Project name	A System for User Friendly Stock Price Prediction and Sentiment Analysis
Supervisor	Doc. RNDr. Irena Holubova, Ph.D.
Annotation	The aim of the project is to develop a generic SW framework, that will enable to analyze information about evolution and future prediction of stock prices of a company with sentiment analysis of articles related to that company. The tool will enable to crawl both stock exchange data and financial articles related to a particular company, apply the respective predictions and sentiment analyses on them and visualize this information in a user-friendly way.

1. OVERVIEW

This document contains the Software Project Proposal for the **Stock Price Prediction and Sentiment Analysis System.** The members of the team are as follows:

- Harun Ćerim
- Ilda Balliu
- Kaan Yöş

The team members already have experience with software development, including building front-end and back-end applications. Taking this into account, we decided to implement a system in which we could integrate these skills and also learn more about machine learning, sentiment analysis and the stock market.

The main purpose of this project is to create a system, served to the users as a web application, through which they will be able to get information about stock prices of companies they choose. The users will also be able to see historical data of the company's stock prices over years. We will use these historical data in the machine learning algorithm to predict the future stocks price of the company. This process will be explained in more details in the next specification. Also, the user will be able to see the latest analyzed articles concerning a specific company. The articles will undergo a sentiment analysis to see if they have a positive, negative or neutral outcome regarding the company. Note that we are aware of other similar software services in the market, such as the Wallet Investor[1], but compared to these platforms our system will also provide the analyzed articles to give the user a deeper insight. In general, this system will be useful for individuals/companies interested in the stock market, current and future investors who want to track companies' progress in the stock market.

The web application will offer the following functionalities:

- searching for a specific company that the user is interested in,
- getting an overview of past and current stock prices of this company (their graph visualizations),

[1] https://walletinvestor.com/

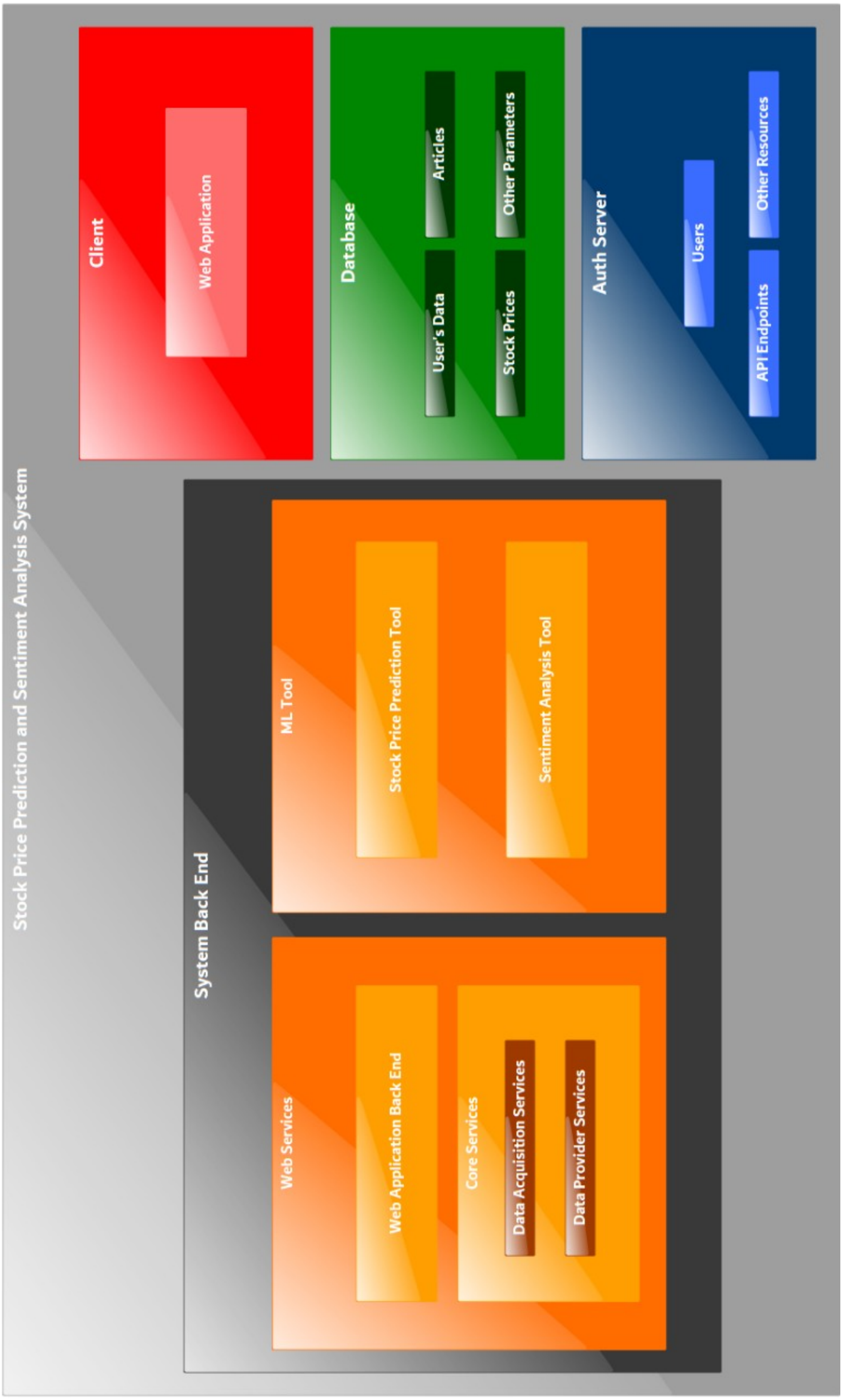
- getting the estimated future stock price based on user-defined parameters (e.g. time period),
- getting articles that might affect the shown price.

If the company the user is looking for is still not in the database, then the user will be informed about it and explained how to ask the administrators to add it. The administrator then will perform all the necessary steps (adding company to the database, training of the machine learning modules etc.).

The system will be proposed as a general modular framework enabling easy replacement of modules, such as e.g., the modules for stock price prediction or data crawling. The modularity will enable future extensions of the selected parts of the project e.g., within Master theses of some of the participating students.

2. SYSTEM UNITS

The system will be divided into smaller implementation units (modules) as depicted in Figure 1.



<u>Figure 1</u> *– Stock Price and Sentiment Analysis System Decomposition View*

Client module is an interface of the system and for the initial version the system will focus on the web application. It is going to have all functionalities of the system available to the end user.

Authentication Server as an individual module will be responsible for keeping resources, users, and APIs secure. The main purpose of this module is to provide token issuing functionality where the end user will get access tokens to use other services provided through the interface.

Database will keep all the information necessary for the System Back End module. There will be stored user's data, financial articles, stock prices, companies' information and other parameters. All the data will be stored in a structured way which means stored by the specification, so it can be easily used by the back-end services and machine learning (ML) tools.

ML tool as the core part of the system will be responsible for predicting stock prices and extracting sentiment from articles which will help the end user in deciding whether some stocks are good to be bought or sold depending on the current stock price, the predicted stock price for a specified date, the average price from a specified period, etc. Classified articles will help the user to check, e.g., whether financial analysts have said something about the company of users' interest. Stock price prediction tool will focus on predicting the stock prices from the previous values and parameters. Sentiment Analysis tool will focus on classifying articles from analysts in the financial world as, e.g., positive, negative, or neutral.

Web Services module is going to deal with providing back-end logic and functionality to front-end application and to ML Tool.

3. HIGH-END FUNCTIONALITIES

The historical data will be obtained from financial API and a custom scraper developed by the team that will get current information from relevant websites for a sufficient time interval. The data will be transformed and stored in a predefined structured way to be used by the prediction tool and for information visualization in the client application. The prediction tool and sentiment analysis tool will consist of algorithms that will be continuously trained by the team during development and new information gathering.

As depicted in Figure 2, the end application will have the following functionalities:

- **Authentication and authorization**
  This involves the registration, logging in/out.
- **Searching, searching history**
  Searching capability for a specified company/set of companies using selected criteria and parameters. Ability to see the previously searched companies/parameters.
- **Check predicted prices**
  Check the predicted price for a specified time and company using visual graph representations.
- **History of prices for a specific period using visual graph representations**
  Ability to see the prices in the past, the trends etc., using visual graph representations.
- **Showing the prices of user's saved companies**
  The application will enable users to save their favorite companies and see them right after logging in, without the need to search again.
- **Showing the related articles and respective sentiment**
  Users will be able to see new analyzed articles for a specific company and to navigate to the source from which these articles are acquired.

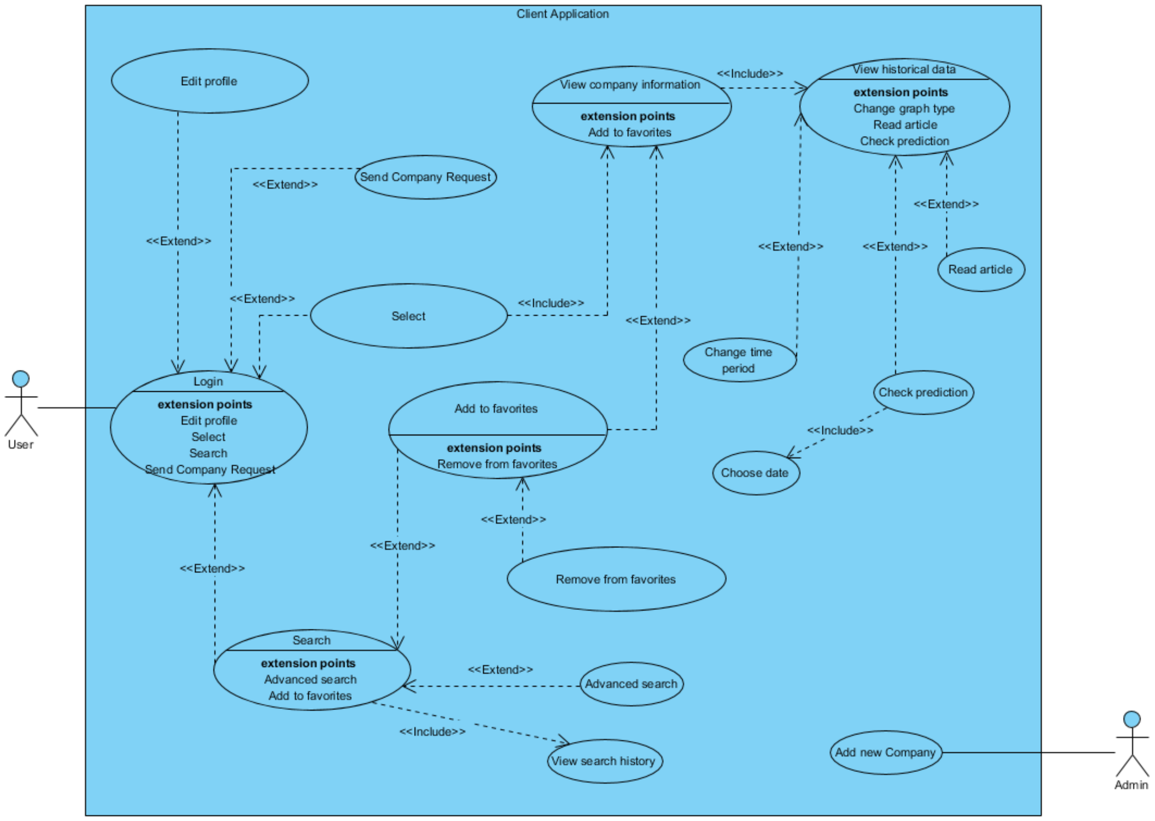


__Figure 2__ – Stock Price and Sentiment Analysis System Use Case Diagram

3.1. Use Case Example

The user first needs to register and then log in to the system. From that point, the user will be able to modify his profile, specify favorite companies that will be available in the dashboard and see the search history. The user will be able to search for a specific company and see its details including; stock prices, latest analyzed articles and choose the date of the prediction. The user can request a new company to be added to the dataset.

4. SOFTWARE PLATFORMS

In this part, there will be described technologies that are going to be used to develop this system.

4.1. Back-end

4.1.1. Web Services

Web services consist of back-end and core services of the web application. The back-end is going to be developed in .NET Core using C# and will be deployed as Lambda application in Amazon Web Services (AWS). For long performing tasks, AWS EC2 will be used as a virtual machine. Data Acquisition and Data Provider Services which belong to Core Services will be developed using .NET Core and C#. Public APIs, such as IEX, will be used to gather historical data about the stock prices and articles from Market Watch.

4.1.2. ML Tool

ML Tool of back-end consists of sentiment analysis and stock price prediction. To predict future stock prices, articles gathered in Core Services must be processed. The Python toolkits called Natural Language Toolkit (NLTK) and TensorFlow will be used to create an article review classifier. The articles will be cleaned and preprocessed before usage in these toolkits. For articles used in the machine learning algorithms must be converted to a numeric representation by vectorization. The, the team will be able to build a classifier and train a model by unsupervised learning which is going to be used for extracting sentiment.

To predict the future prices, the team will develop algorithms for technical analysis which includes reading the charts, articles, historical stock prices, and statistical data. 'Moving average technique' with multiple types of regression algorithms will be used to predict the closing price of stock since profit or loss calculation is generally determined by the closing price of a stock for the day.

In case the application does not have a certain company and the data related to it, the user can send a request and the company will be added by admin. The system will be generic in a way that the back-end algorithm and database structure will not require modification in case of adding additional companies. The requested company will be added, and the model will be re-trained with the new data.

4.2. Database

We are going to use a NoSQL database, most probably DynamoDB. Everything concerning companies, stock prices, articles, parameters, user's data etc., will be stored there.

4.3. Authentication Server

We will be using Amazon Cognito and its' User Pools as the authorization server.

4.4. Client application

The web application will be developed with NodeJS and Angular.

5. SOFTWARE ENGINEERING METHODS

The software will be developed using the agile development model with several key iterations. The first iteration will focus on data scraping and storing data in the database in a structured way. The rest of the iterations will focus on building the project in small portions.

This section will provide an overview of what methods we will use to create the product and how we will perform quality assurance testing.

5.1. Coding Standard

We will follow a coding standard for the development of the project that will increase readability and maintainability. (The coding standard will be presented in the Implementation Plan.)

5.2. Code Reviews

As a team, we will meet the last day of every week to conduct code reviews.

5.3. Object-Oriented Design

We will use the object-oriented method for program design and keep this design updated with fixes for any potential problems we may catch. We will also perform design reviews throughout the project to ensure that the existing design will meet the criteria for the product.

5.4. Change Log

We will keep a track of every change in the project by using *bitbucket* version controller to get a better understatement of versions.

6. PROJECT AREAS

The project focuses on the following areas:

Discrete Mathematics and Algorithms	
	Discrete Mathematics and Algorithms
	Geometry and Mathematical Structure in Computer Science
	Optimization
Theoretical Computer Science	
	Theoretical Computer Science
Software and Data Engineering	
✓	Software Engineering
✓	Software Development
✓	Web Engineering
✓	Database Systems
✓	Analysis and Processing of Large Data
Software Systems	
	System Programming
	Reliable Systems
	Powerful Systems
Mathematical Linguistics	
	Computer and Format Linguistic
	Statistical Methods and Machine Learning in Computer Linguistics
Artificial Intelligence	
	Intelligent Agents
✓	Machine Learning
	Robotics
Computer Games and Computer Graphics	
	Computer Graphics
	Development of Computer Games