Ministry of Education and Science of Ukraine
Ternopil Ivan Puluj National Technical University

Faculty of Computer Information Systems and Software Engineering
(full name of faculty)
Computer Science Department
(full name of department)

QUALIFYING PAPER

For the degree of

Bachelors degree
(degree name)

topic: **Climate data analysis and visualization information system**

Submitted by: fourth year student ____ , group ICH-42
specialty Computer Science
Computer Sciences 122
(code and name of specialty)

	(signature)	Zvaipa Millicent (surname and initials)
Supervisor	(signature)	Nazarevych Oleh (surname and initials)
Standards verified by	(signature)	Shymchuk Hryhorii (surname and initials)
Head of Department	(signature)	Bodnarchuk Ihor (surname and initials)
Reviewer	(signature)	(surname and initials)

Ternopil
2021

Ministry of Education and Science of Ukraine
Ternopil Ivan Puluj National Technical University

Faculty Faculty of Computer Information Systems and Software Engineering
(full name of faculty)

Department Computer Science Department
(full name of department)

APPROVED BY
Head of Department
Bodnarchuk I.O.
(signature) (surname and initials)
« » 20___

ASSIGNMENT
for QUALIFYING PAPER

for the degree of Bachelor
(degree name)

specialty 122 Computer Sciences
(code and name of the specialty)

student Zvaipa Millicent
(surname, name, patronymic)

1. Paper topic Climate data analysis and visualization information system

Paper supervisor Nazarevych Oleh Bohdanovych, PhD, Assoc. Prof.
(surname, name, patronymic, scientific degree, academic rank)

Approved by university order as of «*01*» *01* 2021 № *4/7-62*

2. Student's paper submission deadline 26.06.2021

3. Initial data for the paper

4. Paper contents (list of issues to be developed)
Annotation. 1. Climatology, meteorology and existing systems for data analysis and visualization
2. Selection of software and development of the server part
3. Development of the client part of the system of analysis and visualization of climate data
4. Life safety
Conclusion. Bibliography.

5. List of graphic material (with exact number of required drawings, slides)

6. Advisors of paper chapters

Chapter	Advisor’s surname, initials and position	Signature, date	
		assignment was given by	assignment was received by
Life safety, fundamentals of labor protection			

7. Date of receiving the assignment January 25, 2021

TIME SCHEDULE

LN	Paper stages	Paper stages deadlines	Notes

Student ____________ Zvaipa Millicent
(signature) (surname and initials)

Paper supervisor ____________ Nazarevych Oleh
(signature) (surname and initials)

ANNOTATION

Climate data analysis and visualization information system// Qualification work of the educational level "Bachelor" // Zvaipa Millicent // Ivan Pul'uj Ternopil National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer Science, ICH-42 group / / Ternopil, 2021 // P. 74 , fig – 42, table – 2, bibliography – 20, added – 0.

Keywords: software, interface, MS SQL Server, computer system, JavaScript, C #, data analysis, visualization.

In the bachelors thesis the analysis of scientific researches and existing programs for the analysis and visualization of climatic data is carried out as a result of which urgency and novelty of research work has been proved.

Software tools for development and planning of client-server application architecture, methods of analysis and generalization of MS SQL Server for database creation, C # and JavaScript (ECMAScript 6 standard) and ReactJS v.16.12.0 framework based on it are chosen for methods of analysis and generalization for the client for the application.

The application of object-oriented approach for the design and development of the client part of the program is experimentally substantiated. A computer system for data analysis and visualization was planned and created; the scientific novelty of the research was proved by experiment and generalization.

In particular, the developed application features logical operations on analysis of historical climatic indicators in the client part, which accelerated the process of information visualization for the user.

LIST OF SYMBOLS, UNITS, ABREVIATIONS AND TERMS

DHT – Distributed hash table

LCD – Liquid crystal display

FBD – Function Block Diagram

MQTT – Message Queue Telemetry Transport

ISO – International Organization for Standardization

IEC – International Electrotechnical Commission

IoT – Internet of Things

STOMP – Simple Text Oriented Message Protocol

HTTP – Hyper Text Transfer Protocol

TCP – Transmission Control Protocol

OIC – Open Interconnect Consortium

CES – Consumer Electronics Show

CONTENTS

INTRODUCTION

Actuality of theme. The problem of climate data analysis has became very relevant since the early 2010s. Climate change has a significant impact on various spheres of human activity, both economic (especially in the agricultural sector, which generated 13% of Ukraine's GDP in 2019) and for the social sphere.

One of the most convenient and concise methods of studying and analyzing large data sets is visualization using computer systems. Thanks to the dynamic and static graphs, charts and maps created with the help of software, it is possible to clearly trace trends in climatological and meteorological phenomena, identify features and patterns of processes for further climate change forecasting, which can be used to adjust plans and expected results.

In the 21st century, more and more software is being created that increases the speed and quality of the data analysis and visualization process, but despite the large number of programs created, there are very few special web applications for working with climate data. Among them are GIOVANNI and Earth Observatory, created by NASA, KNMI Climate Explorer – developed and maintained by the Royal Meteorological Institute of the Netherlands and ESA CCI.

Despite the strengths of these web applications, there are many factors that complicate their operation by users (outdated technology, inconvenient user interface, etc.). Therefore, the urgent scientific and technical task is to study the creation of client-server software for analysis and visualization of climate data using modern programming languages and historical climate data visualization, which will create an application that would meet the high modern requirements of computer system architecture and UI / UX design for user convenience.

The aim of the study. The aim of the study is to analyze the methods and tools for creating an effective computer system for analyzing and visualizing climate data.

The main objectives of the study:

- To analyze scientific publications and existing software for analysis and visualization of climatic and meteorological data;

- consider modern tools for creating client-server web applications;

-To investigate the peculiarities of processing and analysis of climatic indicators;

- analyze ways to visualize large data sets;

- to develop and test a computer system for analysis and visualization of climate data, taking into account modern requirements for software development and UI / UX design of web applications.

Object of study – The object of research is the process of developing a program for analysis and visualization of climate data.

Subject of study – The subject of research is models, methods and means of presentation and development of client-server web application for analysis and visualization of climate data.

Research methods – The following methods were used to solve the tasks:

- analysis and generalization – when analyzing existing tools for processing large data sets and their visualization, as well as when reviewing available software for creating a web application;

- Formalization – when substantiating the client-server architecture for the development of a computer system for data analysis and visualization;

- Design and programming – when using selected programming languages, frameworks and libraries to create software;

- Experiment and measurement – to develop a web application for analysis and visualization based on real historical climatic indicators.

The scientific novelty of the obtained results lies in solving the scientific and practical problem of analysis and visualization of climatic indicators, with the following results:

- the means of analysis and visualization of selected climatic and meteorological data are substantiated in detail, which allows to determine and forecast climate trends in accordance with geographical reference and time indicators;

- The expediency of using modern software development tools and interactive methods of data visualization when creating client-server web applications is shown.

The practical significance of the obtained results. The practical value of the results obtained during the study is that the designed and developed system allows processing large arrays of historical climatic data and visualizing the results in the form of user-friendly and understandable graphs and charts, which in turn allows adjusting the developed economic and social programs. Development of enterprises and organizations and use the results obtained when creating new programs.

1. CLIMATOLOGY, METEOROLOGY AND EXISTING SYSTEMS FOR DATA ANALYSIS AND VISUALIZATION

1.1. Climatology and meteorology and their researchers

Science that studies and explains the physical processes and phenomena that occur in the atmosphere during its interaction with the surface of soil, water, vegetation, etc. ("Underlying surface"), is called meteorology. The study of processes and phenomena observed in the atmosphere is of great practical importance and allows studying the patterns of their development, which makes it possible to develop methods for predicting atmospheric processes, and in some cases allows changing their development [1].

The complexity of the subject of meteorological research and the needs of different sectors of the economy has led to the need for different industry disciplines. Based on the division of the subject of study in meteorology, the following subsectors are distinguished:

-Synoptic meteorology (weather forecasting);

-Dynamic meteorology (theoretical foundations of science);

-Actinometrical (study of the modes of flow of solar radiation);

-Climatology.

Qualitatively and quantitatively, the physical state of the atmosphere and the processes occurring in it are expressed in indicators called meteorological elements [2].

The most important for human economic activity are the following elements:

- Surface temperature;
- Surface humidity;
- Atmospheric pressure;
- Air temperature;

- Air humidity;
- Clouds;
- Precipitation;
- Wind.

These elements are in constant interdependence, they are characterized by complex interaction, which is manifested in complex and variable combinations.

The state of the atmosphere over a specific area for a certain period of time, which is determined by the physical processes of its interaction with the underlying surface, is called weather.

Climate is a natural sequence of atmospheric processes created in a particular area as a result of the interaction of solar radiation, atmospheric circulation and physical phenomena occurring on the underlying surface, and determines in this area its characteristic weather regime.

The climate is characterized by long-term weather. The long-term weather regime means not only the prevailing, but also generally possible weather conditions in a given geographical area. The main climate-forming factors are:

- Solar radiation;
- Atmospheric circulation;
- The nature of the backing surface.

As a result of their joint interaction is the formation of climate in different parts of the globe.

The field of science that studies the conditions of climate formation and the climatic regime of countries is called climatology. Climatology studies the relationship between climate-forming factors and their interaction with the underlying surface. Climatology deals with the study of patterns distributed on the surface of the planet of various meteorological phenomena and types of climate. It also examines human impact on the climate.

Climatology studies and describes the climatic conditions of large areas. However, in small areas under the influence of local factors may create certain features of the climate. The set of local features of the climate is called the microclimate of a given geographical location.

The branch of climatology that studies the formation of the climate of the surface layer of air and describes the microclimatic features of certain geographical areas is called microclimatology.

Researchers identify the following main tasks of climatology as a science:

- study of the genesis of climate formation (climate formation) as a result of climate-forming processes and under the influence of geographical factors;
- Description of climates of different geographical areas,their classification and study of distribution;
- Study of climates of historical and geological past (pale climatology);
- Climate change forecast;
- establishing patterns of microclimate formation and its classification;
- creating models of climate change in the future.

Modern scientific meteorology dates back to the XVII century, when the foundations of physics were laid, part of which was originally meteorology. At the same time, the first meteorological instruments were invented and instrumental observations became possible.

During this period, MV Lomonosov singled out meteorology as an independent science with its own tasks and methods, he created the first theory of atmospheric electricity, explored the possibility of scientific weather prediction. In the work “On the Earth’s Layers” Lomonosov was one of the first to express the theory of climate change on our planet in the process of its development.

In 1749, the results of twelve years of climate observations in Europe were published, which gave impetus to intensify further meteorological research.

In the nineteenth century, German scientists A. Humboldt and G. Dove in their research laid the foundations of climatology. A. Humboldt in his scientific work "Space" gave a new definition of climate, which took into account various meteorological factors.

In 1826 the first weather maps were created. The authorship of this method of climate research belongs to the German scientist GV Brandesu. From the middle of the XIX century on the initiative of the French researcher L. Leverier the synoptic method of research of atmospheric processes came into wide use.

This period also includes the organization of the first meteorological research institutes. W. Ferrell (USA) and G. Helmholtz (Germany) lay the foundations of dynamic meteorology. At the end of the XIX century, the study of radiation and electrical processes in the atmosphere intensified.

In the twentieth century, the development of meteorology accelerated; in 1920 L. Richardson created the first mathematical weather forecast. In 1953, computers were used for the first time for simple meteorological calculations.

The next stage in the development of meteorology was in 1961-1967. During this period, due to the increase in the number of processed climatic indicators for data analysis and forecasting, the first computer systems appeared.

At the present stage of development of meteorology and climatology is characterized by the processing of a large mass of data collected by ground and space observation stations. Among modern researchers are the works of P. Krutzen, J. Teixeran de Bohr, M. Milankovich and M.I. Culbids, who studied global climatechange and its impact on the development of the biosphere.

1.2 Types of climatic indicators with their calculation

Climatologically processing of survey makes it possible to obtain more of climatological indicators which characterize the prolonged regime of meteorological quantities with the atmospheric phenomena (weather) for a particular area. The main kind of climatological indicators are as follows:

a)Climatic indicators of individual meteorological elements(quantities) and phenomena (one-dimensional statistical characteristics);

b) Complex climatic indicators (one-dimensional and multi-dimensional statistical characteristics);

c) Indicators of the dated structure of meteorological elements and their complexes (one-dimensional and multi-dimensional statistical characteristics). Climatic indicators of the single meteorological quantities and the phenomena include:

1. Average worth. Within the climatology, the calculation mean of meteorological quantities are considered as the average. With a large figure of members of the meteorological series, it is very advantageous to give a large collection in the structure of a variation series-classes and frequencies of variations.

In meteorology, such statistical indicators of distribution as average, we us mode and median. On a particular variable, the general characteristics of the distribution of data is shown, allows you to identify value (or values –if the mode in the distribution is more than one), which narrates the whole distribution. Mode, median and mean are single values which display the whole data collection, typical of the entire values of the group. We have some general rules where we use the average, in particular:

1. The center of gravity" of the distribution is the mean; each and every value contributes to the definition of the mean when distributions of values are symmetric all over the center point.

2. The mean is greater stable than mode or median. Therefore, when we are finding the most stable measure of the central trend, we can use the average. The mean is the most widely-known measure of the central trend, or arithmetic mean, or simple mean,-just the midpoint value of the series of statistics.

3. To compute it, it is enough to attach the entire values in the distribution and then divide by the number of survey, as we can see it from the formula:

$$M = \frac{\sum x_i}{n} \qquad (1.1)$$

Whereas M is the arithmetic mean, n is the value of indicators.

Median can be stated as a point on the distribution series (an ordered set of numbers of a variable for different observations, for example, from the smallest to the largest number), half of all values are located at this point, and another half of the values after this point. That is, the median number is the value that we use to divide the ordered series in half. If the number of the values is odd, then one of the values is taken-that which is in the exact distribution in the center. When values are even numbers, then take two central values and find the average of those two numbers.

The median number of the distribution function F is a value equal to:

$$F(\tilde{x})=1/2 \qquad (1.2)$$

The probability that a random variable will have a number which is greater than or less than the median is the identical and uniform.

To solve the following problems we use median:

1. The median is the best measure of the central trend, when extreme values affect the mean.

2. When we are finding the exact midpoint, the point "halfway" from the smallest number to the greatest number.

3. We use the median when it’s necessary that certain numbers affects the central trend, but all we know about them is that they’re all "below" or "above" the median.

The value that is common in the distribution is the third measure of the central trend of fashion .As a rule; it represents the greatest typical number. In the distribution fashion is never influenced by utmost values, but by the utmost frequencies of numbers, how often a number of a variable occurs in the distribution.

The specific modal value is calculated by the formula:

$$M_o = x_o + h\frac{f_{mo} - f_{mo-1}}{(f_{mo} - f_{mo-1}) + (f_{mo} - f_{mo+1})}, \qquad (1.3)$$

Where x_0 and h are the lower limit and width of the modal interval,

$f_{m0,\ fm0-1,\ fm0+1}$ -Frequencies (fractions) of modal, premodal and post modal intervals, respectively.

Fashion is used to determine the extent of a central trend. Fashion is not affected by extreme values; it can be defined for open intervals / categories. To identify the mode of analysis of climatic data, you need to build a distribution graph or bar chart.

You can represent the distribution of data (especially one-dimensional distribution) of different types of graphs. Typically, two dimensions (two axes) are used in graphs for distributions. The values of the variables are placed on the X axis, and the frequency of these variables is placed on the Y axis, in absolute or relative values.

The normal distribution describes many natural phenomena, it has become the de facto standard for solving climatic and meteorological problems.

The distribution in which all three measures of the central tendency coincide, ie the mean is equal to the median and is equal to the mode, is called normal. It is also called the Gaussian distribution or "bell-shaped" (Figure 1.1)

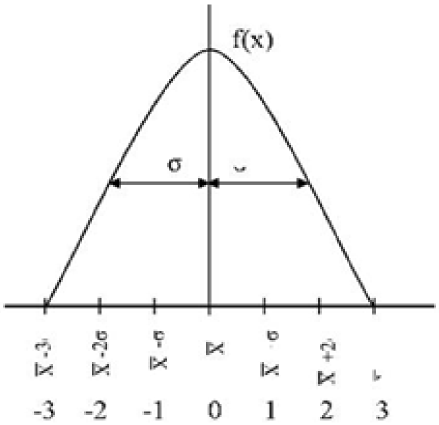


Figure 1.1 – Normal distribution curve (Gaussian curve)

In the normal distribution, most of the data values tend to be grouped "clustered" around the average value. The farther the value from the average, the less likely it is to occur. Of course, the distributions in real life do not exactly correspond to normal. The central limit theorem of probability theory suggests that the set of independent, approximately equally distributed quantities has a normal distribution.

The values in the distribution are also divided along the X axis by standard deviations. The standard deviation is used as an indicator of how "scattered" relative to the mean is the value of a variable in a data set, and is measured in the same units as the variable.

2. Extreme values (maximum and minimum), which characterize the ranges in which the values are meteorological values that are observed for a certain period of time in a particular area. Extreme values of meteorological values are chosen from long-term observations. Determine the absolute maximum, minimum (extremums), average maxima and minima. Average highs and lows are calculated as average long-term values of daily (daily maximum or minimum), monthly (monthly maximum or minimum), annual (annual maximum or minimum).

3. The repetition of different values of meteorological quantities, ie the frequencies of a certain interval in the series of statistical distribution are

expressed as a percentage or fractions of a unit. Repeatability, which is obtained from a long series of observations, is called probability in climatology.

4. Indicators of variability of values of meteorological quantity. This includes the standard deviation, the standard deviation, the coefficient of variability (variation). The root mean square is the sum of the square deviations of individual values of a given quantity from the mean division by the number of 2 variants. Sometimes it is called the mean square or sigma, ie the standard deviation is a number named and is expressed in those units that the arithmetic mean. Remember that it has two signs: plus and minus. The coefficient of variability is the ratio of the standard deviation to the arithmetic mean deviation. The arithmetic mean, the mean square and the mean square (absolute) deviation are called the parameters of the population.

5. Indicators of asymmetry and steepness of the distribution curve are taken in those cases when the distribution of the feature differs from the normal one. The measure of the asymmetry of the distribution is the coefficient A, which is equal to the ratio of the mean cube of the deviation of the magnitude of the sign from the arithmetic mean to the cube of the standard deviation. If (A) <0.25 – the asymmetry is small, if 0.25 <(A) 0.50 – large. The coefficient of excess (excess K) is used to characterize the steepness of the distribution. The coefficient of excess of oscillations from -2 to 2. If K is close to 2, then this indicates that the distribution curve is depressed and can turn into a two-vertex curve; if K = – 2, then the distribution curve consists of two separate curves. For meteorological phenomena, such climatic indicators as the average and the largest number of days with this phenomenon, the average duration of the phenomenon (in hours) are noted.

For the analysis of climatological and metrological data there are the following types of scales:

1. Nominal. A scale whose values cannot be ordered. The only operation that can be applied to data with a nominal scale is a list, a calculation of the frequency of occurrence of a value in the array.

$$A=\{x1,\ldots,xN\}, \qquad (1.4)$$

$$\delta_{ij}=\frac{=1, \text{якщо} x_i = x_j}{=0, \text{якщо} x_i \neq x_j} \Big| \qquad (1.5)$$

2. Ordinal. A variable whose values are by definition ordered (for example, "weak", “moderate", “strong"). The value of such a variable can be compared to identity, to "more or less", but it is impossible to say how much more or less. Example:

$$(x>y)\ i\ (y>z), \qquad (1.6)$$

$$R_i=\sum_{j=1}^{N} C(x_i-x_j) \qquad (1.7)$$

3. Metric scales.

- Relationship scales. The scale of measurements of a quantitative property, variables have all the same properties as intervals, but also have a "natural" zero point of "reference" – for example, the beginning of the time reference, the Kelvin temperature.

- Interval. The values of the variable are arranged as in an ordinal scale, but the differences between the values are meaningful, for example, Celsius temperature, time intervals, and so on. Valid operations are addition and subtraction, but not multiplication or division.

$$\hat{x}_{ij}=\frac{x_{ij}-A}{B}, \qquad (1.8)$$

Where x_{ij} is the j-th coordinate of the i-th vector, A and B are some pre-assigned numbers, which are called characteristic scales. Among the complex climatic indicators are:

1. Meteorological complexes of unambiguous effect, which are characterized by the fact that individual elements are selected in such gradations and combinations (pre-established) that the complex characterizes a quantitatively unambiguous effect. For example, the combination of values of temperature, humidity and wind speed. This complex is used to assess weather comfort for humans. Or the cloud altitude complex is the visibility that determines the conditions of aircraft landing.

2. Mechanical complexes – various combinations of values of various meteorological quantities. An example of such a mechanical complex is the method of complex climatology, developed by EE Fedorov and LA Chubukov. The essence of this method is that the climate of any territory can be manifested only through local weather, and the weather of each observation period is considered as a complex of the whole set of meteorological elements. Determining the gradations of the values of weather elements makes it possible to distinguish the types of weather and calculate the recurrence of each type (gradations are for each meteorological quantity and for each phenomenon). Thus, the climate is expressed through local weather, which is the essence of climate analysis, which is called complex climatology. Using the results of observations, they formulate the local weather formula for each day (each meteorological quantity and phenomenon has its own code), then compile a catalog of weather for a certain period and find the recurrence of different types of weather. The weather catalog makes it possible to follow the nature of weather changes from day to day. The method of complex climatology is successfully used in construction, medical and other applied climatologies.

3. Weather complexes or complexes of meteorological quantities and phenomena, which are fixed in the period of certain natural processes. For

example, such complexes of meteorological elements allow us to speak about the type of weather, which is inherent in such phenomena as droughts, dry spells, and frosts at different synoptic processes. Such methods of natural complexation, which are based on the joint use of synoptic and climatological analysis of atmospheric processes, are called synoptic-climatological and are used in climatological research to study the structure of climate, the reasons for its formation, in synoptic research.

Indicators of the time structure of meteorological series are divided into two groups:

1. Indicators of periodic changes of meteorological quantities and phenomena in time. This includes the daily and annual course of meteorological quantities and phenomena, the amplitude (daily and annual), and the date of occurrence of the values of meteorological elements. The daily and annual course of meteorological elements is expressed by average daily and monthly values. Such indicators are given in climate reference books, meteorological monthly magazines and are calculated on the basis of a long-term series of observations. The daily and annual amplitudes of temperatures are defined as the difference between the highest and lowest values (extremums) of meteorological elements. The timing of certain values of meteorological elements is defined as the date of a stable transition of the values of the element across certain limits (for example, the transition of the average daily temperature through 0, 5, 10, etc.).

2. Indicators of inhomogeneous changes in time of meteorological elements, which are not given in reference books. They are determined by analogy with the corresponding characteristics of random functions, statistical estimates of which they are. In this case, the change of meteorological elements in time is considered as some random process, which obeys the laws of probability theory.

The results of daily observations at meteorological stations are recorded in a special book of the observer, which is the main source of information on the forecast and climate of the territory where the station is located. On the basis of this primary material, which is contained in the observer's book, monthly tables are compiled, in which, in addition to the results of surface term observations, averages and total values, experimental values for days, decades, months, number of days with atmospheric phenomena are given. That is, monthly tables represent the first stage of processing results meteorological observations for the month. The tables are very carefully checked at the meteorological station and in the regional meteorological center. After checking, the monthly tables are ready for printing. Until 1960, they were published in the USSR in the form of meteorological yearbooks.

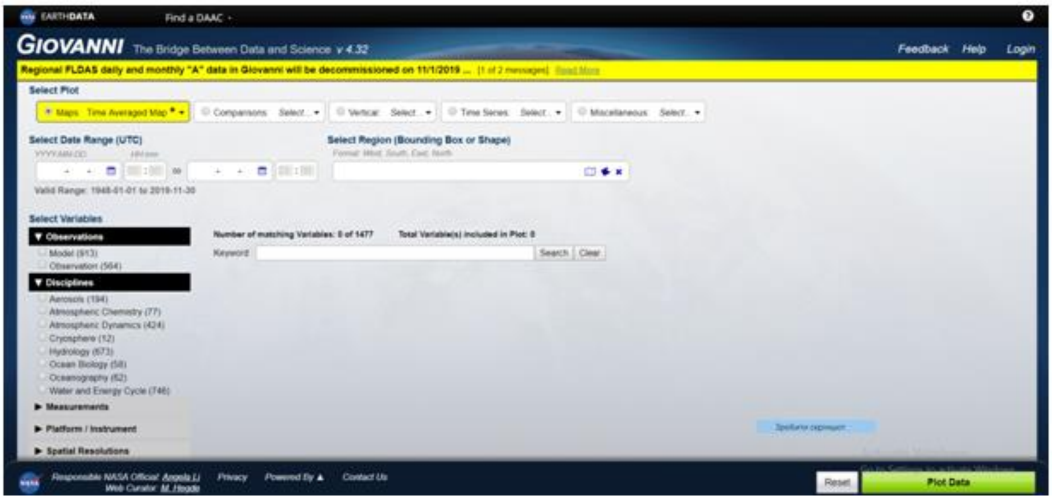
Now regional and state meteorological centers publish meteorological monthly magazines. 13 issues are published per year. Monthly magazines from 1 to 12 months contain data of monthly tables, and in the 13th issue annual data of meteorological observations are published. Meteorological monthly magazines consist of two parts.

1.3 The general review of existing systems for analyzing and visualizing of climate data

The studying examined a number of computer systems for analyzing and visualizing climate data. Giovanni is an abbreviation for GES-DISC(Goddard Science Data and Information Services Center) [4].

ggThe main features of the program are expressed in the abbreviation GrADS-Grid Analysis and Display System. TheGES-DISC interactive online visualization and GIOVANNI analysis system allow you to explore satellite data using sophisticated analysis and visualization.

Among them is the “GIOVANNI” system-a web application developed by GESDISC, which provides a simpler and intuitive method to visualize, to analyze and to access a huge quantity of data from remote sensing of Earth without the need of downloading the data as shownin Figure1.2.



Regional FLDAS daily and monthly "A" data in Giovanni will be decommissioned on 11/1/2019 ... (1 of 2 messages) Read More

Select Plot: ● Maps: Time Averaged Map

Valid Range: 1948-01-01 to 2019-11-30

Number of matching Variables: 0 of 1477

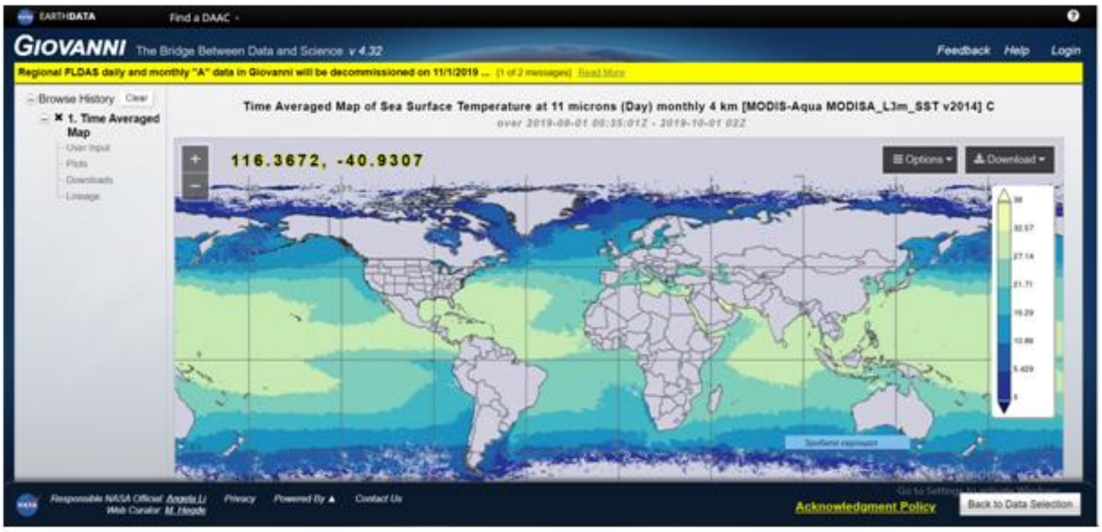
Total Variable(s) included in Plot: 0

Figure 1.2 – Main page of the "GIOVANNI" website

The system allows you to access data from multiple remote sites, supports multiple data formats, including hierarchical data format (HDF), HDF-EOS, network shared data form (netCDF), GRIdded Binary (GRIB), as well as binary and many types of plots, including area, time, How moller animation and images.

In particular, this application allows users to quickly visualize, to interact and to analyze ocean color data from several missions, as well as data sets from many other NASA data archives.

This application has got the ability to visualize the radiometry of satellite remote sensing, pigmentation of phytoplankton, sea surface temperature and sea salinity Shown in Figure1.3.



Regional FLDAS daily and monthly "A" data in Giovanni will be decommissioned on 11/1/2019 ...

Time Averaged Map of Sea Surface Temperature at 11 microns (Day) monthly 4 km [MODIS-Aqua MODISA_L3m_SST v2014] C

over 2019-08-01 00:35:01Z - 2019-10-01 02Z

Figure 1.3 – Visualization of ocean temperature in the program "GIOVANNI"

Animated visualizations of the analyzed data and numerical output of information are also available.

The advantage of an interactive web application for data visualizing and analyzing of the Goddard Data Center and Information Services for Earth Sciences is the acquisition of data from NASA satellites straight on the Internet, without difficulty in traditional methods of data collection and analysis. Scientists used this tool in 2005 on a study of the response of Chesapeake Bayto heavy rain falling the mid-Atlantic in 2005 and 2007.

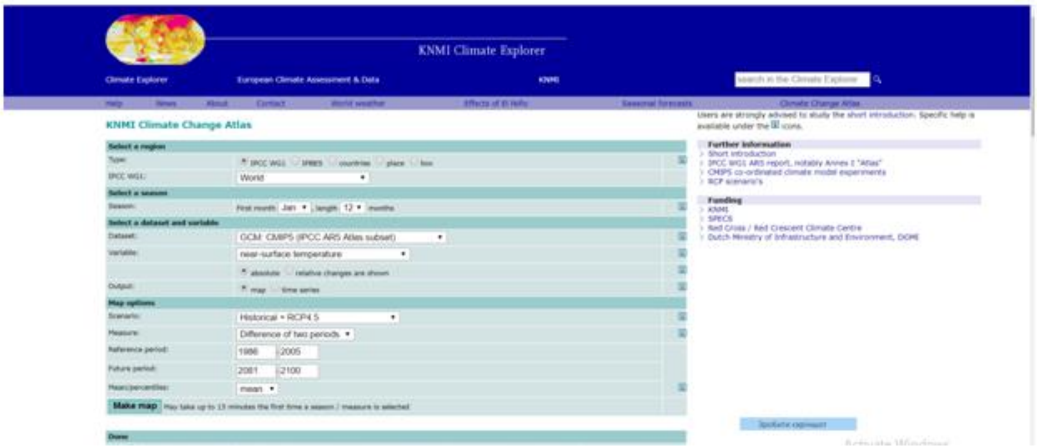
Used in studies of phytoplankton in the Red Sea and gas concentration sover India. In 2008, a cloud cover over Greenland was simulated using the tool.

The Giovanni website offers a variety of tools for educators and users, including the online user’s guide and pages describing availability of data interfaces and story types.

Next, KNMI Climate Explorer was developed. This is a web application for statistical analysis of climate data, developed by the Royal Meteorological Institute of the Netherlands (KNMI). It was created in late 1999 as simpleweb page for the analysis of climate indicators, currently 10 TB of climate data have been processed and dozens of analysis tools have been created. It is now part of the WMO Regional Climate Center at KNMI together with ECA & D.

After free registration, scientists can research and download climate datasets, generate research data, download their own data and perform statistical analysis, research and compare datasets.

Since its inception, this data analysis and visualization system has quickly gained popularity as a tool for climate change research. The form for creating infographics to demonstrate climate change is shown in Figure 1.4.



KNMI Climate Change Atlas

Select a region

Type: ● IPCC WG1 ○ SREX ○ countries ○ place ○ box

IPCC WG1: World

Select a season

Season: First month Jan, length 12 months

Select a dataset and variable

Dataset: GCM: CMIP5 (IPCC AR5 Atlas subset)

Variable: near-surface temperature

● absolute ○ relative changes are shown

Output: ● map ○ time series

Map options

Scenario: Historical + RCP4.5

Measure: Difference of two periods

Reference period: 1986 2005

Future period: 2081 2100

Mean/percentiles: mean

Make map May take up to 15 minutes the first time a season / measure is selected

Done

Users are strongly advised to study the short introduction. Specific help is available under the icons.

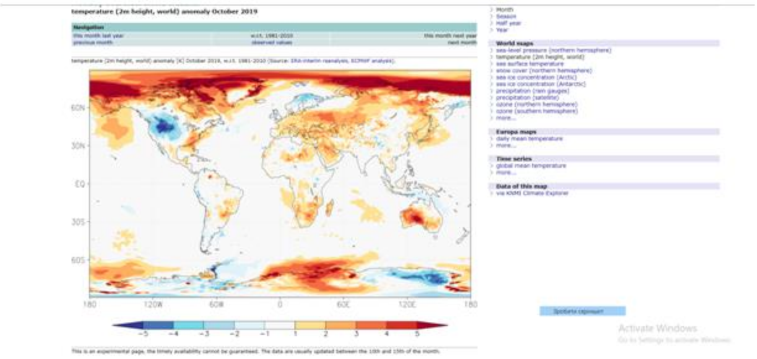
Further information
- Short introduction
- IPCC WG1 AR5 report, notably Annex I "Atlas"
- CMIP5 co-ordinated climate model experiments
- RCP scenario's

Funding
- KNMI
- SPECS
- Red Cross / Red Crescent Climate Centre
- Dutch Ministry of Infrastructure and Environment, DOME

Figure 1.4 – Form for creating info graphics in KNMI Climate Explorer

The application is used by the pale climatic research community to study and download the available analyzed climatic indicators for the selected period, as shown in Figure 1.5.

The application allows you to download high-resolution info graphics to visualize climatic phenomena and create spatial climatic reconstructions. In particular, this tool was used to study and analyze data to create a dendroclimatic atlas of the High Atlas Mountains in Morocco [4].



temperature (2m height, world) anomaly October 2019

Navigation

this month last year | w.r.t. 1981-2010 | this month next year
previous month | observed values | next month

temperature (2m height, world) anomaly [K] October 2019, w.r.t. 1981-2010 (Source: ERA-interim reanalysis, ECMWF analysis).

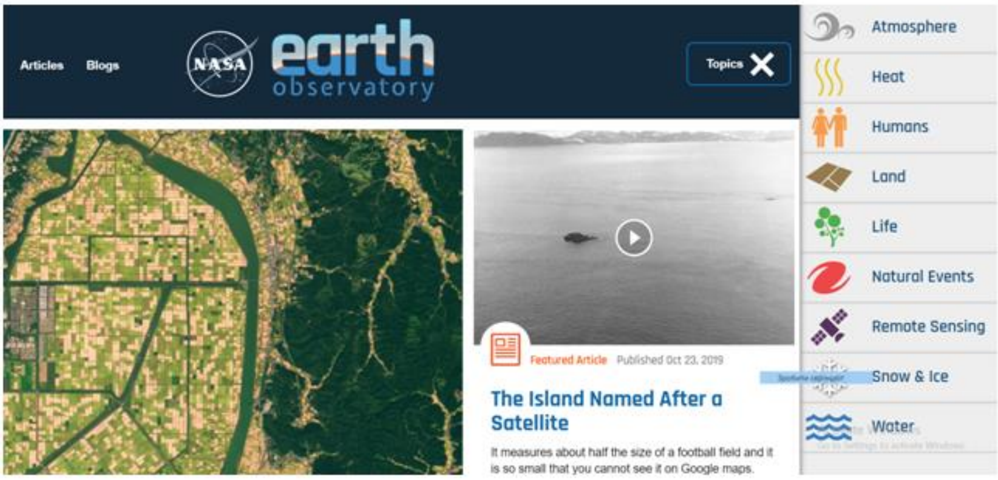
This is an experimental page, the timely availability cannot be guaranteed. The data are usually updated between the 10th and 15th of the month.

60N
30N
EQ
30S
60S
180
120W
60W
0
60E
120E
180
−5
−4
−3
−2
−1
1
2
3
4
5
Month
Season
Half year
Year
World maps
sea-level pressure (northern hemisphere)
temperature (2m height, world)
sea surface temperature
snow cover (northern hemisphere)
sea ice concentration (Arctic)
sea ice concentration (Antarctic)
precipitation (rain gauges)
precipitation (satellite)
ozone (northern hemisphere)
ozone (southern hemisphere)
more...
Europa maps
daily mean temperature
more...
Time series
global mean temperature
more...
Data of this map
via KNMI Climate Explorer

Figure 1.5 – Analysis of climatic indicators for the selected period in the program

A PDF document is available on the website, which includes step-by-step instructions for using the tool.

NASA Earth Observatory is a NASA online resource that was created in 1999 and is an important source of satellite imagery and other scientific information about climate and the environment. Funding comes from the state budget of the United States. The tool is part of the EOS Project Science Bureau, located at the Goddard Space Flight Center.



Featured Article Published Oct 23, 2019

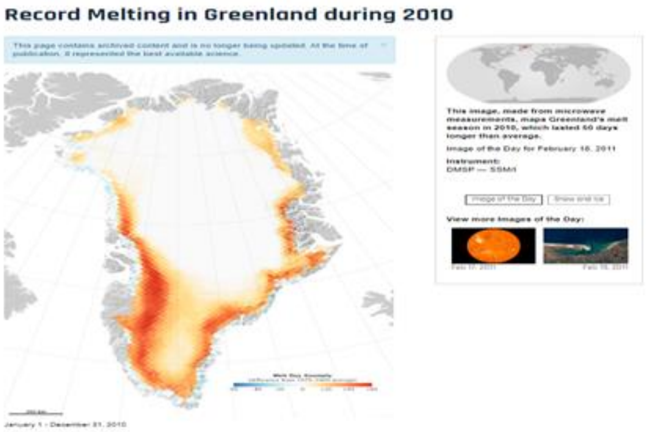
The Island Named After a Satellite

It measures about half the size of a football field and it is so small that you cannot see it on Google maps.

Figure 1.6 – Home NASA Earth Observatory

Here you can track 16 important climatic indicators, such as precipitation, temperature, carbon dioxide level, etc., which are available for selection and navigation on the main page, the image of which is presented in Figure 1.6.

Although the main value of this system for the analysis of climatic and meteorological data is free access to satellite images of the earth's surface and the possibility of its analysis, there are also traditional graphs and charts, such as visualization of temperature anomalies, as shown in Figure 1.7.



Record Melting in Greenland during 2010

This page contains archived content and is no longer being updated. At the time of publication, it represented the best available science.

This image, made from microwave measurements, maps Greenland's melt season in 2010, which lasted 50 days longer than average.

Image of the Day for February 18, 2011

Instrument:
DMSP — SSM/I

View more Images of the Day:

Figure 1.7 – Visualization of temperature anomalies in Greenland in 2010

User manuals and video tutorials on how to use the tool for scientific and educational purposes are available on the main page of the website. It is also worth noting the regular updates of the program with the latest data, which is highly appreciated by researchers and scientists.

ESA CCI (European Space Agency Climate Change Initiative) is a program initiated and created by ESA for global tracking of changes based on meteorological and climatic data collected with the help of artificial satellites. CCI Dashboard – provides a high level of data selection in the archive of the open CCI data portal, including key information for each type of indicator, a quick link to download them via FTP, a graphic image and links to relevant documentation. The image of the tool is given in Figure 1.8.

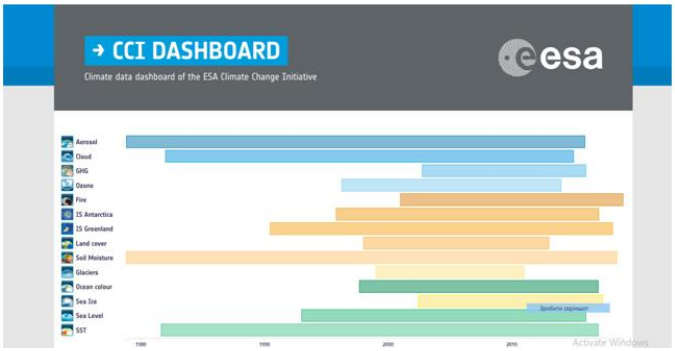


Figure 1.8 – Features of CCI Dashboard

In 2010, the first phase of the program began, launching 10 pilot projects on climate and meteorological data analysis, selected by members of the Climate Science Advisory Board (CSAB) and the ESA Earth Science Advisory Committee (ESAC). The first ESA status reports have been available since early 2011. At this stage, the web application is used to study gas emissions, the Arctic ice sheet, clouds, precipitation, and the state of the world's oceans and the ozone layer of the planet Earth. A number of tools are available in the program for analysis and visualization of data.

CCI Data Search is used to search for and download data from the archives of the open CCI data portal. The tool provides text search and graphical options, which allows users to customize their search query, as shown in Figure 1.9.

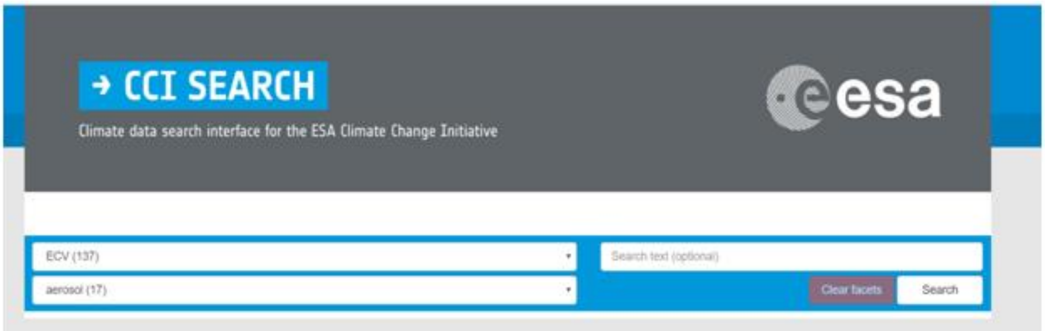


Figure 1.9 – View of the CCI Data Search tool

This computer system uses the CCI Open Data Portal Helpdesk to analyze and resolve technical issues for climate and meteorological data analysis.

1.4 Conclusions to the section

This section discusses the main aspects of climatological and meteorological research and discusses the available systems for data analysis and visualization.

After analyzing the available web applications for working with climate indicators, it was found that these programs work, but have a number of disadvantages:

- The use of outdated and rare technologies for web client development (jQwery v.2.2, HighCharts v.5.5, etc.), which significantly complicates the maintenance, modernization and addition of new functionality;

- User interface that does not meet the modern requirements of UI / UX (User Experience / User Interface) design, which negatively affects the availability and usability of programs;

- Complete absence or limited functionality for analysis and visualization of local climate data;

- Despite the availability of instructions for using these tools, there is a high entry threshold for using the program, which seriously limits the number of users;

- Limited opportunities for visualization of microclimatic and local meteorological indicators. It is also worth noting that despite the fact that the research problem is dealt with by serious research organizations (NASA, ESA, etc.), there is a serious shortage of computer systems for analysis and visualization of climate data.

Thus, the research carried out in this section confirms the relevance of the tasks identified in this master's thesis and the high economic feasibility of creating an alternative application.

2 SELECTION OF SOFTWARE AND DEVELOPMENT OF THE SERVER PART

In order to facilitate the development, maintenance and scaling of the application for analysis and visualization, the software tools were selected below.

2.1. JavaScript programming language and React framework

JavaScript is an object-oriented and prototyping language for software development, which is mostly used in web development [5].

JavaScript differs from other languages for the development of client and server architecture, it belongs to the client side, ie it is performed on the user side, and more specifically the kernel of the operating system browser. JavaScript is commonly used as an embedded language for accessing developed web applications. It is especially used in browsers as a language to add interactivity to web pages.

The main architectural features: dynamic typing, object-prototype paradigm in programming, automatic memory management, functionality or object-orientation.

Using syntax similar to C, JavaScript has different characteristics:

- Access to the type of object during the call;

- Automatic typing;

- Built-in garbage collection;

- Unnamed functions. However, JavaScript does not have some of the required features:

- Creation of modules;

– The ability to manage flows;

- Uniform standards for interfaces;

- Optimized package management system.

The specification (documentary description of syntax, basic concepts and implemented algorithms) of the JavaScript programming language is called "ECMA Script" [6].

The JavaScript kernel contains a number of implementations that provide features such as:

1) The ability to store information in variables;

2) Management of multimedia and interactive capabilities (work with videos, images, animations);

3) Generate content that will be automatically updated;

4) Call and use a certain part of the developed code according to the script and the need that exist on the application page.

The structure of modern JavaScript is as follows:

- Main part – ECMA Script standard,
- Document object model – DOM,
- Browser object model – BOM.

JS is mainly used for front-end programming. If you look in more detail, you can see that JS is used to manage the web application window (browser), change the document tags when working with the DOM, processing various events on the browser page.

In JavaScript, existing data types are divided into two types: primitive and object. There are 5 well-known primitive data types: undefined number, Boolean, string, null and one separate type named object.

JavaScript frameworks are sets of tools and tools for creating dynamic web applications, Smartphone applications, or desktop applications in JavaScript. In common practice, these frameworks are used to develop and create Single-Page Applications, ie everything that happens on the site happens on one page without going directly from it.

ReactJS is a framework in the JavaScript programming language, the code which became available in 2013. This library is used to create and schedule multi-sized web applications where data will change on an ongoing basis [7].

Advantages of ReactJS:

1. Easy to learn. React is much easier to learn given the simplicity and convenience of the available syntax. Developers can use their experience to create HTML. There is no need to study the new TypeScript programming language in detail, as is the case with the Angular framework.

2. Using with ES6 and ES7 ReactJS demonstrates good performance at high loads.

3. Good level of adaptability and flexibility and maximum sensitivity to user needs.

4. Virtual DOM system, which provides the ability to work with HTML, XHTML or XML documents in the project tree, is optimal for creating applications for analysis and visualization.

5. Quite a small amount, because the data running on the client side can also be used on the server side at the same time.

6. Linking data from large to small. This creates a flow of information in which lower-order elements will not affect the parent modules and components.

7. Migration between development versions is quite convenient and simple. The framework developers have also created a codemods tool that can automate much of the migration.

8. This is a JavaScript library with accessible code, which receives constant updates and improvements according to the feedback of developers who use it.

Disadvantages of ReactJS:

1.Insufficient amount of official documentation. Active and the dynamic development of ReactJS has led to the lack of supporting documentation, which is currently unstructured as the large number of developers make personal switch to the repository with no necessary analysis and approach.

2. A fairly high level of understanding of ReactJS is required to build the structure of the website application and the process of integrating the user interface into the MVC template.

JSX is a preprocessor that adds XML syntax to JavaScript. We can use React without JSX, but JSX makes React more elegant.

JSX, an XML/HTML-like syntax used in React, extends ECMAScript so that XML/HTML-like text can coexist with JavaScript code and the React framework. The syntax is designed to use a preprocessor (such as Babel) to convert HTML-like text found in JavaScript files into standard JavaScript objects that will parse the JavaScript kernel.

Basically, using JSX, you can write compressed HTML/XML structures (such as DOM-like tree structures) in the same file as the JavaScript code, and then Babel converts those expressions to JavaScript code. Unlike before, instead of placing JavaScript in HTML, JSX allows you to place HTML in JavaScript. You can embed any Java Script expression in JSX by enclosing it in parentheses.

JSX is an expression. As a result of compilation, the JSX syntax is transformed into regular JavaScript objects. This demonstrates that you can also apply it to if or for statements, specify that it is a variable, and take JSX as arguments and get it from functions.

By default, ReactDOM excludes all values written in JSX before rendering them. This tells us that it is not possible to do what is not clearly written in the code. Before rendering, the code is reduced to a regular line. This helps stop Cross Site scripting attacks.

2.2. Markup languages and HTML / CSS

HTML (or hypertext markup language) is a tag syntax used to develop, build, and visualize web applications and their content. For example, information can be divided into a table, menu, title and other parts, depending on the need to convey it to the user.

HTML, using stylization tables and software features are the main tools for building web applications. It allows:

- Create interactive forms;

- Create a formalized and structural component by marking the features of content or text (headings, menus, tables)

- receive information from the Internet through various hyperlinks;

- Embed images, sound, videos, and other objects in text. CSS (Cascading Styling Table) is a special language used to style web applications developed in web page markup languages.

CSS is most often used for documents that are in HTML, XHTML, and XML. The CSS specifications were created by the World Wide Web Consortium, which develops them.

At the present stage, HTML without styles contains a fairly small selection of tools, which does not always allow you to solve problems and meet current requirements for web application design. CSS is used to solve these problems, which can solve the main tasks related to the style of the page.

This separation is intended to improve perception and ease of use of content, remove duplicates, provide control and monitoring of code display in different states, more structuring and simplifying content, etc. CSS allows you to adapt the site to a variety of conditions (monitors, tablets, smartphones, etc.).

Advantages of using CSS:

- display different designs for one site;

- style caching and minimize application load time;

- ease of design updates;

- allows you to create a perfect layout;

- improving ease of use;

- increase the structuring and orderliness of the pages Modern browsers allow you to quickly process styles that improve the UI design of web applications.

CSS frameworks are powerful tools that can help improve development and design processes. Bootstrap, originally called Twitter Blueprint, was created by Mark Otto and Jacob Thornton and was released on August 19, 2011. This is an external interface with an accessible OneSource code consisting of HTML, CSS and JavaScript. It has a modular structure and uses a SassCSS preprocessor. It is also worth noting that it contains not only CSS, but also a JavaScript framework. Bootstrap has ready-made styles and scripts, for which the developer only needs to, create the necessary stylish classes and attributes of the HTML-element.

The Bootstrap framework supports the latest stable versions of major browsers and platforms. For Windows, Internet Explorer 9-11 and Microsoft Edge are supported, but not IE8.

2.3. C # programming language

C # development language is a structured, widely used, statically typed, object-oriented language for writing the web part of a program, the syntax of which is implemented by developers from Microsoft. C # belongs to the family of development and programming languages C, the process of writing code and syntax is quite similar to C, C ++, Java and other languages.

C # has combined the best of its predecessors in its development environment. In particular, C, C ++, Object Pascal, taking into account the real experience of using and preparing programs. Known problem-containing models that have previously been used in programming languages, namely, multiple class assignments (which are actively used in C), have been specifically excluded.

After studying the characteristics of C #, we can conclude that this language is similar to Java-syntax, it is depicted in the syntax and the main concepts and basics of these development languages.

As in the Java backend language, the developers added the following concepts to the studied C #:

- Virtual machine – a dynamic framework or other basis, such as .Net, launches and executes the developed program;
- Byte code – the prepared code is compiled into a temporary language MSIL ("Microsoft Intermediate Language"), and only later transformed into a machine language in full dependence on the platforms on which the program will run;
- Managed code – given that the developed programs run only in the virtual environment CLR or "Common Language Runtime", there is a possibility for control and control the execution of programs and available mechanisms for stopping and shutting them down, as well as regulate the load and memory usage of the developed software, if necessary – to scale or delete selected areas of memory used by programs.

The C # language is built using component architecture and provides protection for written code. The main features of the language include the following:

- The approach is focused on the division into components;
- Delegates – data encapsulation, which is typical for OOP;
- Indexers for access to components;
- Operators;

C # also inherited the idea of program blocks from C.

C # is considered an object-oriented language The C # programming language is "native" for developing applications in a framework environment. NET because it is tightly and efficiently integrated with it.

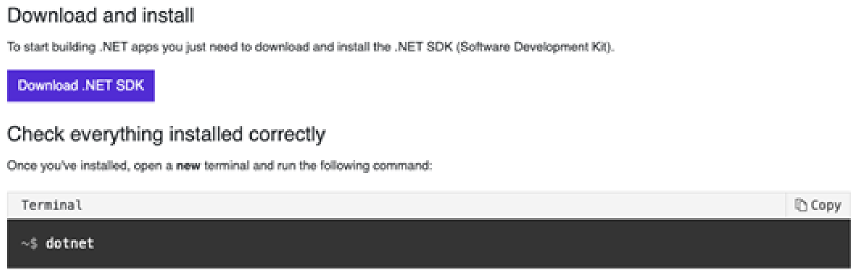
2.4. Development of the server part and creation of a database for the program

The purpose of this work is to create a system for data analysis and visualization. Since the end of the 2010s, there has been a clear trend in the growing popularity of the client-server application approach, with the basic logic of the analysis system focusing on the front-end, while the back-end is responsible for storing information and issuing actually structured but not analyzed data.

When creating a backend part of a website, the first step is to install the necessary software and development environments:

Visual Studio 2017 (with selected when installing .NET Core croses platforms development). We will write the server on .net core 2.0, that's why we need the development environment version> = 2017, because in previous versions the maximum supported version is 1.1 [8].

Then install the SDK (Software Development Kit) .NET, as shown in Figure 2.1.



Download and install

To start building .NET apps you just need to download and install the .NET SDK (Software Development Kit).

Check everything installed correctly

Once you've installed, open a **new** terminal and run the following command:

```
~$ dotnet
```

Figure 2.1 – Installing the .NET SDK

The Microsoft .NET Framework consists of a large number of tools and technologies for developing backend software. The Microsoft .NET Framework is based on the C # programming language [8].

Next, install MS SQL Server to deploy the database (Figure 2.2)

SQL Server 2017 Express edition

Build small, data-driven web and mobile applications up to 10 GB in size with this free, entry-level database.

Figure 2.2 – Installing SQL Server

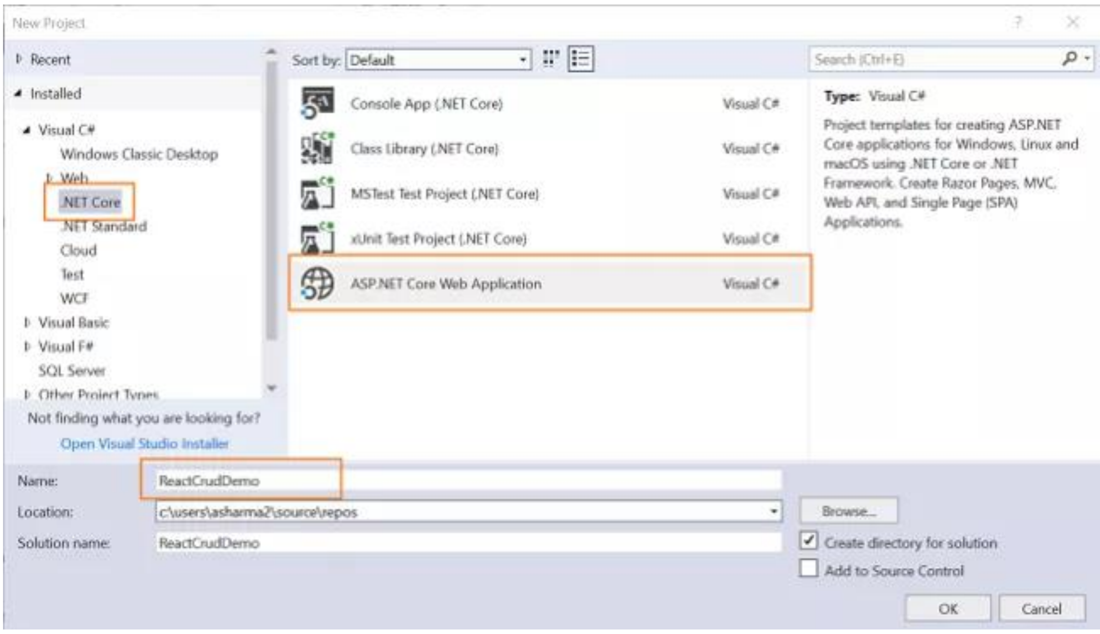
Microsoft SQL Server is a database management application system developed and distributed by Microsoft. Transact-SQL is used to create queries. It is possible use for a variety of (small, medium and large), so it fully meets the requirements of the database of the day of our application.

Open Visual Studio 2017 and select File next New post of this Project. After selecting a project, the New Project dialog box opens. Select .NET Core in the Visual C # menu in the left pane.

Then select "ASP.NET Core Web Application" from the available project types, as shown in Figure 2.3.

The next stage is to choose the template according to which the project was created. From the drop-down menu, select ASP.NET Core version 2.0 and select React.js as the main template. Next, we add data models based on open free climate data and models for user logins. After that, in Visual Studio, open the Package Manager Console and enter the command: enable-migrations.

After executing this command, the Migrations folder will be created in the project, which contains the Configuration’s file. This file contains declarations of the Configuration class of the same name, which sets the migration configuration settings.


New Project
Recent
Installed
Visual C#
Windows Classic Desktop
Web
.NET Core
.NET Standard
Cloud
Test
WCF
Visual Basic
Visual F#
SQL Server
Other Project Types
Not finding what you are looking for?
Open Visual Studio Installer
Sort by: Default
Search (Ctrl+E)
Console App (.NET Core) Visual C#
Class Library (.NET Core) Visual C#
MSTest Test Project (.NET Core) Visual C#
xUnit Test Project (.NET Core) Visual C#
ASP.NET Core Web Application Visual C#
Browse...
OK
Cancel

Type: Visual C#

Project templates for creating ASP.NET Core applications for Windows, Linux and macOS using .NET Core or .NET Framework. Create Razor Pages, MVC, Web API, and Single Page (SPA) Applications.

Name: ReactCrudDemo

Location: c:\users\asharma2\source\repos

Solution name: ReactCrudDemo

☑ Create directory for solution

□ Add to Source Control

Figure 2.3 – Creating ASP.NET Web Applications

In the Seed method, we initialize the database with initial data. Now we need to create the migration itself.

To do this, enter the command: PM in the Package Manager Console followed by the Add-Migration and MigrateDB commands.

Visual Studio will then automatically generate the migration class. In the Up method, using the Create Table method, the "dbo.Users" table is created and configured: creating columns, setting keys.

The Down method removes columns and tables in case they exist. To perform the migration, apply this class by typing the command in the console: PM, followed by Update-Database. After that, the database structure shown in Figure 2.4.

The next step in developing a backend part of the system for analyzing and visualizing climate data is to add the Web API Controller to our application. To do this, open the Controllers folder and perform the following sequence of actions: Add and add New Item.

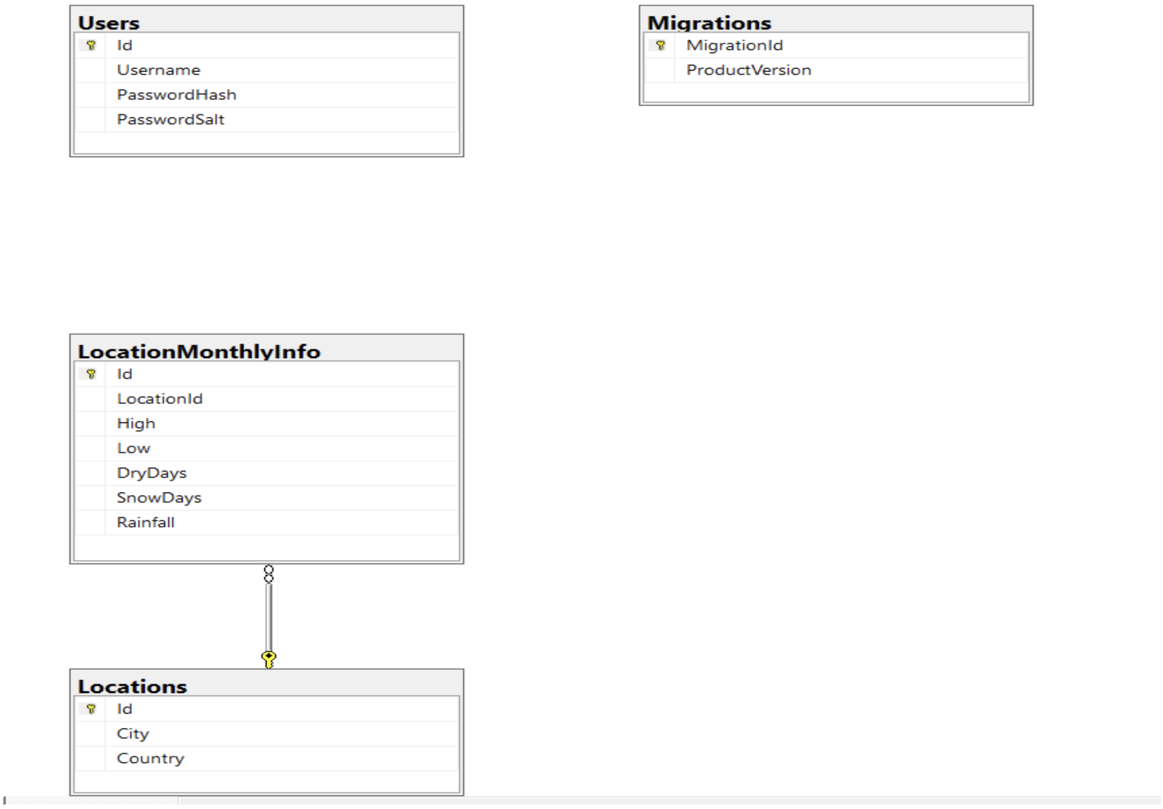
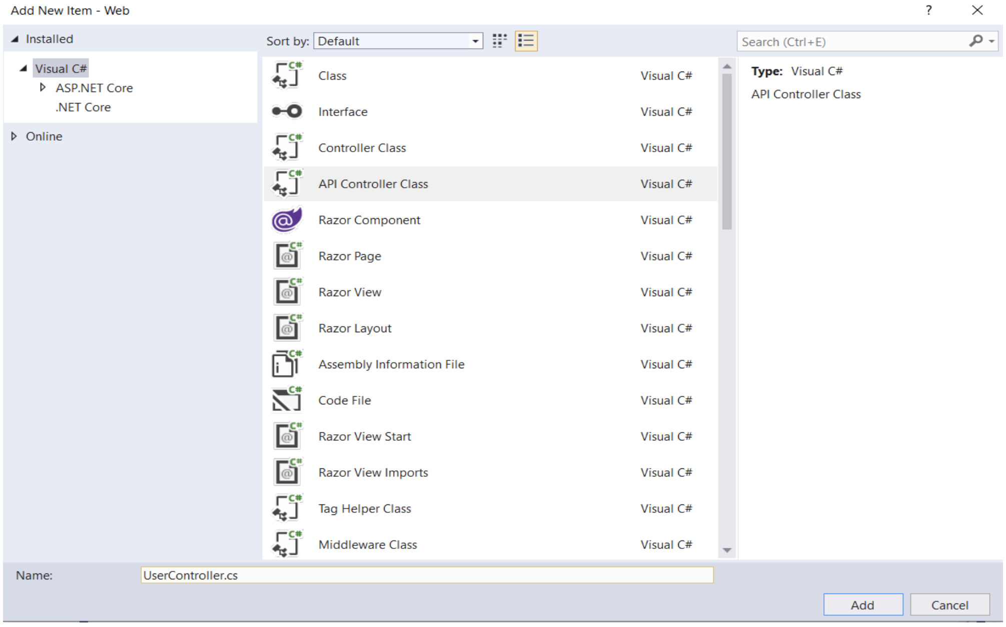


Figure 2.4 – Database structure

Select ASP.NET from the left panel menu, then select the Controllers Class API and give it the name User Controller, as shown in Figure 2.5.



Sort by: Default

Class	Visual C#
Interface	Visual C#
Controller Class	Visual C#
API Controller Class	Visual C#
Razor Component	Visual C#
Razor Page	Visual C#
Razor View	Visual C#
Razor Layout	Visual C#
Assembly Information File	Visual C#
Code File	Visual C#
Razor View Start	Visual C#
Razor View Imports	Visual C#
Tag Helper Class	Visual C#
Middleware Class	Visual C#

Type: Visual C#

API Controller Class

Name: UserController.cs

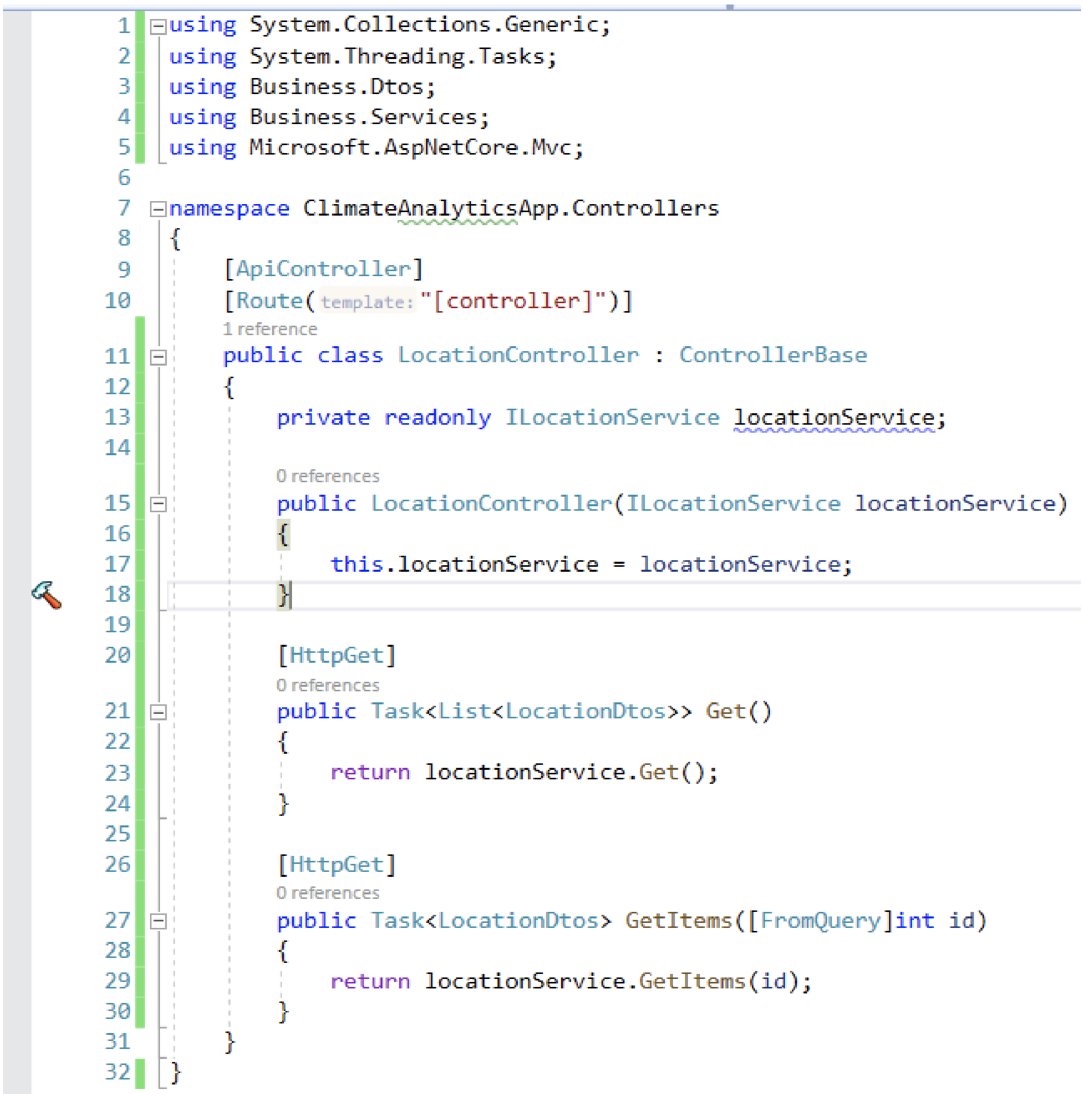
Figure2.5 – Creating a controller Usser

Controller

This controller will have a POST method for user registration.

In the AuthController.cs controller, we add the logic of user logging in to our web application.

We also create a controller to obtain data sets for data analysis and visualization, where we add two methods: the first [HttpGet] to obtain the entire data set (for all available locations) and the second [HttpGet], which takes the query string parameter with LOCATION ID of the selected geographic area. object with nested monthly data sets (Figure 2.6).

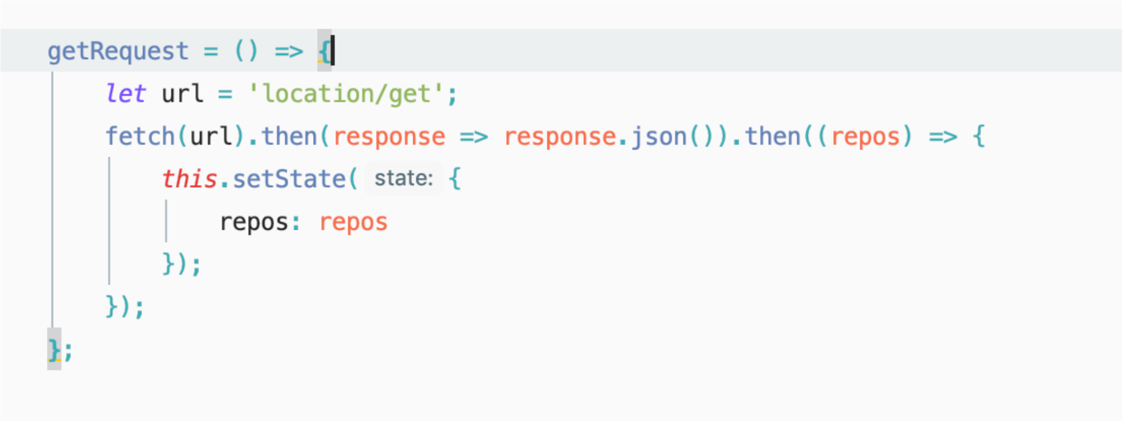


```csharp
using System.Collections.Generic;
using System.Threading.Tasks;
using Business.Dtos;
using Business.Services;
using Microsoft.AspNetCore.Mvc;

namespace ClimateAnalyticsApp.Controllers
{
    [ApiController]
    [Route("[controller]")]
    public class LocationController : ControllerBase
    {
        private readonly ILocationService locationService;

        public LocationController(ILocationService locationService)
        {
            this.locationService = locationService;
        }

        [HttpGet]
        public Task<List<LocationDtos>> Get()
        {
            return locationService.Get();
        }

        [HttpGet]
        public Task<LocationDtos> GetItems([FromQuery]int id)
        {
            return locationService.GetItems(id);
        }
    }
}
```

Figure 2.6 – API methods for obtaining data

In the next iteration of program development when preparing the client part, we need to form a JavaScript file called Service.js, using the fetch()method. The Fetch API is a JavaScript interface for working with HTTP requests and responses. The fetch () method allows you to easily and logically asynchronously receive materials over the network.

Similar functionality has previously been achieved with XMLHttpRequest protefetch () is a much more powerful tool. Fetch () returns a Promise object, which when received a positive response from the server part performs a given function in which the response object is passed as an argument. To obtain the entire data set according to the locations, you need to add the appropriate method, as shown in Figure2.7.



```
getRequest = () => {
    let url = 'location/get';
    fetch(url).then(response => response.json()).then((repos) => {
        this.setState( state: {
            repos: repos
        });
    });
};
```

Figure 2.7 – Getting data using the fetch method

2.5 Conclusions to the section

This section analyzes the programming languages used in the development process for data analysis and visualization, and selects modern development tools: Microsoft SQL Server for database deployment, Microsoft .NET framework with C # development language for the server part of the application, REST API for communication between the server and client parts of the application. The ReactJS framework with the JavaScript programming language is used to write the client part of the website. We use CSS and the React Bootstrap framework for the visual design of the web client.

The above technologies will eventually make it possible to create an effective high-speed tool for visualization of data sets for the system of analysis and visualization of climatic indicators, which ensures the achievement of the goals, set in the work.

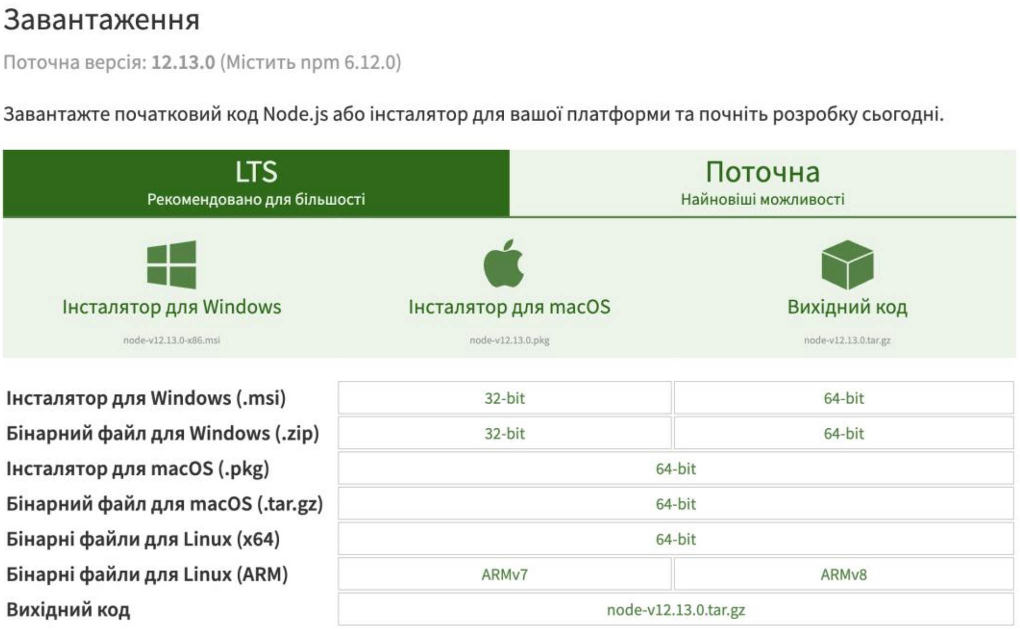
The server part of the computer system for analysis and visualization of climate data is also developed in this section. The use of MS SQL Server 2017 for database deployment is investigated. For implementation of user logging functionality, studied and implemented REST API for interaction with the client part of the program.

All of the above provided good scaling and interoperability of system components, common software interfaces, which in turn will improve system performance and security.

3 DEVELOPMENT OF THE CLIENT PART OF THE SYSTEM OF ANALYSIS AND VISUALIZATION OF CLIMATE DATA

3.1 Create client part form

For frontend development we need to set Node version> = 8.10 and npm version> = 5.6, as shown in Figure 3.1.



Завантаження

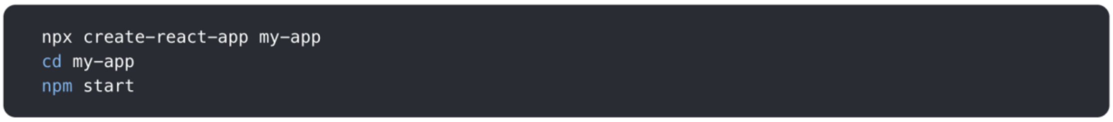
Поточна версія: **12.13.0** (Містить npm 6.12.0)

Завантажте початковий код Node.js або інсталятор для вашої платформи та почніть розробку сьогодні.

Інсталятор для Windows (.msi)	32-bit	64-bit
Бінарний файл для Windows (.zip)	32-bit	64-bit
Інсталятор для macOS (.pkg)	64-bit	
Бінарний файл для macOS (.tar.gz)	64-bit	
Бінарні файли для Linux (x64)	64-bit	
Бінарні файли для Linux (ARM)	ARMv7	ARMv8
Вихідний код	node-v12.13.0.tar.gz	

Figure 3.1 – Installing Node.js

After making the above installation in the selected folder, call the Terminal of our operating system. To create a project, we perform initialization commands (Figure 3.2).

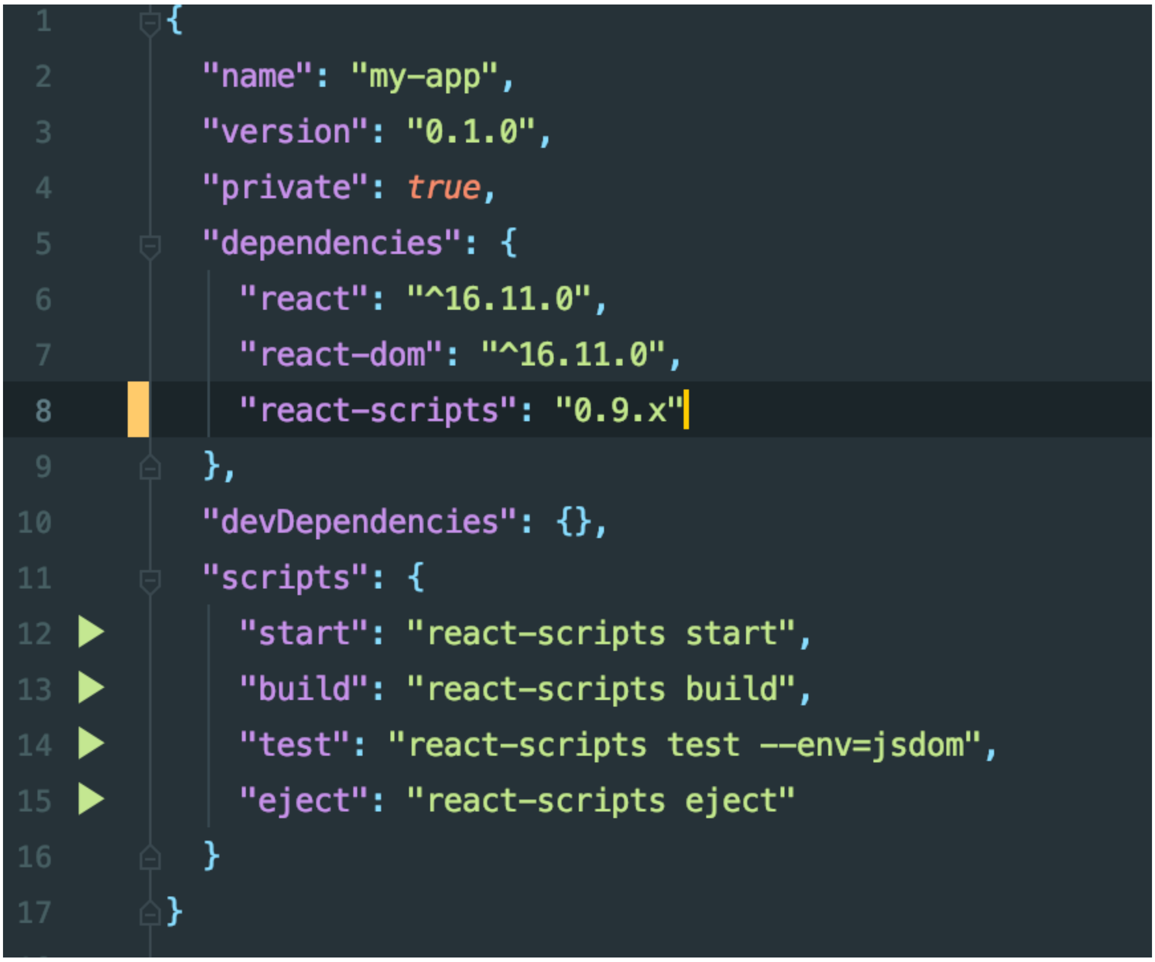


```
npx create-react-app my-app
cd my-app
npm start
```

Figure 3.2 – Teams to create a project

Added the basis of the client part of the web application, created using the Javascript framework React, but this client does not process the logic of the backend and databases, but only provides commands to build the frontend, so it will be used with the previously created backend [8].

When studying the created automatically generated package.json file, we will have access to the structure of program packages, as shown in Figure 3.3.



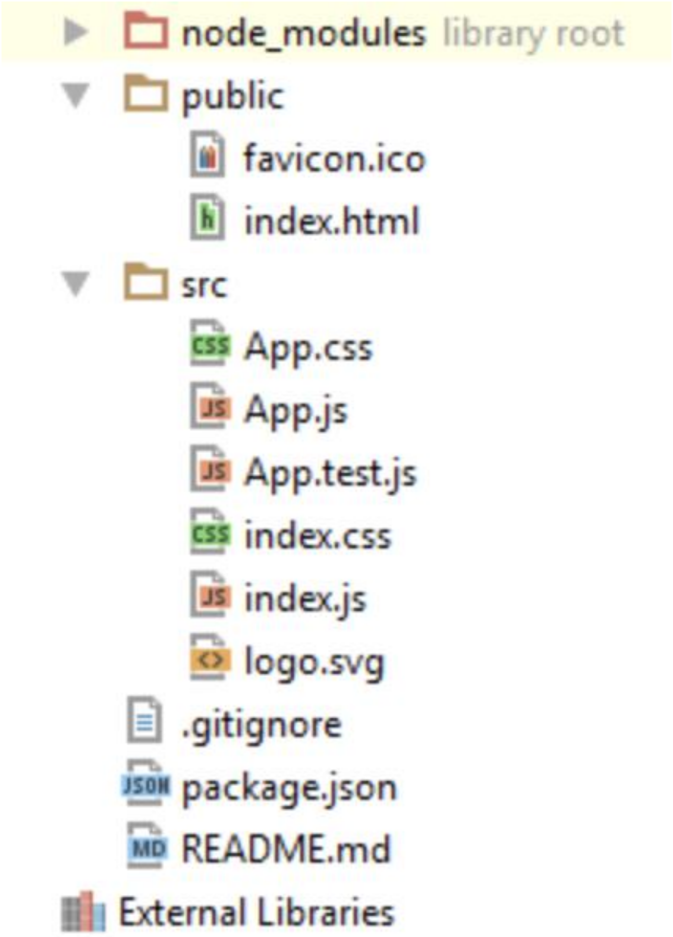
```
{
  "name": "my-app",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
    "react": "^16.11.0",
    "react-dom": "^16.11.0",
    "react-scripts": "0.9.x"
  },
  "devDependencies": {},
  "scripts": {
    "start": "react-scripts start",
    "build": "react-scripts build",
    "test": "react-scripts test --env=jsdom",
    "eject": "react-scripts eject"
  }
}
```

Figure 3.3 – Structure file package.json

This JSON file in the dependencies object lists the dependencies that are required for our application to function and their version.

The scripts object is our declared basic npm commands needed in the website development process.

Also in our IDEA the received structure of the project is specified in Figure 3.4.



```
▶ node_modules library root
▼ public
    favicon.ico
    index.html
▼ src
    App.css
    App.js
    App.test.js
    index.css
    index.js
    logo.svg
  .gitignore
  package.json
  README.md
External Libraries
```

Figure – 3.4 Figure Project structure

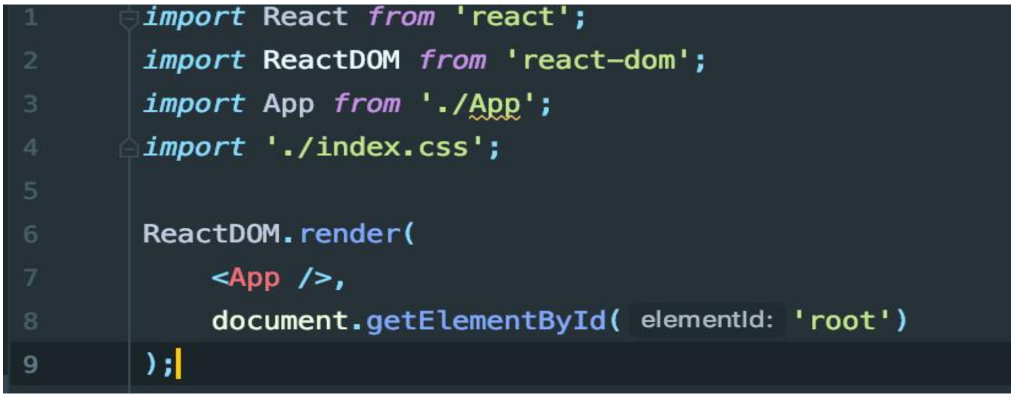
Node modules / – this folder contains all the dependencies of the project, specified in the file package.json and which are installed with the command npm install.

Public / – the contents of this folder are what you need to render to the web application page: public / index.html – the website template and favicon.ico – the favicon that will be displayed on the browser tab.

Src / – here are all the source files with which there will be direct interaction and changes.

Src / index.js is the entry point of the created project.

After the initialization of the project, this file will look like this, as shown in Figure 3.5.



```
import React from 'react';
import ReactDOM from 'react-dom';
import App from './App';
import './index.css';

ReactDOM.render(
    <App />,
    document.getElementById( elementId: 'root')
);
```

Figure 3.5 – File view after initialization

Import module / React / from 'react' and import / ReactDOM / module from 'react-dom' is a connection of React and ReactDOM modules from previously installed node-modules.

Then the App.js component is connected, the import App (module) from './App' is used.

The import './index.css' ribbon connects the CSS file with the basic project styles.

Part of the code:

«ReactDOM.render (

<App />,

document.getElementById ('root')

); »

Contains several parts:

Render () is a method of the ReactDOM module that takes two arguments: the component to be rendered and the HTML "wrapper" in which to place the result. In our project, an element with id = "root", which was specified in public / index.html.

<App /> is an App.js component, an ad using JSX – (or JavaScript Syntaxyses Extensions). This is an HTML-like syntax for creating reactive components.

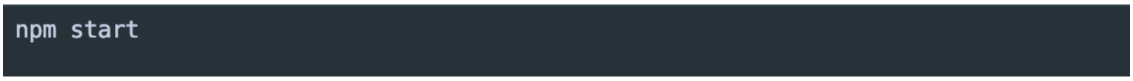
Its syntax is similar to regular HTML, but there are a few differences that have been taken into account when working on the project:

- 1 component is 1 node, ie all component markup must be placed within one tag;

- Odd tags in JSX always need to be closed;

- Instead of the class attribute in JSX className [8] is used.

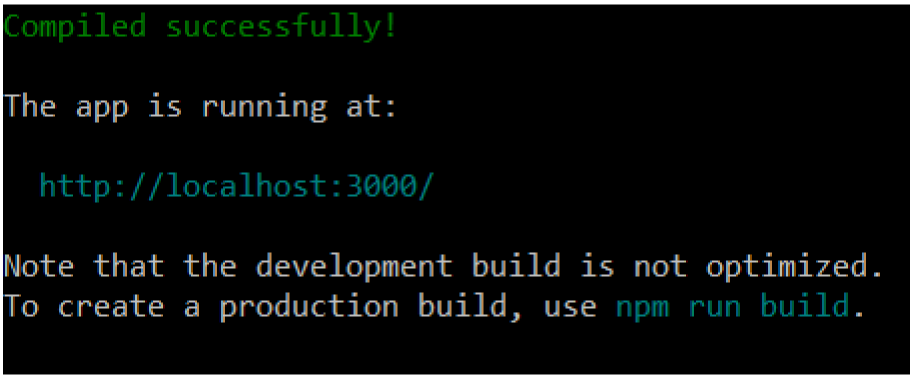
To check the correctness of the installation of basic packages and configure our system for analysis and visualization of data in the project folder in the terminal, execute the command, as shown in Figure 3.6.



```
npm start
```

Figure 3.6 – Application launch command

As a result, we obtain the following conclusion (Figure 3.7):



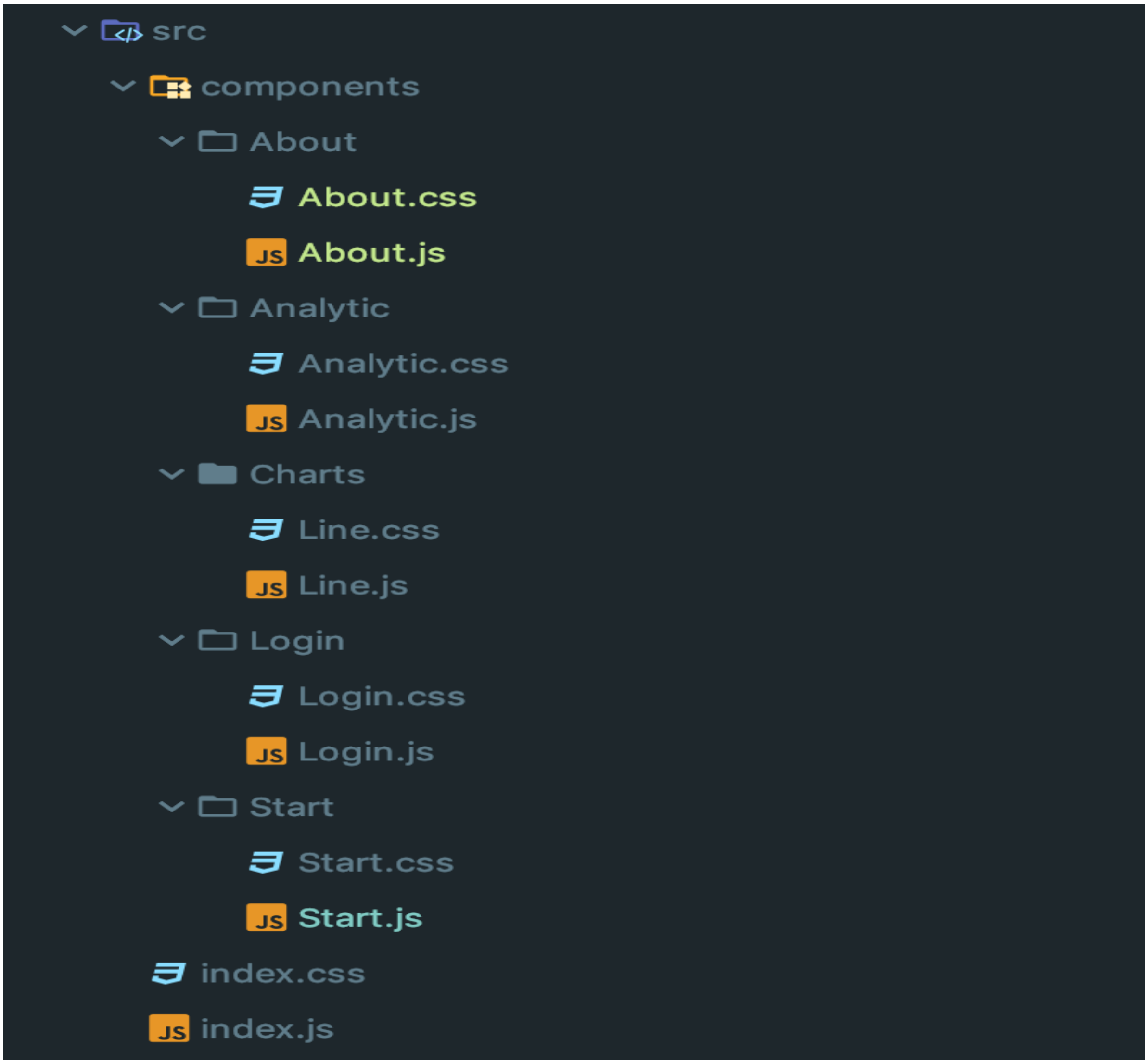
```
Compiled successfully!

The app is running at:

  http://localhost:3000/

Note that the development build is not optimized.
To create a production build, use npm run build.
```

Figure 3.7 – Notification of successful start of the program

The project will also be launched in a browser at http://localhost:3000.

Livereload works in the development environment, i.e. all changes are displayed on the page without reloading it.

The next stage of development of the client part of the system of analysis and visualization of climate data is the creation of JavaScript-components and CSS-files of stylization in accordance with the tasks set before our web application (Figure 3.8).

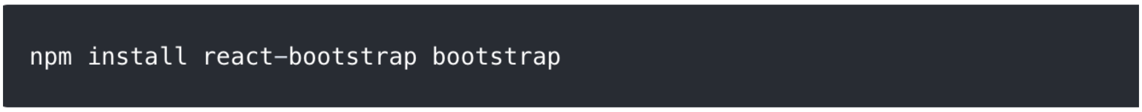


```
src
  components
    About
      About.css
      About.js
    Analytic
      Analytic.css
      Analytic.js
    Charts
      Line.css
      Line.js
    Login
      Login.css
      Login.js
    Start
      Start.css
      Start.js
  index.css
  index.js
```

Figure 3.8 – Created components

The Start folder contains the React component, which will be displayed on the main page of our site. The Login component will be responsible for the client part of the user login process in the program. Analytic files focus on most of the logical operations for analyzing and visualizing climate data. The Charts folder contains code for customizing and configuring selected graphs to display processed data sets. About is a component that will contain the form of feedback of users of the program with the developers and general information about the author of the web application.

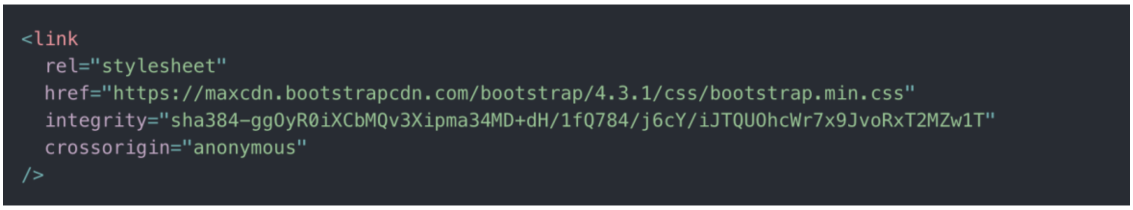
The next step in developing a system for analyzing and visualizing climate data is to connect the React Bootstrap version 4.3. Bootstrap is a framework for styling dynamic web applications. To do this, in the terminal, execute the following command to add the module to the project, as shown in Figure 3.9.



```
npm install react-bootstrap bootstrap
```

Figure 3.9 – Installing the React Bootstrap module

The next step is to add the import 'bootstrap / source / dists / css / bootstrap.min.scss' tape to the src / index.js file so that the functionality of the connected client framework is available in our project. Then add the exterior styles from React Bootstrap, as seen in Figure 3.10.



```
<link
  rel="stylesheet"
  href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
  integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
  crossorigin="anonymous"
/>
```

Figure 3 – 10.Adding external styles

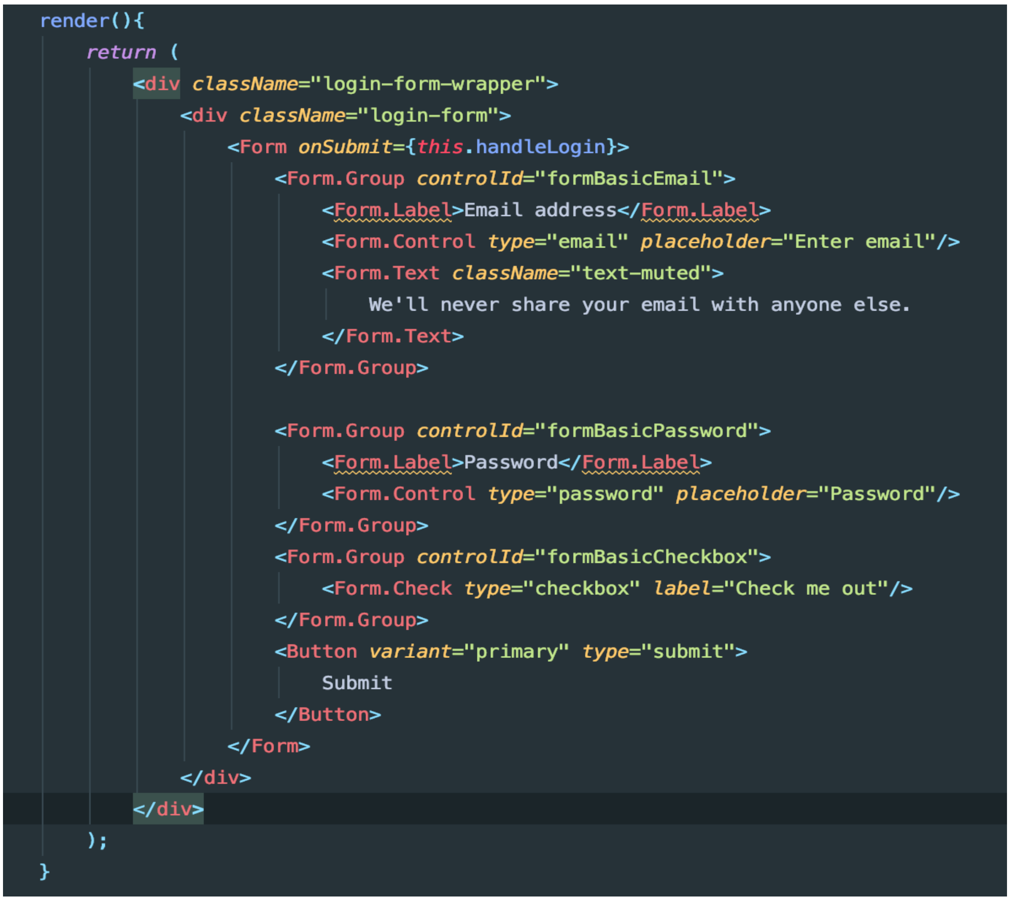
The best way to connect the latest versions of styles to a project is to use a CDN (Content delivery network) as a source of resource files, which receives regular official updates from the developers.

Upon completion of this stage, the result was a locally deployed baseline project at http://localhost:3000 with installed dependencies for further development, including the React framework version 16.10.2 and React Bootstrap version 4.3.

The next stage in the development of a web application for climate data analysis and visualization is the layout of client components and the writing of analysis logic for the data sets obtained from the backend.

When designing the start page, we use the previously connected Bootstrap client framework, in particular, we import the Button component from the 'react-bootstrap / Button' module and add it to the page using JSX syntax and style this part of the website using CSS in the appropriate file. .

Next, we create web markup for our login form for program users, as shown in Figure 3.11.

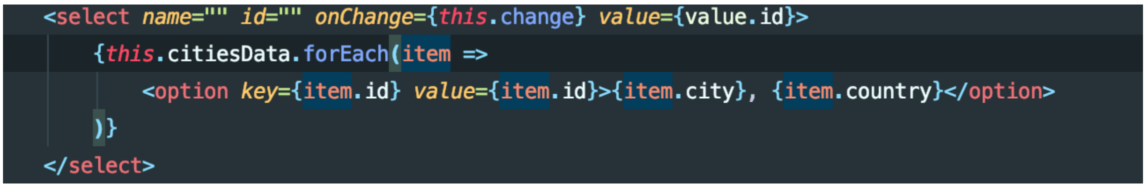


```
render(){
    return (
        <div className="login-form-wrapper">
            <div className="login-form">
                <Form onSubmit={this.handleLogin}>
                    <Form.Group controlId="formBasicEmail">
                        <Form.Label>Email address</Form.Label>
                        <Form.Control type="email" placeholder="Enter email"/>
                        <Form.Text className="text-muted">
                            We'll never share your email with anyone else.
                        </Form.Text>
                    </Form.Group>

                    <Form.Group controlId="formBasicPassword">
                        <Form.Label>Password</Form.Label>
                        <Form.Control type="password" placeholder="Password"/>
                    </Form.Group>
                    <Form.Group controlId="formBasicCheckbox">
                        <Form.Check type="checkbox" label="Check me out"/>
                    </Form.Group>
                    <Button variant="primary" type="submit">
                        Submit
                    </Button>
                </Form>
            </div>
        </div>
    );
}
```

Figure 3.11 – Created web markup

As you can see in the figure above, in addition to the basic tags, the Bootstrap components are also imported – import {Button, Form} from "react-bootstrap".

In the Analytic.js UI component (user interface) part of the client program will be represented by elements that allow users to interact with the logic of the website, in particular select (Figure 3.12).



```
<select name="" id="" onChange={this.change} value={value.id}>
    {this.citiesData.forEach(item =>
        <option key={item.id} value={item.id}>{item.city}, {item.country}</option>
    )}
</select>
```

Figure 3.12 – Elementselect

To select a specific set of information for analysis and display, we also use the Switcher and Checkbox elements, which will allow us to display information after filtering and sorting by selected indicators.

The final component of our climate data analysis and visualization system is the creation of a feedback form in the about component, which is reproduced in Figure 3.13

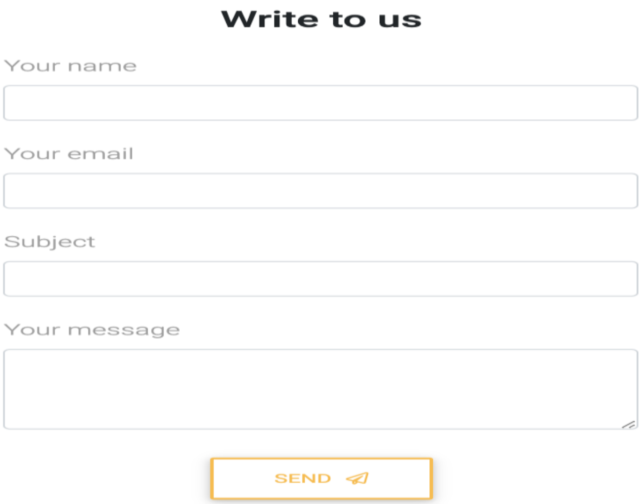


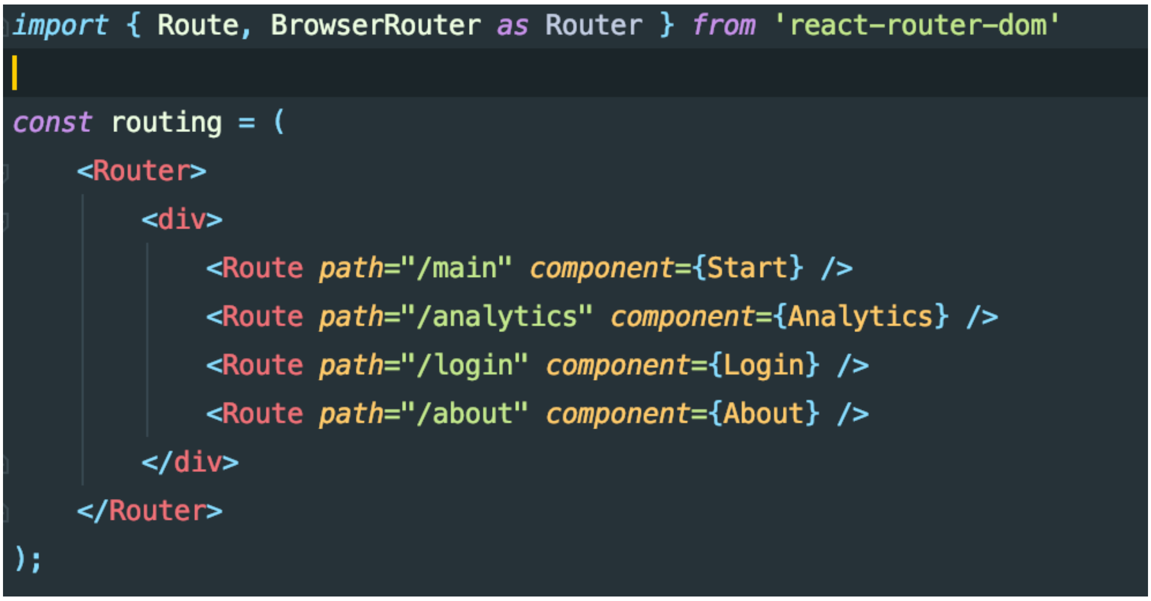
Figure 3.13 – Feedback form

So, at this stage of program development, we performed the layout of the main components of the client part of the program and performed the tasks set before us, obtaining a user-friendly and clear user interface.

In the next stage of development of our system of analysis and visualization of climate data, we add a logical part of the operations that will be performed on the client part of the web application.

In the React framework, there are two approaches to declaring components – functional and class. In our project, we use the class method of declaration, because it is based on the latest version of JavaScript EcmaScript 6, which will allow us to have advanced capabilities when writing code.

To implement dynamic interaction with the user, we first add the ability to navigate our website. To do this, use the module react-router-dom. After adding it to the package.json file, run the npm install command and import it into our main index.js file and configure the router according to the structure of our project (Figure 3.14).



```
import { Route, BrowserRouter as Router } from 'react-router-dom'

const routing = (
    <Router>
        <div>
            <Route path="/main" component={Start} />
            <Route path="/analytics" component={Analytics} />
            <Route path="/login" component={Login} />
            <Route path="/about" component={About} />
        </div>
    </Router>
);
```

Figure 3.14 – Structure of our project

Then we add the handle Click method start the page on the “SIGNIN” button to move to the log in page.

The following methods have been supplemented to the login page:

handleEmailChange- for processing the entered data in the textbox of the email of the user;

handlePasswordChange- is responsible for collecting and processing the log in password;

handleLogin-generates an object from the collected data or information and sends it to the server and incase of a successful login response redirects the user to a page where they can work with analytics.

The next step is to supplement some methods to the analytics component. To analyze the climate data received from the server, which we receive in the form of arrays, we can use the methods available in [removed]

methodarr.find is suitable for us to search for information on a specific unique identifier, its syntax is as follows:

```
let result = arr.find (function(item, index, array) {
//if true – brings back the current component of the array and it then stops the search
//if on all iterations the result is false, returns undefined
});
```

The find method looks for one (first counter) component on which the call back function returns true.

In the case whereby there are more than one element found, we will be using the arr.filter (fn) method (filtering by a certain selected metric).

The syntax of this method is almost the same to find, but filter returns an array of all the elements that satisfies the condition specified in the method.

For iterative search and processing of each following element of an array of climatic historical data we use methods available in the chosen programming language:

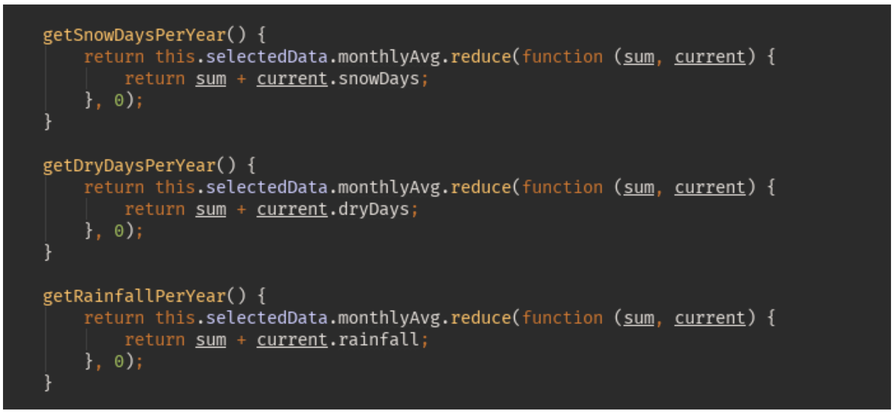
-forEach – this JavaScript method "arr.forEach (callback () (current, thisArg[)") is used for gradual processing of the array. This method foreach iterated component of the array will call the passed call back function:

-Map – the method "arr.map (callback((current, thisArg]))" is used to modify the array. It creates a new instance of the array, which will consist of the results of calling the callback function with the arguments item(I and array)for each element of the climate data array.

-Every – the tool for working with the array "arr.every (callback () (current, thisArg[)" gives the result true if using the callback function returns true for each component of our data array.

-some – the pointer method uses the syntax "arr.some (callback[current, thisArg])", which returns true if the callback function returns true for atleast one element of the array under study.

-Reduce – the function "arr.reduce (callback[, initial Value])" we can use it for gradual processing of all the elements of the array with memorization of temporary computing result. It takes a callback function by iterating to each unit of the array, while maintaining the temporary results of the computing. The callback function includes the following arguments: previousValue, current, indexElement and the actual processed array. In addition to callback to this method, you can pass the argument initialValues [5], as it is shown in Figure3.15.



```
getSnowDaysPerYear() {
    return this.selectedData.monthlyAvg.reduce(function (sum, current) {
        return sum + current.snowDays;
    }, 0);
}

getDryDaysPerYear() {
    return this.selectedData.monthlyAvg.reduce(function (sum, current) {
        return sum + current.dryDays;
    }, 0);
}

getRainfallPerYear() {
    return this.selectedData.monthlyAvg.reduce(function (sum, current) {
        return sum + current.rainfall;
    }, 0);
}
```

Figure 3.15 – Data processing using JavaScript-methods of working with arrays

Thus, a web application was obtained in which the user after the identification process gets access to tools for analyzing historical climate data and gets the output in the form of textual information. Therefore, the next stage of development is to add visualization capabilities.

3.2. Connect ApexCharts and create charts to visualize data

To visualize the data of our climate data analysis system we use ApexCharts – a modern open source JavaScript library for building a graphical data image based on SVG (Scalable Vector Graphics) graphics, the main page of the website is shown in Figure 3.16.

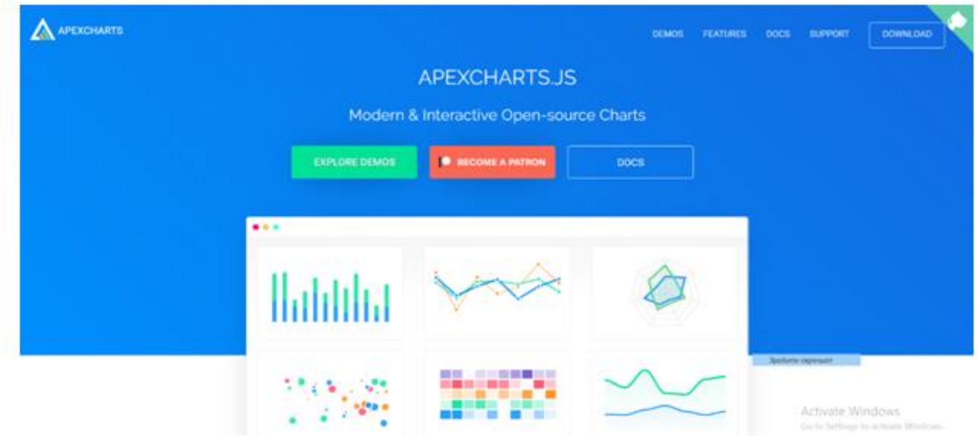
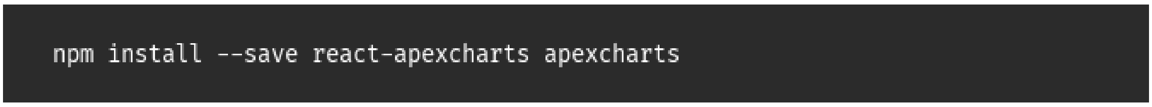


Figure 3.16 – ApexCharts home page

It is free to use and is distributed with a MIT license. A big advantage when choosing ApexCharts was the access to many types of graphs for data visualization and the ability to combine and customize them for user convenience and specific tasks.

First, add the module "apexcharts": "^ 3.10.1" and "react-apexcharts": "^ 1.3.3" to the package.json file and execute the appropriate command in the terminal, as shown in Figure 3.17.



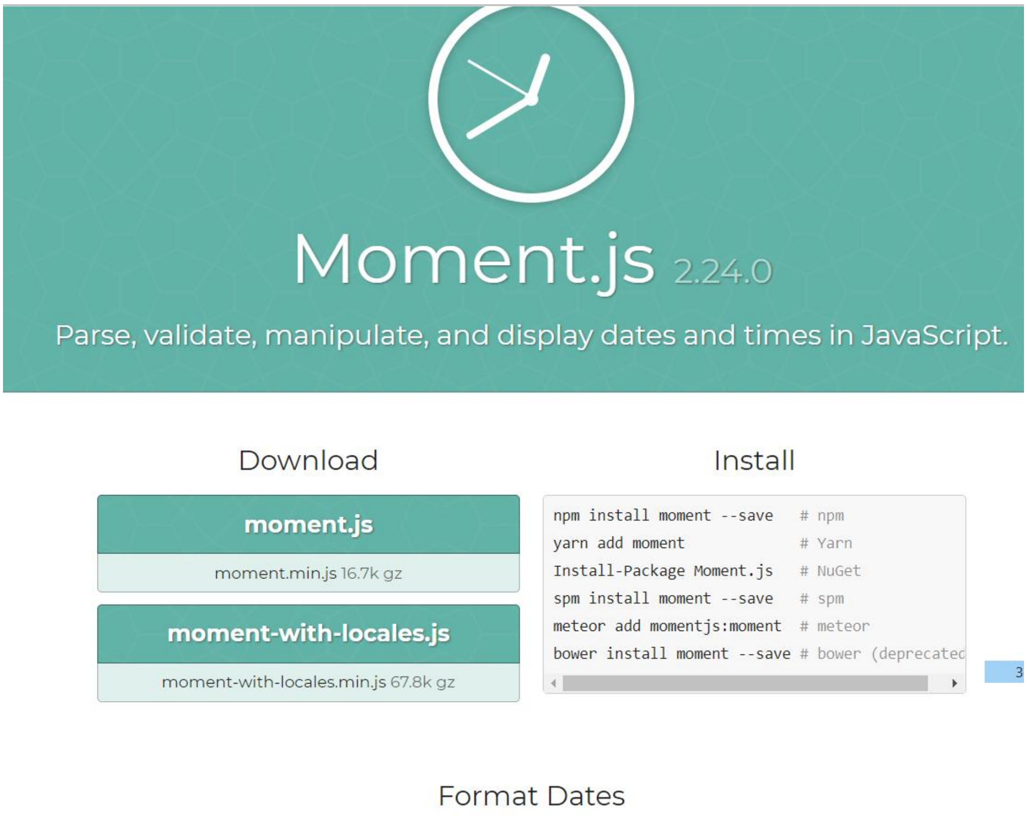
```
npm install --save react-apexcharts apexcharts
```

Figure 3.17 – Installation of ApexCharts packages

After downloading and installing the modules, we import the module into our component by adding the Chart component from the "react-apexcharts" library to the import ribbon.

React-ApexCharts is a wrapper for the ApexCharts component that allows you to connect it as a react.js module.

To unify and customize calendar dates, download and connect the "moment.js" library. In Figure 3.18 shows the main page of the library.



Parse, validate, manipulate, and display dates and times in JavaScript.

```
npm install moment --save    # npm
yarn add moment              # Yarn
Install-Package Moment.js    # NuGet
spm install moment --save    # spm
meteor add momentjs:moment   # meteor
bower install moment --save # bower (deprecatec
```

Figure 3.18 – Home page of the moment.js library

This library is a shell for the standard JavaScript object – Date. It allows you to process dates and times and performs arithmetic operations and comparisons with them [6].

Using the official documentation and the given demonstration examples of diagrams, available on the developer's website at apexcharts.com/docs, we start creating graphs to perform the tasks of visualization of the data set obtained from the server part of the project.

To display the data set of maximum and minimum temperatures, we use a linear type of graph with two data sets and labels on the lines. In Figure Figure 3.19 shows the visualization of an array of data that demonstrates the monthly maximum and minimum values of air temperature for the selected location [9]. This type of chart has a high ability to adjust (display only the highest or lowest values) and customization (change the color scheme, font size, animation speed, etc.).

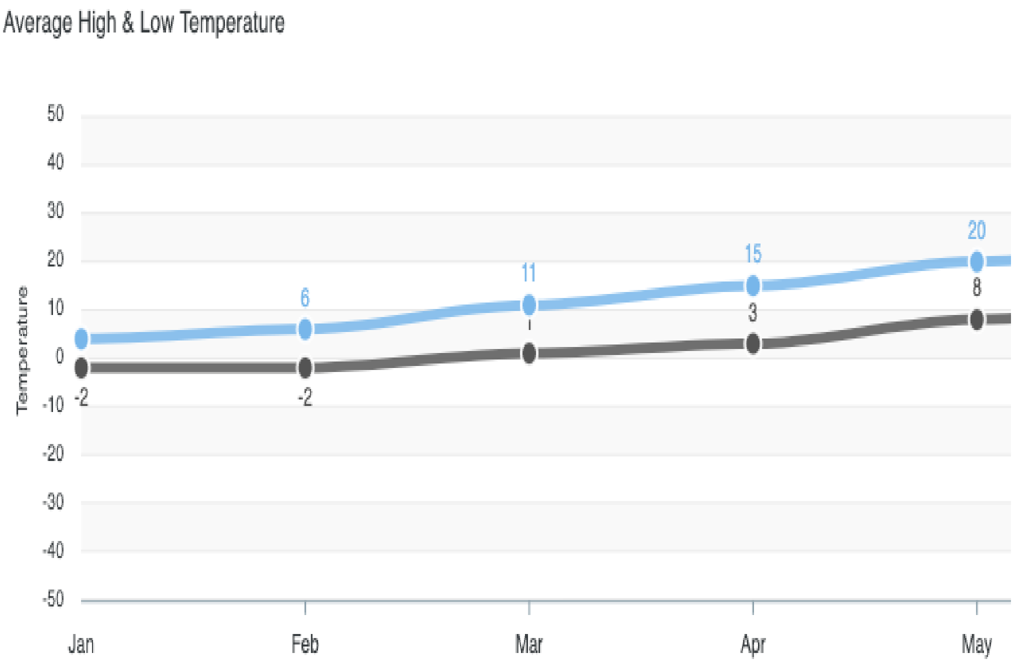
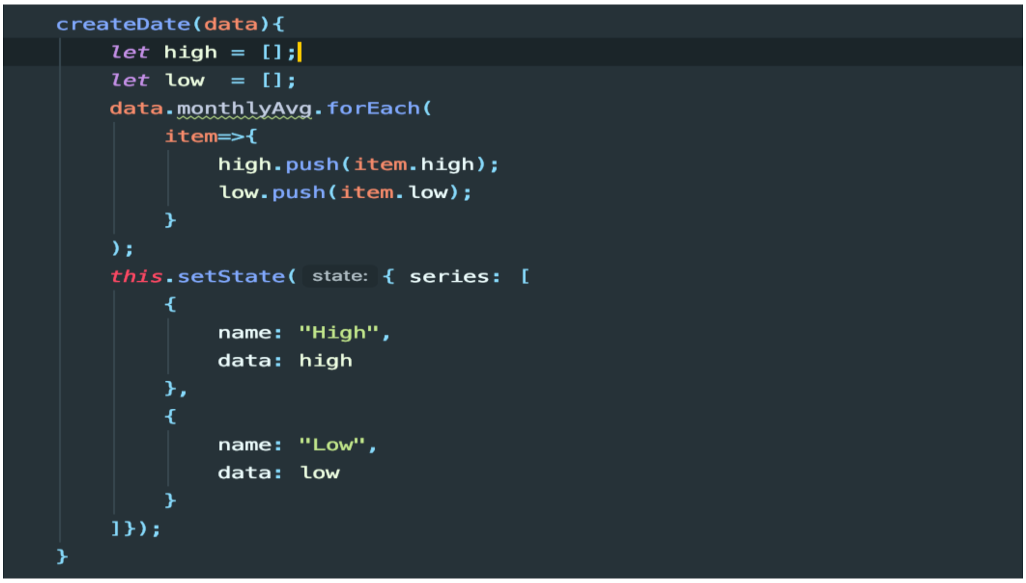


Figure 3.19 – Graph for visualization of temperature values

To process data sets for the given graph, we add a new method, as can be seen from Figure 3.20.



```
createDate(data){
    let high = [];
    let low  = [];
    data.monthlyAvg.forEach(
        item=>{
            high.push(item.high);
            low.push(item.low);
        }
    );
    this.setState( state: { series: [
        {
            name: "High",
            data: high
        },
        {
            name: "Low",
            data: low
        }
    ]});
}
```

Figure 3.20 – Method for processing the analyzed data

To visualize the amount of precipitation, we create a bar graph that shows the trends of seasonality in the amount of precipitation for different control geographical points. In particular, in Figure 3.21column graph shows the annual distribution of precipitation in the selected geographical area.

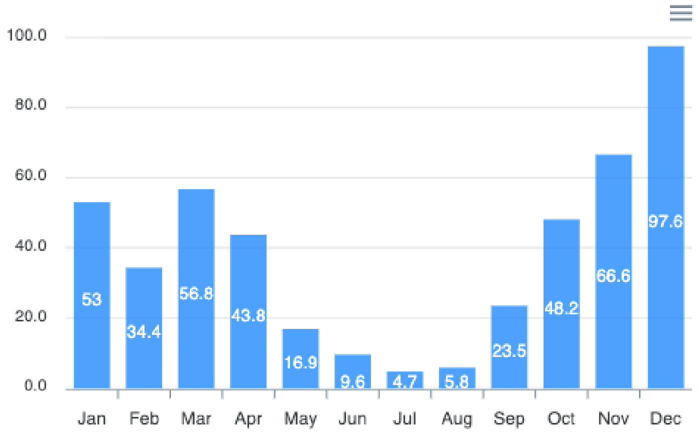


Figure 3.21 – Column graph of precipitation

To demonstrate the ratio of annual precipitation in the form of snow and rain, we use a grouped horizontal bar graph, as in Figure 3.22.

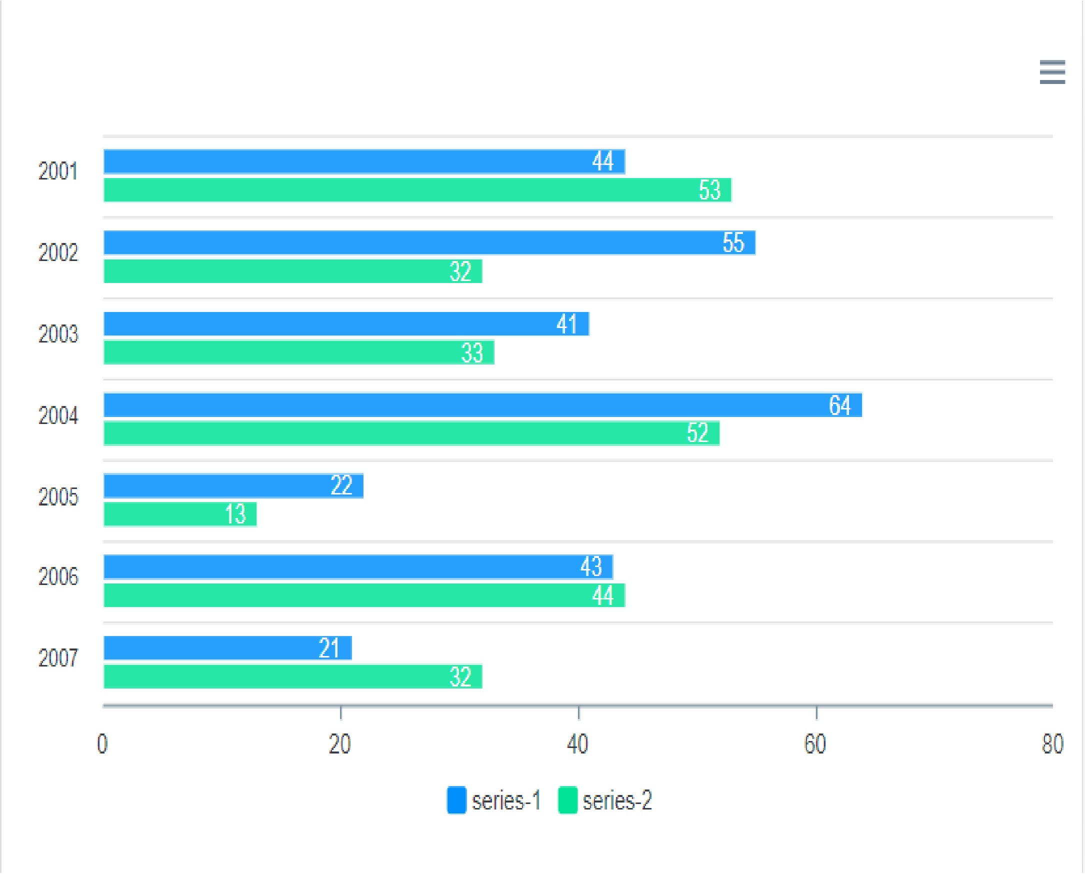


Figure 3.22 – The ratio of annual precipitation

To trace the general trend of the temperature level by geographical months, we create a diagram (Figure 3.23), which shows the temperature variance according to geographical locations.

This diagram shows the climatic distribution of the climatic indicator – the average monthly air temperature in many geographical objects, which allows us to draw conclusions about the most typical values and extreme points [9].

It should also be noted that mixed types of infographics can be used to study the relationships between different climatic indicators.

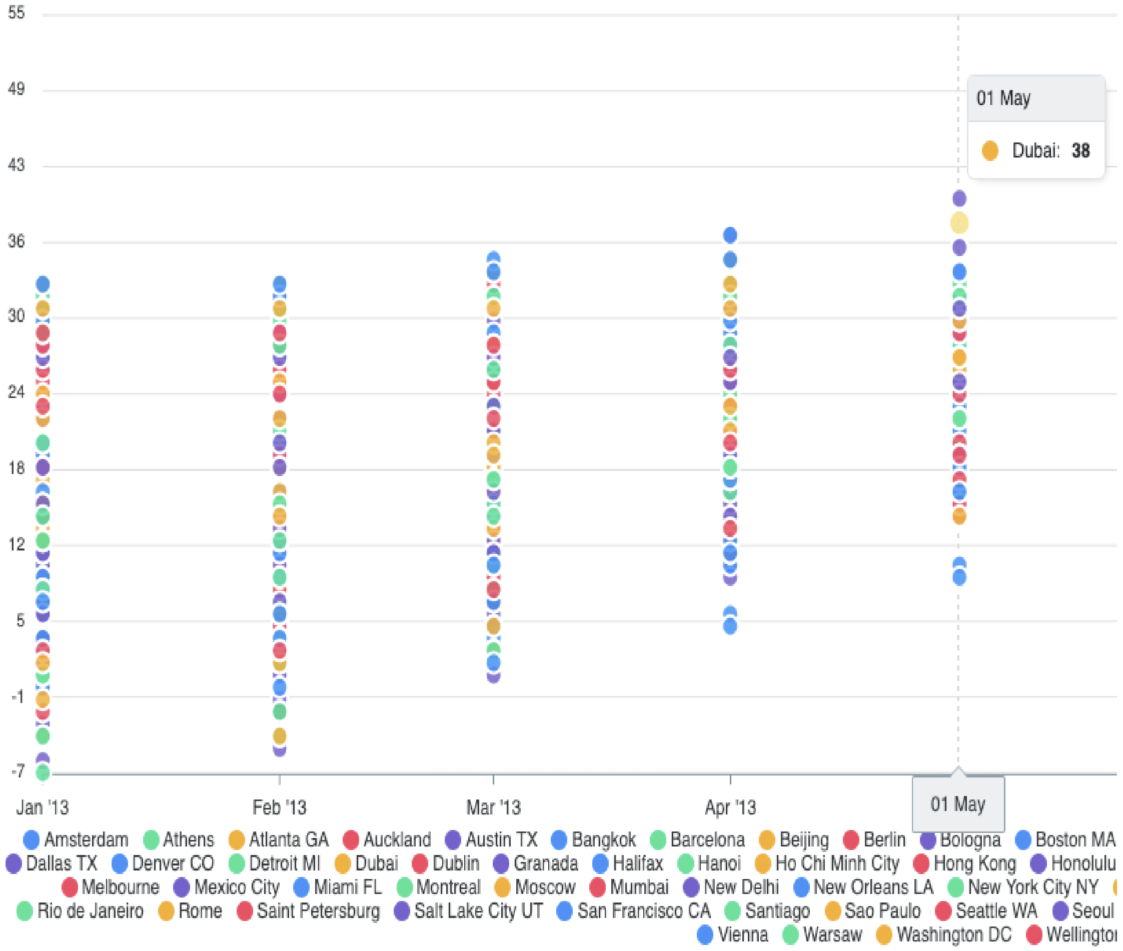
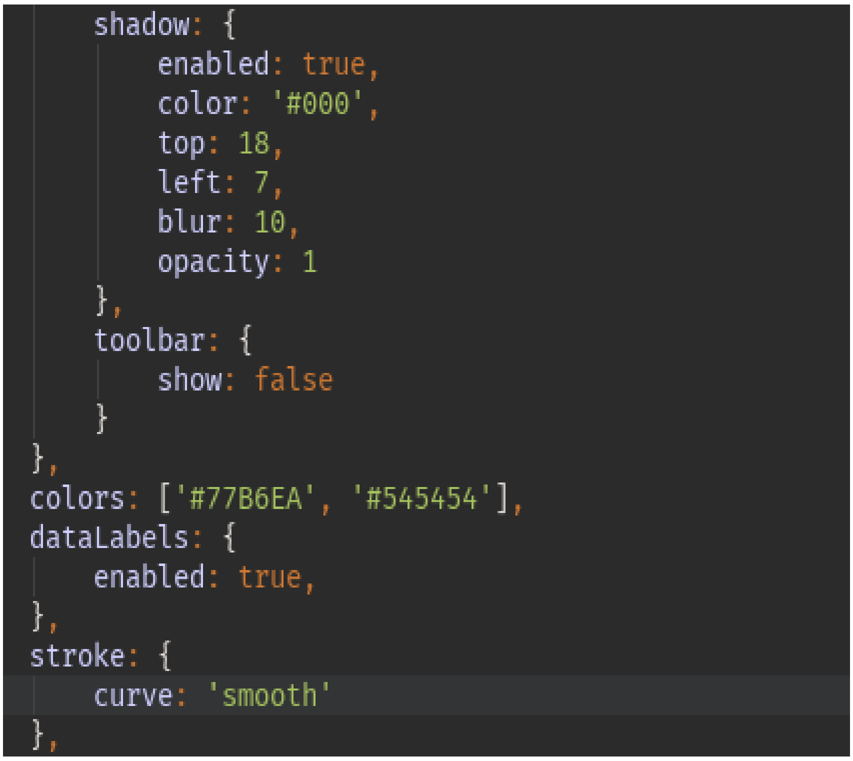


Figure 3.23 – Diagram of temperature variation according to geographical places

An important advantage of this data visualization system is that despite the variety of created graphs, the methods of their configuration and customization are quite similar.

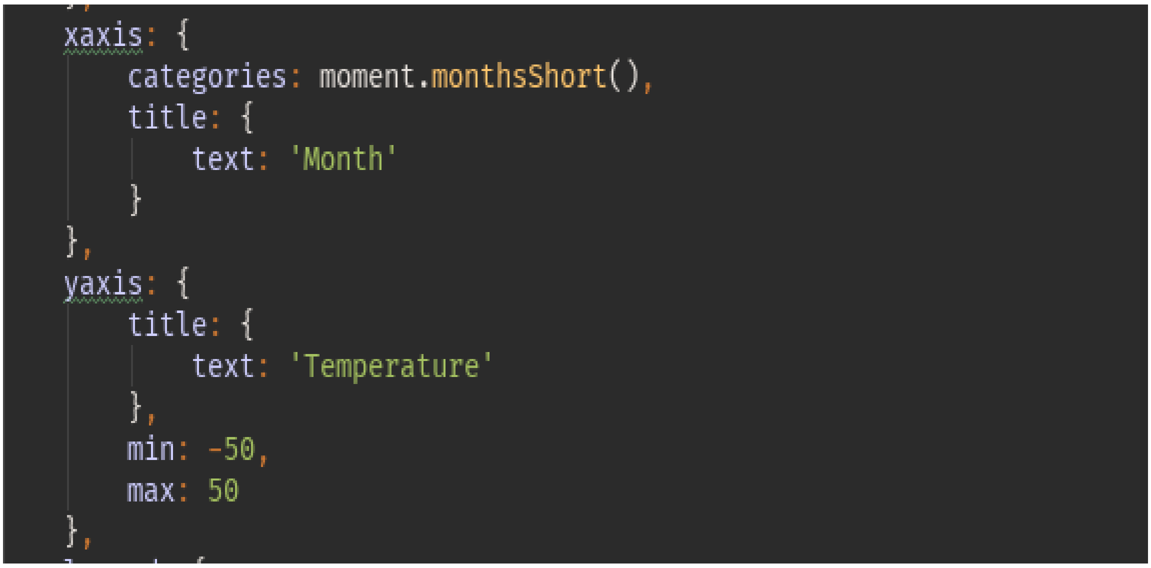
First, adjust the appearance of the canvas graphics, as shown in Figure 3.24.



```
    shadow: {
        enabled: true,
        color: '#000',
        top: 18,
        left: 7,
        blur: 10,
        opacity: 1
    },
    toolbar: {
        show: false
    }
},
colors: ['#77B6EA', '#545454'],
dataLabels: {
    enabled: true,
},
stroke: {
    curve: 'smooth'
},
```

Figure 3.24 – Adjusting the appearance of the graphics canvas

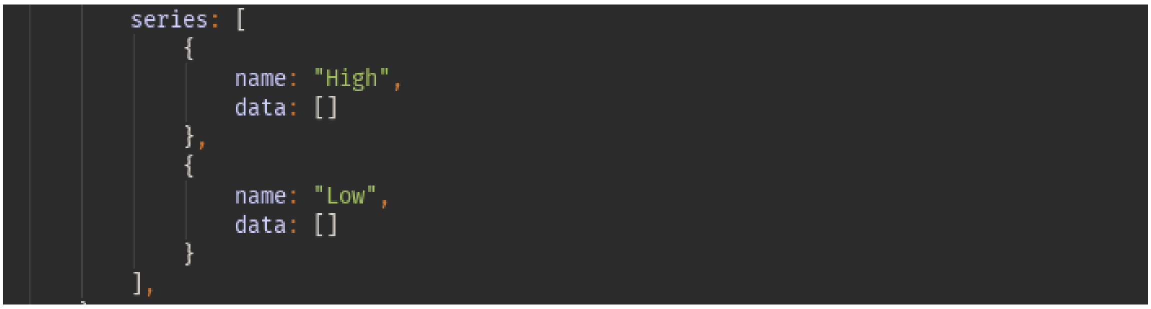
Next, adjust the graph axis to visualize the data, as shown in Figure 3.25.



```
xaxis: {
    categories: moment.monthsShort(),
    title: {
        text: 'Month'
    }
},
yaxis: {
    title: {
        text: 'Temperature'
    },
    min: -50,
    max: 50
},
```

Figure 3.25 – Adjustment of graph axes

To transmit the display, the processed and analyzed data are transmitted using data series (Figure 3.26).



```
        series: [
            {
                name: "High",
                data: []
            },
            {
                name: "Low",
                data: []
            }
        ],
```

Figure 3.26 – Data series

After making all the necessary settings, we get a powerful tool for visualizing data sets for the system of analysis of climatic indicators.

The ApexCharts library gave us a cross-browser and adaptive solution that we used and customized to visualize the patterns of maximum and temperature depending on the geographical location of the objects under study, and created a chart showing the seasonality of precipitation.

In the further development of our application, we will create new graphs to

Demonstrate the analyzed data depending on user needs and available data sets.

3.3 Conclusions of the section

Therefore, in this section the client part of the computer system of analysis and visualization of climate data is researched and developed. The JavaScript programming language (ECMAScript 6 standard) and the ReactJS v.16.12.0 framework based on it were used for development.

In the process of application stylization and creation of a cross-browser and adaptive solution that meets the modern requirements of UI / UX design, the CSS framework React Bootstrap v4.0.1 was analyzed and applied.

The functionality of interaction with the API of the server part is implemented using the Fetch API, which expands the capabilities compared to the use of a standard objectXMLHttpRequest.

To develop the visualization of the obtained and analyzed data, the JavaScript library ApexCharts was used, which uses SVG graphics to draw diagrams, which resulted in a high level of customization and adaptability of dynamic graphs.

The peculiarity of the work is that the main number of logical operations for the analysis of historical climatic indicators we concentrated in the client part, which accelerated the process of visualization of information for the user compared to the classical approach to developing client-server programs, where the main business logic is thus, the scientific novelty and high relevance of research work was confirmed.

4 LIFE SAFETY

Occupational safety and health issues are considered for the design and development phase of climate data analysis and visualization system.

Occupational safety is a system of legal, socio-economic, organizational and technical, sanitary and hygienic and treatment and prevention measures and tools aimed at preserving human life, health and ability to work. Working conditions at the workplace, safety of technological processes, machines, mechanisms, equipment and other means of production, condition of collective and individual protection means used by the employee, as well as sanitary and living conditions must meet the requirements of the law. An employee has the right to refuse the assigned work if a work situation has arisen that is dangerous to his life or health or to the people around him, or to the work environment or the environment. He must immediately notify his immediate supervisor or employer. The existence of such a situation is confirmed, if necessary, by labor protection specialists of the enterprise with the participation of a representative of the trade union of which he is a member or a person authorized by employees on labor protection (if the trade union was not established), as well as an insurance expert [12]. The task of labor protection is to minimize injuries and illnesses of the employee while ensuring comfort with maximum productivity. The main objectives of labor protection are the formation of specialists with the necessary knowledge and practical skills on legal and organizational issues of labor protection, industrial sanitation, safety, fire safety.

4.1 General characteristics of the room and workplace

The development of the analysis and visualization system is performed in a room located on the fourth floor of an eight-storey building with general and local lighting. The room has one-sided lighting, the windows are oriented to the east, the windows have shutters. White ceiling with a reflection coefficient of 0.7, light brick walls with a reflection coefficient of 0.5. There are 4 people working in the room, in accordance with this we obtain input data for the analysis of potentially dangerous and harmful production factors, which are given in table. 4.1.

Table 4.1 – Incoming data

Room parameters	**Value**
Length x width x height	6.6 x 6.1 x 2.7 m
Area	40.26m²
Volume	108,70 m³
Workplace number	**Specifics of work**
I workplace	Front-end programmer (web application client development specialist)
II workplace	Back-end programmer (specialist in the development of the server part of web applications and database design)
III workplace	Business analyst (also acts as a product manager)
IV workplace	UI-UX web designer
Technical means (quantity)	**Name and characteristics**
Monitor (4 pcs.)	HP 22Xi / 21.5 "/ 1920x1080px / IPS
Computer (4 pcs.)	HP ProBook 440 G6, 14 "IPS screen (1920x1080) Full HD, Intel Core i7-8565U (1.8 – 4.6 GHz) / RAM 16 GB / SSD 256 GB
Floor cooler (1 piece)	CRYSTAL YLR3-5V208
Air conditioner (1 piece)	DEKKER DSH105R / G / 26m2 / 2,65kW- 2.9 kW / 25x74.5x19.5 cm / 9 kg
General purpose luminaries (3 pcs.)	The lamp raster built-in 4x18W
Local lamps (4 pcs.)	Delux Decor TF-05/1 x 40W

According to NPAOP 0.00-7.15-18 [14], the area S 'allocated for one workplace with a personal computer must be at least 6 m2 and the volume – at least 20 m3. There are 4 workplaces in the room, which fully meets the required standards.

We calculate the actual values of these indicators by dividing the volume of the room and the total area by the number of employees.

Therefore, based on the results obtained in terms of area and volume, the room meets the standards.

Table 4.2 – Workplace characteristics

№	The name of the parameter	Value	
		in fact	Normative
1.	Height of a working surface, mm	780	680 – 800
2.	Width of a working surface, mm	1500	not less than 600
3.	Depth of a working surface, mm	750	not less than 600
4.	Height of space for legs, mm	750	not less than 600
5.	Width of space for legs, mm	800	not less than 500
6.	Depth of space for legs, mm	750	not less than 450
7.	Seat surface height, mm	480	400 – 500
8.	Seat width, mm	500	not less than 400
9.	Seat depth, mm	500	not less than 400
10.	Height of a basic surface of a back, mm	550	not less than 300
11.	Width of a surface of a back, mm	470	Not less than 380
12.	Length of armrests, mm	300	not less than 250
13.	Width of armrests, mm	60	50 – 70
14.	Distance from eyes to the screen, mm	650	600 – 700

It is possible to draw a conclusion that the sizes of a workplace of the programmer correspond to the established norms, proceeding from the set parameters.

4.2 Analysis of potentially dangerous and harmful production factors in the workplace

When creating a system of analysis and visualization, the work is performed sitting without physical effort, so it belongs to the category of light Ia [15].

Premises for work must be equipped with heating, air conditioning or supply and exhaust ventilation in accordance with DBN B.2.5-67: 2013. Normalized parameters of the microclimate, ionic composition of air, content of harmful substances meet the requirements of LTO 3.3.6.042-99, GN 2152-80, GOST 12.1.005-88, DSTU GOST 12.0.230: 2008 and DSTU GOST 12.4.041: 2006. Ventilation is understood as a set of measures and means designed to ensure meteorological conditions and cleanliness of the air environment that meet hygienic and technical requirements at permanent places and service areas. The main task of ventilation is to remove polluted, humid or heated air from the room and supply clean fresh air.

The sources of noise in the room are the fan of the system unit, laptop and air conditioner. The sound generated by the fan and air conditioner can be classified as constant.

According to DBN B.2.5-28: 2018 the work belongs to the category of visual works. The use of natural, artificial and mixed lighting is envisaged.

The computer is a single-phase consumer of electricity powered by 220V AC from a network with grounded neutral. IBM PC refers to electrical installations up to 1000V closed version; all conductive parts are in the casings.

According to the method of protecting a person from electric shock, computers and peripherals must meet 1 class of protection.

Technical methods of protection against electric shock is reduced to the use of current of safe voltage, protection in case of accidental touching current-carrying parts and against excessive currents, protection in case of voltage transfer to non-current-carrying metal parts of the installation.

Safe voltage is obtained from the high voltage grid (110-120 V) by means of step-down transformers.

Protection against contact with live parts of the installation is achieved by means of insulation, fencing off the use of blocking safety devices and inaccessibility of the location of the installations.

Switchboards are placed in closed metal casings-boxes.

Safety alarm is used in the form of posters and inscriptions. The best light alarms are double, which in the presence of voltage lights a red light, and in its absence – green.

Protection against excessive currents – short circuits and overload currents, which can cause insulation to ignite, is provided by fuses and circuit breakers, and protection against voltage transfer to live parts by means of protective earthing and protective disconnection.

Fire prevention is achieved by eliminating the formation of sources of ignition and combustible environment.

Fires of the following classes are possible in this room: A – combustion of solids, E – combustion of live electrical installations.

4.3 Conclusions to the section

The analysis of working conditions in the considered working room showed that working conditions with the PC correspond to specifications.

Instruction and training of all working safe methods of work is the main prerequisite for a sharp reduction and even complete elimination of injuries.

The main task of training and coaching is that workers before entering the job get the necessary knowledge of safe work and learn the rules of safety.

Due to the fact that this type of work is harmful due to the tension, the software developer is recommended to take a break from work with minor gymnastic and eye exercises.

We also considered the peculiarities of rescue and other urgent work in the elimination of nozzles of major industrial accidents and catastrophes at facilities, which listed the main causes of major industrial accidents and catastrophes at facilities and considered measures to eliminate these consequences.

CONCLUSIONS

Climatic and meteorological indicators were analyzed in the process of researching methods and means of creating a computer system for data analysis and visualization. The study first examines and identifies the strengths and weaknesses of existing climate data analysis systems.

Based on the data obtained for the development of the client-server application, methods of analysis and generalization selected MS SQL Server to create a database to create the server part selected C #, JavaScript language (ECMAScript6) and based on it framework ReactJS v.16.12.0 for the client for the application. The functionality of interaction with the API of the server part is implemented using the Fetch API.

In the process of application stylization and creation of a cross-browser and adaptive solution that meets modern requirements of UI / UX design, the CSS framework React Bootstrap v4.0.1 was analyzed and applied.To develop a visualization of the obtained and analyzed data, the ApexCharts library was used, which uses SVG graphics to create info-graphics.

The method of experiment and generalization proved the scientific novelty of the study, in particular, the peculiarity of the developed application is that for the first time performed logical operations for the analysis of historical climatic indicators in the client part of the program. where the main business logic is focused on the server side.

The study substantiates the cost-effectiveness of implementing a computer system for analysis and visualization of climate data. Based on the analysis of labor protection, safety and environmental requirements, ways to minimize the negative impact of harmful production factors have been identified.

REFERENCES

1. S.I. Reshetchenko. Meteorology and climatology: a textbook. H .: VN KarazinKhNU, 2015. 220 p.

2. Meteorology and climatology: Textbook. State higher educational institution of NLTU of Ukraine. Contributor SP Мельничук. Lviv: NNLTU of Ukraine, 2018. 148 p.

3. Volodarsky E., Kosheva L. Statistical data processing. K: NAU-druk, 2008. 308 p.

4. C. Donald Ahrens and Others. Meteorology Today. Cengage Learning, 2016. 662 p.

5. Flanagan D. JavaScript: a handbook. Dialectics-Williams, 2016. 320 p.

6. Nixon Robin. We create dynamic websites using PHP, MySQL, JavaScript, CSS and HTML5. Peter, 2016. 768 p.

7. Banks A., Porcello E. React and Redux. Functional web development. Peter, 2016. 336 p.

8. Richter J. P55 CLR via C #. Programming on the Microsoft .NET Framework 4.5 in C #. SPb .: Piter, 2013. 896 s.

9. Gladun O. Visualization of information: infographics. Olga Gladun. Bulletin of the KhDADM, 2012, №4. P.11-14.

10. Methodical instructions on the implementation of the economic part of the diploma project for full-time students majoring in 7.091501 – "Computer Systems and Networks" VO Mospan, II Bed. Kremenchuk: KSU Publishing Department. M. Ostrogradsky, 2013. 39 p.

11. The amount of the single contribution depending on the class of occupational risk URL: http://buhgaltera.net.ua/rozmir-edinogo-vnesku-v-zalezhnosti-vid-klasu-profesijnogo-riziku/ (access date: 16.10.2019).

12. Kulakov MA, Lyapun VO, Soft VO etc. Civil defense: a textbook. Kharkiv: Fakt, 2008.

13. Shobotov VM Civil defense: a textbook. K .: Center for Educational Literature, 2004.

14. Norms for determining the categories of premises, buildings and outdoor installations for explosion and fire hazard. NAPB B.03.002-2007. (approved by the order of the Ministry of Emergencies of Ukraine dated 03.12.2007 № 833).

15. LTO 3.3.6.042-99. Sanitary norms of microclimate of industrial premises [Text]. K., 2000. 16 p.

16. Hygienic classification of labor according to indicators of harmfulness and danger of factors of the production environment, severity and intensity of the labor process (approved by the order of the Ministry of Health of Ukraine from 12.08.2014 № 248).

17. Rules of labor protection during operation of electronic computers. NPAOP 0.00-1.28-10 (approved by the order of the State Committee of Ukraine for Industrial Safety, Labor Protection and Mining Supervision dated March 26, 2010 № 65).

18. Hygienic classification of labor according to indicators of harmfulness and danger of factors of the production environment, severity and intensity of the labor process (approved by the order of the Ministry of Health of Ukraine from 08.04.2014 № 248).

19. Rusalovsky AV Legal and organizational issues of labor protection: Textbook. way. K .: University "Ukraine", 2009. 295 p.

20. Tarasova VV Environmental statistics. K .: Center for Educational Literature, 2008. 392 p.