

Agronomix - A Project by Backend Developer

by

Shouvik Barua
21241073

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
February 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:

Shouvik Barua
21241073

Approval

The internship report titled “Agronomix - A Project by backend developer” submitted by

1. Shouvik Barua (21241073)

Of Fall, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on February 03, 2025

Examining Committee:

Supervisor:
(Member)

Md. Tawhid Anwar

Senior Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam, PhD

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

Abstract

This internship report presents the experience gained and contributions made during the internship at DataSoft Systems Bangladesh Limited, focusing on the development of Agronomix, an inventory management system for Amin Mohammad Agro Limited. The report outlines the organizational background, project work, and technologies used, including Java Spring Boot, ReactJS, and MySQL. It details the structured approach taken, from requirements gathering and backend development to implementing advanced features such as batch-based inventory updates, authentication, and reporting APIs. The document also highlights the challenges faced, including adapting to system architecture changes and learning Spring Boot. Furthermore, it discusses the future roadmap of Agronomix, with plans to transition to a SaaS platform and a microservices architecture. The report concludes with key takeaways from the internship, emphasizing the importance of planning, adaptability, and collaborative learning in software development.

Keywords: Backend; Java; SpringBoot; React; Inventory management

Dedication

This report is dedicated to all aspiring software engineers, who dream of making an impact through technology. May this journey of learning and innovation continue to inspire many more.

Acknowledgement

First and foremost, I would like to express my heartfelt gratitude to DataSoft Systems Bangladesh Limited for providing me with the opportunity to intern in such a dynamic and professional environment. The experience I gained during my internship has been invaluable in shaping my technical and professional skills.

I extend my sincere appreciation to my intern supervisor, Abu Naim Mohammad Emrajul Islam, for his constant guidance, support, and mentorship throughout my internship. His insights and encouragement have been instrumental in my learning process. I would also like to thank my primary supervisor, Md. Tawhid Anwar, for his valuable feedback and academic support.

A special thanks to my colleagues and team members at DataSoft, whose willingness to share their knowledge and help me navigate challenges made my journey smoother. Their guidance and collaboration have been crucial in my understanding of real-world software development.

Lastly, I am deeply grateful to my family and friends for their unwavering support and encouragement throughout this journey. Without their belief in me, completing this internship and report would not have been possible.

Table of Contents

Declaration	i
Approval	ii
Abstract	iii
Dedication	iv
Acknowledgment	v
Table of Contents	vi
List of Figures	viii
Nomenclature	viii
1 Introduction	1
1.1 About Internship	1
1.2 About this report	1
1.3 Objective	2
1.3.1 Aim	2
1.3.2 Specific Goals	2
1.4 Methodology	2
1.4.1 Primary Data	2
1.4.2 Secondary Data	2
2 Company Profile	3
2.1 Overview	3
2.2 Vision and Mission	3
2.3 What DataSoft Does	4
2.4 Projects	4
2.4.1 The Anti-Money Laundering Solution Suite	4
2.4.2 Remittance Management Solution REMIT365	4
2.4.3 Microfin360	5
3 Training and Working on Project	6
3.1 Overview	6
3.2 Technology and Framework	6
3.3 Getting Familiar with the Backend Development	7

3.4	Getting to know the business	7
3.5	Learning Resource	9
3.6	Work Environment	9
4	Project Work Plan	12
4.1	Phase 1: Requirements Gathering and System Design (Week 1–2) . .	12
4.1.1	Objective	12
4.2	Phase 2: Backend Development (Year 1)	13
4.2.1	Objective	13
4.3	Future Plan for Agronomix	17
4.3.1	Transition to a SaaS Platform	17
4.3.2	Conversion to Microservices Architecture	18
4.3.3	Additional SaaS Features and Enhancements	19
4.4	Challenges Faced	20
5	Conclusion	21
	Bibliography	22

List of Figures

3.4	Inventory	8
3.1	Procurement work flow	10
3.2	Procurement UI	10
3.3	Cow Profile	11

Chapter 1

Introduction

1.1 About Internship

An internship gives recent graduates and students alike the ability to adapt to real-world workplace procedures and put their knowledge to use. Students have the chance to obtain experience relevant to their course of study. In essence, this facilitates the student's transition from the classroom to the real world of employment. In addition, the companies will get to know their new intern better and be able to assign the intern to the best-suited positions based on their needs. Internships enhance a student's work experience, increasing the value of their CV and leading to future career chances.

The popularity of internships is gradually growing, and some students would rather complete an internship than write a thesis paper. At BRAC University, doing an internship or writing a thesis paper are mandatory for students. Even though fewer students may choose to complete an internship at BRAC, those who do so are nevertheless encouraged to learn as much as they can from the workplace and given sufficient assistance from the school. As long as the specialty is related to their degree, students are allowed to select any concentration for their internship. In order to be qualified, students must have completed more than 72 university credits and the internship must last six months in a respectable organization.

I preferred the internship to the thesis paper preparation. I chose DataSoft Systems Bangladesh Limited has been my internship organization. DataSoft has been a CMMI level 5, ISO 9001:2015 certified emerged as the leading software product and services firm in Bangladesh. DataSoft Systems Bangladesh Limited is an organization that provides Software Development services to its clients. Established in 1999, it is a company that focuses on ERP Consulting And SI, Web Development, and Software Development. I am looking forward to coming out of this experience with as much knowledge as possible.

1.2 About this report

Since this internship is required for me to finish my bachelor's degree, I have to turn in a report that will be used as the basis for my assessment by BRAC University. I've written about my workplace's culture and surroundings in this report. I've written about my time at DataSoft and the lessons I've learned as well. It will give

a thorough rundown of the project I am working on and the activities I am involved in as an intern.

1.3 Objective

These are the details that this report hopes to give its readers:

1.3.1 Aim

The primary objective of this report is to illustrate my work experience and the acquired knowledge.

1.3.2 Specific Goals

- Give a comprehensive overview of the company's background
- List all the services DataSoft Systems Bangladesh Limited offers to clients
- Describe the nature of the work that DataSoft Systems Bangladesh Limited is involved in.
- To demonstrate what I have learned during the internship
- To give a thorough explanation of every task I completed on the project that was given to me.

1.4 Methodology

The report that follows primarily focuses on my work experience and the lessons I learned while working at Data Soft Systems Bangladesh Limited. My personal experiences account for the majority of the data and information presented. Also, certain data has been extracted from particular websites in addition to that. The primary sources of data that have been gathered are:

1.4.1 Primary Data

- My practical experience from work
- Discussions and lessons taken from colleagues and team leaders
- Seminars and team meetings

1.4.2 Secondary Data

- DataSoft Systems Bangladesh Limited WebSite.
- Internet

Chapter 2

Company Profile

2.1 Overview

DataSoft Systems Bangladesh Limited is a CMMI Level - 5 & ISO 9001:2000 certified leading software company in Bangladesh [1]. Established in 1998, DataSoft has had great achievements in offering the most competitive and innovative technical services to business and Government segments. Since its establishment in 1998, stepped into the core area of ICT to meet the challenges of enterprise, governance, and economy. Supported by a team of efficient and competent personnel, DataSoft lays great emphasis on constant development and training of human resources to meet the requirements of the domestic and global markets. DataSoft is in the process of adding new technology in its operational system to increase more clients and to serve the existing clients with a little more than their satisfaction [8].

2.2 Vision and Mission

DataSoft is incorporating new technology to further expand the client base and continue to serve clients with a little more than utmost satisfaction [11].

2.3 What DataSoft Does

As a company, DataSoft has contributed significantly to the digitalization of Bangladesh by designing and implementing critical projects like Chittagong Port Automation [1]. DataSoft was the IT partner of one of the most complicated scientific projects in Bangladesh—the Jute Genome Project. DataSoft also touched the lifestyle of the rural population of Bangladesh by offering comprehensive solutions for the world-famous Micro Credit Solution of Bangladesh [12].

Quality Management methods have been a key focus of DataSoft. A customized quality dashboard is used to track all of DataSoft’s operations. A dedicated team of internal quality auditors at DataSoft continuously reviews procedures and pinpoints areas for improvement. The method of DataSoft quality management is end-to-end as it begins with drawing board solutions and continues through after-sale services [4].

2.4 Projects

DataSoft has numerous projects. Though most of them are confidential, here are some of the project information available for the public [9].

2.4.1 The Anti-Money Laundering Solution Suite

The AML Solution Suite helps your organization to comply with Anti-Money Laundering (AML), Bank Secrecy Act (BSA), and FATCA regulations with ease and simplicity. The solution uses multiple detection methods and an advanced analytics engine to help institutions, large and small, achieve compliance while reducing risk [5].

2.4.2 Remittance Management Solution REMIT365

The Remit365 is a blockchain-enabled web-based application for remittance management. The system has been designed with the following features:

- Dashboard Driven
- System User Management
- Transaction Management
- Money Transfer Operator (MTO)
- AML screening to keep you on the right side of anti-money laundering regulations.
and so on...

[10]

2.4.3 Microfin360

Microfin360 is a financial inclusion workflow-driven Microfinance business solution that includes two modules, Management Information System (MIS) and Financial Information System (FIS). This has been developed to meet the requirements of any financial institution, providing online real-time synchronized financial business automation [2].

Microfin360 drastically reduces the organization's overall cost of reporting at branch and consolidation at area, zone, region, or head office level by eliminating the need for resource expenditure, like paper, printing and shipping, human resource, and entertainment costs [6].

There are lots of other projects as well, covering ERP-Smart Enterprise, Port and Logistics, Marketplace, and more [3].

You can get to know about DataSoft's products here [7].

Chapter 3

Training and Working on Project

3.1 Overview

Being able to work as an intern for a corporation of this size is a major step forward for both one's academic and professional lives. In today's very competitive employment market, having work experience is highly advantageous. When students are eventually done with their studies, it also makes the transition to the workplace easier for them. It's difficult enough to use theoretical knowledge in the actual world. It's crucial to recognize that industry procedures differ as well.

I started working as an intern at DataSoft Systems Bangladesh Limited, and I was assigned to the Japan Team as a software developer. This internship is not only a great opportunity for me to get experience and learn new things while working here, but it is also a prerequisite for BRAC University graduates. I knew very little about how workplaces operated when I first started.

This was a formal training period because interns at DataSoft Systems Bangladesh Limited do not have immediate access to projects. Nevertheless, after three months of working as an intern, I had the good fortune to receive an offer to join as a regular employee. I subsequently had direct access to a project named "Agronomix." This is an web application, specifically build for Amin Mohammad Agro Limited, which is located right beside the BRAC Residential Campus. I worked on the backend of that software. I will not be able to share some portions of the code base I work on due to confidentiality, however I have included multiple screenshots throughout the report.

3.2 Technology and Framework

The following softwares, tools and frameworks are used in the development process of Agronomix:

- Backend : Java Spring Boot
- DataBase : MySQL
- FrontEnd: ReactJs

3.3 Getting Familiar with the Backend Development

I did have a pretty good understand of Java but I didn't have any knowledge about Spring Boot or any other framework. So, in the first two months of my internship period, I spent most of my time studying and learning SpringBoot. At the same time to refresh my Java knowledge I started going through the official Java doc. I also got familiar with some design pattern like "N-tier Architecture", "Inversion of Control", loose coupling and tight coupling etc. I also got familiar with API. How it works, how data is being passed between frontend and backend, how request and responses are happening and data being passed in between different layers. At the same time, I also learnt how to test api with PostMan and Swagger.

3.4 Getting to know the business

Before starting to work in the project "Agronomix", I had to know the business of the software. How the software should work, what are be the functionalities the software should have. So, here is a detailed description about the application. As, I mentioned before it is an inventory management system. The main purpose of Amin Mohammad Agro Limited to have this application is to prevent theft and have a digital receipt of every transaction they are making.

So, first everything they purchase will go through procurement. They can buy two types of products one is Livestock which includes Cow-Buffalo, Goat-Sheep, Chicken, fish and supplies which includes everything starting from food to medicine whatever goods are needed for their livestock. Below is a visual representation of how the procurement works.

As, you can see in the diagram this is a set of steps. First, if anything needs to be purchased, a purchase requisition(PR) will be generated. Then the person who is in charge of purchasing different goods he will be able to approve the request. After that Purchase Order(PO) request will be generated. In purchase order section the client will be able to select from which company they want to order from. After placing the order the admin will be able to approve the order and confirm the order. Finally, when the vehicle reaches the farm, a gate pass will be generated. In the gate pass the user can store the information of the driver for safety purpose. Then finally, quality check will happen in two phase where they thoroughly check the quality of all the goods. One thing to mention, this is a hierarchical process. Though you can see all the options available in the attached Procurement UI picture, all the options will not be visible to all the users. The options will be visible according to the assigned role.

All the purchased livestock have their own profile. Here is a graphical representation of it

As you can see in the figure, Each profile contains detailed information regarding that particular cow.

All the functionalities are same for buffalo as well. The only difference in business between cow-buffalo and goat sheep is, they put the cow-buffalo in line but there is no line for goat-sheep. Also, one thing worth mentioning that, the data processing is not real time here. There is a night batch which is responsible for updating all the information.

Most of the functionalities are same for chicken as well. but the difference is cow-buffalo and goat-sheep are all individual entity. But there is no entity for individual chicken. They have batch for chicken and each batch consists of 100-200 chicken.

And as of writing this report the fish and vermicompost module are still under development.

Next, there is inventory module, where the user can see all the produced and purchased goods. Fig: 3.4 is the ui of the inventor module.

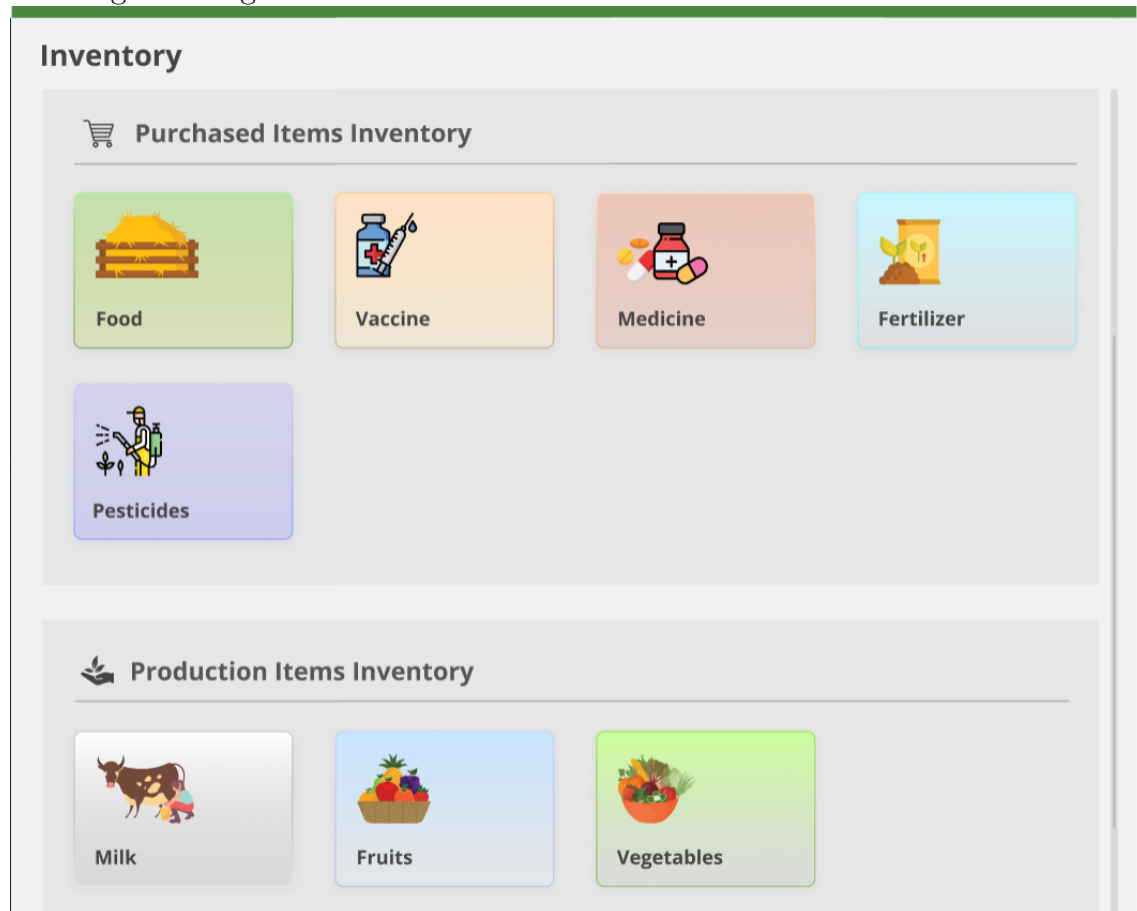


Figure 3.4: Inventory

Though I did not work on the authentication part, it's worth mentioning that to ensure security, Spring security was used.

in order to create connection between front and backend, I have created api for each of this functionalities and passed it to the frontend. And to test those api, Swagger was used.

As the api are not public api, unfortunately, I am unable to provide any picture of the Swagger.

3.5 Learning Resource

My main resource for learning spring boot was documentations and youtube. Also, I got lots of help from my supervisor and other colleagues as well who put lots of time and effort to show me how things work.

3.6 Work Environment

I've learned a lot from my work with DataSoft Systems Bangladesh Limited, which has proven to be an intriguing experience. I made a few new friends, some of whom I have quickly gotten to know well. No one suppresses any ideas, and everyone is really helpful. When I started, I was handled more like an actual full-time employee than an intern. I was able to quickly adapt into their work culture because of this.

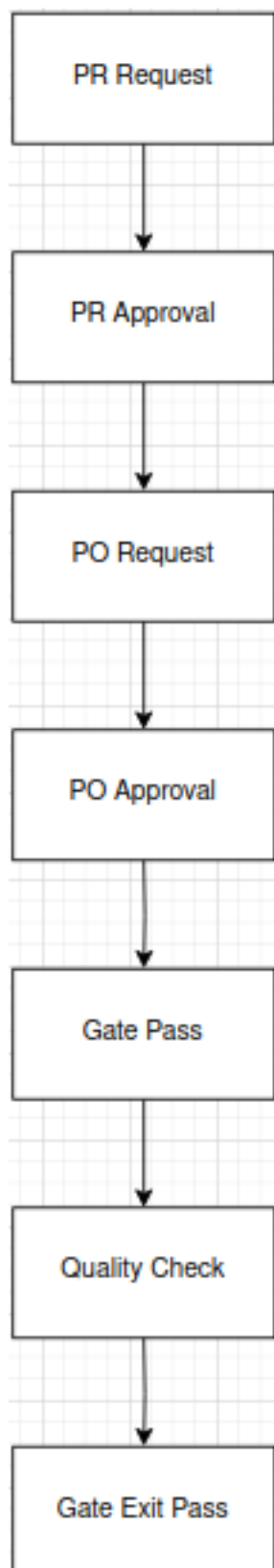


Figure 3.1: Procurement work flow



Figure 3.2: Procurement UI

AMIN MOHAMMAD agro

Dashboard

Live Stock Profile

Cow & Buffalo

Goat & Sheep

Chicken

Pond Profile

Plant Profile

Vermicompost

Food Expense

Vaccine Expense

Medicine Expense

Fertilizer Expense

Pesticides Expense

Pana Expense

Sales

Inventory

Procurement

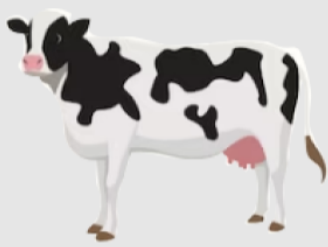
Vendor Profile

Employee Profile

Logout

Individual Cow/Bufalo Profile

EDIT



Tag ID: C1

System ID: MQT0701

Active

Shed 01 Line 01

PR No. #PRN001

Cow

Purchase Date

Aug 24th, 2020

Purchase Price

32000

Birth Date

01-01-2021

Color

Black & White

Gender

Female

Age

28 Months

Breed

Jersey

Weight

480 kg

Breeding Status

Pregnant

Probable Delivery Date

21-09-2023 (28 Days Left)

Next Vaccination

11-09-2023 (10 Days Left)

Previous Vaccination

Vaccine 1, Vaccine 2

Previous Disease

Botulism, BVD

Current Health Status

Sick

Tag Color

Yellow

Type

Type 01

Calf Count

2

Mother ID

C20

Total Expense

120000 BDT

Total Milk Production

501 LTR

Description

Description about the cow/buffalo...

Figure 3.3: Cow Profile

Chapter 4

Project Work Plan

4.1 Phase 1: Requirements Gathering and System Design (Week 1–2)

4.1.1 Objective

Understand the business requirements, define system functionalities, and design the overall architecture.

Task 1.1: Conduct stakeholder meetings to gather requirements. Duration: 3 days
Description: Collaborate with farm managers, supply chain operators, and technical teams to define key features like batch tracking, inventory updates, user management, and reporting.

Task 1.2: Define system functionalities and use cases. Duration: 2 days
Description: Create a detailed list of use cases including adding/removing inventory, batch expiration tracking, and order management.

Task 1.3: Design system architecture and technology stack. Duration: 3 days
Description: Design the system architecture including backend microservices, database schema, and React component hierarchy. Finalize the tech stack (Java Spring Boot, React, MySQL).

Task 1.4: Prepare database schema and ER diagrams. Duration: 2 days
Description: Design the database schema to manage product categories, batch information, inventory logs, and user permissions.

4.2 Phase 2: Backend Development (Year 1)

4.2.1 Objective

Milestone 1: Initial Setup and Core API Development (Month 1–3)

Task 2.1: Set up Spring Boot environment and create base project structure. Duration: 1 week Description: Set up the development environment, project configuration (Maven/Gradle), and version control (Git). Establish the foundation for microservices architecture and project modules.

Task 2.2: Core Inventory Management API Development. Duration: 6 weeks Description: Implement basic CRUD (Create, Read, Update, Delete) functionalities for inventory items, batches, and products. Ensure batch-based updates for products as per inventory needs.

Task 2.3: Develop User Management and Authentication Module. Duration: 4 weeks Description: Implement Spring Security for authentication (using JWT) and authorization (role-based access control). Create APIs for user registration, login, and role assignment.

Task 2.4: Unit Testing for Core Features. Duration: 1 week Description: Write unit tests for core inventory and user management services using JUnit and Mockito.

Milestone 2: Advanced Inventory Features and Optimization (Month 4–6)

Task 2.5: Implement Batch Expiration Tracking. Duration: 4 weeks Description: Create a service that handles expiration tracking for inventory batches, sending notifications or alerts when a batch is near its expiration date.

Task 2.6: Optimize Inventory Update Logic. Duration: 3 weeks Description: Refine the batch-level updates to efficiently manage updates based on the latest available inventory, ensuring correct allocation across multiple batches.

Task 2.7: Implement Bulk Upload and Update Functionality. Duration: 3 weeks Description: Develop an API to handle bulk upload of products and inventory data (e.g., from CSV files), allowing users to update stock levels in bulk.

Task 2.8: Performance Testing and Optimization. Duration: 2 weeks Description: Perform load testing and profiling of the system to identify bottlenecks in the API response times. Optimize database queries and service layers.

Milestone 3: Reporting, Notifications, and External Integrations (Month 7–9)

Task 2.9: Develop Reporting API. Duration: 5 weeks Description: Implement APIs for generating inventory reports, including stock levels, batch status, and expiration data. Enable PDF/CSV export functionality.

Task 2.10: Implement Notification and Alert System. Duration: 4 weeks Description: Set up a notification service to send email or SMS alerts when certain thresholds are met, such as low stock or batch expiration.

Task 2.11: External System Integration (ERP/Accounting). Duration: 3 weeks Description: Integrate the system with external accounting software or ERP systems using APIs to synchronize inventory and financial data.

Task 2.12: Testing and Bug Fixing. Duration: 2 weeks Description: Conduct integration tests for external systems and verify the reliability of the reporting and notification systems.

Milestone 4: Scalability, Security, and Database Optimization (Month 10–12)

Task 2.13: Scale the Microservices Architecture. Duration: 4 weeks Description: Split the application into multiple microservices, isolating modules such as inventory management, user management, and reporting to ensure horizontal scalability.

Task 2.14: Advanced Security Features. Duration: 3 weeks Description: Implement additional security layers such as two-factor authentication (2FA) for sensitive operations, data encryption in the database (for sensitive fields), and rate-limiting to protect against denial-of-service attacks.

Task 2.15: Database Optimization and Indexing. Duration: 3 weeks Description: Analyze and optimize database queries, add indexes to frequently accessed tables (e.g., inventory, batches), and implement database partitioning if necessary for performance improvements.

Task 2.16: Final Testing, Bug Fixing, and Documentation. Duration: 2 weeks Description: Perform a full round of testing (unit, integration, performance, and security testing), fix remaining bugs, and finalize technical documentation for the backend system.

4.3 Future Plan for Agronomix

The future of Agronomix involves scaling the system to meet the needs of a broader audience, transforming it into a multi-tenant **Software as a Service (SaaS)** platform. Additionally, the system will be restructured using **microservices architecture** to ensure scalability, flexibility, and efficient resource management as the platform grows. The following plan outlines the major steps and objectives for this transformation.

4.3.1 Transition to a SaaS Platform

Objective: To transform Agronomix into a multi-tenant SaaS platform that can serve multiple customers (agricultural businesses, farms, cooperatives) from a single infrastructure while ensuring data segregation, scalability, and high availability.

Key Steps

1.1. Multi-Tenancy Implementation:

Description: Implement multi-tenancy to allow multiple organizations to use the same system while keeping their data isolated. This could be done using:

- **Database per tenant:** Each client has a dedicated database to isolate their data.
- **Shared database with tenant ID:** Tenant-specific keys are used to segregate data within the same database.

Objective: Ensure each tenant can manage their own inventory data, users, and preferences without interference from other clients.

1.2. Subscription and Billing Management: **Timeline:** Month 3–4

Description: Integrate a subscription management system for handling customer sign-ups, billing cycles, and plan upgrades/downgrades. Offer tiered subscription plans (e.g., basic, premium, enterprise) with varying levels of feature access, storage, and support.

Objective: Provide seamless billing and invoicing for different pricing plans and manage the customer lifecycle (trials, renewals, cancellations).

1.3. Customer Management Dashboard:

Description: Develop an admin dashboard for managing customers, including onboarding, tenant management, and monitoring subscription statuses. Ensure the platform can handle multiple clients with diverse inventory management needs.

Objective: Provide internal and external tools to manage tenants, monitor system usage, and ensure a streamlined customer experience.

1.4. Cloud Hosting and Scalability:

Description: Migrate the platform to a cloud service provider (e.g., AWS, Google Cloud, Microsoft Azure) to ensure scalability, auto-scaling of resources, and high availability. Leverage cloud tools like AWS.

Objective: Allow the system to automatically scale based on user load, ensuring uptime and performance as more tenants subscribe to the service.

4.3.2 Conversion to Microservices Architecture

Objective: Rearchitect Agronomix to a microservices-based system, enabling independent deployment, scalability, and fault isolation of key components like inventory management, user authentication, and reporting.

Key Steps

2.1. Service Decomposition:

Description: Identify and separate the main features of the application into distinct microservices. Initial services to extract include:

- Inventory Management Service
- Batch Tracking Service
- User Authentication and Authorization Service
- Reporting Service
- Notification Service

Objective: Ensure each service operates independently, can be developed and deployed separately, and communicates with other services through APIs.

2.2. API Gateway Implementation:

Description: Set up an API Gateway (e.g., Kong, AWS API Gateway) to manage requests between microservices and external clients. The API Gateway will handle routing, rate-limiting, and security (authentication and authorization).

Objective: Ensure a unified entry point for clients while managing traffic and controlling access to various microservices.

2.3. Service-to-Service Communication:

Description: Implement communication between microservices using lightweight protocols like gRPC or REST, and manage asynchronous messaging using a message broker like RabbitMQ or Kafka for handling events (e.g., when inventory levels fall below a threshold).

Objective: Enable reliable and efficient communication between services without tight coupling, ensuring smooth interaction between independent services.

2.4. Database Decentralization:

Description: Move from a single database for the entire application to polyglot persistence where each microservice manages its own database as needed (e.g., inventory service might use MySQL, while the notification service could use a NoSQL solution like MongoDB).

Objective: Decouple database dependencies, ensuring that changes or issues in one microservice's database do not affect others.

2.5. Containerization and Orchestration:

Description: Containerize the microservices using Docker and manage them with orchestration tools like Kubernetes or Docker Swarm. This will enable automated

scaling, fault-tolerance, and resource management across the platform.

Objective: Deploy microservices in isolated containers, allowing them to be scaled independently based on demand.

2.6. Monitoring and Logging System:

Description: Implement a centralized logging and monitoring system using tools like Prometheus, Grafana, and ELK Stack (Elasticsearch, Logstash, and Kibana). This will allow real-time monitoring of service health, performance, and alerts for any system failures or performance bottlenecks.

Objective: Ensure operational visibility and quick troubleshooting through effective logging and real-time metrics.

4.3.3 Additional SaaS Features and Enhancements

Objective: To provide a comprehensive SaaS experience by adding features that enhance usability, security, and customer satisfaction.

3.1. Advanced Analytics and Machine Learning:

Description: Integrate analytics features that offer predictive insights into inventory levels, usage trends, and supply chain optimization. Incorporate machine learning models to help tenants forecast inventory needs based on historical data and trends.

Objective: Provide value-added services that give tenants actionable insights and help them make data-driven decisions.

3.3. API for Third-Party Integrations:

Description: Develop a public API that allows third-party developers to integrate their systems (e.g., farm management systems, accounting software) with Agronomix.

Objective: Increase the versatility of the platform by allowing external developers and businesses to extend its functionality and integrate it with their own tools.

4.4 Challenges Faced

One of the major challenges I encountered was adapting to significant change requests within tight deadlines. Initially, the inventory system was designed as a centralized system. However, a major change required transforming it into a decentralized system. This adjustment involved revising every module in the application, including inventory management, batch tracking, reporting, and user access controls. Balancing these changes while maintaining system integrity and performance was a demanding but rewarding experience.

Additionally, learning Java Spring was another significant hurdle. As Spring is a vast and complex ecosystem, it required a steep learning curve to understand its core concepts, frameworks, and integration techniques. Despite these challenges, I managed to adapt and use Spring effectively for the development of Agronomix.

Chapter 5

Conclusion

I have picked up a lot of valuable software development fundamentals over the span of my job. Before beginning the real development phase, it is crucial to plan things out, and it is sage to constantly document everything you do. It is crucial to exercise patience and not hurry things since even the most straightforward and evident edge cases can be easily overlooked. To make it simpler for me to evaluate a problem from several angles, I've learned that it's critical to consider a single concept from multiple angles and to fully understand the business of the software.

I have been exposed to both very simple and complex scenarios, all of which were important to comprehend before moving on to the more complicated aspects of coding. I've been exposed to more complicated problems over the last several months, and I've also spent time understanding and coming up with new development ideas with other developers during their coding and debugging sessions.

I have started this internship on the 15th of January 2024. And from 15th January to 31st March, I worked as an intern. I have joined as a full time employee on the 1st of April. I have covered 1 year of my work as a software developer in this report as it is a part of my Final thesis.

Bibliography

- [1] DataSoft Systems Bangladesh Limited. “About us.” Accessed: 2025-02-11. (2025), [Online]. Available: <https://www.datasoft-bd.com/about-us>.
- [2] DataSoft Systems Bangladesh Limited. “Analytics.” Accessed: 2025-02-11. (2025), [Online]. Available: <https://www.datasoft-bd.com/our-solutions#sec6>.
- [3] DataSoft Systems Bangladesh Limited. “Artificial-intelligence.” Accessed: 2025-02-11. (2025), [Online]. Available: <https://www.datasoft-bd.com/our-solutions#sec2>.
- [4] DataSoft Systems Bangladesh Limited. “Digital-distribution.” Accessed: 2025-02-11. (2025), [Online]. Available: <https://www.datasoft-bd.com/our-solutions#sec2>.
- [5] DataSoft Systems Bangladesh Limited. “Smart-business.” Accessed: 2025-02-11. (2025), [Online]. Available: <https://www.datasoft-bd.com/our-solutions#sec4>.
- [6] DataSoft Systems Bangladesh Limited. “Artificial-intelligence.” Accessed: 2025-02-11. (2025), [Online]. Available: <https://www.datasoft-bd.com/our-solutions#sec7>.
- [7] DataSoft Systems Bangladesh Limited. “Artificial-intelligence.” Accessed: 2025-02-11. (2025), [Online]. Available: <https://www.datasoft-bd.com>.
- [8] DataSoft Systems Bangladesh Limited. “Clients.” Accessed: 2025-02-11. (2025), [Online]. Available: <https://www.datasoft-bd.com/about-us>.
- [9] DataSoft Systems Bangladesh Limited. “Fin-tech.” Accessed: 2025-02-11. (2025), [Online]. Available: <https://www.datasoft-bd.com/our-solutions#sec3>.
- [10] DataSoft Systems Bangladesh Limited. “Mobile-app.” Accessed: 2025-02-11. (2025), [Online]. Available: <https://www.datasoft-bd.com/our-solutions#sec5>.
- [11] DataSoft Systems Bangladesh Limited. “Our-solutions.” Accessed: 2025-02-11. (2025), [Online]. Available: <https://www.datasoft-bd.com/our-solutions>.
- [12] DataSoft Systems Bangladesh Limited. “Port-automation.” Accessed: 2025-02-11. (2025), [Online]. Available: <https://www.datasoft-bd.com/our-solutions#sec1>.