

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ**

**Кафедра комп'ютеризованих систем управління**

ДОПУСТИТИ ДО ЗАХИСТУ  
Завідувач кафедри

\_\_\_\_\_ Олександр ЛИТВИНЕНКО

«\_\_\_» \_\_\_\_\_ 2023 р.

**КВАЛІФІКАЦІЙНА РОБОТА**  
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

**ЗДОБУВАЧА ВИЩОЇ ОСВІТИ  
СТУПЕНЯ «МАГІСТР»**

Тема: \_\_\_\_\_ Система інтеграції даних про метеоумови для малих аеродромів \_\_\_\_\_

Виконавець: \_\_\_\_\_ Дмитро ПРОКОПЧУК \_\_\_\_\_

Керівник: \_\_\_\_\_ Віталій ЗУБОК \_\_\_\_\_

Нормоконтролер: \_\_\_\_\_ Євгеній ТУПОТА \_\_\_\_\_

**Київ 2023**



## 6. Календарний план-графік

№ п/п	Етапи виконання дипломного проєкту	Термін виконання етапів	Примітка
1	Провести аналіз літератури за темою дипломного проєкту та аналіз існуючих систем	02.10.2023- 12.10.2023	
2	Аналіз існуючих технологій	13.10.2023- 28.10.2023	
3	Аналіз компонентів систем інтеграції даних про метеоумови	29.10.2023- 05.11.2023	
4	Програмна системи інтеграції	06.11.2023- 13.11.2023	
5	Програмна реалізація алгоритму роботи серверного додатку	14.11.2023- 25.11.2023	
6	Дослідження характеристик отриманої системи	26.11.2023- 01.12.2023	
7	Написати пояснювальну записку	02.12.2023- 13.12.2023	
8	Підготувати презентацію	14.12.2023- 17.12.2023	
9	Оформити супроводжувальну документацію	18.12.2023 19.12.2023	

## 7. Дата видачі завдання « 02 » жовтня 2023 р.

Керівник дипломного проєкту \_\_\_\_\_ Віталій ЗУБОК  
(підпис)

Завдання прийняв до виконання \_\_\_\_\_ Дмитро ПРОКОПЧУК  
(підпис студента)

## РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи магістра на тему «Система інтеграції даних про метеоумови для малих аеродромів»: 81 с., 18 рис., 24 літературних джерела, 2 додатки.

СИСТЕМА ІНТЕГРАЦІЇ МЕТЕОДАНИХ, МАЛИЙ АЕРОДРОМ,  
МЕТЕОДАНИ, БАГАТОДЖЕРЕЛЬНА ІНТЕГРАЦІЯ.

Мета дослідження – розробка та впровадження системи інтеграції даних про метеоумови на малих аеродромах з метою забезпечення надійного збору, аналізу та використання метеорологічної інформації.

Об'єкт дослідження – процес інтеграції даних про метеоумови для малих аеродромів.

Предмет дослідження – система інтеграції даних про метеоумови для малих аеродромів.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ .....	6
ВСТУП .....	7
РОЗДІЛ 1 ПРИНЦИПИ ІНТЕГРАЦІЇ МЕТЕОДАНИХ НА АЕРОДРОМАХ.....	13
1.1. Огляд наукових досліджень щодо ролі метеоданих у авіаційній безпеці та методах інтеграції метеоданих .....	13
1.2. Огляд сучасних методів опитування і синхронізації даних у метеорології та авіації .....	18
1.3. Аналіз існуючих стандартів і систем інтеграції метеоданих.....	24
1.4. Висновки до розділу.....	29
РОЗДІЛ 2 ПРОЄКТУВАННЯ СИСТЕМИ ІНТЕГРАЦІЇ МЕТЕОДАНИХ.....	30
2.1. Формування дерева функцій системи інтеграції метеоданих на малих аеродромах.....	30
2.2. Архітектура системи та визначення вимог до неї.....	36
2.3. Моделювання процесів опитування та синхронізації метеоданих з різних джерел. ....	40
2.4. Вибір інструментів для аналізу та обробки метеорологічних даних .....	50
2.5. Висновки до розділу.....	53
РОЗДІЛ 3 ПРОГРАМНА РОЗРОБКА СИСТЕМИ ІНТЕГРАЦІЇ ДАНИХ ПРО МЕТЕОУМОВИ НА МАЛИХ АЕРОДРОМАХ .....	55
3.1. Опис роботи основних розроблених модулів системи.....	55
3.2. Розгортання системи на зовнішніх серверах.....	64
3.3. Приклад роботи розробленої системи.....	65
3.4. Висновки до розділу.....	74
ВИСНОВКИ .....	76
СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ.....	79
ДОДАТОК А.....	82
ДОДАТОК Б .....	84

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

*AIXM – Aeronautical Information Exchange Model*

*AMOSS – Aviation Meteorological Observation and Reporting System*

*API – Application Programming Interface*

*ASOS – Automated Surface Observing System*

*ATIS – Automatic Terminal Information Service*

*AWOS – Automated Weather Observing System*

*CSV – Comma-Separated Values*

*GPS – Global Positioning System*

*GUI – Graphical User Interface*

*IDE – Integrated Development Environment*

*IWXXM (ICAO Meteorological Information Exchange Model)*

*JSON – JavaScript Object Notation*

*METAR – Meteorological Aerodrome Report*

*SQL – Structured Query Language*

*SSH – Secure Shell*

*SSL – Secure Sockets Layer*

*SSL/TLS – Secure Sockets Layer/Transport Layer Security*

*TAF – Terminal Aerodrome Forecast*

*UTC – Coordinated Universal Time*

*WSDL – Web Services Description Language*

*XML – Extensible Markup Language*

*XML-RPC – XML Remote Procedure Call*

## ВСТУП

**Актуальність теми.** Кожного року авіаційні компанії розширюють свої маршрути та збільшують обсяги польотів, дотримання стандартів безпеки та отримання точної інформації про метеоумови стають вирішальними факторами для успішної авіаційної діяльності.

Метеодані – це набір інформації, що описує стан атмосферних умов та погодні явища в певному регіоні або місцевості в певний момент часу або протягом певного періоду. Ці дані дуже різноманітні та можуть включати в себе наступну інформацію: температура повітря, вологість, атмосферний тиск, швидкість та напрям вітру, опади, видимість тощо.

Метеодані збираються за допомогою спеціальних метеорологічних приладів, таких як барометри, термометри, анемометри, дозиметри, а також супутникових і радарних систем. Метеорологічне спостереження – це інструментальне вимірювання і візуальні оцінки метеорологічних величин та атмосферних явищ [1, 4]. Вони використовуються для прогнозу погоди, моніторингу клімату, безпеки польотів, розробки стратегій управління екологічними ресурсами, сільського господарства, наукових досліджень і багатьох інших цілей. Метеодані є важливим ресурсом для прийняття рішень та забезпечення безпеки та комфорту життя людей.

На аеродромах використовуються різні типи метеоданих для забезпечення безпеки авіаційних польотів і ефективності аеропортової діяльності. Основні категорії метеоданих, які використовуються на аеродромах, включають [5, 6]:

– метеорологічні дані: включає в себе інформацію про температуру повітря, вологість, атмосферний тиск, швидкість і напрям вітру, видимість і опади. Ці дані важливі для безпеки приземлення та злету літаків, а також для розрахунків під час маневрів на аеродромі;

– дані про стан підстилки: інформація про стан земної поверхні на аеродромі, таку як стан покриття (асфальт, ґрунт), наявність ожеледиці або снігу, якість

асфальту, наявність станцій обігріву та інші параметри. Це важливо для безпеки приземлення і гальмування літаків;

– інформація про зону навколо аеродрому, що включає в себе дані про розташовані поблизу природні перешкоди, такі як гори, дерева, будівлі, які можуть вплинути на безпеку польотів. Вона використовується для розрахунків висоти мінімального безпечного польоту і планування маршрутів з врахуванням метеоумов;

– інформація про освітлення та навігаційні прилади: Це дані про роботу світлових та навігаційних систем на аеродромі та вплив на їх роботу метеоумов, такі як приземні та прильотні світлофори, радіолокаційні системи та інші засоби, що вказують літакам правильний шлях при приземленні і руху на аеродромі.

Проблема метеоданих на малих аеродромах полягає в низці чинників, які можуть ускладнити збір, обробку та використання метеорологічної інформації на аеродромах [7], що може призвести до катастроф [8]. Ось деякі з цих проблем:

– обмеженість обладнання: малі аеродроми можуть бути обмежені фінансовими ресурсами, тому їхнє обладнання для збору метеоданих може бути недостатнім і/або ненадійним;

– доступність даних: не завжди є можливість отримати дані з найсучасніших метеорологічних джерел, особливо на віддалених аеродромах. Це може призвести до нестачі актуальної інформації;

– специфіка малих аеродромів: малі аеродроми можуть мати свої особливості та умови, які потребують індивідуального підходу до збору та використання метеоданих;

– потенційні загрози: метеорологічні умови на малих аеродромах можуть бути більш непередбачуваними, існує ризик погіршення видимості, зміни напрямку та швидкості вітру, а також інші метеорологічні явища, які можуть вплинути на безпеку польоту;

– вимоги до точності: метеорологічні дані на аеродромах повинні бути дуже точними, особливо для прийняття рішень щодо польотів і посадки літаків. Навіть невеликі похибки можуть бути критичними для безпеки;



– інтеграція даних: об'єднання даних з різних джерел та їхня синхронізація можуть бути складним завданням на аеродромах, де обмежені ресурси;

– оперативність: важливо мати можливість отримувати метеодані в режимі реального часу для оперативного реагування на зміни в погодних умовах.

Розв'язання цих проблем вимагає використання сучасних технологій, розвитку інфраструктури метеорологічних систем та налагодження системи інтеграції та обробки метеоданих для малих аеродромів, що покращує безпеку та ефективність польотів.

Авіаційна безпека – це основний пріоритет при обробці метеорологічних даних. Метеоумови, а особливо в окремих регіонах [9, 10, 11, 12] можуть призвести до аварій, погіршити видимість для пілотів, обмежити операційні можливості літаків і навіть призвести до скасування польотів [13, 14]. Саме тому вчені, інженери та авіаційні спеціалісти постійно працюють над розробкою нових технологій і систем, які допоможуть покращити передбачення погоди і забезпечити пілотам та диспетчерам потрібну інформацію для прийняття безпечних рішень, розробити спеціальні підходи для покращення обробки метеоданих [15, 16, 17].

Малі аеродроми відіграють важливу роль у регіональних авіаційних операціях, а також у забезпеченні доступу до віддалених регіонів. Ці аеродроми можуть відрізнятись обмеженими ресурсами і обладнанням, порівняно з великими аеропортами. Тому необхідність у вдосконаленні систем збору, інтеграції та аналізу метеоданих на малих аеродромах є дуже актуальною та важливою. Недостатність точної та надійної інформації про погоду на таких аеродромах може призвести до небезпечних ситуацій під час злету, посадки та руху літаків на землі [18, 19]. Отже, розробка та впровадження систем інтеграції метеоданих стають першочерговим завданням для підвищення безпеки і ефективності авіаційних операцій на малих аеродромах.

Наведемо декілька прикладів аеродромів світу, де особливо важливо враховувати при отриманні метеоданих особливості розміщення аеродрому:

– *Caernarfon*, Велика Британія [20] – розташована в гірському районі, де погодні умови можуть змінюватися дуже швидко, і виникають проблеми з туманами, низькою хмарністю та сильними вітрами, що ускладнює польоти;

– *Gimpo International Airport*, Південна Корея [21] – проблема полягає у сильних бурях, туманах та забрудненні повітря внаслідок промислової діяльності.

– *Kuala Lumpur Subang Abdul Razak Airport*, Малайзія [22] – аеродром розташований у вологому тропічному кліматі, де часто відбуваються дощі та грози, що може призвести до перерв у польотах та ускладнити безпеку.

Основна проблема, яка виникає у контексті метеоданих на малих аеродромах, полягає в обмеженості ресурсів та обладнання на таких аеродромах, а також в потребі в надійному та ефективному зборі, обробці та використанні інформації про погоду [23]. Основна проблема – це відсутність сучасних систем інтеграції, які б забезпечували автоматизований збір та аналіз метеоданих і надавали цю інформацію диспетчерам, пілотам та іншим зацікавленим сторонам.

Мета цього дослідження полягає у розробці та впровадженні системи інтеграції даних про метеоумови на малих аеродромах з метою забезпечення надійного збору, аналізу та використання метеорологічної інформації. Ця система повинна бути спроектована таким чином, щоб враховувати специфічні потреби малих аеродромів і забезпечувати інформаційною підтримкою авіаційних операцій на них. Основною метою є підвищення безпеки польотів, покращення їх ефективності та зменшення ризиків, пов'язаних зі змінами погодних умов.

Розробка системи інтеграції метеоданих на малих аеродромах включає в себе ряд викликів та труднощів. Основні з них включають:

1. Обмежені ресурси: малі аеродроми можуть мати обмежені фінансові ресурси та обладнання для збору та обробки метеоданих. Розробка системи повинна бути ефективною та бюджетно-орієнтованою.

2. Варіабельність погодних умов: метеопередбачення на малих аеродромах ускладнюється варіабельністю погодних умов в регіональних масштабах. Система повинна бути готова обробляти інформацію про непередбачуві зміни в погоді.

3. Безпека даних: забезпечення конфіденційності і безпеки метеоданих є надзвичайно важливою задачею, особливо у вимогливій авіаційній галузі.

4. Інтеграція з існуючими системами: система повинна бути сумісною з існуючими авіаційними системами та стандартами для забезпечення взаємодії з іншими компонентами авіаційного обладнання та програмними засобами.

Урахування цих викликів та труднощів є ключовим етапом у процесі розробки та впровадження системи інтеграції метеоданих на малих аеродромах. Розв'язання цих завдань сприятиме покращенню безпеки та ефективності авіаційних операцій, що є метою цього дослідження.

Наукова новизна цього дослідження полягає в розробці та впровадженні інноваційної системи інтеграції метеоданих на малих аеродромах. Ця система буде спеціально адаптована до потреб таких аеродромів і враховуватиме їхні особливості та обмеження, надасть можливість забезпечити диспетчерів, пілотів та інших учасників авіаційних операцій надійною та актуальною інформацією про погодні умови, що є важливим кроком у покращенні безпеки та ефективності авіаційних операцій на малих аеродромах.

Деякі аспекти наукової новизни включають:

– оптимізація опитування даних: розробка алгоритмів для опитування зовнішніх джерел метеоданих з метою мінімізації часу затримки та підвищення надійності надходження даних;

– синхронізація та інтеграція даних: розробка механізмів для автоматичної синхронізації та інтеграції даних з різних джерел, що дозволяє підтримувати їх узгодженість та надійність;

– аналіз та обробка даних: розробка алгоритмів аналізу метеорологічних даних для виявлення та прогнозу негативних погодних явищ, які можуть впливати на авіаційний рух;

– автоматизація прийняття рішень: розробка інтелектуальних систем для автоматичного прийняття рішень на основі аналізу метеоданих з метою підвищення безпеки авіаційних операцій.

Завдання, які потрібно вирішити в рамках цієї роботи, включають:

- розробити архітектуру системи, включаючи визначення специфікацій для інтеграції зовнішніх джерел метеоданих;
- реалізувати механізми опитування зовнішніх джерел та синхронізації отриманих даних;
- розробити інструменти для аналізу та обробки метеорологічних даних;
- впровадити автоматичну систему прийняття рішень на основі аналізу метеоданих.

З огляду на вище сказане об'єктом дослідження є процес інтеграції даних про метеоумови на малих аеродромах. Предметом дослідження є система інтеграції даних про метеоумови для малих аеродромів.

Розробка системи інтеграції метеоданих на малих аеродромах включає в себе ряд викликів та труднощів:

- обмежені ресурси: малі аеродроми можуть мати обмежені фінансові ресурси та обладнання для збору та обробки метеоданих [23];
- варіабельність погодних умов: метеопередбачення на малих аеродромах ускладнюється варіабельністю погодних умов в регіональних масштабах [19];
- забезпечення безпеки даних, яке є надзвичайно важливою задачею, особливо у вимогливій авіаційній галузі і в умовах обмежених ресурсів окремого аеродрому;
- інтеграція з існуючими системами: система повинна бути сумісною з існуючими авіаційними системами та стандартами для забезпечення взаємодії з іншими компонентами авіаційного обладнання та програмними засобами [17].

Урахування цих викликів та труднощів є ключовим етапом у процесі розробки та впровадження системи інтеграції метеоданих на малих аеродромах. Розв'язання цих завдань сприятиме покращенню безпеки та ефективності авіаційних операцій, що є метою цього дослідження.

## РОЗДІЛ 1

### ПРИНЦИПИ ІНТЕГРАЦІЇ МЕТЕОДАНИХ НА АЕРОДРОМАХ

#### 1.1. Огляд наукових досліджень щодо ролі метеоданих у авіаційній безпеці та методах інтеграції метеоданих

Авіаційна безпека завжди була найвищим пріоритетом в цивільній та військовій авіації.

Погодні умови завжди були ключовим фактором, що впливає на безпеку авіаційних польотів. Метеорологічна інформація, отримана з різних джерел і датчиків, грає вирішальну роль у забезпеченні безпеки та успішності авіаційних місій. Вона надає важливі дані про стан атмосфери, включаючи температуру повітря, вологість, атмосферний тиск, швидкість і напрям вітру, кількість опадів та інші параметри, які можуть суттєво вплинути на безпеку та ефективність авіаційних операцій.

Збір, аналіз та інтеграція метеорологічних даних стали ключовими завданнями для авіаційної галузі. Оператори, пілоти та диспетчери повинні мати доступ до актуальної та достовірної інформації про погоду для прийняття рішень, які стосуються роботи літаків, маршрутів польотів і безпеки пасажирів.

Однак важливо враховувати, що метеодані можуть бути отримані з різних джерел, таких як метеорологічні станції, супутники, датчики на аеродромах та інші. Ці дані можуть бути в різних форматах і містити різні параметри. Тому інтеграція метеорологічних даних стає важливим завданням для забезпечення їхньої спільної обробки та аналізу.

У цьому контексті система інтеграції метеоданих на малих аеродромах виконує критичну роль у забезпеченні доступу до потрібної метеорологічної інформації, її обробці та поданні користувачам. Така система має враховувати різні джерела даних, забезпечувати їхню синхронізацію та надавати зручний інтерфейс для користувачів для роботи з метеорологічною інформацією.

Негода, грози, туман, сильний вітер і інші погодні фактори можуть створити небезпечні ситуації та призвести до аварій. Метеодані грають критичну роль у попередженні цих небезпечних ситуацій та узгодженні безпеки польотів [16, 17, 19, 23].

#### 1.1.1. Роль метеоданих у підвищенні авіаційної безпеки

Погодні умови завжди були ключовим фактором, що впливає на безпеку авіаційних польотів. Метеорологічна інформація, отримана з різних джерел і датчиків, грає вирішальну роль у забезпеченні безпеки та успішності авіаційних місій. Розглянемо роль метеоданих у підвищенні авіаційної безпеки більш детально:

1. Визначення безпечної маршруту і висоти польоту: метеодані надають інформацію про атмосферний тиск, температуру, вологість, швидкість вітру і характеристики хмар, що допомагає пілотам визначити оптимальні маршрути і висоту польоту. Вірне розрахування цих параметрів дозволяє уникнути небезпеки, пов'язаної з турбулентністю, туманом, грозами та іншими метеорологічними умовами.

2. Попередження про небезпеки: метеодані також грають ключову роль у попередженні небезпечних метеорологічних умов, таких як грози, сильні пориви вітру, град та інші. Попередження про небезпеки надаються диспетчерським службам і пілотам, щоб вони могли прийняти необхідні заходи для забезпечення безпеки польоту.

3. Запобігання льодоутворенню: літаки можуть потрапити в льодові умови в атмосфері. Метеодані допомагають ідентифікувати місця льодоутворення, що дає можливість пілотам уникнути цих зон або активувати системи антильодової обробки.

4. Мінімізація впливу турбулентності: турбулентність може бути небезпечною для літаків, особливо для маленьких авіаційних апаратів. Метеодані про турбулентність допомагають пілотам планувати більш безпечні маршрути або зменшувати висоту для мінімізації впливу турбулентності.

5. Підтримка навігації: метеодані використовуються для покращення навігації літаків, визначення місцезнаходження і корекції маршруту у випадку зміни погодних умов.

6. Допомога при аварійних ситуаціях: у випадку аварії чи екстреної ситуації, метеорологічні дані можуть вказувати місце падіння літака або інші важливі деталі, які допомагають рятувальним службам швидше реагувати.

7. Сприяння плануванню та диспетчеризації: диспетчери використовують метеодані для планування руху літаків, розміщення їх на маршрутах та прийняття рішень щодо держав з найкращими метеорологічними умовами для приземлення.

8. Оцінка безпеки перед польотом: пілоти використовують метеодані для оцінки загальної безпеки польоту та прийняття рішення щодо вильоту.

9. Зменшення витрат пального: метеорологічні дані допомагають економити пальне шляхом вибору економічних маршрутів та оптимальної висоти польоту.

10. Збільшення точності приземлення: для авіаційних апаратів, особливо пасажирських літаків, метеорологічні дані є важливими для точного приземлення і запобігання аваріям при поганих погодних умовах.

Загальна роль метеоданих в авіації полягає у тому, що вони забезпечують найважливішу інформацію, яка допомагає зберегти життя, зменшити ризики аварій та підвищити ефективність польотів. Без належної метеорологічної інформації авіаційна безпека була б серйозно під загрозою.

#### 1.1.2. Огляд існуючих методів інтеграції метеоданих

Інтеграція метеоданих є важливим елементом для забезпечення точності та доступності інформації про погоду на аеродромах. Існують різні методи та підходи до інтеграції метеоданих.

Розглянемо основні методи та підходи до інтеграції метеоданих:

1. Централізована система моніторингу: у цій системі всі метеорологічні дані збираються та обробляються на центральному сервері чи в метеорологічному центрі.

Переваги:

- однорідність та зіставимість даних, оскільки вони проходять через одну обробку;

- централізоване керування та моніторинг даних;

- забезпечення доступу до актуальних метеоданих.

Недоліки:

- ризик однієї точки невдачі: якщо центральний сервер вийде з ладу, може виникнути велика проблема;

- збільшені витрати на обладнання та утримання центрального центру обробки.

2. Розподілена система моніторингу: у цій системі різні джерела метеоданих розподіляються на різних аеродромах чи в місцевих пунктах спостереження.

Переваги:

- зменшення ризику великої невдачі через розподіленість джерел даних;

- знижені витрати на трансмісію та обробку даних;

- більша надійність системи в цілому.

Недоліки:

- можливість виникнення різниці у якості та актуальності даних між різними джерелами.

- складність у співпраці між різними локальними системами.

3. Гібридні системи: поєднують в собі як централізовані, так і розподілені методи збору та обробки даних. Наприклад, централізований сервер може бути використаний для обробки даних з різних регіональних аеродромів.

Переваги:

- комбінування переваг централізованих та розподілених систем;

- зменшення ризику невдачі централізованої системи завдяки резервним джерелам даних.

Недоліки:

- Складність розгортання та управління гібридними системами.



4. Використання стандартів обміну даними: використання уніфікованих стандартів обміну метеоданими, таких як *METAR* та *TAF*, забезпечує однорідність та сумісність даних [1, 23].

Переваги:

- однаковий формат даних для різних джерел спостереження;
- легка інтеграція та обробка даних з різних джерел.

Недоліки:

– деякі стандарти можуть бути обмежені щодо видів метеоданих, які вони підтримують.

5. Використання сучасних технологій: хмарні обчислення, Інтернет речей (*IoT*) та штучний інтелект (*AI*), може спростити і прискорити інтеграцію метеоданих [15].

Переваги:

– забезпечення швидкого та автоматизованого обміну даними;

– можливість використовувати аналіз даних для покращення прогнозування погоди та прийняття рішень.

Недоліки:

– вартість впровадження рішення і обслуговування його роботи.

Ці методи інтеграції метеоданих є важливими для забезпечення безпеки та ефективності авіаційних операцій на аеродромах, і їх вибір залежить від конкретних потреб і обмежень кожного аеродрому.

Наведемо перелік деяких систем і стандартів обміну метеоданими, які використовуються в авіації та метеорології. Зауважте, що цей перелік не є вичерпним, і існують інші стандарти і системи, які можуть використовуватися в різних країнах та контекстах:

– *METAR* (*Meteorological Aerodrome Report*) – стандарт для звітування про погодні умови на аеродромах і для авіаційних потреб;

– *TAF* (*Terminal Aerodrome Forecast*) – стандарт для прогнозу погоди на аеродромах для авіаційних потреб;

– *SIGMET* (*Significant Meteorological Information*) – система повідомлень про значущу метеорологічну інформацію, яка може вплинути на безпеку польотів;

- *AIRMET (Airmen's Meteorological Information)* – система повідомлень про метеорологічну інформацію, яка може вплинути на безпеку повітряного руху;
- *AWOS (Automated Weather Observing System)* – автоматизована система спостереження за погодою на аеродромах;
- *ASOS (Automated Surface Observing System)* – автоматизована система спостереження за погодою на поверхні;
- *ADDS (Aviation Digital Data Service)* – система цифрового обміну метеоданими для авіації;
- *GTS (Global Telecommunication System)* – глобальна система телекомунікаційного обміну метеорологічними даними;
- *WMO (World Meteorological Organization) Information System (WIS)* – інформаційна система Всесвітньої метеорологічної організації для обміну метеоданими;
- *GRIB (GRIdded Binary)* – стандарт для обміну географічної метеорологічної інформацією у бінарному форматі;
- *BUFR (Binary Universal Form for the Representation of meteorological data)* – універсальна бінарна форма для представлення метеорологічних даних;
- *NetCDF (Network Common Data Form)* – відкритий стандартний формат для обміну метеорологічними і геофізичними даними;
- *JSON (JavaScript Object Notation)* – легковажний текстовий формат для обміну даними, який також може використовуватися для метеорологічних даних.

## 1.2. Огляд сучасних методів опитування і синхронізації даних у метеорології та авіації

У сучасному світі точність і доступність метеоданих є ключовими факторами для забезпечення безпеки та ефективності авіаційних операцій. Для цього використовуються різні методи опитування метеоданих на аеродромах та методи синхронізації даних між різними джерелами [6, 16, 14].

### 1.2.1. Принципи опитування метеоданих на аеродромах

Опитування метеоданих на аеродромах є важливою складовою процесу забезпечення безпеки і ефективності авіаційних операцій. Основним завданням цього процесу є збір актуальної та достовірної інформації про стан погоди на аеродромах, що включає в себе різні метеорологічні параметри, такі як температура повітря, вологість, атмосферний тиск, швидкість і напрям вітру, наявність опадів і багато інших.

Процес опитування метеоданих на аеродромах починається зі збору інформації від метеорологічних датчиків та приладів, розташованих на аеродромі. Ці дані надходять в реальному часі і вимірюються відповідно до встановлених стандартів і нормативів. Однак одного джерела інформації недостатньо для забезпечення безпеки польотів, оскільки погодні умови можуть різко змінюватися від одного аеродрому до іншого та від одного моменту часу до іншого.

Тому інтеграція даних з різних джерел стає обов'язковою. Сучасні системи опитування метеоданих на аеродромах можуть об'єднувати інформацію з різних метеорологічних станцій, супутників, радарів і навіть метеорологічних моделей. Ця інтеграція дозволяє отримувати більш повну і об'єктивну картину стану погоди в конкретному регіоні.

Крім того, опитування метеоданих повинно бути забезпечене високим рівнем автоматизації та надійності. Це важливо для того, щоб оператори та диспетчери могли оперативнo отримувати інформацію про зміни в погодних умовах і приймати відповідні рішення. Також, надійність процесу опитування метеоданих є ключовою для безпеки пасажирів і польотного екіпажу, оскільки недостовірна інформація про погоду може призвести до серйозних аварійних ситуацій.

Загалом, опитування метеоданих на аеродромах є складним і відповідальним процесом, який вимагає сучасних технологій та високої кваліфікації метеорологічного персоналу. Цей процес є необхідною ланкою у системі забезпечення безпеки авіаційних польотів і сприяє успішному виконанню авіаційних місій. Розглянемо основні принципи та методи опитування метеоданих на аеродромах:

– визначення потреб: перший крок у процесі опитування метеоданих – визначення конкретних потреб аеродрому. Це включає в себе визначення видів даних, які необхідні (такі як температура, атмосферний тиск, вітер, видимість і т. д.), частоту оновлення цих даних і їх точність;

– вибір джерел даних: для опитування метеоданих на аеродромі потрібно вибрати джерела, з яких ці дані будуть отримані. Це можуть бути прилади на самому аеродромі, дані з місцевих метеостанцій, супутникові дані, радари та інші джерела;

– збір та трансмісія даних: після вибору джерел даних, необхідно організувати їх збір та трансмісію до системи обробки метеоданих. Це може включати в себе встановлення метеорологічних приладів на аеродромі, використання датчиків, супутникового зв'язку та інфраструктури для передачі даних;

– системи моніторингу та контролю: слід встановлювати для постійного нагляду за процесом опитування метеоданих. Вони дозволяють вчасно виявляти та усувати можливі неполадки чи помилки у зборі даних;

– стандартизація даних: для забезпечення однорідності та сумісності даних важливо використовувати відповідні стандарти обміну даними, такі як *METAR* чи *TAF*. Це дозволяє легко інтегрувати дані з різних джерел і забезпечити їх спільне використання;

– автоматизація та аналіз: сучасні технології дозволяють автоматизувати багато процесів опитування та обробки метеоданих. Застосування штучного інтелекту та алгоритмів аналізу дозволяє швидше та точніше обробляти дані та надавати прогнози погоди;

– резервування та надійність: для забезпечення неперервного доступу до метеоданих важливо мати системи резервування та надійності. Це включає в себе резервні джерела даних, резервне електроживлення та інші заходи для уникнення перебоїв;

– актуалізація та покращення: процес опитування метеоданих на аеродромах має бути постійно оновлюваним і вдосконалюваним для відповіді на зміни у

вимогах та технологіях. Важливо слідкувати за новітніми розробками у сфері метеорології та авіації.

Здійснення опитування метеоданих на аеродромах відіграє ключову роль у забезпеченні безпеки та ефективності авіаційних операцій, а також у підвищенні точності прогнозів погоди, що є критичним для безпеки польотів.

#### 1.2.2. Методи синхронізації даних між різними джерелами

Синхронізація даних між різними джерелами метеоданих на аеродромах є критично важливим етапом у забезпеченні надійності та актуальності інформації для безпеки польотів [6].

Синхронізація метеоданих з різних джерел – це важливий аспект в процесі інтеграції метеоданих для забезпечення точності та надійності інформації, що використовується в авіаційних додатках та системах. Існують різні методи синхронізації даних, включаючи:

- потокова синхронізація: метеодані надходять в реальному часі з джерела в джерело. Всі зміни в даних оновлюються негайно та автоматично. Це дозволяє забезпечити актуальність інформації, але може стати проблемою при великій кількості джерел, оскільки це може призвести до перевантаження системи;

- пакетна синхронізація: дані оновлюються періодично, наприклад, кожну годину або щодня. Це може зменшити навантаження на систему, але може призвести до затримок у відображенні актуальної інформації;

- синхронізація за запитом: система запитує метеодані лише тоді, коли вони потрібні. Це зменшує обсяг даних, які потрібно синхронізувати, але може призвести до затримок у відповідях на запити.

Проблеми, які можуть виникнути під час синхронізації метеоданих, включають:

- несумісність форматів даних: різні джерела можуть використовувати різні формати для представлення метеоданих, і це може вимагати конвертації даних в єдиний стандартний формат;

- втрати даних: під час синхронізації можуть виникнути проблеми, такі як втрати даних або дублювання;

– затримки: якщо синхронізація не проводиться в режимі реального часу, це може призвести до затримок у відображенні актуальних даних;

– вартість інфраструктури: забезпечення ефективної синхронізації може вимагати великих інвестицій у відповідну інфраструктуру та обладнання;

– безпека даних: під час синхронізації важливо забезпечити захист метеоданих від несанкціонованого доступу або втрати.

Вирішення цих проблем вимагає ретельного проектування системи синхронізації, використання стандартів обміну даними та ретельного тестування для забезпечення якості і надійності метеоданих у авіаційних додатках та системах.

Організація програмно різних типів синхронізації метеоданих вимагає розробки відповідного алгоритму та використання відповідних програмних засобів. Розглянемо, як це можна організувати для кожного типу синхронізації:

#### 1. Поточкова синхронізація:

– необхідно реалізувати програму або модуль, який постійно слідкує за даними з різних джерел метеоданих.

– необхідно використовувати асинхронні запити або сокети для отримання даних в режимі реального часу.

– необхідно обробляти дані, як тільки вони надходять, і оновлюйте систему згідно з ними.

#### 2. Пакетна синхронізація:

– необхідно розробити регулярний розклад для синхронізації даних з різних джерел.

– необхідно використовувати автоматизований процес, який запускається заздалегідь визначеними інтервалами (наприклад, годину або день).

– необхідно отримувати дані з різних джерел, конвертуйте їх у єдиний формат і оновіть систему.

#### 3. Синхронізація за запитом:

– необхідно розробити систему, яка може створювати запити для отримання метеоданих, коли це потрібно.

– необхідно використовувати стандартні протоколи обміну даними, такі як *HTTP* або *FTP*, для отримання інформації з віддалених джерел.

– необхідно проводити запити до джерела метеоданих у відповідь на конкретні запити користувачів або системи автоматизованого прийняття рішень.

У всіх цих методах важливо враховувати стандарти обміну даними, перевірку безпеки та обробку помилок для забезпечення надійності та безпеки обміну метеоданими. Також слід враховувати обсяги даних, їхню швидкість та потреби користувачів при розробці власної системи синхронізації.

Розглянемо основні методи та підходи до синхронізації метеоданих:

– реальний час: один з найважливіших методів синхронізації – це забезпечення реального часу при отриманні та передачі даних. Це означає, що дані мають бути збережені та оновлювані миттєво, щоб вони були актуальними у весь час. Використання систем часової марки може допомогти забезпечити синхронізацію;

– протоколи передачі даних: використання стандартних протоколів передачі даних, таких як *HTTP*, *FTP*, *MQTT* тощо, дозволяє ефективно передавати і синхронізувати дані між різними джерелами. Це полегшує інтеграцію та обмін інформацією;

– мережеві технології: використання мережевих технологій, таких як *Ethernet*, *Wi-Fi*, *5G*, дозволяє стабільно та швидко передавати дані між різними пристроями та системами;

– резервне зберігання: для забезпечення надійності та доступності даних, їх можна зберігати в резервних системах. Це дозволяє уникнути втрати даних у випадку відмови основних джерел;

– автоматична синхронізація: використання автоматичних процесів синхронізації дозволяє підтримувати узгодженість даних між різними джерелами без необхідності вручну втручатися. Це покращує надійність та швидкість обміну даними;

– резервування каналів зв'язку: для забезпечення безперебійного обміну даними можна використовувати резервні канали зв'язку. Це дозволяє уникнути втрати даних у випадку відмови основного каналу;

– алгоритми обробки конфліктів: при одночасній синхронізації можуть виникати конфлікти між різними джерелами даних. Використання відповідних алгоритмів для вирішення конфліктів та збереження цілісності даних є важливим аспектом синхронізації;

– шифрування та захист даних: під час синхронізації даних, особливу увагу слід приділяти захисту даних від несанкціонованого доступу або перехоплення. Використання шифрування та захисту даних може допомогти уникнути загроз безпеці.

Синхронізація даних між різними джерелами метеоданих грає важливу роль у забезпеченні актуальності та доступності інформації для авіаційної безпеки та ефективності польотів. Цей процес вимагає врахування багатьох факторів, таких як швидкість передачі даних, надійність з'єднання, забезпечення захисту інформації та автоматизованих процесів синхронізації.

### 1.3. Аналіз існуючих стандартів і систем інтеграції метеоданих

Аналіз існуючих стандартів і систем інтеграції метеоданих є важливою складовою розробки ефективних систем збору, обробки, інтеграції та поширення метеорологічних даних. У світі існує багато різних стандартів, протоколів та систем, призначених для обміну метеоданими між різними джерелами, такими як метеорологічні станції, супутникові системи, метеорадари, аеродроми та користувачами цих даних. Аналіз цих стандартів та систем допомагає визначити найкращі практики та можливості для створення ефективних інтегрованих рішень у сфері метеорології.

Всесвітня метеорологічна організація (*WMO*) є провідною міжнародною організацією у галузі метеорології та кліматології. *WMO* розробляє та підтримує стандарти обміну метеорологічними даними для забезпечення співробітництва між



різними країнами та організаціями. Один з ключових стандартів *WMO* – це коди *METAR* і *TAF*, які використовуються для звітування метеорологічних даних у цивільній авіації. *METAR* (*Meteorological Aerodrome Report*) містить інформацію про метеорологічні умови на аеродромах, такі як температура, вологість, атмосферний тиск, вітер, видимість та інші параметри. *TAF* (*Terminal Aerodrome Forecast*) надає передбачення погоди на аеродромах.

*METAR* і *TAF* використовуються як стандартизований засіб обміну метеорологічними даними для авіаційної галузі. Вони забезпечують однаковий формат і структуру даних, що дозволяє авіаційним організаціям та пілотам легко розуміти і використовувати ці дані для прийняття рішень та безпечних польотів.

Національне управління океанів і атмосфери (*NOAA*) Сполучених Штатів відіграє важливу роль у розробці та використанні стандартів та систем інтеграції метеоданих. *NOAA* розробляє і підтримує різні протоколи та стандарти для передачі метеорологічних даних, зокрема *Aviation Weather Center (AWC) XML*, що використовується для надання метеоданих авіаційним організаціям та пілотам.

*AWC XML* містить інформацію про температуру повітря, вологість, вітер, опади та інші метеорологічні параметри. Цей стандарт дозволяє легко обмінюватися даними між різними системами та обробляти їх для авіаційних потреб. *NOAA* також надає різні прогностичні моделі та сервіси, які використовуються для передбачення погоди та клімату.

Європейський центр середньострокового прогнозу (*ECMWF*) – це один із найпрестижніших центрів для розробки прогностичних моделей та обміну метеорологічними даними в Європі та світі. *ECMWF* розробляє стандарти та протоколи, які використовуються для інтеграції метеоданих з різних джерел та їх подальшої обробки.

Один з ключових продуктів *ECMWF* – це формат *GRIB (GRIdded Binary)*, який використовується для зберігання та обміну геопросторовими даними, такими як результати прогнозів погоди. *GRIB* дозволяє представляти дані у компактному та стиснутому форматі, що зменшує обсяг даних при зберіганні та передачі. Це робить *GRIB* особливо корисним для потреб метеорології та кліматології.

*Open Geospatial Consortium (OGC)* – це міжнародна організація, яка розробляє та підтримує стандарти для обміну геопросторовими даними, включаючи метеорологічні дані. *OGC* визначає ряд стандартів, таких як *Web Map Service (WMS)*, *Web Feature Service (WFS)*, *Sensor Observation Service (SOS)* та інші, які дозволяють інтегрувати метеорологічні дані з геопросторовими системами.

*WMS* дозволяє отримувати зображення карт, які містять метеорологічні дані, у веб-форматі для подальшого відображення на картографічних веб-сайтах. *WFS* дозволяє отримувати геопросторові дані векторного формату для аналізу та обробки. *SOS* дозволяє отримувати та аналізувати реальні дані спостереження, включаючи метеорологічні дані, з різних джерел.

Поза міжнародними та національними стандартами існують також приватні системи та стандарти, які розроблені різними компаніями та організаціями. Наприклад, *IBM* розробив стандарт *WMO Common Alerting Protocol (CAP)*, який використовується для передачі погодних сповіщень та надзвичайних ситуацій.

Авіаційна галузь є однією з найбільш вимогливих щодо точності, доступності та актуальності метеорологічної інформації. Інтеграція метеоданих є невід'ємною частиною авіаційної безпеки і ефективності польотів. У цьому розділі буде проведений аналіз існуючих систем інтеграції метеоданих, які використовуються в авіаційній галузі, з огляду на їх роль, переваги і обмеження.

*AMOSS* – це комплексна система інтеграції метеоданих, яка спеціалізується на забезпеченні метеорологічної підтримки для авіаційних потреб. Система *AMOSS* розроблена для підтримки різних видів авіаційних діяльностей, включаючи цивільну авіацію, військову авіацію, аеропорти та авіаційні компанії. Головною метою *AMOSS* є надання точної та актуальної метеорологічної інформації для забезпечення безпеки та ефективності авіаційних операцій.

*AMOSS* інтегрує дані з різних джерел, включаючи метеорадари, метеостанції, супутникові системи та інші. Вона використовує стандартизовані формати, такі як *METAR* і *TAF (Terminal Aerodrome Forecast)*, для обміну інформацією між авіаційними організаціями і аеропортами. *AMOSS* також може враховувати

специфічні потреби різних типів польотів, включаючи маршрутні, регіональні та місцеві польоти.

Переваги системи *AMOSS* включають:

- висока точність та актуальність метеоданих.
- можливість інтеграції з різними джерелами інформації.
- спеціалізована підтримка для авіаційних потреб.
- забезпечення безпеки та ефективності авіаційних операцій.

Однак система *AMOSS* також має свої обмеження, включаючи високі витрати на обладнання та підтримку, складність інтеграції з іншими системами та можливість виникнення технічних проблем під час використання.

*AIXM* – це стандарт моделювання обміну інформацією про авіаційні дані, який розроблений Міжнародною організацією цивільної авіації (*ICAO*). Цей стандарт використовується для представлення метеорологічних інформаційних даних у вигляді структурованого і стандартизованого формату, що дозволяє легко обмінювати та інтегрувати ці дані в авіаційних системах.

*AIXM* визначає модель даних, яка описує різні аспекти метеорологічної інформації, такі як спостереження, прогнози та маршрути польотів. Цей стандарт дозволяє створювати структуровані дані про погоду, які можуть бути використані для автоматизованого оброблення та інтеграції в авіаційні системи.

Переваги *AIXM* включають:

- Інтеграція з іншими авіаційними стандартами та системами.
- Можливість створення даних у вигляді географічної інформації, що дозволяє враховувати просторові аспекти метеорологічних даних.
- Стандартизований формат для обміну даними між авіаційними організаціями.

Однак *AIXM* також має свої обмеження, включаючи необхідність великого обсягу даних для представлення повної метеорологічної інформації та складність впровадження у деяких авіаційних системах.

*IWXM* – це модель обміну метеорологічною інформацією, розроблена Міжнародною організацією цивільної авіації (*ICAO*) на основі *AIXM*. Ця модель

спеціалізується на представленні метеорологічних даних в авіаційному специфічному форматі. *IWXXM* включає в себе структуровані дані про спостереження, прогнози, та іншу метеорологічну інформацію.

*IWXXM* спрямована на поліпшення точності та доступності метеорологічних даних для авіаційної галузі. Вона використовує *XML*-подібний формат даних, що дозволяє легко обмінювати та інтегрувати ці дані в авіаційних системах.

Переваги *IWXXM* включають:

– використовує стандартизовані формати для представлення метеорологічної інформації. Це дозволяє різним авіаційним службам та організаціям легко обмінюватися даними та забезпечує єдність інформації в авіаційній галузі.

– спрощує обробку метеорологічних даних за допомогою структурованих форматів. Це дозволяє автоматизувати процеси збору, аналізу та передачі інформації, що підвищує ефективність та точність обробки.

– є стандартом, прийнятим Міжнародною організацією цивільної авіації (*ICAO*), що сприяє єднанню авіаційної галузі на міжнародному рівні. Це робить можливим обмін метеорологічною інформацією між різними країнами та регіонами.

– дозволяє представляти як актуальні метеорологічні спостереження, так і прогнози на різні періоди часу. Це важливо для забезпечення авіапilotів актуальною інформацією для планування та виконання польотів.

– Стандартизований формат даних допомагає зменшити помилки при передачі та обробці інформації, що важливо для безпеки польотів.

– може легко інтегруватися з іншими авіаційними системами, що дозволяє використовувати метеорологічну інформацію для навігації, планування маршрутів та безпеки польотів.

– забезпечує оперативний обмін метеорологічною інформацією, що допомагає пілотам та диспетчерам мати доступ до актуальних даних для прийняття рішень під час польоту.

Завдяки моделі *IWXXM*, авіаційна галузь може забезпечувати авіапilotів та диспетчерів актуальною та структурованою метеорологічною інформацією, що сприяє безпеці та ефективності польотів.

#### 1.4. Висновки до розділу

В розділі надано важливі відомості про роль метеоданих у контексті авіаційної безпеки та методи їх інтеграції.

Визначено, що метеодані відіграють важливу роль у підвищенні безпеки та ефективності авіаційних польотів. Правильна інтерпретація метеоданих дозволяє уникати небезпечних ситуацій та забезпечує надійність авіаційних операцій. Огляд існуючих методів інтеграції метеоданих свідчить про різноманітність підходів до об'єднання метеорологічної інформації з різних джерел. Інтеграція метеоданих вимагає розробки систем, які забезпечують доступність і точність даних.

Аналіз існуючих стандартів і систем інтеграції метеоданих підкреслює необхідність стандартизації та уніфікації процесу обміну метеорологічною інформацією для забезпечення сумісності та зручності взаємодії між системами.

## РОЗДІЛ 2

### ПРОЄКТУВАННЯ СИСТЕМИ ІНТЕГРАЦІЇ МЕТЕОДАНИХ

#### 2.1. Формування дерева функцій системи інтеграції метеоданих на малих аеродромах

##### 2.1.1. Визначення функціональних вимог до системи

Дерево функцій або "функціональне дерево" – це інструмент у системному аналізі та проектуванні, який допомагає розкласти великий проект на менші функціональні компоненти або модулі, щоб легше розуміти і керувати ними. У вас є можливість створити дерево функцій для системи інтеграції метеоданих на аеродромах, щоб краще визначити, які функції системи будуть виконуватися та як вони взаємодіятимуть між собою (рис. 2.1).

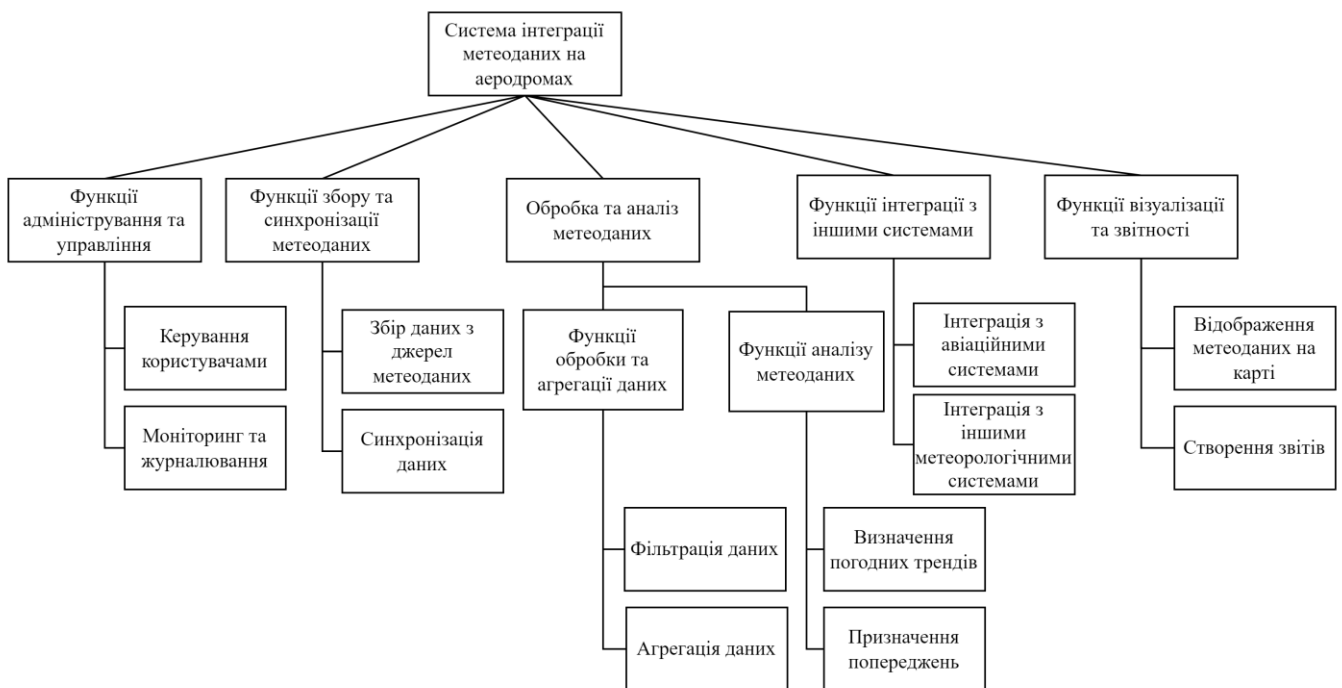


Рис. 2.1. Дерево функцій системи інтеграції даних про метеоумови на малих аеродромах

Першим кроком у розробці системи інтеграції метеоданих на аеродромах є визначення функціональних вимог до системи. Це важливий етап, оскільки визначає, які функції система повинна виконувати для досягнення своєї загальної

мети. Функціональні вимоги визначаються на основі потреб користувачів і вимог галузі авіації. Ось приклади функціональних вимог до системи інтеграції метеоданих:

1. Отримання метеорологічних даних:

– Система повинна забезпечити можливість отримання метеорологічних даних з різних джерел, включаючи сенсори на аеродромах, онлайніві ресурси та метеорологічні агентства.

– Система повинна підтримувати отримання даних в режимі реального часу та архівних даних.

2. Обробка та аналіз даних:

– Система повинна забезпечити можливість фільтрації та калібрування отриманих даних для забезпечення їхньої точності та надійності.

– Система повинна агрегувати дані з різних джерел для створення комплексних метеорологічних звітів та прогнозів.

3. Вибір джерел метеоданих:

– Система повинна мати алгоритми вибору джерел метеоданих в залежності від потреби та доступності даних.

– Система повинна автоматично синхронізувати та інтегрувати дані з різних джерел.

4. Система прийняття рішень:

– Система повинна розробити алгоритми прийняття рішень на основі аналізу метеорологічних даних.

– Система повинна забезпечити автоматичну реакцію на зміни погодних умов, включаючи видачу попереджень та рекомендацій.

5. Інтерфейс користувача:

– Система повинна мати веб-інтерфейс для відображення метеоданих користувачам.

– Система повинна забезпечити можливість користувачам звертатися до системи для отримання актуальних метеорологічних даних.

Це лише загальні функціональні вимоги, і подальший аналіз може визначити додаткові конкретні функції та деталі системи. Важливо зрозуміти потреби користувачів та галузі авіації, щоб визначити, як система повинна функціонувати, і відповідно до цього сформулювати функціональні вимоги.

#### 2.1.2. Функціональне моделювання роботи системи та її компонентів.

Моделювання в рамках методології *IDEF0* (*Integration DEFinition for Function Modeling*) використовується для аналізу та опису функцій системи або процесу. *IDEF0* допомагає структурувати та візуалізувати функції системи та їхні взаємозв'язки, що робить цей метод ідеальним для моделювання бізнес-процесів, проектування систем, та інших завдань, де важливо розуміти, як функції взаємодіють одна з одною.

Основними елементами моделі *IDEF0* є:

1. Функціональний блок (*Function Box*): Це прямокутник, який представляє функцію системи або процесу. Всередині блоку зазвичай зазначається короткий опис функції. Кожен функціональний блок має свій унікальний ідентифікатор.

2. Вхідні стрілки (*Input Arrows*): Стрілки, які входять у функціональний блок і показують вхідні дані або ресурси, які потрібні для виконання функції.

3. Вихідні стрілки (*Output Arrows*): Стрілки, які виходять з функціонального блоку і показують результати або виводи функції.

4. Керуюча стрілка (*Control Arrow*): Стрілка, яка показує, як керування передається між функціональними блоками. Це може бути керуванням послідовністю виконання функцій, управлінням ресурсами, або іншими аспектами.

Процес моделювання в *IDEF0* можна описати наступним чином:

1. Визначення цілей і об'єкта моделювання: Спочатку визначаються цілі моделювання і об'єкт, який буде аналізуватися. Це допомагає зрозуміти, що саме потрібно моделювати та до яких цілей це буде використовуватися.

2. Визначення функцій: Визначаються всі функції, які входять до системи або процесу. Кожна функція представляється окремим функціональним блоком.

3. Визначення вхідних та вихідних стрілок: Для кожного функціонального блоку визначаються вхідні та вихідні стрілки. Вхідні стрілки показують, що



потрібно для виконання функції, а вихідні стрілки показують, що вона виробляє або передає.

4. Визначення керуючих стрілок: Якщо в системі або процесі є керуючі елементи або управління послідовністю функцій, то це моделюється за допомогою керуючих стрілок.

5. Побудова моделі: Всі функціональні блоки та їх зв'язки відображаються на схемі *IDEF0*.

6. Аналіз та оптимізація: Модель аналізується для виявлення можливостей оптимізації та вдосконалення системи або процесу.

7. Документація: Модель документується для подальшого використання та розуміння системи або процесу.

Модель *IDEF0* допомагає командам проекту (рис. 2.2), аналітикам та інженерам розуміти функції системи та їхні зв'язки, що полегшує аналіз, оптимізацію та управління процесами та проектами.

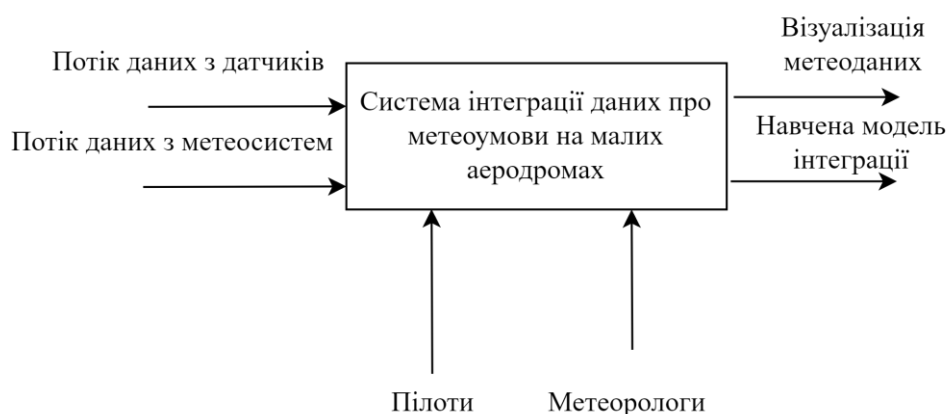


Рис. 2.2. Представлення системи інтеграції даних про метеоумови для малих аеродромів в форматі моделі *IDEF0*

Проведення декомпозиції моделі *IDEF0* для системи інтеграції даних про метеоумови на малих аеродромах – це завдання, яке включає детальний аналіз системи та її функцій. Цей процес важливий для розуміння та опису внутрішньої структури та логіки роботи системи. Декомпозиція моделі *IDEF0* допомагає

розділити систему на більш малий компоненти, модулі та функції, що спрощує аналіз та керування їми.

Першим кроком у проведенні декомпозиції моделі *IDEF0* є визначення вищого рівня функцій, які система повинна виконувати. Наприклад, в системі інтеграції даних про метеоумови на малих аеродромах основною функцією може бути збір, обробка та представлення метеорологічних даних. Ця функція може бути поділена на декілька підфункцій, таких як отримання даних від різних джерел, аналіз та обробка даних, створення графічних представлень інформації та інші.

Далі, кожен з цих підфункцій можна поділити на ще менші компоненти. Наприклад, функція "отримання даних від різних джерел" може включати в себе підфункції, які відповідають за з'єднання з метеорологічними станціями, супутниковими системами, радаром тощо. Кожна з цих підфункцій може бути поділена на ще менші елементи, до яких можна призначити конкретні завдання та обов'язки.

Декомпозиція моделі *IDEF0* дозволяє детально вивчити всі аспекти системи, включаючи взаємодію між різними компонентами, передачу даних та потік робочих процесів. Цей аналіз є важливим етапом у проектуванні системи і дозволяє створити чітку структуру, яка допоможе впорядкувати та оптимізувати роботу системи інтеграції даних про метеоумови на малих аеродромах.

Декомпозиція моделі *IDEF0* також допомагає зрозуміти, які ресурси, обладнання та програмні засоби потрібні для виконання кожної функції та підфункції. Це дозволяє зробити більш точну оцінку витрат і ресурсів, необхідних для реалізації системи, а також розробити план впровадження та розвитку проекту. Модель *IDEF0* використовується для представлення функціональної структури системи і включає рівні декомпозиції, де кожен вищий рівень розбивається на більш детальні підсистеми. Ця модель використовується для представлення функціональної структури системи та включає кілька рівнів декомпозиції, які дозволяють детально розглянути кожен аспект системи. На найвищому рівні, модель представляє загальну функціональність системи або процесу.

Кожен процес на контекстному діаграмі потім розбивається на більш деталізовані підсистеми на наступних рівнях декомпозиції. На кожному рівні декомпозиції, процеси поділяються на більш дрібні, більш управлінні частини. Це дозволяє аналітикам і розробникам системи детально розглянути окремі функції та їх взаємодію з іншими частинами системи.

1. Визначення мети та обсягу системи:

- ясно визначити цільову систему, її мету та основні функціональні області.
- визначити основні вхідні та вихідні дані системи, а також основні ресурси та механізми, які будуть використовуватися.

2. Створення контекстної діаграми:

- намалювати контекстну діаграму на найвищому рівні (A-0), яка показує зовнішні взаємодії системи з користувачами, іншими системами, та оточенням.

3. Розробка діаграми першого рівня (A0):

- розбити систему на основні функції та представити їх у вигляді діаграми першого рівня (рис. 2.3, 2.4).
- кожна функція повинна мати чітко визначені вхідні, вихідні, ресурси та механізми.

4. Опис процесів та взаємодій:

- Для кожної функції другого рівня описати детальні процеси, які відбуваються всередині, та їх взаємодії з іншими підсистемами.

5. Визначення контрольних точок:

- Встановити точки контролю та оцінки працездатності для кожної з функцій на всіх рівнях декомпозиції.

6. Інтеграція з іншими системами:– Визначити, як дані про метеоумови інтегруються з іншими системами аеродрому, включаючи польотні операції, безпеку та логістику.

7. Верифікація та валідація моделі:

- Перевірити модель на консистентність та повноту, забезпечити, що всі важливі функції враховані та правильно представлені.

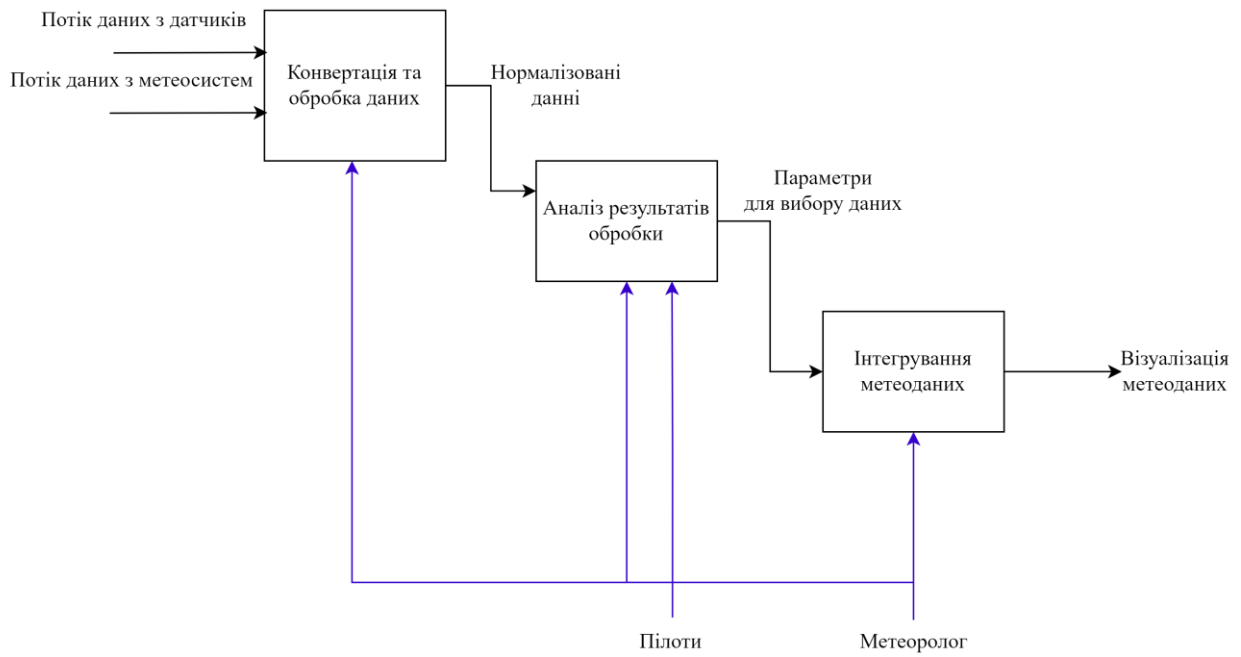


Рис. 2.3. Представлення візуалізації метеоданих в форматі моделі *IDEF0*

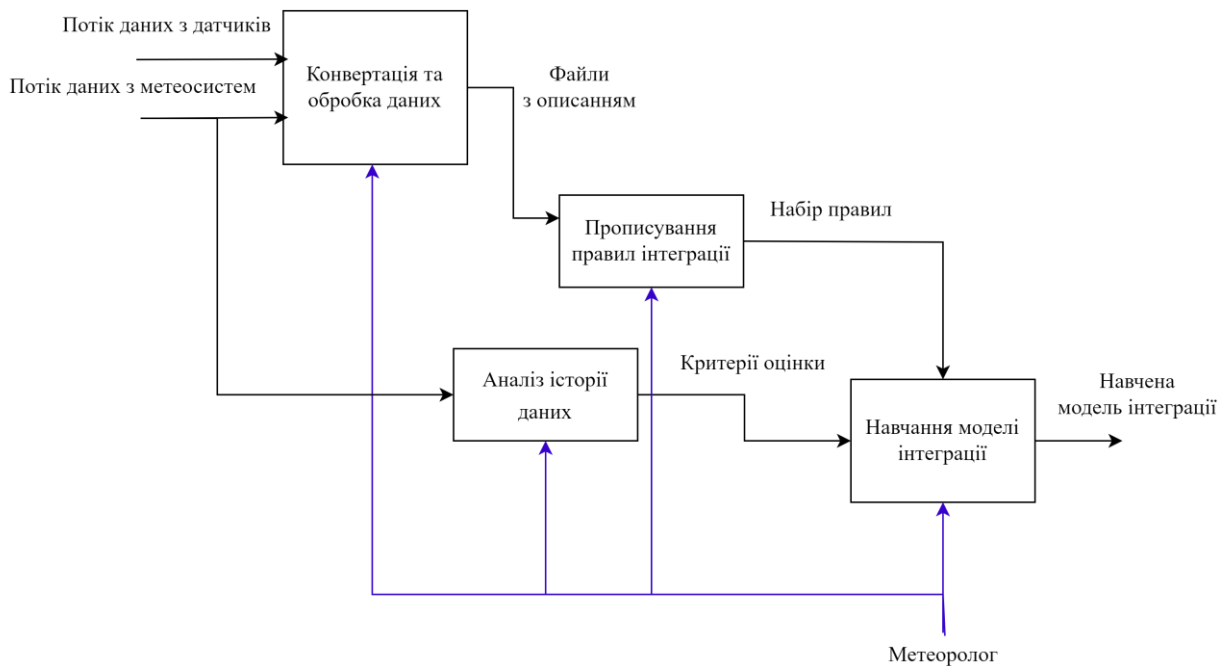


Рис. 2.4. Представлення навчання моделі інтеграції в форматі моделі *IDEF0*

## 2.2. Архітектура системи та визначення вимог до неї

Архітектура системи та визначення вимог до неї є критичним етапом в процесі проектування системи інтеграції метеоданих на аеродромах. Цей етап визначає

структуру, компоненти та спосіб взаємодії всієї системи, щоб вона відповідала функціональним та технічним вимогам.

Для ефективної роботи системи інтеграції метеоданих на аеродромах визначено вибір архітектурного підходу. Один із ключових архітектурних підходів - це багатоуровнева архітектура, що передбачає розділення системи на декілька рівнів, кожен з яких виконує свої функції та має визначені вимоги.

На рівні логіки бізнес-процесів системи необхідно розробити ряд модулів, що відповідають за обробку, аналіз та інтеграцію метеоданих. Ці модулі забезпечують отримання даних з різних джерел, їх обробку, синхронізацію та подальший аналіз.

Під час визначення технічних вимог до системи інтеграції метеоданих необхідно врахувати наступні аспекти:

- продуктивність: система повинна бути здатна обробляти та аналізувати великий обсяг метеоданих в реальному часі. Для цього визначено потребу у потужних серверах і оптимізації запитів до бази даних;

- надійність: забезпечення безперебійної роботи системи має велике значення в авіаційній галузі. Тому було встановлено вимогу до резервного копіювання даних, моніторингу стану системи та автоматичного виявлення та виправлення помилок;

- безпека: забезпечення конфіденційності та цілісності метеоданих є важливим аспектом. Для цього використовуються засоби аутентифікації, авторизації та шифрування даних;

- розширюваність: система повинна бути готовою до росту обсягу даних та користувачів. Для цього розроблено механізми автоматичного масштабування та розширювання інфраструктури;

- сумісність: враховуючи різноманітність джерел метеоданих, система повинна бути сумісною з різними форматами та протоколами обміну даними.

Для оптимізації процесу опитування та синхронізації метеоданих з різних джерел розробляється докладна модель роботи системи. Ця модель включає в себе алгоритми, протоколи та методи синхронізації даних, які забезпечують надійність та точність метеорологічних даних. Моделювання процесів опитування та

синхронізації дозволило виявити потенційні проблеми та вдосконалити роботу системи.

Вибір інструментів для аналізу та обробки метеорологічних даних є важливим етапом проектування системи. Враховуючи обсяг та складність даних, були необхідно обрати потужні інструменти для обробки даних, генерації звітів, візуалізації графіків та аналізу результатів.

#### 2.2.1. Вибір архітектурного підходу до системи

Вибір архітектурного підходу до системи інтеграції метеоданих на аеродромах є важливим етапом в розробці проекту. Архітектурний підхід визначає, як система буде побудована, організована та взаємодіяти з іншими компонентами. Ось детальний розгляд цього етапу:

1. Аналіз вимог до архітектури: Першим кроком є аналіз вимог до системи і визначення того, які функції та характеристики вимагаються для досягнення поставлених завдань. Це включає в себе розуміння потреб користувачів, технічних вимог та обмежень, а також структури даних, які потрібно інтегрувати.

2. Вибір архітектурного стилю: На основі аналізу вимог можна вибрати архітектурний стиль, який найкраще відповідає потребам проекту. Деякі з популярних архітектурних стилів, які можуть бути релевантними для системи інтеграції метеоданих, включають:

– Клієнт-сервер: Деякі компоненти системи можуть бути розглянуті як клієнти, які взаємодіють з централізованим сервером для отримання даних.

– Мікросервіси: Систему може бути побудовано з використанням окремих мікросервісів, кожен з яких відповідає за виконання певної функції.

– Схема з великою кількістю рівнів (*n-tier*): Систему можна розглядати як комбінацію різних рівнів або шарів, які взаємодіють між собою.

3. Визначення архітектурних компонентів: Після вибору архітектурного стилю визначаються конкретні архітектурні компоненти системи. Це можуть бути сервери, бази даних, клієнтські додатки, мікросервіси тощо.

4. Узгодження з вимогами до даних: Однією з ключових задач на цьому етапі є узгодження архітектури з вимогами до даних. Це включає в себе визначення, як дані будуть зберігатися, оброблятися та передаватися між компонентами системи.

5. Оцінка архітектурних рішень: Важливо провести оцінку обраних архітектурних рішень з точки зору їхньої відповідності вимогам, ефективності, масштабованості та інших параметрів. Також слід враховувати технічні обмеження та ресурси, доступні для розробки та підтримки системи.

6. Документація архітектури: Усі вибрані архітектурні рішення та компоненти системи повинні бути детально задокументовані для подальшої розробки, тестування та підтримки.

Вибір архітектурного підходу є критичним для успішної розробки системи інтеграції метеоданих на аеродромах, оскільки він визначає основну структуру та спосіб функціонування системи. Важливо враховувати потреби користувачів, технічні вимоги та інші фактори для вибору найкращого архітектурного рішення.

#### 2.2.2. Визначення технічних вимог до системи

Після визначення архітектурного підходу до системи інтеграції метеоданих на аеродромах, наступним важливим кроком є визначення технічних вимог до самої системи. Технічні вимоги визначають, як система повинна бути реалізована технічно, включаючи апаратні та програмні аспекти. Ось кроки і аспекти, які потрібно врахувати при визначенні технічних вимог:

1. Вибір технологічних платформ: визначення технологічних платформ, на яких буде реалізована система. Це може включати в себе вибір операційної системи, мов програмування, баз даних, фреймворків;

2. Апаратні вимоги: визначення апаратних вимог до обладнання, на якому буде встановлена система. Це включає в себе характеристики серверів, мережевого обладнання, сенсорів та інших пристроїв;

3. Безпека та конфіденційність даних: розробка вимог щодо забезпечення безпеки даних, включаючи методи шифрування, автентифікації, контролю доступу та захисту від вторгнень;

4. Інтеграція з іншими системами: визначення вимог щодо інтеграції з іншими системами, які можуть надавати метеорологічні дані або іншу інформацію;

5. Вимоги до продуктивності: встановлення вимог щодо продуктивності системи, включаючи час відповіді, пропускну спроможність та завантаження;

6. Тестування та якість: визначення вимог до тестування системи, включаючи тестування продуктивності, безпеки та інші аспекти якості;

7. Масштабованість: встановлення вимог щодо можливості масштабування системи у випадку зростання обсягів даних або користувачів;

8. Документація та навчання: вимоги до документації системи, включаючи технічну документацію для розробників та користувачів, а також навчальні матеріали;

9. План підтримки та супроводу: визначення вимог до підтримки, апгрейду та супроводу системи після введення в експлуатацію;

10. Управління змінами: розробка вимог щодо управління змінами в системі, включаючи процедури внесення змін та контроль версій.

Визначення технічних вимог до системи є важливим етапом, оскільки воно визначає, як система буде реалізована та підтримуватися в майбутньому. Ці вимоги допомагають забезпечити успішну розробку та ефективну експлуатацію системи інтеграції метеоданих.

### 2.3. Моделювання процесів опитування та синхронізації метеоданих з різних джерел

В системі передбачено наступні *UML*-діаграми:

– *Use Case Diagram* (діаграма використання): зображуються основні актори (користувачі, системи) і використана функціональність, відповідно до дерева функцій;

– *Activity Diagram* (діаграма активностей): для деталізації складних функцій або процесів, зокрема для функцій, пов'язаних з обробкою метеоданих або синхронізацією;



- *Sequence Diagram* (діаграма послідовності): для ілюстрації взаємодій між різними частинами системи під час виконання функцій;
- *Class Diagram* (діаграма класів): для класів та об'єктів, які використовуються системою для зберігання та обробки метеоданих;
- *Deployment Diagram* (діаграма розгортання): система розгортається на різних серверах, тому необхідна діаграма розгортання для показу архітектури системи;
- *Component Diagram* (діаграма компонентів): для показу компонентів системи та їхніх зв'язків.

Діаграма використання (*Use Case Diagram*) – це вид діаграми *UML*, який використовується для моделювання функціональності системи з точки зору користувачів (акторів) та їхніх взаємодій з системою. Ця діаграма допомагає візуалізувати, які функції або можливості надаються користувачам системи та як користувачі взаємодіють з нею.

Основні елементи діаграми використання:

1. Актори (*Actors*): Актори представляють зовнішніх сутностей, такі як користувачі, інші системи або зовнішні компоненти, які взаємодіють з системою. Актори зображуються у вигляді піктограм людини або будь-якої іншої сутності. У вашій системі для інтеграції метеоданих на аеродромах можуть бути такі актори, як "Метеоролог", "Адміністратор системи", "Пілот" тощо.

2. Використані сценарії (*Use Cases*): Використані сценарії або "функціональність" описують конкретні дії або функції, які можуть бути виконані користувачами або акторами системи. Вони зображуються у вигляді овалів з назвами. Наприклад, використані сценарії можуть включати "Перегляд метеоданих", "Оновлення метеоданих", "Створення звіту" тощо.

3. Зв'язки (*Relationships*): Зв'язки показують, як актори взаємодіють з використаними сценаріями. Зв'язки позначаються стрілками, які вказують на спосіб взаємодії. Наприклад, актор "Метеоролог" може мати зв'язок з використаним сценарієм "Оновлення метеоданих", що показує, що метеоролог може виконувати цей сценарій.

4. Система (*System*): Система (або корисна система) представляє об'єкт, який використовується користувачами для досягнення їхніх цілей. У вашому випадку, це може бути система інтеграції метеоданих на аеродромах.

Діаграма використання допомагає уточнити вимоги користувачів і визначити функціональність системи з точки зору зовнішніх користувачів. Можна створити таку діаграму для вашої системи інтеграції метеоданих на аеродромах, включивши акторів (користувачів та системи), функції, які вони можуть виконувати, і зв'язки між ними (рис. 2.5).

Для системи інтеграції метеоданих на аеродромах можна створити діаграму використання з такими акторами та використаними сценаріями:

Актори (*Actors*):

1. Метеоролог: відповідає за збір, оновлення та аналіз метеорологічних даних на аеродромах;

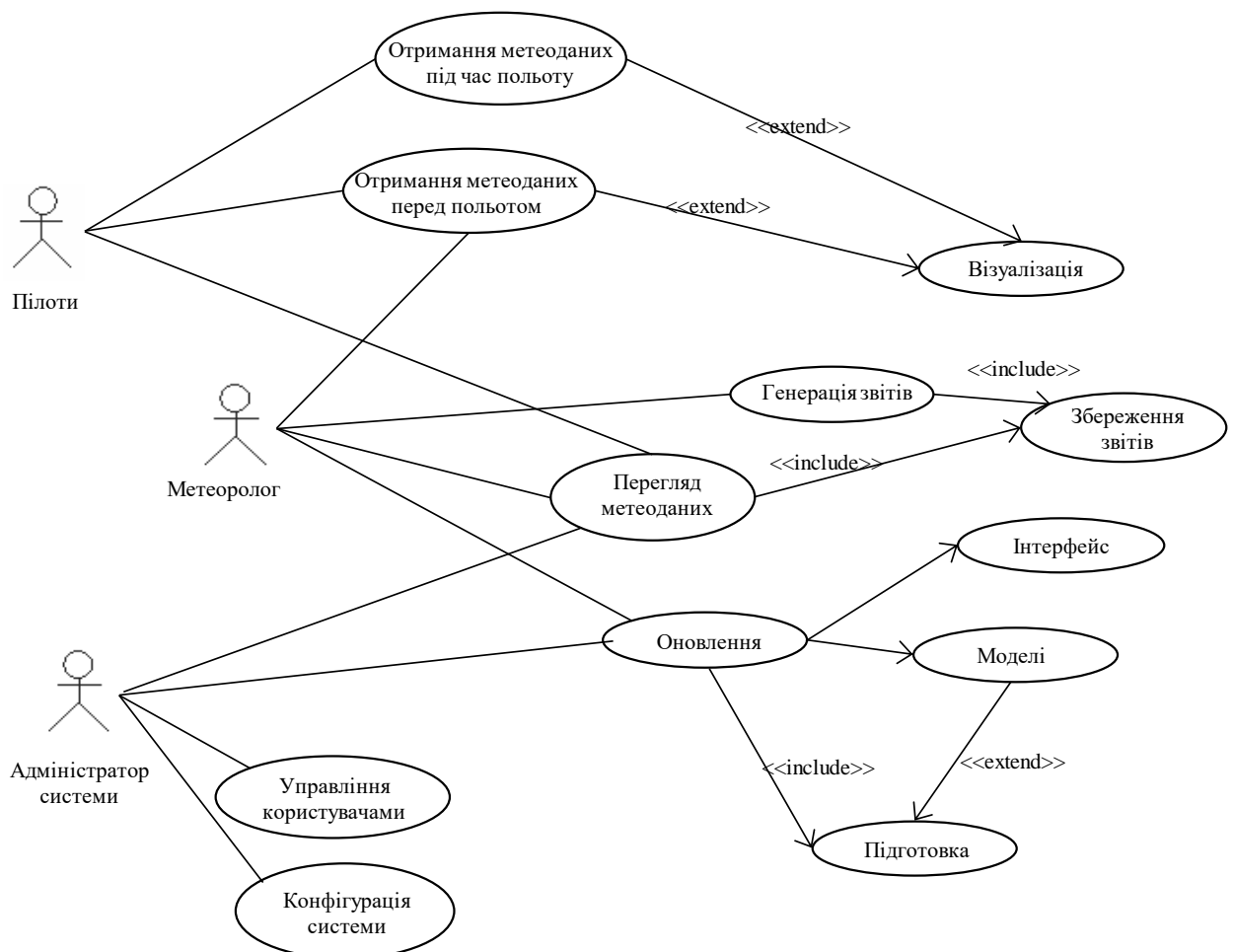


Рис. 2.5. Діаграма використання системи інтеграції даних про метеоумови на малих аеродромах

2. Адміністратор системи: має право доступу до системи та відповідає за її конфігурацію, безпеку та управління користувачами;

3. Пілоти: використовують систему для отримання актуальних метеоданих перед польотами та під час них.

Використані сценарії (*Use Cases*):

1. Перегляд метеоданих: усі актори можуть переглядати актуальні метеорологічні дані для аеродромів;

2. Оновлення метеоданих: метеоролог може завантажувати нові дані з метеорологічних джерел та оновлювати існуючі дані в системі;

3. Конфігурація системи: адміністратор може налаштовувати параметри системи, включаючи інтервали синхронізації та доступні джерела даних;

4. Управління користувачами: адміністратор може додавати, видаляти або змінювати права доступу користувачів до системи;

5. Отримання метеоданих перед польотом: пілот може запитувати актуальні дані перед польотом для прийняття рішень про маршрут та умови польоту;

6. Отримання метеоданих під час польоту: пілот може запитувати дані під час польоту для адаптації до змінних метеорологічних умов;

7. Генерація звітів: метеоролог може створювати звіти на основі зібраних метеоданих.

Ці використані сценарії відображають ключові функції та можливості вашої системи інтеграції метеоданих на аеродромах та взаємодію акторів з нею.

Діаграма активностей (*Activity Diagram*) в *UML* використовується для відображення послідовності дій та активностей в системі. Ваша система інтеграції метеоданих на аеродромах може бути представлена за допомогою такої діаграми для конкретного сценарію, наприклад, "Оновлення метеоданих". Ось опис можливої діаграми активностей для цього сценарію:

Назва діаграми: Оновлення метеоданих (рис. 2.6).

Активності та дії:

1. Початок (*Start*): Початок сценарію;

2. Вибір джерела даних (*Select Data Source*): Метеоролог обирає джерело метеоданих, з якого потрібно отримати оновлену інформацію (наприклад, метеостанція, супутникова система);

3. Запит даних (*Request Data*): Метеоролог створює запит до обраного джерела даних для отримання актуальних метеоданих;

4. Отримання даних (*Receive Data*): Система отримує дані від обраного джерела;

5. Оновлення бази даних (*Update Database*): Отримані дані оновлюють відповідні записи в базі даних;

6. Завершення (*End*): Завершення сценарію.

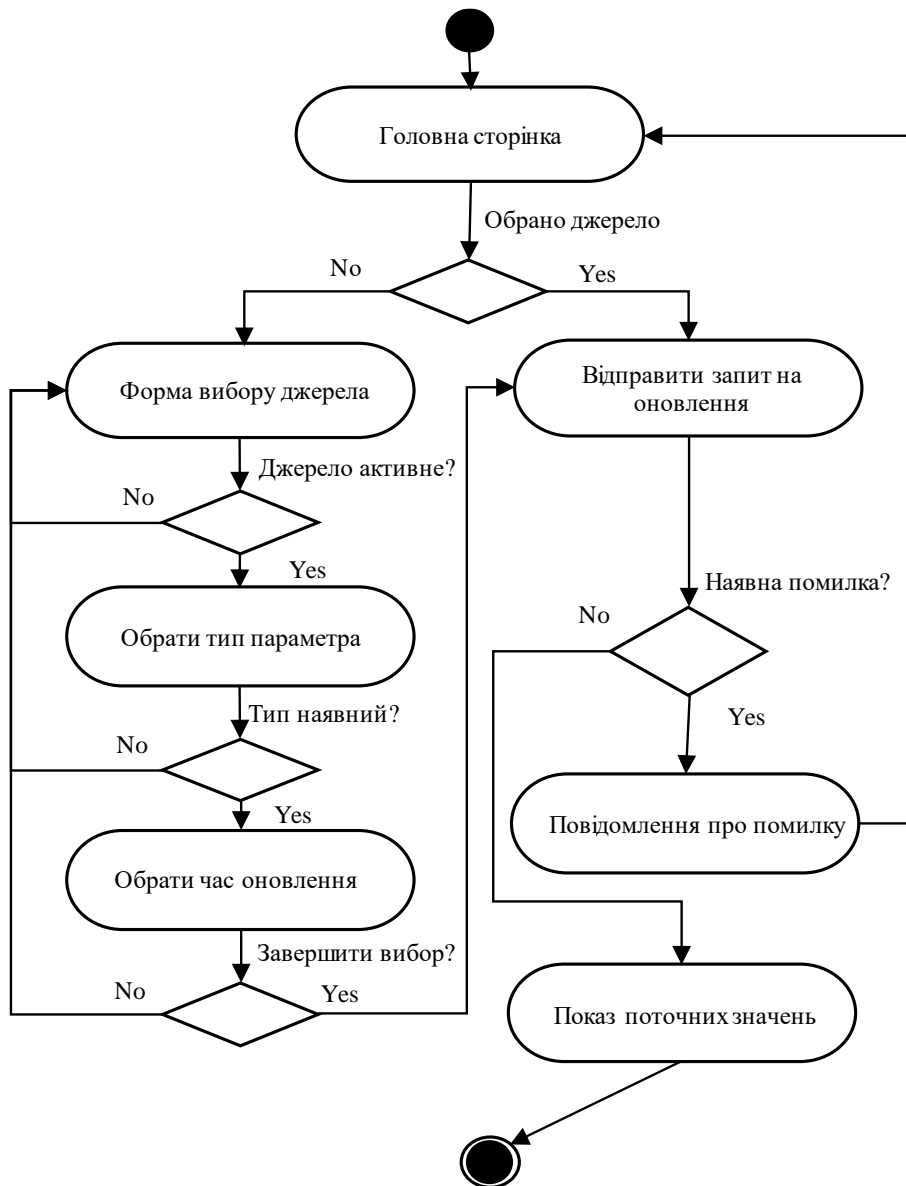


Рис. 2.6. Активності при оновленні метеоданих

### Рішення (*Decisions*):

– Після кожного кроку "Запит даних", може виникнути рішення про те, чи отримані дані вдалося успішно, або потрібно вжити додаткові дії в разі невдалого запити.

### Виключення (*Exceptions*):

– Якщо отримання даних невдається, може виникнути виняток, і система може спробувати інше джерело даних або згенерувати повідомлення про помилку.

Ця діаграма активностей ілюструє послідовність кроків, які виконуються під час сценарію оновлення метеоданих в вашій системі. Вона допомагає візуалізувати логіку та послідовність дій для цього конкретного процесу.

Діаграма послідовності (*Sequence Diagram*) в *UML* використовується для моделювання послідовності повідомлень між об'єктами або компонентами системи в конкретному сценарії. Давайте розглянемо можливу діаграму послідовності для сценарію "Запит метеоданих" у вашій системі інтеграції метеоданих на аеродромах:

### Учасники:

- Користувач: користувач системи, який ініціює запит метеоданих;
- Система інтеграції метеоданих: основна система, яка виконує інтеграцію та обробку метеоданих;
- Джерело метеоданих: Обране джерело, з якого потрібно отримати метеодані (наприклад, метеостанція або супутникова система);

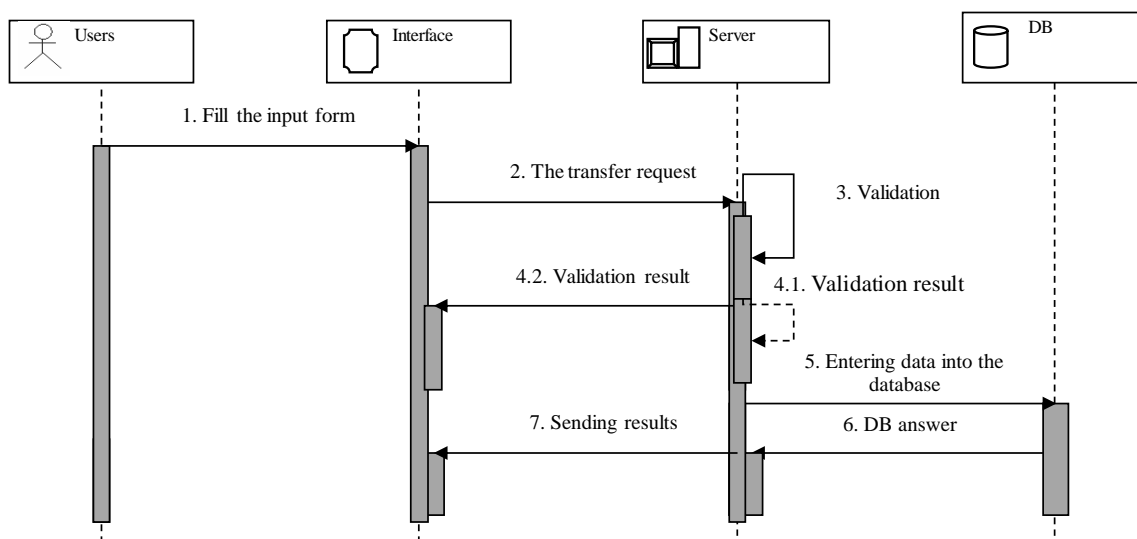


Рис. 2.7. Діаграма послідовності дій при запиті метеоданих

Послідовність подій:

1. Користувач ініціює запит метеоданих, вибравши конкретний аеродром та параметри запиту;
2. Користувач надсилає запит на сервер Системи інтеграції метеоданих;
3. Система інтеграції метеоданих приймає запит та аналізує його параметри;
4. Система інтеграції метеоданих визначає, з якого Джерела метеоданих потрібно отримати дані на основі обраних параметрів;
5. Система інтеграції метеоданих надсилає запит до вибраного Джерела метеоданих;
6. Джерело метеоданих обробляє запит та генерує метеодані;
7. Джерело метеоданих надсилає отримані метеодані до Системи інтеграції метеоданих;
8. Система інтеграції метеоданих отримує метеодані та зберігає їх у внутрішній базі даних;
9. Система інтеграції метеоданих повертає відповідь з метеоданими до Користувача.
10. Користувач отримує результат запиту та відображає метеодані.

Ця діаграма послідовності ілюструє взаємодію між різними учасниками в сценарії "Запит метеоданих" вашої системи. Вона дозволяє візуалізувати, як інформація передається від одного об'єкта до іншого та як обробляються запити на отримання метеоданих.

Діаграма розгортання (*Deployment Diagram*) в *UML* використовується для моделювання фізичного розгортання компонентів та системи на апаратних ресурсах, таких як сервери, комп'ютери, мережеве обладнання тощо. Давайте опишемо діаграму розгортання для вашої системи інтеграції метеоданих на аеродромах.

Компоненти та артефакти:

1. Користувачі: це можуть бути авіаперсонал, диспетчери, метеорологи та інші користувачі системи, які взаємодіють з нею через веб-інтерфейс або інші додатки.
2. Сервер додатку інтеграції метеоданих: цей сервер відповідає за обробку запитів користувачів, інтеграцію з джерелами метеоданих і зберігання даних.

3. Джерела метеоданих: це можуть бути метеостанції, супутникові системи, метеорадари та інші джерела, які надають метеодані.

4. База даних метеоданих: в цій базі даних зберігаються отримані метеодані.

5. Сервер бази даних: цей сервер відповідає за забезпечення доступу до бази даних метеоданих (рис. 2.7).

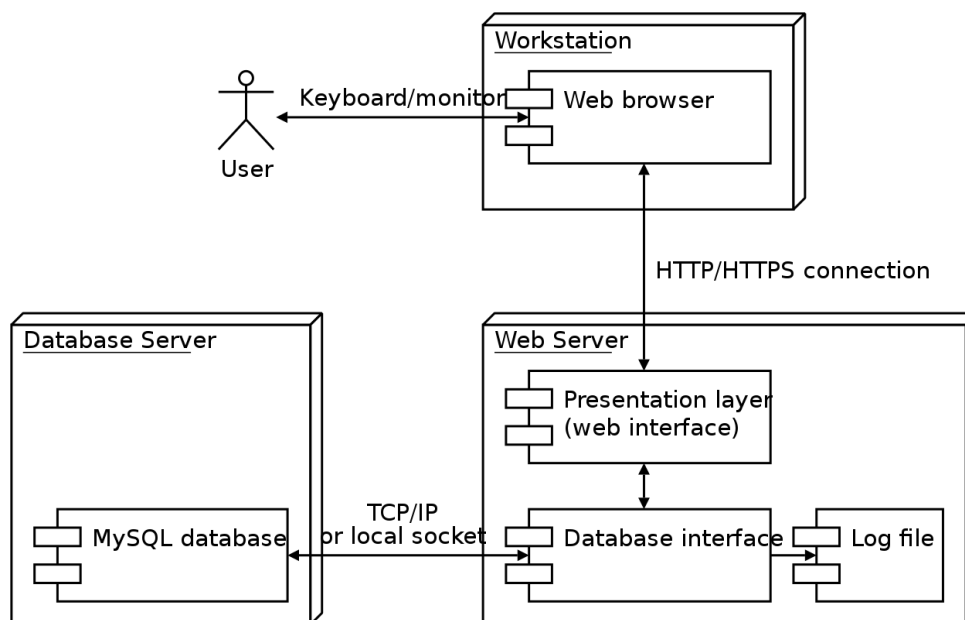


Рис. 2.8. Діаграма розгортання системи

Зв'язки:

– користувачі взаємодіють з Сервером додатку інтеграції метеоданих через мережу, використовуючи веб-інтерфейс або інші додатки;

– сервер додатку інтеграції метеоданих взаємодіє з Джерелами метеоданих, надсилаючи запити для отримання метеоданих;

– сервер додатку інтеграції метеоданих взаємодіє з Сервером бази даних, де зберігаються отримані метеодані;

– джерела метеоданих надсилають метеодані до Сервера додатку інтеграції метеоданих через мережу;

– сервер бази даних забезпечує доступ до бази даних метеоданих для Сервера додатку інтеграції метеоданих.

Ця діаграма розгортання допомагає візуалізувати, як фізичні компоненти вашої системи розташовані та взаємодіють між собою на різних апаратних ресурсах.

Діаграма компонентів (*Component Diagram*) в *UML* використовується для моделювання внутрішньої структури системи та залежностей між її компонентами. Давайте опишемо діаграму компонентів для вашої системи інтеграції метеоданих на аеродромах.

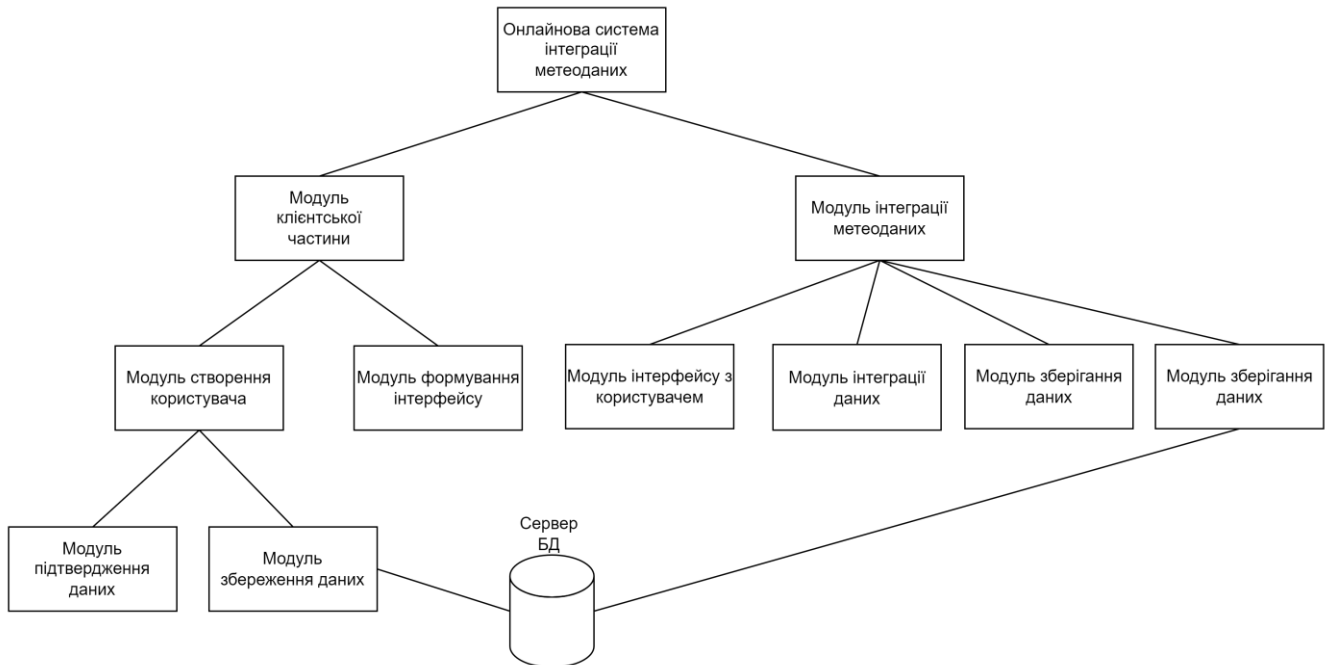


Рис. 2.9. Діаграма компонентів

Компоненти та їхні взаємозв'язки:

1. Користувачі: інтерфейси користувача для взаємодії з системою, такі як веб-інтерфейс, мобільні додатки;
2. Основні модулі онлайн-системи інтеграції метеоданих:
  - модуль інтерфейсу з користувачем: Відповідає за обробку запитів користувачів та відображення інформації на інтерфейсі користувача;
  - модуль аутентифікації та авторизації: Забезпечує аутентифікацію користувачів та керування їх доступом до системи;
  - модуль інтеграції даних: Відповідає за отримання метеоданих з джерел та їхню обробку;
  - модуль зберігання даних: відповідає за зберігання метеоданих в базі даних;
3. Джерела метеоданих: компоненти, що представляють різні джерела метеоданих, такі як метеостанції, супутникові системи, метеорадари;



4. База даних метеоданих: компонент, що представляє базу даних, де зберігаються метеодані.

Зв'язки між компонентами:

– користувачі взаємодіють з системою інтеграції метеоданих через інтерфейси користувача;

– система інтеграції метеоданих взаємодіє з Джерелами метеоданих, отримуючи та обробляючи метеодані;

– система інтеграції метеоданих взаємодіє з Базою даних метеоданих, де зберігаються отримані дані;

– система інтеграції метеоданих також включає в себе модуль для Аутентифікації та авторизації, який контролює доступ користувачів до системи.

Діаграма на рис. 2.8 допомагає краще зрозуміти структуру системи та залежності між її компонентами.

Принципи взаємодії з *API*, які надають метеоданні, можуть бути розглянуті детальніше:

1. Отримання ключа *API*: більшість сервісів, які надають метеодані через *API*, вимагають від користувача отримати ключ доступу (*API key*). Цей ключ ідентифікує користувача і надає доступ до метеорологічних даних. Ключ зазвичай слугує унікальним ідентифікатором користувача.

2. Вибір типу запиту: користувач повинен визначити, який тип запиту він хоче виконати. Це може бути запит поточних погодних умов, прогноз на кілька днів вперед, історичні дані, інформація про погодні явища тощо. Вибір типу запиту визначається параметрами *URL* запиту.

3. Параметри запиту: користувач має встановити параметри запиту, щоб вказати, які саме метеодані він хоче отримати. Наприклад, це може бути конкретний аеродром, географічні координати, діапазон дат тощо. Параметри додаються до *URL* запиту.

4. Виконання запиту: за допомогою створеного *URL* з параметрами та ключем *API* користувач відправляє запит до сервера, який надає метеодані. Зазвичай це виконується за допомогою *HTTP*-запиту (*GET* або *POST*).

5. Отримання відповіді: сервер обробляє запит та відправляє відповідь користувачу. Відповідь може бути у форматі *JSON*, *XML*, *HTML* або іншому, в залежності від *API*. Користувач отримує метеодані, які він запитував.

6. Обробка даних: отримані метеодані можуть бути оброблені на стороні користувача. Це може включати в себе розпакування даних з формату відповіді, конвертацію одиниць вимірювання, аналіз та візуалізацію даних;

7. Перевірка помилок: під час взаємодії з *API* важливо враховувати можливі помилки. Користувач повинен обробити статус-коди відповіді сервера і вжити відповідні заходи в разі помилки, наприклад, повторити запит або звернутися до служби підтримки *API*;

8. Безпека і обмеження: користувач повинен дотримуватися правил безпеки та обмежень, встановлених провайдером *API*. Це може включати в себе обмеження кількості запитів на один ключ *API*, обмеження доступу до певних функцій, використання *HTTPS* для захищеної передачі даних тощо;

9. Оновлення та підтримка: *API* може зазнавати змін та оновлень з часом. Користувач повинен бути готовий до оновлення свого коду та адаптації до нових версій *API*.

Загалом, взаємодія з *API* метеоданих вимагає правильного створення запитів, обробки відповідей і дотримання правил безпеки та обмежень, щоб забезпечити надійний та ефективний доступ до метеорологічних даних.

#### 2.4. Вибір інструментів для аналізу та обробки метеорологічних даних

Для ефективної обробки метеорологічних даних у системі інтеграції на малих аеродромах важливо визначити програмні інструменти та технології, які будуть використовуватися для обробки та аналізу цих даних. Вибір правильних інструментів може суттєво вплинути на продуктивність, точність та надійність системи. Ось декілька ключових аспектів визначення програмних інструментів для обробки метеоданих:

1. Бази даних: необхідно визначити, яка система управління базами даних (СУБД) буде використовуватися для зберігання та управління метеорологічними даними. Популярні СУБД, такі як *PostgreSQL*, *MySQL*, або *NoSQL* бази даних, можуть бути варіантами залежно від потреб системи.

2. Інструменти обробки даних: Необхідно вибрати програмні інструменти для обробки та аналізу метеорологічних даних. Мови програмування, такі як *Python*, *R* або *Java*, можуть бути корисними для розробки алгоритмів обробки та аналізу даних;

3. Візуалізація даних: необхідно розглянути інструменти для створення графіків та візуалізації метеорологічних даних. Бібліотеки для візуалізації даних, такі як *Matplotlib*, *D3.js* або *Tableau*, можуть допомогти представити дані у зрозумілому та зручному форматі.

4. Аналіз даних: необхідно розглянути інструменти для виконання аналізу метеорологічних даних. Це може включати в себе статистичні пакети, бібліотеки для машинного навчання або інші інструменти для обробки даних.

5. Інтеграція з іншими системами: необхідно врахувати можливості інтеграції обраного програмного забезпечення з іншими системами, які можуть надавати метеодані або іншу інформацію.

6. Швидкість та продуктивність: Оберіть інструменти, які дозволять досягти необхідної продуктивності та швидкості обробки даних.

7. Масштабованість: необхідно врахувати можливість масштабування обраного програмного забезпечення для обробки великих обсягів метеоданих у майбутньому.

8. Забезпечення безпеки: важливо визначити заходи безпеки для захисту метеорологічних даних від несанкціонованого доступу та втрати.

9. Підтримка та спільнота: необхідно розглянути наявність активної спільноти користувачів та підтримки для вибраних програмних інструментів.

Вибір програмних інструментів для обробки метеоданих є критичним етапом у процесі розробки систем, які використовують ці дані. Цей вибір має бути

узгодженим не тільки з технічними вимогами до системи, але й з можливостями розробки, ресурсами та довгостроковою стратегією підтримки проекту.

Перш за все, важливо враховувати обсяг метеоданих, які потрібно обробляти. Це можуть бути дані в реальному часі, історичні дані, або комбінація обох. Різні інструменти краще підходять для різних типів даних.

Методані можуть надходити у різноманітних форматах, таких як *CSV*, *XML*, *JSON* або навіть спеціалізовані метеорологічні формати. Обраний інструмент має підтримувати необхідні формати.

Для деяких застосунків, наприклад, прогнозування погоди в реальному часі, важливою є швидкість обробки даних.

Інструменти повинні бути обрані з урахуванням навичок та досвіду команди розробників. Використання складних інструментів без необхідної експертизи може значно ускладнити процес розробки. Такі інструменти мають бути сумісні з іншими системами та технологіями, які використовуються в проекті.

Важливо обрати такі інструменти, які можна буде масштабувати та адаптувати до змінюваних потреб проекту.

Оскільки методані можуть бути важливими для безпеки, інструменти мають забезпечувати належний рівень безпеки даних.

Вибір між вільним програмним забезпеченням (*open-source*) та комерційними рішеннями залежить від бюджету проекту та вимог до ліцензування. Інструменти з активною спільнотою та належною підтримкою зазвичай є надійнішими та забезпечують кращі можливості для навчання та вирішення проблем.

Вибір було зупинено на *MySQL* з наступних причин:

1. відкритий джерело: *MySQL* є відкритою СУБД з вільною ліцензією, що дозволяє безкоштовно використовувати її та мати доступ до вихідного коду. Це робить її економічно вигідним варіантом для проектів з обмеженим бюджетом;

2. Популярність і підтримка: *MySQL* є однією з найбільш популярних СУБД у світі, що означає наявність великої спільноти користувачів та розробників. Це забезпечує активну підтримку, оновлення та розвиток продукту;

3. Висока продуктивність: *MySQL* володіє високою продуктивністю та ефективністю роботи з базами даних. Вона добре справляється з обробкою великих обсягів даних та запитів;

4. Масштабованість: *MySQL* підтримує можливість горизонтального та вертикального масштабування, що дозволяє адаптувати систему до зростання обсягів методаних та користувачів;

5. Широкі можливості для розробки: *MySQL* підтримує багато мов програмування та інтерфейсів, включаючи *JDBC* для *Java*, *SQLAlchemy* для *Python*, та інші, що робить її дружньою для розробників;

6. Надійність та відновлення: *MySQL* має вбудовані засоби для забезпечення надійності та можливості відновлення в разі аварій або втрати даних;

7. Безпека: *MySQL* включає в себе можливості для налаштування доступу, аутентифікації та шифрування даних, що допомагає забезпечити безпеку методаних;

8. Сумісність з стандартами: *MySQL* відповідає багатьом стандартам *SQL*, що дозволяє використовувати існуючий *SQL*-код та інструменти.

Зважаючи на ці фактори, вибір *MySQL* може бути обґрунтованим для системи інтеграції методаних на аеродромах, забезпечуючи високу продуктивність, надійність та доступність у вільному та економічно вигідному пакеті. Однак слід також враховувати конкретні вимоги та обмеження вашого проекту, а також порівнювати *MySQL* з іншими СУБД, щоб обрати найкращий варіант для вашої конкретної задачі.

## 2.5. Висновки до розділу

В розділі надано детальний огляд проектування системи інтеграції методаних на малих аеродромах. Формування дерева функцій системи дозволило чітко визначити функціональні вимоги до системи інтеграції методаних. Це важливий крок у розробці системи, оскільки він дозволяє розподілити функції між різними компонентами та модулями.

Архітектура системи і визначення технічних вимог грають ключову роль у проектуванні. Вибір архітектурного підходу та встановлення технічних вимог допомагають створити структуру системи, яка відповідає вимогам та потребам користувачів. Вибір інструментів для аналізу та обробки метеорологічних даних визначає, як система буде працювати з отриманими даними. Це важливо для забезпечення точності та якості обробки метеоданих.

Моделювання процесів опитування та синхронізації метеоданих з різних джерел є важливою складовою системи інтеграції, що допомагло забезпечити надійність та актуальність метеорологічної інформації.

## РОЗДІЛ 3

### ПРОГРАМНА РОЗРОБКА СИСТЕМИ ІНТЕГРАЦІЇ ДАНИХ ПРО МЕТЕОУМОВИ НА МАЛИХ АЕРОДРОМАХ

#### 3.1. Опис роботи основних розроблених модулів системи

##### 3.1.1. Опис функціональності кожного модуля

Основні модулі системи інтеграції метеоданих на аеродромах можуть включати наступні компоненти:

##### 1. Модуль збору метеоданих:

– відповідає за збір метеоданих з різних джерел, таких як метеостанції, супутники, метеорадари тощо;

– включає підсистеми для збору даних в реальному часі та отримання архівних даних;

##### 2. Модуль обробки даних:

– виконує обробку отриманих метеоданих, включаючи перевірку на достовірність, валідацію, фільтрацію та конвертацію в однорідний формат.

– Може включати алгоритми попередження про погодні явища та розрахунок показників, таких як індекси теплового комфорту.

##### 3. Модуль зберігання даних:

– відповідає за зберігання оброблених метеоданих у базі даних з можливістю історичного зберігання.

– забезпечує доступ до даних для інших компонентів системи та користувачів.

##### 4. Модуль інтеграції з іншими системами:

– дозволяє обмінюватися метеоданими з іншими системами, такими як авіаційні системи управління польотами (ATM), аеропортові системи та системи попередження про погодні небезпеки.

– забезпечує використання стандартів обміну даними, таких як *IWXXM*.

##### 5. Модуль візуалізації та аналізу даних:

- надає інтерфейс користувача для відображення метеоданих на карті, графіках та інших візуальних елементах.

- дозволяє користувачам аналізувати та візуалізувати погодні інформації.

#### 6. Модуль архівування та резервного копіювання:

- забезпечує довгострокове зберігання метеоданих для архівування та аналізу.

- виконує резервне копіювання даних для забезпечення надійності та відновлення у випадку відмови системи.

#### 7. Модуль керування доступом і безпеки:

- забезпечує аутентифікацію та авторизацію користувачів для доступу до метеоданих;

- забезпечує захист даних від несанкціонованого доступу та втрати.

#### 8. Модуль сповіщення та аварійного повідомлення:

- надає можливість відправляти аварійні повідомлення та попередження про погодні небезпеки операторам та відповідним службам;

#### 9. Модуль моніторингу та управління системою:

- забезпечує моніторинг роботи всіх компонентів системи та автоматичну реакцію на виявлені проблеми;

- дозволяє адміністраторам віддалено керувати системою та налаштовувати її параметри.

Це загальний перелік можливих модулів системи інтеграції метеоданих на аеродромах. Залежно від конкретних потреб і вимог, можуть бути розроблені додаткові модулі або внесені зміни до цієї структури.

#### 3.1.2. Опис розробленої бази даних

Нижче представлено текстове описання полів, їх зміст та тип для кожної з таблиць (код створення таблиць БД представлено в Додатку А).

Таблиця "*WeatherData*" (Дані про погоду):

- *ObservationDateTime* (Дата та час спостереження): Це поле зберігає дату та час, коли були зібрані метеодані. Тип даних: *DATETIME*.

- *AirTemperature* (Температура повітря): Це поле містить інформацію про температуру повітря у градусах Цельсія. Тип даних: *FLOAT*.



– *Humidity* (вологість): Це поле зберігає відсоткове значення вологості повітря.

Тип даних: *FLOAT*.

– *AtmosphericPressure* (атмосферний тиск): Це поле містить значення атмосферного тиску у гектопаскалях (гПа). Тип даних: *FLOAT*.

– *WindSpeed* (швидкість вітру): Це поле містить швидкість вітру у метрах на секунду. Тип даних: *FLOAT*.

– *WindDirection* (напрямок вітру): Це поле зберігає напрям вітру у градусах. Тип даних: *FLOAT*.

– *PrecipitationAmount* (кількість опадів): Це поле містить кількість опадів у метрах або міліметрах, які випали протягом певного періоду. Тип даних: *FLOAT*.

– *PrecipitationType* (тип опадів): Це текстове поле містить інформацію про тип опадів (наприклад, дощ, сніг, град і т. д.). Тип даних: *VARCHAR(255)*.

– *OxygenSaturation* (насиченість киснем): Це поле зберігає значення насиченості киснем у відсотках. Тип даних: *FLOAT*.

– *CO2Level* (рівень CO<sub>2</sub>): Це поле містить рівень вуглекислого газу (CO<sub>2</sub>) у повітрі. Тип даних: *FLOAT*.

Таблиця "*DataSources*" (Джерела даних):

– *SourceID* (Ідентифікатор джерела): Це унікальний ідентифікатор для кожного джерела даних. Тип даних: *INT AUTO\_INCREMENT*.

– *SourceName* (Назва джерела): Це поле містить назву джерела даних. Тип даних: *VARCHAR(255)*.

– *SourceDescription* (Опис джерела): Це текстове поле містить докладний опис джерела даних. Тип даних: *TEXT*.

– *SourceAvailability* (Доступність джерела): Це бітове поле, яке вказує, чи доступне джерело даних в даний момент. Тип даних: *BIT*.

– *ConnectionParameters* (Параметри підключення): Це текстове поле містить параметри для підключення до джерела даних. Тип даних: *TEXT*.

Таблиця "*Airports*" (Аеродроми):

– *AirportID* (Ідентифікатор аеродрому): Це унікальний ідентифікатор для кожного аеродрому. Тип даних: *INT AUTO\_INCREMENT*.

– *AirportName* (Назва аеродрому): Це поле містить назву аеродрому. Тип даних: *VARCHAR(255)*.

– *Latitude* (Широта): Це поле містить координату широти аеродрому. Тип даних: *FLOAT*.

– *Longitude* (Довгота): Це поле містить координату довготи аеродрому. Тип даних: *FLOAT*.

– *WeatherStationInfo* (Інформація про метеостанції): Це текстове поле містить інформацію про розміщення метеостанцій на аеродромі. Тип даних: *TEXT*.

– *OtherParameters* (Інші параметри): Це текстове поле для зберігання інших параметрів аеродрому. Тип даних: *TEXT*.

Таблиця "*Users*" (Користувачі):

– *UserID* (Ідентифікатор користувача): Це унікальний ідентифікатор для кожного користувача. Тип даних: *INT AUTO\_INCREMENT*.

– *UserName* (Ім'я користувача): Це поле містить ім'я користувача. Тип даних: *VARCHAR(255)*.

– *Login* (Логін): Це поле містить логін користувача для входу до системи. Тип даних: *VARCHAR(255)*.

– *Password* (Пароль): Це поле містить захешований пароль користувача. Тип даних: *VARCHAR(255)*.

– *AccessRights* (Права доступу): Це поле містить інформацію про рівень доступу користувача (адміністратор, оператор, спостерігач і т. д.). Тип даних: *VARCHAR(50)*.

Таблиця "*EventLog*" (Журнал подій):

– *EventID* (Ідентифікатор події): Це унікальний ідентифікатор для кожної події в журналі. Тип даних: *INT AUTO\_INCREMENT*.

– *EventDateTime* (Дата та час події): Це поле містить дату та час, коли сталася подія. Тип даних: *DATETIME*.

– *EventDescription* (Опис події): Це текстове поле містить опис події або подробиці щодо її виникнення. Тип даних: *TEXT*.

Таблиця "*SystemSettings*" (Налаштування системи):

– *SettingID* (Ідентифікатор налаштування): Це унікальний ідентифікатор для кожного налаштування системи. Тип даних: *INT AUTO\_INCREMENT*.

– *SettingName* (Назва налаштування): Це поле містить назву налаштування. Тип даних: *VARCHAR(255)*.

– *SettingValue* (Значення налаштування): Це поле містить значення налаштування. Тип даних: *VARCHAR(255)*.

Принципи роботи з даними у створеній базі даних можна описати так:

#### 1. Зберігання метеоданих:

– Для зберігання метеоданих використовується таблиця "*WeatherData*". Кожен запис в цій таблиці включає в себе інформацію про різні параметри погоди (температура, вологість, тиск тощо), а також дату та час спостереження.

– Для забезпечення унікальності записів використовується поле "*ObservationDateTime*" як первинний ключ (*PRIMARY KEY*), що гарантує, що в базі даних не буде дублікатів записів за однаковий час спостереження.

#### 2. Зберігання інформації про джерела даних:

– Інформація про джерела даних зберігається в таблиці "*DataSources*". Кожен запис цієї таблиці має ідентифікатор (*SourceID*), назву джерела, опис, інформацію про доступність та параметри підключення.

– Індекс "*SourceID*" є первинним ключем для забезпечення унікальності записів про джерела даних.

#### 3. Зберігання інформації про аеродроми:

– Інформація про аеродроми розміщується в таблиці "*Airports*". Кожен аеродром має свій унікальний ідентифікатор (*AirportID*), назву, координати (широта та довгота), інформацію про метеостанції та інші параметри.

– Поле "*AirportID*" виступає в ролі первинного ключа, що дозволяє ідентифікувати аеродроми.

#### 4. Зберігання інформації про користувачів:

– Інформація про користувачів системи знаходиться в таблиці "*Users*". Кожен користувач має свій унікальний ідентифікатор (*UserID*), ім'я, логін, захешований пароль, права доступу та інші дані.

– Поле "*UserID*" виступає в якості первинного ключа, який ідентифікує користувачів.

#### 5. Зберігання журналу подій:

– Інформація про події системи, такі як помилки, відстеження подій тощо, записується в таблиці "*EventLog*". Кожна подія має свій унікальний ідентифікатор (*EventID*), дату та час події, опис події.

– Поле "*EventID*" є первинним ключем для забезпечення унікальності записів в журналі подій.

Ці принципи дозволяють ефективно зберігати та організовувати дані у базі даних, а також забезпечують цілісність та безпеку інформації. Крім того, вони дають можливість проводити аналіз, пошук та звітність з метеоданих та інших аспектів системи інтеграції метеоданих на аеродромах.

#### 3.1.3. Модуль опитування *API* метеосистем

Користування *API* для оновлення метеоданих важливою частиною системи інтеграції метеоданих на аеродромах. Ось деякі можливі особливості роботи з вказаними *API*:

##### 1. *API OpenWeatherMap* (<https://api.openweathermap.org>):

– зручний доступ до погодних даних: *OpenWeatherMap* надає широкий доступ до різних типів погодних даних, включаючи поточну погоду, прогнози, історичні дані та інше;

– доступність безкоштовного тарифу: Є безкоштовний тариф, який дозволяє отримувати обмежену кількість запитів на місяць, що може бути важливим для малих аеродромів з обмеженими ресурсами;

##### 2. *API Weatherstack* (<http://api.weatherstack.com>):

– простий у використанні: *Weatherstack* надає простий та зрозумілий *API* для отримання погодних даних;

– зручність інтеграції: *API* пропонує план спільно з *OpenWeatherMap*, який дозволяє здійснювати безкоштовні запити, але з обмеженнями;

##### 3. *API AccuWeather* (<https://dataservice.accuweather.com>):

– висока точність: *AccuWeather* славиться своєю високою точністю прогнозів, особливо для спеціалізованих сегментів, таких як авіація;

– платний сервіс: зазвичай *AccuWeather* вимагає плати за користування своїм *API*, що може бути важливим фактором при плануванні бюджету.

Основні особливості роботи з усіма цими *API* включають обробку відповідей *API* (зазвичай у форматі *JSON*), обробку помилок та приватних ключів *API* для автентифікації. Також враховувано обмеження щодо кількості запитів, які можна здійснити протягом певного періоду часу (рис. 3.1).

```
▼ coord:
  lon:      30.8923
  lat:      50.3402
▼ weather:
  ▼ 0:
    id:      804
    main:    "Clouds"
    description: "overcast clouds"
    icon:    "04d"
  base:     "stations"
▼ main:
  temp:     274.06
  feels_like: 269.28
  temp_min: 271.88
  temp_max: 274.06
  pressure: 1021
  humidity: 92
  sea_level: 1021
  grnd_level: 1005
  visibility: 10000
▼ wind:
  speed:    5.17
  deg:      308
  gust:     10.72
▼ clouds:
  all:      85
  dt:       1702632015
▼ sys:
  type:     2
  id:       2082717
  country:  "UA"
  sunrise:  1702619380
  sunset:   1702648389
  timezone: 7200
  id:       711660
  name:     "Boryspil"
  cod:     200
```

Рис. 3.1. Формат *JSON* метеоданих

Для опитування трьох різних сервісів і обирання середніх даних між ними потрібно виконати запити до кожного з цих сервісів, отримати дані і обчислити середнє значення. Код мовою *PHP* наведено в Додатку А.

Для визначення, що один із сервісів випадає з загальної тенденції, можна встановити певний поріг або максимальну відхиленість, яку ви готові прийняти для середнього значення. Якщо різниця між значеннями одного з сервісів і середнім значенням перевищує цей поріг, можна вважати, що цей сервіс випадає з загальної тенденції.

Використання відсотків для визначення *AVG\_Err* може бути зручним підходом, особливо коли маємо справу з різними типами даних і сервісів з різною точністю. Відсотковий показник може забезпечити більшу гнучкість і дозволити легше адаптувати точність до різних умов (рис. 3.2).

Для *AVG\_Err* використовувати відсотки з наступних причин:

1. Визначення відсотку точності: Необхідно розглянути, наскільки точні дані ви хочете отримувати в середньому. В системи встановлено *AVG\_Err* на рівні 5%.

2. Порівняння з реальними значеннями: При отриманні даних їх можна порівняти з іншими сервісами та джерелами. Якщо відхилення вище визначеного відсотку, то можна вважати це неприйнятним і реагувати відповідним чином.

3. Коригування даних: Можна використовувати *AVG\_Err* для автоматичного коригування даних, якщо вони виходять за межі визначеного відсотку. Наприклад, можна видаляти або фільтрувати дані, які не відповідають вашим вимогам до точності.

4. Динамічне налаштування: Використання відсотків дозволяє вам динамічно налаштовувати точність в залежності від умов. Наприклад, можна зменшити *AVG\_Err* в умовах, коли точність особливо важлива (наприклад, в навігації під час польоту).

Відсотковий підхід може бути корисним для різних типів додатків і систем, але важливо зрозуміти, що він може бути менш інтуїтивним, оскільки точність вимірюється у відсотках. Також важливо слідкувати за величиною *AVG\_Err*, оскільки великий відсоток може призвести до значних відхилень в даних.

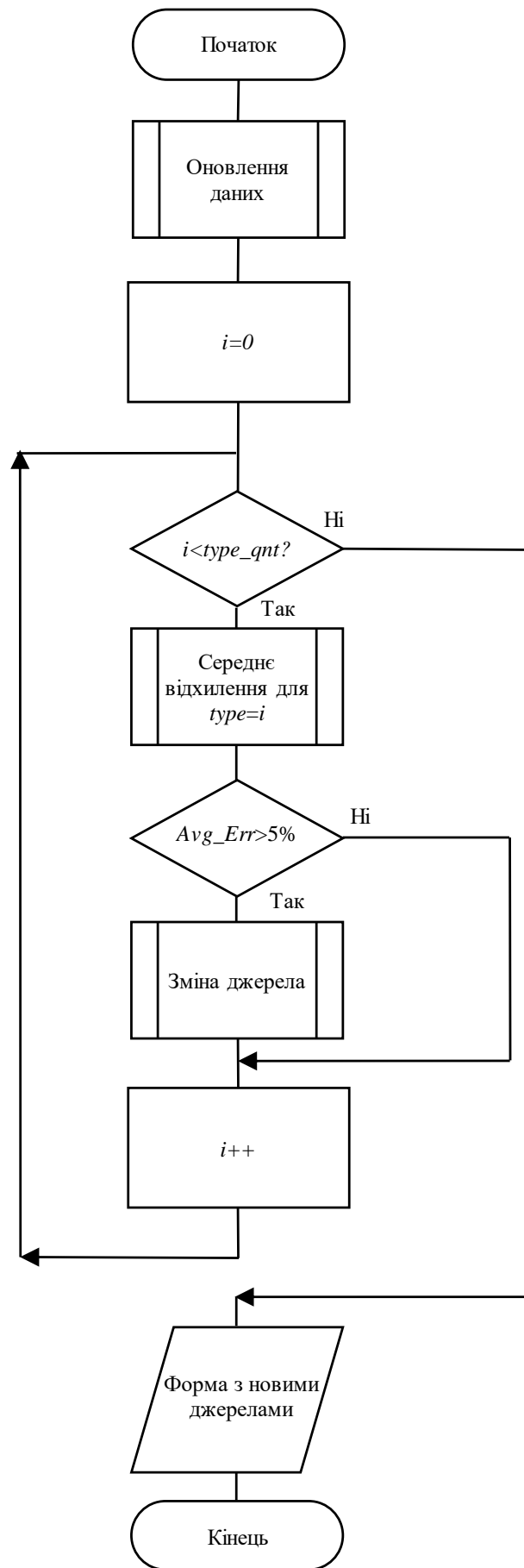


Рис. 3.2. Схема алгоритму зміни джерел метеоданих

## 3.2. Розгортання системи на зовнішніх серверах

Процес розгортання системи на зовнішніх серверах включає наступні кроки:

### 1. Передача файлів:

– Використання *FTP*, *SFTP*, *SCP* чи інших протоколів для передачі файлів веб-додатку з локальної розробки на віддалений хостинг або сервер.

– Можливо, використання систем управління версіями, як-от *Git*, для клонування репозиторію на сервер.

### 2. Конфігурація сервера:

– Налаштування веб-сервера (наприклад, *Apache*, *Nginx*) та інших сервісів, які потрібні для роботи додатку (наприклад, база даних *MySQL*, сервер *Redis* тощо).

– Налаштування файлів конфігурації, включно з файлами *.htaccess*, *php.ini*, та іншими конфігураційними файлами.

### 3. Безпека:

– Встановлення *SSL*-сертифікатів для забезпечення захищеного з'єднання (*HTTPS*).

– Налаштування фаєрволів та інших механізмів безпеки для захисту від зовнішніх атак.

### 4. База даних:

– Імпорт бази даних з локальної системи на віддалений сервер або підключення до зовнішньої бази даних.

– Налаштування резервного копіювання та відновлення бази даних.

5. Залежності: встановлення необхідних бібліотек та залежностей через менеджери пакетів, як-от *Composer* для *PHP*.

6. Тестування та верифікація: перевірка функціональності та виконання на сервері, забезпечення того, що усі компоненти встановлені правильно та додаток працює належним чином.

7. Моніторинг та логування: налаштування систем моніторингу та логування для відстеження стану сервера та застосунку.



8. Оновлення та обслуговування: планування та виконання регулярних оновлень безпеки та обслуговування системи.

### 3.3. Приклад роботи розробленої системи

Вікна входу в систему є одним з ключових елементів будь-якої програмної системи, включаючи систему інтеграції метеоданих на малих аеродромах. Це важливий інтерфейс, який надає користувачам можливість аутентифікації і отримання доступу до функціональності системи. У наступних розділах ми розглянемо докладніше основні компоненти та особливості вікна входу в систему.

На рис. 3.3 надано можливість для розгалуження процесів авторизація та реєстрації.

Система відслідковування  
метеоданих аеродромів

ЛОГІН

РЕЄСТРАЦІЯ



Рис. 3.2. Вікно входу в систему

Основні компоненти вікна входу в систему (рис. 3.4):

1. Поле введення логіна: це текстове поле призначене для введення ідентифікатора користувача (логіна). Він може бути унікальним іменем користувача або електронною поштою, залежно від налаштувань системи;

2. Поле введення паролю: це текстове поле призначене для введення паролю, який відповідає введеному логіну. Пароль повинен бути обов'язково захешованим для забезпечення безпеки користувачів;

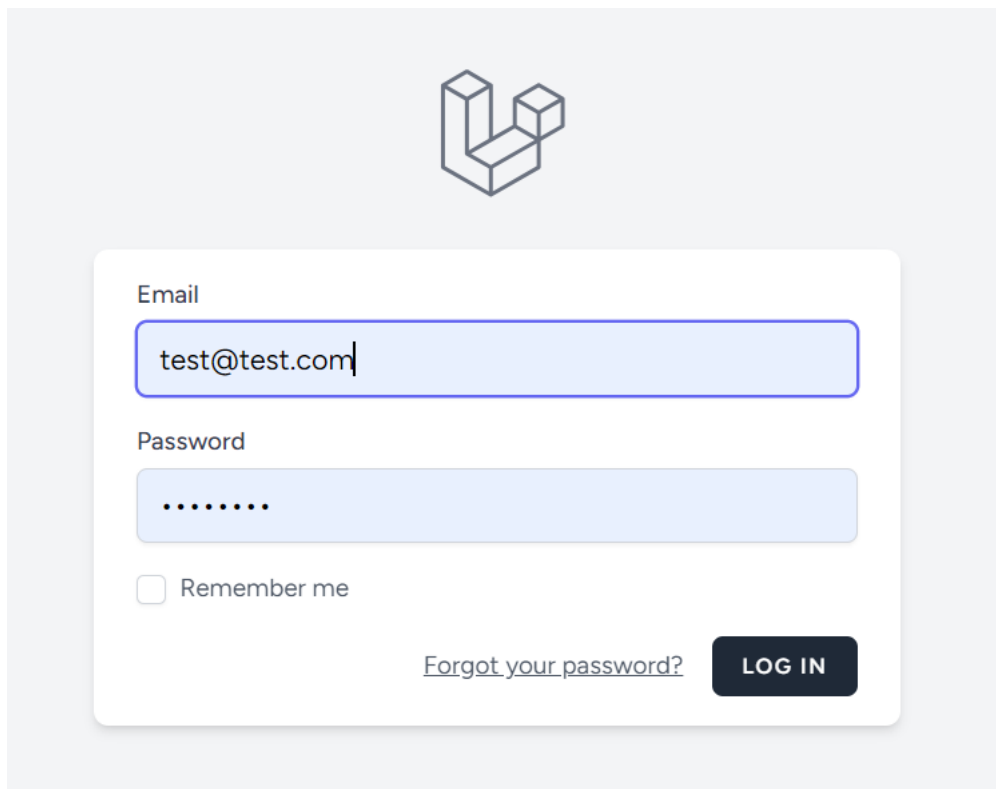


Рис. 3.3. Вікно авторизації

3. Кнопка "Увійти": ця кнопка запускає процес аутентифікації користувача на основі введеного логіна та паролю. Після натискання на кнопку, система перевіряє введені дані та надає доступ користувачу, якщо дані вірні;

4. Посилання на відновлення паролю: деякі системи надають можливість користувачам відновити забуті паролі. Це посилання переходить на відповідну сторінку для встановлення нового паролю;

5. Посилання на реєстрацію: якщо користувач ще не має облікового запису в системі, то він може перейти на сторінку реєстрації для створення нового облікового запису.

Особливості вікна входу в систему:

1. безпека: вікно входу повинно забезпечувати захист від несанкціонованого доступу, включаючи захист від перебору паролів (повторних спроб вводу паролю), обов'язкове шифрування паролів та використання механізмів *CAPTCHA* для визначення, що користувач - це людина, а не автоматизована програма;

2. Доступність: важливо, щоб вікно входу було легко доступним для користувачів, інтуїтивно зрозумілим і сумісним із засобами адаптивного дизайну для різних типів пристроїв;

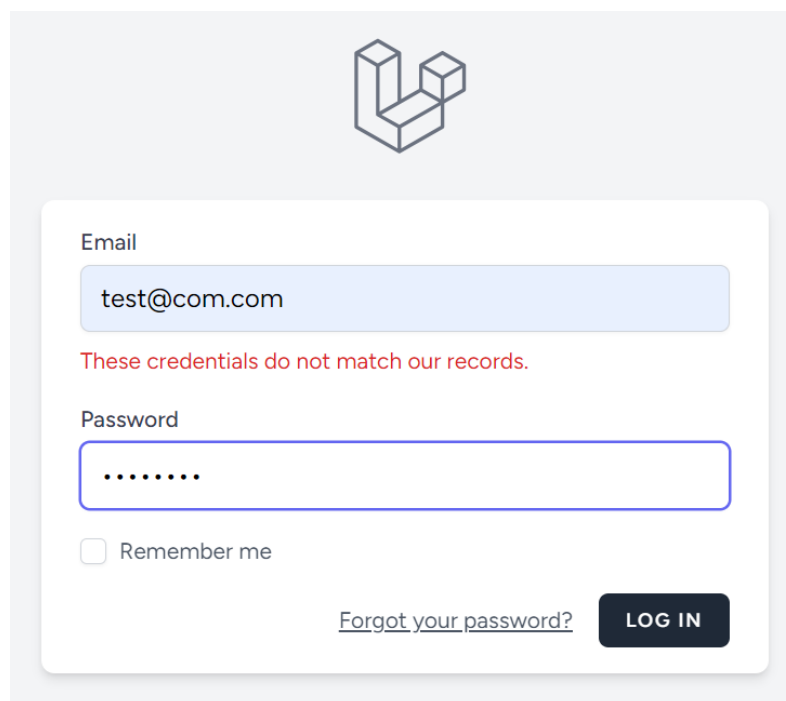
3. Локалізація: якщо система використовується в різних країнах або регіонах, важливо мати можливість змінювати мову та інші параметри вікна входу відповідно до потреб користувачів;

4. Логування та аудит: усі спроби входу в систему повинні реєструватися в журналі подій для подальшого аудиту та виявлення потенційних загроз безпеці;

5. Забезпечення зручності: інтерфейс вікна входу повинен бути максимально зручним та інтуїтивно зрозумілим для користувачів, забезпечуючи їм зручний та швидкий доступ до системи.

В цілому, вікно входу в систему - це важливий елемент, який допомагає забезпечити безпеку та зручність використання системи інтеграції метеоданих на малих аеродромах. Його розробка та налагодження вимагає уваги до деталей та відповідності стандартам безпеки і інтерфейсу.

Реакція системи на помилковий код або незареєстрований електронний адрес може відрізнитися в залежності від політики безпеки та дизайну системи (рис. 3.5).



The image shows a login interface with a logo at the top consisting of three stacked rectangular blocks. Below the logo is a white login form with a light gray border. The form contains the following elements: an 'Email' label above a text input field containing 'test@com.com'; a red error message 'These credentials do not match our records.'; a 'Password' label above a password input field with seven dots; a 'Remember me' checkbox; a 'Forgot your password?' link; and a dark blue 'LOG IN' button.

Рис. 3.5. Вікно з відмовою у вході

Нижче наведено варіанти реакції системи:

1. Відмова в доступі: система може повернути помилку і відмовити в доступі, якщо введений код або електронний адрес не відповідають жодному зареєстрованому обліковому запису. Це є найбільш безпечним варіантом, оскільки не розкриває надто багато інформації;

2. Повідомлення про помилку: система може повідомити користувача про те, що введений код або електронний адрес неправильний. Повідомлення може бути більш або менш докладним, вказуючи, чи помилковий код, чи неправильний адресу електронної пошти;

3. Запит на відновлення паролю: якщо користувач забув свій код або електронну адресу, система може надати можливість відновити пароль або введену інформацію. Це може включати в себе відправку інструкцій на зареєстровану електронну пошту для скидання паролю або знайдення коду;

4. Захист від перебору: система включає механізми захисту від перебору паролів, які запобігають багаторазовим спробам введення коду або адреси. Після певної кількості невдалих спроб, система може блокувати доступ на певний період або вимагати додаткової аутентифікації;

5. Додаткова аутентифікація: в разі сумнівів у валідності коду або адреси, система може вимагати додаткової аутентифікації, такої як введення одноразового коду, використання біометричних даних або інших методів підтвердження.

Реакція системи на помилковий код або незареєстрований електронний адрес повинна бути добре продуманою і забезпечувати безпеку облікових записів користувачів, а також зручність використання для самих користувачів.

За кожним користувачем може бути закріплено декілька аеродромів (рис. 3.6).

Елементи вікна:

1. Користувачі: відображаються зі своїми іменами або ідентифікаторами, і користувач може вибрати конкретного користувача зі списку;

2. Список аеродромів: таблиця, яка містить назви аеродромів або їх ідентифікатори. Пошук аеродромів може бути відфільтрованим за регіоном, країною або іншими параметрами, що допомагає користувачу знайти потрібний аеродром;



## Аеродроми

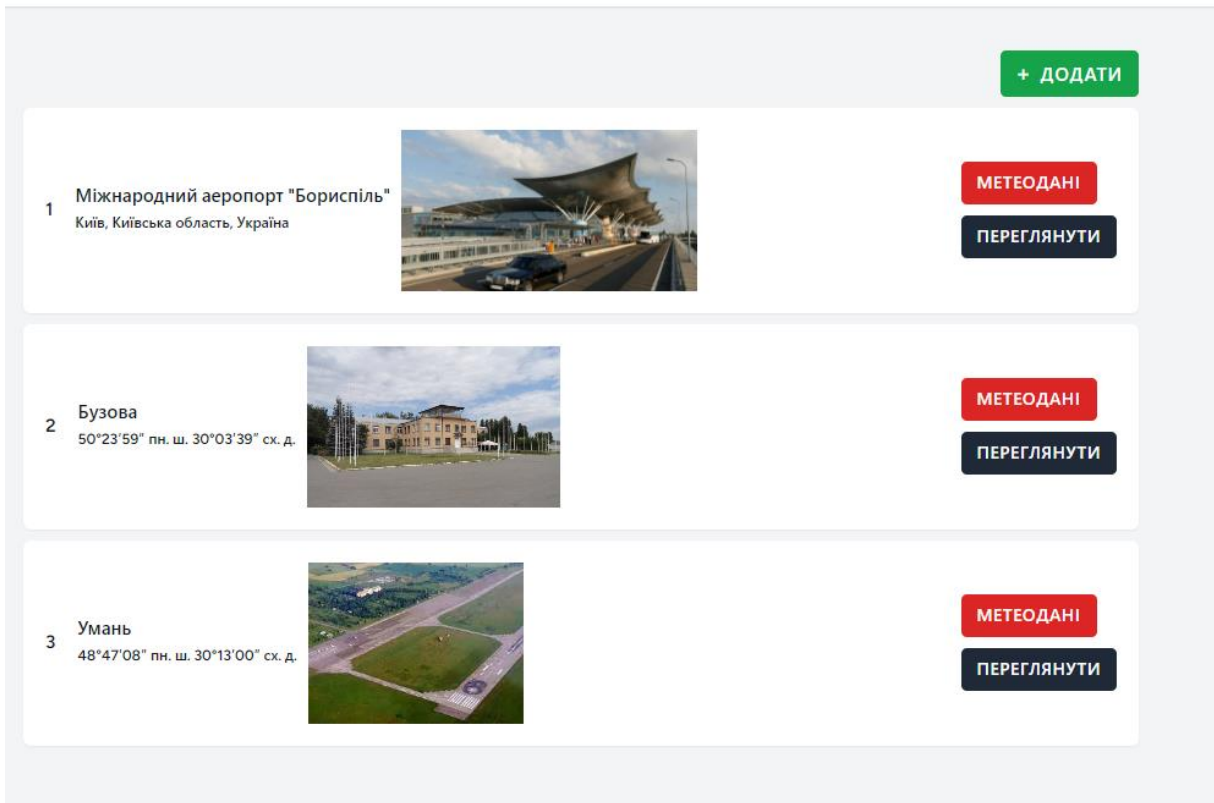


Рис. 3.6. Вікно відображення переліку аеродромів

3. Додати аеродром: це кнопка або елемент інтерфейсу, за допомогою якої користувач може додати новий аеродром до списку закріплених аеродромів для обраного користувача;

4. Видалити аеродром: це кнопка або елемент інтерфейсу, за допомогою якої користувач може видалити обраний аеродром зі списку закріплених аеродромів для обраного користувача.

5. Зберегти зміни: це кнопка або елемент інтерфейсу, яка дозволяє користувачу зберегти зміни у списку закріплених аеродромів.

6. Скасувати: це кнопка або елемент інтерфейсу, яка дозволяє користувачу скасувати зроблені зміни і повернутися до попереднього стану списку закріплених аеродромів.

Робочий процес:

1. Користувач обирає ідентифікатор або ім'я зі списку користувачів.

2. Після вибору користувача відображається список закріплених аеродромів для цього користувача.

3. Користувач може вибрати аеродроми зі списку або додати новий аеродром за допомогою кнопки "Додати аеродром".

4. Користувач може також видалити аеродроми зі списку, якщо вони більше не потрібні, за допомогою кнопки "Видалити аеродром".

5. Після внесення всіх змін користувач натискає кнопку "Зберегти зміни" для збереження оновленого списку закріплених аеродромів.

6. Якщо користувач бажає відмінити зміни, він може натиснути кнопку "Скасувати".

Це вікно дозволяє адміністраторам або користувачам керувати списком закріплених аеродромів для конкретних користувачів, спрощуючи процес налаштування доступу до метеоданих для кожного користувача окремо.

Кожен аеродром має описання, яке вводиться при створенні (рис. 3.7), а також визначаються метеоданні для представлення.

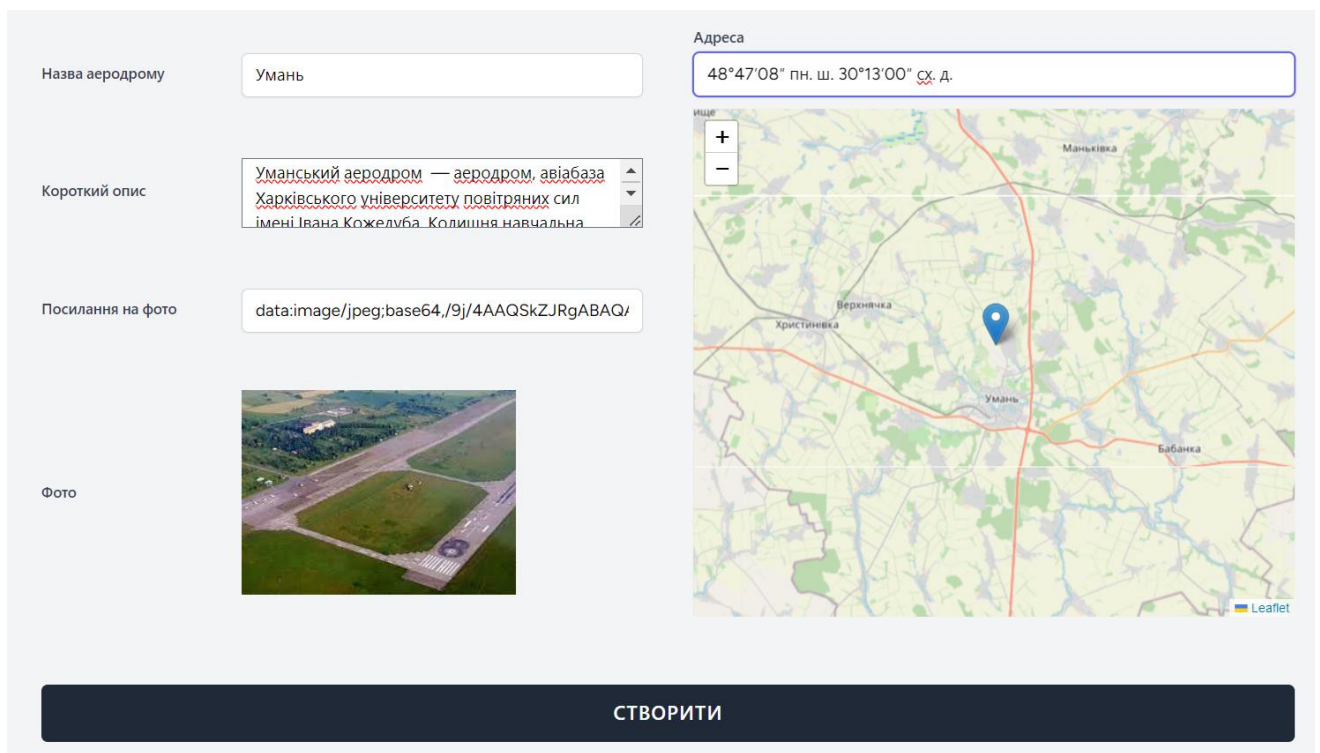


Рис. 3.7. Вікно створення опису нового аеродрому

Вікно створення аеродрому (рис. 3.7) дозволяє користувачам створювати нові аеродроми у системі інтеграції метеоданих. Користувачі повинні ввести наступну інформацію:

- назва аеродрому: ім'я аеродрому або його ідентифікатор.
- географічні координати: широта та довгота, що визначають точне розташування аеродрому на мапі.
- інформація про метеостанції: Опис метеостанції на аеродрому, включаючи її тип, особливості, обладнання та інше.
- інші параметри: додаткова інформація про аеродром, така як висота над рівнем моря, кліматичні умови, тип покриття злітно-посадкових смуг, наявність навігаційного обладнання і т.д.

Користувачі можуть також завантажити фотографії або схеми аеродрому для подальшого візуального представлення.

Вікно відображення метеоданих (рис. 3.8) представляє перелік значень метеоданих, які будуть представлені для кожного аеродрому.

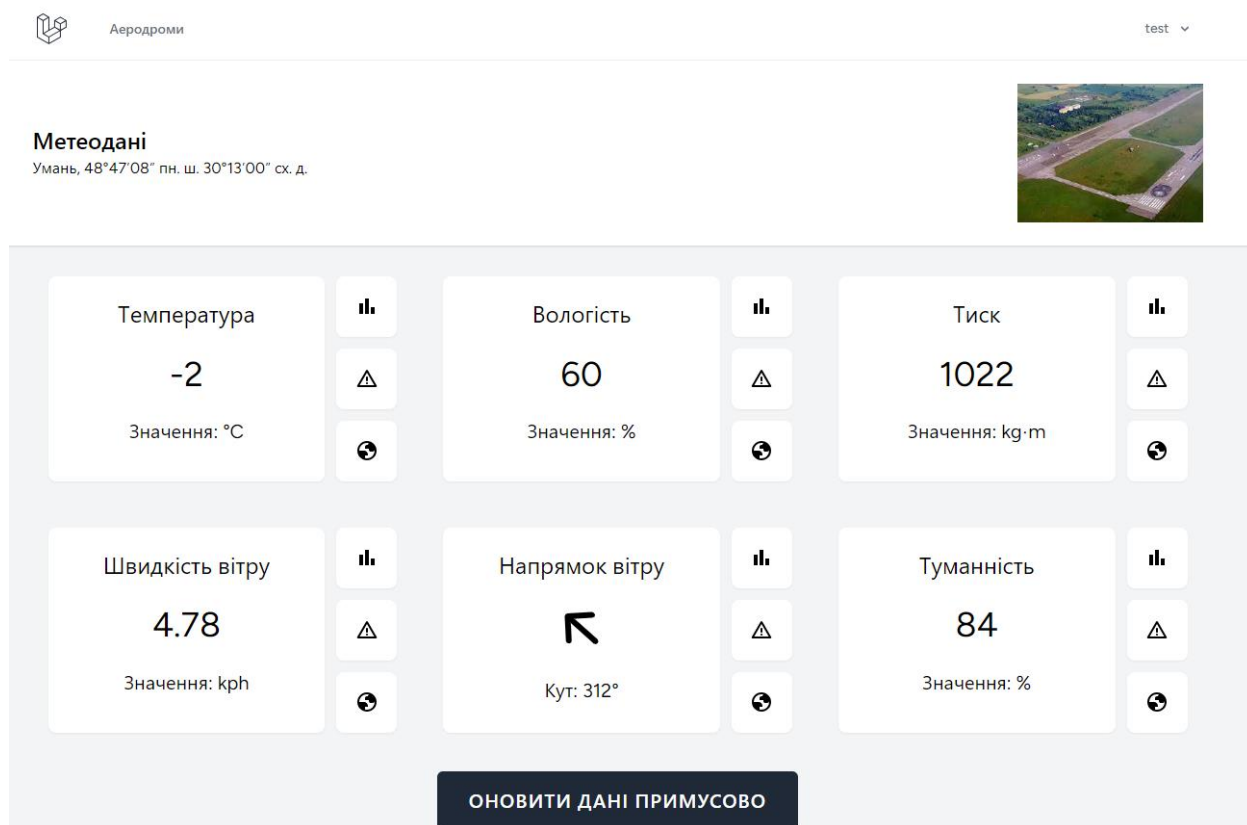


Рис. 3.8. Поточні метеодані для аеродрому

Для кожного аеродрому користувачі можуть встановити наступні параметри метеоданих:

– температура повітря: визначення параметрів та одиниць вимірювання температури повітря (градуси Цельсія, Фаренгейта і т. д.).

– вологість: встановлення одиниць вимірювання вологості повітря (відсотки, грами на кубічний метр і т. д.).

– атмосферний тиск: визначення одиниць вимірювання атмосферного тиску (гектопаскалі, міліметри ртутного стовпа і т. д.).

– швидкість вітру: встановлення одиниць вимірювання швидкості вітру (метри на секунду, кілометри на годину і т. д.).

– напрям вітру: встановлення одиниць вимірювання напрямку вітру (градуси, компасні напрямки і т. д.).

– кількість опадів: Визначення одиниць вимірювання кількості опадів (метри, міліметри, дюйми і т. д.).

– тип опадів: Текстове поле для вибору або введення типу опадів (наприклад, дощ, сніг, град і т. д.).

– насиченість киснем: встановлення одиниць вимірювання насиченості киснем (відсотки, міліграми на літр і т. д.).

Ці налаштування дозволяють визначити формат і одиниці вимірювання метеоданих для кожного аеродрому, щоб система могла коректно отримувати, зберігати та відображати ці дані.

Робота модуля графічного представлення історії значень (рис. 3.9) призначений для візуалізації історії зміни значень метеоданих у часі.

На графіку можна побачити такі компоненти:

– Графік зі зміною напрямку вітру з часом: цей графік відображає, як змінюється напрямок вітру протягом певного періоду часу. Зазвичай він представлений у вигляді кривої лінії або графіку з відзначеннями точок.

– Осі координат: ось  $X$  відповідає часу, а ось  $Y$  - значенню напрямку вітру. Ці осі допомагають інтерпретувати графік і визначити, як змінюється параметр з часом.



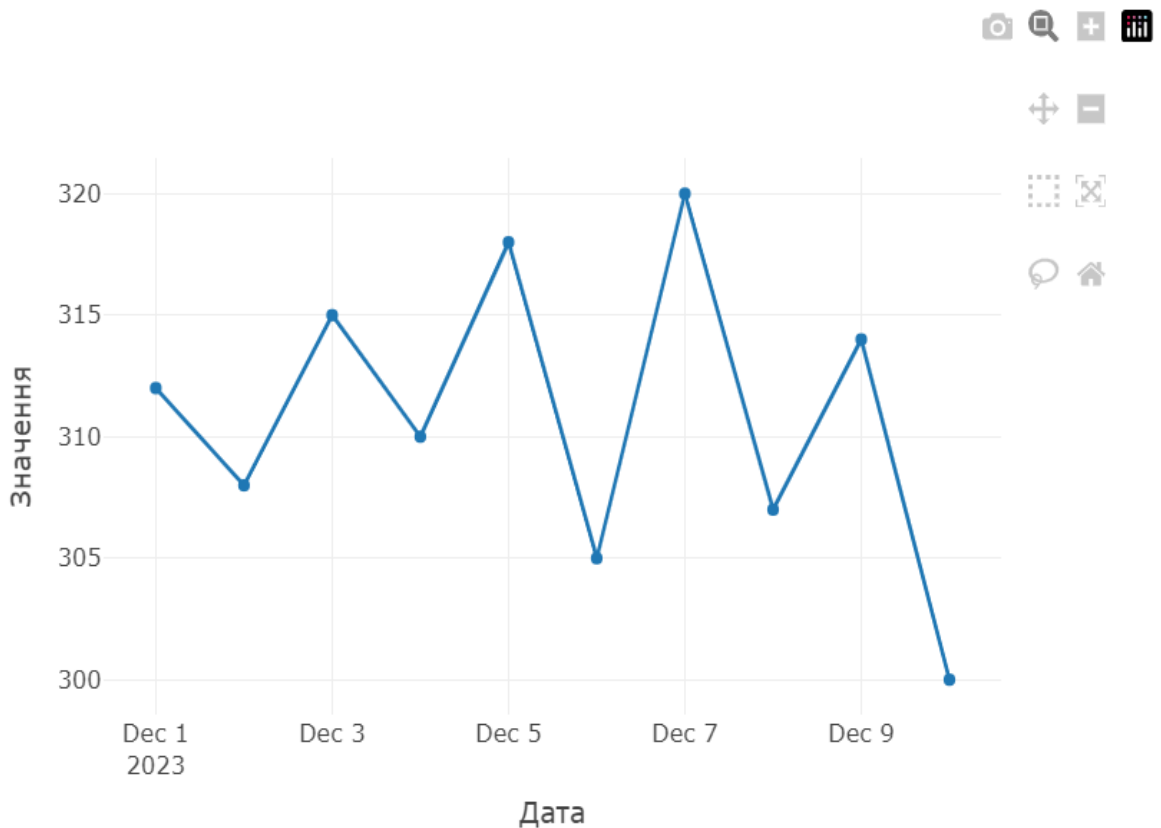


Рис. 3.9. Робота модуля графічного представлення історії значень (на прикладі напрямку вітру)

– легенда або підписи: легенда може вказувати, що саме відображено на графіку і які одиниці вимірювання використовуються. Наприклад, вона може підписувати ось Y як "Напрямок вітру (градуси)".

Цей модуль дозволяє операторам і аналітикам відстежувати та аналізувати зміни в певних метеорологічних параметрах, що допомагає приймати рішення в авіаційній галузі на основі історичних даних.

Згідно схеми алгоритму на рис. 3.1 у разі виходу значення обраного параметру метеоданих за середнє значення зовнішніх даних видається знак оклику (рис. 3.10).

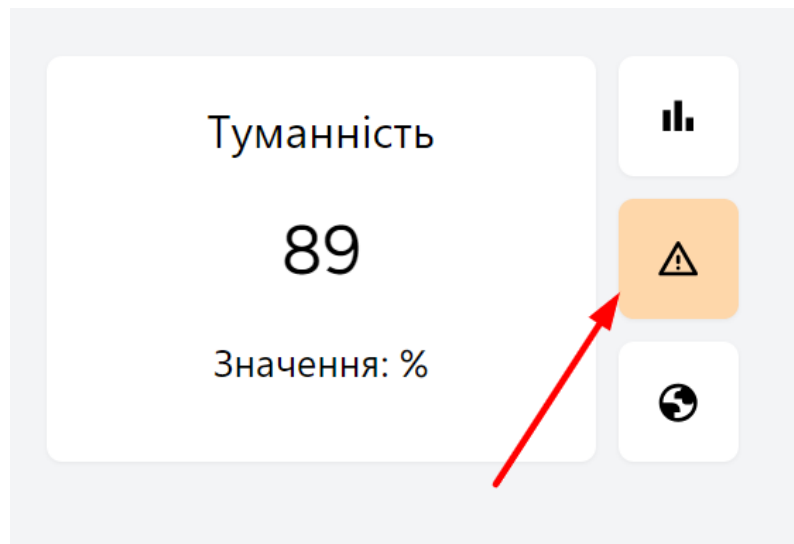


Рис. 3.10. Знак оклику, як попередження про вихід значень джерела за припустимий рівень відхилення

Цей символ може з'явитися біля конкретного значення метеоданих або поруч із повідомленням про статус цього джерела даних. Він вказує на те, що виникла ситуація, коли отримані дані не відповідають очікуваним стандартам або вимогам.

Наприклад, якщо джерело даних надсилає значення температури, яка суттєво вища або нижча за припустимі межі безпеки для авіації, то з'явиться знак оклику, щоб вказати на цю проблему. Користувачі системи можуть вжити відповідних заходів у відповідь на цей сигнал, наприклад, перевірити дані та виправити їх або призупинити використання цього джерела даних до усунення неполадок.

Знак оклику є важливим інструментом для підвищення безпеки та надійності системи інтеграції метеоданих на аеродромах, допомагаючи операторам вчасно реагувати на невідповідності в даних та виконувати вимоги безпеки авіації.

#### 3.4. Висновки до розділу

В розділі представлено процес програмної розробки системи інтеграції даних про метеоумови на малих аеродромах.

Опис роботи основних розроблених модулів системи дозволив краще розуміти функціональність кожного модуля. Це важливо для налагодження та оптимізації

роботи системи. Розроблена база даних є основою для зберігання та управління метеорологічними даними. Вона була створена з урахуванням потреб системи та може забезпечити ефективний доступ до інформації.

Розроблений модуль опитування *API* метеосистем відіграє важливу роль у зборі та оновленні метеоданих та дозволяє отримувати актуальну інформацію з різних джерел та забезпечує надійну інтеграцію даних.

Розгортання системи на зовнішніх серверах робить її доступною та доступною для користувачів. Це дозволяє використовувати систему з будь-якого місця та пристрою з Інтернет-підключенням.

Приклад роботи розробленої системи демонструє її функціональність та можливості. Він ілюструє, як система може бути використана для отримання та обробки метеоданих на малих аеродромах.

## ВИСНОВКИ

Дослідження, розглянуте у даній науковій роботі, спрямоване на створення та впровадження системи інтеграції метеоданих на малих аеродромах. Даний проект є важливим в контексті авіаційної безпеки та ефективності польотів, оскільки точні та актуальні метеорологічні дані є критично важливими для безпечної авіації.

В роботі надано важливі відомості про роль метеоданих у контексті авіаційної безпеки та методи їх інтеграції.

Визначено, що метеодані відіграють важливу роль у підвищенні безпеки та ефективності авіаційних польотів. Правильна інтерпретація метеоданих дозволяє уникати небезпечних ситуацій та забезпечує надійність авіаційних операцій. Огляд існуючих методів інтеграції метеоданих свідчить про різноманітність підходів до об'єднання метеорологічної інформації з різних джерел. Інтеграція метеоданих вимагає розробки систем, які забезпечують доступність і точність даних.

Аналіз існуючих стандартів і систем інтеграції метеоданих підкреслює необхідність стандартизації та уніфікації процесу обміну метеорологічною інформацією для забезпечення сумісності та зручності взаємодії між системами.

Окремо надано детальний огляд проектування системи інтеграції метеоданих на малих аеродромах. Формування дерева функцій системи дозволило чітко визначити функціональні вимоги до системи інтеграції метеоданих. Це важливий крок у розробці системи, оскільки він дозволяє розподілити функції між різними компонентами та модулями.

Архітектура системи і визначення технічних вимог грають ключову роль у проектуванні. Вибір архітектурного підходу та встановлення технічних вимог допомагають створити структуру системи, яка відповідає вимогам та потребам користувачів. Вибір інструментів для аналізу та обробки метеорологічних даних визначає, як система буде працювати з отриманими даними. Це важливо для забезпечення точності та якості обробки метеоданих.

Моделювання процесів опитування та синхронізації метеоданих з різних джерел є важливою складовою системи інтеграції, що допомагло забезпечити надійність та актуальність метеорологічної інформації.

В практичній частині роботи представлено процес програмної розробки системи інтеграції даних про метеоумови на малих аеродромах.

Опис роботи основних розроблених модулів системи дозволив краще розуміти функціональність кожного модуля. Це важливо для налагодження та оптимізації роботи системи. Розроблена база даних є основою для зберігання та управління метеорологічними даними. Вона була створена з урахуванням потреб системи та може забезпечити ефективний доступ до інформації.

Розроблений модуль опитування *API* метеосистем відіграє важливу роль у зборі та оновленні метеоданих та дозволяє отримувати актуальну інформацію з різних джерел та забезпечує надійну інтеграцію даних.

Розгортання системи на зовнішніх серверах робить її доступною та доступною для користувачів. Це дозволяє використовувати систему з будь-якого місця та пристрою з Інтернет-підключенням.

Приклад роботи розробленої системи демонструє її функціональність та можливості. Він ілюструє, як система може бути використана для отримання та обробки метеоданих на малих аеродромах.

Головними висновками даної роботи є:

– роль метеоданих у підвищенні авіаційної безпеки та ефективності польотів визначена як ключова. Наявність точних та актуальних метеоданих дозволяє пілотам та диспетчерам приймати обгрунтовані рішення під час небезпечних метеорологічних умов.

– досліджено існуючі методи інтеграції метеоданих та виявлені їх переваги та обмеження. Системи, які використовуються в авіаційній галузі, включають *AMOSS*, *IWXXM* та інші. Кожен з цих методів має свої особливості та вимоги.

– проектування системи інтеграції метеоданих на малих аеродромах включає в себе визначення функціональних вимог до системи, вибір архітектурного підходу,

розробку бази даних, вибір інструментів для обробки метеорологічних даних та розгортання системи на зовнішніх серверах.

Основною складовою системи є модуль опитування *API* метеосистем, який дозволяє отримувати та оновлювати метеодані з різних джерел. Цей модуль забезпечує надійну інтеграцію даних та забезпечує їх актуальність.

Приклад роботи системи інтеграції метеоданих на малих аеродромах демонструє її функціональність та можливості в реальному середовищі. Система може бути використана для забезпечення безпеки та ефективності авіаційних операцій на малих аеродромах.

В ході виконання роботи:

1) проаналізовано:

– роль метеоданих у забезпеченні безпеки та ефективності авіаційних польотів;

– існуючі методи інтеграції метеоданих на аеродромах;

– проблеми, пов'язані з збором, синхронізацією та доступом до метеорологічних даних на малих аеродромах.

2) розроблено:

– інтегровану систему, яка полегшує інтеграцію різних джерел метеоданих на малих аеродромах;

– базу даних та таблиці для зберігання метеорологічних даних, інформації про джерела даних, аеродроми, користувачів, журнал подій та налаштування системи;

– програмну реалізацію методів синхронізації даних між різними джерелами;

– інтерфейс користувача для зручного доступу та управління системою.

Робота висвітлює важливість інтеграції метеоданих у авіаційній галузі та надає практичні рекомендації щодо розробки та впровадження системи інтеграції метеоданих на малих аеродромах.

## СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бойченко С.В., Іванченко О.В. Положення про дипломні роботи (проекти) випускників Національного авіаційного університету. – К.: НАУ, 2017. – 63 с.
2. ДСТУ 3008-95. Документація. Звіти у сфері науки і техніки. Структура і правила оформлення. – 39 с.
3. Галік О.І. Метеорологічні прилади і методи спостережень. Практикум: Навчальний посібник. – Рівне: НУВГП, 2008. – 134 с.
4. Воронов Г.С., Паламарчук Л.В. Основи метеорології. ч. II. ВПУ КНУ, 2004. – 143 с.
5. Івус Г.П., Семергей-Чумаченко А.Б. Авіаційна метеорологія: Конспект лекцій – Дніпропетровськ: ПБП «Економіка», 2006. – 140 с.
6. Глушков А.В. Технічні засоби гідрометеорологічної служби: конспект лекцій. Одеса: Одеський державний екологічний університет, 2018. 136 с.
7. *Di Vito V., Beran J., Kabrt T., Grzybowski P., Rogalski T., Maslowski P., Montesarchio M. Flight management enabling technologies for single pilot operations in Small Air Transport vehicles in the COAST project (2021) IOP Conf. Ser.: Mater. Sci. Eng. 1024 012089.*
8. Аналіз стану безпеки польотів за результатами розслідування авіаційних подій та інцидентів з цивільними повітряними суднами України та іноземними ПС, що сталися у 2022 році. Національне бюро з розслідування авіаційних подій та інцидентів з цивільними повітряними суднами. – 57 с. Посилання: <https://nbaai.gov.ua/wp-content/uploads/2023/05/2022.pdf>.
9. Волощук В.М., Бойченко С.Г., Щербань І.М. Клімат України. Вид-во Раєвського, Київ, 2003. – 306 с.
10. *Shevchenko, O., Lee, H., Snizhko, S., Mayer, H. Long-term analysis of heat waves in Ukraine (2014) International Journal of Climatology, 34 (5), pp. 1642-1650.*
11. *Demuzere, M., Bechtel, B., Middel, A., Mills, G. Mapping Europe into local climate zones (2019) PLoS ONE, 14 (4), art. no. e0214474. http://www.interscience.wiley.com/jpages/0899-8418.*

12. *Montesarchio1 M., Zollo1 A., Ferrucci1 M., Bucchignani1 E. Latest developments in AWAS: the Advanced Weather Awareness System in the COAST Project (2021) IOP Conference Series: Materials Science and Engineering, Volume 1226, International Conference on Innovation in Aviation & Space to the Satisfaction of the European Citizens (11th EASN 2021).*

13. *Grzybowski P., Rogalski T., Filipowicz M. Experimental verification of the emergency destination definition in Flight Reconfiguration System in the COAST project. (2021) Submitted to EASN 2021 Conference Proceedings, IOP Conference Series: Materials Science and Engineering.*

14. *Grzybowski P., Szpakowska-Peas E. Flight reconfiguration system – an emergency system of the future (2020) Aircraft Engineering and Aerospace Technology, DOI 10.1108/AEAT-03-2020- 0052.*

15. *Saba T., Rehman A., AlGhamdi J.S. Weather forecasting based on hybrid neural model. Appl. Water Sci. 2017, 7, 3869–3874.*

16. *Leelavinodhan P.B., Vecchio M., Antonelli, F.; Maestrini, A.; Brunelli, D. Design and Implementation of an Energy-Efficient Weather Station for Wind Data Collection. Sensors 2021, 21, 3831.*

17. Прусов В. А. Методи прикладного системного аналізу в гідрометеорології: підручник / В. А. Прусов, С. І. Сніжко; Київ. нац. ун-т ім. Тараса Шевченка. – Київ: Прінт Сервіс, 2017. – 701 с.

18. Прусов В.А., Дорошенко А.Ю. Моделювання природних та техногенних процесів в атмосфері. Київ: Наукова думка, 2006. – 160 с.

19. Прусов В.А., Сніжко С.І. Математичне моделювання атмосферних процесів. К.: Ніка-Центр, 2005. – 496 с.

20. *Caernarfon Airport Pleasure Flights.* Посилання: <https://www.visitwales.com/attraction/sightseeing-tour/caernarfon-airport-pleasure-flights-1887769>

21. *Seoul Gimpo International Airport/* Посилання: <https://centreforaviation.com/data/profiles/airports/seoul-gimpo-international-airport-gmp>



22. *Kuala Lumpur Subang Abdul Razak Airport*. Посилання: [https://www.airports-worldwide.com/malaysia/sultan\\_abdul\\_aziz\\_shah\\_intl\\_malaysia.php](https://www.airports-worldwide.com/malaysia/sultan_abdul_aziz_shah_intl_malaysia.php)

23. Сніжко С.І., Прусов В.А. Динамічна метеорологія. Київський національний університет імені Тараса Шевченка, ВПЦ «Київський університет», 2009. – 496 с. *ISBN 966-521-358-X*.

24. Прокопчук Д.Ю. Система інтеграції даних про метеоумови на малих аеродромах // Тези доповідей наук.-практ. конф. “Сучасні тенденції розвитку системного програмування” (23-24 листопада 2023 р.). – К.: НАУ, 2023. – С. 7-8.

## Лістинг коду створення БД

```
CREATE TABLE WeatherData (  
    ObservationDateTime DATETIME,  
    AirTemperature FLOAT,  
    Humidity FLOAT,  
    AtmosphericPressure FLOAT,  
    WindSpeed FLOAT,  
    WindDirection FLOAT,  
    PrecipitationAmount FLOAT,  
    PrecipitationType VARCHAR(255),  
    OxygenSaturation FLOAT,  
    CO2Level FLOAT,  
    PRIMARY KEY (ObservationDateTime)  
);  
-- Створення таблиці "DataSources" для зберігання інформації про джерела даних  
CREATE TABLE DataSources (  
    SourceID INT AUTO_INCREMENT,  
    SourceName VARCHAR(255),  
    SourceDescription TEXT,  
    SourceAvailability BIT,  
    ConnectionParameters TEXT,  
    PRIMARY KEY (SourceID)  
);  
-- Створення таблиці "Airports" для зберігання інформації про аеродроми  
CREATE TABLE Airports (  
    AirportID INT AUTO_INCREMENT,  
    AirportName VARCHAR(255),  
    Latitude FLOAT,
```

*Longitude* *FLOAT*,  
*WeatherStationInfo* *TEXT*,  
*OtherParameters* *TEXT*,  
*PRIMARY KEY* (*AirportID*)

);

-- Створення таблиці "Users" для зберігання інформації про користувачів

*CREATE TABLE* *Users* (  
*UserID* *INT AUTO\_INCREMENT*,  
*UserName* *VARCHAR(255)*,  
*Login* *VARCHAR(255)*,  
*Password* *VARCHAR(255)*,  
*AccessRights* *VARCHAR(50)*,  
*PRIMARY KEY* (*UserID*)

);

-- Створення таблиці "EventLog" для зберігання журналу подій

*CREATE TABLE* *EventLog* (  
*EventID* *INT AUTO\_INCREMENT*,  
*EventDateTime* *DATETIME*,  
*EventDescription* *TEXT*,  
*PRIMARY KEY* (*EventID*)

);

Лістинг коду опитування *API* метесервісів

```
<?php
// Ключі API для трьох різних сервісів погоди
$api_keys = [
    "API_KEY_1",
    "API_KEY_2",
    "API_KEY_3"
];
// Координати місця (наприклад, Київ)
$lat = 50.4501;
$lon = 30.5234;

// Масив для зберігання температур з кожного сервісу
$temperatures = [];

foreach ($api_keys as $api_key) {
    // URL для запиту погоди
    $url = "http://api.weather-service-
1.com/weather?lat=$lat&lon=$lon&appid=$api_key"; // Замініть на відповідний URL

    // Виконання запиту за допомогою функції curl
    $ch = curl_init($url);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    $response = curl_exec($ch);
    curl_close($ch);

    // Розпарсювання відповіді у форматі JSON
    $data = json_decode($response, true);
```

```
// Додавання температури до масиву
if (isset($data["main"]["temp"])) {
    $temperatures[] = $data["main"]["temp"];
}
}

// Обчислення середньої температури
if (!empty($temperatures)) {
    $average_temperature = array_sum($temperatures) / count($temperatures);
    echo "Середня температура: " . $average_temperature . "°C\n";
} else {
    echo "Помилка отримання даних про погоду з усіх сервісів.\n";
}
?>
```