CENG 407
Innovative System Design and Development I

2024-2025 Fall

e-TurFinSAS: Entity Based Turkish Financial Sentiment Analysis System

Team 20

Project Report
Version 1

Names of members:

- Taner Onur UYAR Student Number: 202011014
- Ahmet Eren YAĞLI Student Number: 202011038
- Baha ÖÇALAN Student Number: 202011056
- Ahmet Gökay ÜRKMEZ Student Number: 202011057
- Mert Şerafettin KARGI Student Number: 202011061

Advisor: Dr. Gül TOKDEMİR

Co-Supervisor: Sezer UĞUZ

09.01.2025

Table of Contents

List of Figures

ABSTRACT

This project aims to analyse investor behavior and market sentiment by analysing tweets published on social media about Borsa İstanbul (BIST100) companies with the highest market value in Türkiye. With the entity-based sentiment analysis model to be developed in the Turkish Financial Natural Language Processing (NLP) field, each company in the tweets will be detected with the entity name recognition method and the sentiment will be classified as positive, negative or neutral. The model will provide a decision support mechanism for investors and market analysts, allowing them to better predict market trends, while also allowing companies to evaluate public perception and brand strategies. The project aims to create a widespread impact in economic, commercial and social areas. The financial sentiment analysis model has the potential to be developed as a prototype and used in commercial applications. In addition, it is aimed to introduce the model in media and scientific events and contribute to knowledge accumulation through academic publications. This project will create new research areas in financial text analysis and pave the way for graduate studies and future projects. The study is expected to bring significant innovation in strategic decisions in the Turkish financial sector with the potential to establish early warning systems against financial crises.

Key words:

Sentiment Analysis, Natural Language Processing, Turkish Financial Data, Social Media, Investor Behavior

1. INTRODUCTION

Turkish Financial Sentiment Analysis (SA) is crucial for understanding the impact of investor and public sentiment on stock prices and market dynamics. On digital platforms like social media, investors' emotions can directly influence market behavior. In the Turkish financial sector, sentiment analysis aids in predicting investor behavior, managing risk, and supporting strategic decision-making processes.

The goal of this project is to analyse Twitter/X data related to Turkish companies to identify positive, negative, or neutral sentiments. This analysis helps in gaining insights into market trends and measuring public perception of companies, providing a strategic information source for financial decision-makers and investors.

1.1. Problem Statement

The project, titled *e-TurFinSAS: Entity-Based Turkish Financial Sentiment Analysis System*, seeks to address the challenges of understanding social media sentiment associated with entities (such as companies) in the Turkish financial market. Social media platforms, particularly Twitter, generate massive volumes of data daily, reflecting public opinions and emotions that influence investor behavior, brand perception, and market trends. However, analysing this unstructured data effectively is challenging, particularly for Turkish, which has a complex morphological structure and lacks extensive labeled datasets in financial contexts.

1.2. Background or Related Work

The *e-TurFinSAS* project builds on existing research and methods in Natural Language Processing (NLP), Sentiment Analysis (SA), and Named Entity Recognition (NER), particularly in the Turkish financial domain. Several researchers and studies have explored similar challenges, which form the foundation for this project's development.

- **Named Entity Recognition (NER) in Turkish Texts**
  Research on NER for Turkish is relatively limited due to the language's complex morphological structure.

- **Küçük and Yazıcı (2012)** developed a hybrid NER system for Turkish that combined rule-based and machine learning approaches, adapting the model to various domains such as financial texts, news, and historical documents. This study demonstrated the potential for domain-specific customization of NER systems to improve performance.

- **Dinç (2022)** investigated financial named entity recognition in Turkish news, leveraging deep learning techniques like BERT to label financial entities and evaluate the impact of annotation formats on model performance. This work highlights the effectiveness of domain-specific models tailored for Turkish

❖ **Sentiment Analysis (SA) in Financial Texts**
Sentiment analysis has been widely studied in financial contexts to extract public sentiment from textual data and correlate it with market behaviours.

- **Ahmad and Umar (2023)** compared traditional machine learning models, such as Multinomial Naïve Bayes (MNB) and Logistic Regression (LR), with deep learning models like Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) for SA on financial texts. The deep learning models achieved superior accuracy, illustrating their potential for handling large and complex datasets.

- **Gümüş and Sakar (2021)** combined sentiment analysis of financial news with stock price predictions using LSTM models. Their findings demonstrated that integrating sentiment data into predictive models significantly improved the prediction of stock price trends, particularly during periods of volatility.

❖ **Challenges and Advances in Turkish Financial NLP**

- The limited availability of Turkish financial datasets poses challenges for training robust models. Existing studies have relied on creating annotated datasets, such as those introduced in Dinç’s (2022) research, to address this limitation.

- The integration of sentiment analysis with market data for predictive purposes, as shown by Gümüş and Sakar (2021), underscores the value of combining textual and numerical data for better decision-making

1.3. Solution Statement

The creation of e-TurFinSAS: Entity-Based Turkish Financial Sentiment Analysis System is a possible solution for the challenges encountered with analysing social media sentiment related to companies in the Turkish financial industry. This system analyses sentiment in tweets regarding the top 100 businesses listed on Borsa Istanbul (BIST100) using machine learning models and sophisticated Natural Language Processing (NLP) techniques.

The solution's main phase is a multi-stage process:

- **Data Collection:** Sophisticated APIs and scraping methods are used to collect Turkish tweets on BIST100 companies.

- **Data Preprocessing:** Preparing the data for use on informal social media platforms by cleaning and standardizing it in order to fit Turkish's extensive morphological structure.

- **Entity Recognition:** Using specialized NLP models, this method appropriately locates business names and other relevant entities in the text.

- **Sentiment Classification:** Determining if a sentiment is positive, negative, or neutral by using machine learning models like RNNs, LSTMs, or FinBERT.

- **Decision Support:** Merging financial indicators and sentiment analysis results provide concrete insights for analysts, investors, and businesses.

This approach offers a scalable, reliable, and domain-specific solution while resolving the unique linguistic difficulties of Turkish, such as its agglutinative character. In addition to promoting enhanced financial decision-making, it provides new opportunities for financial natural language processing research, contributes to academic knowledge, and makes it possible to create systems that alert people for financial crises.

1.4. Contribution

Background and Related Work

Two well-known methods in Natural Language Processing (NLP) are Named Entity Recognition (NER) and Sentiment Analysis (SA). In contrast to languages like English or Chinese, their use in Turkish financial documents is yet largely unexplored. The agglutinative nature of the Turkish language and the scarcity of annotated datasets are just two of the issues that have been covered in earlier research in this field. For example, hybrid NER systems, which blend machine learning and rule-based methodologies, have demonstrated potential in enhancing flexibility in a variety of fields, including banking. Similar to this, sentiment analysis has demonstrated efficacy in gleaning insights from textual data through the use of both sophisticated deep learning models (such as LSTM and GRU) and conventional machine learning models (such as Multinomial Naïve Bayes and Logistic Regression).

Even with these developments, current solutions frequently fall short of meeting the complex and ever-changing needs of financial markets, especially when it comes to integrating sentiment analysis with market dynamics. The majority of models were created for general-purpose texts or with an exclusive focus on technical indicators, failing to sufficiently account for investor mood.

Motivation

User-generated content has proliferated due to social media platforms' explosive expansion, especially in the financial markets. Tweets regarding businesses that are listed on the Borsa Istanbul (BIST100) provide important information about market trends, investor mood, and brand perception. By creating an entity-based sentiment analysis system specifically designed for Turkish financial texts, this research seeks to close the gap between sentiment on social media and financial decision-making. The technique will make it possible to analyse social media data in real time and categorize sentiment about BIST100 companies as either favourable, negative, or neutral.

2. LITERATURE REVIEW

2.1. Introduction

Turkish Financial Sentiment Analysis (SA) is crucial for understanding the impact of investor and public sentiment on stock prices and market dynamics. On digital platforms like social media, investors' emotions can directly influence market behavior. In the Turkish financial sector, sentiment analysis aids in predicting investor behavior, managing risk, and supporting strategic decision-making processes.

The goal of this project is to analyse Twitter/X data related to Turkish companies to identify positive, negative, or neutral sentiments. This analysis helps in gaining insights into market trends and measuring public perception of companies, providing a strategic information source for financial decision-makers and investors.

2.2. Sentiment Analysis

Sentiment analysis is an approach used to understand the mood and attitude expressed in texts. Twitter can be thought of as a microblogging platform with a diverse demographic structure, consisting of users from various cultures with countless different dialects and jargon. When texts are written informally or contain social media jargon, sentiment analysis requires a more careful and detailed approach [1], often necessitating the inclusion of emojis and abbreviations in the algorithms. The texts under consideration are classified as positive, neutral, or negative. In this section, we will discuss sentiment analysis, Turkish sentiment analysis, and sentiment analysis in financial data.

2.2.1. Sentiment Analysis Levels

Sentiment analysis can be approached at four levels: Document Level, Sentence Level, Phrase Level, and Aspect Level [2]. For our project, the Aspect Level approach is more suitable, as it is preferred for more specific subjects such as products, stocks, and sectors. Therefore, it is deemed appropriate for tracking the sentiment associated with BIST100 companies. These levels are summarized below.

a. Document Level

At this level, sentiment analysis is performed on the entire document, assigning a single polarity (positive, negative, or neutral) to the entire text. It is commonly used to classify large blocks of text, like book chapters or pages. Both supervised and unsupervised learning methods can be applied at this level. However, this approach is often domain-dependent and may be inadequate for more focused analyses.

b. Sentence Level

In sentence-level analysis, each sentence is evaluated individually, and a sentiment polarity is assigned to each. This is especially useful when a document has a mix of sentiments. Sentence-level analysis requires more training data and processing resources than document-level analysis. It is also crucial for handling more complex tasks, such as conditional sentences or ambiguous statements.

c. Phrase Level

Phrase-level analysis identifies and classifies the sentiment of phrases within the text. This approach is beneficial for texts like multi-line product reviews, where each phrase may convey a distinct sentiment. Phrases are generally shorter and may carry clearer sentiments than full sentences, and this level can also provide insights into demographic and socio-psychological characteristics.

d. Aspect Level

In aspect-level analysis, sentiment is assigned to each aspect within a sentence (such as specific products or stocks). Multiple aspects may exist within a single sentence, and each aspect is analysed individually for sentiment. This allows for a detailed understanding of sentiment towards each aspect, resulting in an aggregated sentiment for the entire sentence.

2.2.2. Sentiment Analysis Process

a. Data Extraction

The first step in Sentiment Analysis is collecting or creating text data for analysis. This data can come from third-party sources, web scraping, or various data types.

b. Data Pre-processing

Unstructured data is cleaned and reduced in size to prepare it for analysis. Steps include:

- Tokenization: Breaking text into smaller units.
- Stop Words Removal: Removing irrelevant words.
- PoS Tagging and Lemmatization: Identifying structural elements and root forms of words.

c. Feature Extraction

i. Bag of Words (BoW)

The BoW model converts text into a numerical vector by creating a vocabulary of unique words and encoding sentences based on word frequency or count. However, it ignores word order, sentence structure, and syntax, which can reduce contextual understanding. TF–IDF is a common extension of BoW that considers the importance of words in context.

ii. Distributed Representation

Distributed representation (or word embedding) represents words in vector space where each vector position contributes information. Common methods include:

- **Word2Vec:** A neural network model with CBOW (predicts current word from context) and Skip-Gram (predicts context from current word).
- **GloVe:** Generates word embeddings by capturing global word co-occurrence information from a corpus, with fast parallelized training.

Other methods include **Doc2Vec** and **FastText**, which capture more contextual information.

d. Feature Selection

Feature selection removes irrelevant or redundant features to improve classification accuracy. Main methods include:

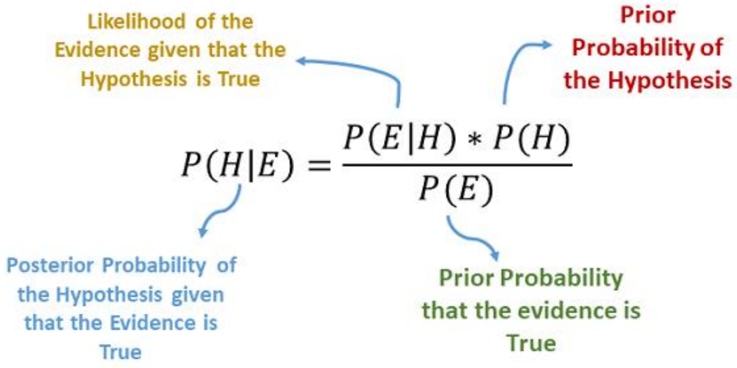
- **Filter Approach:** Selects features based on data properties, without using machine learning. Common measures include Information Gain, Chi-square, and Mutual Information.
- **Wrapper Approach:** Evaluates feature subsets based on machine learning performance but is computationally intensive. Uses methods like Naïve Bayes and SVM with feature subset generation strategies.
- **Embedded Approach:** Integrates feature selection into the model training process, often using decision tree algorithms (e.g., CART, C4.5) and LASSO.
- **Hybrid Approach:** Combines filter and wrapper methods to balance performance and efficiency, often yielding high accuracy for sentiment analysis.[3]

2.3. Sentiment Analysis Methods

a. Machine learning-based approaches

Traditional methods like Naive Bayes and Logistic Regression are widely preferred in sentiment analysis projects due to their speed and relatively low computational requirements [4].

Naive Bayes is known for its efficiency, especially with large datasets. This method classifies text based on probabilities, assuming that each word is independent of the others. Despite its simplicity, it can yield effective results in text classification and sentiment analysis. However, its assumption of word independence can sometimes limit its sensitivity, as it overlooks dependencies within the text.

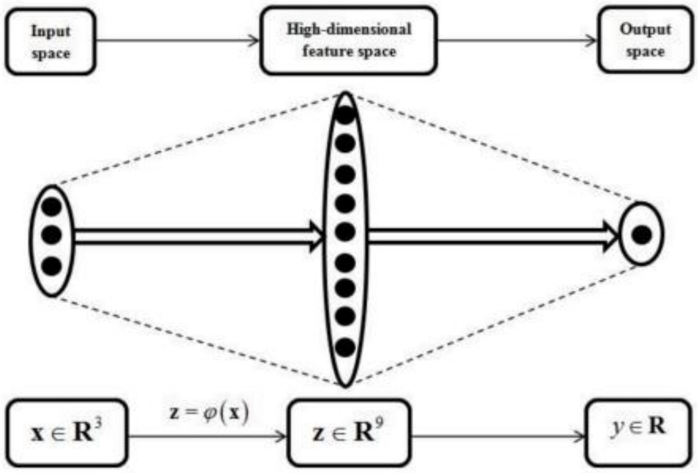


Logistic Regression attempts to classify data using a linear model and calculates the probabilities for class labels. This model can perform well with text data, particularly on datasets with a limited number of features. According to the article, both methods are especially favoured for quick prototyping and basic sentiment analysis tasks, though they may fall short compared to deep learning-based methods like LSTM and CNN.

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost(h_\theta(x^{(i)}), y^{(i)})$$

$$J(\theta) = \frac{1}{m}[\sum_{i=1}^{m} -y^{(i)} log(h_\theta(x^{(i)})) + (1 - y^{(i)}) log(1 - h_\theta(x^{(i)}))]$$

Support Vector Machines (SVMs), developed by Vapnik, have gained increasing acceptance due to their appealing features and strong performance. This method is based on the principle of structural risk minimization, rather than the empirical risk minimization used in traditional machine learning methods, resulting in superior performance. Initially designed to solve classification problems, SVMs have since been adapted to the domain of regression problems as well.



b. Deep learning-based approaches

Deep learning, a subset of machine learning, utilizes deep neural networks and has been widely adopted in sentiment analysis due to its capacity to model complex patterns in text data. Numerous studies review and compare deep learning methods for sentiment analysis. For example, Dang et al. (2020) analysed 32 papers and assessed the performance of models such as Deep Neural Networks (DNN), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN) on multiple datasets. They found that RNN models, particularly with word embeddings, achieved the best results, though at a much higher computational cost compared to CNNs.

Common deep learning models applied in sentiment analysis include CNNs, Long Short-Term Memory (LSTM) networks, and Gated Recurrent Units (GRU). LSTM networks are especially effective for capturing long-term dependencies, making them suitable for sentiment analysis tasks, as noted by Yadav and Vishwakarma (2019), who also highlighted the promise of attention-based networks and Capsule Networks in improving sentiment classification accuracy.

Table 3 Distribution of deep learning papers published in 2020

ID	Algorithm	# of usages	Percentage (%)
1	LSTM	81	35.53
2	CNN	76	33.33
3	GRU	20	8.77
4	RNN	18	7.89
5	Bidirectional Encoder Representations from Transformers (BERT)	7	3.07
6	DNN	4	1.75
7	ReNN	4	1.75
8	Graph Convolutional Neural Network (GCN)	3	1.32
9	Capsule Network (CapsN)	2	0.88
10	Recurrent Convolutional Neural Network (RCNN)	2	0.88
11	Distillation Network (DN)	2	0.88
12	Generative Adversarial Network (GAN)	1	0.44
13	Gated Alternate Neural Network (GANN)	1	0.44
14	Category Attention Network (CAN)	1	0.44
15	Recurrent Memory Neural Network (ReMemNN)	1	0.44
16	Interactive Rule Attention Network (IRAN)	1	0.44
17	Self-Attention based Hierarchical Dilated Convolutional Neural Network (SA-HDCNN)	1	0.44
18	Fusion-Extraction Network (FENet)	1	0.44
19	Deep Q-Network	1	0.44
20	Autoencoder	1	0.44
	Total	228	100

c. Lexicon-Based Approaches

Lexicon-based approaches identify sentiment by scanning for positive or negative words listed in lexicons. This method doesn't require training data but is domain-sensitive, as different domains (e.g., movie reviews vs. Twitter) need tailored lexicons. Lexicons are built via dictionary-based or corpus-based approaches, where the former uses synonym/antonym expansion and the latter identifies sentiment words through co-occurrence in domain-specific corpora.

d. Hybrid Approaches

Hybrid approaches combine machine learning and lexicon-based techniques to leverage the strengths of both methods. Recent research integrates symbolic and subsymbolic AI, applying deep learning to detect patterns and symbolic AI to build commonsense knowledge bases like SenticNet. These approaches aim to enhance natural language understanding by combining pattern recognition with structured knowledge. Hybrid models like LSTM-CNN ensembles also show improved sentiment analysis performance by integrating both model type's strengths.[5]

2.4. Financial Sentiment Analysis

What is Financial Sentiment Analysis?

Financial Sentiment Analysis is a technique used to understand investor sentiment in financial data, particularly in stock predictions. By analysing data from platforms like Twitter, this approach examines the relationship between stock prices and investor moods. Studies show that positive sentiment in tweets can correlate with upward trends in certain stocks, making sentiment analysis a valuable tool in financial forecasting [6].

Key Studies and Methods:

Research in this field explores how social media sentiment affects stock prices, using techniques from natural language processing (NLP) and machine learning. A study published by PLOS ONE analysed the effects of Twitter sentiment on stocks within the Dow Jones Industrial Average. This research utilized event-based analysis to identify significant short-term correlations between sentiment and stock returns during specific events. Such event-based studies indicate that sentiment-driven data, especially during key events, can provide valuable insights into stock valuation [7].

In another notable study, the Federal Reserve developed the Twitter Financial Sentiment Index (TFSI) to observe correlations between financial sentiment and economic variables such as equity returns, bond spreads, and monetary policy. The TFSI leverages FinBERT, a specialized NLP model designed to analyse sentiment in financial text. Models like FinBERT, RNNs, and LSTMs are effective in capturing long-term dependencies in text, making them suitable for understanding market reactions based on sentiment.

These studies highlight the role of advanced NLP and deep learning methods in financial sentiment analysis, demonstrating how models like LSTM, RNN, and BERT can help predict sentiment-based stock movements by analysing social media data.

2.5. Natural Language Processing (NLP) in Turkish Texts and Special Challenges

Studies in Turkish natural language processing (NLP), especially in certain areas such as financial texts, are quite interesting and complex. The agglutinative structure of Turkish, ambiguities and implicit emotional expressions are some of the main difficulties encountered in natural language processing processes.

- **Agglutinative Structure of Turkish**

Derivation of roots with various suffixes in Turkish offers many possibilities that change the meaning and function of the word. This makes morphological parsing difficult, especially for language models. Any word can have hundreds of derivations, which makes word vectors and language comprehension models more complex. Morphological parsing and root finding methods for NLP models should therefore be considered more comprehensively.

- **Ambiguities of Meaning**

In Turkish, the same word can have different meanings with different suffixes or even with the same suffix. In financial texts, terms like “interest” or “stock market” can have different meanings depending on the context. This can be challenging for NLP models trying to understand the correct context. Semantic parsing techniques in particular need to be effective in resolving these ambiguities.

- **Implicit Emotional Expressions**

Emotional expressions in Turkish texts are usually indirect and can be interpreted differently depending on the context. Identifying positive or negative emotional expressions in financial texts is important for decision support systems. Therefore, emotional analysis in Turkish NLP usually requires higher-level features and requires models that require fine-tuning.

2.5.1. Examples of NLP Studies in Turkish Financial Texts in Literature

a. Morphological Segmentation Techniques

Due to the rich morphological structure of Turkish, most NLP methods used in financial texts are based on morphological analysis. Tools such as Zemberek or TRmorph are used to separate the roots and suffixes of Turkish words. Especially in the analysis of financial reports, correct parsing is of great importance.[8][9]

- **ZEMBEREK**

What is Zemberek-NLP?

Zemberek-NLP is an open-source Turkish natural language processing library. It was specifically developed to understand the complex language structure of Turkish and to make sense of texts. The library can perform morphological analysis of Turkish texts, grammar checking, sentence parsing and many more operations. Zemberek is an indispensable resource for Turkish natural language processing projects.

Features of Zemberek:

- Can parse sentences in Turkish texts and identify the structural elements of the sentence.
- Perform morphological analysis of Turkish texts. Morphological analysis provides detailed information about the roots, stems, suffixes and grammatical information of a word.
- It is effective for correcting spelling errors in Turkish texts.
- It breaks the text into understandable parts and separates the words in the text.
- It provides enhanced tools for language analysis and machine learning-based tasks.
- Easy to integrate and use in language processing projects

- **TRmorph**

What is TRmorph?

TRmorph is a relatively complete finite-state morphological analyser for Turkish. The current version of the analyser is licensed under the terms of GNU LGPL.

b. Sentiment Analysis

Studies on Turkish financial news and social media posts often include sentiment analysis algorithms. Language models such as Turkish BERT or XLM-R, trained according to the specific language structure of Turkish, are used to predict sentiment trends in financial news.[10][11]

- **BERT**

What is BERT?

The BERT Algorithm, like many other algorithm updates by Google, was developed to better understand queries and provide more accurate results to its users. The BERT algorithm can be explained as a natural language processing technique that uses artificial intelligence and machine learning technologies together.

For example, before the BERT algorithm update, when a Brazilian traveller made this query on Google about a US visa, they would encounter the Washington Post's news. Because Google could not correctly understand the user's intention here and presented the user with such a result. However, after the BERT algorithm understood the query more accurately, especially with the "to" conjunction in the query, and started to directly present the embassy and consulate pages to the users.

- **XLM-R**

What is XLM-R?

It is a natural language processing model developed by Facebook AI and designed to improve the performance of language modelling across languages. As a multilingual version of the RoBERTa model, it has been pre-trained on 100 different languages, including Turkish. XLM-R shows strong performance on texts in different languages, especially by bridging the gap between language structures and meanings. The model is suitable for working on morphologically rich and structurally diverse languages such as Turkish.

In context-sensitive texts such as Turkish financial news, XLM-R can analyse sentiment trends more accurately thanks to its contextual interpretation power. This model is widely used in tasks such as sentiment analysis, text classification, and translation.

c. Word Embeddings

Word representation models such as Word2Vec and GloVe are used to better capture the meaning in Turkish financial texts. In particular, domain-specific embeddings are useful in capturing the complex semantic relationships in Turkish financial language.[12][13]

- **Word2Vec**

What is Word2Vec?

Word2Vec is an unsupervised and prediction-based model that tries to express words in vector space and there are two main methods used to convert words into vectors.

CBOW: This method uses surrounding words to guess a particular word, thus learning its meaning by considering the context around the words.

Skip-Gram: This method tries to predict surrounding words using a selected word. It can be more effective than CBOW, especially on low datasets.

- **GloVe**

What is GloVe?

It is a model developed by Stanford University that considers the global context of words. Instead of relying on neighbouring words in context like Word2Vec, GloVe creates a vector based on the common occurrences of words in the entire text. In this way, it determines the meanings of words by obtaining more information from their overall distribution.

2.6. Named Entity Recognition (NER)

Named Entity Recognition is used to extract information, question answering and sentiment analysis by getting strings which includes sentence, paragraph, or document, recognizing and classifying the key entities that belong to each category. Names of people, organizations, locations, stocks and dates are some examples.

2.6.1. NER in Financial Texts

The need for clear identification and classification of entities like market terms, companies, financial reports, and news makes Named Entity Recognition more crucial in finance. Because NER allows efficient data extraction by identifying and categorizing key entities, this supports a wide range of financial tasks like sentiment analysis, news and report classification and risk prediction.

2.6.2. NER in Turkish Texts

Using NER for Turkish language is quite challenging compared to English. This is because for few reasons. [14][15]

a. Agglutination

In Turkish, words are formed by adding multiple suffixes to the root word. Also, a word can have more than one suffixes, and this makes morphological variations that affects the complexity of NER system. Sometimes this can create long and hard to understand word forms for traditional tokenizers.

b. Word Order

Turkish is flexible about word order. It means position of key entities can change sentence to sentence.

c. Lack of Resources

Turkish has fewer pre-existing resources for Named Entity Recognition like pre-trained models and articles. The lack of resources makes harder to train a NER model for Turkish language.

2.6.3. NER Methods[16]

a. Rule-Based Methods

This approach is about identifying and creating a set of rules for the grammar of that language. These rules are used to classify entities in the text considering the grammar rules. Rule base methods are great in specific domains where entities are identified clearly, but they can be time consuming and hard to work with large data.

b. Statistical and Machine Learning Methods[17]

Statistical Methods predicts named entities based on training data. Like Hidden Markov Models (HMM) and Conditional Random Fields (CRF). HMM predicts sequence of tags based on observed data. CRF captures dependencies between words in text. They are great as long as training data is good. In Machine learning various algorithms like decision trees or support vector machines (SVM) can be used

c. Deep Learning Based Methods

Recurrent Neural Networks (RNN) and transformer models have ability to model long-term dependencies in the text. To handle with rare words, they capture sequential data. Their best use case is large scale tasks with abundant training data.

d. Hybrid Methods

Combines rules-based and machine learning methods to use the benefits of both. Despite the complexity, this method offers flexibility for various sources. Rule-based can quickly identify entities that are easy to recognize and ML to identify entities that are more complex.

2.6.4. NER Process Steps[18]

a. Data Collection

The first step is making a clear dataset. Dataset contains examples of text with named entities.

b. Data Preprocessing

After the data is collected, data should be cleared from unnecessary words and characters. Then split the text into tokens to work with each part individually. Lemmatization means reducing the word to their root forms and it can be done if it is necessary.

c. Feature Extraction

This step includes word embeddings, contextual features, and lexical features. In word embeddings, words are represented as vectors embeddings to capture semantic meaning. Contextual Features means considering near words for each token, and that gives cues for identifying entities. Feature choices can depend on the NER model that match with the system needs.

d. Model Training

The next step is training a NER model with labeled dataset using ML or deep learning. The Model will learn to recognize patterns and relationships between words in text with their related named entity labels.

e. Model Evaluation

Model evaluated with performance metrics like recall, precision, or F1-score. These metrics Then with the evaluation data, errors or low performance can be revealed and further improvements can be made.

f. Model Improvements

In this step, based on the evaluation data, improvements are made for performance and errors if there is. Training data or model can be modified and adjusted accordingly for the requirements. Entity linking or normalization can be made if output of the NER model needs to be refined for better results.

g. Result

In the last step, model can be tested on a new text. The model will apply preprocessing steps and extract needed data from the text, then the model will predict the named entity labels for each token in the text.

2.7. Data Collection and Preprocessing

2.7.1. Data Sources and Data Collection

It is possible to obtain the Twitter (X) and BIST100 data that we will use in this project by various methods. Some of these methods are:

- **Borsa Istanbul Data API:** In the database and API services offered by Borsa Istanbul, APIs allow you to obtain the prices, volumes, and index values that companies trade at daily or at certain intervals. Obtaining data through official APIs or data services can be one of the most definitive ways to ensure the reliability of the data.
- **Twitter (X) API:** It is possible to collect tweets using specific keywords, hashtags or accounts via Twitter API provided by Twitter (X). Access to past tweets can be accessed with Academic Research Track and a wider data set can be obtained. In addition, Twitter's paid Premium Search API can be used to pull even older tweets or more data. Premium Search API

allows you to perform more in-depth historical searches and can provide data over a wider date range. Main API Functions: BIST100 company names, symbols or specific hashtags can be searched with Search Tweets endpoint. (e.g. “$BIST” or “#BIST100”). Tweets containing specific keywords can be continuously pulled with Filtered Stream Endpoint. In addition, access to the API can be provided using Python libraries such as Tweepy or . These libraries make it easier to pull tweets and structure the data.

- **5.1.3 Web Scraping:** Web scraping is defined as the process of extracting data from pages on the internet. These processes are generally used for data analysis and research. Since it is possible to access current or historical data about BIST100 companies on many financial sites, web scraping is one of the alternatives in the data collection process. In order to collect data such as current stock prices, transaction volumes, and price changes of BIST100 companies through web scraping, web pages that provide such data should be determined. Data obtained from financial sites such as TradingView, Investing.com, Yahoo Finance or financial news sites that include current news and economic situation analyses can be used in analyses about BIST100 companies.

The web scraping process generally consists of the following:

- Creating a Site Map: First, it is determined which pages of the target site will be used to extract information and the HTML structure is analysed. It is defined which HTML elements contain data (for example, <div>, <span>, or <table>).
- Determining HTML Tags and XPath: The HTML elements containing the target data are determined. For example, it can extract certain HTML tags with the methods included in the BeautifulSoup library. Specific elements can be accessed with XPath expressions using Scrapy or Selenium.
- Retrieving and Processing Data: A request is sent to the web page and the necessary information is extracted.

- Cleaning and Saving Data: The retrieved data is usually in raw form. For example, price data can be in string (text) type. Converting the data to numerical form when necessary and saving it to a CSV file or database is done at this stage.

2.7.2. Data Cleaning and Preprocessing

In our project, data preprocessing is a critical step in order to provide meaningful analysis of the data collected from Twitter. Raw data usually contains noise, which reduces the accuracy of the analysis. The proposed data cleaning and preprocessing techniques provide standards for text mining and sentiment analysis.

2.7.3. Data Cleaning

2.7.3.1. Extracting Usernames, URLs, and Numbers:

Raw tweet data often contains usernames (@username), URLs, and meaningless numbers. Extracting usernames and URLs reduces text length and noise, resulting in more accurate sentiment analysis results. This problem can be solved by identifying and removing usernames and URLs using Python Regex or string manipulation methods.

2.7.3.2. Removal of Special Characters and Punctuation Marks:

Special characters (^, &, \, etc.) and punctuation marks that are frequently encountered in the texts we will obtain from Twitter need to be cleaned since they generally do not carry any meaning in data mining. Removing punctuation marks reduces the data size resulting from unnecessary words. The "string.punctuation" and "re.sub" methods [20] can be used to remove special characters and punctuation marks.

2.7.3.3. Uppercase-Lowercase Conversions:

The machine learning algorithms we will use may perceive uppercase and lowercase letters as different words. To prevent this data conflict, it is necessary to convert all text to lowercase and make the data uniform. This conversion can be easily done in Python using the "str.lower()" function.

2.7.3.4. Converting Emoji and Special Characters to Text:

Emojis, which are frequently used on Twitter as well as on every platform in social media, have meaning in terms of sentiment analysis. Because some users can express their feelings and opinions not only through text but also through tools such as emojis and special text. Converting the meanings of these emojis to text is of great importance in terms of Sentiment Analysis. Emojis can be converted into meaningful words using the Python Demoji library.

2.7.4. Data Preprocessing

2.7.4.1. Removing Stop Words:

Stop words (e.g. "and", "but", "one") are words that do not contribute to sentiment analysis. If these words are removed while preserving the meaning of the text, the data we collect will be free from unnecessary data load. In Python, NLTK or spaCy libraries can be used to identify and remove these words.

2.7.4.2. Tokenization:

Tokenization is defined as the process of separating sentences into meaningful words. Tokenization is very important in ensuring that data is divided into meaningful units for sentiment analysis. Tokenization can be performed using NLTK's "word_tokenize()" function or spaCy in this process.

2.7.4.3. Stemming and Lemmatization:

Stemming and stemming techniques are used to reduce words to their most basic roots. For example, the words "working" and "worked" are converted to the same root ("work"). This makes the data we will use more consistent when processing the data and prevents the model from shifting in meaning. NLTK's

"SnowballStemmer" or spaCy's "Lemmatizer" functions can be used for stemming and stemming.

In addition, since the data we will capture will be used in numerical analysis, scaling and normalization are also very important. According to the article Data Preprocessing for Stock Price Prediction Using LSTM and Sentiment Analysis [19], scaling the data within a certain range increases the performance of LSTM models, which are especially strong in time series and text analysis and effective in areas such as stock predictions and sentiment analysis. Tools such as "MinMaxScaler" can help in this process.

As a result, Data Cleaning and Data Preprocessing techniques play a critical role to model the raw data and obtain more consistent results in the project. These techniques that can be used for this project (especially in financial data and social media text analysis) should be adapted with project specific methods to obtain higher quality and meaningful data.

2.8. Previous Works

There have been previous studies attempting to predict the trends of BIST30[21] or BIST100 indices by examining the stocks of multiple large companies that significantly influence these indices, as well as analysing news about them through sentiment analysis. In some stocks, an accuracy rate of 86.56% was achieved using FastText and LSTM [22], and here are more examples with different algorithms with pictures[23].

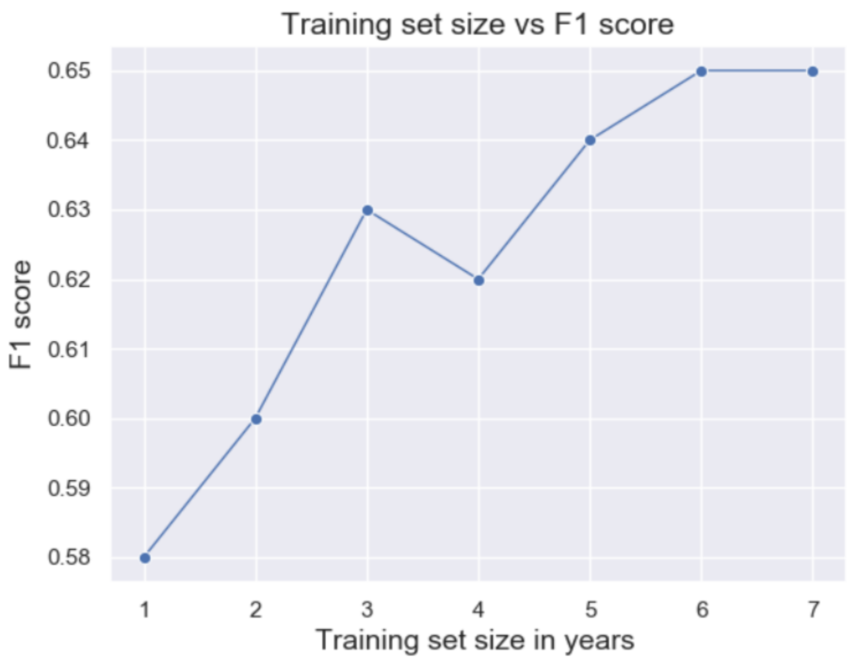
Stocks	Word2Vec	LSTM	Word2Vec+LSTM
AKBNK	84.17	85.39	87.25
ALBRK	87.23	88.05	89.46
GARAN	76.65	78.23	78.98
HALKB	75.96	76.85	77.25
ISCTR	77.16	77.54	79.46
SKBNK	78.57	79.45	79.46
TSKB	74.72	75.93	75.20
VAKBN	78.80	78.64	79.46
YKBNK	77.05	78.12	78.98
Avg.	78.92	79.80	**80.61**

Stocks	FastText	LSTM	FastText+LSTM
AKBNK	89.55	89.23	89.94
ALBRK	89.27	89.00	89.56
GARAN	88.75	88.72	88.15
HALKB	85.43	85.15	85.57
ISCTR	80.27	80.72	80.65
SKBNK	80.78	80.66	81.39
TSKB	81.24	82.80	83.03
VAKBN	87.90	88.24	89.23
YKBNK	88.00	89.37	90.05
Avg.	85.69	85.99	**86.39**

Stocks	Word2Vec	RNN	Word2Vec+RNN
AKBNK	77,25	78,33	80,02
ALBRK	77,10	78,40	80,07
GARAN	78,42	78,77	81,18
HALKB	78,77	78,90	80,11
ISCTR	75,23	75,27	77,00
SKBNK	71,36	72,34	73,87
TSKB	74,30	74,06	75,57
VAKBN	75,50	75,25	76,96
YKBNK	74,80	74,00	75,56
Avg.	75,86	76,15	**77,82**

A comprehensive study has been conducted on sentiment analysis (SA) of financial texts. Sentiment analysis seeks to determine individuals' emotions and attitudes toward specific topics or products using text data. This research extracted sentiment polarity—negative, positive, and neutral—from financial texts through machine learning and deep learning algorithms. The machine learning approach applied Multinomial Naïve Bayes (MNB) and Logistic Regression (LR) classifiers, while deep learning techniques included Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) models. MNB and LR achieved good and very good accuracy rates, respectively, while RNN, LSTM, and GRU demonstrated excellent accuracy rates. The findings suggest that preprocessing steps positively impacted accuracy [24].

In another study for sentiment analysis, FastText was used, a library known for its efficient word vector representations, particularly for morphologically rich languages like Turkish. The FastText model classified Turkish news articles as positive, negative, or neutral based on manually labeled training data of 500 samples [25]. The LSTM-RNN was chosen to capture sequential dependencies in time-series data, making it suitable for stock trend predictions. Experiments on 8 years of data revealed that the model's F1 score, which was around 0.56 without sentiment analysis, increased to 0.65 when sentiment labels were included. The results suggest that the model with sentiment labels better fits actual prices, particularly during periods of high stock price volatility.



Our goal is to improve these accuracy rates by using similar or different models and leveraging the increased amount of data available due to the rise in social media usage. In undertaking sentiment analysis of financial textual data, several challenges are anticipated based on similar studies and prior research in the field. Addressing these issues is essential to enhance model accuracy and efficiency.

First problem we will face is probably Data Quality and Complexity: Financial text data, especially from social media, often includes posts that are either overly lengthy, contain complex or nuanced emotional content, or use non-standard characters and symbols [26]. Such variability in data quality can complicate preprocessing and impact model performance. To mitigate this, rigorous preprocessing steps will be implemented, including data cleaning, tokenization, and feature selection. These steps aim to standardize and simplify the input text data, allowing models to better capture sentiment.

Real-Time Analysis may be the second one, running algorithms on big data such as tweets could be hard to produce some results fast. We can call it as Limited Optimization Techniques: Optimization techniques, such as swarm or bat optimization, can enhance model accuracy and reduce computation time, but these approaches remain underexplored in financial sentiment analysis [27]. To maximize model performance, our study will implement advanced optimization algorithms during the training phase. These techniques aim to refine the model parameters dynamically, potentially improving sentiment classification without sacrificing speed.

High Volatility in Stock Prices: Stock prices are inherently volatile, and sentiment analysis can sometimes be insufficient to capture rapid market shifts. Studies have shown that sentiment analysis models tend to perform better during stable periods but struggle when there is high volatility [28]. This limitation suggests that while sentiment data adds value, further enhancements in feature engineering and model selection are necessary to account for extreme market changes.

3. PROJECT PLAN

Start Date: 30/09/2024		WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7	WEEK 8	WEEK 9	WEEK 10	WEEK 11	WEEK 12	WEEK 13	WEEK 14	WEEK 15	WEEK 16
Procedural Steps	**Current Status**	30.09.2024	07.10.2024	14.10.2024	21.10.2024	28.10.2024	04.11.2024	11.11.2024	18.11.2024	25.11.2024	02.12.2024	09.12.2024	16.12.2024	23.12.2024	30.12.2024	06.01.2025	13.01.2025
Team Setup	Done																
Project Proposal Form	Done																
Project Selection Form	Done																
GitHub Repository	Done																
Project Work Plan	Done																
Literature Review	Done																
Software Requirements Specification	Done																
Project Webpage	Done																
Software Design Description	Done																
Project Report	Done																
Presentation	Waiting…																

Figure 1. Project Work Plan

4. SOFTWARE REQUIREMENTS SPECIFICATION

4.1. Introduction

4.1.1. Purpose

The purpose of this project is to conduct entity-based sentiment analysis of social media posts (tweets) about the top 100 companies listed on the Borsa Istanbul (BIST100), which have the highest market value and trading volume. By analysing these posts, the project aims to classify sentiments as positive, negative, or neutral, with the following objectives:

- Supporting investors in their decision-making processes.
- Better predicting market trends.
- Enabling companies to evaluate their social media image.
- Developing early warning mechanisms for financial crises or unexpected events.
- Providing a new data source for testing behavioural finance theories.

4.1.2. Scope of Project

This project aims to deeply analyse the relationship between social media sentiment and financial market dynamics in Turkey. It focuses on the 100 companies with the highest market value and trading volume traded on Borsa Istanbul (BIST100). Within the scope of the project, Turkish tweets shared on the Twitter/X platform regarding these companies will be collected and an asset-based sentiment analysis will be performed on these tweets. As a result of the analysis, it is aimed to obtain concrete findings regarding the impact of the perception formed in social media on financial markets and investor behavior by determining the positive, negative or neutral sentiments of the posts. In order for the project to achieve these goals, advanced Natural Language Processing (NLP) and Machine Learning (ML) techniques will be used, considering the unique challenges of Turkish.

In order to address the complex language features of Turkish, such as its agglutinative structure and morphological richness, pre-trained models such as BERT will be utilized, and these models will be customized specifically for Turkish. In addition, a powerful preprocessing process will be developed to clean elements such as noise (slang, abbreviations, emojis) commonly seen in tweets and to standardize the texts. Within the scope of the project, the scope of the model will be expanded to correctly understand the terms and jargons frequently used in financial texts.

The performance of the sentiment analysis model will be evaluated using well-known metrics such as accuracy, F1 score, precision, recall, and AUC-ROC curve. The findings from this project will be usable in various fields. For example, it will provide a clearer picture of public perception towards companies to help investors make more informed decisions. Companies will be able to evaluate their social media image and brand perception with this data and develop improvement strategies. In addition, this project will create a new data source that will contribute to behavioural finance studies to examine the impact of emotional reactions on financial decisions.

However, the project also aims to contribute to the development of decision support systems. In order to create early warning mechanisms, especially for financial crises or unexpected events, the data obtained from sentiment analysis will be integrated with traditional financial indicators. In this way, more effective methods for predicting market trends and volatility can be developed. In general, this project aims to provide valuable tools and insights for investors, analysts and companies by combining social media analytics with financial market research.

Glossary Table of SRS

Term	Definition
BIST100	Collecting Turkish-language tweets about BIST100 companies from the Twitter/X platform.
Sentiment Analysis	Developing an entity-based sentiment analysis model tailored to the morphological richness of the Turkish language and financial texts.
Natural Language Processing (NLP)	A subfield of artificial intelligence focused on enabling computers to understand and process human language.
Machine Learning (ML)	A field of artificial intelligence that enables computers to learn from data without explicit programming.
Entity-Based Sentiment Analysis	A method of analysing sentiments specifically associated with an entity (e.g., a company).
Twitter/X	A social media platform where users share short text-based posts for communication.
F1 Score	A metric that balances precision and recall measuring a model's performance.
Pre-trained Model	A machine learning model that has been trained on a large dataset and can be finetuned for a specific task.
Morphological Richness	The ability of languages like Turkish to form derivatives of words through the addition of suffixes, creating complexity in language processing.

4.1.3. Overall Description

4.1.3.1. Product Perspective

This project utilizes advanced NLP and ML techniques to perform entity-based sentiment analysis on tweets about the top 100 companies listed on Borsa Istanbul (BIST100). By classifying tweets as positive, negative, or neutral, the system provides actionable insights into social media sentiment surrounding these companies.

Designed as a decision-support tool, it aids investors and market analysts in predicting trends and making data-driven decisions. Companies can also evaluate their public perception to refine strategies. Additionally, the system supports early warning mechanisms for detecting potential risks, ensuring timely and informed actions.

Tailored to the Turkish financial ecosystem, this product leverages big data analytics to enhance market analysis and strategic planning.

4.1.3.2. Software Methodology

We decided to use the agile methodology while doing our project. Agile methodology is a flexible and collaborative approach used in software development and project management. In short, agility is a paradigm shift in the usual way of working. The word agile has passed into Turkish as "çeviklik". The concept of agile, which is the ability to adapt quickly to changing conditions; Although it is defined as method, methodology, method or project management in most sources, it is a way of thinking contrary to these. At the core of this methodology are the principles that individuals and interactions are more valuable than processes, a working product is more important than comprehensive documentation, customer collaboration is more important than contract negotiations and adapting to change is more valuable than following a plan. Agile aims to deliver projects faster, increase customer satisfaction, and adapt quickly to changing requirements through small, manageable work cycles (e.g., sprints). Although it was initially developed for software development, today it is widely used in different sectors such as marketing, finance and manufacturing. Adopting a continuous improvement and value-oriented approach, Agile enables teams to work more efficiently and effectively.

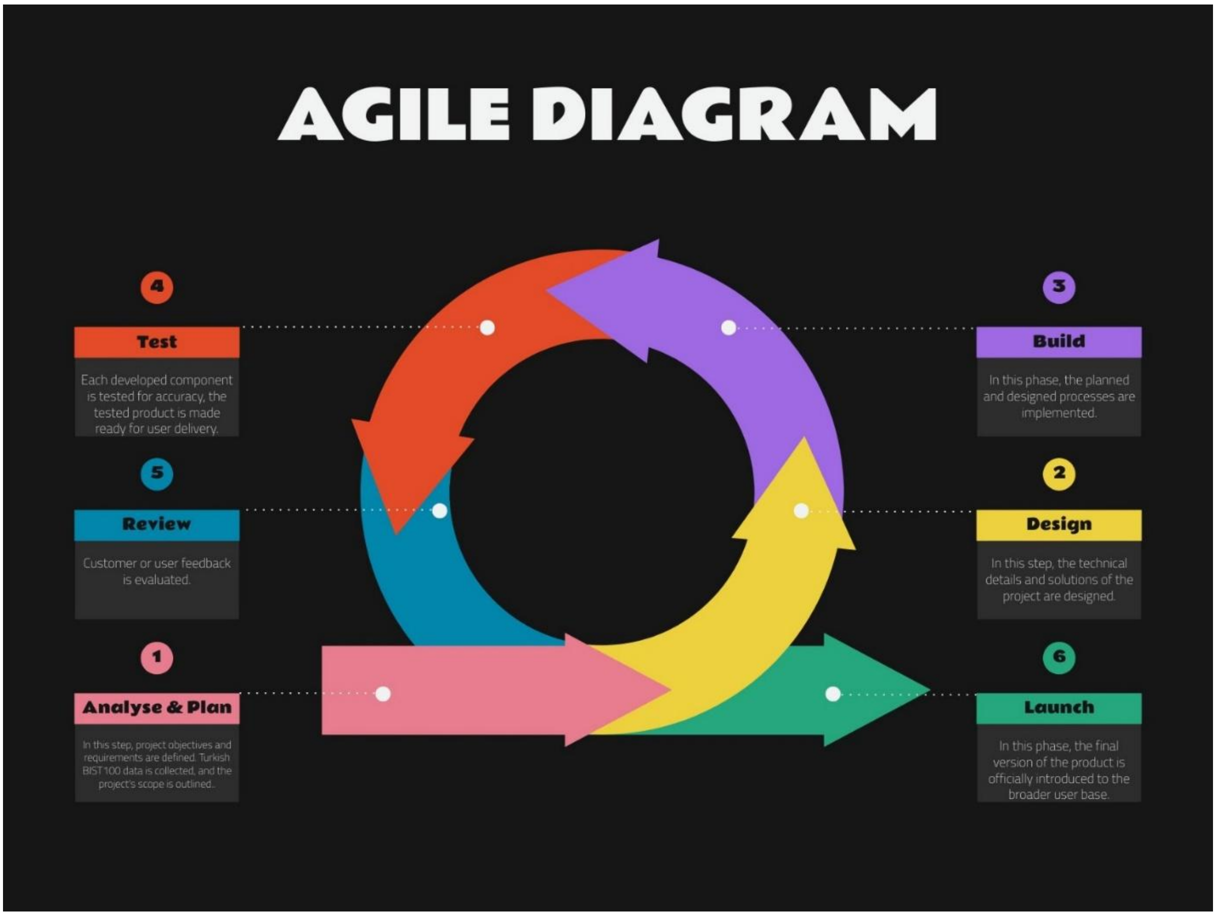


Figure 2. Agile Diagram

● In the **Design** phase, decisions are made regarding which technologies and methods will be used for the project. Since a Turkish NLP model will be created at this stage, care is taken to select methods that are effective for Turkish, and answers are sought to the question of how these methods and the literature can be contributed to.

● In the **Build** phase, the project is implemented by following the previous two phases as closely as possible. This phase includes the NLP model, UI, backend, and all other remaining project components.

● In the **Test & Deploy** phase, all components of the final version of the project are tested, and it is prepared for release.

● In the **Review** phase, the final version of the project is shared with stakeholders and optionally with potential users, and feedback is collected. Stakeholders make new decisions regarding the project's shortcomings and additional requests, marking the end of the cycle for the next iteration. After this step, the project can either be published or move into a new agile cycle, with all steps repeated.

● In the **Launch** phase, the final version of the completed project is published. After the launch, maintenance and updates are carried out regularly.

4.1.3.3. User Characteristics

Participants

● Participants must have a professional or academic interest in financial markets or Natural Language Processing (NLP).

● Participants must be proficient in Turkish language, as the analysed data and simulation outputs will be in Turkish.

● Participants must have a basic understanding of sentiment analysis concepts and methodologies.

● Participants must be familiar with the Borsa Istanbul (BIST100) or financial data interpretation.

4.1.4. Requirements Specification

4.1.4.1. External Interface Requirements

a. User interfaces

The user interface will be available for:

• Android devices with Android 7.0 (Nougat) or higher. [29]

• iOS devices with iOS 12.0 or later. [30]

The interface will be simple and lightweight, designed primarily for querying sentiment analysis results regarding BIST100 companies.

b. Hardware interfaces

There are no specific hardware interface requirements. The application can run on any modern smartphone that meets the minimum OS requirements.

Minimum hardware specifications:

- Android: At least 1 GB RAM and 50 MB of free storage. [29]

- iOS: At least 1 GB RAM and 50 MB of free storage. [30]

Network connectivity is required to access backend services.

c. Software interfaces

The application does not depend on any external software interfaces or libraries installed on the user's device.

d. Communications interfaces

The application requires an active internet connection to communicate with backend cloud services.

- Communication will be over secure HTTPS protocols.

- The app will send user queries to the backend API and receive processed sentiment analysis results.

4.1.4.2. Functional Requirements

1. Login/Register Use Case

Use Case:

● Login

● Register

● Login as Admin

● Exit

Diagram:

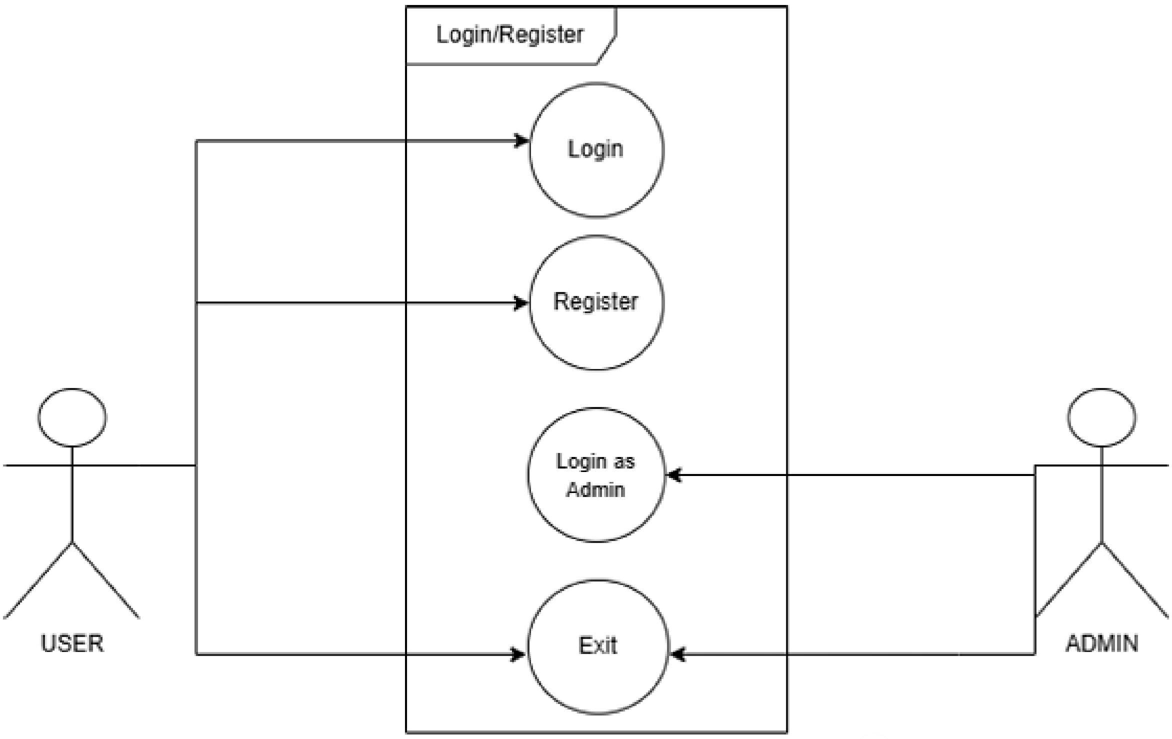


Figure 3. Login/Register Use Case Diagram

Brief Description:

The Login/Register use case diagram illustrates the core functionalities for system entry and exit processes for two types of users: Participants and Admins. Both user roles can exit the system, but their other actions differ:

● Participants can initiate the system without logging in.

● Admins are required to log in using their credentials, specifically a username and password.

Step-by-Step Description:

Participant Access:

● The Participant is allowed to start the system without the need for login credentials.

● They can proceed directly to use functions that do not require authentication.

Admin Login Process:

● The admin must log in using a valid Admin username and password.

● If the provided password is invalid for the given username, the system will prompt the admin to re-enter their credentials until successful login.

Exiting the System:

● Both Participants and Admins have the option to exit the system at any point by selecting the "Exit" function.

2. *Admin Use Case*

Use Case:

● Update data

● Change data frequency

● Change data

● Change NLP method

Diagram:

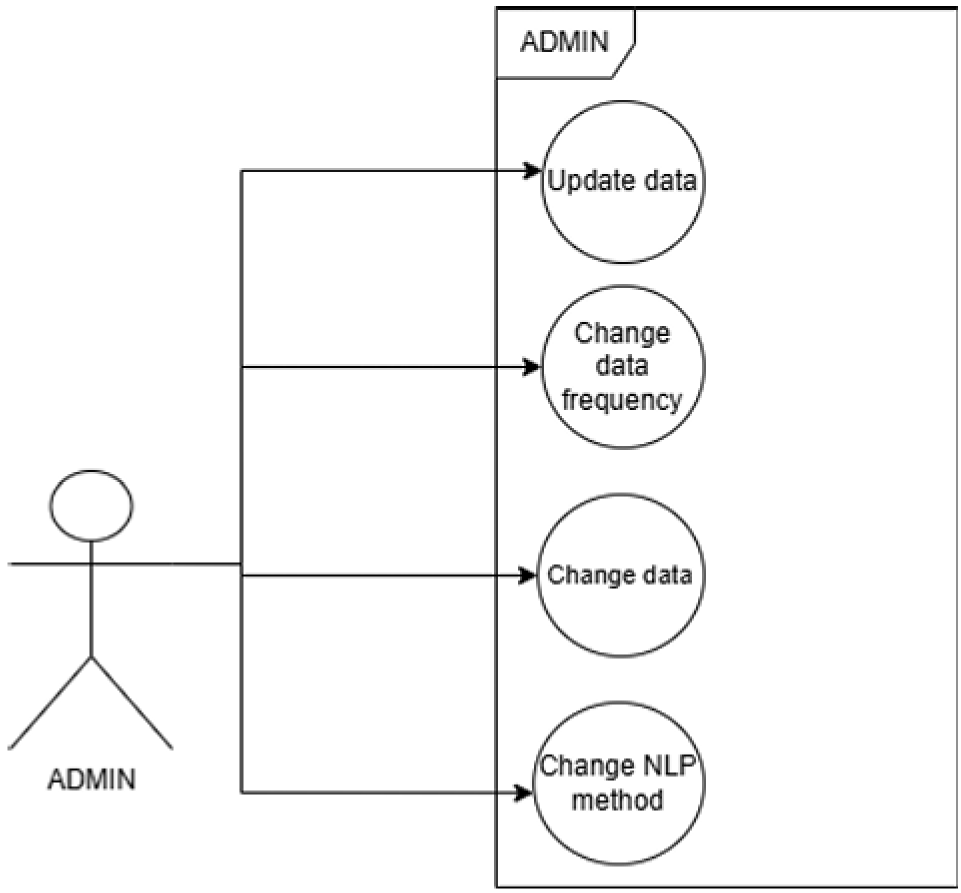


Figure 4. Admin Use Case Diagram

Brief Description:

The use case diagram illustrates the functionalities available to an Admin in the system, focusing on data management and system configuration. The admin role is pivotal, as it oversees modifying data sources, updating processing methods, and adjusting system settings for optimal performance.

Step-by-Step Description:

Admin Login Requirement:

● Only Admins can access these functionalities. They must log in to ensure secure operations.

● The system verifies the admin's credentials before granting access.

Update Data:

● Admins can trigger updates to the data repository.

● This functionality ensures the most recent data (e.g., tweets or financial texts) is available for analysis.

Change Data Frequency:

● Admins can modify how often the system fetches and processes new data.

● This could involve scheduling updates daily, weekly, or in real time, depending on user requirements.

Change Data:

● This allows Admins to replace or alter data sources.

● For instance, changing the data provider or transitioning from historical data to live data streams.

Change NLP Method:

● Admins can switch or upgrade the Natural Language Processing (NLP) technique used in sentiment analysis.

● For example, transitioning from a pre-trained model to a custom-built model tailored for Turkish financial texts

3. Main Page Use Case

Use Case:

● View BIST100 Analysis

● Compare analysis of 2 BIST100 companies

● Add BIST100 companies to favourites

● Display details of stock market shares

Diagram:

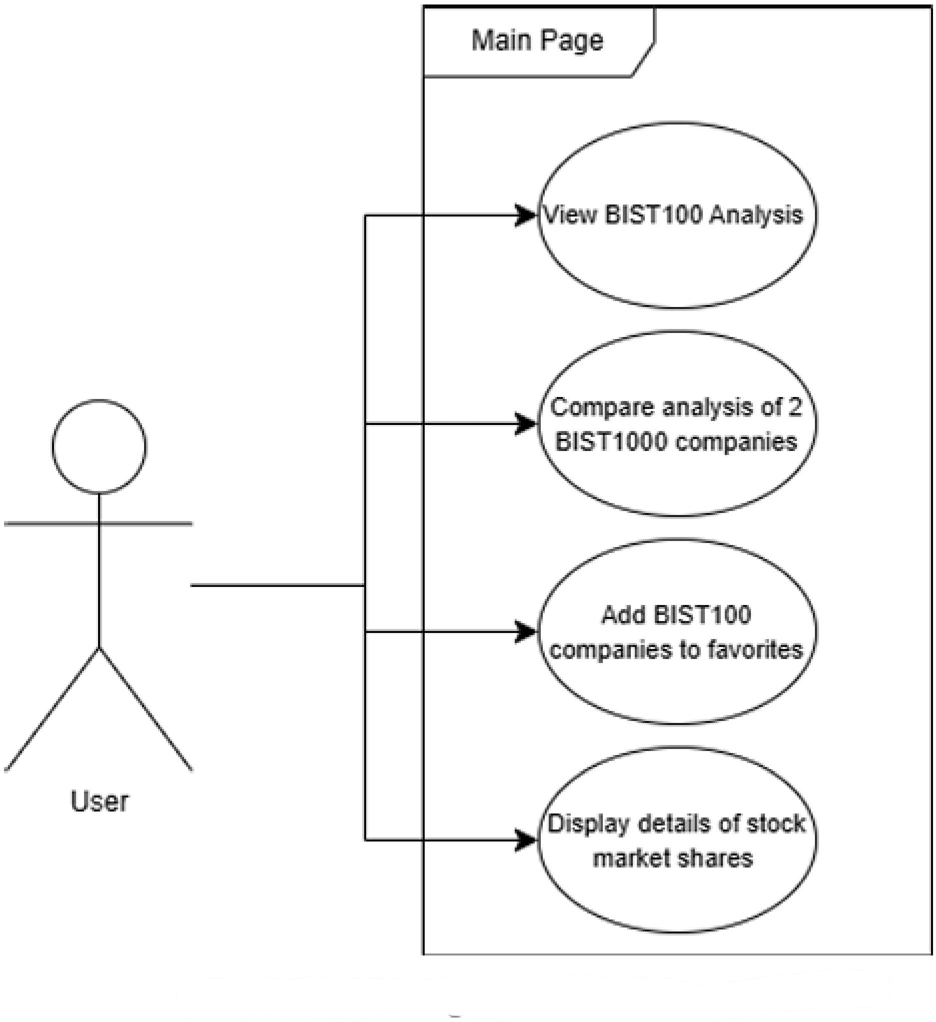


Figure 5. Main Page Use Case Diagram

Brief Description:

The Main Page use case diagram illustrates the core functionalities for system options of BIST100 companies.

Step-by-Step Description:

View Companies Analysis:

● The feature that welcomes users.

● Thanks to that feature users can view all analyses of BIST100 companies and their share price prediction without any effort after the sign in.

● Share price prediction is displayed according to the user's term choice, short - medium - long.

Comparing 2 stock market shares:

● Users can compare shares' prices and analysis via that feature

● Also help users understand the difference between shares through graphics etc.

Add any of the BIST100 company to favourites:

● Users can add BIST100 companies they are interested in to their favourites and access them quickly.

Display of a BIST100 stock share:

● Users redirect to the details page that include news, short - medium - long term analysis etc. about the share.

4. Notification Use Case

Use Case:

● Set Notification Preferences

● Receive Notification Preferences

● Sent Notification

● Read Notification

Diagram:

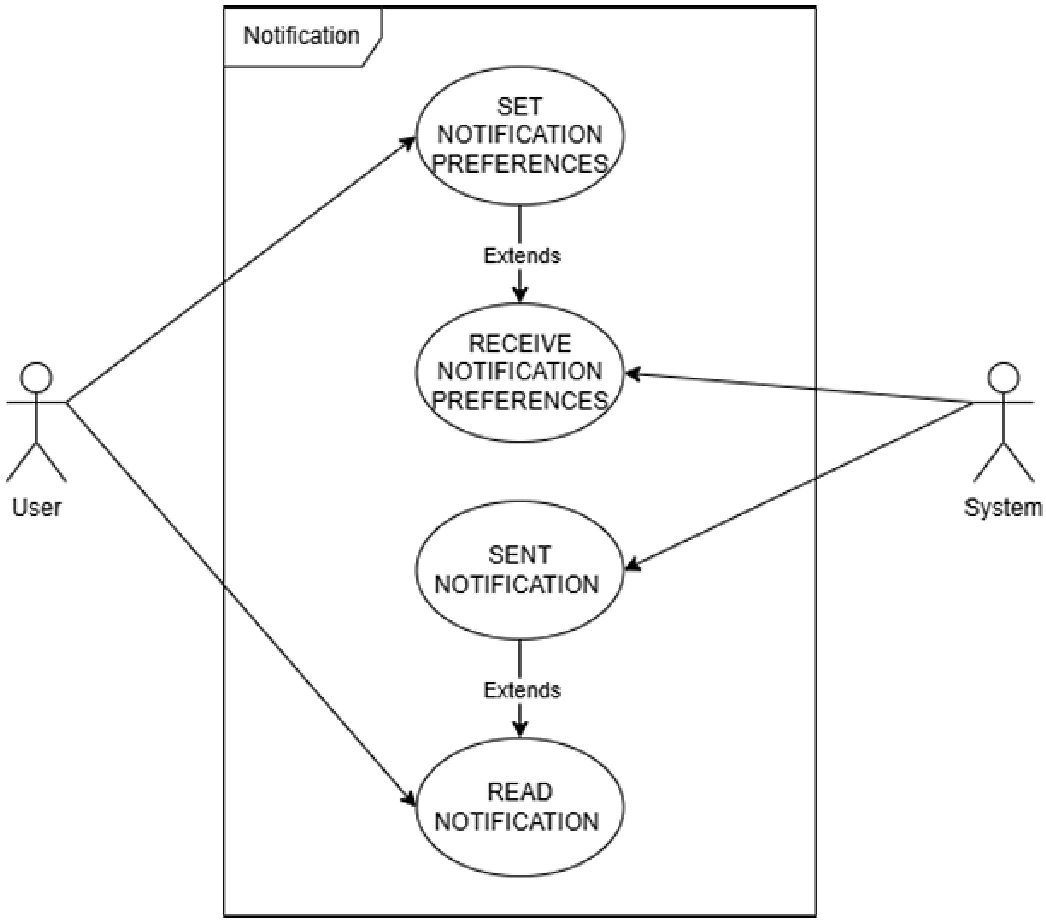


Figure 6. Notification Use Case Diagram

Brief Description:

The Notification System use case diagram illustrates the main methods of sending notifications to the user, through the user's notification preferences.

Step-by-Step Description:

Set Notification Preferences:

● The feature that users can select notification preference.

● Users can select notification content and shipping type.

● Users can change preferences and turn off notifications any time.

Receive Notification Preferences:

● System can access each user's notification preference data.

● System receives each user’s selected notification content and shipping type.

Sent Notification:

● The system groups notifications according to their types and each user is sent notifications from the types they have selected.

● The system sends user-selected notifications using the user-selected notification shipping type.

Read Notification:

● Users can read the notifications via the sending method the user prefers.

5. Display Details of Stock Market Shares Use Case

Use Case:

● Market Statistics

● Performance History

● Real Time Stock Details

● View News and Analysis about Stock

Diagram:

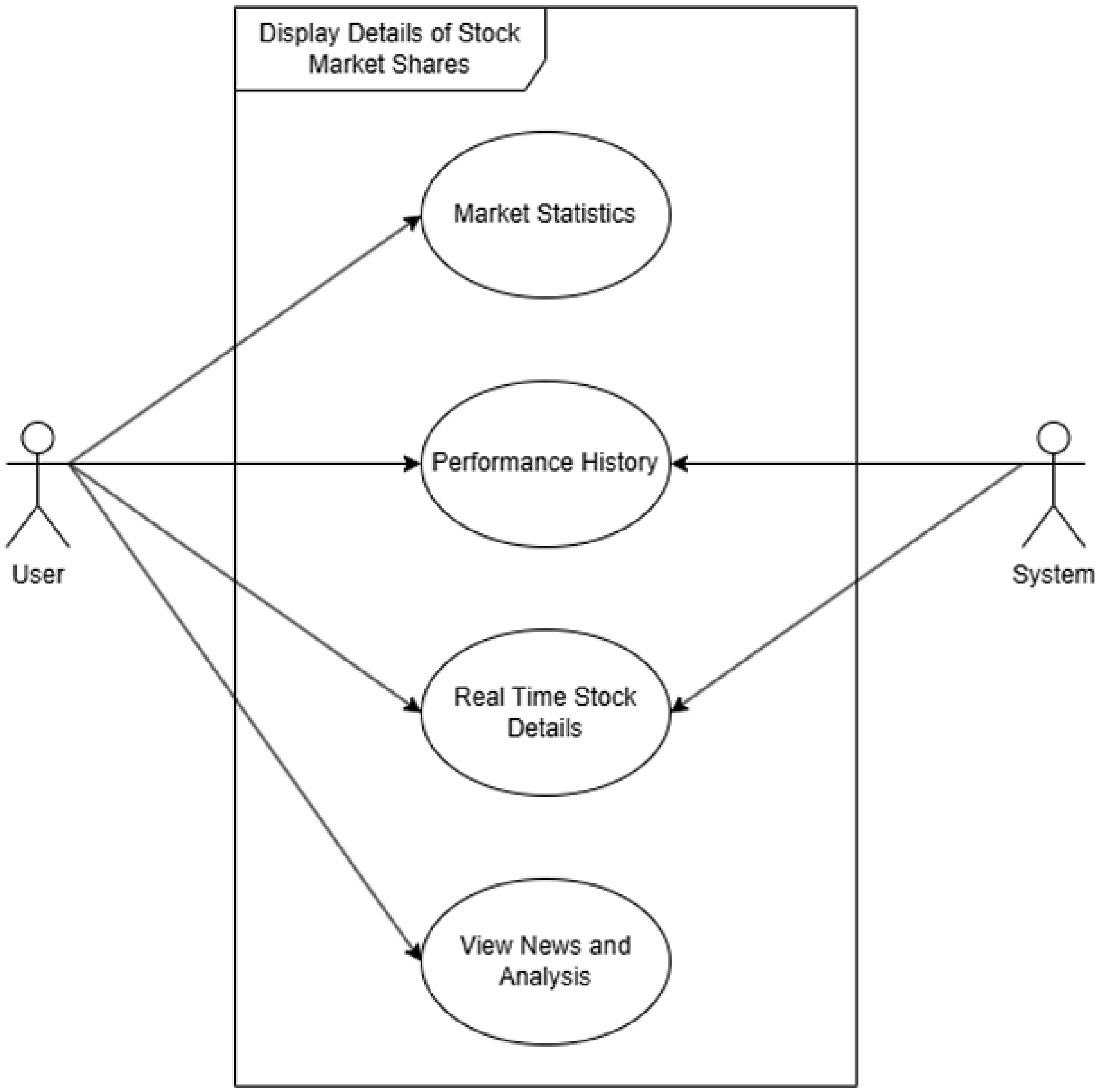


Figure 7. Display Details of Stock Market Shares Use Case Diagram

Brief Description:

The Display Details of Stock Market Shares use case diagram illustrates the ability of users to view variety of details about stock market shares for companies listed on BIST100.

Step-by-Step Description:

View Statistics:

● The user select company or a sector from BIST100.

● The user can see market capitalization, trading volume and other metrics to evaluate the current financial situation of the company or sector.

Performance History:

● The system displays trends in certain time periods decided by the user.

● Users can select a certain time period and access historical stock price data.

Real Time Stock Details:

● The system gives live updates about stock price and other metrics.

● Users can view real time data in stock performance and use it to make decisions.

View News and Analysis about Stock:

● The user can see news and read analysis about the stock or sector they choose.

● This feature allows users to keep up with new updates in stocks and sectors.

4.1.4.3. Performance Requirements

Since the analysis model runs on one computer, and the results will reach the user by application, it does not require good performance hardware. An Android or iOS device such as today's world devices will be good enough, more details on the below.

4.1.4.4. Software system attributes

1. ***Portability***

• The mobile app is designed to run on both Android 7.0+ and iOS 12.0+.

• The results which pre-trained model produces reach through backend, cloud platform. It's independent.

2. ***Performance***

• The model API endpoints must provide an average response time of <1 second under load.
• The application must be designed with efficient, optimized algorithms to keep fluency.

3. ***Usability***

• Company-specific sentiment analysis.

• Error messages (e.g., "Network Unavailable") must provide actionable guidance to the user.

4. ***Adaptability***

• The system is designed to integrate new features, such as expanded sentiment categories or support for additional languages (e.g., English), without major architectural changes.

5. ***Scalability***

• Backend services must dynamically scale to support 5x the current user load during market events.

6. ***Security***

• All data exchanged between the mobile app and backend must be encrypted using TLS 1.2 or higher.

• Sentiment analysis results and user queries must be anonymized and stored in compliance with Turkish data protection laws (KVKK).

4.1.4.5. Safety Requirements

1. Data Integrity

• Tweets fetched for analysis must not be altered during processing to ensure sentiment results are trustworthy.

2. User Safety

• Users should be warned against over-reliance on sentiment trends, emphasizing that the tool complements, not replaces, professional financial analysis.

3. System Reliability

• Backup servers must replicate critical data every hour to minimize loss in case of failure.

4.Ethical Considerations

• The system must prevent misuse of sentiment analysis (e.g., for manipulating stock markets) by flagging abnormal patterns in social media data.

5. SOFTWARE DESIGN DESCRIPTION

5.1. Introduction

5.1.1. Purpose

The purpose of the e-TurFinSAS project is to create an entity-based Financial Sentiment Analysis system with Turkish texts from X, TradingView, KAP and Investing related to BIST100. System analyses Turkish content, and classify sentiment as positive, neutral and negative regarding BIST100 companies. The meaning of this project for users are assisting them in making better decisions about their investments, enhancing prediction of financial trends and stock market dynamics. The goal of this Software Design Description (SDD) is to provide technical guidance for the development of the project. This document outlines the architectural and design details of the system. The document acts as reference for future development, ensuring alignment with the project's purpose and requirements, with the respect to information in the document.

5.1.2. Scope

This Software Design Description outlines the design and architecture of the "e-TurFinSAS: Entity-Based Turkish Financial Sentiment Analysis System." The document provides a detailed description of the system's objectives, functional and nonfunctional requirements, design decisions, and architectural components. It aims to serve as a comprehensive reference for stakeholders, including developers, project managers, and future maintainers, ensuring alignment on the technical aspects of the system.

The scope of the document includes:

- **System Overview:** Define the purpose of the system, which is to perform entity-based sentiment analysis on Turkish financial texts, particularly social media data, to classify sentiment as positive, negative, or neutral.

- **Functional Requirements:** Details the expected behavior of the system, including the collection, processing, and analysis of social media data for sentiment classification.

- **Design Considerations:** Discusses challenges such as the morphological complexity of Turkish, data preprocessing, and model training for the financial domain.

- **Design Approach:** Describe the high-level architecture of the system, including backend data collection, natural language processing (NLP) pipelines, and model integration for sentiment analysis. Focus on modularity and scalability to ensure adaptability to future requirements.

- **System Architecture:** Describes the software components, their interactions, and how they integrate to meet project goals.

- **Technical Constraints:** Identifies limitations such as dataset availability, processing power, and real-time analysis requirements.

- **Evaluation Metrics:** Defines the success criteria, including accuracy, precision, recall, and F1 score for sentiment analysis models.

- **Risk Assessment:** Addresses potential risks and mitigation strategies related to data privacy, ethical considerations, and model performance.

This document provides a foundation for developing a robust and efficient system that meets the technical and functional requirements of the project. It serves as a reference for the development team and ensures alignment across all phases of the project lifecycle.

5.1.3. Glossary

Glossary Table of SDD

Term	**Definition**
Accuracy	Measure of the frequency of accurate sentiment analysis predictions.
Activity Diagram	Graphical representation of workflows and processes within the system.
Admin Page	An interface for the system that lets administrators change configurations, data, and methods.
API (Application Programming Interface)	Set of protocols for interacting with software applications or data sources.

BIST100	The top 100 companies listed on Borsa Istanbul, a major Turkish stock market index.
Class Diagram	A diagram illustrating the structure and relationships of classes in the system architecture.
Comparison Feature	Function that enables users to compare stock performance of different companies.
Dataset	Structured collection of the data used for training and testing sentiment analysis models.
e-TurFinSAS	Entity-Based Turkish Financial Sentiment Analysis System designed for BIST100 companies.
Entity-Based Sentiment Analysis	Analysing sentiment specific to entities such as companies or stocks within financial contexts.
F1 Score	The harmonic mean of precision and recall, used as an evaluation metric for sentiment analysis.
Firebase	A backend-as-a-service platform used for database and application management.
Investing	Financial website that provides market data, used as a data source in this project.
KAP	Public Disclosure Platform (Kamuyu Aydınlatma Platformu), a source for Turkish financial data.
Main Page	The central interface for users to access system functionalities like analysis and comparisons.
NLP (Natural Language Processing)	The field of AI focused on the interaction between computers and human language.
Precision	A metric to evaluate the relevance of positive sentiment predictions made by the system.
Preprocessing	Steps taken to clean and prepare raw data for analysis, such as tokenization or normalization.
Recall	A metric to assess the system's ability to identify all relevant sentiment instances.
SDD	Software Design Document.
Sentiment Analysis	Process to determine the emotional tone (positive, negative, or neutral) within text data.
TensorFlow	Open-source machine learning framework used for model development in this project.
TradingView	Platform that provides stock market analysis and data, utilized in this project.
X (formerly Twitter)	Social media platform used as a data source for gathering public sentiment about companies.

5.1.4. Motivation

We are senior Computer Engineering students who are interested in Data Analytics, NLP (Natural language processing) and Finance. To further develop our ideas on these topics, we are taking the NLP course and conducting research and studies in this field. Our goal is to contribute to the Turkish Stock Exchange and Turkish NLP technologies by combining our knowledge in these fields and providing accessible and useful analysis to users and companies. 8 In order to achieve our goal, we have chosen familiar platforms and tools such as Python, TensorFlow and Firebase. In finance, we have taken and continue to take Turkish data from reliable and popular sources such as KAP, X (formerly Twitter) and Investing, taking care to clean the data. In the sentiment analysis part, we are trying to carefully research and develop up-to-date methods for Turkish, which is a complex language. This project combines NLP technologies and the field of finance, while making a significant contribution to the literature for the Turkish language and the Turkish stock market.

5.1.5. Overview of document

Here are the remaining chapters and what they include: Section 2 is about Architectural Design. It explains how the project was developed. This part also has the class diagram of the system and the architecture design for the simulation. It talks about things like actors, exceptions, basic steps, priorities, and conditions before and after. There’s also an activity diagram for the scenario generator in this section. Section 3 is Use Case Realization. It shows and explains a block diagram of the system. This diagram is made based on the use cases in the SRS document. Section 4 is about the Environment. Here, we show example frames of the environment from the prototype and describe the scenario.

5.2. Architecture Design

5.2.1. Simulation Design Approach

We decided to use the agile methodology while doing our project. Agile methodology is a flexible and collaborative approach used in software development and project management. In short, agility is a paradigm shift in the usual way of working. The word agile has passed into Turkish as "çeviklik". The concept of agile, which is the ability to adapt quickly to changing conditions; Although it is defined as method, methodology, method or project management in most sources, it is a way of thinking contrary to these. At the core of this methodology are the principles that individuals and interactions are more valuable than processes, a working product is more important than comprehensive documentation, customer collaboration is more important than contract negotiations and adapting to change is more valuable than following a plan. Agile aims to deliver projects faster, increase customer satisfaction, and adapt quickly to changing requirements through small, manageable work cycles (e.g., sprints). Although it was initially developed for software development, today it is widely used in different sectors such as marketing, finance and manufacturing. Adopting a continuous improvement and value-oriented approach, Agile enables teams to work more efficiently and effectively.

5.2.1.1. Class Diagram

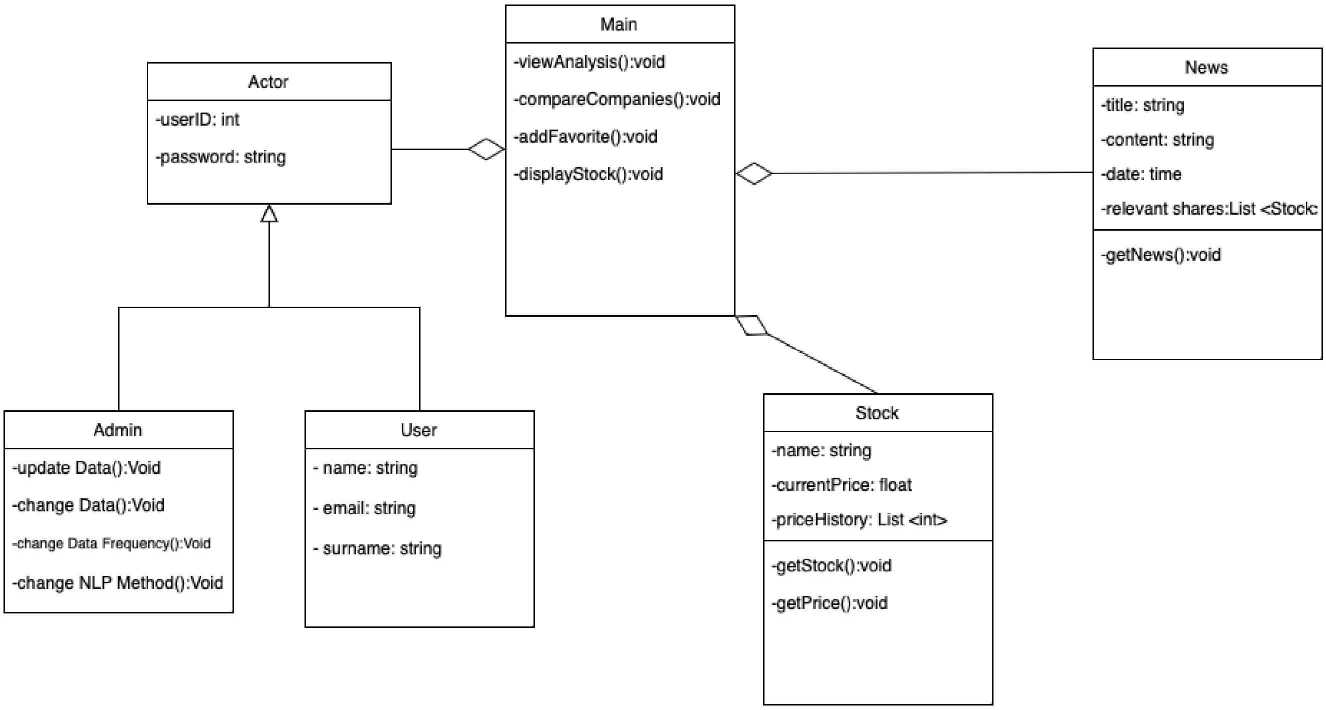


Figure 8. Class Diagram of e-TurFinSAS

The class diagram illustrates the structure and relationships of the e-TurFinSAS system components. The Main Class serves as the central system, managing the connections and interactions between the various components, such as Actor, Stock, and News. It facilitates the flow of data and ensures seamless operation across the system. The Actor Class represents all individuals interacting with the system. The Admin Class, a specialized Actor, is responsible for managing and maintaining the system, ensuring security, and overseeing user activities. The User Class represents regular users who can register, log in, and request sentiment analysis for stocks. Users can also view historical data and trends. Figure 2. Class Diagram of e-TurFinSAS 12

The Stock Class contains information about the stocks being analysed, including their names, symbols, and relevant metadata. This class interacts with the News Class for retrieving stock-specific content for sentiment analysis. The News Class represents text data sources like articles, tweets, or reports used for analysis. This class provides the input data needed for sentiment labelling and prediction in the backend.

Relationships between classes include the Main Class connecting Actors with Stock and News classes, ensuring that users and admins can interact with the system effectively. Admins manage Stocks and News, while Users request sentiment analysis and view results derived from these components. This structure ensures that all components work together cohesively, allowing users to seamlessly interact with the system while admins oversee operations and maintain its reliability

5.2.2. Architecture Design of Simulation

5.2.2.1. Register Page

Summary: The system adds to new users into DB. Users need to input their email, password, name-surname and phone number information to register successfully.

Actor: User

Precondition: User must not already have an account.

Basic Sequence:

1. User opens the application and navigates to the Register page.
2. User enters their email, password, and other required details.
3. User clicks the "Register" button.
4. System validates the input and creates a new account if all information is valid.

Exception: If any input data is invalid program shows a relevant error message.

Post Condition: A new account is created; the user is redirected to the login page.

Priority: High

Activity Diagram:

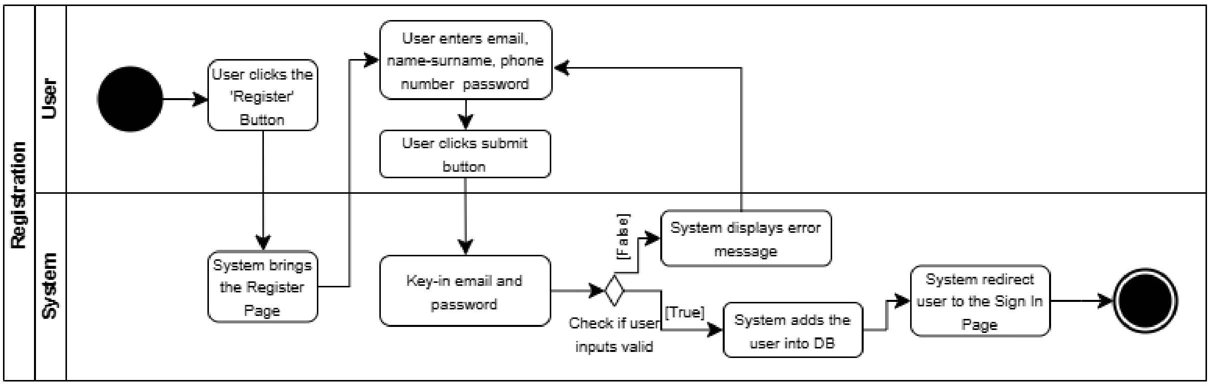


Figure 9. Activity Diagram of Register Page

5.2.2.2. Login Page

Summary: The system allows users or admin to enter their email and password to sign in the system.

Actor: User

Precondition: User must have an account.

Basic Sequence:

1. User opens the application and navigates to the Login page.

2. User enters their email and password.

3. User clicks the "Login" button.

4. System verifies the credentials and grants access if valid.

Exception: If any input data is invalid program shows a relevant error message.

Post Condition: Users log in and are redirected to the main page.

Priority: High

Activity Diagram:

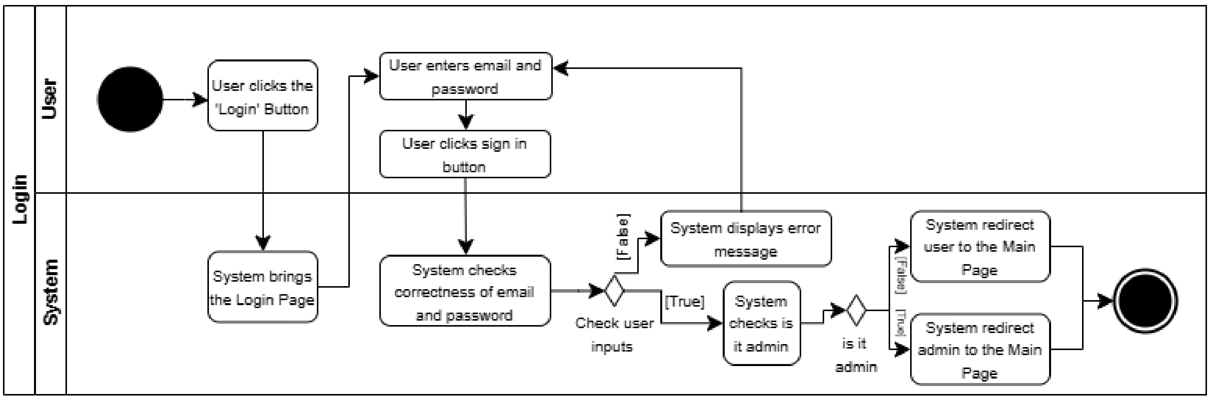


Figure 10. Activity Diagram of Login Page

5.2.2.3. Admin Page

Summary: The system is used by admin. Admin can perform operations such as update data, change data frequency, change data, change NLP method from this page.

Actor: Admin

Precondition: Administrator must run the program and log in as admin.

Basic Sequence:

1. Admin click to enter the Admin Page with the admin account.

2. After logging in, the admin selects one of the options: update data, change data frequency, change data, change NLP method.

3. After the admin makes the desired changes, admin approves the changes.

4. The system saves the changes made.

Exception: Problems may arise with changes made.

Post Condition: Changes made by admin will be saved.

Priority: Medium

Activity Diagram:

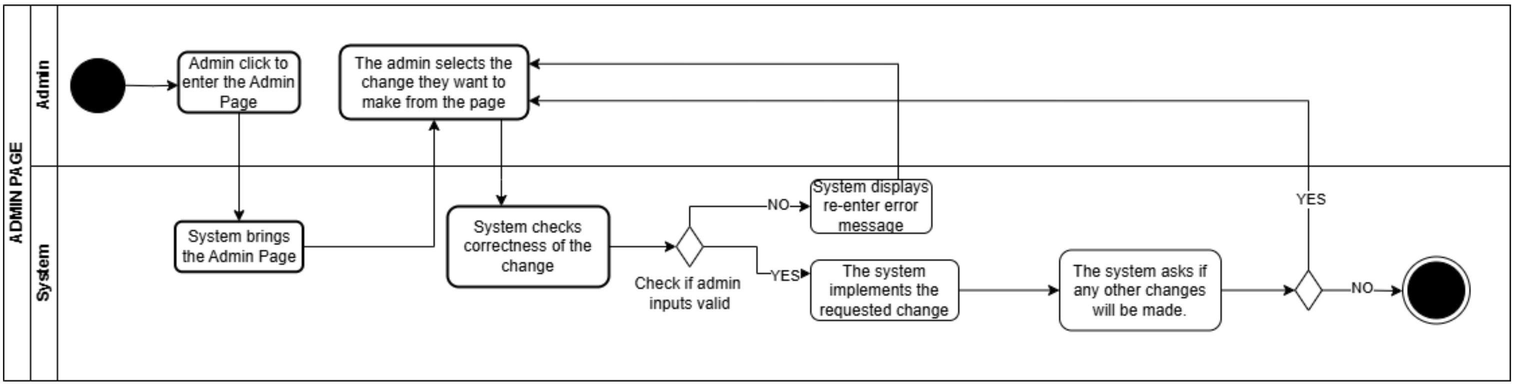


Figure 11. Activity Diagram of Admin Page

5.2.2.4. Compare Companies

Summary: This system is used by user and admin. User can compare stock of companies however he wants. Admin can change some features here such as supporting graphics.

Actor: User, Admin

Precondition: User must click the compare button to open the page.

Basic Sequence:

1. User must be on the Main Page.

2. User must click the button Compare.

3. User must choose the companies which he wants to compare their stocks.

4. Admin can do changes on the model, the data period which model takes as input.

5. User can exit from the system by selecting exit button.

Exception: Database connection, connection to the model can be failed due to API, internet or some backend failures.

Post Condition: None

Priority: Low

Activity Diagram:

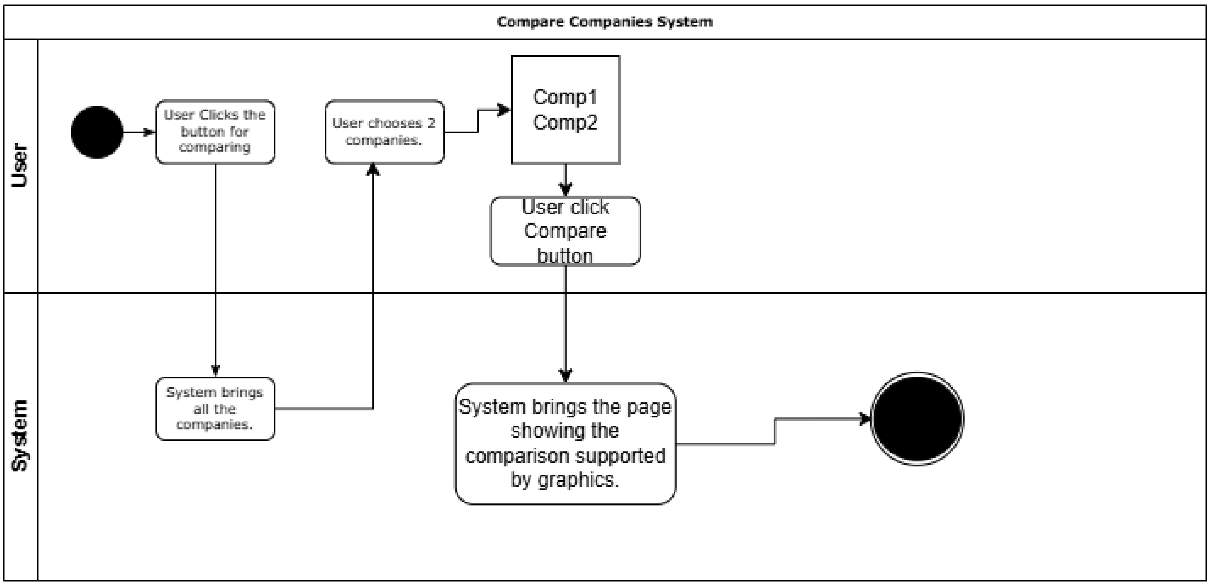


Figure 12. Activity Diagram of Compare Companies System

5.2.2.5. Main Page

Summary: Main functionalities of the system can be seen and accessed in this page. Users can view BIST100 Analysis, compare the analysis of two companies, add these companies to favourites and see the details of stock market shares.

Actors: User, Admin.

Precondition: User must login to the program.

Basic Sequence:

1. User must be logged on to the program in order to access the main page.
2. After login, system shows the choices the user can make.
3. User can see various functionalities and choose to view BIST100 Analysis, compare the analysis of two companies, add the desired companies to favourites or see the details of stock market shares.
4. After the choice of the user, the system directs the user to that page.
5. After the user completes his task, he can turn back to the main page.
6. User can choose another functionality or logout and close the program.

Exception: Network Problems and user session timeout.

Post Condition: None

Priority: High

Activity Diagram:

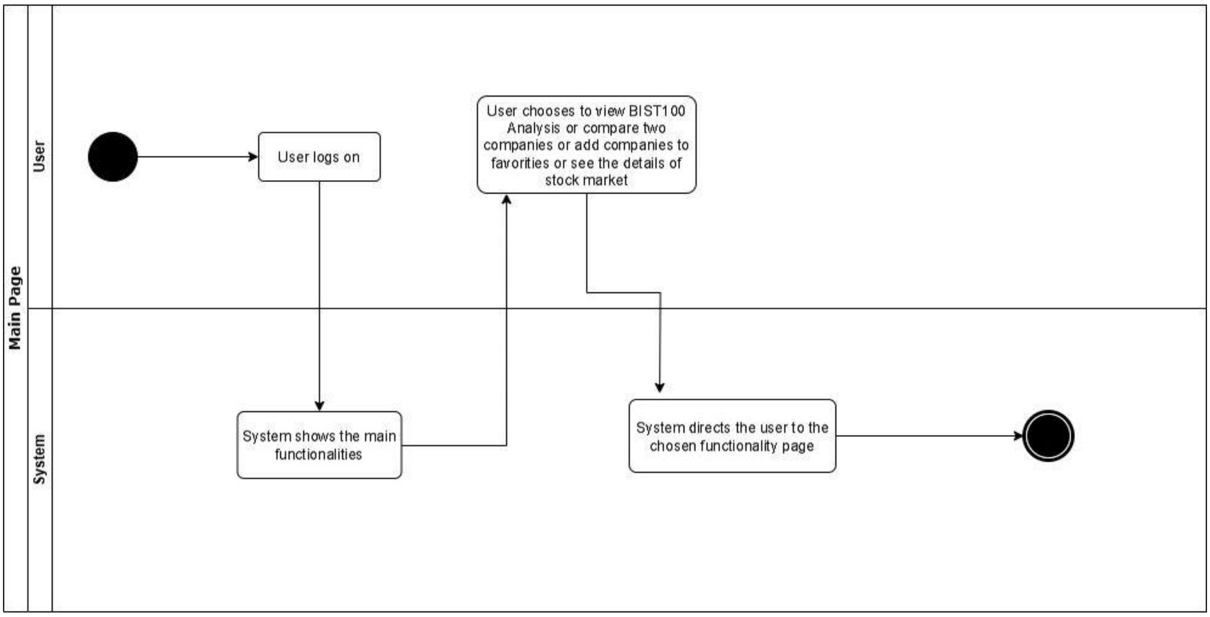


Figure 13. Activity Diagram of Main Page

5.3. Use Case Realizations

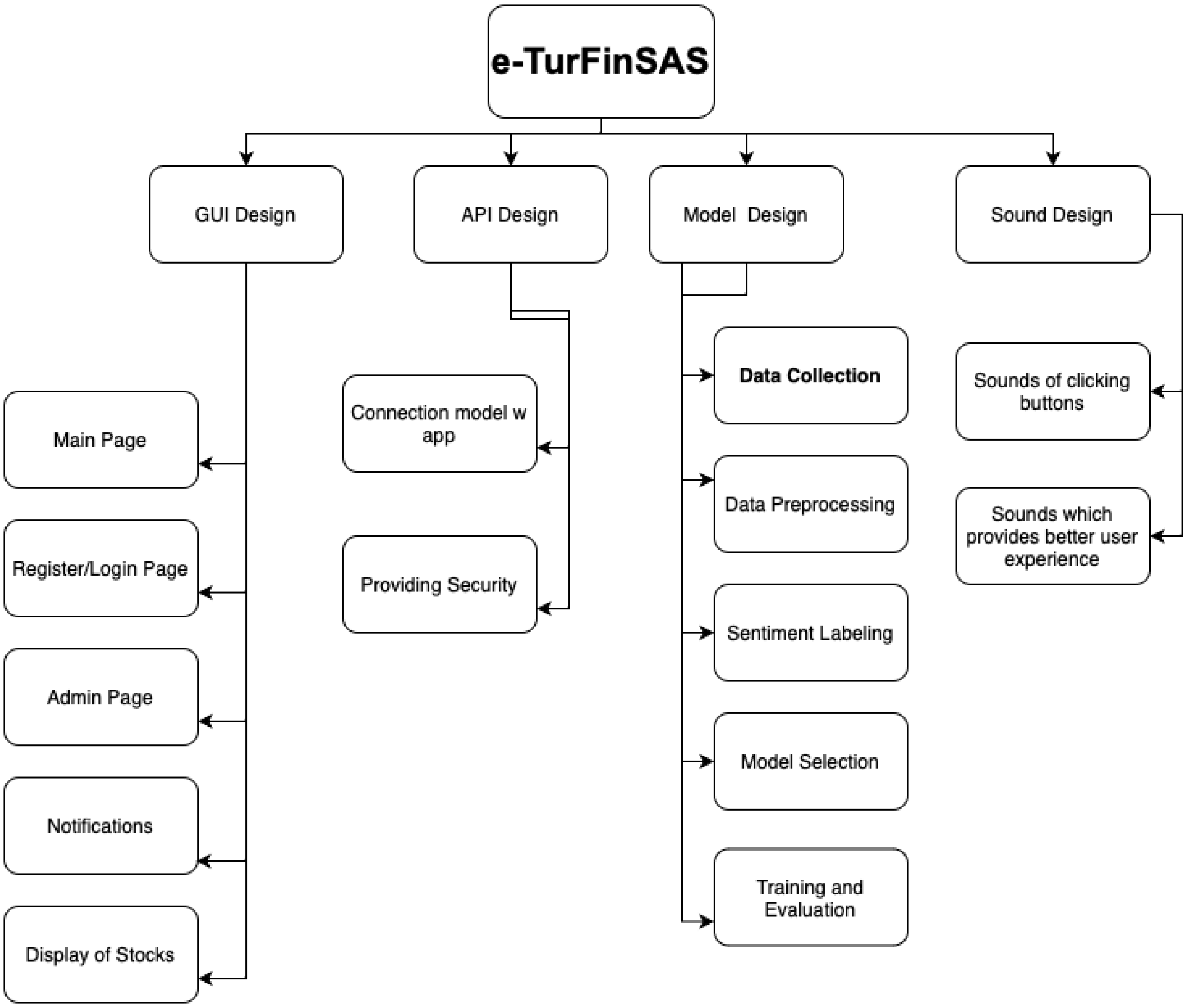


Figure 14. Project Components of e-TurFinSAS

5.3.1. Brief Description of the Project Components

5.3.1.1. GUI Design

GUI design is responsible for interaction between the actors and the system. There are 5 sub-systems in this design which are Main, Register/Login, Admin pages, Notification and display of stocks. Main Page is the actor can reach all activities s/he wants. Register/Login is where s/he can sign up or sign in. Admin Page is where the admin, which is determined by us, can do some important changes on the model, application and more. Notifications are for letting the user know about stocks, or application. Display of Stocks is the page where the user can view the analysis of the model of stocks which s/he wants.

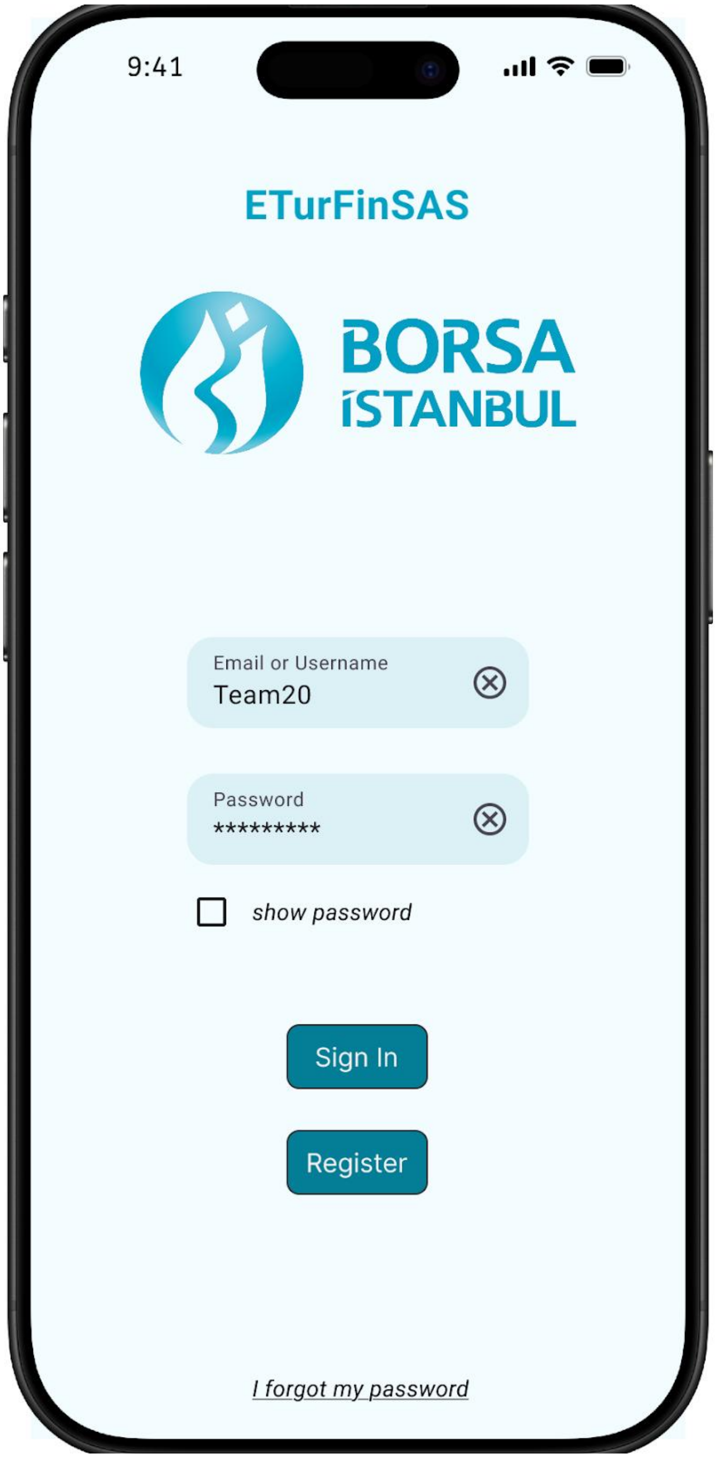


Figure 15. Prototype UI Design of Login Page

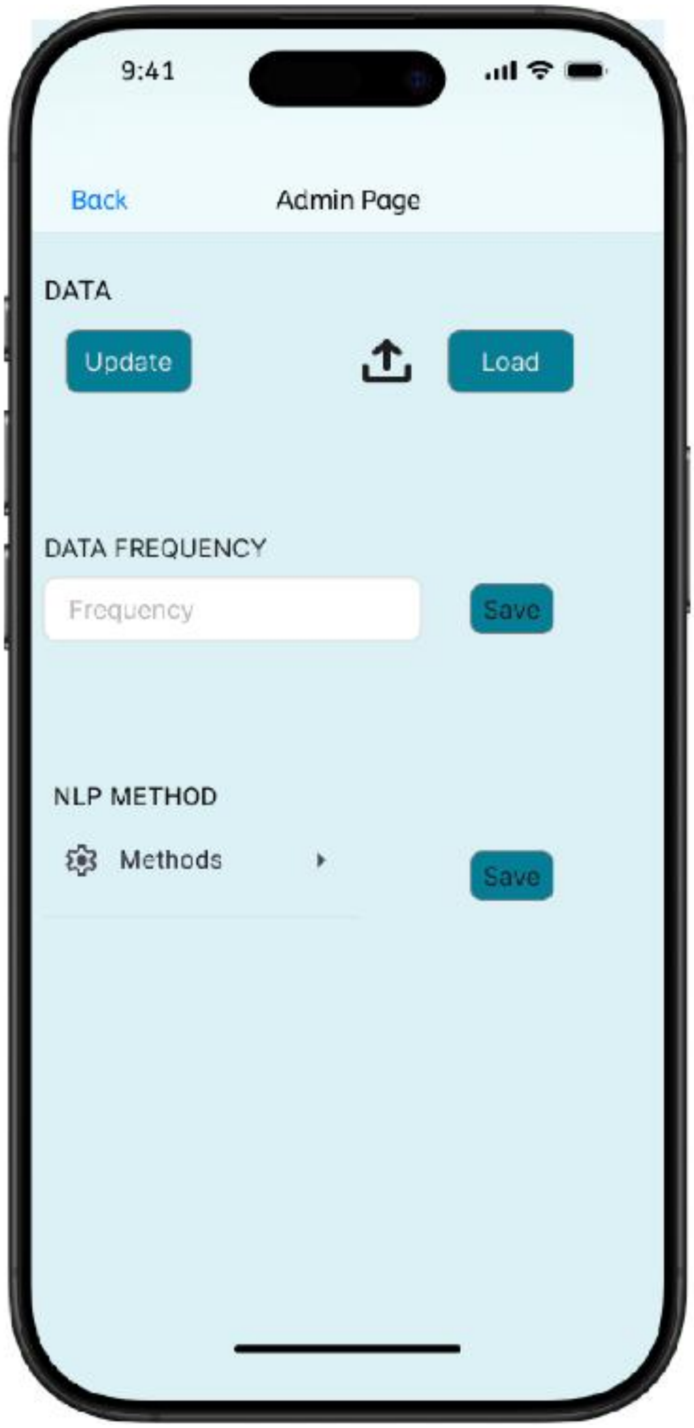


Figure 16. Prototype UI Design of Admin Page

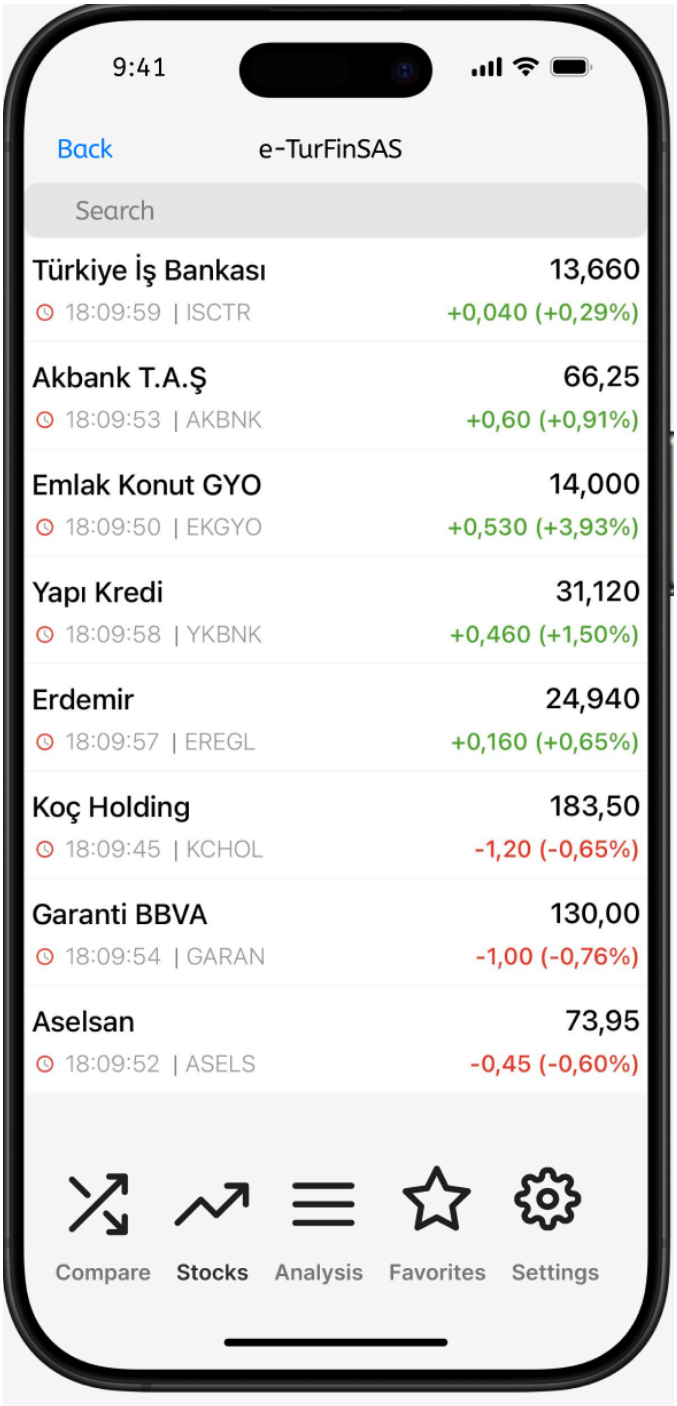


Figure 17. Prototype UI Design of Main Page

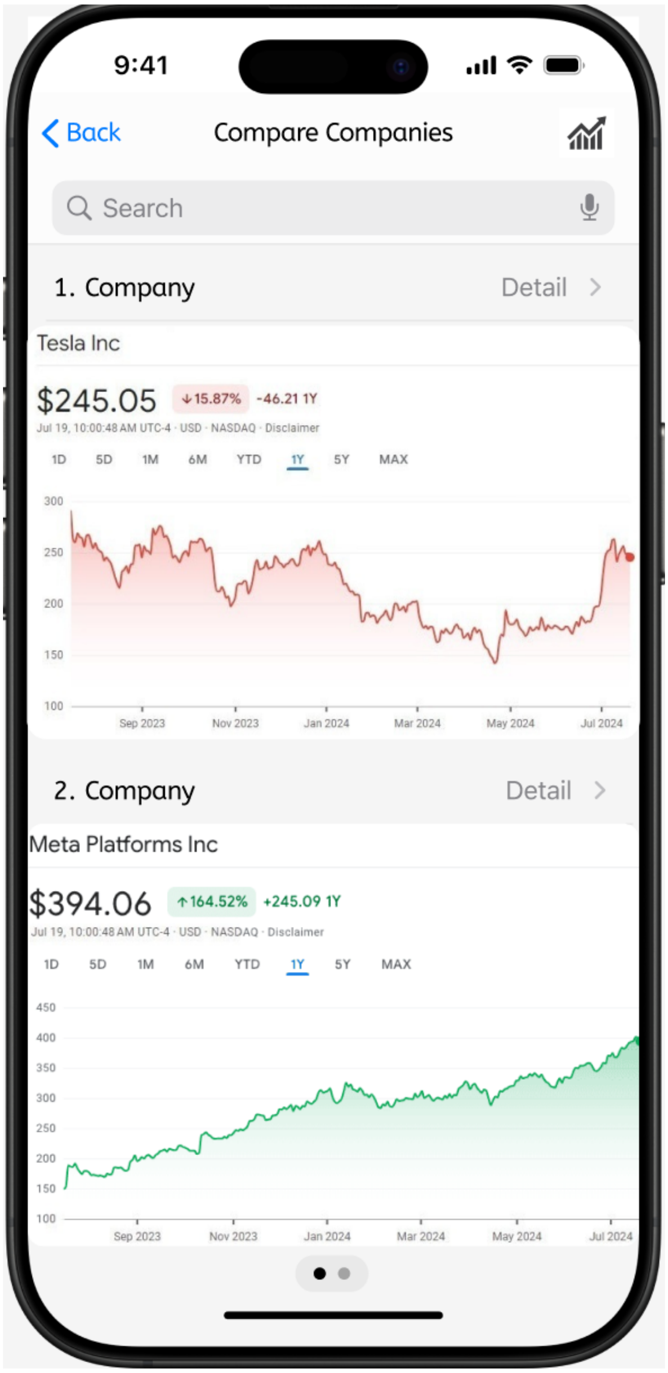


Figure 18. Prototype UI Design of Compare Companies

5.3.1.2. API Design

The API Design module is responsible for facilitating communication between the application and the backend servers where the sentiment analysis is performed. This system ensures efficient data transfer and accurate responses without burdening the user's device.

- **Request Module:** Accepts user input, such as the stock name and related text data, and forwards it to the backend for analysis.
- **Response Module:** Retrieves sentiment analysis results from the server, including sentiment classification (positive, negative, or neutral) and confidence scores, and delivers them to the application.

5.3.1.3. Model Design

The Model Design module is responsible for processing text data and performing sentiment analysis using advanced machine learning models. The system operates entirely on the backend, leveraging pre-trained transformer-based models for high accuracy.

5.3.1.4. Sound Design

The Sound Design module is responsible for enhancing the user experience through audio feedback, providing cues for various system interactions and processes.

6. CONCLUSION

In this project, we have planned the way we will develop our project to classify sentiments about BIST100 companies as positive, negative, or neutral, then train the model to try to predict the future stock prices according to current data provided by twitter. We may have a lot of problems such as:

- Handling high-frequency changes in stock prices, particularly during volatile periods.
- Optimizing the system for real-time sentiment analysis while maintaining accuracy.
- Expanding lexicons and datasets to cover less frequently used financial jargon and expressions.
- Handling noise data because of social media to improve accuracy of the model.

In conclusion, this project represents a significant step toward integrating social media sentiment with financial analysis in Turkey, but further refinements are necessary to realize its full potential.

Acknowledgement

We have been supported by our supervisor Mrs. Tokdemir and co-supervisor Mr. Uğuz.

7.REFERENCES

[1] Omar Y. Adwan () , Marwan Al-Tawil, Ammar M. Huneiti, Rawan A. Shahin, Abeer A. Abu Zayed, Razan H. Al-Dibsi, "Twitter Sentiment Analysis Approaches: A Survey", Aug 14 2020, Kassel, Germany, [Online], Available: https://www.learntechlib.org/p/217980/ [Accessed: 09.01.2025]
[2] Mayur Wankhade1,2 · Annavarapu Chandra Sekhara Rao1,2 · Chaitanya Kulkarni1,2, "A survey on sentiment analysis methods, applications, and challenges", 07 February 2022, [Online], Available: https://link.springer.com/article/10.1007/S10462-022-10144-1 [Accessed: 09.01.2025]
[3] Marouane Birjali ∗ , Mohammed Kasri, Abderrahim Beni-Hssane, "A comprehensive survey on sentiment analysis: Approaches, challenges and trends ", 17 August 2021, [Online], Available: https://www.sciencedirect.com/science/article/pii/S095070512100397X [Accessed: 09.01.2025]
[4] Parastoo Golpour 1 , Majid Ghayour-Mobarhan 2,3, Azadeh Saki 1,* , Habibollah Esmaily 4 , Ali Taghipour 4,5, Mohammad Tajfard 4,6, Hamideh Ghazizadeh 2,7, Mohsen Moohebati 3 and Gordon A. Ferns, "Comparison of Support Vector Machine, Naïve Bayes and Logistic Regression for Assessing the Necessity for Coronary Angiography ", 4 September 2020, [Online], Available: https://www.mdpi.com/1660-4601/17/18/6449 [Accessed: 09.01.2025]
[5] Alexander Ligthart1 · Cagatay Catal2 · Bedir Tekinerdogan1, "Systematic reviews in sentiment analysis: a tertiary study", pp. 4997-5053, 2021, https://link.springer.com/article/10.1007/s10462-021-09973-3
[6] Travis Adams, Andrea Ajello, Diego Silva, Francisco Vazquez-Grande, "More than Words: Twitter Chatter and Financial Market Sentiment ", 03 March 2021. [Online]. Available: https://www.federalreserve.gov/econres/feds/files/2023034pap.pdf
[Accessed 9.01.2025]
[7] Gabriele Ranco1 , Darko Aleksovski2 *, Guido Caldarelli1,3,4, Miha Grčar2 , Igor Mozetič2, "The Effects of Twitter Sentiment on Stock Price Returns", September 21, 2015. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0138441
[Accessed 9.01.2025]
[8] Avar, B., & Ceylan, H., "*Morphological Analysis and Applications in Turkish Natural Language Processing"*,2017
[9] Sak, H., Güngör, T., & Saraçlar, M., "*Turkish Language Resources: A Collection of Tools and Resources for Turkish Natural Language Processing*",2008
[10] Öztürk, N., & Ayvaz, S., "*Sentiment Analysis on Twitter: A Text Mining Approach to the Syrian Refugee Crisis in Turkey*", 4(3),pp. 251-272,2008
[11] Avvad, H., & Ereren, E.,"*Sentiment Analysis in Turkish Tweets Using Different Machine Learning Algorithms"*,01 October 2024, [Online]. Available: Artificial Intelligence Theory and Applications » Makale » Sentiment Analysis in Turkish Tweets Using Different Machine Learning Algorithms
[Accessed 9.01.2025]
[12] Eren, İ., & Yıldırım, Ö., "*Domain-specific Word Embeddings for Turkish Financial Texts*", In Proceedings of the 27th Signal Processing and Communications Applications Conference,2019
[13] Şahin, M., & Gülşen, Ö., "*Analysis of Financial News with Word Embeddings for Turkish Language"*, Journal of Language Technology and Computational Linguistics.2021
[14]Zeynep Banu Özger, Banu Diri, "Türkçe Dokümanlar İçin Kural Tabanlı Varlık İsmi Tanıma", 24 June 2016, [Online]. Available:
https://dergipark.org.tr/en/pub/tbbmd/issue/22245/238798
[Accessed 9.01.2025]
[15]Kübra Adalı, A. Cüneyd Tantuğ, "Annotation of Financial Entities Using A Comprehensive Scheme in Turkish". 29 August 2022, [Online]. Available:
https://ieeexplore.ieee.org/abstract/document/9864782
[Accessed 9.01.2025]
[16]Zulkarnain, Tsarina Dwi Putri, "Intelligent transportation systems (ITS): A systematic review using a Natural Language Processing (NLP) approach". [Online]. Available:
https://www.cell.com/action/showPdf?pii=S2405-8440%2821%2902718-3

[Accessed 9.01.2025]
[17] Pierre Lison, Aliaksandr Hubin, Jeremy Barnes, Samia Touileb, "Named Entity Recognition without Labelled Data: A Weak Supervision Approach". [Online]. Available: https://arxiv.org/pdf/2004.14723
[Accessed 9.01.2025]
[18]Abid All Awan, "What is Named Entity Recognition (NER)? Methods, Use Cases, and Challenges". 13 September 2023, [Online]. Available: https://www.datacamp.com/blog/what-is-named-entity-recognition-ner?utm_source=google&utm_medium=paid_search&utm_campaignid=19589720821&utm_adgroupid=152984011814&utm_device=c&utm_keyword=&utm_matchtype=&utm_network=g&utm_adposition=&utm_creative=719914245961&utm_targetid=dsa-2222697810918&utm_loc_interest_ms=&utm_loc_physical_ms=9210493&utm_content=DSA~blog~Machine-Learning&utm_campaign=230119_1-sea~dsa~tofu_2-b2c_3-row-p1_4-prc_5-na_6-na_7-le_8-pdsh-go_9-nb-e_10-na_11-na-
[Accessed 9.01.2025]
[19] Pardeep Kumar, Kanwal Garg, "Data Cleaning of Raw Tweets for Sentiment Analysis", 01 September 2020, [Online]. Available: https://ieeexplore.ieee.org/document/9181326
[Accessed 9.01.2025]
[20] Aditya Singh Rajpurohit, Harshada Mhaske, Pradnya Sangitbabu Gaikwad, "Data Preprocessing for Stock Price Prediction Using LSTM and Sentiment Analysis", 04 May 2023, [Online]. Available: https://ieeexplore.ieee.org/document/10112026
[Accessed 9.01.2025]
[21],[25] Muhlis Sarıyer, Ahmet Akıl, Feyza Nur Bulgurcu, Fatih Emin Öge, Murat Can Ganiz, "Individual Stock Price Prediction by Using KAP and Twitter Sentiments with Machine Learning for BIST30". 23 September 2022, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9894172
[Accessed 9.01.2025]
[22],[23] Zeynep Hilal Kilimci; Ramazan Duvar, "An Efficient Word Embedding and Deep Learning Based Model to Forecast the Direction of Stock Exchange Market Using Twitter and Financial News Sites: A Case of Istanbul Stock Exchange (BIST 100)". 09 October 2020, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9218927
[Accessed 9.01.2025]
[24],[26],[27] Ahmad, H. O., & Umar, S. U. (2023)." Sentiment analysis of financial textual data using machine learning and deep learning models", 2023, [Online]. Available: https://www.informatica.si/index.php/informatica/article/view/4673 [Accessed 9.01.2025]
[28]Adnan GUMUS , C. Okan SAKAR, "Stock Market Prediction in Istanbul Stock Exchange by Combining Stock Price Information and Sentiment Analysis", 30.01.2021, [Online]. Available: https://dergipark.org.tr/en/pub/jeps/issue/59877/683952 [Accessed 9.01.2025]
[29] Google Play App Requirements: https://support.google.com/googleplay/android developer/answer/11926878?hl=en
[30] Apple iOS App Store Requirements: https://developer.apple.com/news/upcoming requirements/

- Use Case Explanation: https://www.techtarget.com/searchsoftwarequality/definition/use-case

- Functional & Non-Functional Requirements Explanation: https://www.jamasoftware.com/requirements-management-guide/writing requirements/functional-requirements-examples-and-templates

- Software Requirement Specification Example Template: https://view.officeapps.live.com/op/view.aspx?src=https%3A%2F%2Fwww.cse.msu.edu%2F~cse435%2FHandouts%2FSRSExample-webapp.doc&wdOrigin=BROWSELINK

- For drawing Use Case Diagrams (Figure 3-7.): https://app.diagrams.net/

• For drawing Agile Diagram (Figure 2.): https://wepik.com/

- OpenAI ChatGPT: https://chatgpt.com/
- Tweepy: https://docs.tweepy.org/en/stable/
- Beautiful Soup: https://www.crummy.com/software/BeautifulSoup/bs4/doc/
- Scrapy: https://docs.scrapy.org/en/latest/
- Selenium: https://www.selenium.dev/documentation/
- Demoji: https://pypi.org/project/demoji/
- NLTK: https://www.nltk.org/
- spaCy: https://spacy.io/
- IBM: https://www.ibm.com/topics/named-entity-recognition