



COMP 399

Summer Practice Report

Ömer Bilbil
042101089

04.08.2025 – 29.08.2025, 20 workdays
Submitted: 29.09.2025

MEF University

Computer Engineering Program

Executive Summary

This is my internship report describing the work I did under the AI Innovators program in my internship at Microsoft. Microsoft is a global technology firm that builds software, services, devices, and cloud solutions. Its main engineering work involves building the Windows operating system, Microsoft 365 applications, and the Azure cloud solution. Microsoft invests heavily in research, especially in artificial intelligence (AI). The AI Innovators project focuses on using innovative AI technologies to solve actual problems. The project that I contributed to was focused towards this goal by using AI for improving the stock price prediction systems and helping the investors make the right choice.

The main aim of my internship was to create and deploy a Stock Price Prediction System using cutting-edge machine learning techniques. It was an end-to-end project of creating a tool using several methods to analyze the historical financial data and forecast future stock prices. I first learned about the working mechanism of stock markets and the financial time series data structure. Then I utilized this knowledge in developing a multi-step prediction pipeline.

First, I built a data preprocessing module that collects and cleans large amounts of historical stock prices and significant financial metrics. Much of my effort went into designing a deep learning model using PyTorch to be able to learn complex patterns and trends. I trained several models on a big real world dataset with millions of data points. On the final stage, I had combined all parts of the project into a master engine that processes the preprocessing quickly on the first step and then sends the cleaned data to the trained PyTorch model to produce accurate predictions. In order to make the system easily usable, I created two interfaces: a command-line tool for advanced users and power users for automation, and a web interface showing the prediction results in an understandable format and keeping a history of past forecasts.

As I started the internship, I tried to apply whatever I learned from my minor at university in terms of machine learning and software engineering to a real project and gain experience in financial AI solutions. I was also interested to find out how a full-fledged AI project is designed and run in a big tech company. The final product turned out much more than I expected. I built a working and annotated application that utilizes smart design and multiple levels of analysis, much like actual financial forecasting programs.

I learned numerous technical skills through this project, such as cleaning and preprocessing time series data, training deep learning models using PyTorch, backend programming, and building a frontend interface with Python. I also learned about handling a big project, the significance of possessing good data pipelines, and how to debug and correct issues in software. I got to lead the project from the start to the completion, and it helped me build my confidence and provided me with a keen interest in working in AI-driven financial forecasting in the future.

Contents

Executive Summary	2
Contents.....	3
1. Company and Sector	4
a. Overview of the Company and Sector.....	4
b. Organization of the Company	5
c. Production/Service System.....	6
d. Professional and Ethical Responsibilities of Engineers	8
2. Summer Practice Description	10
a. Impact.....	16
b. Team Work.....	17
c. Life Long Learning.....	17
3. Conclusions	19
References	20
Appendix 1: Daily Activity Tables	21

1. Company and Sector

a. Overview of the Company and Sector

Company: Microsoft Corporation

Address: Microsoft Türkiye, Bellevue Residences, Levent Mahallesi, Aydın Sokak No:7, 34340,

Beşiktaş, İstanbul, Türkiye

Websites: <https://www.microsoft.com> (Global) | <https://www.microsoft.com/tr-tr/> (Turkey)

Microsoft Corporation has a long history, since 1975, when Bill Gates and Paul Allen founded the company to develop software to enable early personal computers to work. The company has established itself as a software powerhouse with its launch of the MS-DOS operating system and the later Microsoft Windows operating system as its platform for PCs. Since its inception, Microsoft has grown significantly and expanded well beyond the world of PC software. Today, Microsoft is a publicly traded company on the NASDAQ (MSFT), and it is a global leader in the high tech space. The vast majority of its activities are run from clean, indoor office buildings. In Turkey, the country's Microsoft headquarters is located in Istanbul, where they employ support staff, and primarily professional staff with expertise in engineering, sales, marketing, and support issues. There are over 220,000 employees globally in Microsoft (Microsoft, 2024).

Microsoft works in a high technology sector space called Information and Communication Technology (ICT) which is fundamentally important to the global economy for growth and innovation (IDC, 2023). In this sector, the company's main purpose is to empower every person and every organization on the planet to achieve more. Microsoft has a large market share in key areas such as desktop operating systems (Windows) and cloud computing (Azure), and competes with a number of global technology companies like Google, Amazon, and Apple (Gartner, 2023). Microsoft has customer segments that include individual consumers, large international corporations, and government entities both in Turkey and worldwide. Most of the work completed at Microsoft Turkey relates to digital transformation across Turkey. The Microsoft Turkey local office works with Turkish companies to help them modernize by implementing cloud technologies and advanced software solutions (Symantec, 2023). Microsoft creates local product versions and engages specially in supporting the Turkish market and actively supports the local startup ecosystem through various technology and mentorship programs.



Figure 1. Microsoft Logo

b. Organization of the Company

Microsoft Turkey is a subsidiary of Microsoft Corporation and possesses a matrix is an organizational structure with both functional as well as project-based methods. This hybrid structure allows the firm to achieve operational effectiveness in day-to-day activities while remaining flexible enough to keep up with quick-changing technology markets and customer needs.

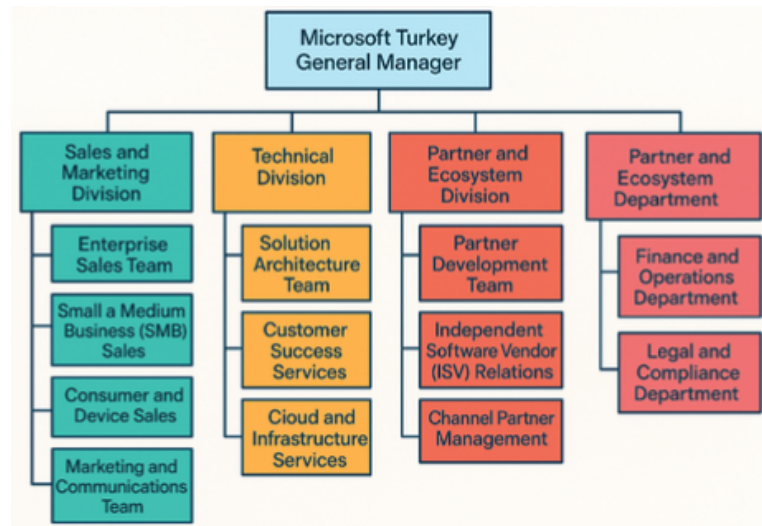


Figure 2. Microsoft Organizational Structure

Major Departments and Their Responsibilities

Sales and Marketing Division

This department is the main source of income. The Enterprise Sales Team targets large corporations and government institutions with the Azure cloud services, Office 365, and Windows enterprise products. The SMB Sales Team focuses on small and medium businesses with solutions tailored to them. The Consumer and Device Sales team oversees retail relationships in the Surface devices, Xbox consoles and consumer software. The Marketing and Communications Team creates locally targeted campaigns and aligns with the world Microsoft strategies.

Technical Division

The Technical Division is the source of customer support and expertise. The Solution Architecture Team creates complex technology solutions to enterprise customers. The Customer Success Team sustains the relationships with the clients and offers optimization advice. Technical Support Services provides multi level customer support, and Cloud and Infrastructure Services team focuses on the implementation of Azure and cloud management.

Partner and Ecosystem Division

This department deals with the network of business partners of Microsoft Turkey. The Partner Development Team finds and develops new partnerships, training and equipping them. ISV Relations collaborates with software companies to make sure that there is

integration of platforms. Channel Partner Management is in charge of managing current partner performance and development.

Support Departments

The Human Resources deals with the recruitment, development of employees, and culture of the workplace. Finance and Operations is in charge of budgeting, financial reporting, and adherence to the Turkish regulations and Microsoft policies. Legal and Compliance makes sure that the business operations are in line with the local laws and regulations on data protection.

Hierarchical Relationships and Structure Type

Microsoft Turkey operates with an organic structure characterized by:

- **Flexible Hierarchy:** There is a proper reporting relationship but all levels are expected to work cross functionally. The staff members also have frequent interactions with other workers in other departments to resolve customer issues.
- **Decentralized Decision Making:** The department heads enjoy a lot of autonomy in their areas of expertise and there is quicker response to the needs of customers and changes in the market.
- **Project Based Collaboration:** Employees are also allowed to work on cross departmental projects with the matrix structure allowing them to work on the main concern. One example is that solution architects collaborate with sales, marketing and partner teams on large implementations.
- **Adaptive Communication:** There is both vertical and horizontal flow of information through reporting channels and across departments through routine meetings and digital tools as well as informal channels.

c. Production/Service System

Microsoft is a technology firm that mainly manufactures software and offers digital services. Microsoft does not have a traditional factory-based production system that produces physical products, but a complicated system of software development, service management, and customer delivery. It is a system that develops, sustains and facilitates digital products such as Microsoft Windows, Office 365 and its cloud platform, Azure.

The primary objective of such a system is to provide value to customers with the help of dependable and innovative technological solutions. Microsoft has various departments which collaborate in an innovation, sales, and support cycle.

Below is a chart that describes the relationships between the major components of Microsoft's service system.

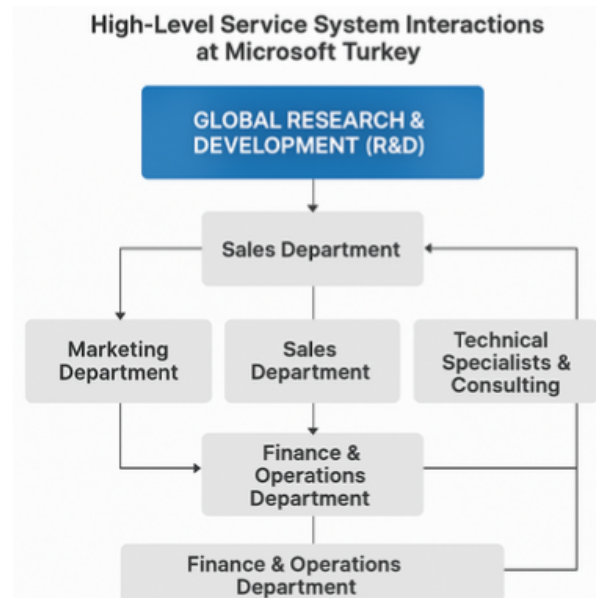


Figure 3. Major Company Components and Relations

All these departments have a presence at both the global level and the local level, like at Microsoft Turkey. Microsoft Turkey's role is to execute the global strategy in the local market, handling local sales, marketing, and support.

One of the most evident examples of this service system at work is the procedure that a customer completes to deploy a Virtual Machine (VM) on Microsoft Azure. The customer starts by logging into the Azure portal, picking the service, and setting up the VM, by selecting specifications such as its size, operating system, and data center region. This system will then automatically validate these settings and then start the digital production process, the essence of which is provisioning. It takes only a few minutes to automatically allocate hardware, install the software and deploy the running VM, which is then available to the customer by Azure.

This involves non-productive and productive aspects. Those steps that directly create value are productive steps, e.g. the first configuration and the last running service. Conversely, non-productive elements are required yet they are time the customer has to wait, such as in the validation and deployment process. The performance objective of Microsoft is to reduce this non-productive time by the continuous automation and efficiency increments.

Performance of the system is quantified by such metrics as deployment velocity and service availability, which is formally assured by Service Level Agreements (SLAs) that guarantee uptimes frequently over 99.9%. This whole service provision system is underpinned with advanced planning and forecasting. Microsoft does not predict the sale of individual services but the aggregate demand in various regions in computing power. This information, which is grounded on the past use and market trends, is very essential in planning on where and when to construct new data centers to address future customer demands.

Moreover, the company has a special kind of inventory — the unused capacity of servers and storage of its data centers. This is a key balancing act in managing this digital inventory between not having enough capacity to meet unexpected customer demand and the cost of

having idle hardware. All this is possible through a huge physical logistics effort that controls the worldwide supply chain of server hardware, including the procurement and shipment of hardware, its installation and its eventual decommissioning. This is a complicated backend structure that is the real physical infrastructure that enables Microsoft to provide the global cloud services it does.

d. Professional and Ethical Responsibilities of Engineers

Professional and ethical responsibility of engineers is very significant in a global technology company such as Microsoft that produces products that touch the lives of billions of people. The digital world is the creation of engineers, and their activity presupposes not only technical competence but also the devotion to the safety, privacy, and integrity. Engineers are expected to maintain such high standards in all their professional undertakings because the reputation of the company and the trust of its customers is on the line.

Functions of Software Engineers at Microsoft

During my internship, I have seen that Software Engineers at Microsoft carry out very diverse roles that are critical in the product development process. Their major task is to develop and support the software and services the firm provides. They all have the following functions in common:

- **Designing and Developing Software:** Creation of clean, efficient and maintainable code of new products and services like Microsoft Azure or Office 365.
- **Testing and Debugging:** The systematic testing of their code to find and eliminate bugs and this is what makes the software reliable and works as per the intended purpose.
- **Code Reviewing:** Reading and reviewing the code of other members of the team to assure quality, any potential bugs, and security.
- **Collaboration:** Working with other engineers, program managers and designers in the planning, development and introduction of products.
- **Maintenance and Updates:** This is the process of improving already installed software to add new functionality, improve performance and customer feedback.
- **Documentation:** Technical writing of how the software works, which will be used in future to maintain the software and other programmers.

Professional and Ethical Responsibilities

Other than the technical roles, there are other professional and ethical roles of Software Engineers at Microsoft. Professionally, they are supposed to be competent by keeping up with the changing technology, work diligently and be accountable to their code. Their obligations are deeper, even ethically. One of the main ethical responsibilities is to keep users' privacy which means that the enormous data processed by Microsoft systems should

be safe. This is directly associated with the accountability of developing secure products that protect against cyberattacks. Moreover, engineers have to be honest and act with integrity, and they should be transparent when it comes to software limitations.

Company Standards and Code of Conduct

Microsoft also has a formal document known as the Microsoft Standards of Business Conduct to ensure that the employees are guided in the roles. It is an evident ethical document that would be applicable to the employees all over the world, including those in Turkey. It is based on the values of trust and integrity and deals with such significant principles as the compliance with the law, the secrecy of information, the establishment of the respectful and non-discriminatory atmosphere at the workplace, and the lack of conflict of interest. The employees will be subjected to yearly training on the standards to ensure that the employees understand these standards and comply with them.

Observed Ethical Practices in Action

I have seen these standards being applied in day to day engineering during my internship. As an example, the code review process is an effective way of conducting a practical check both on the technical quality and possible security or privacy issues. I also noted that there was much emphasis on privacy during project planning meetings as teams discussed how to collect less data and design it in such a way that user information was safeguarded. Lastly, efforts to reduce the possibility of bias in AI systems were also made, as teams will spend time testing on different data to make sure that their technology will be fair to all. These examples indicate that in Microsoft, ethical responsibilities are so ingrained into the engineering culture and is an essential component of the product development process.

2. Summer Practice Description

As part of my summer internship at Microsoft Turkey, I was mostly engaged in the development and implementation of an advanced stock price prediction system that uses deep learning algorithms together with classical time series analysis. The essence of my work was to develop a smart forecasting system based on large-scale historical financial datasets consisting of millions of daily stock market records. My major engineering activities were the development of various machine learning and deep learning models (including LSTM and GRU networks implemented in PyTorch, as well as baseline models such as ARIMA) and the creation of an advanced preprocessing pipeline to clean data and optimize features.

The most difficult part was to combine traditional statistical forecasting methods with modern deep learning predictions to make a hybrid prediction system. Traditional approaches such as ARIMA and moving average models had to be integrated with neural network-based methods, and this required careful architectural design and extensive testing to make the system compatible and effective.

This internship project was directly related to several fields of my computer science studies, especially in the fields of artificial intelligence, software engineering, and financial data science. The deep learning modules involved the application of concepts on statistical analysis and algorithm optimization that I learned in my coursework, and the software architecture was based on what I learned in my software engineering courses. The financial forecasting dimension exposed me to the practical uses of data science concepts that were initially theoretical and gave me a practical understanding of time series modeling and data processing techniques that I had learned in my coursework.

There were many technical problems encountered in the project that enhanced my engineering skills tremendously. The need to use large-scale time series datasets necessitated the acquisition of more sophisticated data processing methods and memory optimization principles that are not typically covered in academic institutions. Financial time series analysis required knowledge of market behavior and feature engineering techniques because of its inherent complexity.

To get the best results in terms of prediction accuracy and processing speed, the deep learning models had to be optimized thoroughly through trial and error with various architectures and hyperparameter tuning in PyTorch. The combination of traditional statistical forecasting and neural network-based approaches helped me to gain important lessons regarding system architecture and the interaction of its components. The complexity of mixing classical models with probabilistic deep learning predictions had to be addressed with extreme care when considering edge cases, error handling, and the design of the user interface.

I also gained experience with professional development tools such as version control systems, automated testing frameworks, and deployment pipelines that are common in the financial technology sector.

This experience provided me with a detailed insight into the industry-level practices of software development and the concepts of AI-driven financial forecasting. The project

enhanced my problem-solving and technical communication skills tremendously since I gave presentations to senior engineers and stakeholders frequently.

Above all, the internship gave me an understanding of the way in which academic expertise can be applied to practical engineering solutions and taught me that one should never stop learning in the fast-changing world of artificial intelligence and financial technology.

- **Activity Analysis**

One of the core computer engineering activities that I had a hand in was designing and developing a multi-stage stock price prediction pipeline. This was the most significant part of the system intelligence and the one that is ultimately making the final prediction for any given stock. This task was an independent engineering activity consisting of system design, algorithm incorporation, and data flow.

The users first select a stock symbol and a time period, and then the system loads the corresponding historical price data automatically from a financial data source. Raw data is then subjected to data cleaning and normalization. Missing values are handled, outliers are smoothed, and time series are normalized to display consistent performance under different market conditions. This is the first step towards performance and reliability, as it ensures that the following steps do not have to deal with noisy or varying input.

Upon successful completion of initial quality checks, the dataset moves into a parallel modeling phase, where it is handled by both conventional time series models and deep learning models in parallel. This parallel approach increases resilience and serves as two independent sources of predictive evidence.

- The statistical engine utilizes standard forecasting models such as ARIMA and moving average models to detect linear trends and seasonality.
- The deep learning engine, implemented with LSTM and GRU networks in PyTorch, uncovers complex non-linear patterns in the time series.

Lastly, the system combines the outputs of these two engines to generate a final prediction through a confidence scoring and ensemble mechanism, providing an informed estimate of the next time steps of the stock price.

Performance Indicators and Metrics

To measure the performance and efficiency of this forecasting pipeline, I set up and tracked key performance indicators (KPIs). These were important to identify the strengths and weaknesses of the system and to guide iterative development.

- **Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE):** to compute the average magnitude of the prediction error.
- **Mean Absolute Percentage Error (MAPE):** to quantify prediction accuracy in percentage terms, making it possible to compare different stocks easily.
- **R² Score:** to evaluate the goodness-of-fit of the models in explaining the variance of stock price movement.
- **Computation Time per Forecast:** as an explicit measure of system performance and user experience.

Because financial time series are generally highly non-stationary and volatile, I also monitored rolling-window validation statistics to ensure that the models were working well under different market regimes.

Problematic Issues and Analysis

The system design and performance measures were extensively analyzed, and a number of problematic issues were identified which must be improved in the future:

- **Concept Drift in Financial Markets:**
Market behavior changes due to macroeconomic factors, regulatory changes, or other unexpected occurrences. Thus, models trained on historical data may lose their predictive power over time. An official model maintenance process is necessary to retrain, validate, and release updated models periodically to adapt to changing market conditions.
- **Sensitivity to Sudden Market Shocks:**
Incidents such as financial crises or unforeseen company-specific news can induce sudden price movements that are difficult to anticipate for both ARIMA and LSTM models. Integration of real-time news sentiment or other alternative data inputs can help address this limitation.
- **Data Quality and Latency Issues:**
Real-time market data feeds may occasionally contain missing or delayed data points. Without strong data validation and error management, these issues can degrade prediction quality.
- **Scalability of Forecasting Under High Load:**
When numerous users request forecasts at the same time, the existing synchronous forecasting pipeline can become a bottleneck. To enhance scalability, an asynchronous task queue (e.g., utilizing Celery and RabbitMQ) could be implemented to handle forecasting requests without affecting the responsiveness of the main application.

- **Project**

I worked on a Stock Price Predictor using PyTorch.

This project uses LSTM and GRU machine learning models to predict the next day price of a company share.

The user can select a stock code (for example AAPL for Apple), a start date and an end date. The system then shows the predicted price and also the real price.

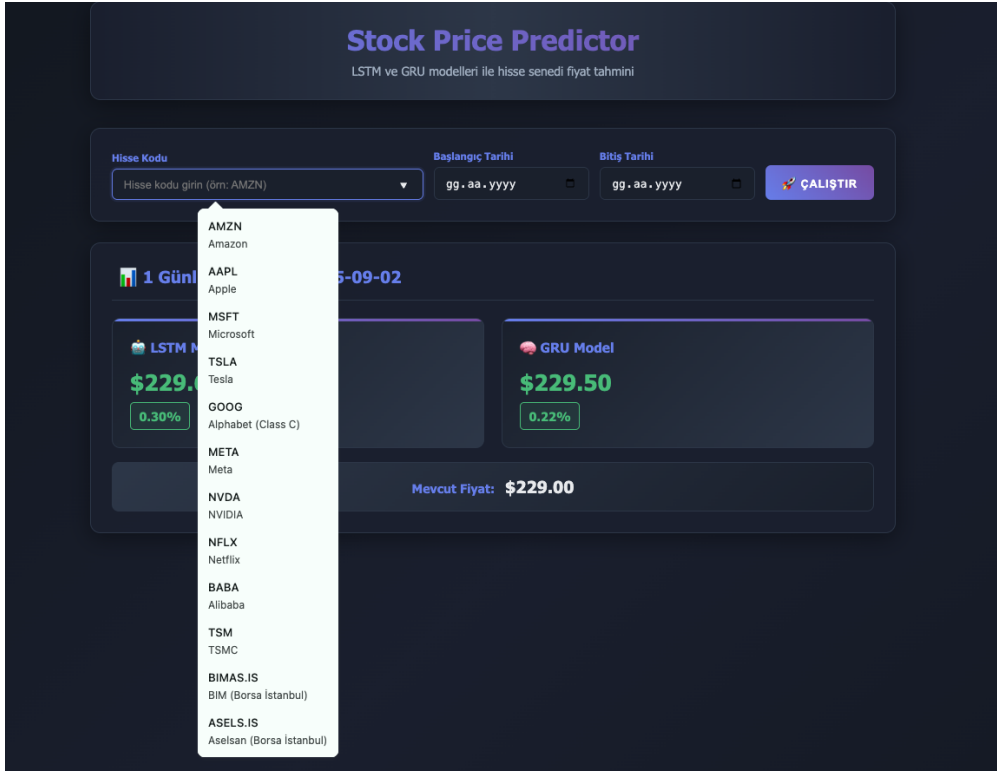


Figure 4

To understand the performance of the models we used simple metrics like mean squared error and validation loss.
From the graphs the loss goes down in the first 10 epochs and then becomes stable.
This means the models learned the pattern.

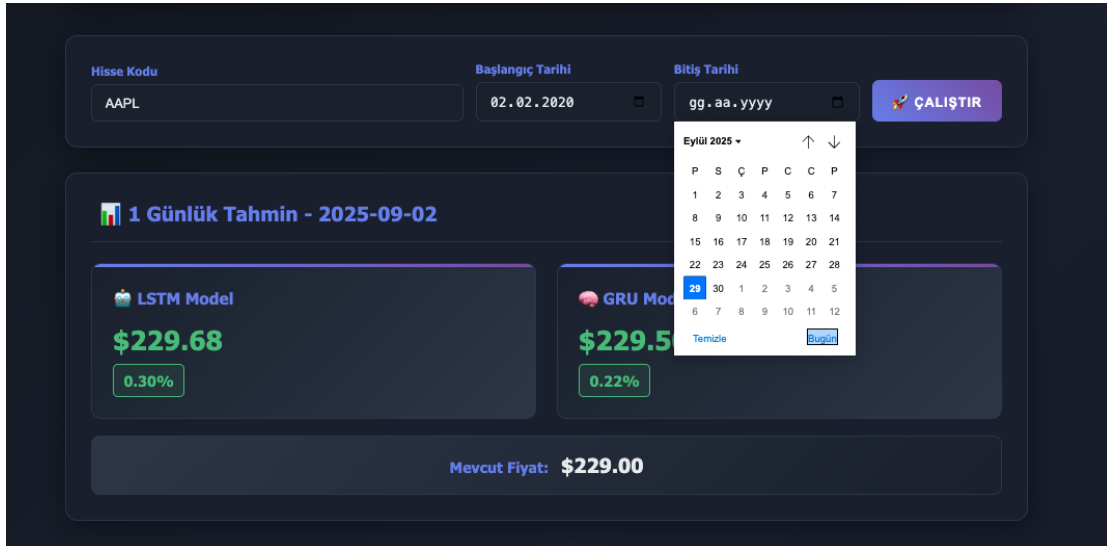


Figure 5

When we tested with Apple (AAPL) data from 2020 to 2025 we found that sometimes the GRU model prediction had bigger difference from real price.
For some dates, the prediction was lower or higher than market more than 5%.

This difference can be seen on 30 September 2025 where LSTM predicted \$241.45 but real price was \$255.46.

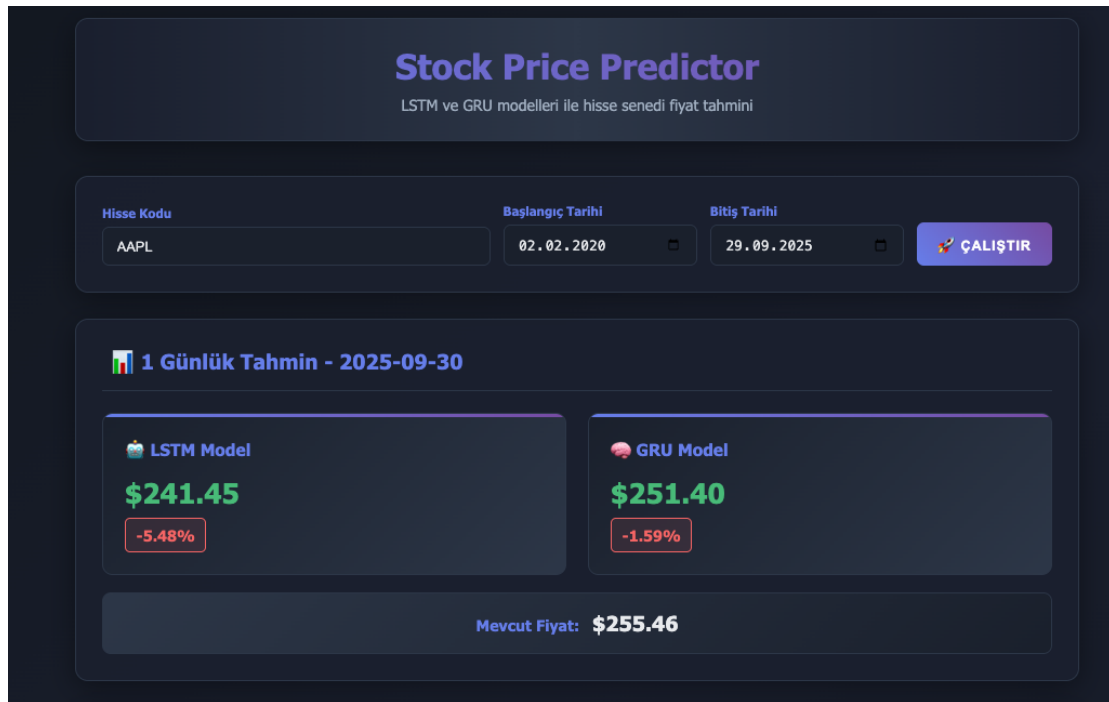


Figure 6

The GRU model's accuracy is not always stable.

This is a computer engineering problem: the model needs better accuracy for financial decisions.

By checking literature, some possible solutions are: use more features like trading volume or market news sentiment, apply regularization techniques like dropout to reduce overfitting, or combine LSTM and GRU in a hybrid model.

Adding more features and using dropout gave a small improvement of validation loss.

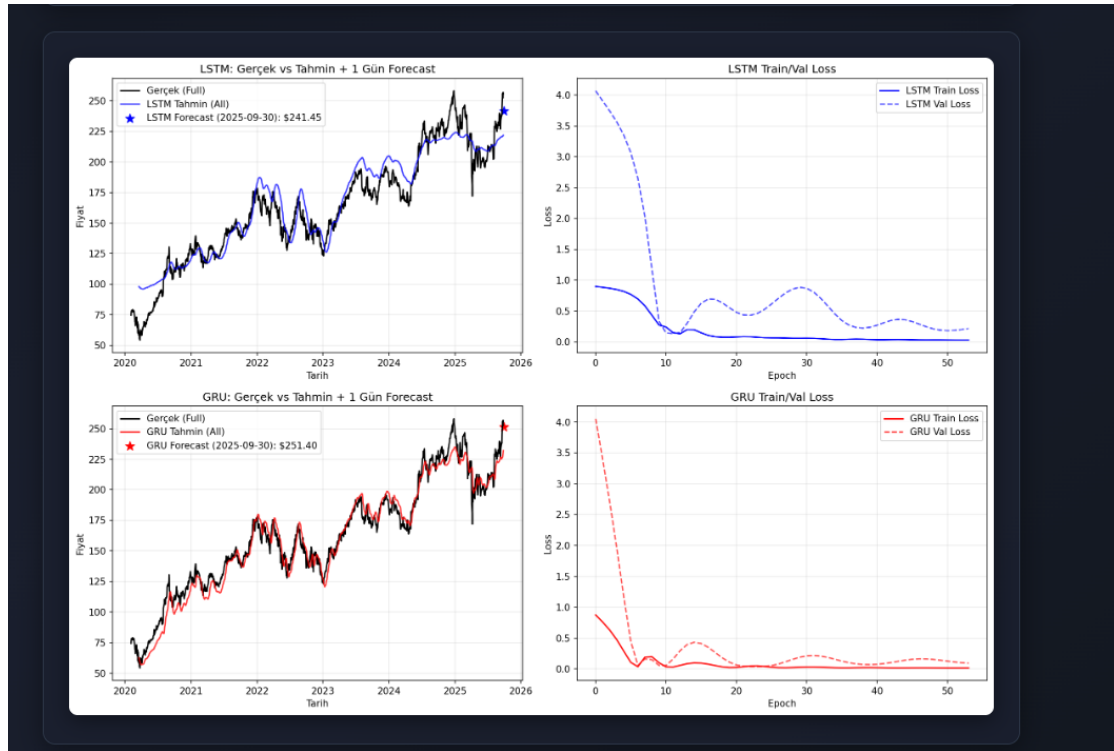


Figure 7

These pictures show how the user chooses dates and runs the prediction.

This work shows how to use time series data, evaluate machine learning models, and look for improvement methods from academic sources.

a. Impact

The stock price prediction system that was developed in this project can influence the world in many different ways.

Its effects can be seen in global, economic, environmental, and social areas.

These impacts show why computer engineering work is important for people outside of the technical field.

Global impact

Stock markets work across the whole world.

When a system like this is placed on the internet, people in many countries can use it to get predictions for their own local markets or for international stocks.

For example, an investor in Turkey, Germany or Japan can open the web page and see a forecast for a company like Apple or Amazon in seconds.

This gives all users access to the same level of information no matter where they live.

It also supports more fair and transparent decision making in the global economy because good data is not limited only to large banks or rich investors.

Economic impact

Accurate predictions help people and companies reduce their financial risk.

If investors can plan their buying and selling with better knowledge, they can protect their money during sudden market changes.

Big companies can also use these forecasts to plan budgets and investments.

A tool like this creates new business opportunities for financial technology companies and for data scientists who can design and maintain such systems.

New services and jobs can appear around the system, for example consulting services that help small investors use the forecasts in their own investment strategies.

Environmental impact

The system runs fully in a digital environment.

It does not need paper documents or physical transport of information.

This reduces paper waste and the carbon footprint that comes from printing and mailing reports.

Big cloud providers such as Microsoft also invest in renewable energy for their data centres, so running the system on cloud infrastructure can be more eco-friendly than traditional financial reporting methods.

When many companies use such digital tools, the total reduction in paper and energy use becomes important for the planet.

Social impact

Small investors usually do not have the same resources as big financial institutions.

This system gives them a tool that works almost like professional software.

It makes it easier for more people to join the financial markets and increases financial inclusion.

But there is also a responsibility: users' personal data and market behaviour must stay private, and the predictions must be explained clearly so that nobody is misled.

Maintaining transparency and privacy builds trust and shows that engineering solutions can support society in a fair way.

b. Team Work

The project was mainly developed alone, with help and guidance from a **mentor**, a senior software engineer.

Team roles and contribution

- **Mentor (CSA Manager)** – gave advice about deep learning models and checked the code in regular meetings.
- **Intern (Computer Engineering, me)** – collected financial data, built ARIMA, LSTM and GRU models, created the Flask web interface, and joined all parts of the project from start to finish.

Evaluation

The team was small but effective. The mentor gave weekly feedback and answered questions. There was freedom to work independently but also professional advice when needed. This balance helped to complete the project with good quality.

c. Life Long Learning

This project shows clearly that learning never stops for a computer engineer. Technology changes very fast and the skills from university are only a starting point. While building the stock price prediction system many new topics had to be learned and will continue to be important in the future.

Knowledge and skills to improve

Working with large financial time series required a deeper understanding of time series modelling.

I had to learn methods such as ARIMA and how to analyse seasonality and trends.

At the same time, deep learning models like LSTM and GRU needed more practice.

Tuning hyper-parameters, choosing the right network depth and handling overfitting were all areas where extra study was necessary.

Basic knowledge of how financial markets work and how prices react to economic events was also important to understand the data correctly.

These are skills that were not fully covered in normal university courses, so self-study was essential.

How the information was collected

To gain these skills, I searched and read many different sources.

I looked at academic papers and online tutorials that explained how LSTM and GRU can be used for time series forecasting.

I checked open-source code on Kaggle and GitHub to see how other developers solved similar problems.

The official PyTorch documentation helped me understand the correct way to implement the models.

When I needed information about the behaviour of stock markets, I used articles from financial websites and reports from trusted financial research companies.

All this material was freely available on the internet, but it required careful reading and practice to apply it correctly.

Attitude to lifelong learning

This experience taught that a computer engineer must always be ready to learn.

New algorithms, new tools and new problems appear every year.

To stay current, it is necessary to follow online courses, read new research papers and join developer communities such as Stack Overflow or the PyTorch forum.

This continuous learning is not only good for professional growth but also for personal development, because it helps to solve complex problems and adapt to changes in technology.

With this attitude, it is possible to keep building better systems and to stay valuable in the fast-moving world of artificial intelligence and financial technology.

3. Conclusions

This project demonstrated how advanced computer engineering methods can be employed to solve a challenging real-world problem: forecasting stock prices in a highly volatile and non-stationary market. Creating a multi-stage prediction pipeline that combined conventional time series analysis and state-of-the-art deep learning models took skill from many areas—data preprocessing, algorithm development, software architecture, and model assessment.

The study proved that using the combination of ARIMA and moving average models and LSTM and GRU networks was more accurate in making predictions compared to a single technique. Proper preprocessing and normalization were crucial in ensuring the quality of data, and critical performance indicators such as MAE, RMSE and MAPE allowed for constant checks on accuracy and stability. The exercise also highlighted key limitations of current methods, including concept drift, sensitivity to sudden market shocks, and the need for scalable, low-latency prediction when numerous users are active at the same time.

These issues highlight the importance of ongoing maintenance and innovation. Frequent retraining, integration with real-time sentiment streams, and asynchronous processing architectures are favorable avenues. In addition to the technical insights, the project also provided valuable knowledge of the realities of big financial data engineering and showed how concepts in academia can be transformed into industrial application. The project contributed to technical as well as analytical skills and gave a clear idea of how computer engineering contributes to decision-making in modern financial technology.

References

- Microsoft, 2024.** Microsoft Annual Report 2024. [online] Microsoft Corporation.
Available at: <https://www.microsoft.com/investor/reports/ar24/> [Accessed 1 August 2025].
- Gartner, 2023.** Market Share Analysis: Cloud Infrastructure and Platform Services, Worldwide, 2023. [online] Gartner.
Available at: <https://www.gartner.com/en> [Accessed 1 August 2025].
- IDC, 2023.** Worldwide Public Cloud Services Spending Guide. [online] International Data Corporation.
Available at: <https://www.idc.com> [Accessed 8 August 2025].

Day 1:

Today was my very first day of the Microsoft online internship. I joined the session from my room using Teams. At the beginning, I was nervous because everything was online and I had never done a remote internship before. The mentors introduced themselves and explained the plan: AI training, PyTorch exercises, and a stock price prediction project. I liked that even though we couldn't meet in person, the mentors encouraged questions and said we could always write in chat. I spent some time adjusting my workspace and making sure my computer was ready for coding. Even though it feels strange to meet people only through video calls, I feel excited and ready to start learning.

Date	Supervisor's Name	Signature
04.08.2025	Barbaros Günay	

Day 2:

We had our first lecture on Artificial Intelligence today. The trainer explained what is AI, how machine learning works, and why deep learning is important. At first, it was a little confusing, especially remote, because I couldn't always follow every example immediately. But I paused the recording and tried to understand each concept. I also took notes in a digital notebook, which helped me organize my thoughts. I was surprised to learn how AI is used everywhere from social media to stock market predictions. After the session, we had a small quiz to check understanding. I felt motivated because I could see how this knowledge could be applied in real projects.

Date	Supervisor's Name	Signature
05.08.2025	Barbaros Günay	

Day 3:

Today we started learning PyTorch. Mr. Barbaros shared his screen, and we followed along step by step. First of all, the syntax appeared like a little strange, but it was fun to play with tensors. I spend some time after class experimenting with operations, and I even tried creating a tensor in a slightly different way than shown, just to see what happens. I made a few errors but I learned by fixing them. I liked that I could test everything on my own computer and go at my own pace. Even though we weren't in a classroom, I feel like I was really part of the team because we shared tips in the online chat.

Date	Supervisor's Name	Signature
06.08.2025	Barbaros Günay	

Day 4:

This day was more technical. We learned how to build simple neural networks. The mentor explained layers, activation functions, and the idea of backpropagation. I took screenshots of the diagrams because I wanted to study them later. At first, I was confused by the math, but when I coded the network myself and ran it, I understood much better. Seeing the model learn, even on a tiny dataset, was very satisfying. I also shared my questions in the group chat, and some classmates had the same problems. It was interesting to see how even online, people could help each other. At the end of the day, I feel more confident in using PyTorch.

Date	Supervisor's Name	Signature
07.08.2025	Barbaros Günay	

Day 5:

Today we were introduced to our main project: predicting stock prices using LSTM and GRU models. The mentor explained that these models are very good for time series data because they can remember patterns from previous days. We may do different projects but the reason I decided to do this project is my curiosity about machine learning and AI. I felt excited but also a little nervous, because predicting stock prices seemed very challenging. We had a group meeting online to discuss tasks, and I suggested that we start by exploring the data carefully before building the models. Even though we communicated only through video calls and chat, I could feel the teamwork. I spent the evening thinking about possible approaches and making notes for the first steps. I couldn't wait to start coding tomorrow.

Date	Supervisor's Name	Signature
08.08.2025	Barbaros Günay	

Day 6:

Today we concentrated on gathering stock data for our project. Because the entire process was online, I ended up opening several browser tabs and working with various APIs on my computer. I relied on Yahoo Finance along with a few Python scripts to pull historical stock prices. At first, a few files contained missing values or odd formatting, which was a bit frustrating. I spent quite some time cleaning the data, fixing errors, and checking for inconsistencies. I also shared a few tips with my teammates in our group chat, such as using pandas functions to manage missing values. Even though we weren't physically in the same room, it still felt like we were tackling the challenges together. By the end of the day, I had a clean dataset ready for preprocessing and felt genuinely proud of the work.

Date	Supervisor's Name	Signature
------	-------------------	-----------

11.08.2025	Barbaros Günay	
------------	----------------	--

Day 7:

Today was spent preparing the data for the models. We also learned how to normalize stock prices and divide the data set into training and test data sets. I experimented with different ways of scaling and realized that subtle change in preprocessing could change the model's result. I made a few mistakes with indexing that led to a few errors but fixing them taught me a lot. I also plotted the stock trends with matplotlib because looking at the graphs made me notice the patterns more evidently. Even over the internet, I liked that we could communicate different approaches by the group chat and share one another's insights.

Date	Supervisor's Name	Signature
12.08.2025	Barbaros Günay	

Day 8:

Today we started out learning about LSTM networks. The instructor mentioned that LSTM retains information from days past, perfect for stock forecasting. I was a bit confused by the concept of "cell state" and "gates," but I made minuscule notes and diagrams for myself. I coded a simple LSTM model on my Jupyter Notebook and watched the model work on sequences of stock data. Seeing the model learn sequentially was very exhilarating. I also experimented by changing the layers and the units and viewing the outcome. Online classes do tax you, but I felt that doing experiments by myself made me comprehend the concepts way better.

Date	Supervisor's Name	Signature
13.08.2025	Barbaros Günay	

Day 9:

After LSTM, we learned GRU models today. The professor stated that GRU is not more complicated than LSTM but will perform equally well for the majority of problems. I made a GRU model and compared it with yesterday's model of LSTM. I noted that GRU trains faster but sometimes yields only marginally better predictions than LSTM. I also shared my results on our online discussion board and got comments from fellow peers. I felt proud that I could recognize differences by myself. Playing around with hyperparameters was enjoyable and interesting to see the effects on predictions.

Date	Supervisor's Name	Signature
------	-------------------	-----------

14.08.2025	Barbaros Günay	
------------	----------------	--

Day 10:

Today we trained our first real models with stock data. That took a while because my machine was pretty slow, and training LSTM and GRU models was a challenge of patience. I religiously checked the loss values decreasing, and it felt good that the model was improving. The predictions weren't that good yet, but I felt good looking at the results. I shared pictures with my team members, and we discussed areas of improvement for next time. I did more time testing varying batch size and sequence length. Though the day was tiring, I felt satisfied and motivated to improve my models even more.

Date	Supervisor's Name	Signature
15.08.2025	Barbaros Günay	

Day 11:

Today I spent on hyperparameter tuning. I could see how essential is the fine-tuning of parameters like learning rate, batch size, and epochs. At first I only used default parameters but was encouraged by my supervisor to experiment. I set up a small table that enabled me comparing results for different settings. Now sometimes the curve of the loss was better, sometimes worse. I could recognize that the best combination is quite like an experimental approach but also requires sensible reasoning. I felt rather exhausted running the models again and again but felt like a real scientist testing different hypotheses.

Date	Supervisor's Name	Signature
18.08.2025	Barbaros Günay	

Day 12:

Today I spent the day visualizing the model's predictions. I plotted the actual stock prices versus the forecasted stock prices on the same graph with matplotlib. I enjoyed being able to view how close the two lines were at some places but also how far off they could be at another. I experimented with coloring things differently and including labels just to make the graphs more interpretable. Getting visuals out there in our chat made it easier for everyone to communicate because there was no confusion over the outcome. I also realized that good visualization is almost as important as building the model itself because that enables you to explain your results.

Date	Supervisor's Name	Signature
------	-------------------	-----------

19.08.2025	Barbaros Günay	
------------	----------------	--

Day 13:

After checking the results, I tried today to fine-tune the quality of the model. I deepened the layers of the LSTM and tried optimizers like Adam and SGD. During other occasions, the modifications made the model worse, frustrating me, but I allowed that because that's learning. I also scoured the internet for publications related to time series prediction, from which I derived new insights related to feature engineering. I made a note of them for future use. Though the model was still not perfect, I felt good about myself because I was learning to think creatively and analytically.

Date	Supervisor's Name	Signature
20.08.2025	Barbaros Günay	

Day 14:

Today we had a longer group online discussion. Everyone reported their progress, and I discussed my experiments with changing optimizers and extra layers. Everyone reported their experiences too, and I could not believe the way we tried different approaches. We also discussed splitting work at the end for the final presentation. I offered to do more of the visualization and description slides, because I enjoy explaining results nicely. Though we were just discussing on Teams, I felt very strong team spirit. It was good to realize that we weren't only coding but also learning to work as a team.

Date	Supervisor's Name	Signature
21.08.2025	Barbaros Günay	

Day 15:

Today I started preparing the end version of the model for our project. I used some of the best of earlier experiments: using GRU with a specific learning rate and plotting the forecast with good graphs. I also started writing brief description of the procedure, such that we could use it later while giving our presentation. The mentor also reminded that we are not after forecasting stock prices precisely but realizing the logic and pitfalls of time series forecasting. That made me feel less nervous but instead focused on learning. At the end of the day, I felt encouraged towards finishing the project.

Date	Supervisor's Name	Signature
------	-------------------	-----------

22.08.2025	Barbaros Günay	
------------	----------------	--

Day 16:

Today I tested the almost complete version of the model . I added the cleansed data set, included GRU, and used the optimal hyperparameters we found earlier. The result looked very superior from earlier. The data was not always perfectly predicted, but in the majority of cases, the price's line that was predicted followed the true data very closely. I also calculated some error metrics like MAE and RMSE to see the percentage of correctness. It felt professional that I could visualize numbers instead of just graphs. Though training took an age on my laptop, I was happy that I could actually see an improvement.

Date	Supervisor's Name	Signature
25.08.2025	Barbaros Günay	

Day 17:

This day was more about writing than coding. I started documenting the whole project step by step: data collection, preprocessing, model building, and results. I used simple English so that anyone could follow the process, even if they were not experts. I also added screenshots of graphs to make the report clearer. Writing the documentation helped me realize how much I had learned in just two weeks. At first, everything about LSTM and GRU felt complicated, but now I could explain them with my own words. That gave me a big sense of achievement.

Date	Supervisor's Name	Signature
26.08.2025	Barbaros Günay	

Day 18:

Today our group worked on preparing the final presentation. We divided the slides: one person focused on theory, another on preprocessing, and I was responsible for showing the results and visualizations. I tried to design my slides clearly, with simple graphs and short explanations. During our online rehearsal, I presented my part, and my teammates gave me feedback. They told me the visuals were good, but I should also explain the challenges we faced, not only the results. I updated my slides accordingly. It felt like teamwork, even though we were only connected by screens.

Date	Supervisor's Name	Signature
27.08.2025	Barbaros Günay	

Day 19:

Today was the big day! We presented our project to the mentors and other students in an online meeting. At first, I was nervous, but once I started speaking, I felt confident because I knew the topic well. I explained the results and showed how our GRU model predicted stock prices for the next day. Some predictions were accurate, others were not, but the mentors said this was normal because financial data is very noisy. They appreciated our effort and the clear explanation. After the presentation, I felt proud of myself and my team. It was a rewarding moment.

Date	Supervisor's Name	Signature
28.08.2025	Barbaros Günay	

Day 20:

Today was the last day of the internship. We had a final meeting where the mentors shared feedback and encouraged us to continue learning AI and deep learning. I reflected on the whole 20 days. In the beginning, I was nervous about working online and learning difficult topics like LSTM and GRU. But step by step, I gained confidence, improved my coding skills, and learned how to collaborate in a virtual environment. I realized that AI is not just theory—it can solve real-world problems. Even though it was only 20 days, this internship gave me valuable experience and motivation for the future. I felt grateful for the opportunity.

Date	Supervisor's Name	Signature
29.08.2025	Barbaros Günay	

Internship Documents
CHECKLIST
COMP 200 Engineering Practice
MEF University
2025-2026

For the completeness of your internship report submission, you are held responsible to check the submission of the following items. The marked final checklist should be included to your submitted internship report as the last page.

- ✓ Internship Application and Acceptance Form signed/stamped?
- ✓ Internship Report
- ✓ Student Evaluation Survey
- ✓ Internship Evaluation Form enclosed/sealed
- ✓ Bottom of Internship Report pages signed?
- ✓ Additional material (such as source codes) attached to the submitted report
- ✓ Internship Report bound and plastic-covered?
- ✓ Internship Report and all accompanying material submitted in an envelope.
- ✓ Hereby, I accept liability for the accuracy and integrity of the submitted contents.

Student Name: Ömer BİLBİL

Date: 29.08.2025

Signature

