suited to abstract concepts and technical information.

Computer graphics are pictures and films created using computers. Usually, the term refers to computer-generated image data created with help from specialized graphical hardware and software. It is a vast and recent area in computer science. The phrase was coined in 1960, by computer graphics researchers Verne Hudson and William Fetter of Boeing. It is often abbreviated as CG, though sometimes erroneously referred to as CGI. Important topics in computer graphics include user interface design, sprite graphics, vector graphics, 3D modelling, shaders, GPU design, implicit surface visualization with ray tracing, and computer vision, among others.

The overall methodology depends heavily on the underlying sciences of geometry, optics, and physics. Computer graphics is responsible for displaying art and image data effectively and meaningfully to the user. It is also used for processing image data received from the physical world. Computer graphic development has had a significant impact on many types of media and has revolutionized animation, movies, advertising, video games, and graphic design generally.

1.5 OpenGL API

Open Graphics Library (OpenGL) is a cross-language, cross-platform application programming interface (API) for rendering 2D and 3D vector graphics. The API is typically used to interact with a graphics processing unit (GPU), to achieve hardware-accelerated rendering. Silicon Graphics Inc., (SGI) began developing OpenGL in 1991 and released it on June 30, 1992; applications use it extensively in the fields of computer-aided design (CAD), virtual reality, scientific visualization, information visualization, flight simulation, and video games. Since 2006 OpenGL has been managed by the non-profit technology consortium Khronos Group.

The OpenGL specification describes an abstract API for drawing 2D and 3D graphics. Although it is possible for the API to be implemented entirely in software, it is designed to be implemented mostly or entirely in hardware. The API is defined as a set of functions which