

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційних радіотехнологій та технічного захисту інформації

Кафедра Радіотехнологій інформаційно-комунікаційних систем

АТЕСТАЦІЙНА РОБОТА Пояснювальна записка

Рівень вищої освіти другий(магістерський)

ГЮІК.467957.000ПЗ

(позначення документа)

Дослідження особливостей аналізу та агрегації синоптичного прогнозу

(тема)

Виконав:

студент II курсу, групи ІКТм -19-1

Плотников О. Ю.

(прізвище, ініціали)

Спеціальність

122 Комп'ютерні науки

(код і повна назва спеціальності)

Тип програми

освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма

Інформаційно-комунікаційні технології

(повна назва освітньої програми)

Керівник доцент Бітченко О.М.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри РТІКС

(підпис)

Цопа О.І.

(прізвище, ініціали)

2020 р.

Не містить відомостей заборонених для відкритого публікування.

Студент

О. Ю. Плотников

Керівник

О. М. Бітченко

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет Інформаційних радіотехнологій та технічного захисту інформації
Кафедра Радіотехнологій інформаційно-комунікаційних систем
Рівень вищої освіти другий (магістерський)
Спеціальність 122 Комп'ютерні науки
Тип програми Освітньо-професійна
Освітня програма Інформаційно-комунікаційні технології

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

«_____» _____ 2020 р.

ЗАВДАННЯ НА АТЕСТАЦІЙНУ РОБОТУ

студентові ПЛОТНИКОВУ Олексію Юрійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи ДОСЛІДЖЕННЯ ОСОБЛИВОСТЕЙ АНАЛІЗУТА
АГРЕГАЦІЇ СИНОПТИЧНОГО ПРОГНОЗУ

затверджена наказом по університету від 2 листопада 2020 р. № 1506Ст

2. Термін подання студентом проекту (роботи) 12 грудня 2020 р.

3. Вихідні дані до проекту (роботи)

3.1 Проаналізувати моделі прогнозів погоди та їх функції

3.2 Проаналізувати методи побудови архітектури full-stack додатків

3.3 Рекомендований час відгуку системи 3...4 секунди

4. Перелік питань, що потрібно опрацювати в роботі

Реферат. Перелік умовних позначень, символів, одиниць, скорочень і термінів.

Вступ. 4.1 Огляд та аналіз сервісів для прогнозування погоди. 4.2 Теоретичний

розділ. 4.3 Проектний розділ. 4.4 Практичний розділ. Висновки. Перелік

посилань. Додатки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри)

Комп'ютерна презентація

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

| Найменування розділу | Консультант (посада, прізвище, ім'я, по-батькові) | Позначка консультанта про виконання розділу | |
|----------------------|--|---|------|
| | | підпис | дата |
| Основна частина | доц. Бітченко Олександр Миколайович | | |
| | | | |
| | | | |

КАЛЕНДАРНИЙ ПЛАН

| № | Назва етапів роботи | Терміни виконання етапів роботи | Примітка |
|----|--|---------------------------------|----------|
| 1 | Вступ | 5.09-10.09 | Виконано |
| 2 | Огляд та аналіз сервісів для прогнозування погоди. | 11.09-20.09 | Виконано |
| 3 | Теоретичний розділ. | 21.09-30.09 | Виконано |
| 4 | Проектний розділ. | 1.10-20.10 | Виконано |
| 5 | Практичний розділ | 21.10-11.11 | Виконано |
| 6 | Реферат | 12.11-15.11 | Виконано |
| 7 | Перелік умовних позначень, символів, одиниць, скорочень і термінів | 16.11-17.11 | Виконано |
| 8 | Висновки | 18.11-20.11 | Виконано |
| 9 | Оформлення пояснювальної записки | 20.11-30.11 | Виконано |
| 10 | Оформлення презентації | 1.12-11.12 | Виконано |
| 11 | Подання роботи на кафедрі | 12.12.2019 | Виконано |

Дата видачі завдання **4 вересня 2020 р**

Студент _____
(підпис)

Керівник роботи _____
(підпис)

доц. Бітченко О.М.
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка до атестаційної роботи містить 93 сторінки тексту, 5 таблиць, 58 рисунків, 24 джерела посилання та 3 додатки.

ПРОГНОЗ ПОГОДИ, АНАЛІЗ ДАНИХ, БАЗА ДАНИХ, SQL, JAVASCRIPT, NODE.JS, REACT.

В роботі виконано огляд моделей прогнозування погоди, обрано метод оцінки моделей, що буде відповідати вимогам мети дослідження

Задачею дослідження є проектування системи, яка за допомогою обробки даних моделей прогнозу погоди буде надавати корисну інформацію користувачу щодо точності прогнозу моделі та задовольняти поставлену мету дослідження.

У результаті роботи спроектована система аналізу даних прогнозу погоди та реалізований її прототип.

При розробці прототипу системи були використані: теорія проектування реляційних баз даних, основні принципи нормалізації реляційних баз даних; мова запитів SQL, мова програмування JavaScript, Node.js, React.js.

ABSTRACT

Explanatory note to the certification work: 93 pages, 5 tables, 58 figures, 24 sources.

WEATHER FORECAST, DATA ANALYSIS, DATABASE, SQL,
JAVASCRIPT, NODE.JS, REACT.

The goal of research is focused on review weather forecast models and choosing a method of models estimation.

The task of research is focused on designing a system that, by processing the data of weather forecast models, will provide useful information to the user on the accuracy of the model forecast and will satisfy set goals for the research.

As a result, a weather forecast data analysis system was designed and its working prototype was implemented.

In developing the prototype system were used: the theory of relational database design, the basic principles of normalization of relational databases; SQL query language, JavaScript programming language, Node.js, React.js.

ЗМІСТ

| | |
|---|----|
| Перелік умовних позначень, символів, одиниць, скорочень і термінів..... | 7 |
| Вступ..... | 8 |
| 1 Огляд та аналіз сервісів для прогнозування погоди..... | 10 |
| 1.1 Сервіс Weather Underground..... | 10 |
| 1.2 Сервіс Яндекс»..... | 13 |
| 1.3 Сервіс AccuWeather..... | 16 |
| 1.4 Сервіс OpenWeatherMap..... | 19 |
| 1.5 Сервіс yr.no..... | 22 |
| 1.6 Сервіс ForecastAdvisor..... | 24 |
| 1.7 Сервіс slimyhorror..... | 27 |
| 1.8 Мета та постановка задачі..... | 28 |
| 2 Теоретичний розділ | 29 |
| 2.1 Опис методологій прогнозів погоди | 29 |
| 2.2 Опис методології прогнозів погоди різної завчасності..... | 30 |
| 2.3 Оцінка похибки при прогнозуванні погоди..... | 32 |
| 2.4 Обґрунтування вибору засобів реалізації програмного продукту..... | 36 |
| 2.4.1 JavaScript..... | 36 |
| 2.4.2 Node.js..... | 38 |
| 2.4.3 Express.js..... | 40 |
| 2.4.4 React..... | 40 |
| 2.4.5 Bootstrap..... | 43 |
| 2.4.6 MySQL..... | 45 |
| 3 Проектний розділ..... | 48 |
| 3.1 Формування вимог до системи..... | 48 |
| 3.2 Розробка структури бази даних | 48 |
| 4 Практичний розділ..... | 52 |
| 4.1 Реалізація прототипу системи..... | 52 |
| 4.2 Розробка front-end частини..... | 52 |

| | |
|--|----|
| 4.3 Розробка back-end частини..... | 57 |
| 4.4 Інструкція користувача | 60 |
| Висновки | 66 |
| Перелік джерел посилання..... | 67 |
| Додатки..... | 70 |
| Додаток А Програмні коди..... | 71 |
| Додаток Б Копії слайдів..... | 82 |
| Додаток В Відомість атестаційної роботи..... | 92 |

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ,
ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

SQL – Structured Query Language – мова структурованих запитів

IDE – Integrated Development Environment – інтегроване середовище розробки

PWS – Private Weather Stations – приватні метеостанції

API – Application Programming Interface – набір визначень підпрограм

JSON – JavaScript Object Notation – запис об'єктів JavaScript

XML – eXtensible Markup Language – розширювана мова розмітки

HTML – HyperText Markup Language – мова розмітки гіпертексту

CSS – Cascading Style Sheets – каскадні таблиці стилів

DOM – Document Object Model – об'єктна модель документа

ВСТУП

Прогноз погоди – науково обґрунтоване припущення про майбутній стан погоди в певному пункті або регіоні на певний період. Складається метеорологічними службами на основі методів метеорології.

Прогнозами погоди складеними в гідрометцентрах користуються в різних галузях людської діяльності: на транспорті, у сільському господарстві, комунальні служби та служби надзвичайних ситуацій. У часи масового поширення засобів і приладів комунікації популярність зміщується в бік інтернет–сервісів та мобільних застосунків.

Спираючись на деякі критерії при перегляди сайтів прогнозів, можна завжди знайти найбільш точну погоду. Багато сайтів надають прогнози на 7, 14, 30 і навіть 90 днів. Але якщо розраховуєте отримати найбільш точні дані, не варто орієнтуватися на такі далекі дослідження. Людство ще не навчилося настільки точно прогнозувати погоду в далекій перспективі.

Якщо ресурс показує не тільки цифри, але і «відчуття» – це добре. Деякі сайти прогнозу погоди дозволяють дізнатися, як саме відчувається температура. Тому що іноді різниця по відчуттю і за фактом може досягати десятка–другого градусів.

Сайт з мобільним додатком найбільш зручний для постійного використання. Мобільні додатки дозволяють дивитися поточну температуру та інші дані прямо на екрані смартфона.

В ідеалі варто вивчати «професійні» прогнози погоди для військових або аеропортів. Деякі інтернет–ресурси співпрацюють з різними офіційними структурами. Наприклад, з аеропортами або військовими. Вимоги до таких прогнозів набагато вище, ніж до «цивільним», але доступні вони далеко не скрізь.

Метеостанції, метеорологічні служби розраховують, аналізують, розробляють, прогнозують на конкретний період часу характер атмосферних явищ, щоб люди були готові до всіх погодних умов, так як більшість сфер людського життя залежать від прогнозу погоди.

Метою даної атестаційної роботи є розробка системи та алгоритму розрахунку точності прогнозів погоди різного типу.

Для досягнення поставленої мети в роботі сформульовані наступні задачі:

проаналізувати сучасні моделі прогнозів погоди;

вести статистичні дані щодо прогнозу погоди на окремому періоді часу; провести огляд програмних рішень для розробки full-stack

додатку; розробити систему з працюючим інтерфейсом користувача.

ПОГЛЯД ТА АНАЛІЗ СЕРВІСІВ ДЛЯ ПРОГНОЗУВАННЯ ПОГОДИ

1.1 Сервіс Weather Underground

Weather Underground - це комерційний метеорологічний сервіс, що надає інформацію про погоду в режимі реального часу через Інтернет [1]. Weather Underground надає звіти про погоду для більшості великих міст світу на своєму веб-сайті, а також місцеві звіти про погоду для газет і сторонніх сайтах. Його інформація надходить з Національної метеорологічної служби і більше 250 000 персональних метеостанцій (PWS). Weather Underground належить компанії Weather, дочірньої компанії IBM. Головний інтерфейс Weather Underground представлено на рисунку 1.1.

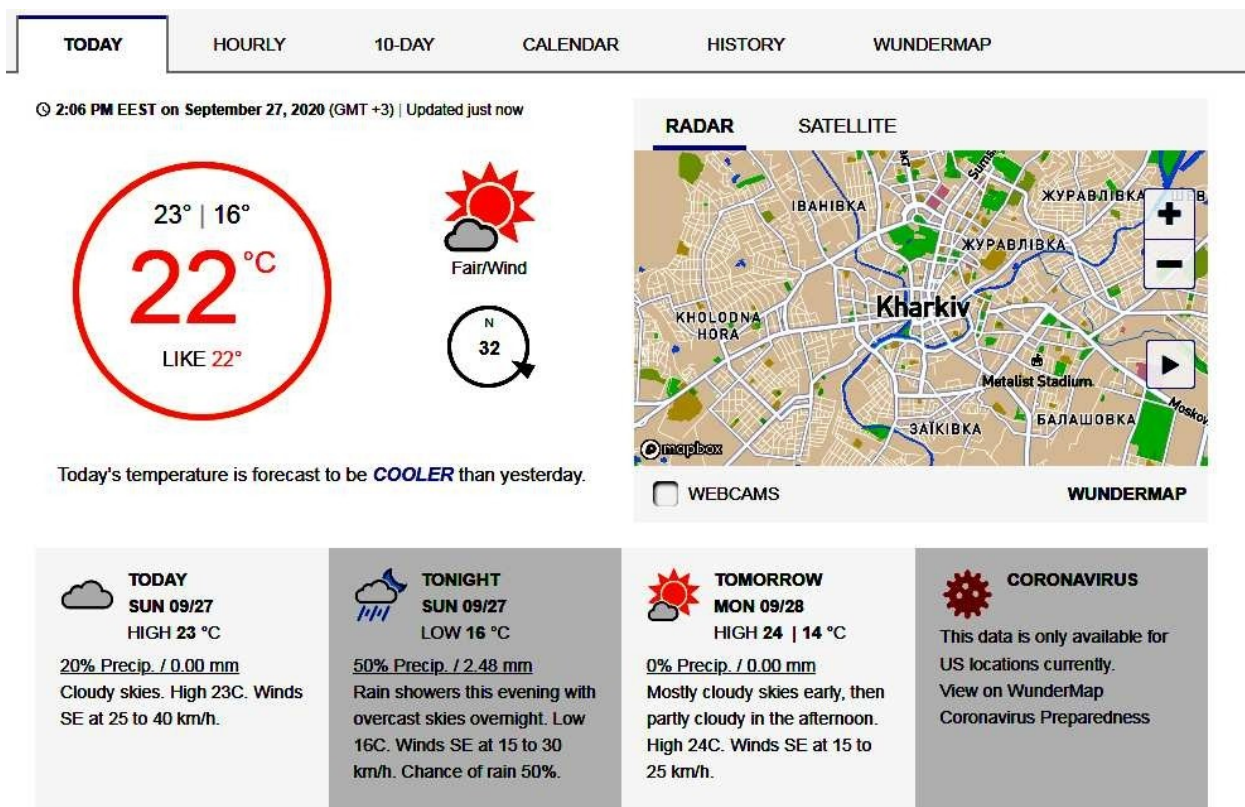


Рисунок 1.1 - Головний інтерфейс Weather Underground

На сайті представлений погодинний прогноз (рисунок 1.2), а також прогноз погоди на 10 днів вперед (рисунок 1.3).

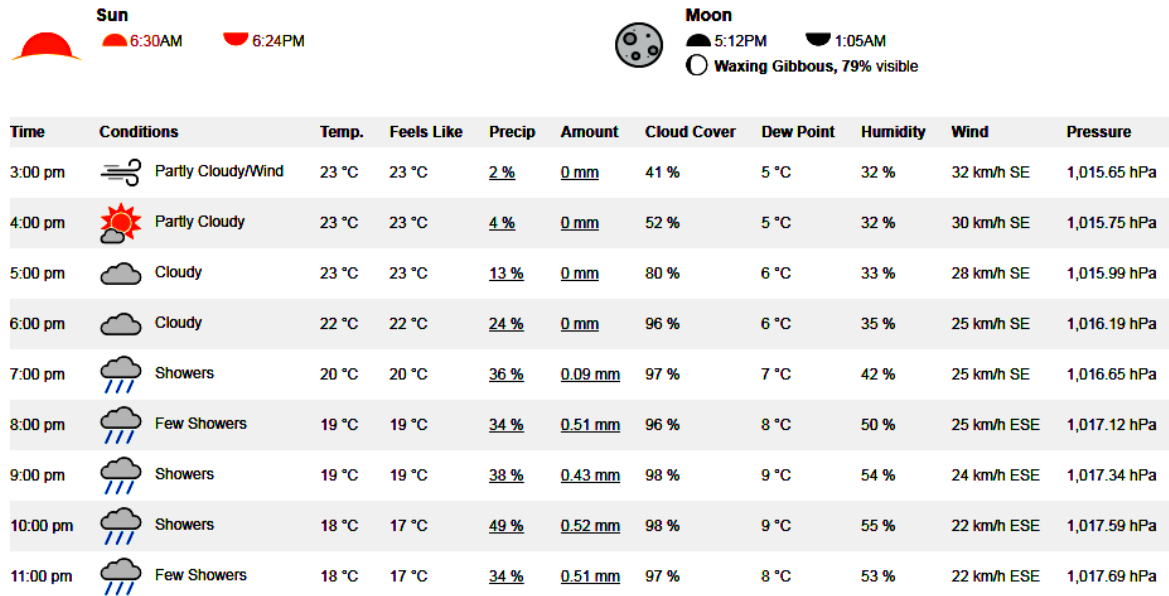


Рисунок 1.2 – Погодинний прогноз погоди Weather Underground

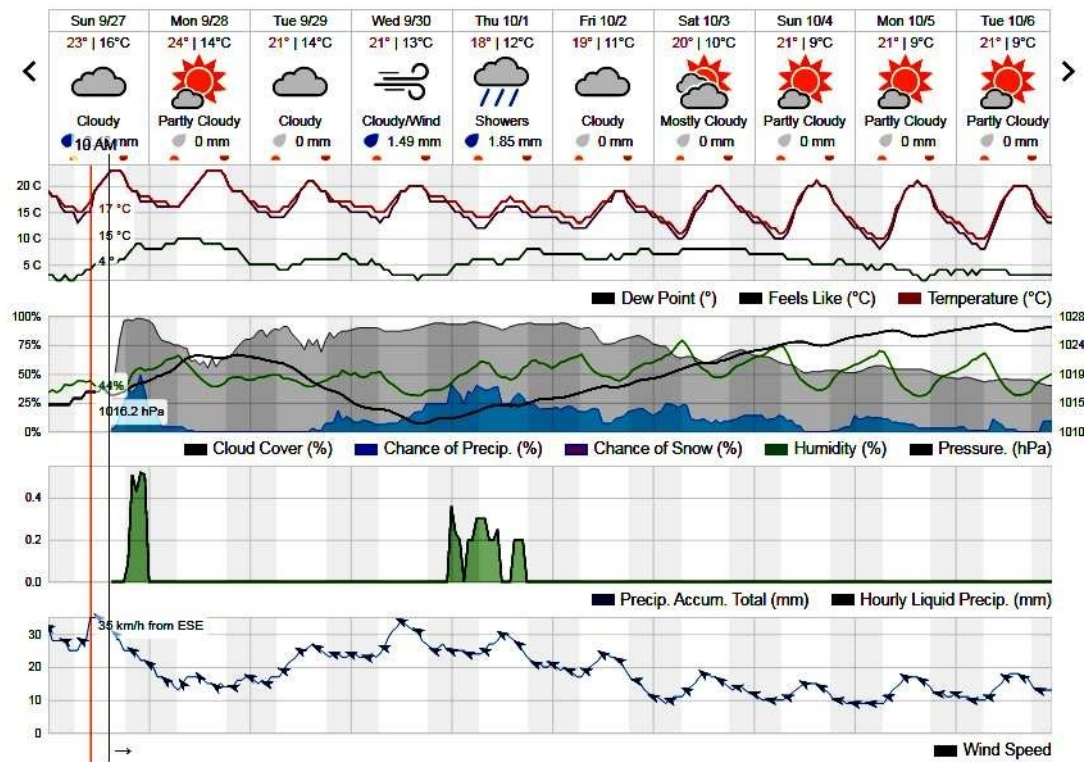


Рисунок 1.3 – Прогноз погоди Weather Underground на 10 днів вперед

Веб-журнали (блоги) були однією з основних функцій Weather Underground, дозволяючи користувачам сайту створювати блоги про погоду, повсякденному житті і про все інше. Джефф Мастерс почав перший блог 14 квітня 2005, і він публікує записи в блозі майже кожен день. З 2007 по початок

2017 Річард Б. Руд вів блоги про зміну клімату та реакції суспільства з новими записами щотижня.

3 квітня 2017 року Weather Underground закрила всі блоги учасників, SMS-оповіщення і ретрансляцію погодного радіо NOAA.

Weather Underground використовує понад 250 000 персональних метеорологічних станцій по всьому світу.

WunderMap Weather Underground накладає дані про погоду від персональних метеорологічних станцій та офіційних станцій на базі Mapbox Map і забезпечує безліч інтерактивних та динамічно оновлюваних погодних та екологічних рівнів.

Меню вибору для відображення інформації представлено на рисунку 1.4.

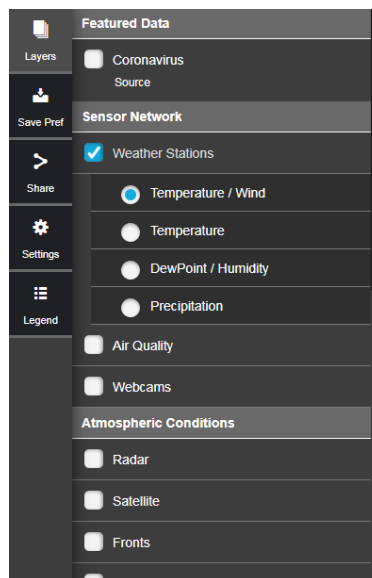


Рисунок 1.4 – Меню вибору WunderMap

При активації опції Weather Stations на карті показано розташування метеостанцій, наприклад в харківській області (рисунок 1.5).

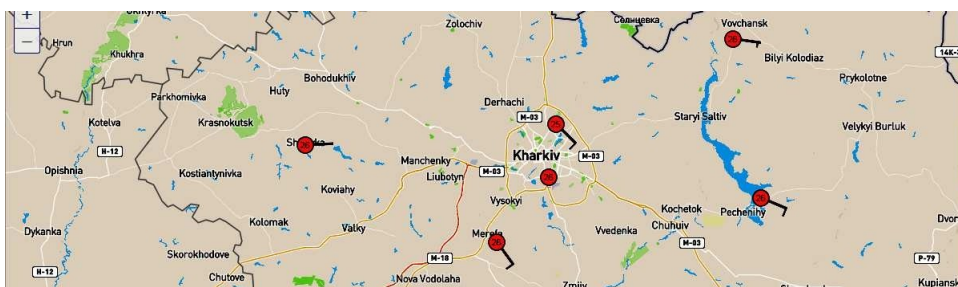


Рисунок 1.5 – Розташування метеостанцій у вибраному регіоні

Приклад загальних погодніх показань представлено на рисунку 1.6.

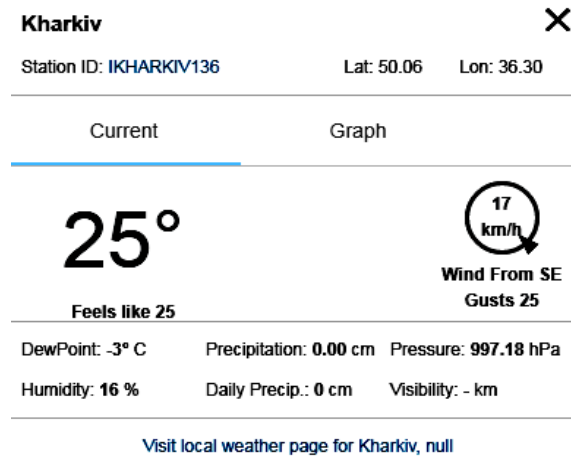


Рисунок 1.6 – Загальна погодна інформація WunderMap

Associated Press використовує Weather Underground для подання національних звітів про погоду.

Weather Underground має кілька розширень Google Chrome та додатки для iPhone, iPad та Android. Був розроблений додаток для пристроїв Roku, який було видалено. У лютому 2015 року Weather Underground випустило додаток для iOS під назвою Storm. Серед інших додатків Weather Underground – WunderStation для iPad та WunderMap для iOS та Android. У 2017 році Weather Underground скасувало підтримку "Шторму" на користь програми "Штормовий радар", випущеної The Weather Channel Interactive у червні 2017 року.

31 грудня 2018 року Weather Underground припинили підтримку свого інтерфейса для прикладного програмування, чим ще більше зменшили спектр своїх послуг.

1.2 Сервіс «Яндекс»

«Яндекс» - російська транснаціональна компанія в галузі інформаційних технологій, зареєстрована в Нідерландах і що володіє однойменною пошуковою системою в Мережі, інтернет-порталами і службами в декількох країнах. У листопаді 2015 року Яндекс розробив власну технологію прогнозу погоди -

«Meteum». Вона дозволяє будувати прогноз погоди з точністю до будинку, завдяки об'єднанню класичних моделей метеопрогнозування і технологій машинного навчання [2].

Meteum - це технологія для розрахунку прогнозу погоди, розроблена в Яндексі. За допомогою алгоритмів машинного навчання Meteum порівнює дані різних прогностичних моделей, а потім будує власний прогноз.

Meteum розраховує прогноз в два етапи.

Етап 1. Побудова власного прогнозу

Яндекс.Погода отримує дані з метеостанцій і супутників. Щоб підвищити точність прогнозу, сервіс використовує докладні дані від Яндекс.Карт про тип підстильної поверхні.

Сервіс будує власний прогноз по прогностичній моделі. Модель Яндекс.Погода формує точний прогноз не тільки для регіонів і великих міст, а й для окремих районів міст, станцій метро, дачних селищ і т. д.

Етап 2. Розрахунок фінальної версії прогнозу

Сервіс отримує відомості про прогнози, побудованих за допомогою інших прогностичних моделей: американських, канадських, японських і європейських.

Яндекс.Погода порівнює прогнози і фактичні дані про погоду за допомогою алгоритмів машинного навчання. Алгоритми визначають модель з найточнішим прогнозом для конкретних метеорологічних умов і регіону. Якщо жодна модель не надає високоточного прогнозу, Meteum комбінує дані найбільш точних прогнозів для побудови власного.

За допомогою Meteum сервіс розраховує прогноз погоди для регіонів і міст усього світу:

супутники і станції спостереження по всьому світу збирають дані про погоду;

дані надходять в метеорологічні центри;

метеорологи вносять дані в прогностичні моделі і обробляють на комп'ютерах.

Кожна прогностична модель враховує особливості циркуляції атмосфери, випаровування і конденсації вологи, нагрівання землі і вплив багатьох інших чинників. Після обробки даних метеорологи отримують прогноз температури, опадів, швидкості і напрямку вітру та інших атмосферних явищ.

Жодну прогностичну модель не можна назвати абсолютно точною, у кожній є свої переваги і недоліки. Тому для розрахунку прогнозів Яндекс.Погода використовує технологію Meteum, в якій прогностичне моделювання поєднується з машинним навчанням.

На сервісі представлені дані про погоду в даний час і на найближчі години і дні.

Дані про погоду оновлюються не рідше, ніж раз на три години. Частота оновлення інформації про погоду в конкретному місті залежить від часу поновлення даних на найближчій до нього метеостанції.

Головний інтерфейс Яндекс.Погода представлено на рисунку 1.7.

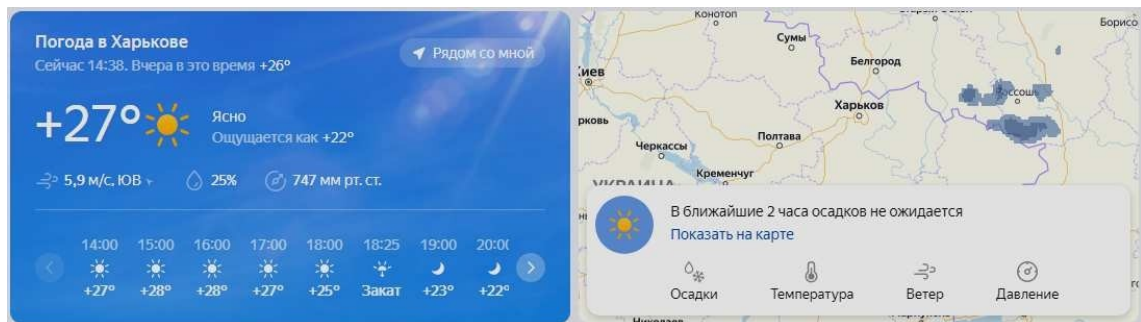


Рисунок 1.7 – Головний інтерфейс Яндекс.Погода

На сайті представлений погодинний прогноз на 10 днів наперед (рисунок 1.8) та на місяць (рисунок 1.9).



Рисунок 1.8 – Прогноз погоди на 10 днів Яндекс.Погода

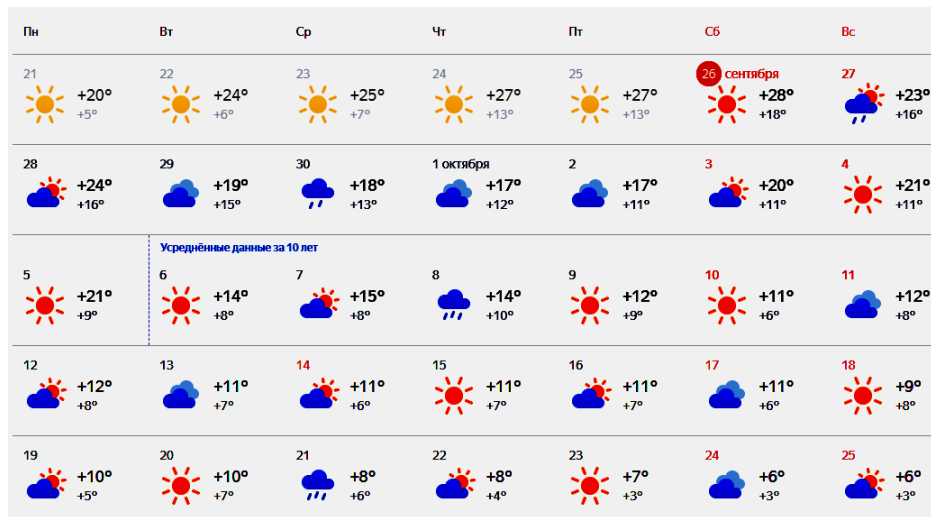


Рисунок 1.9 – Прогноз погоди на місяць Яндекс.Погода

1.3 Сервіс AccuWeather Inc.

AccuWeather Inc. - американська медіакомпанія, що надає комерційні послуги з прогнозування погоди по всьому світу.

AccuWeather надає прогнози погоди, погодні попередження, а також додаткові метеорологічні продукти і послуги клієнтами по всьому світу в ЗМІ та бізнесі, включаючи більше половини компаній зі списку Fortune 500 і тисячі інших компаній по всьому світу. Він також керує безкоштовним веб-сайтом AccuWeather.com, постачальником погоди в Інтернеті [3].

Прогнози і служби попереджень AccuWeather засновані на інформації про погоду, отриманої з численних джерел, включаючи дані спостережень за погодою і дані, зібрані Національної метеорологічної службою і метеорологічними організаціями за межами США, а також на інформації, наданій неметеорологічними організаціями, такими як Агентство з охорони навколишнього середовища і збройні сили.

AccuWeather управляє цілодобовим комерційним метеорологічним каналом, відомим як The Local AccuWeather Channel. Канал Local AccuWeather був запущений в 2005 році і в даний час транслюється по всій країні, включаючи Нью-Йорк, Лос-Анджелес, Чикаго, Філадельфії, Сан-Франциско і Х'юстон. Мережа транслює попередньо записані національні і регіональні прогнози погоди,

аналіз поточних погодних явищ і новини, пов'язані з погодою, а також місцеві погодні сегменти.

AccuWeather щодня випускає місцеві відео про погоду для використання на їх власному веб-сайті, на місцевому каналі AccuWeather, в Інтернеті, мобільних додатках і веб-сайтах. Мобільний додаток має щохвилинний прогноз, а також збирає дані спостережень за погодою з краудсорсингу.

Компанія розширилася на міжнародному рівні. AccuWeather вступила в спільне підприємство з Huafeng Media Group, отримавши виняткові права на надання прогнозів, зроблених Китайською метеорологічною адміністрацією, урядовим агентством, контролюючим Huafeng. Крім послуг з прогнозування для окремих споживачів, AccuWeather надає підприємствам прогнозні аналітичні послуги, пов'язані з погодою, такі як визначення того, як погодні умови вплинули на минулу історію продажів, і консультування підприємств по адаптації їх стратегії продажів до майбутніх погодних явищ.

AccuWeather створила уніфіковану і запатентовану систему видимої температури, і використовувала цю величину в своїх прогнозах і спостереженнях. Формула для розрахунку цього значення враховує вплив температури, вітру, вологості, інтенсивності сонячного світла, хмарності, опадів і висоти на людське тіло, аналогічно рідко використовуваної Wet-Bulb Globe Temperature. AccuWeather отримав патент США на RealFeel Temperature, але формула є комерційною таємницею і не розглядалася іншими метеорологічними органами.

Загальна погодна інформація AccuWeather представлена на рисунку 1.10.



Рисунок 1.10 – Загальна погодна інформація AccuWeather

AccuWeather також надає інформацію про якість повітря. Приклад можна побачити на рисунку 1.11.

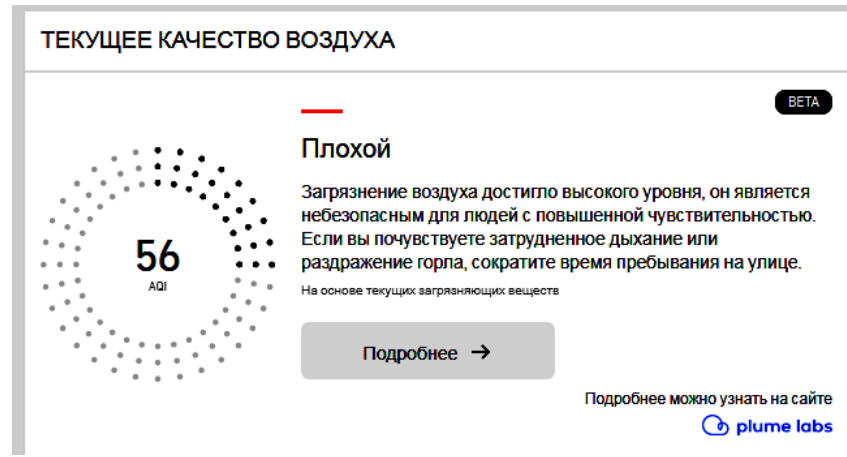


Рисунок 1.11 – Якість повітря обраної території

На рисунку 1.12 можна побачити температуру на поточний та наступний день.

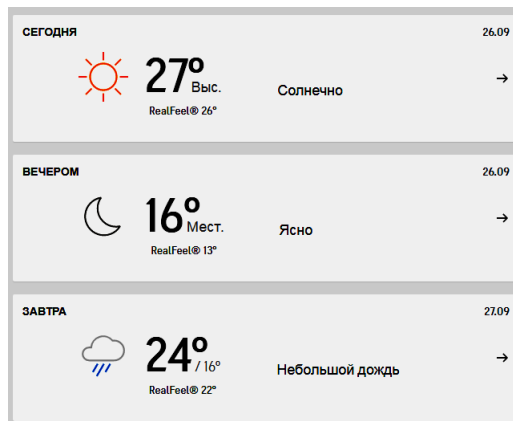


Рисунок 1.12 – Температура повітря поточного і наступного дня

На сайті представлена можливість переглянути ІК-область спектра і показати відносну теплоту об'єктів (рисунок 1.13).

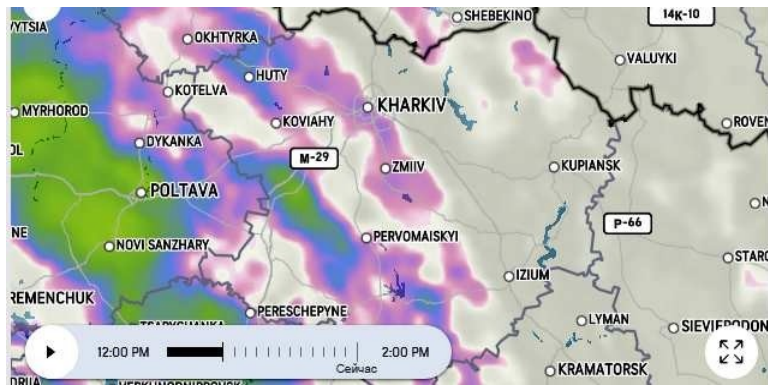


Рисунок 1.13 – ІК-область спектра зі супутника

Також на сайті представлений почасовий прогноз погоди (рисунок 1.14) та прогноз погоди на тиждень вперед (рисунок 1.15).

| | | | | | |
|-------------|--|-----|---------------|-----------------------------------|-----|
| 15 27.09 | | 23° | RealFeel® 22° | Небольшая облачность | 40% |
| 16 27.09 | | 23° | RealFeel® 20° | Преимущественно облачно с ливнями | 44% |
| 17 27.09 | | 23° | RealFeel® 20° | Ливни | 40% |
| 18 27.09 | | 22° | RealFeel® 20° | Облачно | 37% |
| 19 27.09 | | 21° | RealFeel® 19° | Облачно | 40% |
| 20 27.09 | | 18° | RealFeel® 15° | Ливни | 57% |
| 21 27.09 | | 17° | RealFeel® 13° | Ливни | 57% |
| 22 27.09 | | 18° | RealFeel® 16° | Облачно | 49% |
| 23 27.09 | | 18° | RealFeel® 16° | Облачно | 34% |

Рисунок 1.14 – Погодинний прогноз погоди AccuWeather

| | | | | |
|-------------|--|----------|---------------------------------|-----|
| ВС 27.09 | | 23° /15° | Небольшой дождь | 55% |
| ПН 28.09 | | 24° /15° | Временами переменная облачность | 3% |
| ВТ 29.09 | | 22° /15° | Временами переменная облачность | 14% |
| СР 30.09 | | 19° /13° | Небольшая облачность | 25% |
| ЧТ 01.10 | | 17° /13° | Облачно | 25% |
| ПТ 02.10 | | 17° /9° | Небольшая облачность | 18% |
| СБ 03.10 | | 21° /11° | Ливень | 55% |
| ВС 04.10 | | 21° /9° | Преимущественно облачно | 4% |

Рисунок 1.15 – Прогноз погоди AccuWeather на тиждень вперед

1.4 Сервіс OpenWeatherMap

OpenWeatherMap - онлайн сервіс, який надає платний (є функціонально обмежена безкоштовна версія) АРІ для доступу до даних про поточну погоду, прогнозами і історичними даними. Як джерело даних використовуються офіційні метеорологічні служби, дані з метеостанцій аеропортів, і дані з приватних

метеостанцій [4].

Інформація обробляються OpenWeatherMap, після чого, на основі даних будується прогноз погоди і погодні карти, наприклад карти хмарності та опадів. Основною ідеєю сервісу OWM є використання приватних погодних станцій, які допомагають підвищити точність вихідної погодної інформації і, як наслідок, точність прогнозів погоди.

Натхненниками OpenWeatherMap були картографічний сервіс OpenStreetMap і Вікіпедія, створюють вільну інформацію і надають її безкоштовно для всіх. В даний час OWM є комерційною організацією.

OpenWeatherMap використовує платний API (є функціонально обмежений безкоштовний доступ), щоб надати дані поточної погоди, прогнозу і карт з погодними явищами, такими як хмари, вітер, тиск і опади. Всі погодні дані можуть бути отримані в форматах JSON, XML або HTML

Поточні погодні дані. Поточні погодні дані можуть бути знайдені по місту (понад 200.000 доступних міст) або географічними координатами. Дані оновлюються кожні 10 хвилин.

Прогнози. Прогнози можуть бути знайдені по місту (понад 200.000 доступних міст) або географічними координатами.

OpenWeatherMap надає наступні види прогнозних даних:

- хвилинний прогноз на 1 годину;
- часовий прогноз на 4 дня;
- денний прогноз на 16 днів;
- кліматичний прогноз на 30 днів.

Пошук. Система геокодування OpenWeatherMap дозволяє знайти міста за назвою, країні, поштовим індексом або географічних координат. Пошук можливий за частиною імені міста. Для того, щоб результати пошуку були більш точними, назва міста і країни повинні бути розділені комою.

Погодні карти. OpenWeatherMap надає безліч карт погоди, включаючи карти опадів, хмарності, атмосферного тиску, температури, вітру і багато інших. Карты можуть бути підключені до мобільних додатків і веб-сайтів. Сервіс

використовує карти OpenStreetMap для побудови погодних карт.

Приклад головного інтерфейсу OpenWeatherMap можна побачити на рисунку 1.16.

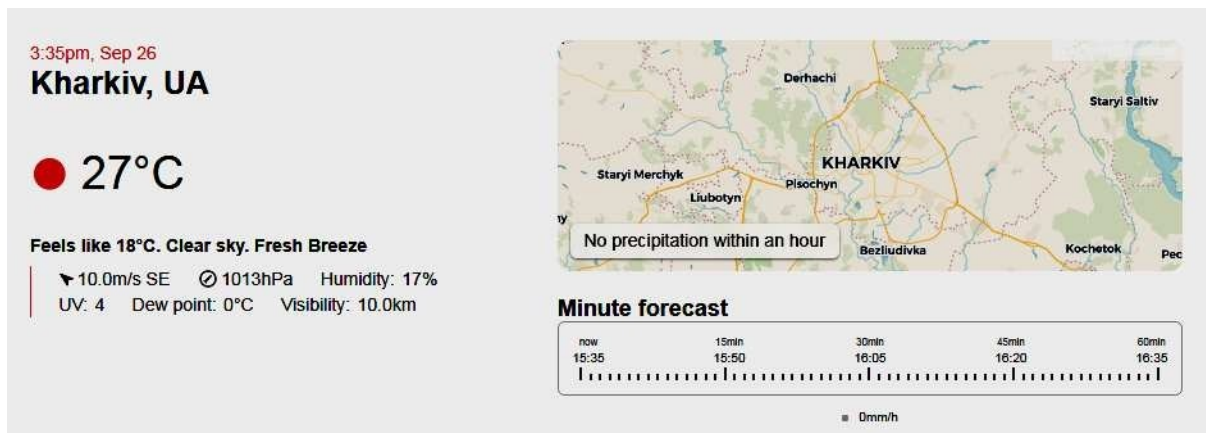


Рисунок 1.16 – Головний інтерфейс OpenWeatherMap

На сайті представлена можливість розгляду почасового прогнозу погоди (рисунок 1.17) та прогнозу погоди на 8 днів вперед (рисунок 1.18).

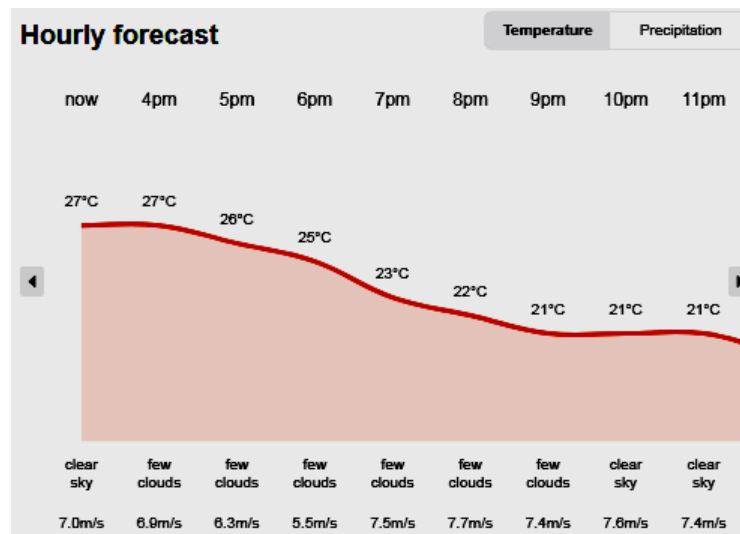


Рисунок 1.17 – Почасовий прогноз погоди OpenWeatherMap

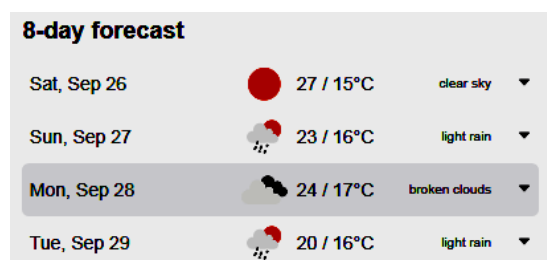


Рисунок 1.18 – Прогноз погоди на 8 днів вперед

1.3.5 Сервіс yr.no

Yr.no - це веб-сайт і мобільний додаток для прогнозування погоди та поширення інших типів метеорологічної інформації, розміщене Норвезькою радіомовною корпорацією у співпраці з Норвезьким метеорологічним інститутом.

yr.no надає прогнози погоди по всьому світу. На веб-сайті також публікуються різні типи інформації і новин, пов'язаних з погодою [5].

На додаток до даних Норвезького метеорологічного інституту yr.no використовує відкриті дані різних співробітників, таких як:

європейська організація з експлуатації метеорологічних супутників;

GeoNames;

норвезький полярний інститут;

він також збирає інформацію з різних типів приватних метеостанцій.

Головне меню навігації yr.no зображено на рисунку 1.19.

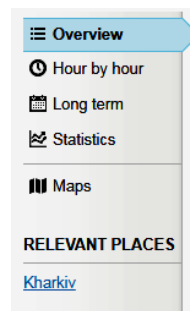


Рисунок 1.19 – Головне меню навігації yr.no

Загальний перегляд температури на поточний день та наступний день можна побачити на рисунку 1.20.

| Today, Saturday 26/09/2020 | | | | |
|-----------------------------|----------|-------|---------------|--|
| Time | Forecast | Temp. | Precipitation | Wind |
| 16:00–18:00 | | 26° | 0 mm | Moderate breeze, 7 m/s from east-southeast |
| 18:00–00:00 | | 24° | 0 mm | Moderate breeze, 6 m/s from east-southeast |
| Tomorrow, Sunday 27/09/2020 | | | | |
| Time | Forecast | Temp. | Precipitation | Wind |
| 00:00–06:00 | | 20° | 0 mm | Moderate breeze, 7 m/s from east-southeast |
| 06:00–12:00 | | 16° | 0 mm | Moderate breeze, 7 m/s from east-southeast |
| 12:00–18:00 | | 21° | 1.6 mm | Fresh breeze, 8 m/s from southeast |
| 18:00–00:00 | | 18° | 3.1 mm | Moderate breeze, 7 m/s from east-southeast |

Рисунок 1.20 – Погодна інформація на поточний і наступний день

Приклад метеографіку зображений на рисунку 1.21.

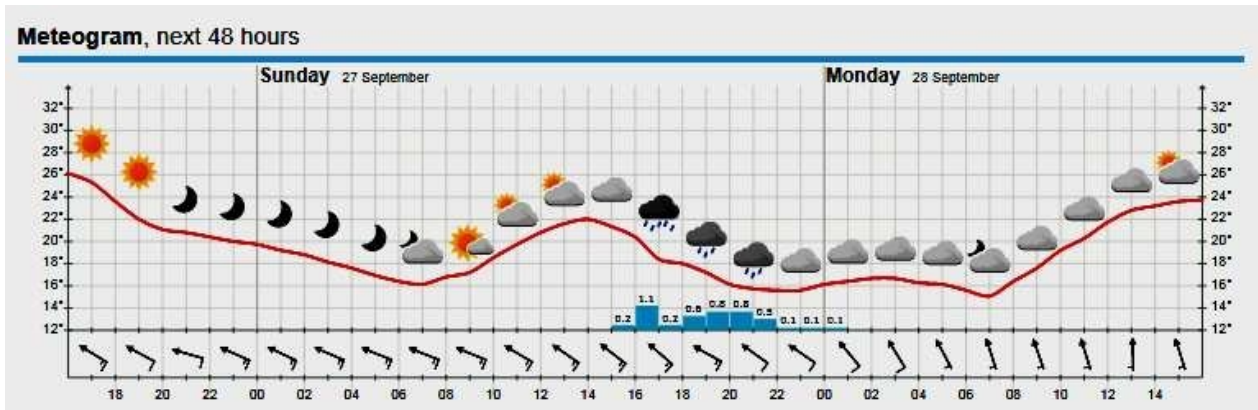


Рисунок 1.21 – Метеографік на поточний день

Погодинна погодна інформація зображена на рисунку 1.22.

| Today and tonight, 26 September 2020 | | | | |
|--------------------------------------|----------|-------|---------|--|
| Time | Forecast | Temp. | Precip. | Wind |
| Saturday 16:00 | | 26° | 0 mm | Moderate breeze, 7 m/s from east-southeast |
| Saturday 17:00 | | 25° | 0 mm | Moderate breeze, 6 m/s from east-southeast |
| Saturday 18:00 | | 24° | 0 mm | Moderate breeze, 6 m/s from east-southeast |
| Saturday 19:00 | | 22° | 0 mm | Moderate breeze, 6 m/s from east-southeast |
| Saturday 20:00 | | 21° | 0 mm | Moderate breeze, 6 m/s from east-southeast |
| Saturday 21:00 | | 21° | 0 mm | Moderate breeze, 6 m/s from east-southeast |
| Saturday 22:00 | | 20° | 0 mm | Moderate breeze, 7 m/s from east-southeast |
| Saturday 23:00 | | 20° | 0 mm | Moderate breeze, 7 m/s from east-southeast |
| Sunday 00:00 | | 20° | 0 mm | Moderate breeze, 7 m/s from east-southeast |
| Sunday 01:00 | | 19° | 0 mm | Moderate breeze, 7 m/s from east-southeast |
| Sunday 02:00 | | 19° | 0 mm | Moderate breeze, 7 m/s from east-southeast |
| Sunday 03:00 | | 18° | 0 mm | Moderate breeze, 7 m/s from east-southeast |
| Sunday 04:00 | | 18° | 0 mm | Moderate breeze, 7 m/s from east-southeast |
| Sunday 05:00 | | 17° | 0 mm | Moderate breeze, 7 m/s from east-southeast |

Рисунок 1.22 – Погодинна погодна інформація на поточний день

На сайті представлена можливість розгляду прогнозу погоди на 8 днів вперед (рисунок 1.23).

| Long term forecast | | | | | | | | |
|---------------------------------|---------------------------------|---------------------------------|-----------------------------------|-------------------------------|-----------------------------|-------------------------------|-----------------------------|-----------------------------|
| Sunday 27 September 12-18 | Monday 28 September 12-18 | Tuesday 29 September 9-15 | Wednesday 30 September 9-15 | Thursday 1 October 9-15 | Friday 2 October 9-15 | Saturday 3 October 9-15 | Sunday 4 October 9-15 | Monday 5 October 9-15 |
| | | | | | | | | |
| 21° | 22° | 17° | 14° | 12° | 13° | 13° | 14° | 14° |
| 1.6 mm | 0 mm | 0.3 mm | 0 mm | 0.4 mm | 1.1 mm | 0 mm | 0.1 mm | 0 mm |
| | | | | | | | | |

Рисунок 1.23 - Прогноз погоди на 8 днів вперед

1.6 Сервіс ForecastAdvisor

ForecastAdvisor - продукт для всіх, кому цікава погода. Він відповідає не тільки на питання «Яка буде погода?» шляхом надання прогнозів бази даних цифрових прогнозів Національної служби погоди, а також «Хто надає найбільш точний прогноз погоди для мого міста?». Це робиться в чистому, зручному для читання форматі [6].

Процент точності. Загальний відсоток точності обчислюється з відсотків точності за один-три дні для високої температури, низької температури, значка прогнозу опадів (дощу та снігу) і текстового прогнозу опадів (дощу та снігу). Точність температури - це відсоток прогнозів в межах трьох градусів. Точність опадів - це відсоток вірних прогнозів. Прогнози збираються ввечері.

Аналіз точності прогнозів погоди зображено на рисунку 1.24.

| Weather forecast accuracy for Columbus, Ohio | |
|--|----------------------------|
| Last Month (August) | Last Year (2019) |
| Weather Underground | The Weather Channel |
| 81.39% | 81.71% |
| The Weather Channel | Weather Underground |
| 80.56% | 79.04% |
| Foreca | Foreca |
| 77.22% | 78.82% |
| AccuWeather | AccuWeather |
| 76.67% | 77.42% |
| Aeris Weather | NWS Digital Forecast |
| 73.61% | 73.21% |
| NWS Digital Forecast | Wetter |
| 69.77% | 67.74% |
| Wetter | World Weather Online |
| 63.61% | 57.42% |
| World Weather Online | Weather News |
| 56.39% | 55.79% |
| Open Weathermap | Persistence |
| 55.83% | 41.06% |
| Weather News | |
| 55.56% | |
| Persistence | |
| 47.96% | |

Рисунок 1.24 – Аналіз точності прогнозів погоди

Підтвердження прогнозів і спостережень. Виконується серія аудитів, щоб гарантувати, що прогнози і спостереження, які дуже мало ймовірні або явно недійсні, виключені з аналізу. Це досягається за рахунок використання заздалегідь встановлених порогових значень, які дозволяють ідентифікувати спостереження і прогнози, що виходять за межі розумних діапазонів.

Якісні спостереження необхідні для отримання значущих і корисних даних. Спостереження витягуються з місцевої інформації з контрольованою якістю, що надається Національним центром кліматичних даних і іншими метеорологічними агентствами, дані яких відповідають певним стандартам якості. Процес контролю якості в цьому наборі даних включає більше 50 перевірок достовірності, які включають перевірки екстремальних значень, внутрішні (в рамках спостереження) перевірки узгодженості і тимчасові (у порівнянні з іншим наглядом для тієї ж станції) перевірки безперервності.

Зміни температури протягом минулого місяця і року можна побачити на рисунку 1.25.

| Temperature | Last Month | Last Year |
|---|-------------------|------------------|
| Average change in high temperature day-to-day | 2.8° | 6.2° |
| Average change in low temperature day-to-day | 5.7° | 7.5° |

Рисунок 1.25 – Зміни максимальної і мінімальної температури

Зміни опадів протягом минулого місяця і року можна побачити на рисунку 1.26.

| Precipitation | Last Month | Last Year |
|--------------------------------------|-------------------|------------------|
| Chance of dry day after a precip day | 38% | 59% |
| Chance of precip day after a dry day | 18% | 28% |

Рисунок 1.26 – Зміни посушливого дня після дня опадів та дощу після сухого дня

Координати широти / довготи або фактичні назви метеостанцій зазвичай використовуються, щоб гарантувати, що прогнози охоплюють аналізоване розташування. Цей процес застосовується в більш ніж 800 населених пунктах США. Прогнози зазвичай беруться з загальнодоступних веб-сайтів постачальників і їх API.

ForecastWatch розраховує точність і якість прогнозів погоди. ForecastWatch збирає більше 40 000 прогнозів кожен день від Accuweather, Національної служби погоди і інших для більш ніж 800 міст США і 20 міст Канади і порівнює їх з тим, що відбулося насправді. З цієї інформацією ForecastWatch допомагає підприємства, компаніям та приватним особам:

оцінити постачальників прогнозів погоди;

поліпшити процес прийняття рішень при використанні прогнозів погоди в якості вихідних даних;

покращити прогнози погоди, надаючи корисні відгуки;

підвищити якість веб-сайтів з прогнозами погоди;

надати додаткові дані для прийняття рішень і аналізу ризиків.

Прогноз погоди на 5 днів вперед зображено на рисунку 1.27.



Рисунок 1.27 – Прогноз погоди на 5 днів вперед

Погода грає важливу роль в економіці. Сотні тисяч підприємств і приватних осіб щодня використовують прогнози погоди для прийняття рішень, які мають реальний економічний ефект. Сотні тисяч підприємств і приватних осіб могли б використовувати прогнози погоди як частину свого щоденного процесу планування, якби вплив погоди на їхні підприємства або ризики використання прогнозів погоди можна було б кількісно оцінити.

ForecastAdvisor виконує розширені обчислення точності і навичок. Кліматологічні прогнози погоди - це прогнози, які передбачають, що погода буде відповідати середнім кліматичними показниками. Постійні прогнози погоди - це прогнози, які передбачають, що завтра, наступного дня і так далі будуть точно такими ж, як сьогодні. Такі прогнози вважаються «некваліфікованими», оскільки для їх складання не потрібно ніяких навичок. Вони використовуються для розрахунку точності прогнозів погоди, складених метеорологами.

Недоліки: цей ресурс оцінює виключно по містам США, тому там відсутня величезна кількість населених пунктів в базі.

1.7 Сервіс slimyhorror

Проект від slimyhorror аналізує прогнози вже більше 5000 днів поспіль.

Служба погоди BBC пропонує незліченну кількість прогнозів погоди на своєму веб-сайті. Одна з найбільш корисних функцій - це локалізовані звіти про погоду [7]. Це дає п'ятиденний прогноз погоди для будь-якого місця в Великобританії. Проблема в тому, невідомо наскільки точні ці прогнози?

П'ятиденні прогнози погоди також включають погоду на сьогодні. Таким чином, кожен день комп'ютерний сценарій зчитує останній п'ятиденний прогноз і порівнює сьогоднішню погоду з тим, що BBC сказала в своїх попередніх прогнозах.

Згодом, у міру того як програма збирає все більше і більше прогнозів погоди, можливо побачити, наскільки часто прогнози погоди виявлялися вірними.

На рисунку 1.28 показана точність прогнозів погоди BBC.

| Bournemouth | | | | | | | | |
|--------------|------|------|------|------|------|------|------|----------------------|
| | | | | | | | | Days monitored: 5816 |
| Days ahead | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| BBC accuracy | 58 % | 49 % | 45 % | 40 % | 38 % | 37 % | 33 % | 31 % |
| BBC (yearly) | 74 % | 66 % | 60 % | 55 % | 56 % | 48 % | 43 % | 40 % |
| My accuracy | 33 % | 28 % | 28 % | 26 % | 26 % | 26 % | 25 % | 24 % |

| Cambridge | | | | | | | | |
|--------------|------|------|------|------|------|------|------|----------------------|
| | | | | | | | | Days monitored: 5835 |
| Days ahead | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| BBC accuracy | 58 % | 49 % | 45 % | 40 % | 43 % | 38 % | 37 % | 37 % |
| BBC (yearly) | 66 % | 64 % | 57 % | 48 % | 45 % | 35 % | 33 % | 32 % |
| My accuracy | 34 % | 29 % | 28 % | 28 % | 27 % | 26 % | 27 % | 26 % |

| Cardiff | | | | | | | | |
|--------------|------|------|------|------|------|------|------|----------------------|
| | | | | | | | | Days monitored: 5852 |
| Days ahead | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| BBC accuracy | 59 % | 49 % | 44 % | 39 % | 36 % | 34 % | 33 % | 31 % |
| BBC (yearly) | 68 % | 64 % | 58 % | 52 % | 49 % | 43 % | 46 % | 37 % |
| My accuracy | 33 % | 28 % | 27 % | 26 % | 25 % | 24 % | 24 % | 24 % |

| Durham | | | | | | | | |
|--------------|------|------|------|------|------|------|------|----------------------|
| | | | | | | | | Days monitored: 5784 |
| Days ahead | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| BBC accuracy | 55 % | 46 % | 41 % | 38 % | 38 % | 36 % | 35 % | 36 % |
| BBC (yearly) | 65 % | 56 % | 52 % | 46 % | 42 % | 38 % | 35 % | 34 % |
| My accuracy | 33 % | 30 % | 29 % | 26 % | 26 % | 26 % | 25 % | 26 % |

Рисунок 1.28 – Точність прогнозу BBC по різним територіям

Таким чином, якщо прогноз «Часткова хмарність», то «Сонячні інтервали» також вважаються еквівалентними. Аналогічним чином, «слабкий дощ» і «мжичка» вважаються досить близькими, щоб зробити точний прогноз.

«Річна» точність заснована на результатах за останні 365 днів, щоб показати, чи стає ВВС краще або гірше згодом. Погода, передбачена ВВС на чотири дні, була точна всього на 30-40%, тобто вони помилялися приблизно два рази з трьох.

Недоліки: цей ресурс оцінює тільки прогнози від ВВС та виключно по містам Великобританії, тому там відсутня величезна кількість населених пунктів в базі.

1.8 Мета та постановка задачі

Деякий час існував проект MyCli.me, нині закритий. Однак у нього були постійні проблеми зі збором даних та рейтингова система (вибір найкращого прогнозу) була заснована на науковому методі.

Крім того існує багато початкових проектів, але всі вони обмежені або за часом, або по місту, або за провайдеру.

Таким чином, на підставі аналізу недоліків аналогів, було прийнято рішення робити свій проект. Головне завдання – збирати прогнози, збирати фактичну погоду і оцінювати процент точності на основі науково обґрунтованої формули.

Прогнози погоди повинні оцінюватися окремо для дня і ночі. Виправданість добового прогнозу визначається як середнє із значень справджуваності прогнозів на ніч і на день. Оцінка успішності прогнозів погоди за календарний період (місяць, сезон, рік і т.д.) полягає в обчисленні їх середньої справджуваності за відповідний період.

2 ОРЕТИЧНИЙ РОЗДІЛ

2.1 Опис методологій прогнозів погоди

Прогноз синоптичного положення, тобто температури, опадів, вологості і полів тиску спирається на закони, що описують рух атмосфери як рідини. Дані закони відносяться до такого розділу фізики як гідродинаміка, яка включає в себе рівняння руху, стану газів, перенесення вологості, збереження маси. Вирішуючи ці рівняння чисельними методами для конкретного стану атмосфери, можна отримати числові значення майбутніх полів тиску, температури, вітру. В якості вихідних даних для підстановки в рівняння є виміряні в даний момент параметри атмосфери. Чисельне інтегрування рівнянь зобов'язана своєю точністю сучасним обчислювальним машинам, які можуть обробляти значну кількість вхідних даних. Проте, для цілей чисельного рішення рівнянь і швидкої інтерпретації цих рішень, необхідно використовувати максимально можливі ресурси.

Прогнози діляться по завчасності періоду, на який дається прогноз:

прогноз поточної погоди – від 0 до 2 годин;

надкороткострокові – до 12 годин;

короткострокові – від 12 до 72 годин;

середньострокові – від 72 годин до 10 діб;

з розширеним терміном – від 10 до 30 діб;

довгострокові – від 30 діб до двох років; включають місячний, тримісячний (або 90–добовий), сезонний орієнтовні прогнози, які представляють собою опис усереднених метеорологічних параметрів, виражених у вигляді відхилення (девіація, коливання, аномалія) від кліматичних величин відповідно для цього місяця (не обов'язково для майбутнього місяця), для цього 90–добового періоду (не обов'язково для майбутнього 90–добового періоду), для цього сезону;

прогноз клімату – понад два роки;

прогноз клімату на рік–десятиліття (опис очікуваних параметрів кліма-

ту, пов'язаних зі зміною міжрічних і десятирічних кліматичних аномалій, а також аномалій кількох десятків років);

прогноз клімату (опис очікуваного в майбутньому клімату з урахуванням впливу природних і антропогенних факторів).

Виправданість прогнозів тим нижче, чим вище завчасність.

Прогнози погоди поділяються за типами залежно від цілей, для яких вони розроблені:

прогнози загального користування (публікуються на інтернет-сайтах, озвучених по ТБ і Радіо) містять коротку інформацію про хмарність, атмосферних опадах, атмосферні явища, вітри, температурі, вологості повітря і атмосферний тиск; режимах роботи підприємств.

авіаційні прогнози містять детальну характеристику вітру, видимості, атмосферних явищ, хмарності, температури повітря;

морські і річкові прогнози містять детальну характеристику вітру, хвилювання, атмосферних явищ, температури повітря;

сільськогосподарські (агрометеорологічні) прогнози містять детальну характеристику атмосферних опадів і температури повітря.

Три етапи завчасності прогнозу погоди: короткостроковий прогноз – 1–3 діб, середньостроковий – 4–10 діб, місяць і сезон – довгостроковий.

2.2 Опис методології прогнозів погоди різної завчасності

Прогноз погоди на короткі терміни. До моменту початку складання прогнозу на 1–3 дні є: аналіз синоптичного положення у поверхні Землі, майбутні поля тиску і температури на пару днів вперед, отримані в результаті чисельного інтегрування рівнянь термодинаміки атмосфери. За цим слідує другий етап – складання безпосередньо прогнозу погоди для певної ділянки території. Користуючись прогностичними полями тиску, синоптик визначає майбутнє становище циклонів і антициклонів, фронтів і повітряних мас. Також враховуються по-

ру року і / або місяць, які впливають на поправки в розрахунках. Для прогнозу зливових опадів і гроз використовується інформація від метеорологічних радіолокаторів. Сучасна виправданість прогнозу досить велика – в 4 рази вище, ніж при відсутності аналізу стану атмосфери. Ще більш висока виправданість штормових попереджень про небезпечні явища природи (вітер, заметіль, туман, ожеледь, заморозки, шторм, повінь); вона оцінюється приблизно в 90...95 %.

Середньостроковий прогноз погоди. Прогнози на середні терміни (4...10 днів) спираються на поточну інформацію про розвиток синоптичних процесів протягом усього сезону, а також на майбутні поля тиску і температури на 1...5 діб вперед, розраховані за допомогою інтегрування рівнянь динаміки і термодинаміки атмосфери, з використанням в якості початкових даних фактичних спостережень в момент часу, вихідний для початку інтегрування. Однак в прогнозах на 5–10 днів вже немає можливості прогнозувати зміни погоди по днях, так як неможливо простежити за зміною властивостей повітряних мас протягом усього прогнозованого періоду. Тому прогноуються середня температура (або аномалія температури) і середня кількість опадів на майбутні 10 днів, а також величини максимальної та мінімальної температури протягом майбутніх 5–7 діб. Середньостроковий прогноз складається з використанням статистичних методів, які вимагають великих обчислювальних ресурсів, тому його здійснення можливе тільки при наявності досить продуктивних ЕОМ.

Довгостроковий прогноз погоди. В даний час встановлено, що передбачуваність характеру погоди в якийсь день в даному місці не перевищує двох тижнів. Іншими словами, як би не удосконалювалися методи прогнозу погоди, принципово неможливо за межами двох тижнів передбачити конкретне явище погоди, наприклад, чи буде в такому–то місці дощ такого–то числа. Це пов'язано, по–перше, з тим, що ми ніколи не знаємо точно початковий стан атмосфери і, по–друге, що при інтегруванні рівнянь динаміки і термодинаміки атмосфери ми допускаємо певні помилки при їх чисельному рішенні, при наближеному описі фізичних процесів, що відбуваються в атмосфері, її взаємодії з поверхнею, що підстилає і космосом. Таким чином, немає надії отримати метод про-

гнозу, який би дозволив оцінити погоду по днях на майбутній місяць або сезон і тим більше на великі терміни. Це, однак, не означає, що ставити задачу прогнозу погоди на місяць або сезон безглуздо. Просто під прогнозом погоди потрібно розуміти прогноз більш загальних характеристик погоди майбутнього: найбільш часто говорять про місячних і сезонних аномаліях температури і опадів. Але завдання визначення загальних характеристик погоди на місяць і сезон поки ще далека від задовільного дозволу. Тому головне завдання метеорології в даний час – створення надійного методу довгострокового прогнозу погоди. Найперші спроби застосування для довгострокових прогнозів інерції в ході атмосферних процесів, тобто збереження знака поточної аномалії температури на місяць і сезон вперед, не дали обнадійливих результатів [15].

2.3 Оцінка похибки при прогнозуванні погоди

В процесі вимірювання неможливо відразу отримати істинне значення вимірюваної величини, і результати будь-яких двох повторних вимірювань будуть відрізнятися один від одного. Причини розбіжностей можуть бути найрізноманітнішими, але умовно їх можна розділити на дві групи.

Перша група розбіжностей результатів вимірювання – можливі зміни властивостей самого вимірюваного об'єкта. Наприклад, при вимірюванні довжини розмір предмета може змінитися під дією температури – добре відоме властивість тел розширюватися або зменшуватися при зміні температури. В інших видах вимірювання зустрічається та ж сама ситуація, під впливом температури може змінитися тиск в замкнутому просторі газу, може змінитися опір провідника, коефіцієнт відбиття поверхні.

Друга група розбіжностей – недосконалість засобів вимірювань, недосконалість методики вимірювань або недостатня кваліфікація і ретельність роботи оператора. Цю тезу досить очевидний, проте, оцінюючи похибки вимірювань, нерідко забувають про те, що ці фактори потрібно враховувати в комплексі. Вимірювальна практика показує, що грубим приладом можна отримати досить

близькі до дійсних значень результати за рахунок вдосконалення методики або мистецтва оператора. І навпаки, найточніший прилад дасть помилкові результати, якщо в процесі вимірювання не дотримуються передумови реалізації методу.

За останні десятиліття світове метеорологічне співтовариство досягло значних успіхів у розвитку технологій чисельного прогнозу погоди. Проте, повністю позбутися від помилок в прогнозах погоди неможливо. Існують такі причини помилок в прогнозі: неповнота і неточність наших знань про поточний стан атмосфери, недосконалість використовуваних прогностичних методів і моделей, проблема передбачуваності.

Головним в помилках прогнозу виступає не людський, а цілий ряд технологічних факторів.

Нерівномірність покриття метеостанціями різних територій зумовлює використання інтерполяції для таких територій, щоб привести до стандартного кроку в моделі що використовується. Найдрібніша сітка глобальних моделей 50 x 50 км. Регіональні моделі локально масштабують її до 10 x 10 км, або 2,5 x 2,5 км. На початок XXI століття на планеті Земля існувало більше 8 тис. метеорологічних станцій. Але покриття земної поверхні ними вкрай нерівномірне, значний брак відчувається в Південній півкулі та на акваторіях Світового океану віддалених від головних транспортних магістралей. Нерівномірність безпосередніх спостережень вирівнюється даними, що їх отримують з метеорологічних супутників. Але не всі метеоеlementи супутникова метеорологія передає так само якісно, як це може зробити навіть автоматична метеостанція на поверхні Землі. Наприклад, температурні показники, на відміну від хмарного покриття, особливо динаміку якого краще спостерігати з космосу. Проте супутникова апаратура і комп'ютерні розрахунки дозволяють відновлювати висотний профіль температури і вологості в часі.

Дискретність спостережень у часі. Стандартно вони робляться через 3 години. Автоматичні метеостанції можуть це робити кожні 5 секунд, але використовувані моделі все одно користуються тригодинною інтерполяцією даних.

Так само тренд розвитку тієї чи іншої метеоподії може як пришвидшуватись, так і сповільнюватись. І це може залишитись поза прогностичними обчисленнями. Синоптики використовують спеціальні прийоми коригування власних прогнозів. Одним з яких є обраховування модельних прогнозів на 12 та 24 години. Нічний прогноз перевіряють о 9 годині ранку. Якщо він досить точний, то й аналогічному полудневому можна довіряти. В іншому випадку потрібно перерахувати й внести корективи.

Наявність атмосферних процесів, що погано описуються математичними рівняннями (процеси тертя і турбулентності в нижньому 1,5 км шарі тропосфери). Для їхнього опису використовують параметризацію (математичне наближення). До недавнього часу зливи з купчасто-дощових хмар описувались саме таким чином, але в сучасних моделях параметризацію вже замінено рівняннями. Важливим невирішеним питанням залишається теорія утворення й розвитку тропічних циклонів — циклоногенезу.

Неповнота і неточність наших знань про поточний стан атмосфери. Щоб прогнозувати погоду треба, перш за все, знати, що відбувається в атмосфері зараз, в початковий момент часу. Дані гідрометеорологічних спостережень – «сировину» для розрахунку прогнозу погоди. Щоб підготувати прогноз на пару днів вперед, треба мати дані про фактичну погоду на території з масштабами декількох тисяч кілометрів. А прогноз на тиждень і далі вимагає вже інформації про те, що відбувається з погодою на всій земній кулі. При довгостроковому прогнозі доводиться розглядати практично всю кліматичну систему, в яку входять атмосфера, океан і верхній шар суші.

Поточний стан атмосфери завжди відомий лише приблизно, так як спостереження за атмосферою є неповними і неточними – спостережна мережа досить рідкісна і дані спостережень містять помилки. Великі області планети залишаються слабо освітленими даними спостережень. На території нашої країни щільність і оснащеність спостережної мережі також залишають бажати кращого. Можна підвищувати щільність мережі і зменшувати похибки вимірювань, але не нескінченно – можливості такої деталізації обмежені, тому наше знання

поточного стану атмосфери ніколи не буде повним.

Недосконалість використовуваних прогностичних методів і моделей. Основним прогностичним інструментом сьогодні стають чисельні моделі атмосфери – вони успішно відтворюють багато властивостей атмосфери і стають все більш досконалішими. Сучасна прогностична модель атмосфери з себе являє складний програмний комплекс, який вирішує систему рівнянь, що описує еволюцію атмосфери, тобто розраховує температуру, вологість, вітер і інші параметри на різних висотах в різних точках земної кулі. У моделі враховуються процеси термогідродинаміки, перетворень вологи, радіаційно–хмарні взаємодії, складні процеси в прикордонному шарі атмосфери і на кордоні з її поверхнею, що підстилає і т.д. Деякі фізичні процеси в моделях не враховуються свідомо через те, що вони менше впливають на успішність прогнозу. Їх розрахунок вимагає великих обчислювальних ресурсів (розрахунок прогнозів погоди – одна з найбільш «жадібних» до комп'ютерних ресурсів завдань). Серед факторів, що впливають можна виділити більш і менш важливі.

На сьогоднішній день автоматизовані прогностичні технології не здатні прогнозувати деякі погодні явища. Це пов'язано з тим, що багато явищ погоди, включаючи небезпечні явища, мають локальний характер і складну природу освіти, яку в даний час важко описати формально для повної автоматизації прогнозу з прийнятним рівнем успішності. З цієї причини цілий ряд явищ погоди прогнозуються в основному фахівцями–синоптиками на місцях, які добре знають умови їх утворення і розвитку в конкретному регіоні. Результати модельних розрахунків синоптики використовують як основу для складання остаточних, «офіційних» прогнозів погоди, які передбачають синтез досвіду фахівців–прогнозистів і результатів різних прогностичних технологій.

Проблема передбачуваності. Проблема передбачуваності стала усвідомлюватися вже після перших чисельних експериментів з моделювання еволюції атмосфери на довгі терміни. За межами приблизно двох тижнів помилки деталізованого по днях модельного прогнозу виростають до рівня помилок випадкового прогнозу. Так виявляються об'єктивні обмеження на можливість точно

прогнозувати конкретний хід еволюції атмосферних процесів на досить тривалих інтервалах часу. Обмеження пов'язане з тим, що початкові умови для розрахунку прогнозу завжди містять похибки і початкові помилки мають тенденцію зростати протягом періоду прогнозу через нестійкість атмосферних процесів. Практична передбачуваність атмосфери залежить від цілого ряду чинників, в тому числі і від структури атмосферних течій (тобто від поточної погодної ситуації) – в деяких випадках розвиток погодних процесів добре «проглядається» на кілька діб вперед, а буває, що і прогноз на завтра виявляється вельми крихким.

Все це не означає, що ми нічого не можемо сказати про майбутнє атмосфери за межами пари тижнів. Можемо, але прогнози на довгі терміни формулюються в іншій формі і вимоги до них інші – як правило для довгострокових прогнозів використовуються імовірнісна формулювання і представлення результатів у термінах середніх за період величин. Формально нічого не варто деталізувати, наприклад, прогноз на кілька тижнів вперед по добі або навіть по хвилинах, але ця «точність» буде незабезпеченої реальними можливостями сучасних прогностичних технологій [16].

2.4 Обґрунтування вибору засобів реалізації програмного продукту

2.4.1 JavaScript

JavaScript, є мовою програмування, яка відповідає специфікації ECMAScript.

Основні архітектурні риси: динамічна типізація, слабка типізація, автоматичне керування пам'яттю, прототипне програмування, функції як об'єкти першого класу.

Поряд з HTML та CSS, JavaScript є однією з основних технологій Всесвітньої мережі. JavaScript забезпечує інтерактивні веб-сторінки та є важливою частиною веб-додатків. Переважна більшість веб-сайтів використовують його для

поведінки на стороні клієнта, і всі основні веб-браузери мають спеціальний механізм JavaScript для його виконання.

Як мова багатопарадигми, JavaScript підтримує функціональні та імперативні стилі програмування. Він має інтерфейси прикладного програмування (API) для роботи з текстом, датами, регулярними виразами, стандартними структурами даних та об'єктною моделлю документа (DOM). Однак сама мова не включає жодного вводу/виводу, наприклад мережевих, сховищних чи графічних засобів, оскільки середовище хоста (зазвичай веб-браузер) забезпечує ці API.

Спочатку JavaScript використовувався лише у веб-браузерах, але зараз він вбудовані в деякі сервери, як правило, через Node.js. Вони також вбудовані в різноманітні програми, створені за допомогою фреймворків.

Незважаючи на те, що між JavaScript та Java є подібність, включаючи назву мови, синтаксис та відповідні стандартні бібліотеки, ці дві мови відрізняються за дизайном та функціоналом.

JavaScript є мовою сценаріїв на стороні клієнта в Інтернеті, для цього його використовує 95% веб-сайтів. Скрипти вбудовані в документи HTML або включені до них і взаємодіють із DOM. Усі основні веб-браузери мають вбудований механізм JavaScript, який виконує код на пристрої користувача.

В JavaScript:

всі ідентифікатори чутливі до регістру;

в назвах змінних можна використовувати букви, підкреслення, символ долара, арабські цифри;

назви змінних не можуть починатися з цифри;

для оформлення однорядкових коментарів використовуються //, багаторядкові і внутрішторочні коментарі починаються з /* і закінчуються */.

Об'єктна модель браузера. Об'єктна модель браузера – браузер-специфічна частина мови, що є прошарком між ядром і об'єктною моделлю документа. Основне призначення об'єктної моделі браузера – управління вікнами браузера і забезпечення їх взаємодії. Кожне з вікон браузера представляється

об'єктом window, центральним об'єктом DOM.

Крім управління вікнами, в рамках об'єктної моделі браузера, браузерами зазвичай забезпечується підтримка наступних сутностей:

- управління кадрами;
- підтримка затримки у виконанні коду і зациклення з затримкою;
- системні діалоги;
- управління адресою відкритої сторінки;
- управління інформацією про браузері;
- управління інформацією про параметри монітора;
- обмежене управління історією перегляду сторінок;
- підтримка роботи з HTTP cookie.

Об'єктна модель документа. Об'єктна модель документа – інтерфейс програмування додатків для HTML і XML-документів. Згідно DOM, документ (наприклад, веб-сторінка) може бути представлений у вигляді дерева об'єктів, що володіють рядом властивостей, які дозволяють виробляти з ним різні маніпуляції:

- генерація і додавання вузлів;
- отримання вузлів;
- зміна вузлів;
- зміна зв'язків між вузлами;
- видалення вузлів [17].

2.4.2 Node.js

Node або Node.js – програмна платформа, заснована на движку V8, що перетворює JavaScript з вузькоспеціалізованого мови в мову загального призначення. Node.js дозволяє розробникам використовувати JavaScript для написання інструментів командного рядка та для сценаріїв на стороні сервера – запуску сценаріїв на стороні сервера для створення динамічного вмісту веб-сторінки до того, як сторінка буде відправлена у веб-браузер користувача. Отже, Node.js

об'єднує розробку веб-додатків навколо однієї мови програмування, а не різних мов для серверних та клієнтських сценаріїв.

Хоча `.js` є стандартним розширенням імені файлу для коду JavaScript, назва "Node.js" не посилається на певний файл у цьому контексті, а є просто назвою продукту. Node.js має керовану подіями архітектуру, здатну до асинхронного вводу/виводу. Ці варіанти дизайну спрямовані на оптимізацію пропускну здатності та масштабованості у веб-додатках з багатьма операціями введення-виведення, а також для веб-додатків у режимі реального часу.

Архітектура платформи. Node.js забезпечує програмування подій на веб-серверах, що дозволяє розвивати швидкі веб-сервери в JavaScript. Розробники можуть створювати масштабовані сервери без використання потоків, використовуючи спрощену модель програмування, керованого подіями, яке використовує зворотні виклики для сигналізації про завершення завдання. Node.js пов'язує простоту мови сценаріїв (JavaScript) з потужністю мережевого програмування Unix.

Node.js був побудований поверх механізму JavaScript Google V8 JavaScript, оскільки він був відкритим за ліцензією BSD. Він знає основи Інтернету, такі як HTTP, DNS, TCP.

Для Node.js існує тисячі бібліотек з відкритим кодом, більшість з яких розміщені на веб-сайті npm. Спільнота розробників Node.js має два основних списки розсилки та IRC-канал `# node.js` на freenode.

Було розроблено веб-фреймворки, щоб пришвидшити розробку додатків. До таких фреймворків належать Connect, Express.js, Socket.IO, Feathers.js, Koa.js, Hapi.js, Sails.js, Meteor, Derby та багато інших. Також створено різні пакети для взаємодії з іншими мовами або середовищами виконання, такими як Microsoft .NET.

Сучасні середовища розробки середовища для робочого столу надають функції редагування та налагодження спеціально для програм Node.js. Такі IDE включають Atom, Brackets, JetBrains WebStorm, Microsoft Visual Studio (з інструментами Node.js для Visual Studio, або TypeScript з визначеннями вузлів,

NetBeans, Nodeclipse, Enide Studio та Visual Studio Code. Деякі веб-інтерфейси IDE також підтримують Node.js, такі як Codeanywhere, Codenvy, Cloud9 IDE, Koding та редактор візуального потоку в Node-RED.

Node.js підтримується на багатьох хмарних платформах хостингу, таких як Jelastic, Google Cloud Platform, AWS Elastic Beanstalk, Joyent та інших [18].

2.4.3 Express.js

Express.js, або просто Express, – це мінімалістичний і гнучкий веб-фреймворк для додатків Node.js, що надає великий набір функцій для мобільних і веб-додатків, випущений як безкоштовне програмне забезпечення з відкритим кодом за ліцензією MIT. Його називали де-факто стандартною серверною структурою для Node.js.

Оригінальний автор, TJ Holowaychuk, описав його як сервер, натхненний Sinatra, що означає, що він відносно мінімальний з багатьма функціями, доступними як плагіни. Express може бути backend'ом для програмного стека, разом з базою даних MongoDB і каркасом Vue.js, React або AngularJS для frontend'a.

Маючи в своєму розпорядженні безліч службових методів HTTP і проміжних оброблювачів, створити надійний API можна швидко і легко. Express надає тонкий шар фундаментальних функцій веб-додатків, які не заважають працювати з функціями Node.js [19].

2.4.4 React

React (також відомий як React.js або ReactJS) – це бібліотека JavaScript з відкритим кодом, інтерфейс, для побудови користувальницьких інтерфейсів або компонентів інтерфейсу. Він підтримується Facebook та спільнотою окремих розробників та компаній. React можна використовувати як основу при розробці односторінкових або мобільних додатків. Однак React займається лише наданням даних в DOM, тому створення React-програм зазвичай вимагає викорис-

тання додаткових бібліотек для управління станом та маршрутизації. Redux та React Router є відповідними прикладами таких бібліотек.

React код складається з сутностей, які називаються компонентами. Компоненти можуть бути відтворені до певного елемента в DOM за допомогою бібліотеки React DOM. Під час візуалізації компонента можна передати значення, які відомі як "prop". Два основних способи оголошення компонентів у React – це функціональні компоненти та компоненти на основі класів.

Крім вхідних даних (доступних через `this.props`), компонент підтримує внутрішні дані стану (доступні через `this.state`). Коли дані стану компонента зміняться, React ще раз викличе `render ()` і оновить відрендерену розмітку.

Оскільки логіка компонента написана на JavaScript, а не міститься в шаблонах, можна з легкістю передавати найрізноманітніші дані по всьому додатком і тримати стан поза DOM [20].

Віртуальний DOM. Ще однією помітною особливістю є використання віртуальної об'єктної моделі документа або віртуальної DOM. React створює кеш-структуру даних в пам'яті, обчислює отримані різниці, а потім ефективно оновлює відображуваний DOM браузера. Це дозволяє програмісту писати код так, ніби вся сторінка відображається при кожній зміні, тоді як бібліотеки React відображають лише підкомпоненти, які насправді змінюються. Цей вибірковий візуалізація забезпечує значне підвищення продуктивності. Це економить зусилля з перерахунку стилю CSS, макета сторінки та рендерингу для всієї сторінки.

Методи життєвого циклу. Методи життєвого циклу використовують форму підключення, що дозволяє виконувати код у задані точки протягом життя компонента:

`shouldComponentUpdate` дозволяє розробнику запобігти непотрібному повторному рендерингу компонента, повертаючи значення `false`, якщо візуалізація не потрібна;

`componentDidMount` викликається, як тільки компонент "змонтова-

ний" (компонент був створений в інтерфейсі користувача, часто шляхом асоціювання його з DOM-вузлом). Це зазвичай використовується для запуску завантаження даних з віддаленого джерела через API;

`componentWillUnmount` викликається безпосередньо перед тим, як компонент буде зруйнований або "демонтований". Це зазвичай використовується для очищення залежностей від компонента, що вимагають ресурсів, і які не будуть просто видалені при демонтажі компонента (наприклад, видалення будь-яких екземплярів `setInterval ()`, які пов'язані з компонентом, або "eventListener", встановленого на " документ "через наявність компонента);

`render` є найважливішим методом життєвого циклу і єдиним необхідним у будь-якому компоненті. Зазвичай він викликається кожного разу, коли стан компонента оновлюється, що повинно відображатися в інтерфейсі користувача.

JSX, або JavaScript XML, є розширенням синтаксису мови JavaScript. Подібно до зовнішнього вигляду HTML, JSX надає спосіб структурувати візуалізацію компонентів за допомогою синтаксису, знайомого багатьом розробникам. Компоненти React, як правило, пишуться за допомогою JSX, хоча їх не обов'язково (компоненти також можуть бути написані на чистому JavaScript). JSX подібний до іншого синтаксису розширення, створеного Facebook для PHP під назвою XHP.

Вкладені елементи. Кілька елементів на одному рівні потрібно обернути в один елемент React, такий як елемент `<div>`, фрагмент, позначений `<Fragment>` або у його скороченому вигляді `<>`, або повернути у вигляді масиву.

Атрибути. JSX надає ряд атрибутів елементів, призначених для віддзеркалення тих, що надаються HTML. Спеціальні атрибути також можуть бути передані компоненту. Всі атрибути будуть отримані компонентом як `props`.

Вирази JavaScript. Вирази JavaScript можна використовувати всередині JSX із фігурними дужками `{}`.

Архітектура за межами HTML. Основна архітектура React застосовується

крім рендеринга HTML у браузері. Наприклад, Facebook має динамічні діаграми, які відображаються на теги `<canvas>`, а Netflix і PayPal використовують універсальне завантаження для надання однакових HTML-кодів як на сервері, так і на клієнті.

React Hooks. Hooks – це функції, які дозволяють розробникам "підключатись" до стану реагування та функцій життєвого циклу з функціональних компонентів. Вони роблять код зрозумілим. Хуки не працюють всередині класів – вони дозволяють використовувати React без класів.

React надає кілька вбудованих хуків, таких як `useState`, `useContext`, `useReducer` і `useEffect` та інші. Усі вони зазначені в Довідковому документі по API Hooks. `useState` та `useEffect`, які найчастіше використовуються, призначені для управління станами та побічними ефектами відповідно у React Components.

React не намагається надати повну "бібліотеку програм". Він розроблений спеціально для побудови користувальницьких інтерфейсів, тому не включає багато інструментів, які деякі розробники можуть вважати необхідними для створення додатка. Це дозволяє вибрати, яку бібліотеку розробник воліє виконувати такі завдання, як забезпечення доступу до мережі або локального зберігання даних.

React може працювати на сервері, використовуючи Node.js і на мобільних платформах, використовуючи React Native [21].

2.4.5 Bootstrap

Bootstrap – це безкоштовний фреймворк CSS з відкритим кодом, спрямований на адаптивну мобільну веб-розробку. Він містить шаблони дизайну на основі CSS та (за бажанням) JavaScript для типографіки, форм, кнопок, навігації та інших компонентів інтерфейсу.

Bootstrap – це бібліотека HTML, CSS та JS, яка зосереджена на спрощенні розробки інформативних веб-сторінок (на відміну від веб-програм). Основною метою його додавання до веб-проекту є застосування вибору Bootstrap кольору,

розміру, шрифту та макета до цього проекту. Після додавання до проекту Bootstrap надає базові визначення стилів для всіх елементів HTML. Результат – однаковий вигляд для таблиць та елементів форми у веб-браузерах.

Крім того, розробники можуть скористатися перевагами класів CSS, визначених у Bootstrap, для подальшої настройки зовнішнього вигляду їх вмісту. Наприклад, Bootstrap підготував для світлих та темних кольорів таблиці, заголовки сторінок, видатніші цитати та текст із виділенням.

Компоненти JavaScript надають додаткові елементи інтерфейсу користувача, такі як діалогові вікна, підказки та каруселі. Кожен компонент Bootstrap складається із структури HTML, декларацій CSS, а в деяких випадках супроводжує код JavaScript. Вони також розширюють функціональність деяких існуючих елементів інтерфейсу, включаючи, наприклад, функцію автоматичного заповнення для полів введення.

Найвизначнішими компонентами Bootstrap є компоненти макета, оскільки вони впливають на цілу веб-сторінку. Основний компонент макета називається "Контейнер", оскільки в ньому розміщується кожен інший елемент сторінки [22].

Основні інструменти Bootstrap:

сітки – заздалегідь задані розміри колонок, які можна відразу ж використовувати, наприклад, ширина колонки 140 px відноситься до класу `.span2` (`.col-md-2` в третій версії фреймворка), який можна використовувати в CSS-описі документа;

шаблони – фіксований або адаптативний шаблон документа;

типографіка – опис шрифтів, визначення деяких класів для шрифтів;

медіа – надає деяке упорядкування зображень та відео;

таблиці – засоби оформлення таблиць;

форми – класи для оформлення форм і деяких подій, що відбуваються з

ними;

навігація – класи оформлення для панелей, вкладок, переходу по сторінках, меню і панелі інструментів;

алерт – оформлення діалогових вікон, підказок і спливаючих вікон.

Відновлений за допомогою React. React–Bootstrap замінює Bootstrap JavaScript. Кожен компонент був побудований з нуля як справжній компонент React, без непотрібних залежностей, таких як jQuery.

Будучи однією з найстаріших бібліотек React, React–Bootstrap еволюціонував і розвивався поряд з React, що робить його чудовим вибором як основу вашого інтерфейсу.

Bootstrap по суті. Створений з урахуванням сумісності, Bootstrap охоплює ядро початкового завантаження та прагне бути сумісними з найбільшою в світі екосистемою інтерфейсу.

Покладаючись повністю на таблицю стилів Bootstrap, React–Bootstrap просто працює з тисячами вже улюблених тем Bootstrap.

Доступний за замовчуванням. Модель компонента React дає нам більше контролю над формою та функціями кожного компонента.

Кожен компонент реалізований з урахуванням доступності. Результат – набір доступних за замовчуванням компонентів, порівняно з тим, що можливо із звичайного Bootstrap [23].

2.4.6 MySQL

MySQL – це система управління реляційними базами даних із відкритим кодом (СУБД). Реляційна база даних організовує дані в одну або кілька таблиць даних, в яких типи даних можуть бути пов'язані між собою; ці відносини допомагають структурувати дані. SQL – це мова, яку програмісти використовують для створення, модифікації та вилучення даних з реляційної бази даних, а також контролю доступу користувачів до бази даних. На додаток до реляційних баз даних та SQL, СУБД, як MySQL, працює з операційною системою для реалізації реляційної бази даних у системі зберігання даних комп'ютера, управляє користувачами, забезпечує доступ до мережі та полегшує тестування цілісності бази даних та створення резервних копій.

MySQL має окремі клієнти, які дозволяють користувачам безпосередньо

взаємодіяти з базою даних MySQL за допомогою SQL, але частіше MySQL використовується з іншими програмами для реалізації програм, які потребують можливості реляційних баз даних. MySQL є компонентом стеку програмного забезпечення для веб-додатків. MySQL використовується багатьма веб-додатками, керованими базами даних, включаючи Drupal, Joomla, phpBB та WordPress. MySQL також використовується багатьма популярними веб-сайтами, включаючи Facebook, Flickr, MediaWiki, Twitter, та YouTube.

MySQL портована на велику кількість платформ: AIX, BSDi, FreeBSD, HP-UX, Linux, macOS, NetBSD, OpenBSD, OS / 2 Warp, SGI IRIX, Solaris, SunOS, SCO OpenServer, UnixWare, Tru64, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Server 2003, WinCE, Windows Vista, Windows 7 і Windows 10. Існує також порт MySQL до OpenVMS. На офіційному сайті СУБД для вільного завантаження надаються не тільки вихідні коди, а й відкомпілювалися і оптимізовані під конкретні операційні системи готові виконувати модулі СУБД MySQL.

MySQL має API та конектори для мов Delphi, C, C ++, Ейфель, Java, Лисп, Perl, PHP, Python, Ruby, Smalltalk, Компонентний Паскаль, Tcl і Lua, бібліотеки для мов платформи .NET, а також забезпечує підтримку для ODBC за допомогою ODBC-драйвера MyODBC.

Підтримку можна отримати в офіційному посібнику. Безкоштовна підтримка додатково доступна на різних каналах IRC та форумах. Oracle пропонує платну підтримку через свої продукти MySQL Enterprise. Вони відрізняються за обсягом послуг та ціною. Крім того, існує низка незалежних організацій, які надають підтримку та послуги.

MySQL вважається гарним рішенням для малих і середніх застосувань. Коди сервера компілюються на багатьох платформах. У UNIX-системах розкриваються найповніші можливості сервера, тому що там є підтримка багатопотосності, яка підвищує продуктивність систему в цілому.

Можливості сервера MySQL:

простота у встановленні та використанні;

підтримується необмежена кількість користувачів, що одночасно працюють із БД;

кількість записів у таблицях може досягати 50 млн;

висока швидкість виконання команд;

наявність простої і ефективної системи безпеки.

MySQL Workbench. MySQL Workbench – це інтегроване середовище для MySQL. Він був розроблений MySQL AB і дозволяє користувачам графічно управляти базами даних MySQL та візуально проектувати структури баз даних.

MySQL Workbench доступний у двох виданнях: звичайному безкоштовному і відкритому вихідному коді Community Edition, який можна завантажити з веб-сайту MySQL, та запатентованому Standard Edition, який розширює та вдосконалює набір функцій Community Edition [24].

3 ПРОЕКТНИЙ РОЗДІЛ

3.1 Формування вимог до системи

У процесі розробки інформаційної системи були створені такі вимоги:

система повинна надавати користувачу детальний опис моделей та прогнозу погоди;

система повинна застосовувати запити для отримання інформації з баз

даних;

система повинна мати можливість легкого масштабування;

система має опрацьовувати запити користувача зі швидкістю часу відгуку не більше 3 секунд;

система повинна мати функціонал обробки помилок з виведенням зрозумілого повідомлення про проблему;

система повинна трансформувати отримані дані в узагальнені об'єкти.

3.2 Розробка структури бази даних

Для створення автоматизованої інформаційної системи було обрано систему керування базами даних MySQL. Схема бази даних у MySQL Workbench представлена на рисунку 3.1.

Таблиця «forecast_type» потрібна для визначення короткостроковий чи довгостроковий був прогноз. Приклад записів таблиці наведено у таблиці 3.1.

Таблиця 3.1 – Приклад записів таблиці «forecast_type»

| Id type | Type description |
|---------|------------------|
| 1 | 1 day |
| 2 | 3 days |
| 3 | 5+ days |

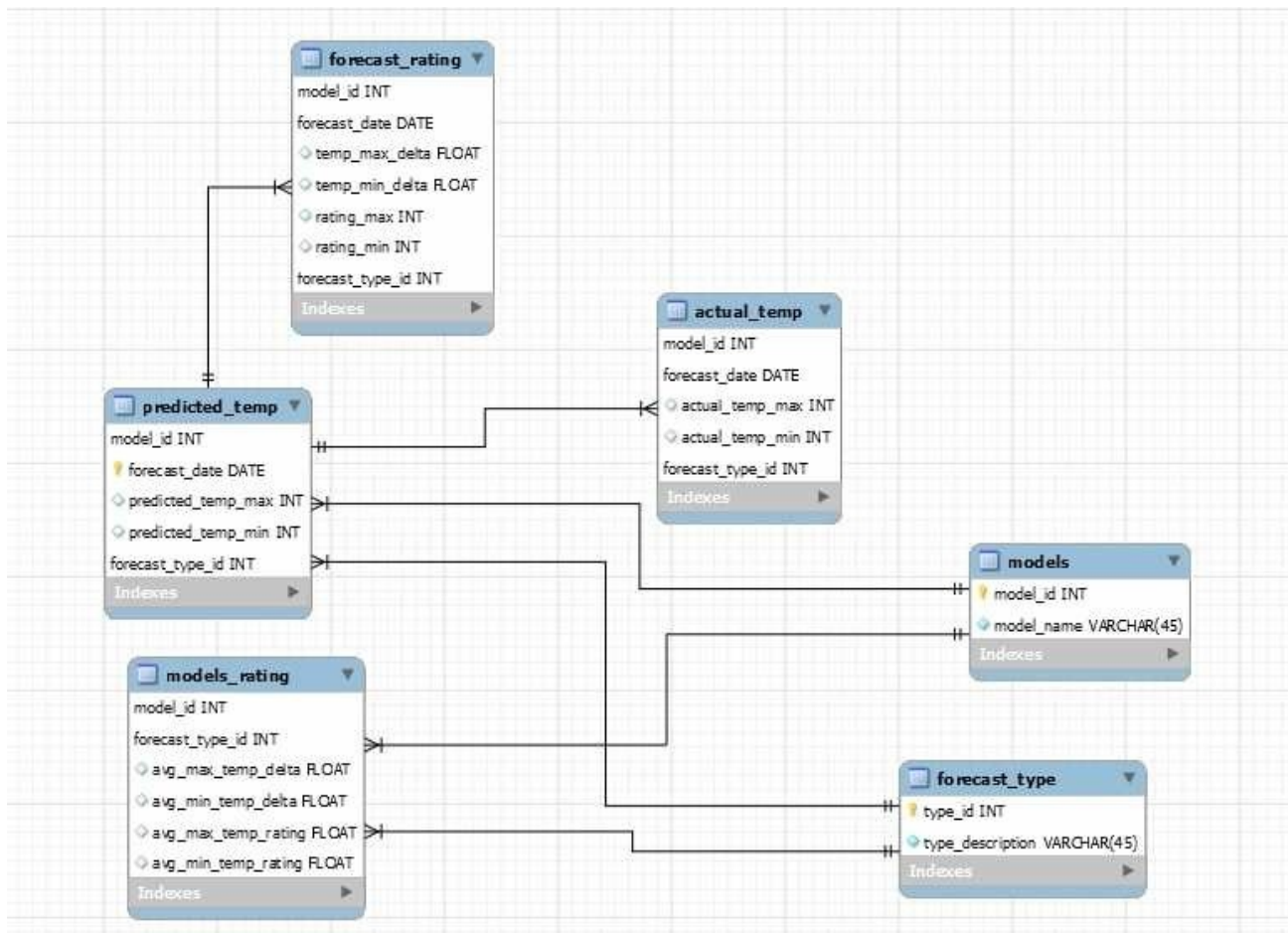


Рисунок 3.1 – Схема бази даних у MySQL Workbench

У таблицю «predicted_temp» заносяться прогностичні дані з різних моделей щодо прогнозів погоди на 1, 3 та 5+ днів вперед. Приклад записів в ній наведено в таблиці 3.2.

Таблиця 3.2 – Приклад записів таблиці «predicted_temp»

| Model id | Forecast date | Predicted maximum temperature | Predicted minimum temperature | Type id |
|----------|---------------|-------------------------------|-------------------------------|---------|
| 1 | 2020-11-16 | -2 | -4 | 1 |
| 2 | 2020-11-16 | 0 | -5 | 1 |
| 1 | 2020-11-19 | -1 | -5 | 2 |

Коли настають терміни прогнозу, фактична погода заносяться в таблицю 3.3 «actual_temp».

Таблиця 3.3 – Приклад записів таблиці «actual_temp»

| Model id | Forecast date | Predicted maximum temperature | Predicted minimum temperature | Type id |
|----------|---------------|-------------------------------|-------------------------------|---------|
| 1 | 2020-11-16 | -3 | -5 | 1 |
| 2 | 2020-11-16 | -1 | -4 | 1 |
| 1 | 2020-11-19 | 1 | -4 | 2 |

Всі прогнози виконані в одні і ті ж дні, маючи одні й ті ж вихідні дані і синоптичну ситуацію.

З 15 листопада цього року було порівняно прогнози не тільки максимальної, а й мінімальної температури. Під максимальною температурою тут мається на увазі максимум, зафіксований в період з 9 до 21 години, тобто днем.

Система виноує розрахунок середньої похибки температури (різниця фактичної і прогнозованої температури по модулю).

На результаті середньої похибки температури розраховується оцінка прогнозу.

Критерії оцінок наступні: прогноз на 5 – помилка не більше 1.5 градусів, 4 – не більше 2.5 градусів, 3 – 3.5 градуса. Помилка більш 3.5 градусів – це оцінка 2.

Всі розрахунки записуються в таблицю «forecast_rating» (таблиця 3.4).

Таблиця 3.4 – Приклад записів таблиці «forecast_rating»

| Model id | Forecast date | Maximum temperature delta | Minimum temperature delta | Maximum temperature rating | Minimum temperature rating | Forecast type id |
|----------|---------------|---------------------------|---------------------------|----------------------------|----------------------------|------------------|
| 1 | 2020-11-16 | 1 | 1 | 5 | 5 | 1 |
| 2 | 2020-11-19 | 3 | 1 | 2 | 5 | 2 |
| 1 | 2020-11-20 | 1 | 3 | 5 | 2 | 3 |

Розрахунок загальної похибки, оцінки моделі та типу прогнозу записується в таблицю «models_rating» (таблиця 3.5).

Таблиця 3.5 – Приклад записів таблиці «models_rating»

| Model id | Average maximum temperature delta | Average minimum temperature delta | Average Maximum temperature rating | Average Minimum temperature rating | Forecast type id |
|----------|-----------------------------------|-----------------------------------|------------------------------------|------------------------------------|------------------|
| 1 | 1.56 | 0.78 | 4.22 | 5 | 1 |
| 2 | 1.67 | 1.67 | 4 | 4 | 2 |
| 1 | 1 | 1.33 | 4.67 | 4 | 3 |

4 ПРАКТИЧНИЙ РОЗДІЛ

4.1 Реалізація прототипу системи

За допомогою IDE Visual Studio Code та мови JavaScript була реалізована fullstack система.

В результаті була сформована така структура пакетів:

«assets» – пакет для зберігання зображень і шрифтів, які будуть використовуватися в системі;

«common» – пакет для зберігання констант, які відображаються на інтерфейсі користувача та значення, які використовуються в логічних операціях системи;

«components» – пакет пласту моделей, який слугує для відображення частин інтерфейсу користувача;

«containers» – пакет з класами реалізації усіх зазначених у проектуванні сутностей, а також головний клас «Content»;

«server» – пакет сервісного пласту для класів, що виконують задачу роботи з базою даних та зв'язування frontend та backend частин системи;

«service» – пакет з класами методів, кожен з яких використовується в процесі обробки у класах пакету «containers».

Структуру класів наведено на рисунках 4.1, 4.2 та 4.3.

4.2 Розробка front-end частини

Front-end частини була розроблена за допомогою бібліотеки JavaScript – React.js. Проект був розділений на логічну частину (контейнери) та візуальну частину (компоненти).

Саме вона отримує запити від користувача та за необхідності передає їх на сервер для обробки. Після цього повертає отриманий результат користувачеві у зрозумілому та сприятливому вигляді.

```

  ▾ src
    > assets \ fonts \ Roboto
    ▾ common
      JS constants.js
      JS databaseValues.js
      JS menuValues.js
    ▾ components
      ▾ ActualTempTable
        # ActualTempTable.css
        JS ActualTempTable.js
      ▾ ForecastRatingTable
        # ForecastRatingTable.css
        JS ForecastRatingTable.js
      ▾ Menu
        # Menu.css
        JS Menu.js
      ▾ Model
        # Model.css
        JS Model.js
      ▾ PredictedTempTable
        # PredictedTempTable.css
        JS PredictedTempTable.js

```

Рисунок 4.1 – Структура пакетів «assets», «common» та «components»

```

  ▾ src
    > assets \ fonts \ Roboto
    > common
    > components
    ▾ containers
      ▾ Accuweather
        JS Accuweather.js
      ▾ Content
        JS Content.js
      ▾ CurrentView
        JS CurrentView.js
      ▾ Meteo
        JS Meteo.js
      ▾ OpenWeatherMap
        JS OpenWeatherMap.js
      ▾ Tables
        JS Tables.js
      ▾ Wunderground
        JS Wunderground.js
      ▾ Yandex.Pogoda
        JS YandexPogoda.js
      ▾ Yr.no
        JS YrNo.js

```

Рисунок 4.2 – Структура пакета «containers»

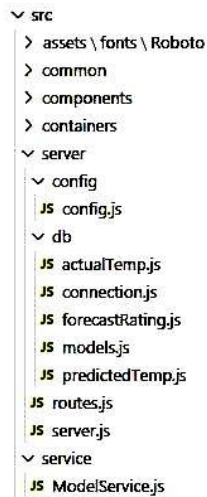


Рисунок 4.3 – Структура пакетів «server» та «service»

Розглянемо відправлення запиту в базу даних на приклади таблиці «predicted_temp» (рисунок 4.4).

```

selectPredictedTempTable() {
  fetch("/api/predictedTemp/", {
    method: "GET",
  })
  .then((response) => response.json())
  .then((jsonData) => {
    this.setState({
      predictedTempTable: jsonData,
    });
  })
  .catch((error) => {
    console.error(error);
  });
}
}

```

Рисунок 4.4 – Метод selectPredictedTempTable
для відправлення запиту в базу даних

Цей метод використовує інтерфейс JavaScript fetch API, який слугує для роботи із запитом та відповідями HTTP.

Fetch звертається до локальної URL localhost:3000/predictedTemp, яка зазначена в back-end частині та повертає JSON файл з записами таблиці, які після буде оброблено та виведено на інтерфейс користувача в зрозумілому вигляді.

```

<Accuweather
  predictedTempTable={predictedTempTable}
  actualTempTable={actualTempTable}
  modelsRatingTable={modelsRatingTable}
/>

```

Рисунок 4.5 – Передача записів таблиці «predicted_temp» в контейнер
Accuweather для обробки

```

<Tables
  predictedTempTable={predictedTempTable}
  actualTempTable={actualTempTable}
  forecastRatingTable={forecastRatingTable}
/>

```

Рисунок 4.6 – Передача записів таблиці «predicted_temp» в компонент Tables для відображення на інтерфейсі користувача

Наступним кроком буде розрахунок записів таблиці «forecast_rating». Для цього був написаний метод sortForecastRating, який працює с записами таблиць «predicted_temp» та «actual_temp», які було отримано заздалегіть (рисунок 4.7).

```

sortForecastRating() {
  const { predictedTempTable, actualTempTable } = this.state;

  let forecastRatingArray = [];

  predictedTempTable.forEach((predictedTempRecord) => {
    actualTempTable.some((actualTempRecord) => {
      if (
        actualTempRecord.model_id === predictedTempRecord.model_id &&
        actualTempRecord.forecast_date ===
          predictedTempRecord.forecast_date &&
        actualTempRecord.forecast_type_id ===
          predictedTempRecord.forecast_type_id
      ) {
        let tempMaxDelta = Math.abs(
          actualTempRecord.actual_temp_max -
            predictedTempRecord.predicted_temp_max
        );
        let tempMinDelta = Math.abs(
          actualTempRecord.actual_temp_min -
            predictedTempRecord.predicted_temp_min
        );
        forecastRatingArray.push({
          model_id: predictedTempRecord.model_id,
          model_name: predictedTempRecord.model_name,
          forecast_date: new Date(
            predictedTempRecord.forecast_date
          ).toLocaleDateString("en-CA"),
          forecast_type_id: predictedTempRecord.forecast_type_id,
          temp_max_delta: tempMaxDelta,
          temp_min_delta: tempMinDelta,
          rating_max: this.calculateForecastRating(tempMaxDelta),
          rating_min: this.calculateForecastRating(tempMinDelta),
          type_description: predictedTempRecord.type_description,
        });
      }
    });
  });
  return 0;
}
}

```

Рисунок 4.7 – Метод sortForecastRating

Метод виконує задачу розрахунку похибки при прогнозуванні температури та оцінки прогнозу та формує об'єкт записів таблиці.

Наступний метод insertForecastRecords відправляє HTTP запит PUT для

запису сформованого об'єкту в таблицю «forecast_rating» (рисунок 4.8).

```
insertForecastRatingRecords() {
  const { forecastRatingTable } = this.state;

  forecastRatingTable.forEach((forecastRatingRecord) => {
    fetch("/api/forecastRating/", {
      method: "put",
      headers: {
        "Content-Type": "application/json",
      },
      body: JSON.stringify(forecastRatingRecord),
    });
  });
}
```

Рисунок 4.8 – Метод insertForecastRecords

Наступний метод insertModelsRecords також відправляє HTTP запит PUT для розрахунку та запису середньої похибки прогнозу температури та середньої оцінки кожної с моделей в таблицю «models_rating». Головний функціонал цього методу відбувається в SQL запиті (рисунок 4.9).

```
insertModelsRatingRecords() {
  fetch("/api/modelsRating/", {
    method: "put",
    headers: {
      "Content-Type": "application/json",
    },
  });
}
```

Рисунок 4.9 – Метод insertForecastRecords

Фінальним кроком буде метод selectModelsRatingTable, який викликається після натиску кнопки меню на інтерфейсі користувача (рисунок 4.10, 4.11).

```
<Col
  className={`menu_models-rating-button button ${activeButton(
    MenuValues.MODELS_RATING
  )}`}
  onClick={() => loadView(MenuValues.MODELS_RATING)}
>
  {MenuConstants.MODELS_RATING_VIEW}
</Col>
```

Рисунок 4.10 – Виклик методу по натиску на кнопку меню

```

selectModelsRatingTable() {
  fetch("/api/modelsRating/", {
    method: "GET",
  })
  .then((response) => response.json())
  .then((jsonData) => {
    this.setState({
      modelsRatingTable: jsonData,
    });
  })
  .catch((error) => {
    console.error(error);
  });
}
}

```

Рисунок 4.11 – Метод selectModelsRatingTable

4.3 Розробка back-end частини

Для розробки серверної частини було обрано мова програмування JavaScript. Node.js проект був створений з усіма залежностями, в першу чергу з Express.js, який працює з підключенням бази даних MySQL.

Для кожної таблиці бази даних було створено окремі пакети

Як вже було зазначено в розділі 4.2, front-end використовує локальні URL, які зазначені в пакеті «routes», який отримує вхідні запити користувача (рисунок 4.12).

```

router.get("/api/predictedTemp/", async (req, res) => {
  let response = await predictedTempRecords(req.params);
  res.send(response);
});

router.get("/api/actualTemp/", async (req, res) => {
  let response = await actualTempRecords(req.params);
  res.send(response);
});

router.put("/api/forecastRating/", bodyParser, async (req, res) => {
  let response = await forecastRating.insertForecastRatingRecord(req.body);
  res.send(response);
});

router.get("/api/modelsRating/", async (req, res) => {
  let response = await models.getModelsRating();
  res.send(response);
});

router.put("/api/modelsRating/", bodyParser, async (req, res) => {
  let response = await models.insertModelsRating(req.body);
  res.send(response);
});

```

Рисунок 4.12 – Пакет «routes»

Коли приходити GET запит з front-end на URL localhost:3000/predictedTemp, щодо отримання записів таблиці, викликається метод predictedTempRecords, який знаходиться в пакеті «db/predictedTemp» (рисунок 4.13).

```
const predictedTempRecords = async (params) => {
  return new Promise((resolve, reject) => {
    pool.query(
      `select ${PredictedTemp.MODEL_ID},
        ${Models.MODEL_NAME},
        ${PredictedTemp.FORECAST_DATE},
        ${PredictedTemp.PREDICTED_TEMP_MAX},
        ${PredictedTemp.PREDICTED_TEMP_MIN},
        ${PredictedTemp.FORECAST_TYPE_ID},
        ${ForecastType.FORECAST_TYPE_DESCRIPTION}
      from
        ${PredictedTemp.TABLE_NAME},
        ${Models.TABLE_NAME},
        ${ForecastType.TABLE_NAME}
      WHERE
        ${Models.MODEL_ID} = ${PredictedTemp.MODEL_ID} AND
        ${ForecastType.FORECAST_TYPE_ID} = ${PredictedTemp.FORECAST_TYPE_ID}` ,
      (err, result) => {
        if (err) {
          return reject(err);
        } else {
          return resolve(result);
        }
      }
    );
  });
};
```

Рисунок 4.13 – Метод predictedTempRecords, який знаходиться в пакеті predictedTemp

Цей метод відправляє SQL запит SELECT, який повертає записи таблиці «predicted_temp» для подальшого виведення на екран.

PUT запит з front-end на URL localhost:3000/forecastRating викликає метод insertForecastRatingRecord, який знаходиться в пакеті «db/forecastRating» (рисунок 4.14).

Цей метод записує об'єкт даних, який надійшов з front-end в таблицю «forecast_rating».

PUT запит з front-end на URL localhost:3000/models викликає метод insertModelsRating, який знаходиться в пакеті «db/models» (рисунок 4. 15).

```

const insertForecastRatingRecord = (params) => {
  return new Promise((resolve, reject) => {
    pool.query(
      `INSERT INTO ${ForecastRating.TABLE_NAME} (
        ${ForecastRating.MODEL_ID},
        ${ForecastRating.FORECAST_DATE},
        ${ForecastRating.TEMP_MAX_DELTA},
        ${ForecastRating.TEMP_MIN_DELTA},
        ${ForecastRating.RATING_MAX},
        ${ForecastRating.RATING_MIN},
        ${ForecastRating.FORECAST_TYPE_ID})
      values (
        ${params.model_id},
        '${params.forecast_date}',
        ${params.temp_max_delta},
        ${params.temp_min_delta},
        ${params.rating_max},
        ${params.rating_min},
        ${params.forecast_type_id}
      )
      on duplicate key update
        ${ForecastRating.TEMP_MAX_DELTA}=${params.temp_max_delta},
        ${ForecastRating.TEMP_MIN_DELTA}=${params.temp_min_delta},
        ${ForecastRating.RATING_MAX}=${params.rating_max},
        ${ForecastRating.RATING_MIN}=${params.rating_min}`
    , (err, result) => {
      if (err) {
        return reject(err);
      } else {
        return resolve(result);
      }
    }
  );
});
};

```

Рисунок 4.14 – Метод insertForecastRatingRecord, який знаходиться в пакеті forecastRating

```

const insertModelsRating = async (params) => {
  return new Promise((resolve, reject) => {
    pool.query(
      `insert into ${ModelsRating.TABLE_NAME}
      (
        model_id,
        forecast_type_id,
        ${ModelsRating.AVG_MAX_TEMP_DELTA},
        ${ModelsRating.AVG_MIN_TEMP_DELTA},
        ${ModelsRating.AVG_MAX_TEMP_RATING},
        ${ModelsRating.AVG_MIN_TEMP_RATING})
      select result.model_id,
      result.forecast_type_id,
      result.avg_max_temp_delta,
      result.avg_min_temp_delta,
      result.avg_max_temp_rating,
      result.avg_min_temp_rating
      from (select
        ${PredictedTemp.MODEL_ID},
        ${PredictedTemp.FORECAST_TYPE_ID},
        ROUND(avg(${ForecastRating.TEMP_MAX_DELTA}), 2) as avg_max_temp_delta,
        ROUND(avg(${ForecastRating.TEMP_MIN_DELTA}), 2) as avg_min_temp_delta,
        ROUND(avg(${ForecastRating.RATING_MAX}), 2) as avg_max_temp_rating,
        ROUND(avg(${ForecastRating.RATING_MIN}), 2) as avg_min_temp_rating
        from ${PredictedTemp.TABLE_NAME}, ${ForecastRating.TABLE_NAME}
      where
        ${PredictedTemp.MODEL_ID} = ${ForecastRating.MODEL_ID} AND
        ${PredictedTemp.FORECAST_DATE} = ${ForecastRating.FORECAST_DATE} AND
        ${PredictedTemp.FORECAST_TYPE_ID} = ${ForecastRating.FORECAST_TYPE_ID} group by model_id, forecast_type_id) as result
      on duplicate key update
        avg_max_temp_delta = result.avg_max_temp_delta,
        avg_min_temp_delta = result.avg_min_temp_delta,
        avg_max_temp_rating = result.avg_max_temp_rating,
        avg_min_temp_rating = result.avg_min_temp_rating`
    , (err, result) => {
      if (err) {
        return reject(err);
      } else {
        return resolve(result);
      }
    }
  );
});

```

Рисунок 4.15 – Метод insertModelsRating, який знаходиться в пакеті models

Цей метод відправляє SQL запит INSERT, який розраховує середнє арифметичне похибки прогнозу температури та оцінки по кожній моделі та типу прогнозу і записує в таблицю «models_rating».

GET запит з front-end на URL localhost:3000/models викликає метод getModelsRating, який знаходиться в пакеті «db/models» (рисунок 4.16).

```
const getModelsRating = async (params) => {
  return new Promise((resolve, reject) => {
    pool.query(
      `select
        ${ModelsRating.MODEL_ID},
        ${ModelsRating.FORECAST_TYPE_ID},
        ${Models.MODEL_NAME},
        ${ForecastType.FORECAST_TYPE_DESCRIPTION},
        ${ModelsRating.AVG_MAX_TEMP_DELTA},
        ${ModelsRating.AVG_MIN_TEMP_DELTA},
        ${ModelsRating.AVG_MAX_TEMP_RATING},
        ${ModelsRating.AVG_MIN_TEMP_RATING}
      from
        ${ModelsRating.TABLE_NAME},
        ${Models.TABLE_NAME},
        ${ForecastType.TABLE_NAME}
      where
        ${ModelsRating.MODEL_ID} = ${Models.MODEL_ID} AND
        ${ModelsRating.FORECAST_TYPE_ID} = ${ForecastType.FORECAST_TYPE_ID}`,
      (err, result) => {
        if (err) {
          return reject(err);
        } else {
          return resolve(result);
        }
      }
    );
  });
};
```

Рисунок 4.16 – Метод getModelsRating, який знаходиться в пакеті models

Цей метод відправляє SQL запит SELECT, який повертає записи таблиці «models_rating» для подальшого виведення на екран.

4.4 Інструкція користувача

Так як система була реалізована у виді веб-додатку, після запуску відкриється інтерфейс користувача на URL <http://localhost:3000>.

Головна сторінка додатку включає в собі меню та компонент виводу (рисунок 4.17). На вкладці «ALL TABLES» в компоненті виводу доступний перегляд таблиць системи.

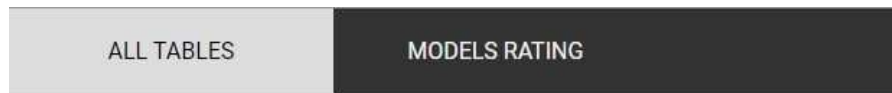


Рисунок 4.16 – Головне меню з обраною опцією «ALL TABLES»

Приклад виводу таблиць можна побачити на рисунку 4.18 та 4.19.

| Predicted Temperature | | | | |
|-----------------------|---------------|------------------------------|------------------------------|---------------|
| Model Name | Forecast Date | Maximum Forecast Temperature | Minimum Forecast Temperature | Forecast type |
| Accuweather | 2020-11-16 | -2 | -4 | 1 day |
| Accuweather | 2020-11-17 | -2 | -4 | 1 day |
| Accuweather | 2020-11-18 | -1 | -5 | 3 days |
| Accuweather | 2020-11-19 | -1 | -5 | 3 days |
| Accuweather | 2020-11-20 | 0 | -1 | 5+ days |

Рисунок 4.18 – Приклад виводу таблиці «predicted_temp» на інтерфейс користувача

| Forecast Rating | | | | | | |
|-----------------|---------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|---------------|
| Model Name | Forecast Date | Maximum Temperature Difference | Minimum Temperature Difference | Minimum Temperature Difference | Minimum Temperature Difference | Forecast type |
| Accuweather | 2020-11-16 | 1 | 1 | 5 | 5 | 1 day |
| Accuweather | 2020-11-17 | 1 | 1 | 5 | 5 | 1 day |
| Accuweather | 2020-11-18 | 1 | 1 | 5 | 5 | 3 days |
| Accuweather | 2020-11-19 | 2 | 1 | 4 | 5 | 3 days |
| Accuweather | 2020-11-20 | 1 | 3 | 5 | 2 | 5+ days |

Рисунок 4.19 – Приклад виводу таблиці «forecast_rating» на інтерфейс користувача

У майбутньому планується розробити функціонал для роботи с таблицями бази даних, тобто додавання, редагування та видалення записів.

На вкладці «MODELS RATING» (рисунок 4.20) формується компонент виводу середньої похибки прогнозу температури та середньої оцінки по кожній моделі та типу прогнозу.

Рисунок 4.20 – Головне меню з обраною опцією «MODELS RATING»

Приклади сформованих компонентів можна побачити на рисунках 4.21...4.26.

Accuweather

| Forecast type | Average maximum temperature difference | Average minimum temperature difference | Average maximum temperature rating | Average minimum temperature rating |
|---------------|--|--|------------------------------------|------------------------------------|
| 1 day | 1.56 | 0.78 | 4.22 | 5 |
| 3 days | 1.33 | 0.67 | 4.67 | 5 |
| 5+ days | 1 | 1.33 | 4.67 | 4 |

Рисунок 4.21 – Компонент моделі Accuweather

Wunderground

| Forecast type | Average maximum temperature difference | Average minimum temperature difference | Average maximum temperature rating | Average minimum temperature rating |
|---------------|--|--|------------------------------------|------------------------------------|
| 1 day | 0.78 | 1.56 | 4.89 | 4 |
| 3 days | 1.67 | 1.67 | 4 | 4 |
| 5+ days | 1.67 | 1.67 | 4 | 4 |

Рисунок 4.22 – Компонент моделі Wunderground

Meteo.ua

| Forecast type | Average maximum temperature difference | Average minimum temperature difference | Average maximum temperature rating | Average minimum temperature rating |
|---------------|--|--|------------------------------------|------------------------------------|
| 1 day | 0.89 | 1.11 | 4.67 | 4.44 |
| 3 days | 1.67 | 0.33 | 4 | 5 |
| 5+ days | 1.67 | 0.33 | 4 | 5 |

Рисунок 4.23 – Компонент моделі Meteo.ua

OpenWeatherMap

| Forecast type | Average maximum temperature difference | Average minimum temperature difference | Average maximum temperature rating | Average minimum temperature rating |
|---------------|--|--|------------------------------------|------------------------------------|
| 1 day | 1 | 0.67 | 4.67 | 5 |
| 3 days | 1 | 0.67 | 4.67 | 5 |
| 5+ days | 0.67 | 0.67 | 5 | 5 |

Рисунок 4.24 – Компонент моделі OpenWeatherMap

yr.no

| Forecast type | Average maximum temperature difference | Average minimum temperature difference | Average maximum temperature rating | Average minimum temperature rating |
|---------------|--|--|------------------------------------|------------------------------------|
| 1 day | 1 | 0.78 | 4.78 | 4.89 |
| 3 days | 0.67 | 1.33 | 5 | 4.67 |
| 5+ days | 1 | 1.67 | 4.67 | 4.33 |

Рисунок 4.25 – Компонент моделі yr.no

Yandex.Pogoda

| Forecast type | Average maximum temperature difference | Average minimum temperature difference | Average maximum temperature rating | Average minimum temperature rating |
|---------------|--|--|------------------------------------|------------------------------------|
| 1 day | 0.78 | 1.56 | 4.89 | 4.22 |
| 3 days | 1 | 2.33 | 4.67 | 3 |
| 5+ days | 1.33 | 3 | 4.67 | 2.67 |

Рисунок 4.26 – Компонент моделі Yandex.Pogoda

Таким чином, користувач може проаналізувати результати дослідження і побачити, яка з моделей являється найбільш точною в тому чи іншому типі прогнозу.

Результати дослідження в виді діаграм можна побачити на наступних рисунках 4.27...4.29.

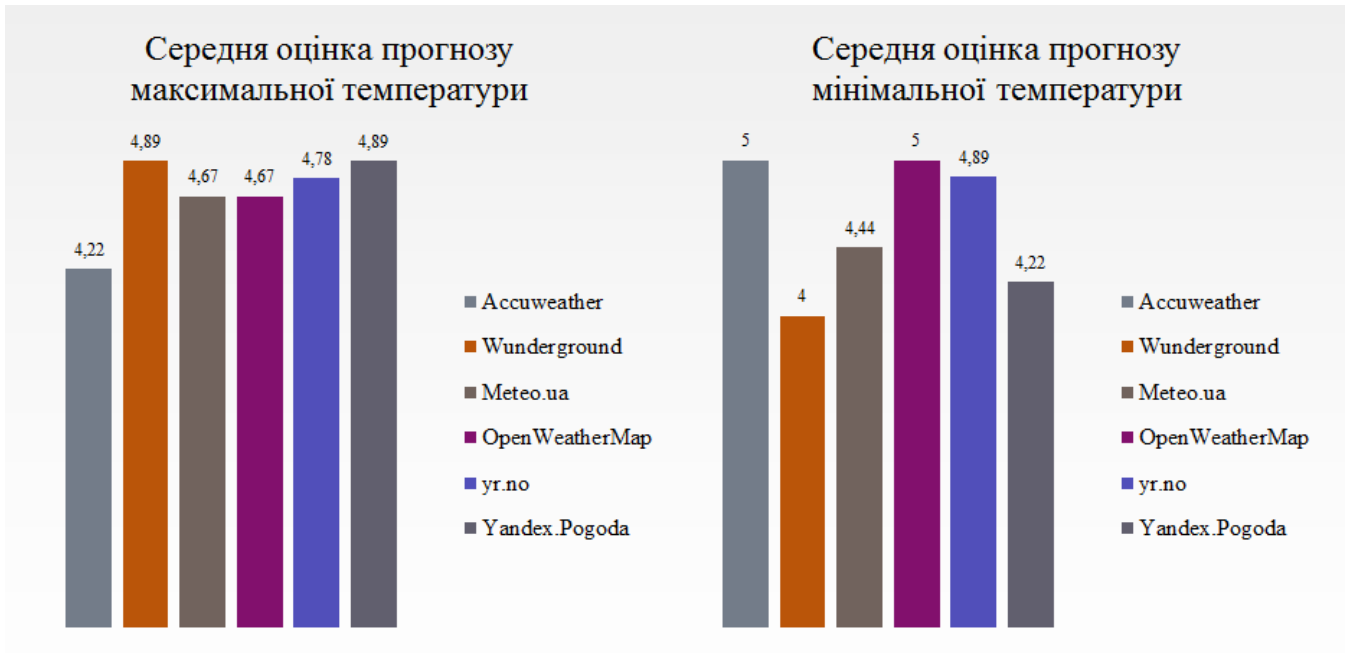


Рисунок 4.27 – Результати дослідження прогнозів максимальної і мінімальної температури на 1 день

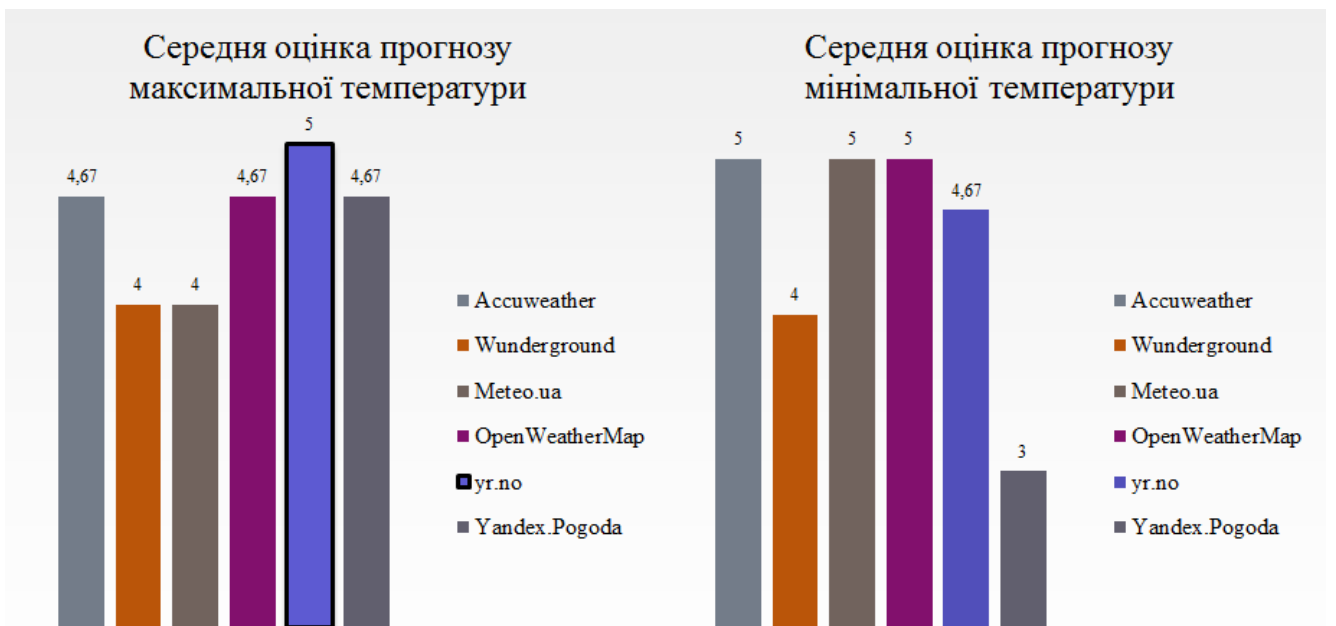


Рисунок 4.28 – Результати дослідження прогнозів максимальної і мінімальної температури на 3 дня

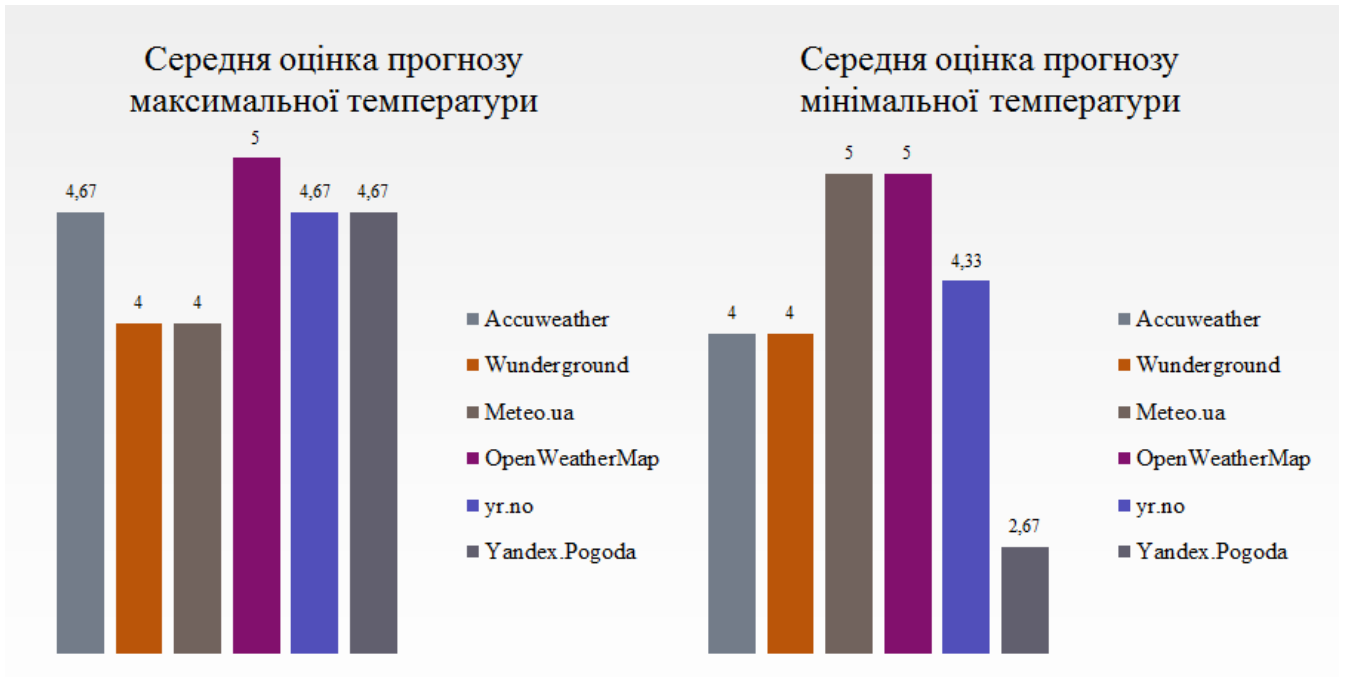


Рисунок 4.29 – Результати дослідження прогнозів максимальної і мінімальної температури на 5 чи більше днів

ВИСНОВКИ

В результаті виконання атестаційної роботи була досліджена предметна область короткострокового та довгострокового прогнозу погоди, висунута проблема та визначений об'єкт, мета та задача дослідження для її виправлення.

Було проведено дослідження різноманітних моделей прогнозу погоди з наведенням їх прикладів та порівнянням.

Були сформовані вимоги до системи та вибрані інструменти для розробки, а саме мова програмування JavaScript, React, Node.js.

Спираючись на висунуті вимоги, була спроектована універсальна система аналізу прогнозів погоди та реалізований її прототип, використовуючи зазначені інструменти розробки. Система складається з узагальнених сутностей, що були сформовані спираючись на дослідження предметної області.

Усі поставлені задачі виконані в повному обсязі.

Розробка атестаційної роботи розширила знання мови програмування JavaScript, дала додатковий досвід роботи з React, закріпила навички роботи з базою даних MySQL та Node.js.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Wunderground Kharkiv, Ukraine Weather Condition [Електронний ресурс] / Wunderground – Режим доступу. <https://www.wunderground.com> – Загол. з екрану (дата звернення: 26.09.2020).
2. Wunderground WunderMap [Електронний ресурс] / Wunderground – Режим доступу. <https://www.wunderground.com> – Загол. з екрану (дата звернення: 26.09.2020).
3. Wikipedia Weather Underground (weather service) [Електронний ресурс] / Wikipedia – Режим доступу. <https://en.wikipedia.org/wiki/> – Загол. з екрану (дата звернення: 26.09.2020).
4. Wikipedia Яндекс [Електронний ресурс] / Wikipedia – Режим доступу. <https://ru.wikipedia.org/wiki/> – Загол. з екрану (дата звернення: 26.09.2020).
5. Yandex Pogoda FAQ [Електронний ресурс] / Yandex – Режим доступу. <https://yandex.ua/> – Загол. з екрану (дата звернення: 26.09.2020).
6. Yandex Погода, Харьков [Електронний ресурс] / Yandex – Режим доступу. <https://yandex.ua/> – Загол. з екрану (дата звернення: 26.09.2020).
7. Wikipedia AccuWeather [Електронний ресурс] / Wikipedia – Режим доступу. <https://en.wikipedia.org/wiki/> – Загол. з екрану (дата звернення: 26.09.2020).
8. AccuWeather Харьков, Харьковская, Украина [Електронний ресурс] / AccuWeather – Режим доступу. <https://www.accuweather.com/> – Загол. з екрану (дата звернення: 26.09.2020).
9. Wikipedia OpenWeatherMap [Електронний ресурс] / Wikipedia – Режим доступу. <https://ru.wikipedia.org/wiki/> – Загол. з екрану (дата звернення: 26.09.2020).
10. OpenWeatherMap Weather forecast [Електронний ресурс] / OpenWeatherMap – Режим доступу. <https://openweathermap.org/> – Загол. з екрану (дата звернення: 26.09.2020).
11. Wikipedia Yr.no [Електронний ресурс] / Wikipedia – Режим доступу.

<https://en.wikipedia.org/wiki/> – Загол. з екрану (дата звернення: 26.09.2020).

12. Yr – Weather forecast for Kharkiv [Електронний ресурс] / Yr.no – Режим доступу. <https://www.yr.no/> – Загол. з екрану (дата звернення: 26.09.2020).

13. ForecastAdvisor [Електронний ресурс] / ForecastAdvisor – Режим доступу. <https://www.forecastadvisor.com/> – Загол. з екрану (дата звернення: 26.09.2020).

14. Weather.Slimyhorror [Електронний ресурс] / Weather.Slimyhorror – Режим доступу. <https://weather.slimyhorror.com/> – Загол. з екрану (дата звернення: 26.09.2020).

15. Wikipedia Прогноз погоди [Електронний ресурс] / Wikipedia – Режим доступу. <https://en.wikipedia.org/wiki/> – Загол. з екрану (дата звернення: 18.11.2020).

16. Wikipedia Прогноз погоди [Електронний ресурс] / Wikipedia – Режим доступу. <https://en.wikipedia.org/wiki/> – Загол. з екрану (дата звернення: 18.11.2020).

17. Wikipedia JavaScript [Електронний ресурс] / Wikipedia – Режим доступу. <https://en.wikipedia.org/wiki/> – Загол. з екрану (дата звернення: 06.12.2020).

18. Wikipedia Node.js [Електронний ресурс] / Wikipedia – Режим доступу. <https://en.wikipedia.org/wiki/> – Загол. з екрану (дата звернення: 06.12.2020).

19. Express JS [Електронний ресурс] / Express JS – Режим доступу. <https://expressjs.com/> – Загол. з екрану (дата звернення: 06.12.2020).

20. React [Електронний ресурс] / React – Режим доступу. <https://reactjs.org/> – Загол. з екрану (дата звернення: 06.12.2020).

21. Wikipedia React [Електронний ресурс] / Wikipedia – Режим доступу. <https://en.wikipedia.org/wiki/> – Загол. з екрану (дата звернення: 06.12.2020).

22. Wikipedia Bootstrap (front–end framework) [Електронний ресурс] / Wikipedia – Режим доступу. <https://en.wikipedia.org/wiki/> – Загол. з екрану (дата звернення: 06.12.2020).

23. React Bootstrap [Електронний ресурс] / React Bootstrap – Режим доступу. <https://react-bootstrap.github.io/> – Загол. з екрану (дата звернення: 06.12.2020).

24. Wikipedia MySQL [Електронний ресурс] / Wikipedia – Режим доступу.
<https://en.wikipedia.org/wiki/> – Загол. з екрану (дата звернення: 07.12.2020).

