

An Internship Journey in Web and App Development

by

Sharara Samin Khan
18101341

This internship report is submitted to the Department of Computer Science and Engineering in partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
October 2025

© 2025. Brac University
All rights reserved.

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Sharara Samin Khan

18101341

Approval

The internship report titled “An Internship Journey in Web and App Development” submitted by Sharara Samin Khan (18101341) of Summer, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science in October 2025.

Examining Committee:

Supervisor:
(Member)

Arif Shakil

Lecturer
Department of Computer Science and Engineering
BRAC University

Thesis Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD

Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Non-Disclosure Agreement

This Non-Disclosure Agreement (the “Agreement”) is made and entered into on this 1st day of March, 2025 (“Effective Date”), by and between:

Baula Innovation, having its principal place of business at Plot 45, Navana Tower (4th floor), Gulshan 1, Dhaka 1212, (“Discloser” or “Baula Innovation”);

and

Sharara Samin Khan, a student at Brac University, (“Recipient”).

Collectively, the “Parties.”

0.0.1 Definition of Confidential Information

“Confidential Information” means any non-public information disclosed by the Discloser to the Recipient, whether oral, written, electronic, or otherwise, including but not limited to information related to legal materials, software development and design, business or software architecture, software not yet known to the public, and information about clients or prospective clients. Confidential Information does not include information that (a) is or becomes publicly available without breach of this Agreement; (b) is lawfully received from a third party without restriction and without breach of any obligation; or (c) is independently developed by the Recipient without reference to Confidential Information.

0.0.2 Obligations of Recipient

The Recipient agrees:

- Not to disclose any Confidential Information or any information derived therefrom to any third party, except as required by law (in which case Recipient shall, where legally permissible, notify the Discloser promptly).
- To use the Confidential Information solely for the purpose for which it was disclosed.
- To take all reasonable precautions, at least as great as the precautions taken to protect its own confidential information, to prevent unauthorized use or disclosure.
- To immediately notify the Discloser of any unauthorized disclosure or use.

0.0.3 Ownership

All rights, title, and interest in and to the Confidential Information shall remain with the Discloser and/or its licensors. Nothing in this Agreement shall be construed as granting any rights to the Recipient, by license or otherwise, under any patent, copyright, trademark, or trade secret of the Discloser.

0.0.4 Term

The Recipient’s duty to maintain the confidentiality of Confidential Information shall survive for a period of three (3) years after the Effective Date or until such information no longer qualifies as Confidential Information, whichever occurs first.

0.0.5 General Provisions

- This Agreement shall be governed by and construed in accordance with the laws of Bangladesh.
- Any failure by either Party to enforce any provision of this Agreement shall not constitute a waiver of that provision or any other provision.
- If any provision of this Agreement is held invalid or unenforceable, the remaining provisions shall continue in full force and effect.
- This Agreement may not be assigned by either Party without the prior written consent of the other Party.

0.0.6 Execution

IN WITNESS WHEREOF, the Parties have executed this Agreement as of the Effective Date

Baula Innovation

Signature Date

Sharara Samin Khan Signature Date

Abstract

This report provides an overview of my internship at Baula Innovations, a technology startup located in Dhaka, where I served as a Web and App Development Intern. Throughout my time there, I assisted in developing Appayon, a digital food delivery service that specializes in home catering. The internship aimed to enhance my knowledge and application of contemporary web development methodologies, particularly Agile, and to engage in backend development as a component of the app creation process. During my role, I worked on various features of the platform, which allowed me to gain valuable insights into the complete product development life cycle and the practical challenges associated with constructing scalable web applications. This report details my contributions, the technologies employed, and the challenges faced during my internship experience.

Keywords: Back-end; ASP.NET Core; Database; Web Development; Agile; Appayon

Acknowledgment

Praise be to Allah the Almighty.

I sincerely thank Mr. Arif Shakil for his supervision and guidance throughout the preparation of this report.

I express my heartfelt appreciation to my parents for their unwavering support and effort to guide me through every stage of my life.

Lastly, I would like to express my sincere gratitude to Baula Innovations for giving me the opportunity to complete my internship as a web development intern.

Table of Contents

Declaration	i
Approval	ii
Ethics Statement	iii
0.0.1 Definition of Confidential Information	iii
0.0.2 Obligations of Recipient	iii
0.0.3 Ownership	iii
0.0.4 Term	iii
0.0.5 General Provisions	iv
0.0.6 Execution	iv
Abstract	v
Acknowledgment	vi
Table of Contents	vii
List of Figures	ix
Nomenclature	ix
1 Introduction	1
1.1 Company Profile	1
1.2 Product Overview – Appayon	2
1.3 Objectives of the Internship	3
1.4 Reason for Choosing Baula Innovations and the Technology Stack . .	4
1.5 Scope of Work	4
1.6 Team Structure	4
1.7 Methodology	5
1.8 Report Outline	5
2 Technical Background	6
2.1 Overview of Technical Environment	6
2.2 Technologies and Tools Used	6
2.3 System Architecture and Design Patterns	7
2.4 Development Methodology (Agile Approach)	8
2.5 Target Technology Stack	9
2.6 Technical Resources Consulted	10
2.7 Summary	10

3	Projects and Responsibilities	11
3.1	Overview of Assigned Tasks	11
3.2	Backend Development (ASP.NET Core, C#)	11
3.3	Database Design and Management (MySQL)	13
3.4	API Development and Testing (Postman)	13
3.5	Frontend Integration and UI Collaboration	15
3.6	Version Control and Team Collaboration (Git, Azure Repos)	16
3.7	Problem Solving and Debugging	16
3.8	Key Learning Moments	17
4	Internship Outcomes and Reflections	18
4.1	Technical Skills Acquired	18
4.2	Professional and Interpersonal Development	19
4.3	Exposure to Industry Practices	19
4.4	Application of Academic Knowledge	20
4.5	Overall Learning Experience	20
5	Challenges and Limitations	22
5.1	Technical Challenges	22
5.2	Communication and Remote Work Challenges	23
5.3	Time Management and Personal Constraints	24
5.4	Limitations of the Internship Scope	24
6	Conclusion and Future Goals	25
6.1	Summary of Internship Experience	25
6.2	Recommendations for Baula Innovations and Appayon	25
6.3	Future Career Goals	26
	Bibliography	28

List of Figures

1.1	Baula Innovations Company Logo	1
1.2	Baula Innovations Company Logo	2
1.3	System Architecture of Appayon Platform	3
2.1	Layered Architecture of Appayon Backend	8
2.2	Agile Workflow (Sprint Cycle Overview)	9
2.3	Sprint Task Tracking on Azure Boards	9
2.4	Sprint Planning and Tracking via Azure DevOps	9
3.1	Example of API Implementation in ASP.NET Core (Promos Controller)	12
3.2	MenuItems Table Structure in MySQL Workbench	13
3.3	API Testing Using Postman (GET Request Example)	14
3.4	Customer View of Food Item Description on Appayon Client	15
3.5	Appayon Admin Dashboard Overview	15
3.6	Version Control Workflow on Azure Repos	16
5.1	Example of Debugging and Log Tracing in Visual Studio Using Serilog	22

Chapter 1

Introduction

1.1 Company Profile



Figure 1.1: Baula Innovations Company Logo

Baula Innovations[1]. is a technology startup founded in 2023 by S M Arifuzzaman, with Md. Omar Khasru and Nafiza Nusrat as co-founders. The company was established with the goal of creating innovative digital products that address real-world challenges through modern technology and user-centric design

The company's mission is *"to harness technological advancements to tackle real-world challenges, enhancing productivity and enriching lives"*. Baula Innovations [1]envisions becoming a global leader in technological innovation, with a strong commitment to excellence, sustainability, and positive societal impact.

The company's mission is *"to harness technological advancements to tackle real-world challenges, enhancing productivity and enriching lives."* Baula Innovations envisions becoming a global leader in technological innovation, with a strong commitment to excellence, sustainability, and positive societal impact.

At present, the company's flagship project is Appayon, a platform designed to connect home-based food service providers — such as caterers, bakers, and culinary entrepreneurs — with customers looking for quality, home-cooked meals. Unlike traditional food delivery apps that primarily partner with restaurants, Appayon would provide a dedicated marketplace for small, independent food businesses to manage orders and deliveries.

Baula Innovations specializes in software development, web and mobile applications, and digital solutions for niche markets in Bangladesh. The company operates fully remotely with a small but dynamic team of eight members. Its collaborative work

culture emphasizes transparency and continuous learning. Daily stand-up meetings, where team members share progress and challenges, help foster open communication and problem-solving.

In the growing digital marketplace of Bangladesh, Baula Innovations is growing as a smart and flexible tech company in Bangladesh, targeting niche sectors like home-based food services through its creative product offerings.

1.2 Product Overview – Appayon

This section presents an overview of Appayon, the main project I contributed to during my internship.

Appayon is a web and mobile-based platform that connects customers with home-based food service providers. It allows customers to explore a variety of meal options, place orders, and schedule deliveries—all within a smooth digital experience—and is made to support small-scale food enterprises, home caterers, and bakeries.

Appayon provides food entrepreneurs with a polished platform to handle orders, advertise products, and expand their clientele. Without the expense of running a restaurant or handling separate logistics, small enterprises can expand with the help of integrated order management and delivery services.

For customers, Appayon provides access to authentic, home-cooked meals that are rarely available through conventional restaurant-based delivery apps.



Figure 1.2: Baula Innovations Company Logo

Core features include:

- Vendor and customer registration
- Menu and order management
- Delivery tracking and status updates
- Review and rating system

The product is currently in the web development phase, along with mobile application development.

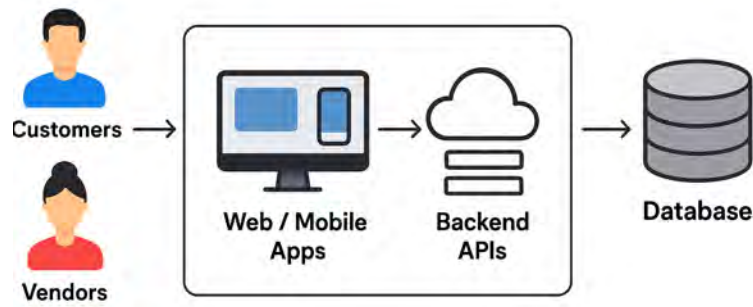


Figure 1.3: System Architecture of Appayon Platform

1.3 Objectives of the Internship

The main goal of my internship at Baula Innovations was to gain practical experience in full-stack software development, with a focus on backend development for the Appayon platform. My role centered on learning and applying modern .NET technologies, including ASP.NET Core and MySQL Server, to build scalable and efficient web services. **Technical Goals:**

- Develop and test backend components using ASP.NET Core and C#
- Manage and optimize MySQL databases
- Participate in system design and architecture discussions
- Collaborate with frontend developers to ensure seamless data flow
- Apply version control and coding best practices

Professional Goals:

- Strengthen teamwork and communication skills within a remote setup
- Adapt to Agile workflows and daily stand-up practices
- Understand the full product development cycle from concept to deployment

In addition to understanding full-stack software development, I aimed to deepen my knowledge of key backend concepts such as *dependency injection*, *middleware configuration*, *model binding*, and *database mapping*. I also set a goal to become comfortable using *logging and monitoring tools like Serilog*, and to understand *how layered architecture and repository patterns contribute to clean and scalable code design*.

1.4 Reason for Choosing Baula Innovations and the Technology Stack

I chose **Baula Innovations** as my internship organization because of its dynamic startup culture and commitment to developing impactful digital solutions for real-world problems. As a growing tech company, Baula provides a unique opportunity to work closely with experienced developers while contributing meaningfully to live projects. The Appayon platform particularly caught my attention since it supports my interest in software solutions that empower local businesses and communities in Bangladesh.

Another major reason behind my choice was the company's **modern technology stack**. Baula Innovations uses **ASP.NET Core** for backend development, **MySQL** for database management, and **Azure DevOps** for project collaboration — all of which are powerful, industry-relevant technologies. Since these tools are widely adopted in enterprise-level software development and offer strong scalability, security, and performance, I wanted to gain some hands-on experience in them.

By working with these technologies, I aimed to:

- Improve my skills in **C# and ASP.NET Core**, which are necessary for full-stack and enterprise systems.
- Learn **real-world database management** and **API development**.
- Understand Azure DevOps version control and Agile workflows.
- Experience **team collaboration** in a professional software environment.

All things considered, selecting Baula Innovations and its technology stack gave me the chance to integrate my academic background with corporate practices, creating the perfect setting for me to develop as a future software engineer.

1.5 Scope of Work

My internship's primary focus was on Appayon platform backend development and system integration. I helped manage database relationships, implement APIs, and ensure secure communication between system layers. My work also involved assisting with frontend integration, performing testing through Postman, and participating in daily stand-ups and sprint reviews.

1.6 Team Structure

At Baula Innovations, I worked as a backend development intern under the supervision of Md. Omar Khasru (Senior Backend Developer).

Team Composition:

- Founder & Lead: S M Arifuzzaman
- Senior Backend Developer: Md. Omar Khasru

- Senior Frontend Developer: Nafiza Nusrat
- Junior Frontend Developers: Sadat Ahmed and Tabassum Chowdhury

Interns: Sharara Khan (myself, backend), Farjana Yeasmin Sumaiya (backend), and Arfin Azad (frontend)

Teamwork at Baula Innovations is truly collaborative. While each member specializes in a particular area, our daily check-ins and regular feedback sessions ensure that everyone stays aligned and progress remains continuous.

1.7 Methodology

The internship followed an Agile development methodology. The team conducted daily morning stand-ups through Microsoft Teams, where each member discussed progress, upcoming tasks, and challenges.

All project management and sprint tracking were handled using Azure DevOps, which helped assign, monitor, and review tasks efficiently. Code collaboration was managed through Git-based repositories, and all pull requests underwent review before merging.

This iterative and feedback-driven approach provided flexibility, encouraged collaboration, and allowed continuous improvement throughout the internship period.

1.8 Report Outline

This report is organized into six chapters.

- Chapter 1 introduces the company, product, internship objectives, team structure, and methodology.
- Chapter 2 presents the technical background, tools, and development environment used.
- Chapter 3 explains the specific projects and responsibilities undertaken during the internship.
- Chapter 4 discusses internship outcomes and key reflections.
- Chapter 5 highlights challenges and limitations.
- Chapter 6 concludes with overall insights and future career goals.

Chapter 2

Technical Background

2.1 Overview of Technical Environment

Described in this chapter are the core technologies, tools, development practices, and architectural patterns used while interning at Baula Innovations.

As mentioned earlier, I , as an intern in development, worked with a full-stack environment that emphasized scalability, security, and maintainability.

The project Appayon combined modern backend frameworks, responsive frontend design, and collaborative cloud-based development practices.

The resources consulted and the main technical issues resolved during the development process are also included in this section.

2.2 Technologies and Tools Used

Programming Languages and Frameworks

During the internship, I worked primarily with the following languages and frameworks:

- C# for backend development using ASP.NET Core.
- JavaScript, HTML, and CSS for frontend development and styling.
- Razor Pages and Blade templates (within the ASP.NET ecosystem) for dynamic web interfaces.
- Bootstrap and Semantic UI for building responsive and consistent user interfaces.

Backend Technologies

- ASP.NET Core was used to develop RESTful APIs and implement business logic.
- MySQL Server served as the main relational database management system.
- Postman was used to test and document API endpoints during development.

Frontend Technologies

- Bootstrap and Semantic UI were used for rapid UI development.
- The team also planned to adopt React and Blazor for frontend enhancements and React Native for mobile application development.

Version Control and Collaboration

- TortoiseGit and Azure Repos (Git) were used for source-code management and version control.
- Pull Requests via Azure DevOps facilitated team-based code reviews and collaborative merging.

Development Tools

- Microsoft Visual Studio for backend development.
- Azure DevOps for project tracking, code deployment, and build automation.
- Microsoft Teams for daily stand-up meetings and communication among team members.

2.3 System Architecture and Design Patterns

The development team followed a hybrid architecture model combining multiple modern patterns for maintainability and scalability.

- **Model-View-Controller (MVC) design pattern:** Used to separate concerns between user interface, data, and business logic, which improved maintainability and testability.
- **REST API-based communication:** Backend services exposed secure API endpoints to serve both web and mobile clients.
- **Layered Architecture:** The solution was organized into distinct layers—Controllers (UI logic), Services (business logic), and Repositories (data access)—to ensure modularity and reusability.

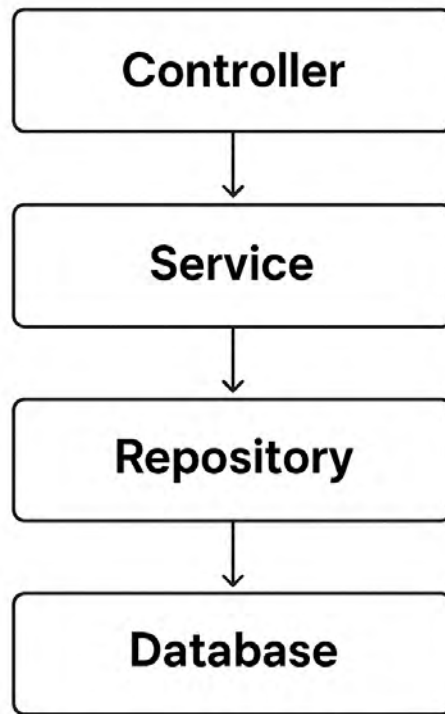


Figure 2.1: Layered Architecture of Appayon Backend

- **Authentication Middleware:** Implemented within the .NET pipeline to manage user roles and permissions, ensuring secure access.

To maintain modularity and reusability, the project also applied the **Repository and Unit of Work patterns**, which abstracted data access logic and improved maintainability. **Autofac** was used as the dependency injection container to manage object lifetimes efficiently across layers. Additionally, **Serilog** was integrated for structured logging and debugging, ensuring that exceptions and API events were traceable in both local and staging environments.

2.4 Development Methodology (Agile Approach)

Baula Innovations followed an Agile development methodology, characterized by iterative progress, teamwork, and constant feedback.

Every team member was able to exchange updates, pinpoint obstacles, and organise daily work through Microsoft Teams' daily morning stand-up meetings.

Azure Boards were used to manage sprint tasks, assign bugs, and monitor progress through user stories. Each sprint included backend deliverables like API completion, code review, and testing checkpoints. This process helped me understand *continuous integration*, version control workflows, and *how incremental updates are managed within Agile pipelines*.

In a fast-paced startup environment, this Agile process encouraged adaptability, accountability, and regular communication.



Figure 2.2: Agile Workflow (Sprint Cycle Overview)

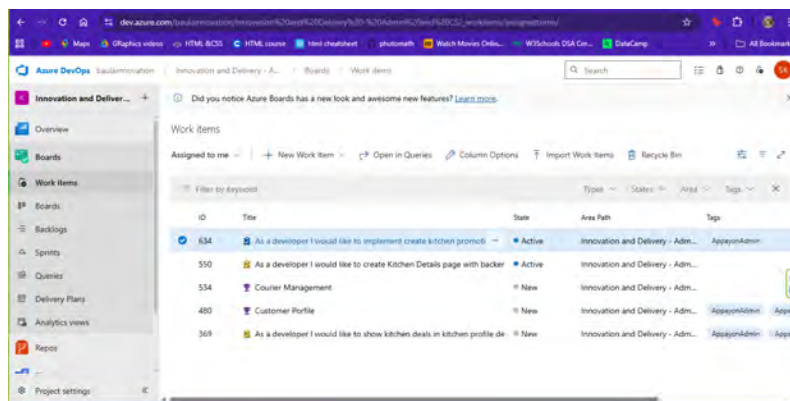


Figure 2.3: Sprint Task Tracking on Azure Boards

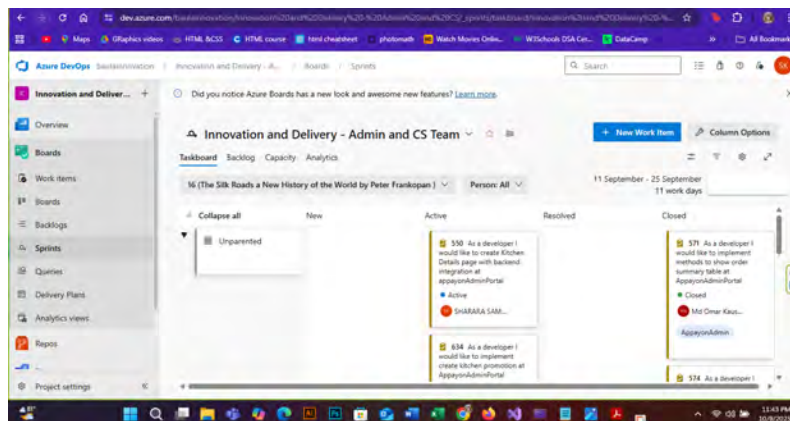


Figure 2.4: Sprint Planning and Tracking via Azure DevOps

2.5 Target Technology Stack

The Appayon platform was designed with a full-stack architecture suitable for scalable web and mobile development.

The following table summarizes the technology stack used or planned for the project:

- Backend: ASP.NET Core, C#

- Database: MySQL Server
- Frontend: React, Blazor, Bootstrap, Semantic UI
- Mobile React Native
- Hosting & Deployment: Azure Cloud / Local Server (for staging and testing)

This stack ensured maintainability, performance, and compatibility across multiple client platforms.

2.6 Technical Resources Consulted

Throughout the internship, I consulted several technical resources to improve understanding and solve development-related problems:

- **Microsoft .NET Documentation**[3] – for framework usage, middleware setup, and API implementation.
- **Stack Overflow** [4]– for debugging errors and exploring community-based solutions.
- **ReactJS Official Documentation** [2] – to understand modern component-based frontend principles.
- **W3Schools** [5]– for quick references to HTML, CSS, and JavaScript concepts.

These resources helped me learn framework conventions, syntax, and how to integrate different technologies into a cohesive solution.

2.7 Summary

This chapter examined the technical ecosystem, tools, and development practices used in the Appayon project at Baula Innovations.

Hands-on experience with contemporary full-stack programming was gained by working with a professional tech stack that included ASP.NET Core, MySQL Server, Bootstrap, and Azure, while adhering to Agile principles and MVC/REST architectures.

Overcoming challenges such as authentication setup and performance tuning deepened my understanding of production-level software engineering and prepared me for future enterprise-scale development work.

Chapter 3

Projects and Responsibilities

3.1 Overview of Assigned Tasks

I was mostly tasked with working on the backend development of Appayon, a web-based meal delivery service for home caterers, during my time at Baula Innovations. My responsibilities involved developing backend components, integrating databases, testing APIs, and collaborating with the frontend and DevOps teams.

I worked under the supervision of Md. Omar Khasru (Senior Backend Developer), who guided me in applying real-world development principles and improving my coding practices. Tasks were assigned through Azure DevOps, and progress was tracked in daily stand-up meetings.

My key responsibilities included:

- Developing and maintaining RESTful APIs for core features such as menu management, categories, ratings, and promotions.
- Managing MySQL databases — designing tables, writing queries, and optimizing data retrieval.
- Performing API testing using Postman.
- Collaborating with frontend developers to ensure smooth data flow between the interface and backend.
- Using Git for version control and participating in team-based code reviews.
- Debugging issues raised during testing and improving overall performance.
- These assignments gave me an in-depth understanding of how a real-world software project is structured and maintained in a professional environment.

3.2 Backend Development (ASP.NET Core, C#)

Backend development formed the core of my work during the internship. I used ASP.NET Core with C# to implement the application's logic and data-processing functionality. The backend followed the MVC and layered architecture, where controllers handled HTTP requests, services contained business logic, and repositories interacted with the database.

Some of the main features I contributed to included:

- **Menu Management:** Implemented APIs for creating, updating, deleting, and retrieving food items. Each menu item was linked to a category and included attributes such as name, price, description, availability, and image URL.
- **Category Management:** Developed endpoints for category creation and visibility toggling.
- **Promotional Offers:** Added functionalities for managing promotional discounts and enabling filtering by status (active/inactive).
- **Ratings and Reviews:** Worked on data retrieval and pagination logic for user feedback, ensuring optimized queries for large datasets.

I also applied best practices such as:

- **Dependency Injection** for maintaining modular code.
- **Asynchronous programming** (async/await) for efficient API responses.
- **Validation attributes** and exception handling **middleware** to ensure clean and secure API design. Through this process, I gained hands-on experience in backend architecture and the software development lifecycle.

Each controller I developed used *model binding* to process data from frontend requests, while *DTOs (Data Transfer Objects)* were used to ensure secure and clean data exchange between layers. Through the *Repository and Unit of Work patterns*, I learned how to separate concerns and make database operations more efficient. *Dependency Injection* simplified service management and testing, while *Serilog* provided structured logs for tracing API behavior during development.

```

208 [HttpPost]
209 [ValidateAntiForgeryToken]
210 public async Task<JsonResult> GetPromos(Guid KitchenId)
211 {
212     var tableModel = new DataTablesWithFilterRequestModel(Request);
213     var data = await this.promotionService.GetKitchenPromosAsync(tableModel, KitchenId);
214     return Json(data);
215 }
216
217 [HttpPost]
218 [ValidateAntiForgeryToken]
219 public async Task<ActionResult> AddPromos(PromosViewModel model)
220 {
221     try
222     {
223         if (ModelState.IsValid)
224         {
225             var result = await this.promotionService.AddPromosAsync(model);
226             if (result)
227             {
228                 return Json(new { success = true, message = "Promotion added successfully!" });
229             }
230             else
231             {
232                 return Json(new { success = false, message = "Failed to add promotion!" });
233             }
234         }
235         else
236         {
237             return Json(new { success = false, message = "Invalid data submitted!" });
238         }
239     }
240     catch (Exception ex)
241     {
242         logger.LogError(ex, "Error in AddPromos");
243         return Json(new { success = false, message = "An error occurred while adding the promotion." });
244     }
245 }

```

Figure 3.1: Example of API Implementation in ASP.NET Core (Promos Controller)

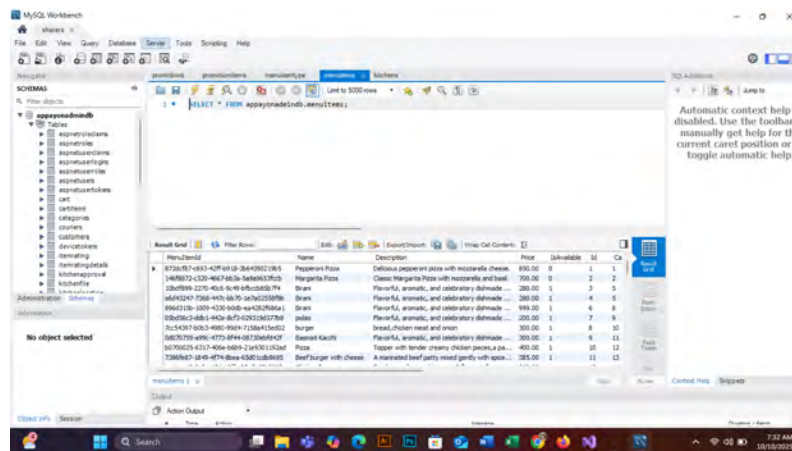
3.3 Database Design and Management (MySQL)

Database management was another important aspect of my role. The Appayon system used MySQL Server for data storage and retrieval. I learned how to design relational tables and establish relationships between entities such as Users, Menu-Items, Categories, Orders, and Reviews.

My tasks included:

- Writing SQL scripts to create, update, and query tables.
- Designing schema relationships using primary and foreign keys.
- Performing query optimization to reduce redundant data calls.
- Using Entity Framework Core (EF Core) as the ORM to map classes with database tables.
- Testing data integrity during CRUD operations. I also worked on ensuring that the database structure was consistent with the backend logic. For example, whenever new modules such as “Promotions” or “Ratings” were added, I updated corresponding migration files and seeded the database with test data.

These tasks gave me valuable experience with both direct SQL operations and ORM-based data handling in a professional project environment.



The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' pane displays a tree view of databases, with 'appayon' selected. The 'appayon' database is expanded, showing various tables including 'MenuItems'. The 'MenuItems' table is selected, and its structure is displayed in the main window. The table has columns: MenuItemId, Name, Description, Price, Availability, and Category. The 'MenuItems' table is highlighted in blue. The 'MenuItems' table structure is shown in the main window, displaying columns: MenuItemId, Name, Description, Price, Availability, and Category. The table contains several rows of data, including items like 'Pepperoni Pizza', 'Margherita Pizza', 'Bran', 'Pancake', 'Burger', 'Smoothie', 'Taco', and 'Beef Burger'.

MenuItemId	Name	Description	Price	Availability	Category
1	Pepperoni Pizza	Delicious pepperoni pizza with mozzarella cheese.	15.99	1	1
2	Margherita Pizza	Classic Margherita Pizza with tomato sauce and basil.	12.99	1	1
3	Bran	Pancake, aromatic, and celebratory delishious...	280.00	1	3
4	Pancake	Pancake, aromatic, and celebratory delishious...	280.00	1	4
5	Burger	Pancake, aromatic, and celebratory delishious...	999.00	1	5
6	Smoothie	Pancake, aromatic, and celebratory delishious...	200.00	1	6
7	Taco	Beef, chicken, meat and more...	300.00	1	7
8	Beef Burger	Pancake, aromatic, and celebratory delishious...	300.00	1	8
9	Taco	Tupper with tender shrimp chicken penne...	400.00	1	9
10	Beef Burger	A marinated beef patty mixed gently with sp...	385.00	1	10

Figure 3.2: MenuItems Table Structure in MySQL Workbench

3.4 API Development and Testing (Postman)

Testing and validating APIs was a routine part of my internship work. I used Postman to test various endpoints developed for Appayon.

The API development process generally followed these steps:

- Design API endpoints according to requirements.
- Implement the endpoint using controllers and services in ASP.NET Core.
- Run unit and integration tests using Postman to verify expected outputs.

- Document endpoints for frontend developers to integrate efficiently.

Some key tested APIs included:

- **Menu Item Management API** – to handle CRUD operations for food items.
- **Category API** – to retrieve visible categories for frontend dropdowns.
- **Promo API** – to manage promotional campaigns dynamically.
- **Review API** – to display ratings and feedback with pagination.

Testing through Postman allowed me to identify errors early in development and ensure that endpoints adhered to proper HTTP status codes and JSON response structures.

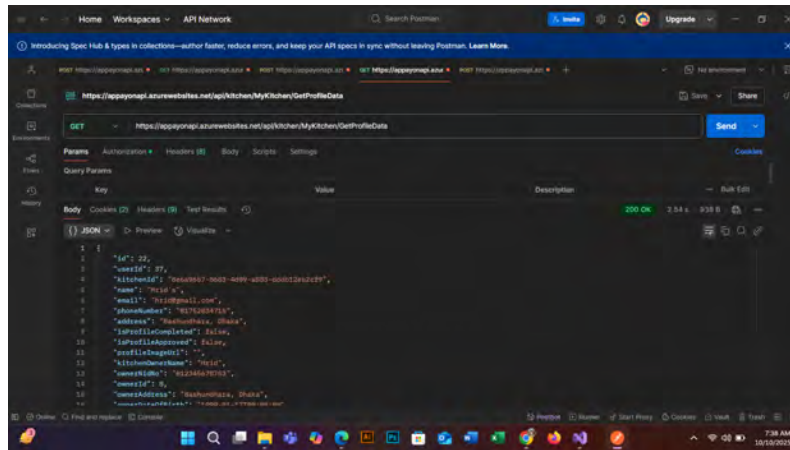


Figure 3.3: API Testing Using Postman (GET Request Example)

3.5 Frontend Integration and UI Collaboration

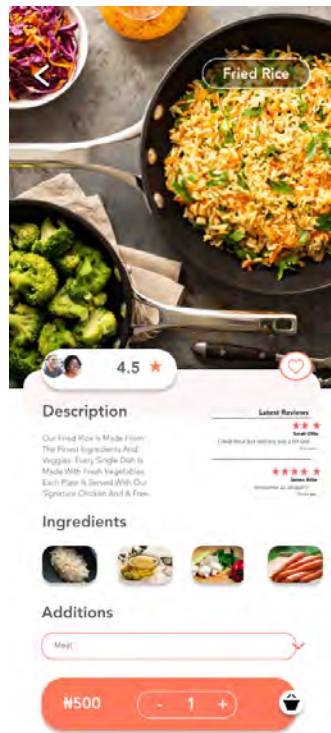


Figure 3.4: Customer View of Food Item Description on Appayon Client

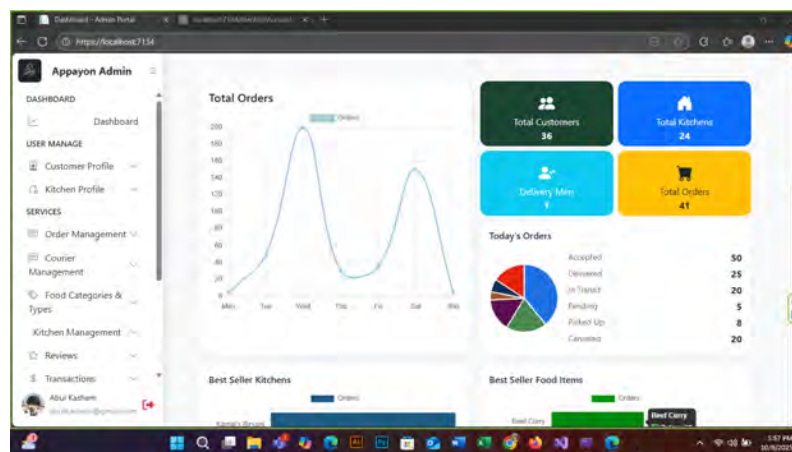


Figure 3.5: Appayon Admin Dashboard Overview

Although my main responsibility was backend development, I frequently collaborated with the frontend developers to ensure consistent data exchange between both layers.

Using Semantic UI and Bootstrap for layout design, the frontend team built dynamic pages that relied on my backend APIs. I often assisted them by:

- Explaining API endpoint behavior and request parameters.
- Adjusting JSON response formats for better frontend parsing.

- Testing integration results after frontend implementation.

Additionally, I gained exposure to AJAX and Fetch API mechanisms used by the frontend to call backend endpoints asynchronously. This collaboration helped me understand how backend logic directly affects user experience and application responsiveness.

3.6 Version Control and Team Collaboration (Git, Azure Repos)

For version control, Baula Innovations used Azure Repos along with Git to manage its projects. Every team member had an assigned branch for their tasks, and all code modifications were reviewed before being merged into the main branch. My responsibilities included:

- Regularly pushing updates to feature branches.
- Creating **pull requests** (PRs) for code review.
- Merging approved code after peer and supervisor verification.
- Resolving merge conflicts and maintaining branch consistency.

We also used **Azure DevOps Boards** to track daily progress, assign user stories, and review completed tasks. Transparency, accountability, and effective team communication were made possible by this workflow.

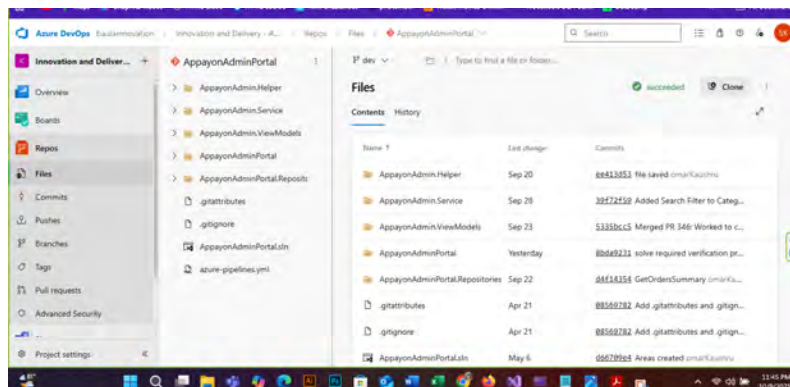


Figure 3.6: Version Control Workflow on Azure Repos

3.7 Problem Solving and Debugging

An essential component of the learning process was debugging. I encountered several issues, from broken queries and data mismatches to API response errors. With supervision, I have acquired knowledge about how to apply systematic debugging techniques.

Serilog logs and *Visual Studio's integrated debugging tools* were used to track errors. Watch windows, breakpoints, and immediate evaluations aided in identifying logic issues quickly. In order to properly address serialisation or validation issues during

API testing, I also learnt how to inspect HTTP request payloads and model binding mismatches.

Through these experiences, I became more skilled at swiftly recognising the root causes and effectively implementing solutions. The process has taught me how crucial it is to address software errors with patience, logical thinking, and documentation.

3.8 Key Learning Moments

Throughout my time working on Appayon, I gained several important lessons that shaped my professional and technical growth, some of which are:

- Understanding the complete workflow of enterprise-grade web development.
- Writing clean, reusable, and well-documented backend code.
- Managing team-based projects through Azure DevOps and Git.
- Applying Agile practices effectively in a remote working environment.
- Collaborating across roles—backend, frontend, and QA—to achieve common goals.

In addition to enhancing my technical proficiency, the real-world experience built confidence in my ability to operate in a team-oriented, production-level setting. It taught me how theoretical knowledge from coursework can be translated into applications and professional teamwork in actual life.

Chapter 4

Internship Outcomes and Reflections

4.1 Technical Skills Acquired

I greatly improved my skill set, during my internship, by putting my academic understanding to use in real-world development circumstances.

Key technical skills I developed, working on Appyaon, include:

- **Backend Development:** Practical experience using ASP.NET Core (C#) for building RESTful APIs, managing controllers, and implementing middleware for authentication and authorization.
- **Database Management:** Designed and maintained relational databases using MySQL, optimized queries, and applied ORM concepts through Entity Framework Core.
- **API Testing and Validation:** Used Postman to test, debug, and document APIs, ensuring accuracy and consistency across endpoints.
- **Architecture Understanding:** Gained a clear understanding of the MVC and Layered Architecture patterns used to separate concerns in large-scale projects.
- **Version Control and CI/CD:** Applied Git and Azure DevOps tools for source control, code reviews, and branch management, ensuring smooth collaboration and safe deployment processes. Additional technical competencies I developed include:
 - Applying **model binding and validation** to handle HTTP requests securely.
 - Understanding the **Repository and Unit of Work patterns** for data abstraction.
 - Managing environment-specific configurations through **appsettings.json** and dependency registration.

4.2 Professional and Interpersonal Development

My internship experience also helped me grow beyond technical knowledge — it developed my professionalism, adaptability, and teamwork skills. Working remotely in a startup environment required me to communicate clearly, manage my time efficiently, and collaborate with others from different technical backgrounds.

Professional development highlights:

- **Communication:** Regular stand-up meetings and code review sessions helped me express my ideas, report progress, and receive feedback constructively.
- **Team Collaboration:** Coordinating with frontend, backend, and QA members taught me how teamwork drives project success.
- **Time Management:** Balancing multiple tasks in parallel taught me how to plan daily goals effectively and meet deadlines under pressure.
- **Accountability:** Every task was assigned individually in Azure DevOps, which helped me take ownership of my work and track my progress systematically.

These experiences strengthened my interpersonal confidence and improved my ability to operate efficiently in a professional environment.

4.3 Exposure to Industry Practices

The internship introduced me to various real-world development practices that are standard in the software industry. I learned how professional teams organize their workflows, follow Agile methodologies, and maintain production-quality standards in every phase of development.

Some of the **industry-standard practices** I was exposed to include:

- **Agile Software Development:** Participating in daily stand-ups, sprint planning, and task review cycles to ensure iterative progress.
- **Code Review and Quality Control:** Submitting pull requests (PRs) and resolving feedback from senior developers before merging code into the main branch.
- **Version Control Discipline:** Following branching strategies (feature, develop, and main) and maintaining clean commit histories.
- **Testing and Documentation:** Writing test cases, validating API endpoints, and maintaining Postman collections for reference.
- **Continuous Learning Culture:** Adapting to new frameworks and solving unexpected problems independently while maintaining communication with the team.
- **Code Review Pipelines:** Creating pull requests and resolving review comments before merges.

- **Structured Logging:** Using Serilog to track system events and exceptions.
- **Layered Testing:** Observing how unit tests and integration tests ensure API reliability.
- **Configuration Management:** Maintaining environment-based settings and connection strings securely.

This exposure provided me with valuable insights into how professional software companies maintain efficiency, transparency, and quality in their operations.

4.4 Application of Academic Knowledge

My computer science curriculum gave me a solid foundation that came in very handy for the internship. I was able to apply a number of the concepts that I had studied in class to actual development projects.

Examples of academic knowledge applied:

- **Object-Oriented Programming (OOP):** Principles like inheritance, encapsulation, and polymorphism were used in C# while developing backend services.
- **Database Concepts:** Applied knowledge gained on normalization, foreign key relationships, and query optimization from Database Systems courses.
- **Software Engineering Principles:** Implemented modular design and version control practices taught in Software Engineering classes.
- **Data Structures and Algorithms:** Optimised database operations and API response times through the use of efficient data-handling techniques that were previously learned.
- **Web Technologies:** Built and styled interfaces using HTML, CSS, and JavaScript, supported by Bootstrap and Semantic UI. These were learnt from working on group projects.

I gained a fresh understanding of the usefulness of my undergraduate education after witnessing how these theoretical subjects were directly applied to software development in the actual world.

4.5 Overall Learning Experience

Overall, my internship at Baula Innovations was a transformative experience that strengthened my technical, professional, and personal skills. I learned not only how software is developed in an industry setting but also how collaboration, consistency, and problem-solving contribute to the success of a project.

This experience taught me:

- How to work in a **structured yet flexible startup environment**.

- The importance of **clear communication** and **task ownership**.
- The value of **writing clean, maintainable code** and **documenting work** for team understanding.
- How to adapt quickly, learn from mistakes, and continually improve.

The *Appayon* project allowed me to contribute meaningfully to a real digital product while learning technologies like ASP.NET Core, MySQL, and Azure DevOps. This hands-on experience has prepared me for future professional roles and strengthened my confidence as a software developer.

Chapter 5

Challenges and Limitations

5.1 Technical Challenges

As a web and app development intern, I encountered multiple technical challenges while working on the *Appayon* platform. Since this was my first professional exposure to large-scale software development, several aspects required independent learning, debugging, and problem-solving.

Key technical challenges included:

Learning ASP.NET Core Framework:

Initially, understanding the ASP.NET Core structure—controllers, dependency injection, middleware, and routing—was overwhelming. I overcame this by following official Microsoft documentation, online tutorials, and hands-on experimentation with sample projects. **Implementing Secure Authentication:**

Configuring **JWT (JSON Web Tokens)** for secure user authentication and role-based authorization was complex. It took multiple attempts to correctly handle token generation, validation, and user session management without breaking the middleware pipeline. **Database Integration with Entity Framework:**

Managing relationships between entities (such as kitchens, menu items, and users) in **MySQL** through **Entity Framework Core** initially caused several migration and query errors. Over time, I learned how to use migrations, navigation properties, and query optimization techniques effectively. **Debugging and API Testing:**

Testing and debugging APIs using **Postman** often revealed unexpected issues in request-response handling. By logging API activities using **Serilog** and analyzing server responses, I gradually developed an understanding of how to trace and resolve backend logic errors efficiently.

```
207 at Microsoft.EntityFrameworkCore.ChangeTracking.Internal.StateManager.SaveChangesAsync(StateManager stateManager, Boolean acceptAllChangesOnSuccess, CancellationTok
208 at MySQL.EntityFrameworkCore.Storage.Internal.MySQLExecutionStrategy.ExecuteAsync[TState,TResult](TState state, Func<TState,TResult> func, CancellationToken can
209 at Microsoft.EntityFrameworkCore.DbContext.SaveChangesAsync(Boolean acceptAllChangesOnSuccess, CancellationToken cancellationToken)
210 2023-10-22 22:48 [Info] Error sending rules and rules
211 Microsoft.EntityFrameworkCore.Update.Internal.UpdateBatchExecutor: Could not save changes. Please configure your entity type accordingly.
212 ---- MySQL.Data.MySqlClient.MySqlException (0x8000000d): Unknown column 'AdminId' in 'field list'
213 at MySQL.Data.MySqlClient.MySqlStream.ReadResultAsync(CancellationToken cancellationToken)
214 at MySQL.Data.MySqlClient.NativeDriver.GetResultAsync(Int32 affectedRows, Int64 insertId, Boolean executeAsync)
215 at MySQL.Data.MySqlClient.Driver.GetResultAsync(Int32 statementId, Int32 affectedRows, Int64 insertId, Boolean executeAsync)
216 at MySQL.Data.MySqlClient.Driver.GetResultAsync(Int32 statementId, Boolean force, Boolean executeAsync)
217 at MySQL.Data.MySqlClient.MySqlDataReader.NextResultAsync(CancellationToken cancellationToken)
218 at MySQL.Data.MySqlClient.MySqlDataReader.NextResultAsync(CancellationToken cancellationToken)
219 at MySQL.Data.MySqlClient.MySqlCommand.ExecuteReaderAsync(CommandBehavior behavior, Boolean executeAsync, CancellationToken cancellationToken)
220 at MySQL.Data.MySqlClient.MySqlCommand.ExecuteReaderAsync(CommandBehavior behavior, Boolean executeAsync, CancellationToken cancellationToken)
221 at MySQL.Data.MySqlClient.MySqlCommand.ExecuteReaderAsync(CommandBehavior behavior, Boolean executeAsync, CancellationToken cancellationToken)
222 at Microsoft.EntityFrameworkCore.Storage.RelationalCommand.ExecuteReaderAsync(RelationalCommandParameterObject parameterObject, CancellationToken cancellatio
223 at Microsoft.EntityFrameworkCore.Update.ReaderModificationCommandBatch.ExecuteAsync(IRelationalConnection connection, CancellationToken cancellationToken)
224 at Microsoft.EntityFrameworkCore.Update.Internal.BatchExecutor.ExecuteAsync(IEnumerable<ICommandBatch> commandBatches, IRelationalConnection connection, CancellationTok
225 at Microsoft.EntityFrameworkCore.Update.Internal.BatchExecutor.ExecuteAsync(IEnumerable<ICommandBatch> commandBatches, IRelationalConnection connection, CancellationTok
226 at Microsoft.EntityFrameworkCore.ChangeTracking.Internal.StateManager.SaveChangesAsync(List<IEntry> entriesToSave, CancellationToken cancellationToken)
227 at Microsoft.EntityFrameworkCore.Storage.RelationalCommand.ExecuteReaderAsync(RelationalCommandParameterObject parameterObject, CancellationToken cancellatio
228 at MySQL.EntityFrameworkCore.Storage.Internal.MySQLExecutionStrategy.ExecuteAsync[TState,TResult](TState state, Func<TState,TResult> func, CancellationToken can
229 at Microsoft.EntityFrameworkCore.DbContext.SaveChangesAsync(Boolean acceptAllChangesOnSuccess, CancellationToken cancellationToken)
230 at AppayonAdminPortal.Repositories.UnitOfWork.CommitAsync() in E:\Appayon\AppayonAdminPortal\AppayonAdminPortal\Repositories\UnitOfWork\UnitOfWork
231 at AppayonAdminPortal.Repositories.UnitOfWork.CommitAsync() in E:\Appayon\AppayonAdminPortal\AppayonAdminPortal\Repositories\UnitOfWork\UnitOfWork
232 at AppayonAdminPortal.Repositories.UnitOfWork.CommitAsync() in E:\Appayon\AppayonAdminPortal\AppayonAdminPortal\Repositories\UnitOfWork\UnitOfWork
233 at AppayonAdminService.Services.UserService.SaveMenuItemsAsync() in E:\Appayon\AppayonAdminPortal\AppayonAdminService\Services\UserService.cs:line 207
```

Figure 5.1: Example of Debugging and Log Tracing in Visual Studio Using Serilog

Database Schema Design and Data Relationships:

Designing a scalable and efficient database schema for Appayon was a key challenge, especially since the platform required managing multiple entities such as kitchens, menu items, promotions, and customer reviews.

Initially, some relationships—like those between Kitchen, MenuItem, and Promo tables—led to complex joins and data inconsistencies. To address this, I collaborated with my supervisor to refine the entity relationships using **Entity Framework Core** conventions such as **navigation properties** and **foreign key constraints**. This improved query readability and ensured proper referential integrity across tables. I also learned to use **data seeding** for testing and applied **migrations** effectively to keep schema changes synchronized with the database.

Performance Optimization:

As the platform was expected to scale, performance optimization became a priority. I learned to write more efficient queries using `Include()` for eager loading in **Entity Framework** and minimized redundant API calls.

In order to increase data retrieval speed and guarantee seamless frontend performance during testing, I also explored lazy loading and caching techniques. These technical challenges taught me the importance of patience, systematic debugging, and continuously seeking solutions through community documentation and peer collaboration.

5.2 Communication and Remote Work Challenges

Baula Innovations operates as a **fully remote startup**, which comes with both advantages and unique challenges. While the remote setup allowed flexibility, it required disciplined communication and self-motivation.

Some of the main communication challenges were:

- **Coordination Across Schedules:**

Working with team members in different time zones and personal routines sometimes led to delays in feedback or task updates. I learned to manage this by keeping my Azure DevOps board updated and communicating proactively through **Microsoft Teams**.

- **Limited Face-to-Face Interaction:**

It was initially difficult to build rapport and fully understand team dynamics without in-person meetings. Over time, consistent participation in daily stand-ups helped bridge this gap.

- **Clarifying Technical Requirements:**

At times, the written instructions for tasks were so short that I had to follow up with inquiries to make sure they were done correctly. Learning how to communicate clearly and confirm details became a key skill during this experience.

Despite these obstacles, I adapted by maintaining active communication, documenting my progress clearly, and seeking help through chat discussions when needed. This improved my confidence and ability to communicate professionally in a remote environment, and I anticipate that it will be even simpler for me in a physical setting.

5.3 Time Management and Personal Constraints

One of the biggest obstacles I faced was juggling my internship duties with my personal obligations. It was initially challenging to establish a distinct workspace and sustain constant focus throughout working hours when living in a joint family setting.

Furthermore, since many of the tools and frameworks (like ASP.NET Core and Azure DevOps) were new to me, I needed extra time outside work hours to study and practice them. Managing this workload required strong discipline and planning. To overcome this:

- I created a structured **daily schedule** to allocate time for learning and development.
- I set short, achievable goals for each day to maintain steady progress.
- I practiced prioritizing tasks to ensure critical work items were completed before deadlines.

Through consistent effort, I learned valuable lessons in self-management and adaptability — skills that are crucial for any software developer working remotely.

5.4 Limitations of the Internship Scope

Although my internship was highly beneficial, it had certain limitations due to project timelines and defined responsibilities.

Some limitations included:

- **Limited Exposure to Frontend Development:**

My primary focus was backend development, so I could not explore the full frontend stack or mobile app implementation using React Native as much as planned.

- **Project Still in Development Stage:**

Since *Appayon* was still under construction, I couldn't witness the final deployment process or post-launch performance analytics during my internship period.

- **Restricted Access to Production-Level Systems:**

As an intern, I worked mostly within the staging and development environments. Security policies prevented interns from direct access to production servers, which limited my understanding of real-time monitoring and deployment.

Despite these constraints, the internship still provided extensive technical and professional exposure. It equipped me with the confidence and knowledge to take on future projects involving complete software lifecycle management.

Chapter 6

Conclusion and Future Goals

6.1 Summary of Internship Experience

My internship at **Baula Innovations** was an invaluable experience that bridged the gap between academic learning and practical industry exposure. I had the chance to learn how a real-world software project develops from concept and architecture design to implementation and testing while working on the Appayon platform. I actively participated in backend development using ASP.NET Core and MySQL during my internship. I also learnt about Postman API testing and Azure DevOps collaboration workflows. I understood how enterprise-level projects follow systematic approaches like **Agile development**, code reviews, and continuous integration. Beyond technical growth, this internship taught me the importance of teamwork, effective communication, and adaptability in a professional environment. Self-regulation and determination were necessary when working remotely, and regular mentorship from senior developers refined my problem-solving and project management skills. To sum up, my internship experience at Baula Innovations not only improved my technical proficiency but also shaped my professional outlook — reinforcing my interest in pursuing a long-term career in software development.

6.2 Recommendations for Baula Innovations and Appayon

I've placed some helpful recommendations, in my opinion, that might better the company's internship program and the ongoing Appayon project even further.

For Baula Innovations (Internship Process):

- **Structured Onboarding:** New interns can become more acquainted with tools like Azure DevOps, TortoiseGit, and the project structure more rapidly by participating in a brief onboarding session or reading through some provided documentations.
- **Mentorship Sessions:** Organising routine one-on-one or small-group meetings for progress evaluations would give interns more precise feedback and make requirements clear.

- **Technical Workshops:** To improve interns' technical comprehension, brief internal workshops on optimal coding methods, deployment, or debugging in.NET could be arranged.

For the Appayon Project:

- **Enhanced Documentation:** Maintaining updated technical and functional documentation would improve project scalability and onboarding for future developers.
- **Expanded Testing Coverage:** Implementing automated testing frameworks (like xUnit or NUnit) could help ensure consistent quality assurance as the project grows. This would be done in the future, long after the app is launched.
- **User Interface Enhancements:** A more unified and visually appealing UI for both vendors and customers would make *Appayon* stand out in a competitive food delivery market.
- **Performance Optimization:** Monitoring API performance and database queries using profiling tools would help maintain efficiency as the number of users increases.

These recommendations are based on my personal experience and observations as a developer-in-training. Implementing such improvements would further strengthen both the internship program and the success of *Appayon* as a product.

6.3 Future Career Goals

This internship has given me a clear sense of direction for my future career path. The combination of technical exposure, professional collaboration, and problem-solving opportunities inspired me to continue pursuing **software engineering and full-stack development** as my primary field.

In the near future, I aim to:

- **Deepen my expertise in ASP.NET Core** and explore **microservices architecture** for scalable backend systems.
- **Advance my frontend development skills** by mastering **React** and **Blazor** to become more versatile as a developer.
- **Pursue cloud certification** in **Microsoft Azure** or **AWS** to strengthen my understanding of modern deployment and DevOps pipelines.
- **Contribute to open-source projects** and expand my professional portfolio with real-world solutions.
- **Continue lifelong learning**, keeping up with evolving technologies in web and app development to remain adaptable in the fast-changing software industry.

Long-term, my goal is to become a **Software Engineer or Full-Stack Developer** capable of designing and building complete enterprise solutions that make a positive impact in people's lives — much like *Appayon* aims to do for small home-based businesses in Bangladesh.

Bibliography

- [1] B. Innovations, *About baula innovations*, Accessed: October 2025, 2023. [Online]. Available: <https://baulainnovation.com/>.
- [2] Meta, *Reactjs official documentation*, Accessed: October 2025, 2025. [Online]. Available: <https://react.dev/>.
- [3] Microsoft, *Microsoft .net documentation*, Accessed: October 2025, 2025. [Online]. Available: <https://learn.microsoft.com/en-us/dotnet/>.
- [4] Stack Overflow Community, *Stack overflow – developer q&a platform*, Accessed: October 2025, 2025. [Online]. Available: <https://stackoverflow.com/>.
- [5] W3Schools, *W3schools online web tutorials*, Accessed: October 2025, 2025. [Online]. Available: <https://www.w3schools.com/>.

@miscbaulaWebsite, author = Baula Innovations, title = About Baula Innovations, year = 2023, url = <https://baulainnovation.com/>, note = Accessed: October 2025
@miscmicrosoftDocs, author = Microsoft, title = Microsoft .NET Documentation, year = 2025, url = <https://learn.microsoft.com/en-us/dotnet/>, note = Accessed: October 2025

@miscstackoverflow, author = Stack Overflow Community, title = Stack Overflow – Developer Q&A Platform, year = 2025, url = <https://stackoverflow.com/>, note = Accessed: October 2025

@miscreactDocs, author = Meta, title = ReactJS Official Documentation, year = 2025, url = <https://react.dev/>, note = Accessed: October 2025

@miscw3schools, author = W3Schools, title = W3Schools Online Web Tutorials, year = 2025, url = <https://www.w3schools.com/>, note = Accessed: October 2025