

**The British College**

**Kathmandu**

**Coursework Submission Coversheet**

(Individual coursework only)

**Faculty of Arts, Environment and Technology**

**77261181**

**LBU Student Id:**

Award name: BSc (Hons) Computing

Module Code: COMP607

Module name: Production Project

Module run:

Coursework title: Secured Messaging Application using ASP.NET Core and SignalR.

Due Date:

Module Leader: (In LBU)

Module Supervisor: (In TBC) Mr. Resham Bahadur Pun

**TURNITIN** Checked: Yes No

Submission date & time:

Total Word Count:

Total Number of Pages:

**Academic Honesty Statement**: I confirm that this coursework is completely my own work, not submitted elsewhere at Leeds Beckett University or any other place. I've properly cited all information sources used in my assignment, and I understand that not doing so is considered academic dishonesty.

Signed: Date:

**Teacher’s Signature: Date:**

**This grade is temporary and needs to be checked and confirmed by the appropriate exam organization.**

**Teacher’s Feedback**



**Secured Messaging Application**

**using ASP.NET Core and SignalR.**

**Module: Production Project**

**Date:**

**BSc. (Hons) in Computing**

**[Level 6: 2nd Semester]**

**Submitted To: Mr. Resham Bahadur Pun**

**Submitted By: Umang Shrestha**

1. **Abstract**

The project's goal is to create a secure, user-friendly messaging app using ASP.NET Core and SignalR, prioritizing user privacy and security. It aims to offer a reliable, easy-to-use platform for real-time chats while safeguarding user privacy. ASP.NET Core, a flexible and efficient web framework, is used to build a scalable backend. This framework manages server resources effectively and supports diverse application needs. SignalR, a library for real-time communication, is integrated into the app for smooth, instant messaging between users.

To boost security, modern encryption techniques provide end-to-end encryption, ensuring data remains private and safe from unauthorized access. Key project achievements include designing a user-friendly chat interface, developing a secure encryption system, and effectively integrating real-time messaging with SignalR. The final app showcases the potential for building a secure, high-performance messaging platform using these technologies.

The project's success enables further enhancements and potential applications in sectors needing secure communication, such as healthcare, finance, and government organizations requiring confidential channels to share sensitive information.

1. **Acknowledgements**

I am grateful to my project supervisor, Resham Bahadur Pun, for unwavering guidance and support throughout my Secured Messaging Application project. His knowledge and expertise had been instrumental in helping me understand ASP.NET Core MVC and SignalR. His continuous feedback and constructive criticism had been critical in ensuring the project met high standards. Without his help, it would be difficult to complete the project completely.

The incredible faculty of the Computer Science department at The British College had been a true inspiration to me because of providing an excellent foundation and an ever-evolving learning environment that ignited my passion for exploring and excelling in the world of computer science. Thank you, Computer Science faculty at The British College, for lighting up my path towards success and beyond. A special thanks went to my fellow classmates and friends who offered their insights, suggestions, and helped me troubleshoot various issues during the development of the application. Their friendship and excitement made this journey an enjoyable and rewarding experience.

In conclusion, I am grateful for the opportunity to have worked on this project, which not only enhanced my technical skills but also deepened my understanding of real-world challenges and solutions in the field of secure communication systems.

1. **Introduction**

In today's era of online interactions, the importance of safety and discretion has grown substantially. The emergence of numerous social networking and messaging services has transformed how we connect with one another; however, it has also brought forth worries regarding information protection and personal privacy. As a result, it is vital to create strong, dependable communication tools that guarantee the preservation and accuracy of user information. As a Computer Science student, I have undertaken the project of developing a Secured Messaging Application using ASP.NET Core MVC and SignalR. This project aims to create a web-based messaging platform that offers users end-to-end encryption, secure authentication, and a user-friendly interface. The main motivation behind this project is to provide users with a messaging application that prioritizes their security and privacy, while also offering a seamless and efficient communication experience.

The ASP.NET Core MVC constitutes a robust and adaptable structure that facilitates the creation of web apps employing the Model-View-Controller (MVC) architectural paradigm. This offers an excellent basis for developing a protected communication system, as it promotes distinct responsibility divisions, code testing, and ongoing code management. In contrast, SignalR serves as a live interaction tool that enables the incorporation of instantaneous messaging features in web-based applications. This undertaking seeks to combine SignalR with the ASP.NET Core MVC to establish a messaging solution exhibiting minimal lag and superior efficiency. Performance is a key aspect of .NET. With each new release, the .NET team and community contributors dedicate efforts to enhance performance, enabling .NET applications to run faster and consume fewer resources (Conroy, 2022).

Throughout the duration of this endeavor, multiple protective measures have been put in place to safeguard user information and maintain the privacy of communications. This encompasses employing end-to-end encryption, robust password hashing, and token-driven authentication. Moreover, the app offers a user-centric interface equipped with straightforward navigation and messaging functions, facilitating seamless interaction between users. As demands for both functionality and performance have increased, our dependence on .NET as an essential component of our infrastructure has become even more profound (Watson, 2023). This analysis will deliver a detailed examination of the developmental stages and assorted elements involved in constructing the Secure Communication Application. It will encompass aspects such as the project's goals, system framework, design and execution of assorted security attributes, as well as the assessment and appraisal of the app. Upon concluding this analysis, readers should possess a comprehensive comprehension of the technical facets and hurdles associated with developing a protected messaging application using ASP.NET Core MVC and SignalR. Employing SignalR makes it simple to develop a real-time chatroom application accessible by multiple users simultaneously, while also ensuring an excellent user experience (Patil, 2022).

1. **Review of Literature**
2. Messaging Platforms

* In the annals of history, one may find that the genesis of messaging platforms emerged during the latter part of the 20th century. Jarkko Oikarinen, a pioneer in this field, conceived the Internet Relay Chat (IRC) back in 1988. As decades flew by, these innovative platforms gradually cemented their status as indispensable pillars in the realm of contemporary communication.
* Smartphones ascendancy ushered in a new era. Groundbreaking messaging services, including WhatsApp, Facebook Messenger, and WeChat, emerged, revolutionizing the landscape. These cutting-edge platforms boasted an array of features, encompassing multimedia sharing, video conferencing, and impregnable encryption, as noted by Church & De Oliveira in 2013. If your communication is strong, it becomes simpler to create a marketing campaign that fosters the market. This campaign consists of a series of content offerings that allow potential customers to explore your solutions gradually. When your content nurturing program is based on a unified message, it forms the basis on which potential customers will make a purchase (Regensburg, 2014) .

1. Communication Advancements

* The trajectory of messaging platform development intertwines intimately with advancements in communication tech. Groundbreaking protocols, such as WebSocket, have facilitated seamless, two-way interactions between clients and servers, as expounded by Fette & Melnikov in 2011. In the years that followed, real-time messaging libraries, SignalR being a prime example, have increasingly simplified the process of crafting real-time web applications, as elucidated by Roth in 2012.

1. Design and Usability of Messaging Platforms

* Investigations delving into messaging platform design and user experience emphasize the import of centering users in design principles, crafting accessible interfaces, and devising malleable layouts. Furthermore, scholarly inquiries have underscored the vital role that privacy and security play in influencing user adoption rates and satisfaction levels with regard to these messaging platforms.

1. Web Development Frameworks

* Hailing from Microsoft's stable, ASP.NET Core is emerging as an open-source, versatile web development framework that has captured the interest of many, owing to its performance, adaptability, and compatibility with an array of programming language. This powerful framework empowers developers, enabling them to craft scalable, maintainable web applications, with messaging platforms being a notable example.

1. Real-time Web Solutions

* As the years rolled by, real-time web solutions experienced a substantial surge, propelled by the insatiable appetite for instantaneous communication and cooperation. SignalR, an ingenious real-time messaging library tailored for ASP.NET Core, rose to prominence as a go-to option for developers crafting messaging platforms and a myriad of real-time web applications. By untangling the intricate web of real-time communication, SignalR allows developers to laser-focus on the core logic of their applications.