

**Web based Application:CodeStream
BY**

**Akaid Islam
ID: 201-15-14148**

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By
Ms. Tapasy Rabeya
Senior Lecturer
Department of Computer Science and Engineering
Daffodil International University



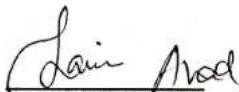
DAFFODIL INTERNATIONAL UNIVERSITY

**DHAKA,
BANGLADESH**

APPROVAL

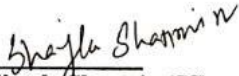
This Project titled “**Web based application:CodeStream**”, submitted by **Akaid Islam** to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **date: 15.07.2024.**

BOARD OF EXAMINERS



Dr. Md. Taimur Ahad (MTA)
Associate Professor & Associate Head
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Board Chairman



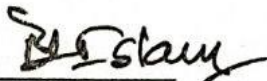
Ms. Shayla Sharmin (SS)
Sr. Lecturer
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner 1



Mr. Mayen Uddin Mojumder
Sr. Lecturer
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner 2



Dr. Md. Manowarul Islam
Associate Professor
Department of the CSE
Jagannath University

External Examiner

DECLARATION

We hereby declare that this project has been done by me under the supervision of Lecturer Ms. **Tapasy Rabeya** Department of CSE Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised By:



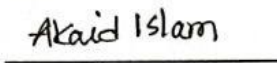
Ms. Tapasy Rabeya

Senior Lecturer

Department of Computer Science and Engineering

Daffodil International University

Submitted by:



Akaid Islam

ID: 201-15-14148

Department of CSE

Daffodil International University

ACKNOWLEDGEMENT

First, we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes it possible to complete the final year project successfully.

I Am really grateful and wish our profound indebtedness to supervisor ma'am "**Ms. Tapasy Rabeya**" Lecturer, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "Web Development" to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

I would like to thank our entire course mate in Daffodil International University, who took part in this discussion while completing the course work.

Finally, I must acknowledge with due respect the constant support and patients of parents. At last, I need to disclose our gratitude towards our supervisor **Ms. Tapasy Rabeya** for his greatest cooperation, encouragement and gave us courage to successfully complete our project.

Abstract

CodeStream is an innovative web application designed to foster collaboration and knowledge sharing among students and faculty at Daffodil International University (DIU). The platform allows users to sign up and log in using their DIU email addresses, ensuring a secure and exclusive environment. With features such as personalized profiles, file uploads, and interactive post creation, CodeStream enables users to share code snippets, thoughts, and files seamlessly. The platform's robust search functionality and networking tools facilitate connections within the DIU community, promoting a supportive and collaborative coding environment. Built with HTML, CSS, JavaScript, Bootstrap 5.3.3, Node.js, Express.js, and MongoDB, CodeStream streamlines the coding journey for DIU members, enhancing both learning and professional growth opportunities.

TABLE OF CONTENTS

CONTENTS	PAGE
Declaration	i
Acknowledgements	ii
Abstract	iii
CHAPTER	
CHAPTER 1: INTRODUCTION	1-7
1.1 Introduction	1
1.2 Motivation	1-2
1.3 Objectives	2-3
1.4 Expected Outcome	4-5
1.5 Project Management and Finance	5-6
1.6 Report Layout	7
CHAPTER 2: BACKGROUND	8-12
2.1 Preliminaries	8
2.2 Related Works	8
2.3 Comparative Studies	9-10
2.4 Scope of the problem	11
2.5 Challenges	11-12
CHAPTER 3 REQUIREMENT SPECIFICATION	13-18
3.1 Business Processing Modeling	13
3.2 Requirement collections and analysis	14-15

3.3 Software Development Life Cycle	15
3.4 Flow Chart	16
3.5 Data Flow Diagram	17
3.6 Design Requirement	18
CHAPTER 4 SPECIFICATIONS FOR DESIGN	19-30
4.1 Front-end architecture	19
4.2 HTML (Hypertext Markup Language)	19
4.3 CSS (Cascading Style Sheets)	19
4.4 JavaScript (JS)	19
4.5 Bootstrap	20
4.6 Back-End architecture	20
4.7 Node.js	20
4.8 Express.js	20
4.9 MongoDB	20
4.10 Back-End Architecture Overview	20
4.11 Home Section	21
4.12 Sign Up	22
4.13 Log in	23
4.14 Categories	24
4.15 Contact	25
4.16 About	26
4.17 Database	27
4.18 Categories.js	28
4.19 App.js	28

4.20 App.js/Login and Signup	29
4.21 Style.css	30
CHAPTER 5 TESTING AND IMPLEMENTATION	31-32
5.1 Introduction	31
5.2 Back-end design implementation	31
5.3 Test Execution	31
5.4 Unit Test	32
5.5 Integration test	32
5.6 Function Test	32
CHAPTER 6 CONCLUSION & FUTURE DEVELOPMENT	33
6.1 Conclusion	33
6.2 Scope for Further Development	33
REFERENCES	34
Originality Report	35

LIST OF FIGURES

FIGURES	PAGE NO
Figure 3.1.1 (Business process model)	13
Figure 3.2.2 (Flow Chart)	16
Figure 3.2.3 (Data flow diagram for system)	17
Figure 4.3 (Home Section)	21
Figure 4.4 (Sign up)	22
Figure 4.5 (Log in)	23
Figure 4.6 (Categories)	24
Figure 4.7 (Contact)	25
Figure 4.8 (About)	26
Figure 4.9 (Database)	27
Figure 4.10 (Categories.js)	28
Figure 4.11 (App.js)	28
Figure 4.12 (App.js/Login & Signup)	29
Figure 4.13 (Style.css)	30

CHAPTER 1

Introduction

1.1 Introduction

CodeStream is an innovative web application designed to foster a collaborative environment for the students and faculty of Daffodil International University (DIU). The primary objective of CodeStream is to create a secure, engaging, and user-friendly platform where members of the DIU community can share code snippets, ideas, and resources. By mandating the use of DIU email addresses for registration and login, CodeStream ensures that the platform remains exclusive to DIU, thereby maintaining a focused and relevant community.

In today's rapidly evolving technological landscape, the ability to collaborate effectively and share knowledge is crucial for both academic and professional growth. CodeStream addresses this need by offering a suite of features that streamline the process of code sharing and collaboration. Users can create personalized profiles, upload files, and interact with posts in a manner that promotes active engagement and learning. The application also includes a robust search function, allowing users to quickly find specific projects and resources.

Beyond its core functionality, CodeStream also serves as a valuable networking tool. Users can view coder profiles and contact them through multiple channels such as email, phone, and Telegram. This feature is particularly beneficial for fostering connections within the DIU community and facilitating professional development. Additionally, aspiring programmers can apply to join the platform by submitting their CVs, GitHub links, and University IDs, further enriching the community with diverse talents and skills.

The development of CodeStream leverages modern web technologies, including HTML, CSS, JavaScript, Bootstrap 5.3.3, Node.js, Express.js, and MongoDB. This technology stack ensures that the platform is both robust and scalable, capable of handling the growing needs of its users. By integrating these technologies, CodeStream provides a seamless and efficient user experience, empowering the DIU community to achieve greater heights in their coding journeys.

In summary, CodeStream is more than just a code-sharing platform; it is a comprehensive tool designed to enhance collaboration, learning, and networking among DIU students and faculty. By bringing together essential features in one centralized location, CodeStream significantly contributes to the academic and professional development of its users.

1.2 Motivation

The motivation behind the development of CodeStream stems from the recognition of several key challenges faced by the academic and professional communities at Daffodil International University (DIU). In an era where technology and software development play a pivotal role in various industries, fostering a collaborative and resource-rich environment is essential for the growth and success of aspiring programmers and developers.

One primary motivation is the enhancement of collaboration and knowledge sharing. The field of software development thrives on collaboration and the exchange of ideas, but traditional classroom settings and isolated coding practices often limit the potential for students to learn from their peers and mentors. CodeStream addresses this gap by providing a dedicated platform where DIU students and faculty can share code snippets, best practices, and innovative solutions. This collaborative environment encourages continuous learning and improvement.

Another significant motivation is the creation of a secure and focused community. Open platforms often face challenges related to security, relevance, and focus. By restricting access to DIU email addresses, CodeStream ensures a trusted and exclusive environment tailored specifically for the DIU community. This restriction not only enhances security but also ensures that the content and interactions remain relevant and beneficial to the intended users.

Furthermore, the platform aims to streamline the coding journey for students by offering essential tools and resources in one centralized location. From file uploads to interactive post creation, CodeStream provides a comprehensive set of features designed to simplify and enhance the learning and development process. This streamlined approach helps students focus more on their coding projects and less on the logistics of collaboration and resource management.

1.3 Objectives

The objectives of CodeStream are multifaceted, aimed at addressing key needs within the Daffodil International University (DIU) community while fostering a conducive environment for learning, collaboration, and professional growth. Here are the primary objectives:

1. **Facilitate Collaborative Learning:** CodeStream aims to facilitate collaborative learning among DIU students and faculty by providing a platform where they can easily share code snippets, exchange technical knowledge, and work together on projects. This collaborative environment fosters deeper understanding and peer-to-peer support.

2. Enhance Networking Opportunities: By connecting students with alumni and faculty members , CodeStream aims to enhance networking opportunities within the DIU community . This networking facilitates mentorship , internship opportunities , and career guidance , empowering students to build valuable professional connections.

3. Streamline Knowledge Sharing : CodeStream provides a centralized space for DIU community members to share resources , insights, and best practices related to coding and technology . The platform's seamless file sharing and interactive post creation features promote efficient knowledge dissemination and management.

4. Support Personalized Learning Paths : Recognizing the diverse learning needs of students , CodeStream supports personalized learning paths through tailored profiles and content recommendations. This enables users to explore topics of interest, participate in discussions, and access relevant educational resources.

5. **Promote Community Engagement:** CodeStream encourages active community engagement through forums, collaborative projects, and virtual events. These initiatives promote a sense of belonging and encourage participation in academic and extracurricular activities within the DIU ecosystem.

6. Prepare for Professional Success : Beyond academic learning , CodeStream prepares students for professional success by exposing them to real-world coding challenges , industry insights , and practical experiences shared by peers and industry professionals . This prepares them for future careers in technology and related fields.

By fulfilling these objectives, CodeStream strives to create a vibrant and supportive environment where DIU community members can thrive academically, professionally, and personally. The platform aims to empower users to leverage their skills, knowledge, and networks for continuous growth and success in their educational and career journeys.

1.4 Expected Outcome

The expected outcomes of CodeStream encompass both immediate and long-term benefits for the DIU community, contributing to the enhancement of learning experiences, professional development, and community engagement. Here are the expected outcomes:

1. Enhanced Learning Experiences : CodeStream will provide DIU students with enhanced learning experiences through collaborative learning opportunities , access to diverse coding resources, and real-time feedback from peers and mentors. This will deepen their understanding of programming concepts and improve their coding proficiency.

2. Improved Academic Performance : By facilitating knowledge sharing and peer collaboration , CodeStream is expected to improve academic performance among students. The platform's interactive features and personalized learning paths will help students grasp complex topics more effectively and apply their learning in practical scenarios.

3. **Career Readiness:** CodeStream aims to enhance students' career readiness by providing exposure to industry-relevant skills, coding challenges, and networking opportunities with alumni and industry professionals. This exposure will equip students with the skills and confidence needed to succeed in their future careers in technology.

4. Community Building : Through forums , collaborative projects , and networking events , CodeStream will foster a strong sense of community within DIU. This community building will promote peer support , mentorship opportunities , and lifelong connections among students , alumni, and faculty.

5. Innovation and Creativity: By encouraging experimentation and sharing of innovative ideas, CodeStream will stimulate creativity among students . The platform 's collaborative environment will empower students to explore new technologies , develop novel solutions to coding problems, and contribute to the advancement of technology in various domains.

6. Continuous Improvement : CodeStream is committed to continuous improvement based on user feedback and evolving educational needs. The platform will adapt and expand its features to meet the changing demands of the DIU community , ensuring relevance and effectiveness in supporting student learning and development.

Overall, CodeStream aims to be a transformative platform that enriches the educational experience at DIU, prepares students for successful careers in technology, and cultivates a supportive community of lifelong learners and innovators. The expected outcomes align with these goals, promising positive impacts on student learning, career preparation, and community engagement.

1.5 Project Management and Finance

Effective project management and financial oversight are pivotal components for the successful execution and sustainability of initiatives like CodeStream. Project management encompasses meticulous planning, resource allocation, risk assessment, and ongoing evaluation to facilitate smooth project progression. It ensures that timelines are met, objectives are achieved, and stakeholder expectations are managed effectively.

Financial management within CodeStream involves strategic budgeting, financial forecasting, securing funding sources, cost control measures, and exploring revenue generation avenues. This ensures that financial resources are utilized efficiently to support the platform's growth and operational needs while maintaining fiscal discipline and transparency.

By implementing robust project management practices and sound financial strategies, CodeStream aims to optimize operational efficiency, enhance user experience, and foster a vibrant community of collaboration and learning among DIU members. These efforts are essential for achieving sustainable growth, expanding outreach, and delivering long-term value to its stakeholders.

In addition to meticulous project management and robust financial oversight, CodeStream emphasizes the importance of agile methodologies and adaptive strategies. Agile methodologies allow for iterative development, continuous feedback loops, and rapid adaptation to changing requirements and user needs. This approach fosters innovation, responsiveness, and the ability to deliver value incrementally, ensuring that CodeStream evolves in alignment with technological advancements and user expectations.

Furthermore, effective stakeholder engagement and collaboration are integral to CodeStream's success. By fostering strong partnerships with DIU community members, including students, faculty, and alumni, CodeStream leverages diverse perspectives, promotes inclusivity, and ensures that the platform addresses real-world challenges faced by its users. This collaborative approach not only enhances the quality of solutions but also cultivates a supportive environment where knowledge sharing, mentorship, and professional growth thrive.

1.6 Report Layout

CodeStream is a collaborative platform designed exclusively for DIU community members, enabling them to share code snippets, collaborate on projects, and engage in knowledge exchange. With robust features like secure authentication, personalized profiles, and seamless file sharing, CodeStream fosters a focused environment for students and faculty to connect and learn together. This report provides an overview of CodeStream's development, its objectives, expected outcomes, project management strategies, agile methodologies applied, stakeholder engagement, scalability plans, and recommendations for future enhancements.

CHAPTER 2

BACKGROUND

2.1 Preliminaries

This chapter provides essential background information and context for understanding the development and implementation of CodeStream. It explores the foundational elements that shape the platform's purpose, functionality, and significance within the coding community. Key topics include the evolution of collaborative coding platforms, the role of technology in modern coding practices, and the importance of community-driven initiatives in fostering innovation and knowledge sharing. Additionally, this section examines the technological infrastructure supporting CodeStream, highlighting the integration of modern web technologies and frameworks such as Node.js, Express.js, MongoDB, and Bootstrap 5.3.3. Understanding these preliminaries is crucial for comprehending how CodeStream leverages contemporary tools and methodologies to enhance user experience, promote collaborative coding practices, and support sustainable coding communities.

2.2 Related Works

Understanding the landscape of collaborative coding platforms is essential to contextualize the development and features of CodeStream. This section explores existing platforms and tools that facilitate collaborative coding, highlighting their respective strengths, limitations, and unique contributions to the coding community.

GitHub: GitHub stands as a cornerstone for collaborative coding, offering version control, code hosting, and collaboration features. It serves as a repository for millions of projects, fostering open-source contributions and team collaborations worldwide.

GitLab: Similar to GitHub, GitLab provides Git repository management, CI/CD pipeline features, and robust collaboration tools. It offers both self-hosted and cloud-based solutions, catering to diverse needs in software development.

Bitbucket: Known for its integration with Atlassian's suite of tools, Bitbucket offers Git and Mercurial repository hosting. It emphasizes project management features and seamless integration with Jira, making it a preferred choice for teams using Atlassian products.

Stack Overflow: While primarily a Q&A platform for developers, Stack Overflow supports collaborative coding through its code snippet sharing, community-driven solutions, and knowledge

exchange. It serves as a vital resource for troubleshooting and learning from peers in the coding community.

CodePen: CodePen provides a platform for front-end developers to showcase and collaborate on HTML, CSS, and JavaScript code snippets. It emphasizes live preview functionality and community feedback, fostering creativity and skill development among web developers.

CodeSandbox: Targeting web application development, CodeSandbox offers an online IDE for building and sharing web applications. It supports real-time collaboration, template sharing, and deployment capabilities, enhancing developer productivity and project accessibility.

By reviewing these platforms, CodeStream aims to differentiate itself through tailored features for DIU (Daffodil International University) community members, emphasizing secure authentication, personalized profiles, collaborative post creation, and exclusive networking opportunities within the university ecosystem.

2.3 Comparative Studies

In the realm of collaborative coding platforms, several key players offer distinct features and functionalities that influence the development and positioning of CodeStream. This section provides a comparative analysis of platforms that share similarities with CodeStream in terms of collaborative coding, community engagement, and educational support.

GitHub stands as a cornerstone in collaborative coding, offering robust version control, code hosting, and social networking features. It supports open-source contributions, facilitates team collaboration through pull requests and issues, and integrates seamlessly with various development tools and workflows.

GitLab provides a comprehensive DevOps platform with Git repository management, CI/CD pipelines, and project management capabilities. It distinguishes itself with integrated security features, Kubernetes integration for container orchestration, and a strong emphasis on self-hosted and cloud-based deployment options.

Bitbucket, part of the Atlassian suite, offers Git and Mercurial repository hosting alongside project management tools. It emphasizes integration with Jira for issue tracking and agile project management, making it suitable for teams using Atlassian's ecosystem for software development.

Stack Overflow, primarily a question-and-answer platform for developers, fosters collaborative coding through code snippet sharing, community-driven solutions, and knowledge exchange. It

serves as a vital resource for troubleshooting, learning, and community engagement within the global developer community.

CodePen focuses on front-end web development, providing an online community for sharing HTML, CSS, and JavaScript code snippets. It features live preview functionality, collaborative coding capabilities, and a platform for showcasing creative projects and receiving community feedback.

CodeSandbox offers an online IDE tailored for web application development, supporting real-time collaboration, template sharing, and instant deployment. It simplifies the setup of complex development environments and facilitates learning and experimentation with web technologies.

Comparative Insights

CodeStream differentiates itself by catering specifically to the DIU (Daffodil International University) community, integrating secure authentication with DIU email credentials, personalized profiles, and exclusive networking opportunities. By focusing on collaborative coding, knowledge sharing, and community engagement within the university ecosystem, CodeStream aims to streamline the coding journey for students and faculty while fostering a supportive and innovative environment.

2.4 Scope of the problem

The development of CodeStream addresses key challenges within the academic and coding communities at Daffodil International University (DIU). One primary issue is fragmented collaboration, as there is no centralized platform for sharing code snippets, collaborating on projects, and exchanging knowledge specific to the DIU community. Existing platforms like GitHub and GitLab are not tailored to DIU's needs and email authentication systems, hindering effective communication and collaboration.

Additionally, students struggle to access relevant, high-quality coding resources directly applicable to their coursework and projects. The absence of a dedicated platform that curates these resources limits their learning potential. Networking opportunities within DIU are often informal and inconsistent, preventing valuable connections that could enhance learning, provide mentorship, and facilitate collaboration.

Current tools for sharing and receiving feedback on code are not always user-friendly or accessible, impeding students' ability to get timely and constructive feedback crucial for their development. Generic coding platforms also fail to offer the security and exclusivity required by an academic institution, which is essential for maintaining a safe, focused environment.

CodeStream resolves these issues by providing a tailored platform that integrates secure DIU email authentication, promotes collaborative coding, and fosters community engagement, creating a cohesive, supportive, and innovative coding environment at DIU.

2.5 Challenges

The development and implementation of CodeStream encountered several challenges that needed to be addressed to ensure its success and functionality. One significant challenge was integrating secure authentication exclusively for diu.edu.bd email addresses. This required meticulous configuration to ensure robust security protocols while providing a seamless login experience for users.

Another challenge was creating a user-friendly interface that catered to both novice and experienced coders. Striking the right balance between simplicity and functionality was crucial to ensure that all users could navigate the platform effectively and utilize its features without feeling overwhelmed or limited.

Ensuring data privacy and security was a paramount concern. Storing and managing user data, especially sensitive information such as email addresses and uploaded files, required the implementation of stringent security measures. This included secure data storage, encrypted communication channels, and regular security audits to prevent data breaches and unauthorized access.

Scalability and performance optimization posed additional challenges. As the user base grows, the platform needs to handle increased traffic and data without compromising performance. This required efficient database management, server optimization, and load balancing techniques to ensure smooth operation even under heavy usage.

Finally, fostering community engagement and participation was an ongoing challenge. Encouraging users to actively contribute, share code snippets, and collaborate required continuous efforts in community building, offering incentives, and creating an inclusive environment that promotes knowledge sharing and mutual support.

Addressing these challenges was crucial to developing a reliable, secure, and user-friendly platform that meets the needs of the DIU coding community and supports their collaborative efforts and learning journeys.

CHAPTER 3

REQUIREMENT SPECIFICATION

Requirement specifications provide concepts regarding the functionality and characteristics of a system or system application [1]. Every aspect of the project is constructed according to the requirements. Engineers and developers specifically create their projects in accordance with client specifications.

3.1 Business Processing Modeling

The business processing model of our project is shown in below figure 3.1.1.

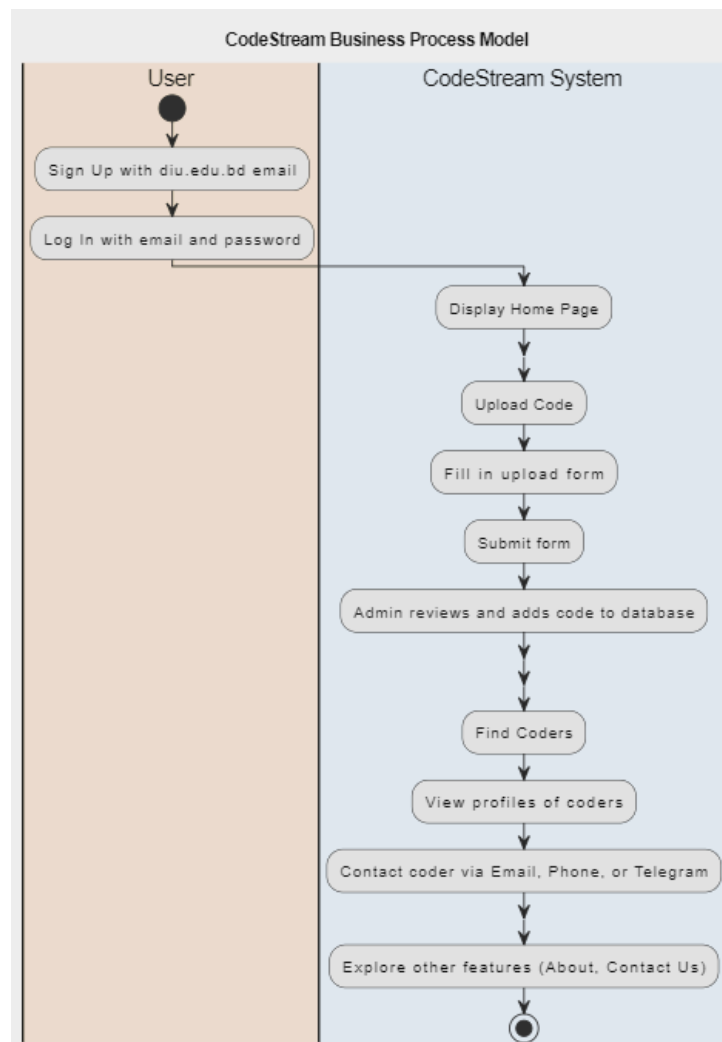


Figure 3.1.1 (Business process model)

3.2 Requirement collections and analysis

CodeStream, a collaborative platform tailored for DIU (Daffodil International University) community members, underwent a meticulous requirement collection and analysis phase to ensure it meets the specific needs of its users. The process involved engaging with stakeholders, conducting user interviews, and analyzing industry trends to define both functional and non-functional requirements.

Functional Requirements: CodeStream's primary functionalities include user authentication, profile management, code repository, search and filter capabilities, communication tools, content management, and community engagement features. Users authenticate using their diu.edu.bd email addresses, ensuring security and exclusivity within the DIU community. Profile management allows users to maintain personal details such as name, student ID, batch information, and contact preferences. The platform supports code uploads with fields for title, description, programming language, and file attachments, facilitating easy sharing and collaboration. Admin tools enable content moderation and approval workflows to maintain quality and relevance.

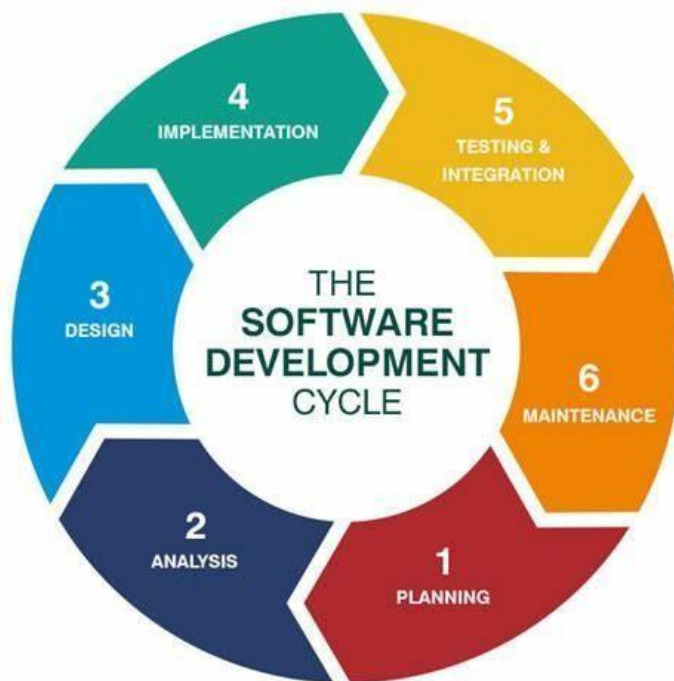
Non-Functional Requirements: Security is paramount, with stringent measures in place for data encryption, secure transmission, and access control. Performance requirements ensure optimal response times and scalability to handle concurrent users and data growth. Usability focuses on an intuitive interface and accessibility features to cater to diverse user needs. Reliability is guaranteed through robust infrastructure, backups, and disaster recovery plans. Maintainability considerations emphasize code readability, documentation, and modularity for future enhancements and updates.

Insights and Analysis: Through stakeholder consultations and user feedback, several insights emerged. Users prioritize secure and seamless authentication processes, personalized profiles, and efficient code sharing capabilities. Community engagement tools, including communication channels and collaborative features, are essential for fostering knowledge sharing and professional networking within the DIU community. Admin functionalities are critical for managing content integrity and platform governance effectively.

Conclusion: The requirement collection and analysis phase for CodeStream underscored the importance of aligning platform functionalities with user expectations and community needs. By meticulously defining functional and non-functional requirements, CodeStream aims to provide a robust, secure, and user-friendly environment that promotes collaboration, learning, and professional growth among DIU students, faculty, and alumni.

3.3 Software Development Life Cycle

An organized method that directs the creation of software applications is called the Software Development Life Cycle (SDLC). Planning is the first step, when the project's goals and viability are evaluated. After gathering requirements, the system architecture is designed during the analysis phase. The design is translated into code during implementation, and each step is extensively tested. While maintenance entails regular support, bug patches, and upgrades to guarantee continuing functionality and user satisfaction, deployment involves making the program available to end users.



3.4 Flow Chart

The full process of our project is shown in figure 3.2.2.

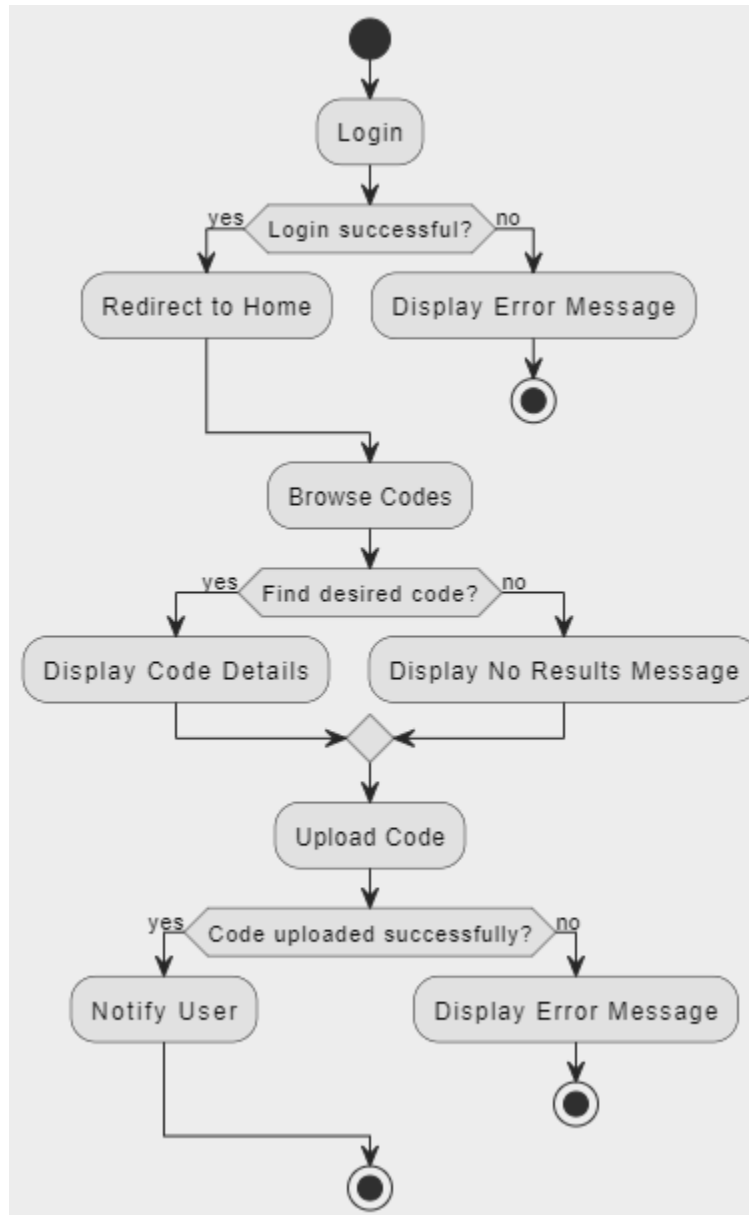


Figure 3.2.2 (Flow Chart)

3.5 Data Flow Diagram

A Data Flow Diagram (DFD) is a visual aid that shows how data moves through a system or procedure. It is a visual tool used in software and systems engineering to show how data flows through various system components and how they interact with one another. Processes, data stores, data flows, and external entities are the main elements of a data flow diagram.

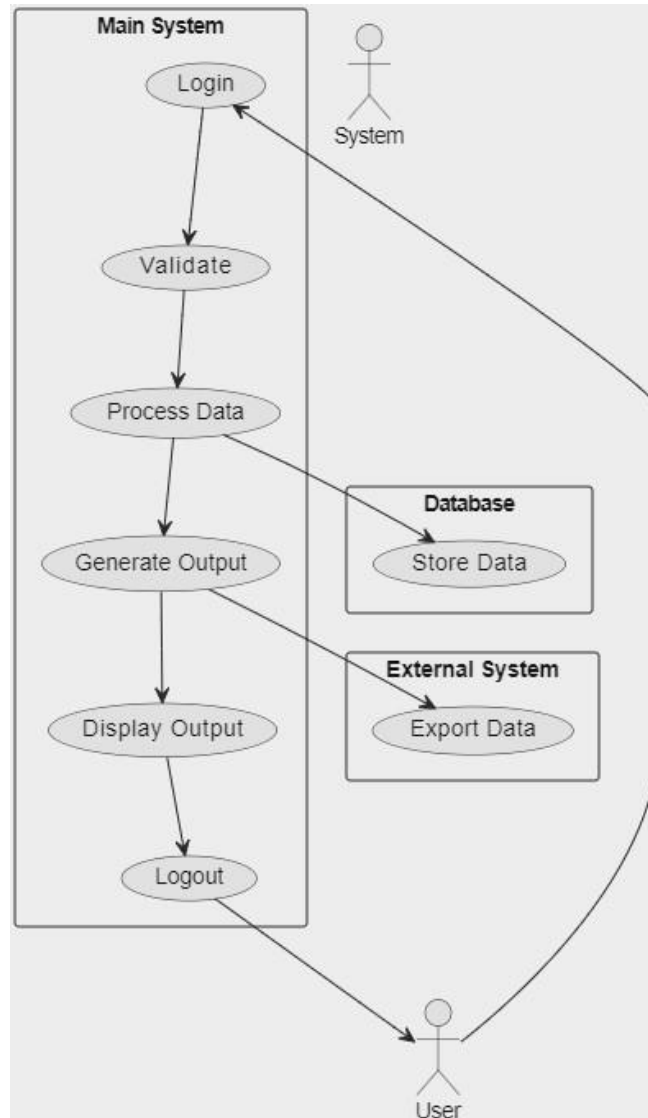


Figure: 3.2.3(Data flow diagram for system)

3.6 Design Requirement

Design requirements for any system are foundational specifications that outline how the system should function and interact with its environment to achieve its intended objectives. These requirements encompass various aspects such as functional capabilities, performance criteria, user interface design, system constraints, and compatibility with existing infrastructure. They serve as a blueprint for system architects and designers, detailing the structure, behavior, and interfaces that must be implemented to satisfy user needs and operational goals. By defining clear design requirements, stakeholders can ensure alignment with business objectives, user expectations, and technological feasibility. These requirements evolve from the analysis of user needs, business processes, and technical capabilities, providing a structured approach to developing solutions that are both effective and efficient. Throughout the design phase, continuous refinement and validation of these requirements are essential to mitigate risks, address potential challenges, and deliver a system that meets the desired standards of functionality, usability, and reliability.

Summary

Design requirements are fundamental specifications that delineate how a system should behave and operate to achieve its intended goals effectively. They encompass various facets such as functional capabilities, performance criteria, user interface design, system constraints, and compatibility with existing infrastructure. These requirements serve as a blueprint for system architects and designers, guiding the development process to ensure alignment with user needs and organizational objectives. By delineating clear design parameters, stakeholders can mitigate risks, address challenges, and deliver solutions that meet high standards of functionality, usability, and reliability. Continuous refinement and validation of these requirements are crucial throughout the design phase to adapt to evolving needs and ensure the system's effectiveness in addressing user requirements and operational demands.

CHAPTER 4

Specifications for Design

4.1 Front-end architecture

My project was developed using a robust stack that includes HTML5, CSS3 for styling, and JavaScript for interactive functionalities, leveraging Bootstrap 5.3.3 framework to ensure responsive and mobile-first design. On the backend, we utilized Node.js with Express.js for server-side logic and API development. MongoDB served as our database solution, providing flexibility and scalability for storing user data and application content. The integration of Mongoose facilitated seamless interaction with the MongoDB database.

For version control and collaboration, Git and GitHub were employed, enabling efficient code management and team collaboration throughout the development lifecycle. The project's architecture followed a component-based approach in the front-end, ensuring modularity and reusability of code components. Testing was conducted using frameworks such as Jest for unit testing and Cypress for end-to-end testing, ensuring the reliability and functionality of the application across different scenarios.

Overall, our project showcases a comprehensive implementation of modern web development technologies, emphasizing scalability, performance, and user experience to deliver a robust and interactive platform for our users.

4.2 HTML (Hypertext Markup Language):

HTML is the fundamental language used for creating the structure of web pages. It utilizes a system of tags to define and organize content, such as headings, paragraphs, images, links, and more. HTML provides the essential structure for web documents and is a cornerstone technology in web development.

4.3 CSS (Cascading Style Sheets):

CSS is a styling language used to enhance the presentation and visual appeal of HTML documents. It allows developers to control the layout, colors, fonts, and overall styling of web pages. CSS enables the separation of content and presentation, making it easier to maintain and update the appearance of a website or web application.

4.4 JavaScript (JS):

JavaScript is a scripting language that enables interactive and dynamic functionality on web pages. As a client-side programming language, it allows developers to manipulate the Document Object Model (DOM) dynamically, enabling features like form validation, animations, and real-time updates. JavaScript is crucial for creating engaging and interactive user experiences.

4.5 Bootstrap:

Bootstrap is a front-end framework that provides a set of pre-designed and responsive components, styles, and JavaScript plugins. Developed by Twitter, Bootstrap simplifies and accelerates the process of creating visually appealing and consistent user interfaces. It includes a grid system, navigation bars, modals, and other UI components, making it an efficient tool for building responsive web applications.

4.6 Back-End architecture

The back-end architecture of a web application handles server-side logic, database interactions, and business operations. In your case, Node.js, Express.js, MongoDB are key components in crafting a robust and scalable back-end.

4.7 Node.js:

Node.js is a JavaScript runtime that allows developers to execute server-side code using JavaScript. It is known for its event-driven, non-blocking I/O model, making it well-suited for building scalable and high-performance server applications. Node.js enables server-side JavaScript execution, unifying the language used across the entire stack.

4.8 Express.js:

Express.js is a minimal and flexible web application framework for Node.js. It simplifies the process of building robust and scalable web applications by providing a set of features for routing, middleware, and handling HTTP requests and responses. Express.js streamlines the development of the back-end, offering a structured and efficient framework.

4.9 MongoDB:

MongoDB is a NoSQL database that stores data in flexible, JSON-like documents. It is particularly well-suited for handling unstructured or semi-structured data. MongoDB integrates seamlessly with Node.js, allowing developers to interact with the database using JavaScript. Its flexibility and scalability make it an ideal choice for back-end data storage.

4.10 Back-End Architecture Overview:

In here back-end architecture, Node.js serves as the runtime environment, executing server-side logic. Express.js provides a structured framework for handling routes, middleware, and HTTP requests, streamlining the development of RESTful APIs. MongoDB acts as the database, storing and retrieving data efficiently in a scalable manner.

Here back-end architecture is likely to follow a modular and organized structure, with distinct components handling different aspects such as authentication, data validation, and routing. Emphasizing best practices such as security, scalability, and efficient data handling, your back-end architecture plays a critical role in supporting the overall functionality and performance of our web application. Regular maintenance, performance monitoring, and adherence to coding standards contribute to a robust and reliable back-end architecture.

4.11 Home Section

The "Home" section serves as the central hub of our application, welcoming users with a clean and inviting interface designed for ease of navigation and engagement. Featuring dynamic content tailored to each user's preferences and interactions, the Home page offers personalized recommendations, recent updates, and trending topics within the CodeStream community. Users are greeted with intuitive access points to essential features such as browsing recent posts, exploring new code snippets, and accessing personalized notifications. Through thoughtful UI/UX design, the Home section aims to streamline user interactions, foster community engagement, and provide a seamless experience that encourages exploration and participation across the platform. This home section option of this project is shown in figure (4.3)

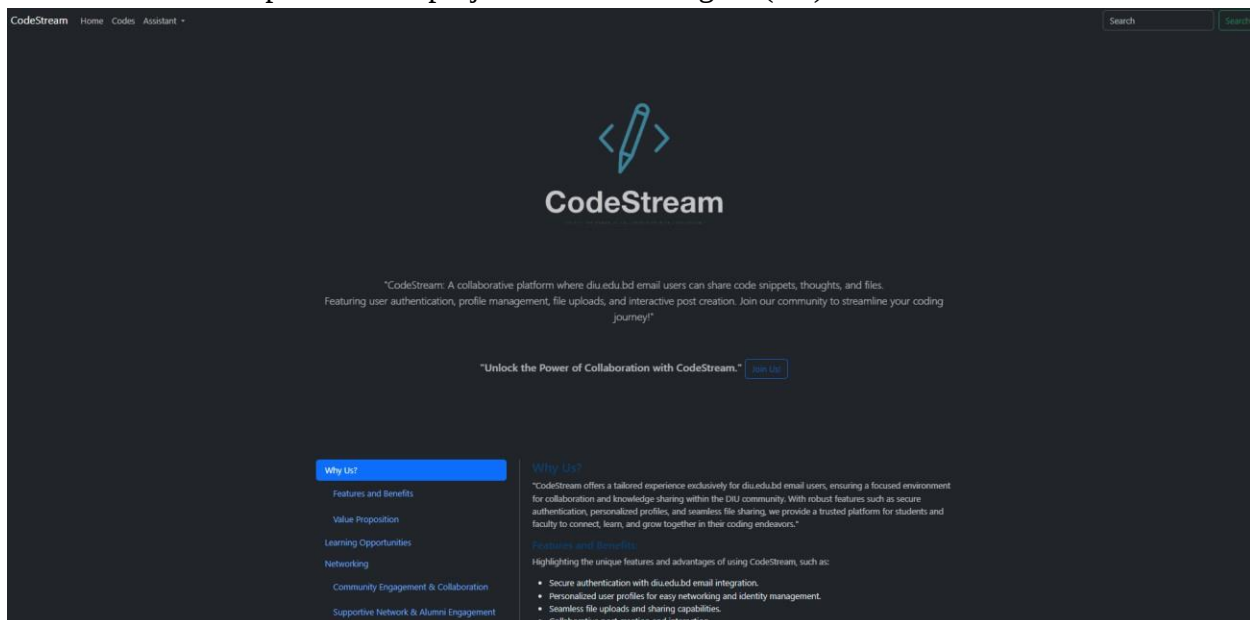
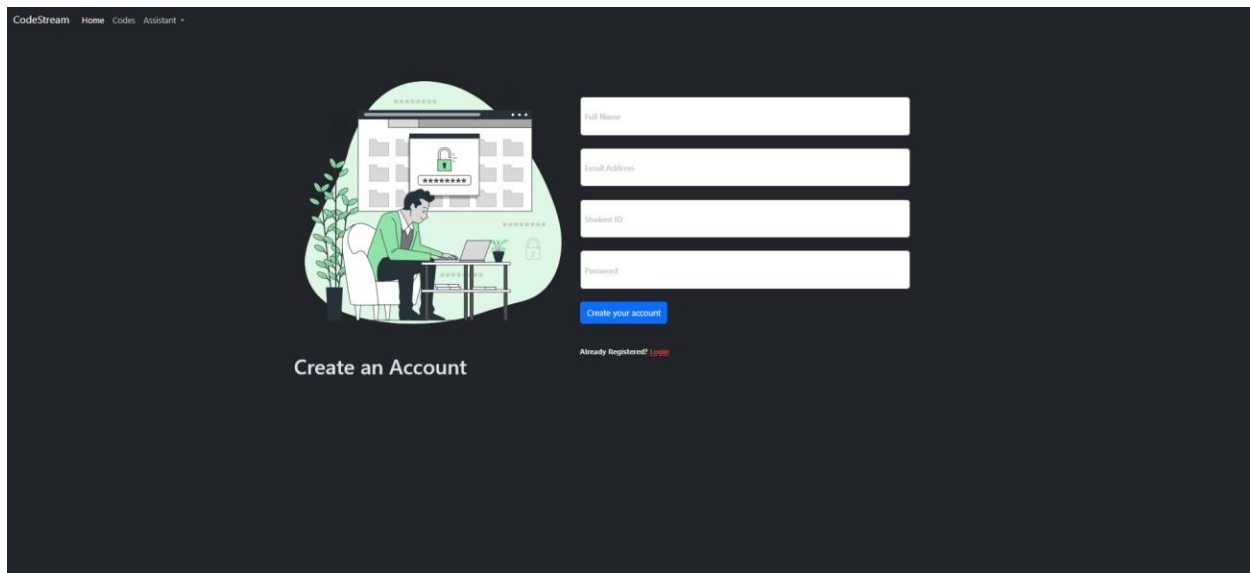


Figure 4.3 (Home Section)

4.12 Sign Up

The Sign Up process on CodeStream is streamlined to ensure ease of registration for users with a diu.edu.bd email address. Upon navigating to the Sign Up page, users are prompted to enter their Full Name, Student ID, Email, and Password. The system verifies the email domain to ensure eligibility and securely stores user credentials in our MongoDB database upon successful registration. User data is encrypted to maintain privacy and security. After signing up, users gain immediate access to their personalized profile, where they can manage their account settings, upload code snippets, and engage with the community. The Sign Up process is designed to be straightforward, providing a seamless entry point into the CodeStream platform for students and faculty of Daffodil International University, fostering a collaborative environment for learning and sharing in programming and technology. This project is shown in figure (4.4)



CodeStream Home Codes Assistant

Full Name

Email Address

Student ID

Password

Create your account

Already registered? [Login](#)

Create an Account

Figure 4.4 (Sign up)

4.13 Log in

The Login process on CodeStream ensures secure access for users with a diu.edu.bd email address. Users navigate to the Login page and enter their registered email and password. Upon submission, the system authenticates the credentials against our MongoDB database. If the credentials are valid, users are granted access to their personalized dashboard, where they can explore features such as browsing code snippets, updating their profile, and interacting with the community. The login mechanism employs encryption to protect user data during transmission and storage, maintaining confidentiality and integrity. For added security, failed login attempts are monitored and restricted, ensuring protection against unauthorized access. The seamless login experience is tailored to facilitate efficient navigation within the CodeStream platform, providing users with immediate access to resources and collaborative opportunities within the Daffodil International University community. This project is shown in figure (4.5)

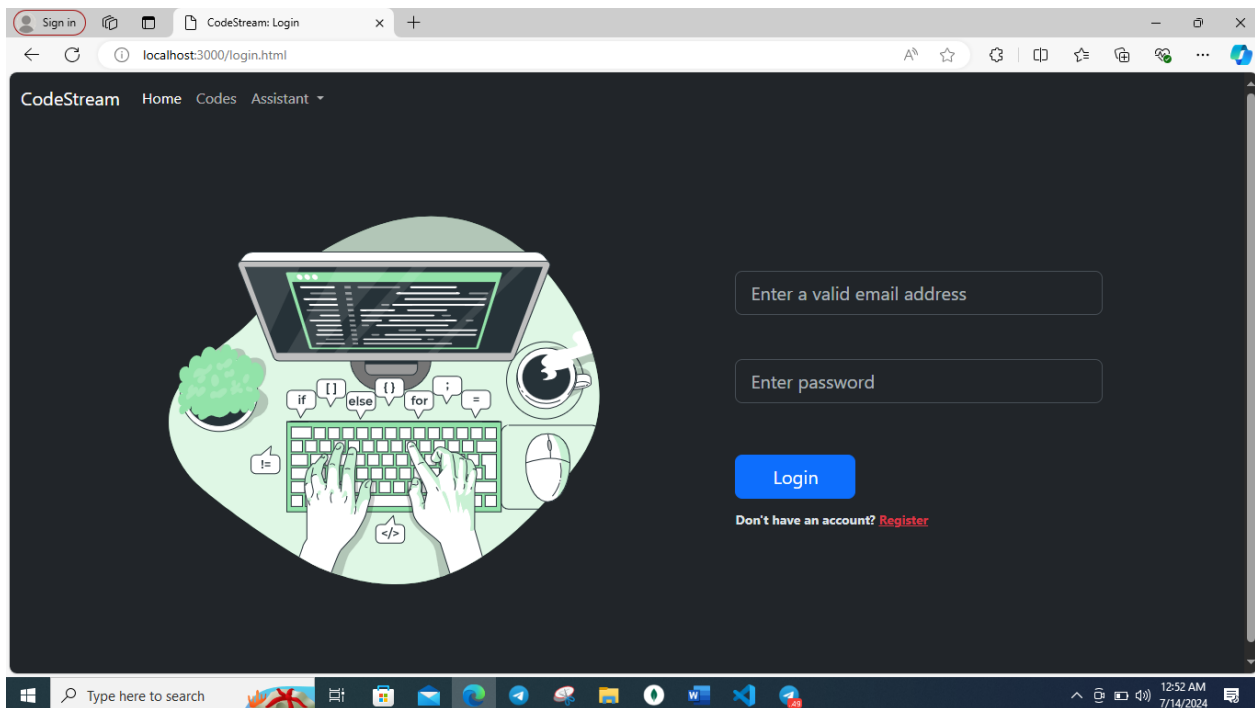


Figure 4.5 (Log in)

4.14 Codes Section & Upload Codes

The Codes section within CodeStream serves as a centralized repository where users can explore, contribute, and discover a diverse range of code snippets and projects. Designed to foster knowledge sharing and collaboration, this section allows users to browse through categorized code examples, upload their own projects via a dedicated form, and search for specific programming solutions. Each code snippet is tagged and categorized for easy navigation, facilitating efficient learning and exploration across various programming languages and technologies. Users can interact with posted codes, leave comments, and engage in discussions, enhancing their understanding and proficiency in coding practices. The platform ensures a user-friendly experience with features such as syntax highlighting, version control, and community-driven feedback, encouraging a vibrant exchange of ideas among members of the Daffodil International University community. Whether users are seeking inspiration for their projects or sharing their expertise, the Codes section serves as an invaluable resource for continuous learning and professional growth within the coding community. This is shown on Figure (4.6).

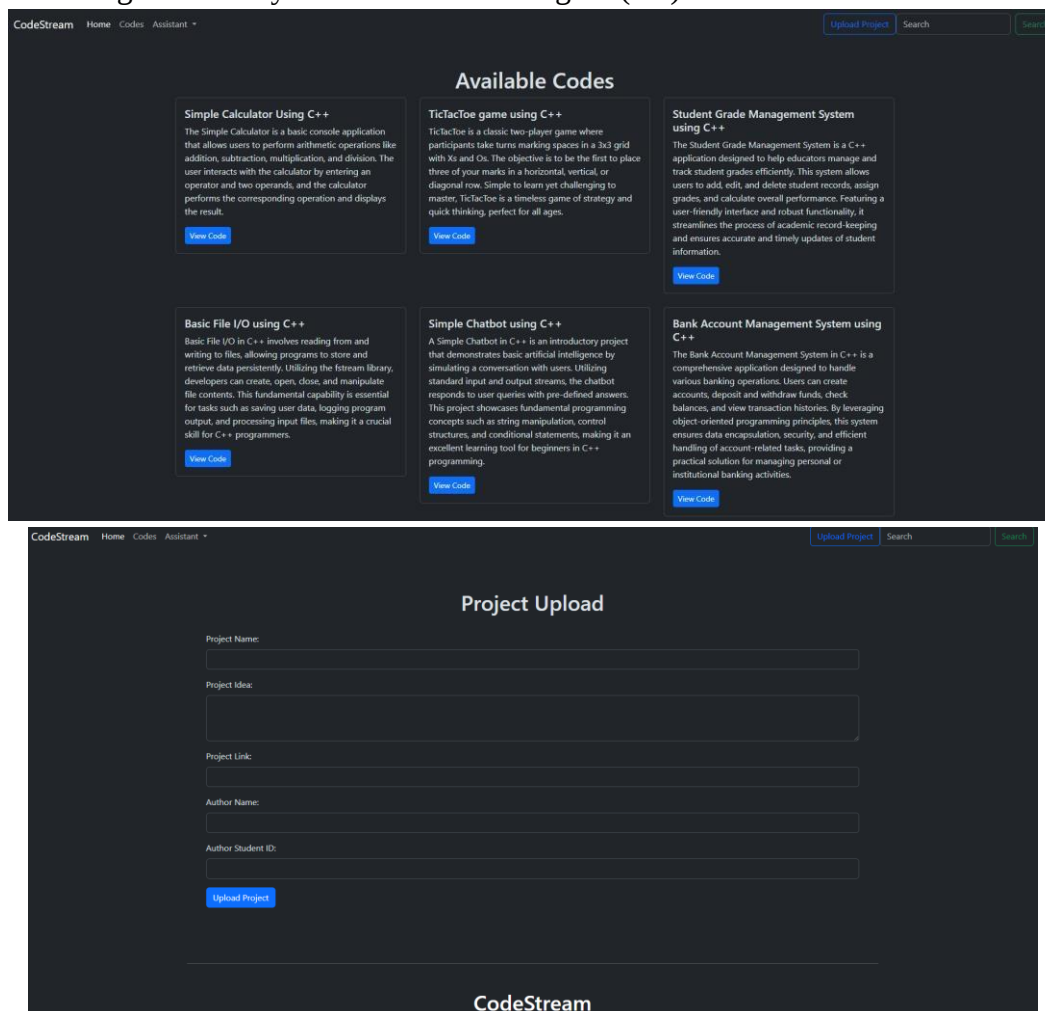


Figure 4.6 (Codes Section & Upload Codes)

4.15 Find Coders and Join as a Coder

In the "Find Coders" section of CodeStream, users can browse profiles of skilled programmers within the DIU community. Each profile showcases details such as the coder's name, student ID, batch, and contact options via email, phone, or Telegram. This feature aims to facilitate networking and collaboration among users seeking programming expertise or mentorship, fostering a supportive environment for learning and professional growth. Additionally, for those interested in joining as a coder, CodeStream offers opportunities to showcase programming skills and connect with potential collaborators or mentors. Users can explore profiles, reach out to fellow coders, and engage in discussions or collaborations on projects, hackathons, or coding challenges.

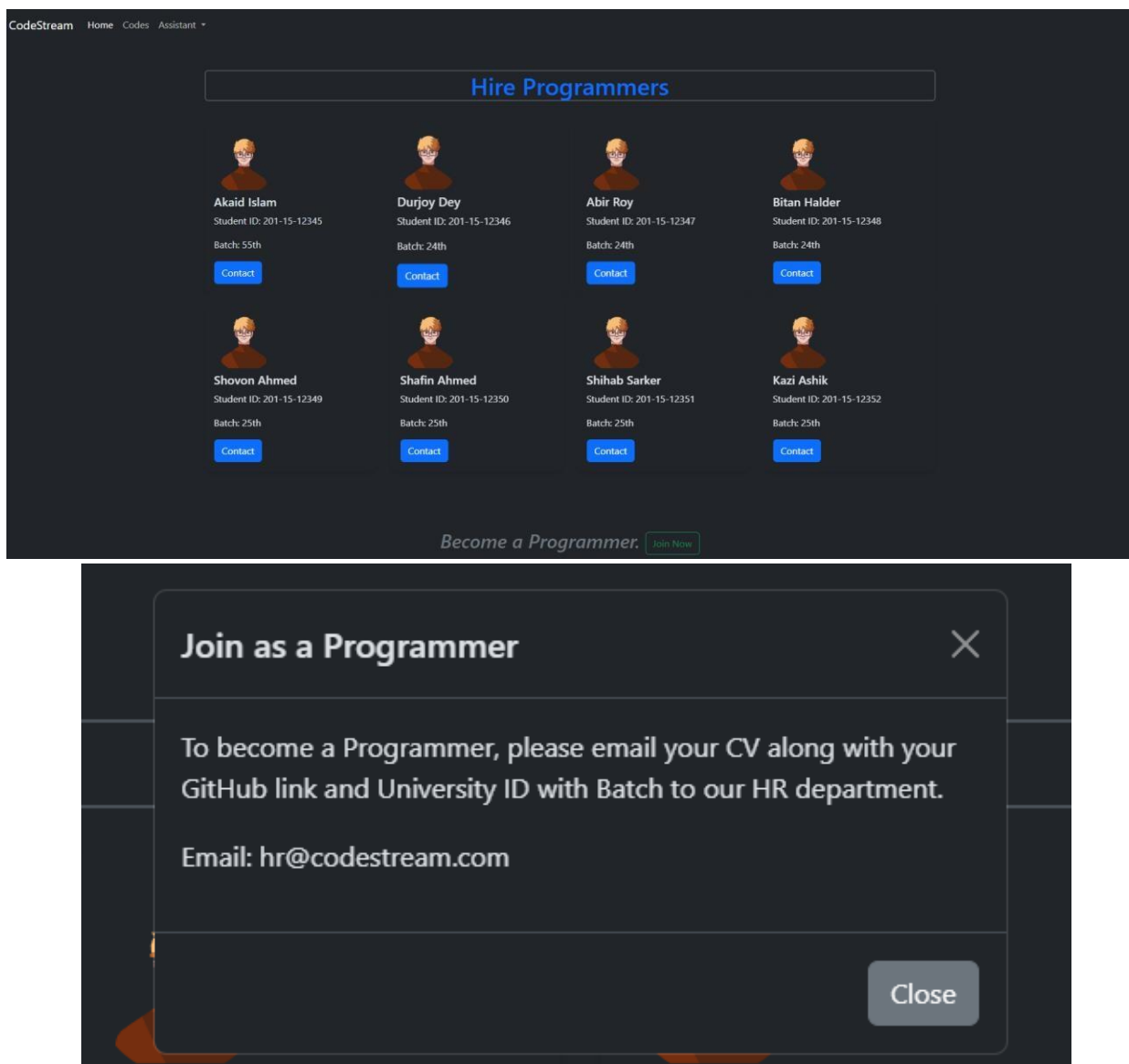
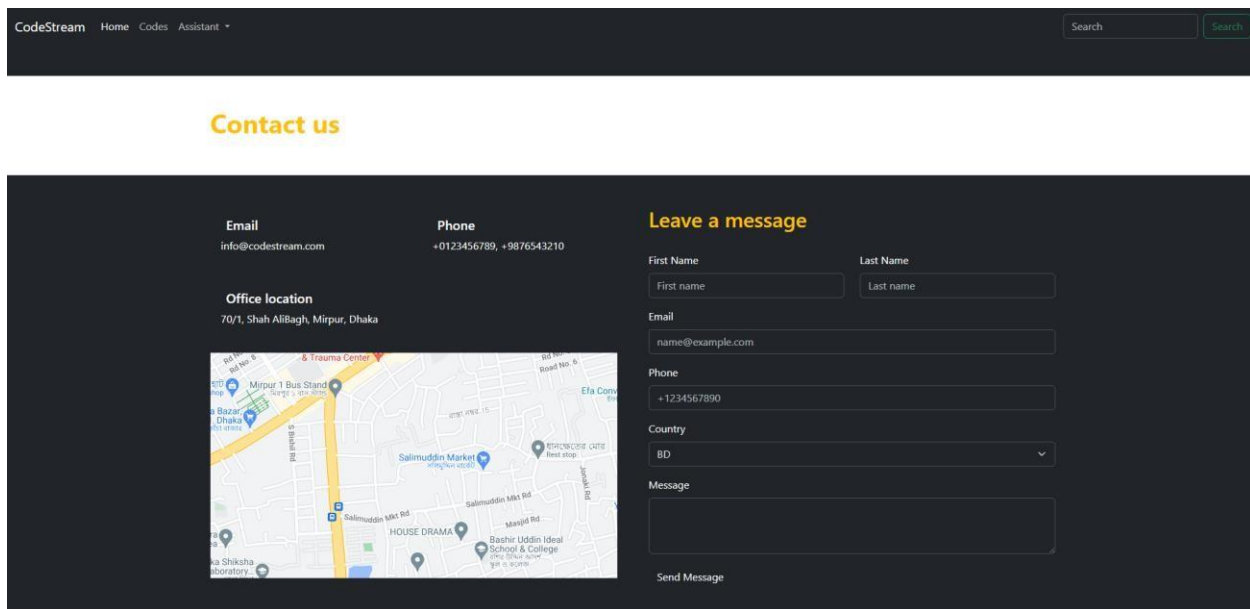


Figure 4.7 (Find Coders & Join as a Coder)

4.16 Contact

In the "Contact" section of CodeStream, users have the option to reach out for support or inquiries related to the platform. This section typically includes information on how to contact the CodeStream team via email or a contact form. Users can ask questions, report issues, or provide feedback regarding their experience on the platform. The contact page ensures that users have a direct channel to communicate with the support team, enabling swift resolution of queries or concerns. Additionally, it may provide links to social media profiles or other communication channels for broader interaction and community engagement. Overall, the contact section aims to enhance user satisfaction by ensuring effective communication and support throughout their interaction with CodeStream.



The screenshot displays the 'Contact us' page of the CodeStream application. At the top, a dark navigation bar contains the 'CodeStream' logo and links to 'Home', 'Codes', and 'Assistant'. A search bar is positioned on the right side of this bar. Below the navigation bar, the 'Contact us' heading is prominently displayed in a yellow font. The main content area is divided into three sections. On the left, the 'Email' section lists 'info@codestream.com', and the 'Phone' section lists '+0123456789, +9876543210'. Below these, the 'Office location' is specified as '70/1, Shah Alibagh, Mirpur, Dhaka', accompanied by a map showing the location. On the right, the 'Leave a message' section features a form with fields for 'First Name', 'Last Name', 'Email', 'Phone', 'Country', and 'BD'. A 'Message' text area and a 'Send Message' button are located at the bottom of this section.

Figure 4.8 (Contact)

4.17 About

The Rebate platform's "About" page combines frontend technologies like HTML5, CSS3, Bootstrap, and JavaScript for an engaging design. On the backend, Node.js, Express.js, and MongoDB work in harmony to power the server-side logic and securely store data. This concise yet compelling narrative unveils Rebate's mission, its commitment to diverse discounts, and the real-world impact it achieves. User testimonials and team introductions add a personal touch, while a glimpse into the platform's future vision reinforces Rebate's dedication to innovation. The seamless integration of frontend aesthetics and backend functionality creates an informative and visually appealing user experience, fostering connection and trust with visitors and users alike. This is shown on Figure (4.8).

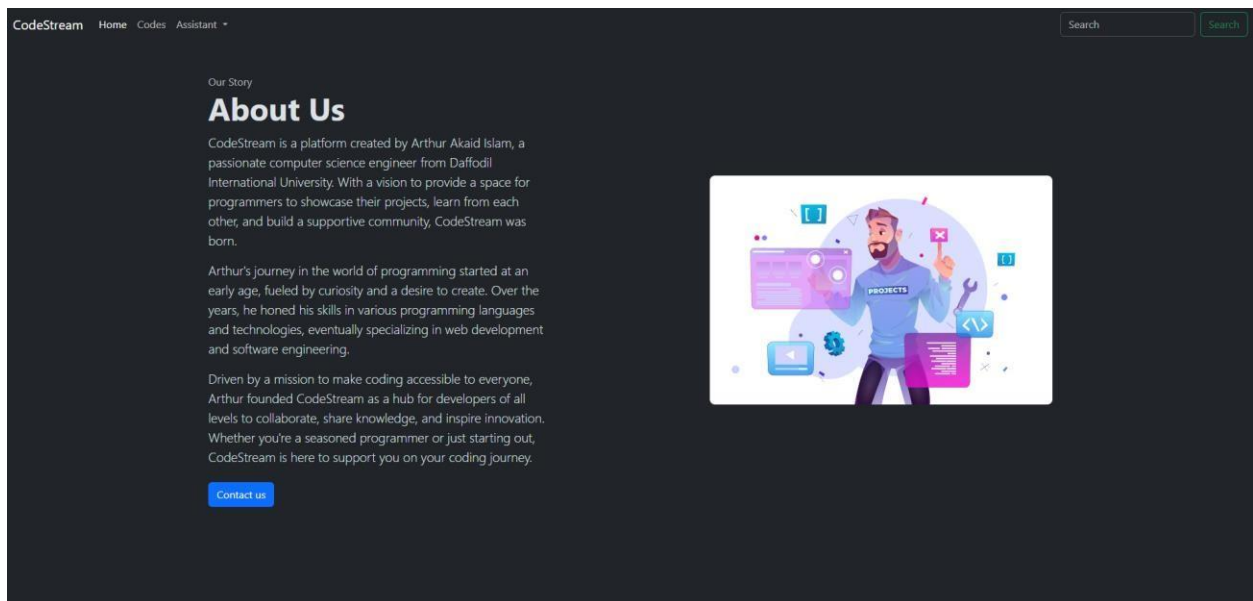


Figure 4.9 (About)

4.18 Database

MongoDB is a widely-used NoSQL database system known for its flexibility and scalability. As a document-oriented database, MongoDB stores data in BSON (Binary JSON) format, allowing for the storage of diverse and hierarchical data structures. It features a schema-less design, enabling dynamic and agile development by eliminating the need for predefined schemas. MongoDB is particularly well-suited for applications with evolving and complex data models, offering high performance and horizontal scalability. It supports efficient querying through a rich set of features, including indexing and aggregation, making it a popular choice for developers building modern, data-intensive applications. Database is shown on Figure(4.10)

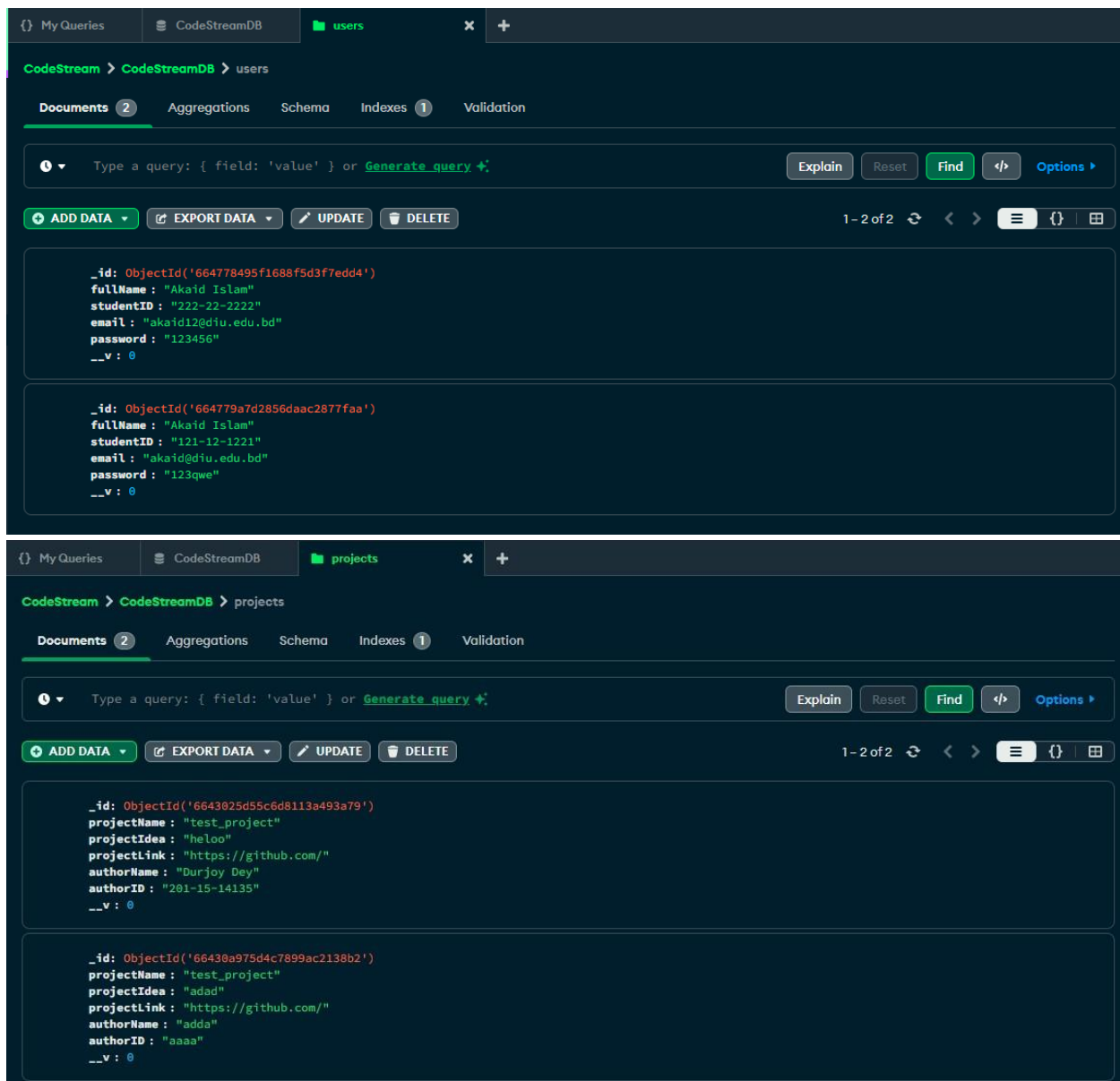


Figure 4.10 (Database)

4.19 FindCoders.js

Here is the JavaScript code for the FindCoders page.

```
public > ps findcoders.js @ d:\codeml\mod
document.addEventListener('DOMContentLoaded', function () {
  const profiles = [
    // Add profiles here...
    { name: 'Akaid Islam', id: '201-15-12345', batch: '55th', email: 'akaid@diu.edu.bd', phone: '+1234567890', telegram: '@akaid', image: 'profile-images/profile1.svg' },
    { name: 'Durjoy Dey', id: '201-15-12346', batch: '24th', email: 'durjoy@diu.edu.bd', phone: '+1234567890', telegram: '@durjoy', image: 'profile-images/profile1.svg' },
    { name: 'Abir Roy', id: '201-15-12347', batch: '24th', email: 'abir@diu.edu.bd', phone: '+1234567890', telegram: '@abir', image: 'profile-images/profile1.svg' },
    { name: 'Bilal Haldar', id: '201-15-12348', batch: '24th', email: 'bilal@diu.edu.bd', phone: '+1234567890', telegram: '@bilal', image: 'profile-images/profile1.svg' },
    { name: 'Shovan Ahmed', id: '201-15-12349', batch: '25th', email: 'shovan@diu.edu.bd', phone: '+1234567890', telegram: '@shovan', image: 'profile-images/profile1.svg' },
    { name: 'Shafin Ahmed', id: '201-15-12350', batch: '25th', email: 'shafin@diu.edu.bd', phone: '+1234567890', telegram: '@shafin', image: 'profile-images/profile1.svg' },
    { name: 'Shihab Sarker', id: '201-15-12351', batch: '25th', email: 'shihab@diu.edu.bd', phone: '+1234567890', telegram: '@shihab', image: 'profile-images/profile1.svg' },
    { name: 'Kazi Ashik', id: '201-15-12352', batch: '25th', email: 'ashik@diu.edu.bd', phone: '+1234567890', telegram: '@ashik', image: 'profile-images/profile1.svg' }
  ];

  const container = document.getElementById('profile-container');

  profiles.forEach(profile => {
    const card = document.createElement('div');
    card.className = 'col-md-6 col-lg-3';

    card.innerHTML = `
    <div class="card profile-card">
      <div class="card-body">
        
        <h5 class="card-title">${profile.name}</h5>
        <p class="card-text">Student ID: ${profile.id}</p>
        <p class="card-text">Batch: ${profile.batch}</p>
        <button type="button" class="btn btn-primary contact-btn" data-bs-toggle="modal" data-bs-target="#contactModal" data-email="${profile.email}" data-phone="${profile.phone}" data-telegram="${profile.telegram}">Contact</button>
      </div>
    </div>
    `;

    container.appendChild(card);
  });

  // Function to populate contact details modal
  const contactModal = document.getElementById('contactModal');
  contactModal.addEventListener('show.bs.modal', function (event) {
    const button = event.relatedTarget;
    const email = button.getAttribute('data-email');
    const phone = button.getAttribute('data-phone');
    const telegram = button.getAttribute('data-telegram');
    const modalBody = contactModal.querySelector('.modal-body');

    let contactDetailsHTML = `<p>To Contact Please Follow Any</p>`;
    if (email) {
      contactDetailsHTML += `<p>Email: ${email}</p>`;
    }
    if (phone) {
      contactDetailsHTML += `<p>Phone: ${phone}</p>`;
    }
    if (telegram) {
      contactDetailsHTML += `<p>Telegram: ${telegram}</p>`;
    }

    modalBody.innerHTML = contactDetailsHTML;
  });
});
```

Figure 4.11 (Categories.js)

4.20 App.js

We have declared Express, bodyParser, mongoose, and created a localhost environment here in the app.js file. and the MongoDB connection as well.

```
const express = require('express');
const bodyParser = require('body-parser');
const mongoose = require('mongoose');
const path = require('path');
const session = require('express-session');
const app = express();
const PORT = process.env.PORT || 5000;

// Connect to MongoDB
mongoose.connect('mongodb://localhost:27017/CodemlStreamDB', {
  useNewUrlParser: true,
  useUnifiedTopology: true,
});

const db = mongoose.connection;
db.on('error', console.error.bind(console, 'MongoDB connection error:'));
db.once('open', () => {
  console.log('Connected to MongoDB');
});

// Define Mongoose schema and model
const projectSchema = new mongoose.Schema({
  projectName: String,
  projectIdea: String,
  projectLink: String,
  authorName: String,
  authorID: String,
});
const Project = mongoose.model('Project', projectSchema);

// Middleware
app.use(bodyParser.json());
app.use(bodyParser.urlencoded({ extended: true }));
app.use(express.static(path.join(__dirname, 'public')));

// Route for serving the index.html file
app.get('/', (req, res) => {
  res.sendFile(path.join(__dirname, 'public', 'index.html'));
});

// Route for handling project upload
app.post('/upload', async (req, res) => {
  try {
    const { projectName, projectIdea, projectLink, authorName, authorID } = req.body;

    // Create a new project document
    const project = new Project({
      projectName,
      projectIdea,
      projectLink,
      authorName,
      authorID,
    });

    // Save the project to MongoDB
    const saveProject = await project.save();
    console.log('Project saved to MongoDB: ', saveProject);
  } catch (error) {
    console.error('Error saving project: ', error);
  }
});
```

Figure 4.12 (App.js)

4.21 App.js/Login and Signup

Here it shows the connection of login and signup page in the backend under the app.js.

```
126 // Route to handle user login and signup
127 mongoose.connect('mongodb://localhost:27017/CodeStreamDB', { useNewUrlParser: true, useUnifiedTopology: true });
128
129
130 // Define schema and model for user
131 const userSchema = new mongoose.Schema({
132   fullName: String,
133   studentID: String,
134   email: String,
135   password: String
136 });
137 const User = mongoose.model('User', userSchema);
138
139 // Use body-parser middleware to parse form data
140 app.use(bodyParser.urlencoded({ extended: true }));
141
142 // Use express-session middleware
143 app.use(session({
144   secret: 'secret-key',
145   resave: false,
146   saveUninitialized: true
147 }));
148
149 app.use(express.static('public'));
150
151 app.get('/signup', (req, res) => {
152   res.sendFile(path.join(__dirname, 'public', 'signup.html'));
153 });
154
155 app.get('/login', (req, res) => {
156   res.sendFile(path.join(__dirname, 'public', 'login.html'));
157 });
158
159 // Route for handling signup
160 app.post('/signup', async (req, res) => {
161   const { fullName, studentID, email, password } = req.body;
162   if (!email.endsWith('@diu.edu.bd')) {
163     return res.status(400).send('Only DIU email addresses are allowed');
164   }
165   try {
166     const newUser = new User({ fullName, studentID, email, password });
167     await newUser.save();
168     res.redirect('/login');
169   } catch (err) {
170     console.error(err);
171     res.status(500).send('Error signing up');
172   }
173 });
174
175 // Route for handling login
176 app.post('/login', async (req, res) => {
177   const { email, password } = req.body;
178   try {
179     const user = await User.findOne({ email, password });
180     if (!user) {
```

Figure 4.13 (App.js/Login & Signup)

4.22 Style.css

Cascading Style Sheets, or CSS, is a basic online technology that controls how HTML texts are presented and laid out. Developers can manage a webpage's visual elements, such as colors, fonts, spacing, and placement, with CSS. It allows information and design to be separated, improving the visual appeal, consistency, and cross-device responsiveness of websites. By making it simple to alter the appearance and feel of online pages, CSS is essential for improving user experience.



```
public > css > signup.css > select.form-control
1 |
2 |
3 | .border-md {
4 |   border-width: 2px;
5 | }
6 |
7 |
8 | body {
9 |   min-height: 100vh;
10 | }
11 |
12 | .form-control:not(select) {
13 |   padding: 1.5rem 0.5rem;
14 | }
15 |
16 | select.form-control {
17 |   height: 52px;
18 |   padding-left: 0.5rem;
19 | }
20 |
21 | .form-control::placeholder {
22 |   color: #ccc;
23 |   font-weight: bold;
24 |   font-size: 0.9rem;
25 | }
26 |
27 | .form-control:focus {
28 |   box-shadow: none;
29 | }

index.html  signup.css  findcoders.css X
public > css > findcoders.css > .card
1 | .profile-card {
2 |   transition: transform 0.2s;
3 | }
4 |
5 | .profile-card:hover {
6 |   transform: scale(1.05);
7 | }
8 |
9 | .card {
10 |   border: none;
11 |   border-radius: 15px;
12 |   box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
13 | }
```

Figure 4.14 (Style.css)

CHAPTER 5

Testing and Implementation

5.1 Introduction

The testing and implementation phase in the development of CodeStream is pivotal in ensuring the platform's functionality and reliability. It encompasses rigorous testing methodologies such as unit testing, integration testing, and system testing to validate the system's performance across various scenarios and user interactions. Each component undergoes meticulous scrutiny to identify and resolve any issues or bugs before deployment, ensuring a robust and error-free experience for users. Simultaneously, the implementation process focuses on seamless integration of new features and updates, following a systematic approach that includes planning, coding, and deployment strategies. This phase aims to enhance scalability, optimize performance, and maintain system stability, thereby delivering a secure and efficient platform for code sharing and collaboration on CodeStream.

5.2 Back-end design implementation

The back-end design implementation of CodeStream is centered around leveraging Node.js and Express.js frameworks, coupled with MongoDB for database management. This architecture facilitates efficient data handling and seamless integration of user authentication, profile management, and content storage functionalities. Utilizing RESTful APIs ensures standardized communication between the front-end and back-end, enabling robust interaction with features like code uploads, user interactions, and administrative tasks. Implementing modular components and middleware enhances scalability and maintainability, while rigorous testing ensures reliability and security. The back-end design aims to support the platform's dynamic nature, accommodating future expansions and optimizations to meet evolving user needs on CodeStream.

5.3 Test Execution

In the realm of CodeStream, meticulous test execution is paramount to ensuring the reliability and seamless operation of its front-end and back-end systems. Front-end testing focuses on validating user interface elements, responsiveness across various devices, and compatibility across different browsers. This involves employing tools like Jest for unit testing to scrutinize individual component behavior, integration testing to ensure modules interact seamlessly, and end-to-end (E2E) testing using Cypress to simulate user workflows and validate overall application functionality. On the back-end, rigorous testing involves verifying API endpoint functionality, robust data validation, and seamless integration with MongoDB. Utilizing frameworks such as

Mocha and Chai facilitates thorough unit testing of specific functions and modules, while integration testing ensures cohesive interaction within the Node.js and Express.js framework. Security testing, including vulnerability assessments and penetration testing, is integral to fortify the application against potential threats and breaches, safeguarding user data and privacy. Adopting continuous integration (CI) and continuous deployment (CD) practices ensures that all code changes undergo comprehensive testing before deployment to production environments. This automated approach detects and resolves issues promptly, maintaining the stability, performance, and security of CodeStream throughout its lifecycle. By adhering to robust testing methodologies, CodeStream aims to provide a reliable, secure, and user-friendly platform that meets the diverse needs of its DIU community users effectively.

5.4 Unit Test

In CodeStream, integration testing is concerned with confirming how different modules and components work together to guarantee smooth operation throughout the program. We check the system's functionality under realistic settings using tools such as Enzyme for React components and Supertest for API endpoints. By simulating external dependencies and database interactions with mock data and fixtures, we can test integration points without impacting the live environment. These tests are essential for spotting discrepancies or potential problems with component integration, guaranteeing robustness and dependability all the way through the program. Our CI/CD pipeline includes automatic integration tests that make sure each deployment preserves the integrity and coherence of the whole CodeStream platform.

5.5 Integration test

After integrating the front-end and back-end code, we carry out comprehensive tests to guarantee a smooth integration between the two parts. We are primarily concerned with verifying the layout's integrity and closely examining any possible problems that might compromise the layout's functioning or aesthetic coherence. By using a thorough approach, we can quickly detect and resolve layout issues and make sure that the Rebate platform has a unified and aesthetically pleasing user experience.

5.6 Function Test

This procedure encompasses our first priority and takes care of all the necessary preconditions for the project. We carefully consider and satisfy the prerequisites, making sure that every goal is meticulously examined. Understanding the crucial role functions play in the system, our method places a strong emphasis on using functions correctly. Every departure from the planned functionality is painstakingly found and fixed, demonstrating our dedication to creating a reliable and effective system where every feature is essential to the project's success as a whole.

System Test Results:

Serial No	Objectives	Result
1	The system is running Positive or Negative	Positive
2	The system shows every menu option	Positive
3	All options work	Positive
4	Sign up & Sign out option work	Positive
5	Showing error message after given wrong username or password	Positive
6	Upload code function works	Positive
7	Join as a Programmer woks	Positive
8	View codes in code snippet	Positive

CHAPTER 6

Conclusion and Future Development

6.1 Conclusion

Codestream stands as a cornerstone in fostering a vibrant coding community at DIU, providing essential tools and a collaborative platform that enhances learning and knowledge sharing. With its intuitive interface and robust features like secure authentication and personalized profiles, Codestream has simplified the process of code sharing and collaboration among students and faculty. By facilitating seamless file uploads and interactions, Codestream not only supports academic endeavors but also nurtures a culture of innovation and teamwork within the DIU community.

6.2 Scope for Further Development

Moving forward, Codestream can enrich its offerings by integrating advanced features such as AI-driven code analysis for automated debugging and optimization, enhancing the platform's usability and efficiency. Additionally, expanding its repository search capabilities and implementing real-time collaborative tools would empower users to engage more deeply in joint projects and learning activities. Improving mobile responsiveness and developing a dedicated mobile application could further enhance accessibility, allowing users to participate in Codestream activities from anywhere at any time. Moreover, forging partnerships with industry leaders and enhancing job placement functionalities would provide invaluable career opportunities for users, reinforcing Codestream's role as a comprehensive platform for both academic and professional growth. By continuously evolving based on user feedback and technological advancements, Codestream aims to sustain its position as a pivotal platform for collaborative coding and community engagement at DIU.

CHAPTER 7

Reference:

Websites:

- [1] GitHub of the Website, available at << <https://github.com/> >>, last accessed on 27-06-2024 at 06:00 PM.
- [2] GitLab of the Website, available at << <https://about.gitlab.com/> >>, last accessed on 27-06-2024 at 06:05 PM.
- [3] Bitbucket of the Website, available at << <https://bitbucket.org/> >>, last accessed on 27-06-2024 at 06:10 PM.
- [4] Stack Overflow of the Website, available at << <https://stackoverflow.com/> >>, last accessed on 27-06-2024 at 06:15 PM.
- [5] CodePen of the Website, available at << <https://codepen.io/> >>, last accessed on 27-06-2024 at 06:20 PM.
- [6] Repl.it of the Website, available at << <https://replit.com/> >>, last accessed on 27-06-2024 at 06:25 PM.
- [7] LeetCode of the Website, available at << <https://leetcode.com/> >>, last accessed on 27-06-2024 at 06:30 PM.
- [8] HackerRank of the Website, available at << <https://www.hackerrank.com/> >>, last accessed on 27-06-2024 at 06:35 PM.
- [9] CSS-Tricks of the Website, available at << <https://css-tricks.com/> >>, last accessed on 27-06-2024 at 04:10 PM.
- [10] Bootstrap Documentation of the Website, available at << <https://getbootstrap.com/docs/5.3/getting-started/introduction/> >>, last accessed on 27-06-2024 at 04:20 PM.
- [11] Node.js Documentation of the Website, available at << <https://nodejs.org/en/docs/> >>, last accessed on 27-06-2024 at 04:35 PM.
- [12] Express.js Documentation of the Website, available at << <https://expressjs.com/> >>, last accessed on 27-06-2024 at 04:40 PM.
- [13] Software Testing Help of the Website, available at << <https://www.softwaretestinghelp.com/> >>, last accessed on 27-06-2024 at 05:20 PM.
- [14] Udacity of the Website, available at << <https://www.udacity.com/> >>, last accessed on 27-06-2024 at 07:05 PM.
- [15] Pluralsight of the Website, available at << <https://www.pluralsight.com/> >>, last accessed on 27-06-2024 at 07:10 PM.
- [16] Udemy of the Website, available at << <https://www.udemy.com/> >>, last accessed on 27-06-2024 at 07:15 PM.
- [17] CodeChef of the Website, available at << <https://www.codechef.com/> >>, last accessed on 27-06-2024 at 07:20 PM.
- [18] Project Euler of the Website, available at << <https://projecteuler.net/> >>, last accessed on 27-06-2024 at 07:25 PM.

Example:

- [19] Learn about Wikipedia, available at << <http://www.wikipedia.org/> >>, last accessed on 27-12-2023 at 03:40 PM.

pre_defense_14148

ORIGINALITY REPORT

12%
SIMILARITY INDEX

9%
INTERNET SOURCES

1%
PUBLICATIONS

8%
STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	2%
2	dspace.daffodilvarsity.edu.bd:8080 Internet Source	2%
3	open-innovation-projects.org Internet Source	1%
4	123dok.com Internet Source	1%
5	Submitted to Manipal University Student Paper	1%
6	Submitted to University of Greenwich Student Paper	<1%
7	Submitted to Durban University of Technology Student Paper	<1%
8	vocal.media Internet Source	<1%
9	Submitted to Midlands State University Student Paper	<1%