

DOI: 10.32347/2412-9933.2024.60.56-70

УДК 004.7:004.4:004.8

Жовнір Юрій Іванович

Аспірант кафедри інформаційних систем та мереж,

<https://orcid.org/0009-0006-6186-2861>

Національний університет «Львівська політехніка», Львів

Грибовський Олег Миколайович

Аспірант кафедри інформаційних систем та мереж,

<https://orcid.org/0009-0005-6318-3611>

Національний університет «Львівська політехніка», Львів

Орлов Микола Вікторович

Аспірант кафедри інформаційних систем та мереж,

<https://orcid.org/0009-0007-9835-6177>

Національний університет «Львівська політехніка», Львів

Дуда Олексій Михайлович

Кандидат технічних наук, доцент, доцент кафедри комп'ютерних наук,

<https://orcid.org/0000-0003-2007-1271>

Тернопільський технічний університет ім. Івана Пулюя, Тернопіль

Кунанець Наталія Едуардівна

Доктор наук із соціальних комунікацій, професор, професор кафедри інформаційних систем та мереж,

<https://orcid.org/0000-0003-3007-2462>

Національний університет «Львівська політехніка», Львів

МЕТОДОЛОГІЯ РОЗРОБЛЕННЯ ТА СУПРОВОДУ ІНФОРМАЦІЙНИХ СИСТЕМ, БАЗОВАНИХ НА ТЕХНОЛОГІЇ ІНТЕРНЕТУ РЕЧЕЙ

***Анотація.** Стаття присвячена дослідженню методологій розроблення та супроводу інформаційних систем, що базуються на технології Інтернету речей, із особливим акцентом на використанні методології DevOps. Технологія Інтернет речей швидко розвивається, проникаючи у різні галузі, включаючи управління інфраструктурою. Сучасні IoT-системи вимагають високого рівня надійності, масштабованості та безпеки, оскільки вони взаємодіють із великою кількістю пристроїв та опрацьовують значні обсяги даних у реальному масштабі часу. Тому актуальність статті полягає у необхідності визначення найефективніших методологій для проектування, підтримки та оновлення IoT-систем з урахуванням складності таких рішень і швидкого розвитку технологій. Результати дослідження засвідчують, що методологія DevOps є однією з найбільш придатних для розроблення IoT-систем, оскільки забезпечує автоматизацію процесів, безперервну інтеграцію, регулярні оновлення та високу стабільність систем. Використання DevOps дає змогу значно підвищити продуктивність команд розробників та операційних фахівців, які працюють над IoT-рішеннями. У дослідженні розглянуто основні етапи розроблення IoT-систем, включаючи аналіз вимог, архітектурне проектування, тестування, впровадження, а також їхнє обслуговування. Описано сучасні підходи до забезпечення безпеки, інтеграції та масштабованості IoT-систем, які включають автоматизований моніторинг і керування оновленнями програмного й апаратного забезпечення. У статті також представлено аналіз реальних прикладів використання методології DevOps у низці проєктів вітчизняних та зарубіжних IT-компаній. Наприклад, розглянуто успішні кейси впровадження DevOps для безпекових систем житлових комплексів, систем управління водопостачанням і водовідведенням у великих містах, систем контролю мікроклімату у промислових будівлях і офісах, а також систем управління освітленням у міських умовах. Результати дослідження свідчать про високу ефективність використання DevOps для IoT-проєктів, що вимагають стабільності, гнучкості та масштабованості. Використання цієї методології дає змогу оптимізувати процеси розроблення та супроводу, забезпечити безперебійне функціонування IoT-систем і підвищити рівень кібербезпеки, що є критично важливим для сучасних інфраструктурних рішень. Використання методології DevOps у таких проєктах, як безпекові системи житлових комплексів, системи управління водопостачанням і водовідведенням, контролю мікроклімату і освітлення у містах, демонструє її ефективність. Методологія DevOps*

уможливорює легко масштабувати інформаційні системи відповідно до зростаючих потреб міст, що є критично важливим для інфраструктурних рішень, де кількість підключених пристроїв може постійно зростати.

Ключові слова: інтернет речей (IoT); методологія розроблення; супровід інформаційних систем; DevOps; Agile; інфраструктурні IoT-системи; безпекові системи; системи водопостачання і водовідведення; контроль мікроклімату; управління освітленням; вітчизняні IT-компанії; зарубіжні IT-компанії; автоматизація процесів; безпека IoT; масштабованість

Вступ

Використання сучасної методології розроблення та супроводу інформаційних систем, що базуються на технології Інтернету речей (IoT), зумовлена стрімким зростанням ринку IoT-рішень та їх інтеграцією в різні сфери економіки, промисловості та побуту. IoT-системи активно використовуються для автоматизації виробничих процесів, управління інфраструктурою, моніторингу стану здоров'я, логістики та управління ресурсами. За оцінками аналітиків, кількість підключених IoT пристроїв у світовому масштабі продовжує стрімко зростати, що формує попит на ефективні методи розроблення та супроводу таких систем.

На прикладі реальних IT-проєктів, що втілюються як вітчизняними, так і та зарубіжними компаніями можна чітко відслідковувати актуальні тенденції цієї проблеми. Компанія SoftServe, ELEKS, ASTRA реалізують рішення для автоматизації «розумних» будівель, що включають функції дистанційного контролю систем безпеки, управління освітленням, кліматичними параметрами, водопостачанням та водовідведенням. Такі проєкти вимагають ретельного розроблення архітектури IoT-систем, забезпечення безпеки даних, а також постійного моніторингу та супроводу. Компанія IBM працює над розробленням IoT-платформ для управління виробничими процесами в промисловості. Це охоплює збір та аналіз даних з обладнання в режимі реального масштабу часу, що дає змогу оптимізувати виробництво, знижувати витрати та підвищувати продуктивність. У цьому випадку методологія супроводу й обслуговування IoT-систем є критично важливою для забезпечення їх стабільної та ефективної роботи.

Розвиток технологій IoT та зростання попиту на «розумні» рішення обумовлюють необхідність розроблення чітких методологій, які забезпечують безперервність, масштабованість та безпеку IoT-систем, що є актуальним як для вітчизняного, так і для глобальних ринків

Аналіз останніх досліджень і публікацій

Аналіз наукових публікацій з проблематики розроблення та супроводу інформаційних систем на

основі технології IoT засвідчує, що інтерес до цього напрямку значно зріс у зв'язку з розвитком цих технологій і потребою в ефективних підходах до управління складними системами. У публікаціях розглядаються як загальні питання розроблення IoT-систем, так і специфічні аспекти застосування методологій, таких як DevOps, Agile та інші підходи до такого роду IT-інновацій.

Застосування методології DevOps в інформаційних системах на основі IoT

Низка досліджень акцентує увагу на перевагах методології DevOps для розроблення IoT-систем, зокрема на його здатності забезпечити стабільність і автоматизацію процесів їх супроводу. Автори [1] вважають важливим аспектом реалізації IT-проєктів інтеграцію безперервної доставки і тестування, які є критичними для безперервної підтримки і оновлення IoT-систем. У роботі проаналізовано автоматизацію процесів збирання, тестування та розгортання з метою прискорення випуску високої якості і стабільності програмного забезпечення, акцентуючи увагу на важливості безпеки та відповідності в середовищі DevOps. Дослідники стверджують [2], що методологія DevOps допомагає знизити ризик збоїв роботи пристроїв та забезпечити безперервний моніторинг, що особливо важливо для IoT-систем у промислових і виробничих галузях. Інші автори підкреслюють важливість безпеки в IoT системах і демонструють, як DevOps-практики допомагають швидко впроваджувати оновлення безпеки та захищати систему від кіберзагроз. Аналізуючи поточні проблеми безпеки та обмеження, пов'язані з пристроями IoT, авторами [3] також подається ілюстрація того, як пристрої безпечно обмінюються конфіденційною інформацією. У статті [4] наголошується, що моніторинг пристроїв для безперебійного підключення для швидкого розгортання та оновлення створює значні проблеми, для вирішення яких доцільно використовувати методологію DevOps як набір методів розроблення програмного забезпечення, які включають ефективну співпрацю між командами розробників (Dev) і операціоністів (Ops) для автоматизації процесу доставки і підвищення ефективності та якості.

Agile-підходи та гнучка розробка IoT-рішень

Agile, як і його складова Scrum, також часто розглядаються як кращі методології для

IoT-проектів, особливо у випадках, коли йдеться про складні та динамічні вимоги. Науковці [5] відзначають, що Agile дає змогу швидко адаптуватися до змін у специфікаціях проекту, а Scrum забезпечує короткі ітерації, що уможливають швидко отримувати зворотний зв'язок. Водночас в деяких дослідженнях наголошується, що через специфіку IoT-систем гнучкі методології потребують додаткових заходів для забезпечення інтеграції між апаратною та програмною складовими. Аналізується зокрема масштабування методології на рівні підприємства, особливості впровадження Agile в організаціях, керування Agile-командами та вимірювання успіху Agile-проектів, способи подолання труднощів. Дослідниця [6] відзначає, що модель водоспаду базується на послідовному підході до роботи, що робить її ідеальною для проектів із фіксованими вимогами та чіткими часовими рамками, яка спочатку використовувалася для розробки програмного забезпечення, а згодом отримала широке застосування в багатьох галузях. У роботах багатьох авторів розглядаються системні методології до розроблення та супроводу IoT-систем [7], зокрема мова йде про методології DevOps, Agile та Waterfall, і досліджується їхня ефективність у проектах, де потрібно поєднувати програмні та апаратні компоненти.

Особливості розроблення IoT у контексті Lean

У низці публікацій [8] розглядається використання Lean-методології для невеликих або стартапів IoT-проектів. Lean-підхід орієнтований на оптимізацію ресурсів і зменшення витрат, що особливо актуально для компаній з обмеженими бюджетами. Дослідники [9] зазначають, що Lean дає змогу швидко створювати прототипи і тестувати рішення, що є важливим для систем на базі IoT, оскільки створення прототипів допомагає швидко перевірити працездатність ідеї та знизити ризик інвестицій у сумнівну технологію.

Waterfall і V-модель для стабільних IoT-проектів

Публікація [10], присвячена традиційним методологіям Waterfall та V-моделі, підкреслює їхню значущість для стабільних і критично важливих IoT-проектів, де зміни щодо вимог мінімальні або зовсім не припустимі. У дослідженні [11] показано, що послідовність етапів і контроль кожного етапу розроблення забезпечують передбачуваність і уможливають краще контролювати якість кінцевого продукту, що є критично важливим для IoT-рішень у медицині, автомобільній промисловості та інших галузях з підвищеними вимогами до рівня безпеки.

Кібербезпека та захист даних у IoT-системах

Кібербезпека є важливим аспектом IoT-систем, і багато авторів вивчають, як методології DevOps,

Agile та інші підходи можуть сприяти підвищенню захисту. Наприклад, у публікації авторів [12] розглядається інтеграція практик безпеки у DevOps для IoT-систем, включаючи автоматизовані оновлення і моніторинг у реальному масштабі часу, що допомагає [13] вчасно виявляти потенційні загрози й оперативно на них реагувати. Аналіз наукових публікацій свідчить, що вибір методології розроблення та супроводу IoT-систем має базуватися на потребах конкретного проекту, його галузі, ресурсах і вимогах до гнучкості та масштабованості. DevOps та Agile часто згадуються як оптимальні для динамічних і великих IoT-систем, тоді як традиційні Waterfall та V-модель залишаються актуальними для стабільних проектів з високими вимогами до надійності.

Мета дослідження

Мета дослідження – проаналізувати, обґрунтувати ефективність методологій розроблення та супроводу інформаційних систем, заснованих на технології IoT, що були втілені у реальних проектах низки вітчизняних та зарубіжних ІТ-компаній. Дослідження має на меті визначити кращі практики, основні виклики й особливості, які забезпечують стабільну роботу, безпеку та масштабованість IoT-систем.

Об'єктом дослідження є процеси розроблення, впровадження та супроводу інформаційних систем, заснованих на технології Інтернету речей (IoT), на прикладі реальних проектів вітчизняних та зарубіжних ІТ-компаній.

Предметом дослідження є методологічні підходи, принципи, інструменти та практики, що застосовуються під час розроблення та супроводу IoT-систем, їхні особливості, ефективність та вплив на якість, безпеку і масштабованість інформаційних систем в умовах реальних проектів.

Викладення основного матеріалу

У сучасному ІТ-середовищі вибір методології розроблення та супроводу проектів відіграє ключову роль у досягненні успіху. Кожен підхід має свої специфічні характеристики, що впливають на процеси розроблення, тестування, розгортання і підтримки продукту, адаптуючи їх до різних потреб та умов. Усі методології пропонують певні переваги та обмеження, що визначає їхню доцільність у певних типах проектів. Тому для прийняття обґрунтованого рішення необхідно глибоко розуміти особливості методологій, їхні сильні та слабкі сторони, а також обставини, в яких їх найдоцільніше використовувати. На сьогодні популярними є методології Agile, DevOps, Waterfall, Lean, V-модель та Scrum, кожна з яких має своє призначення й особливості (таблиця). Agile і Scrum орієнтовані на

швидку адаптацію до змін, DevOps фокусується на автоматизації та безперервному розгортанні, Waterfall забезпечує чітку послідовність етапів, Lean оптимізує витрати, а V-модель акцентує увагу на тестуванні та якості. Аналіз наявних методологій розроблення та супроводу IoT-систем включає розгляд основних підходів, якими є Agile, DevOps, Waterfall, Lean, які широко застосовуються для проектів різної складності та сфери застосування.

Agile гнучка методологія розроблення, що орієнтована на швидкі релізи й адаптацію до змін. Agile фокусується на ітеративному розробленні з частими релізами та постійним зворотним зв'язком від користувачів. Це забезпечує гнучкість і швидке реагування на зміни, підходить для динамічних проектів, де вимоги часто змінюються.

Забезпечує гнучкість у розробленні, що є особливо важливим у швидко змінюваному середовищі IoT. Часті релізи дають змогу швидко впроваджувати зміни, тестувати нові функції та адаптувати систему до нових умов. Інтенсивна комунікація з користувачами допомагає краще зрозуміти їхні потреби та підвищити якість продукту.

Проте Agile не завжди враховує специфічні вимоги апаратної частини IoT-проектів, що може ускладнити процес розроблення. Часті зміни можуть призвести до нестабільності в системах, що потребують високого рівня надійності. Методологія використовується в проектах, де вимоги можуть часто змінюватися або де потрібно швидко впроваджувати нові функції, наприклад, у розробленні веб- та мобільних застосунків. Agile підходить для IoT-проектів, що вимагають швидкої адаптації до змін і можуть працювати в умовах невизначеності. Методологія добре підходить для розроблення та супроводу програмного забезпечення в інформаційних системах на основі технології IoT, але при цьому може мати суттєві обмеження в апаратній складовій проекту.

Методологія **DevOps** поєднує розроблення й операційні процеси, зосереджується на автоматизації, безперервній інтеграції та доставці, що дає змогу швидко випускати оновлення і підтримувати стабільність системи. DevOps передбачає спільну роботу між командами розробників і операційних фахівців та підтримує автоматизацію процесів, що значно спрощує розгортання та супровід IoT-систем, в яких потрібно обслуговувати велику кількість пристроїв. Забезпечує безперервну інтеграцію та доставку, що сприяє швидкому оновленню програмного забезпечення, зменшує час простою та мінімізує ризики помилок. Методологія містить інструменти для моніторингу та підтримки безпеки IoT-систем у режимі реального часу.

Методологія DevOps потребує високого рівня автоматизації та певного рівня підготовки команди,

може бути витратною для малих проектів або стартапів через потребу в додаткових ресурсах та високих технологіях.

DevOps є ідеальним варіантом для великих IoT-проектів, де потрібна стабільність, масштабованість та гнучкість. Особливо вона корисна для виробничих і промислових IoT-систем, що вимагають безперервної роботи. Ідеально підходить для великих проектів, що вимагають швидкого розгортання оновлень, таких як хмарні сервіси класу SaaS (програмне забезпечення як послуга), та IoT-систем.

Waterfall (Каскадна модель) традиційна каскадна методологія, що має чітку послідовність етапів розроблення, вимагає детально визначених вимог на початку проекту і є менш гнучкою. Підходить для проектів із чітко визначеними та стабільними вимогами. Для неї характерний лінійний підхід, де кожен етап слідує один за одним, не переходячи до наступного без завершення попереднього. Вимоги визначаються на початку проекту і рідко змінюються.

Водночас каскадна модель не забезпечує гнучкості до змін, що робить її менш придатною для динамічного середовища IoT. Повільний процес розроблення обумовлюється необхідністю послідовної реалізації етапів, що ускладнює впровадження змін у вимогах.

Методологія використовується в проектах з чіткими і стабільними вимогами, наприклад, у державних та великих корпоративних IT-проектах, де важливий повний контроль над етапами. Каскадна модель може бути застосовна для проектів, де вимоги є стабільними і зміни на етапі розроблення мінімальні. Наприклад, для побудови IoT-систем в умовах традиційних промислових процесів, де специфікації визначені чітко і заздалегідь.

Lean – методологія, орієнтована на мінімізацію витрат і швидке досягнення результату. Використовується для швидкого створення прототипів і оптимізації ресурсів. Найкраще підходить для стартапів і невеликих проектів з обмеженими бюджетами. Методологія Lean орієнтована на оптимізацію ресурсів і зменшення витрат, включає швидке створення прототипів та ітеративне розроблення.

Lean передбачає оптимізацію використання ресурсів, фокусуючись на мінімізації витрат, що робить її привабливою для IoT-проектів з обмеженим бюджетом. Придатна для швидкого прототипування, допомагає швидко тестувати ідеї та адаптувати їх до потреб ринку.

Проте методологія Lean може обмежити можливості масштабування проекту, оскільки зосереджується на мінімальних функціональних вимогах. У низці випадків якість проекту може постраждати, оскільки Lean орієнтована на швидке досягнення результату з мінімумом витрат.

Часто використовується в стартапах та невеликих компаніях, а також у проєктах, де важлива швидкість виходу на ринок і економія ресурсів.

Методологія з назвою **V-модель** – один з варіантів каскадної методології, що акцентує увагу на тестуванні кожного етапу розробки. Забезпечує високу якість і надійність продукту, підходить для критичних проєктів, таких як медичні або автомобільні IoT-системи, модифікацію каскадної методології з посиленням акцентом на тестуванні на кожному етапі. Кожен етап розроблення має відповідний етап тестування, що забезпечує високу якість. Підходить для критичних проєктів, де потрібно забезпечити надійність і точність кожного компоненту системи. Проте вимагає чітко визначених вимог на початку проєкту, що є обмеженням для IoT-систем у динамічних умовах. Не забезпечує гнучкості для внесення змін у процесі розроблення, що може обмежити адаптивність. Методологія **Scrum** є частиною методології Agile, що базується на коротких ітераціях (спринтах), з частим релізом оновлень і регулярною взаємодією з клієнтами. Підходить для динамічних проєктів, особливо в споживчому секторі, де потрібна швидка адаптація до ринку. Scrum є частиною Agile і працює на основі коротких ітерацій (спринтів), під час яких команда виконує певний обсяг робіт, а потім отримує зворотний зв'язок від замовника.

Scrum зосереджується на коротких спринтах з частим релізом функцій, що уможлиблює швидко адаптувати IoT-систему до змін і нових вимог. Включає регулярну взаємодію з клієнтами, що допомагає отримати зворотний зв'язок і поліпшити функціональність.

Проте вимагає високої дисципліни і достатнього рівня комунікації, що може бути складно організувати для великих IoT-проєктів з розподіленими командами.

Використовується у веброботі, мобільних застосунках і проєктах, де швидкість впровадження нових функцій є ключовою. Методологія Scrum ефективна для динамічних проєктів, де часті зміни є звичними, і широко використовується для IoT-проєктів у сфері споживчих рішень, де потрібно швидко реагувати на ринкові зміни.

Кожна з розглянутих вище методологій має свої переваги та обмеження для розроблення IoT-систем. Agile і Scrum є корисними для динамічних проєктів, що вимагають високої гнучкості, тоді як методологія DevOps забезпечує стабільність та ефективність для великих, масштабованих систем. Методологія Waterfall і V можуть бути придатні для проєктів з чітко визначеними вимогами, а Lean – для IoT-проєктів з обмеженими ресурсами або в умовах швидкого запуску.

Вибір методології залежить від галузі застосування IoT-системи, вимог до її гнучкості, надійності, масштабу та доступності ресурсів.

Вибір на користь DevOps як методології для розроблення і супроводу IT-проєктів із розроблення інформаційних систем, базованих на технології IoT, обумовлений низкою вагомих переваг, які особливо актуальні у сучасних динамічних умовах розвитку IT-сфери та зростаючої складності інформаційних систем. Використання методології DevOps сприяє автоматизації багатьох операційних завдань, таких як тестування, інтеграція, доставка та розгортання. Це значно прискорює розроблення і допомагає командам частіше та з меншою кількістю помилок випускати нові версії продукту. Автоматизація мінімізує потребу в ручній праці, підвищуючи ефективність роботи команди.

Методологія DevOps забезпечує впровадження практики безперервної інтеграції та доставки (CI/CD), що уможлиблює швидко впроваджувати зміни у систему і водночас підтримувати її стабільність. Це особливо важливо для великих проєктів, де нові функції та оновлення потребують регулярного і плавного розгортання без порушення роботи наявної системи.

Методологія DevOps передбачає об'єднання команд розробників та операційних фахівців, що сприяє більш тісній їх взаємодії. Це своєю чергою допомагає швидше вирішувати проблеми, обмінюватися інформацією та координувати дії, що особливо важливо для проєктів із розподіленими командами.

Завдяки автоматизованому моніторингу та швидкому розгортанню оновлень, методологія DevOps дає змогу знижувати час простоїв та мінімізувати ризик відмов системи. Це особливо важливо для критичних систем, де стабільність та безперервність роботи є пріоритетом.

Вона забезпечує легку адаптацію до змін в жорстких умовах виконання проєкту та уможлиблює масштабувати систему відповідно до зростаючих потреб без необхідності перебудовувати процеси. Це особливо важливо для проєктів, що ростуть і розширюються, як-от хмарні сервіси або IoT-рішення.

DevOps інтегрує процеси забезпечення безпеки на всіх етапах розроблення та супроводу, що знижує ризик вразливостей та дає змогу ефективно протидіяти кіберзагрозам. Інструменти DevOps забезпечують моніторинг та автоматизоване впровадження оновлень безпеки. Постійне тестування і швидкий зворотний зв'язок у процесі розроблення допомагають виявляти і виправляти помилки на ранніх етапах, що значно покращує якість кінцевого продукту і знижує ризики виходу на ринок із недосконалим рішенням.

Таблиця – SWOT аналіз методологій управління проектами

Методологія	Сильні сторони	Слабкі сторони	Сфери застосування
Agile	Гнучкість у розробці, часті релізи, швидке адаптування до змін	Неврахування специфіки апаратної частини, можливі нестабільності	Підходить для динамічних проєктів з часто змінюваними вимогами
DevOps	Автоматизація процесів, безперервна інтеграція та доставка, моніторинг у реальному часі	Висока вартість впровадження, потреба у високій автоматизації	Оптимально для великих, масштабованих IoT-систем у промислових умовах
Waterfall	Чітка структура, контроль кожного етапу, підходить для стабільних проєктів	Низька гнучкість, тривалий процес розроблення	Для проєктів з фіксованими вимогами, стабільних умовах
Lean	Оптимізація ресурсів, швидке прототипування, мінімізація витрат	Обмежена масштабованість, можливі компроміси у якості	Для стартапів та малих проєктів з обмеженими ресурсами
V-модель	Чітка структура з акцентом на тестуванні, висока якість продукту	Вимога чітко визначених вимог, низька гнучкість	Для критичних проєктів з високими вимогами до надійності
Scrum	Часті релізи, короткі спринти, регулярна взаємодія з клієнтами	Вимога високого рівня комунікації, недостатня структура для великих проєктів	Підходить для динамічних проєктів з частими змінами, споживчих IoT-рішень

Загалом, DevOps є вибором для проєктів, де важлива швидкість розробки, стабільність, можливість регулярного оновлення та висока якість. Методологія особливо корисна для масштабованих, надскладних та критично важливих систем, які потребують надійної роботи в режимі 24/7. Завдяки поєднанню автоматизації, гнучкості та покращеної співпраці, використання методології DevOps підвищує ефективність розроблення та супроводу проєктів, що робить її оптимальним вибором для сучасних IT-компаній.

IoT-системи зазвичай включають велику кількість пристроїв, підключених до мережі, що робить процес оновлення програмного забезпечення на кожному з них складним і трудомістким. Методологія DevOps уможливило автоматизувати цей процес, забезпечуючи централізоване розгортання та оновлення програмного забезпечення на всіх пристроях одночасно. Для забезпечення стабільної роботи IoT-систем необхідний постійний моніторинг стану пристроїв та системи в цілому.

DevOps-інструменти дають змогу інтегрувати процеси моніторингу і аналізу у реальному масштабі часу, що дає змогу швидко виявляти та усувати можливі проблеми, зменшуючи час простоїв.

DevOps підтримує швидке масштабування IoT-систем, що є критичним для розширення мережі пристроїв або збільшення обсягу опрацювання даних.

Отже, використання методології DevOps при розробленні IoT-систем забезпечує ефективність, безпеку та надійність таких рішень, дозволяючи швидко адаптуватися до змін і забезпечувати

стабільну роботу великих мереж підключених пристроїв.

Розглянемо кілька прикладів того, як методологія DevOps була успішно використана в поєднанні з інформаційними технологіями класу IoT в реальних проєктах низку IT-компаній з розроблення програмного забезпечення та провайдерів Інтернет-послуг.

Дослідження підтверджують доцільність і ефективність застосування методології DevOps, що сприяло успішному впровадженню IoT-рішень.

У всіх наведених нижче прикладах інформаційних систем автори статті були безпосередніми їх розробниками, виконуючи ролі системних аналітиків, системних проєктувальників, системних інтеграторів та адміністраторів.

Інформаційна безпекова система житлового комплексу.

Виробник пристосував у проєкті інформаційної безпекової системи житлового комплексу практики DevOps для покращення якості та надійності розроблених в цьому контексті програмних продуктів. Було реалізовано CI/CD-пайплайн для автоматизації тестування та розгортання, що дає змогу частіше випускати оновлення для датчиків руху, термостатів та камер безпеки. Це загалом позитивно вплинуло на задоволеність клієнтів і прискорило вихід нових безпекових функцій.

У цьому проєкті система аудіо-, відеонагляду включає низку ключових компонентів, які забезпечують повний спектр функцій для моніторингу і управління безпекою. На перших етапах реалізації системи були встановлені сучасні

камери відеоспостереження, Axis Q6215-LE, які підтримують роздільну здатність до 4K, функції панорамування, нахилу, зуму та нічного бачення. Окрім камер, у загальних зонах будинку були розміщені мікрофони Shure MX396, які забезпечують високоякісний запис аудіо (рис. 1).

Збирання даних здійснюється через шлюзи IoT, представлені Raspberry Pi 4, які підключені до камер та мікрофонів. Ці пристрої виконують первинне опрацювання даних, включаючи виявлення руху і шуму. Raspberry Pi використовує такі програмні рішення OpenCV для опрацювання відео і TensorFlow Lite. Це уможливило зменшити затримки опрацювання і обсяг даних, що передаються до хмари.

Опрацювання і зберігання даних здійснюється в хмарній інфраструктурі. Віртуальні машини на базі AWS EC2 використовуються для централізованого опрацювання відео та аудіоданих. Спочатку використовуються інстанси типу t3.large, але в разі потреби можливе масштабування до інстансів типу c5.2xlarge для опрацювання більших обсягів даних.

Додатково, функції AWS Lambda використовуються для опрацювання подій, таких як сповіщення про виявлення руху або звуку, що

уможливило автоматично запускати опрацювання даних без потреби в постійному моніторингу з боку користувачів.

Дані з камер і мікрофонів завантажуються в хмарне сховище AWS S3, де зберігаються у форматі, який уможливило реалізацію швидкого доступу і перегляду архівів. Для зберігання метаданих, таких як налаштування камер, записів подій і користувацьких даних, використовується AWS RDS для PostgreSQL.

Для аналітики відео- та аудіоданих використовуються сервіси Amazon Rekognition і Amazon Kinesis. Rekognition забезпечує розпізнавання обличчя і аналіз відео, тоді як Kinesis дає змогу опрацьовувати відео- та аудіопотоки в реальному масштабі часу, інтегруючись з Rekognition для додаткового аналізу.

Користувачі можуть взаємодіяти із системою через мобільний застосунок, розроблений за допомогою React Native, який забезпечує доступ до відео в реальному часі, отримання сповіщень і налаштування системи. Вебінтерфейс, створений за допомогою React.js, дає змогу переглядати архіви, аналізувати дані і управляти системою безпеки.

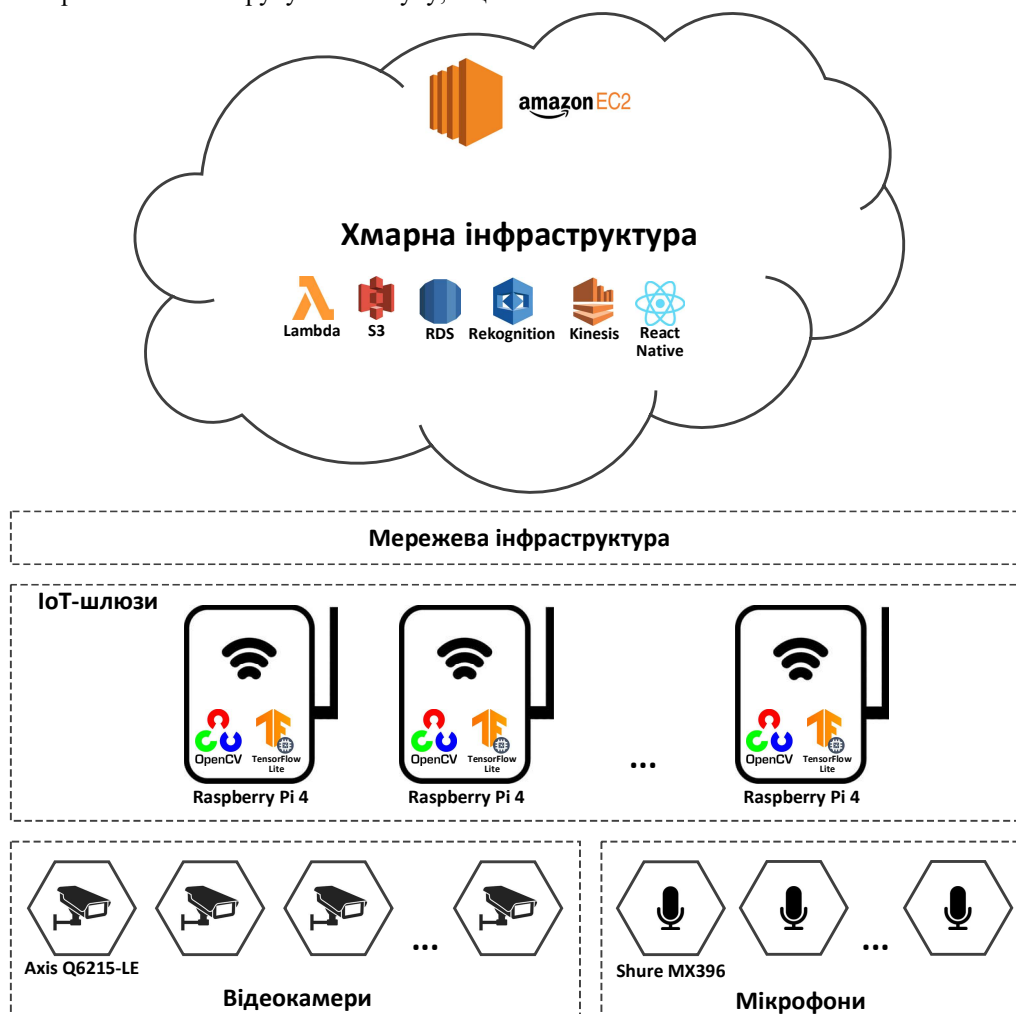


Рисунок 1 – Компоненти системи аудіо-, відеонагляду

Інформаційна система контролю та управління водопостачанням та водовідведенням для великих міст на основі технологій IoT.

У цьому проекті система контролю водо- та тепlopостачання великих міст поєднує сенсорний рівень, мережевий рівень та хмарну інфраструктуру. Сенсорний рівень своєю чергою умовно поділений на три підрівні.

Підрівень датчиків містить лічильники витрат води та тепла, інтегровані у відповідні міські ресурсні мережі. При цьому використовуються механічні й інтелектуальні лічильники витрат спожитих ресурсів та послуг. Інтелектуальні лічильники підключаються до IoT-пристроїв

засобами промислового протоколу M-BUS, а механічні за допомогою інтерфейсів RS485 та RS232 до аналогових та імпульсних входів представлених на відповідному підрівні (рис. 2).

Для регулювання параметрів процесів водо- та тепlopостачання були інстальовані різнотипові IoT-пристрої з підключеними лічильниками. Водночас мережевий рівень містить повсюдну міську мережу, яка за допомогою LAN, 3G/4G/5G, WiFi, WiMax або Bluetooth забезпечує безперешкодний доступ IoT-пристроїв до мережі Інтернет. Взаємодія IoT-пристроїв з повсюдною міською мережею може відбуватись на основі комунікаційних технологій IPv6, 6LowPAN, Zigbee, Z-Wave, NFC та LoRaWAN.

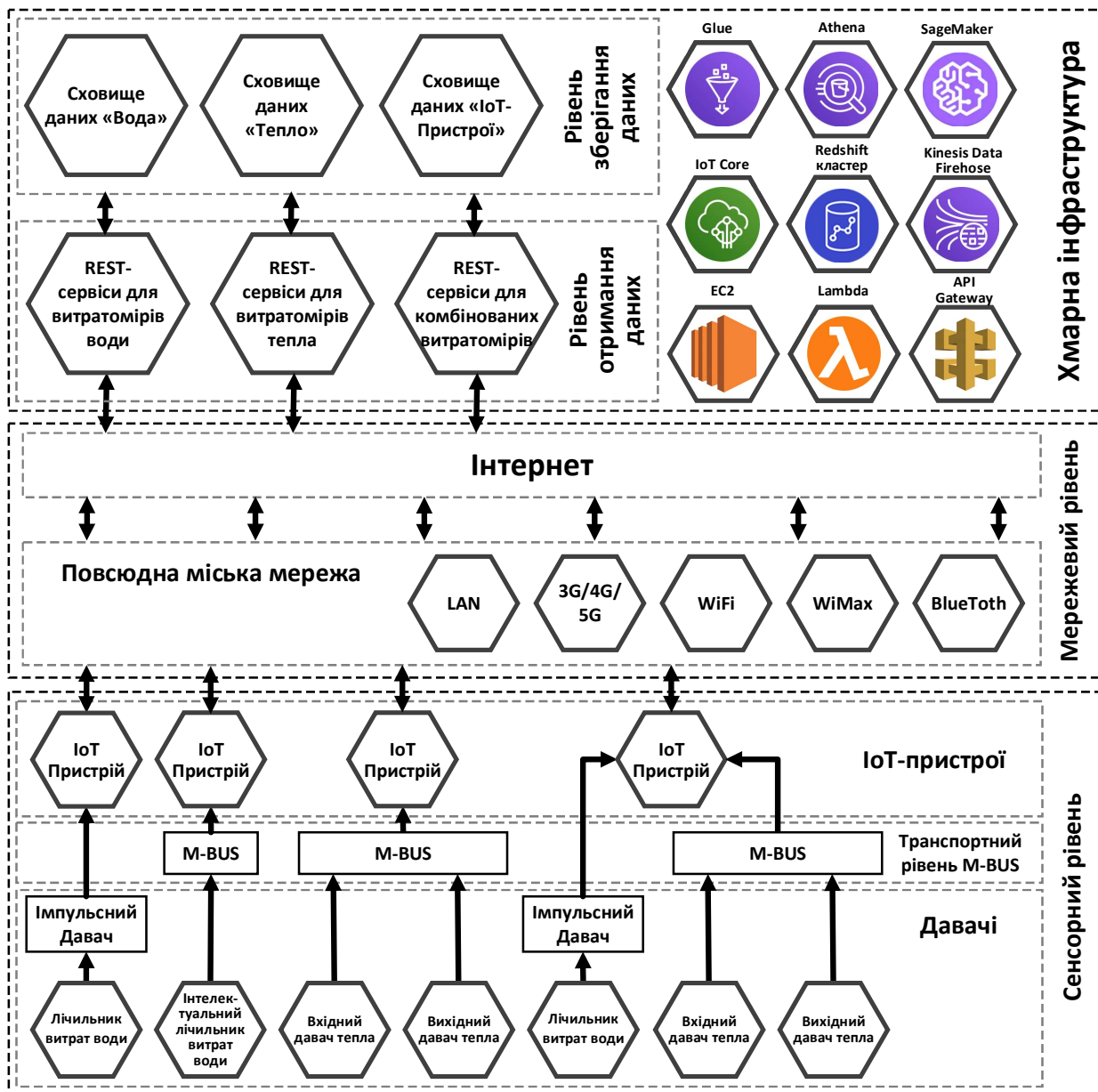


Рисунок 2 — Компоненти системи контролю водо- та тепlopостачанням великих міст

На базі хмарної платформи реалізовані два рівні. Зокрема, для взаємодії з IoT-пристроями та отримання даних на окремому рівні реалізовані

REST-сервіси. При цьому для формування віртуальних серверів для запуску застосунків задіяно AWS EC2, для виконання невеликих функцій задіяно

безсерверну обчислювальну платформу AWS Lambda, а для створення REST API використано сервіс AWS API Gateway. Використання AWS EC2 дає змогу автоматично масштабувати кількість інстансів вгору або вниз відповідно до поточного навантаження на інформаційну систему. Це своєю чергою забезпечує оптимальне використання ресурсів і знижує фінансові витрати на впровадження та використання інфраструктури. Для управління IoT-пристроями задіяно сервіс AWS IoT Core, що знижує фінансові витрати на впровадження та використання інфраструктури. Для управління IoT-пристроями задіяно сервіс AWS IoT Core. Накопичені набори даних передаються на наступний рівень та зберігаються в розподіленому масштабованому сховищі даних. При цьому для кожного IoT-пристрою продукується окремий набір інформаційних сутностей, які групуються в тематичних базах даних. Для високопродуктивного та оперативного виявлення й індексування структури зібраних від IoT-пристроїв даних використано AWS Glue, яка інтегрує потужний набір машинних алгоритмів та процедур автоматизації процесів створення, запуску і використання конвеєрів машинного опрацювання даних. Щоб спростити процеси збирання, трансформації та доставки потокових даних використано Kinesis Data Firehose. Для багатовимірного аналізу метаданих було використано AWS Redshift кластер, а для аналітичного опрацювання даних задіяно AWS Sage Maker. Використання хмарних сервісів AWS Glue, Redshift та SageMaker дає змогу описувати

інфраструктуру як код, спрощуючи процедури управління нею та масштабування.

Інформаційна система контролю мікроклімату в приміщеннях на основі технології IoT.

У проєкті інформаційної системи контролю мікроклімату в приміщеннях були використані хмарні сервіси Amazon Web Services для автоматизації процесів, забезпечення ефективності роботи системи та її масштабування. Використання Kinesis Data Streams автоматизує процедури збирання, опрацювання та доставки потокових даних з IoT-пристроїв.

У цьому проєкті система контролю мікроклімату в приміщеннях поєднує низку ключових компонентів, які забезпечують повний спектр функцій для спостереження та регулювання повітряного середовища. На перших етапах реалізації системи використано IoT-пристрої на базі мікроконтролерів Arduino UNO R3 (CH340) та Arduino Leonardo (ATmega32u4). Для відбору показників повітряного середовища було використано групу датчиків, за допомогою яких відбувається відбір даних IoT-пристроями. Зокрема, DHT11, DHT22 та BME280 для вимірювання температури та вологості повітряного середовища, LM35 для вимірювання температури повітряного середовища, MQ-135 для вимірювання концентрації шкідливих газів. Окрім цих давачів, у приміщеннях були розміщені давачі PM2.5, які забезпечують процес вимірювання концентрації твердих частинок у повітряному середовищі (рис. 3).

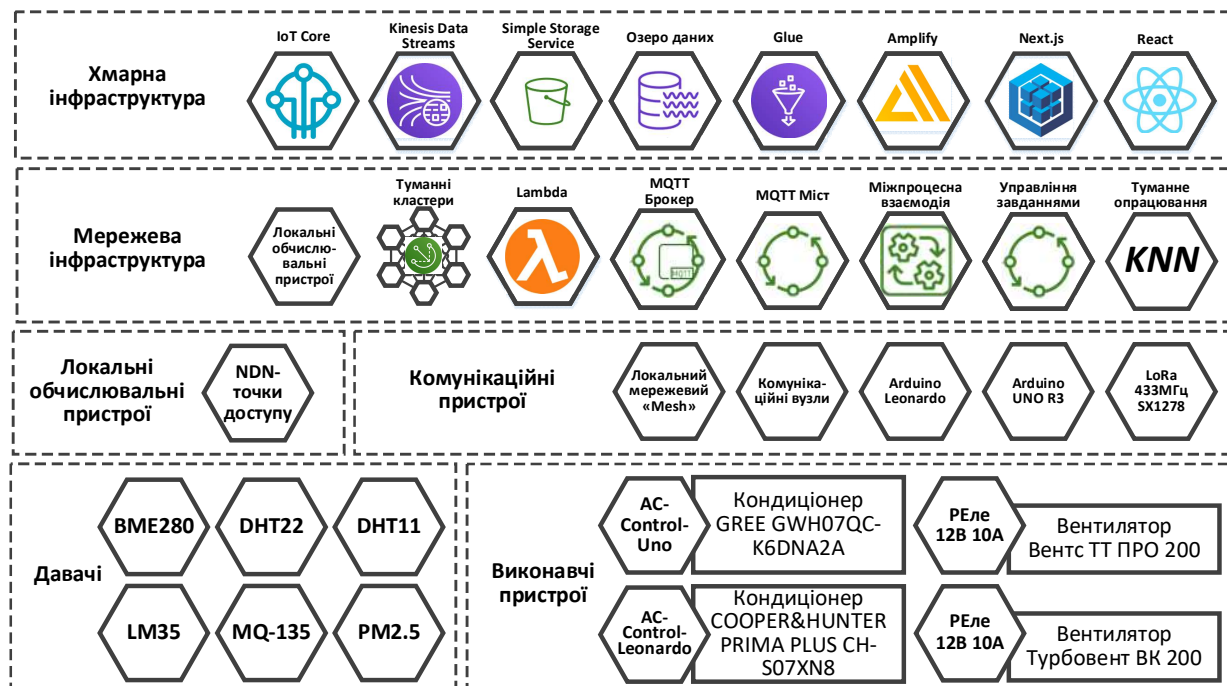


Рисунок 3 – Компоненти системи контролю мікроклімату в приміщеннях

Для регулювання параметрів повітряного середовища приміщень використовувались системи вентиляції та кондиціонування. Керування кондиціонерами MQTT Брокер в Greengrass-групі дає можливість обміну даними між туманним рівнем та IoT-пристроями за допомогою мережевого протоколу MQTT. Обчислювальні засоби мережевої інфраструктури задіяно для автоматизації процесів управління системами вентиляції та кондиціонування приміщень. При цьому використано метод k-найближчих сусідів для виявлення аномалій у зібраних даних показників якості повітря приміщень.

Для отримання даних від туманних кластерів у хмарній інфраструктурі було розгорнуто службу AWS IoT Core. Для забезпечення процесів передавання зібраних з IoT-пристроїв даних у сховища використано керувану службу потокової передачі AWS Kinesis Data Streams, що дало змогу автоматизувати процедури збирання, опрацювання та доставки поточкових даних з IoT-пристроїв. У процесі зберігання інформації використано сховище даних у необробленому вигляді AWS Simple Storage Service. На його основі було сформовано озера даних. Щоб уникнути перетворення озера даних на «болото даних» було використано платформу для опрацювання великих за обсягом наборів та колекцій даних AWS Glue, що дало змогу автоматизувати створення ETL-процесів опрацювання даних. Використання хмарних сервісів Amazon передбачає використання підходу «інфраструктура як код» для створення й управління інфраструктурою. Це дає можливість автоматизувати процес розгортання та

масштабування інформаційної системи контролю мікроклімату в приміщеннях. Для швидкого створення та управління вебзастосунками використано Next.js на базі AWS Amplify. Для формування користувацьких інтерфейсів застосовано React.

Інформаційна система управління освітленням для великих міст на основі технологій IoT

Інтегрована система використання технологій Інтернету речей (IoT) та методології DevOps у розробленні інтелектуальної системи управління освітленням є ефективним перспективним підходом в забезпеченні інтеграції, автоматизації та оптимізації процесів керування освітленням міст, селищ та територіальних громадах. Такі системи мають можливість автоматично адаптувати рівень освітлення на основі обширних наборів даних, забезпечуючи при цьому високу енергоефективність і підвищений комфорт мешканців.

Інформаційна система управління освітленням для великих міст на основі технологій IoT здійснює спостереження за допомогою датчиків освітлення, зокрема, Hamamatsu S1223 та S1336, Osram BPW34, Vishay Vactrol, TDK PL-S, OMRON E3X, Sharp PD330. Водночас відбувається спостереження середовища за допомогою датчиків температури, атмосферних явищ або вітру. Система може автоматично вмикати або вимикати світло залежно від присутності людей у приміщеннях та на відповідних територіях залежно від часу доби. Для цього використовуються датчики руху, зокрема, Panasonic AMN 34111, Panasonic AMN 12112J та Honeywell HFS-D (рис. 4).

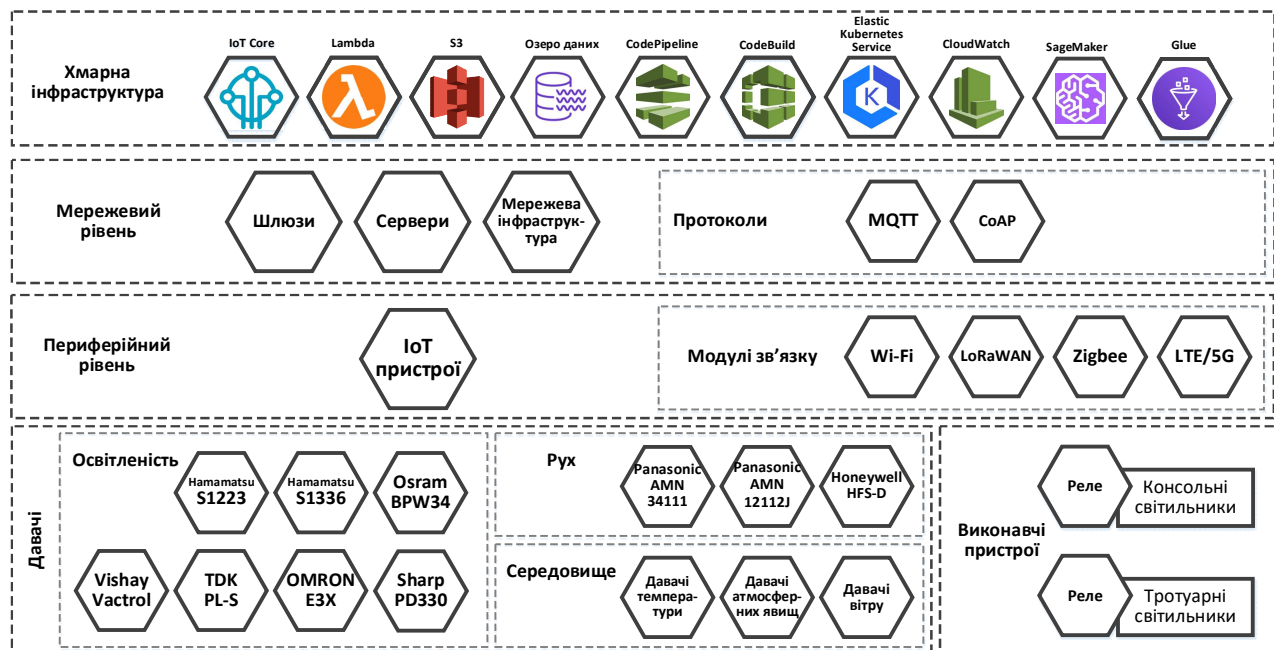


Рисунок 4 – Компоненти системи управління освітленням

Ця функція дає можливість суттєво економити електроенергію. Використовуючи смартфон або комп'ютер, оператори чи потенційний користувач може змінювати налаштування освітлення, перебуваючи у будь-якій локації. При цьому за допомогою реле відбувається вмикання чи вимикання консольних або тротуарних світильників.

Процеси відбору даних та управління освітлювальними приладами відбуваються з використанням IoT-пристроїв, які застосовують різноманітні модулі зв'язку – Wi-Fi, LoRaWAN, Zigbee або LTE/5G. DevOps практики автоматизують процеси інтеграції нових функцій у систему освітлення та постійно оновлюють її компоненти. Це своєю чергою забезпечує швидке виправлення помилок і додавання нових функцій. Такий підхід забезпечує постійний моніторинг стану мережі IoT-пристроїв, дає змогу вчасно реагувати на неполадки та системно оптимізувати роботу системи освітлення. Інтеграція систем, базованих на технологіях IoT та методології DevOps, забезпечує ефективність та гнучкість системи освітлення в умовах швидкозмінного середовища. Завдяки IoT система отримує можливість збирання даних у режимі реального часу, автоматизації процесів та управління інфраструктурою, а завдяки методології DevOps підтримується постійна інтеграція нових функцій та безперервне вдосконалення розлогої IT інфраструктури.

Мережевий рівень інформаційної системи управління освітленням для великих міст інтегрує шлюзи, сервери та різноманітну мережеву інфраструктуру, що використовує мережеві протоколи MQTT та CoAP для процесів агрегації даних IoT-пристроїв.

З метою формування безпечного і масштабованого каналу для збирання даних IoT-пристроїв на базі хмарної інфраструктури розгорнуто AWS IoT Core, що дало можливість управляти підключеннями пристроїв, а також шифрувати і фільтрувати дані перед передаванням. Функції опрацювання даних в режимі реального часу, які запускаються у відповідь на повідомлення від IoT-пристроїв, реалізовано засобами AWS Lambda – це дало змогу опрацювати та очищати дані перед збереженням. Комбінація методології DevOps та інформаційних систем, реалізованих на базі розлогих мережевих структур IoT, є ключовою при створенні ефективних, автоматизованих і безпечних розумних систем освітлення. При цьому технології IoT забезпечують збирання даних і автоматизацію процесів освітлення в реальному масштабі часу, тоді як практики методології DevOps дають можливість швидко впроваджувати нові функції та оптимізувати роботу системи шляхом постійної оперативної інтеграції та моніторингу.

Для зберігання неопрацьованих та історичних даних, що надходять від IoT-пристроїв, використано AWS S3, що забезпечує високоефективне та надійне зберігання великих обсягів інформації при організації озера даних (Data Lake). Для автоматизації процесів розгортання і тестування мікросервісів та інфраструктури задіяно AWS CodePipeline, що дало змогу інтегрувати стадії збирання, тестування, перевірки безпеки та розгортання. Для компіляції коду, запуску тестів та створення контейнерів у процесі подальшого розгортання використано AWS CodeBuild.

Amazon Elastic Kubernetes Service було використано для оркестрації контейнерів. Це дало змогу масштабувати мікросервіси в контейнерах, забезпечуючи при цьому балансування навантаження, керування ресурсами і забезпечити процедури автоматичного відновлення контейнерів. Для моніторингу продуктивності, управління метриками та налаштування повідомлень було задіяно Amazon CloudWatch. Інформаційна система аналізує дані від сенсорів та дані відповідних часових рядів і оптимізує роботу освітлення, адаптуючи її до поточних умов і потреб. При цьому ефективно реалізується функція прогнозування споживання енергії і керування освітленням. При розробленні моделей машинного навчання для прогнозування та оптимізації енергоспоживання використано Amazon SageMaker, що застосовується для навчання моделей на історичних даних. А для автоматизованого опрацювання даних перед завантаженням у аналітичну систему задіяно ETL-службу AWS Glue. У розумній системі освітлення для житлового комплексу з різними типами будівель (житлові будинки, офісні приміщення, торгові центри) реалізована функція автоматичного вмикання/вимикання освітлення залежно від часу доби, наявності людей у приміщеннях та відкритій території, а також від рівня природного освітлення. Зниження споживання енергії досягається за рахунок адаптивного регулювання інтенсивності освітлення.

Впровадження програмних оновлень відбувається засобами DevOps. Розробники працюють над реалізацією нової функції – можливості дистанційного керування освітленням за допомогою голосових команд. Набір практик DevOps забезпечує тестування цієї функції на тестових серверах, перевірку безпеки й автоматичне розгортання в умовах реально діючої системи. Використовується набір автоматичних тестів, які перевіряють роботу оновленої системи освітлення в різних сценаріях (день, ніч, критичні ситуації). Практики DevOps при цьому забезпечують постійний моніторинг оновленої системи. У разі виявлення аномалій у роботі освітлювальних пристроїв або збільшення споживання енергії,

система автоматично відправляє повідомлення інженерам для оперативного опрацювання ситуації та її виправлення.

Комбінація інформаційних систем керування освітленням з використанням інформаційних технологій IoT та методології DevOps у системі освітлення забезпечує високу ефективність, безперервне вдосконалення та швидке реагування на виклики та нові потреби користувачів. Така інтеграція уможливорює реально знижувати енергоспоживання, забезпечувати зручне управління освітленням та швидко додавати нові функції без переривання роботи системи. Система проходить стендове тестування базових функцій і найближчим часом буде передана в дослідну експлуатацію на території декількох міських житлових комплексів.

Наведені реальні приклади яскраво ілюструють універсальність методології DevOps у задоволенні різноманітних потреб масштабних IoT-проектів у різних галузях. Незалежно від сфери тієї чи іншої аплікації методологія DevOps виявилась цінним союзником у прискоренні інновацій та підвищенні надійності систем, базованих на інформаційних технологіях класу IoT.

Висновки

У статті проведено системний аналіз методологій розроблення та супроводу інформаційних систем, заснованих на технології Інтернету речей (IoT), зокрема на прикладах вітчизняних та зарубіжних IT-компаній. Отримані результати дослідження свідчать про важливість вибору відповідної методології, яка враховує специфіку IoT-проектів, а також їхню потребу в гнучкості, масштабованості та безпеці.

DevOps та Agile є найбільш придатними методологіями для сучасних IoT-проектів, які вимагають безперервної інтеграції, доставки та адаптації до змінних умов. DevOps забезпечує автоматизацію, стабільність та швидкість розгортання, що є критичним для великих і масштабованих IoT-рішень.

Методології Waterfall та V залишаються актуальними для проектів із чітко визначеними та стабільними вимогами, наприклад, у галузях із високими вимогами до надійності та безпеки (медицина, автомобільна промисловість). Їхня послідовність та контроль над кожним етапом забезпечують передбачуваність та контроль якості.

Lean є оптимальним вибором для невеликих проектів або стартапів, що потребують мінімізації витрат і швидкого прототипування. Ця методологія сприяє ефективному використанню ресурсів і швидкій перевірці концепцій, що особливо важливо для IoT-проектів з обмеженими бюджетами.

Agile та Scrum забезпечують гнучкість і допомагають швидко адаптувати IoT-системи до змін. Це сприяє задоволенню потреб кінцевих користувачів і забезпечує ефективну комунікацію та зворотний зв'язок, що важливо для рішень у динамічних ринкових умовах.

Для забезпечення кібербезпеки та захисту даних IoT-систем, особливо в умовах зростання кількості підключених пристроїв, необхідно інтегрувати заходи безпеки на кожному етапі розробки та супроводу, що забезпечується методологією DevOps.

Загалом, при виборі методології враховується специфіка проекту, його масштаб і вимоги до безпеки. Використання відповідної методології дає змогу забезпечити стабільну роботу IoT-систем, оптимізувати процеси розроблення і супроводу та підвищити загальну ефективність IT-проектів.

Аналіз використання методології DevOps у контексті розроблення і супроводу IoT-систем для житлових, міських та інфраструктурних потреб демонструє її високу ефективність і доцільність. Використання DevOps у проектах для безпекових систем житлового комплексу, систем контролю водопостачання та водовідведення, управління мікрокліматом і освітленням у великих містах обґрунтовується нижченаведеними факторами.

Використання методології DevOps є доцільним для розроблення інформаційних безпекових систем, оскільки забезпечує безперервний моніторинг, автоматизоване оновлення і високу стабільність системи. Безпекові системи потребують безперебійної роботи і миттєвого реагування на загрози. DevOps уможливорює автоматизувати розгортання оновлень для забезпечення кібербезпеки та швидко реагувати на потенційні вразливості, що є критичним для захисту житлових комплексів.

Для інформаційних систем управління водопостачанням та водовідведенням важливо забезпечити безперебійне постачання і ефективне управління ресурсами, а також контролювати якість води. DevOps підтримує швидке масштабування, автоматизоване тестування та моніторинг, що допомагає ефективно управляти системами, розміщеними на численних об'єктах по всьому місту. Постійний моніторинг і можливість безперервного оновлення систем допомагають запобігти простою і знижують ризик аварій.

Методологія DevOps підходить для розроблення інформаційних систем управління мікрокліматом у великих будівлях чи промислових об'єктах, де потрібно підтримувати постійну температуру, вологість і рівень CO₂. За допомогою методології DevOps можна впровадити автоматизовані процеси, які дають змогу не лише збирати дані в реальному масштабі часу, а й відразу

коригувати налаштування для підтримки оптимального мікроклімату. Методологія DevOps забезпечує своєчасне оновлення систем управління мікрокліматом, щоб уникнути збоїв і втрат енергії.

Управління міським освітленням потребує оптимального використання ресурсів і надійного контролю над системами в реальному масштабі часу. Завдяки використанню методології DevOps можна легко масштабувати інформаційну систему, інтегрувати нові функції, автоматизувати процеси моніторингу і контролю, що допомагає ефективно керувати освітленням у великих міських зонах. Автоматичне оновлення і безперервний моніторинг дають змогу своєчасно впроваджувати оновлення та оптимізувати споживання енергії, знижуючи витрати.

Використання методології DevOps у таких проєктах, як безпекові системи житлових комплексів, системи управління водопостачанням і водовідведенням, контролю мікроклімату і освітлення у містах, демонструє її ефективність.

Методологія DevOps дає змогу легко масштабувати інформаційні системи відповідно до зростаючих потреб міст, що є критично важливим для інфраструктурних рішень, де кількість підключених пристроїв може постійно зростати. DevOps забезпечує високу стабільність завдяки безперервному моніторингу й автоматизованому оновленню, що мінімізує ризик збоїв у критичних інфраструктурних системах. У разі виявлення загроз або несправностей, методологія DevOps забезпечує швидке оновлення та усунення вразливостей, що є важливим для безпеки і стабільності таких систем.

Завдяки автоматизації багатьох процесів DevOps допомагає оптимізувати використання енергетичних та людських ресурсів, що особливо актуально для великих інфраструктурних систем.

Застосування DevOps у цих проєктах допомагає забезпечити безперервну і надійну роботу IoT-систем, які є критично важливими для сучасного міста.

Список літератури

1. Kim, G., Debois, P., Willis, J., Humble, J., Allspaw, J. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations. Portland: IT Revolution Press, 2016. 480 с.
2. Perry, Y. DevOps: The Complete Guide to Culture, Technology and Tools. URL: <https://bluexp.netapp.com/devops>.
3. Varadharajan, V., Bansal, S. Data Security and Privacy in the Internet of Things (IoT) Environment. 2016. 10.1007/978-3-319-33124-9_11. URL: <https://api.semanticscholar.org/CorpusID:168449484>.
4. DevOps and IoT: How to Integrate Them for Success. URL: <https://www.mindbrowser.com/integrating-devops-and-iot/>.
5. Breyter, M. Agile Product and Project Management. Apress, 2022. URL: <https://www.perlego.com/book/4513995/agile-product-and-project-management-a-stepbystep-guide-to-building-the-right-products-right-pdf> (Original work published 2022).
6. Klein, M. Waterfall Model Methodology: Everything You Need to Know. URL: <https://project-management.com/waterfall-model/>.
7. Chantzis, F., Stais, I., Calderon, P., Deirmentzoglou, E., Woods, B. Practical IoT Hacking. San Francisco: No Starch Press, 2021. 434 с.
8. Докладний посібник з методологій Lean та Agile. URL: <https://worksection.com/ua/blog/agile-vs-lean.html>.
9. Riepina, I. Using the Lean manufacturing methodology to improve the quality of the enterprise's business processes = Використання методології Lean production для підвищення якості бізнес-процесів підприємства. *Менеджмент*. 2023. Вип. 1 (37). С. 39–49.
10. Коваленко, О. О., Денисюк, А. В., Бажан, В. М. Порівняльний аналіз методологій розробки ІТ продукту. *Вісник Харківського національного університету імені В. Н. Каразіна. Серія: Інформаційні технології*. 2023. Вип. 25. С. 45–55.
11. Adeagbo, M., Akinsola, J.E.T., Awoseyi, A., Kasali, F. Project Implementation Decision Using Software Development Life Cycle Models: A Comparative Approach. *Journal of Computer Science and Its Application*. 2021. 28. DOI: 10.4314/jcsia.v28i1.10.
12. Cybersecurity for Industrial Internet of Things: Architecture, Models and Lessons Learned. URL: https://paris1.hal.science/hal-03967801/file/Cybersecurity_for_Industrial_Internet_of_Things_Architecture_Models_and_Lessons_Learned.pdf.
13. Models and Methods of Information and Control System Cybersecurity for Smart Buildings. URL: https://www.researchgate.net/publication/371044780_Models_and_Methods_of_Information_and_Control_System_Cybersecurity_for_Smart_Buildings.
14. Models and Methods of Information and Control System Cybersecurity for Smart Buildings. ieved from https://www.researchgate.net/publication/371044780_Models_and_Methods_of_Information_and_Control_System_Cybersecurity_for_Smart_Buildings

Стаття надійшла до редколегії 11.11.2024

Zhovnir Yuri Ivanovich

Postgraduate student of the Department of Information Systems and Networks,

<https://orcid.org/0009-0006-6186-2861>

Lviv Polytechnic National University, Lviv

Hrybovskiy Oleg Mykolayovych

Postgraduate student of the Department of Information Systems and Networks,

<https://orcid.org/0009-0005-6318-3611>

Lviv Polytechnic National University, Lviv

Orlov Mykola Viktorovych

Postgraduate student of the Department of Information Systems and Networks,

<https://orcid.org/0009-0007-9835-6177>

Lviv Polytechnic National University, Lviv

Duda Oleksiy Mykhailovych

Associate Professor of the Department of Computer Sciences,

<https://orcid.org/0000-0003-2007-1271>

Ternopil Technical University named after Ivan Pulyuya, m. Ternopil

Kunanets Natalia Eduardivna

DSc (Social Communications), Professor, Professor of the Department of Information Systems and Networks,

<https://orcid.org/0000-0003-3007-2462>

Lviv Polytechnic National University, Lviv

METHODOLOGY OF DEVELOPMENT AND MAINTENANCE OF INFORMATION SYSTEMS BASED ON THE TECHNOLOGY OF THE INTERNET OF THINGS

Abstract. *The article is devoted to the study of development and maintenance methodologies for information systems based on the Internet of Things (IoT) technology, with a particular focus on the use of the DevOps methodology. The Internet of Things is rapidly evolving, penetrating various industries, including infrastructure management, security systems, resource management, and water supply and sewage control. Modern IoT systems require a high level of reliability, scalability, and security as they interact with numerous devices and process large volumes of real-time data. Therefore, the relevance of this study lies in the need to determine the most effective methodologies for designing, supporting, and updating IoT systems, considering the complexity of such solutions and the rapid technological advancements. Research results show that the DevOps methodology is one of the most suitable for developing IoT systems, as it enables process automation, continuous integration, regular updates, and high system stability. Using DevOps significantly enhances the productivity of development and operations teams working on IoT solutions. The study examines the main stages of IoT system development, including requirements analysis, architectural design, testing, deployment, and maintenance. Modern approaches to ensuring the security, integration, and scalability of IoT systems are described, which include automated monitoring and managing updates of software and hardware. The article also provides an overview of real examples of the DevOps methodology application in several projects by domestic and foreign IT companies. For instance, successful cases of DevOps implementation in residential complex security systems, water supply and sewage control systems for large cities, microclimate control systems in industrial buildings and offices, and city lighting control systems are reviewed. The research results demonstrate the high effectiveness of using DevOps for IoT projects that require stability, flexibility, and scalability. Applying this methodology allows optimizing development and maintenance processes, ensuring the uninterrupted operation of IoT systems, and enhancing cybersecurity levels, which is critically important for modern infrastructure solutions. The use of the DevOps methodology in projects such as security systems for residential complexes, water supply and sewage management systems, as well as city climate control and lighting systems, demonstrates its effectiveness. The DevOps methodology enables easy scaling of information systems to meet the growing needs of cities, which is critically important for infrastructure solutions where the number of connected devices may continuously increase.*

Keywords: *Internet of Things (IoT); development methodology; information systems maintenance; DevOps; Agile; infrastructure IoT systems; security systems; water supply and sewage systems; climate control; lighting management; domestic IT companies; foreign IT companies; process automation; IoT security; scalability*

References

1. Kim, G., Humble, J., Debois, P., Willis, J. & Forsgren, N. (2021). *The DevOps handbook: How to create world-class agility, reliability, & security in technology organizations*. It Revolution.
2. Perry, Y. (2020). DevOps: The Complete Guide to Culture, Technology and Tools, website. URL: <https://bluexp.netapp.com/devops>.
3. Varadharajan, V. & Bansal, S. (2016). Data security and privacy in the internet of things (iot) environment. *Connectivity frameworks for smart devices: the internet of things from a distributed computing perspective*, 261–281. https://doi.org/10.1007/978-3-319-33124-9_11.

4. DevOps and IoT: How to Integrate Them for Success, website. (2024). URL: <https://www.mindbrowser.com/integrating-devops-and-iot/>
5. Breyter, M. (2022). Agile Product and Project Management. Apress.
6. Klein, M. (2024). Waterfall Model Methodology: Everything You Need to Know, website. URL: <https://project-management.com/waterfall-model/>
7. Chantzis, F., Stais, I., Calderon, P., Deirmentzoglou, E. & Woods, B. (2021). Practical IoT hacking: the definitive guide to attacking the internet of things. No Starch Press.
8. A detailed guide to Lean and Agile methodologies, website. (2024). URL: <https://worksection.com/ua/blog/agile-vs-lean.html>
9. Riepina, I. (2023). Using the Lean manufacturing methodology to improve the quality of the enterprise's business processes. *Management*, 37 (1), 39–49.
10. Kovalenko, O. O., Denysiuk, A. V. & Bazhan, V. M. (2023). Comparative analysis of IT product development methodologies. *Bulletin of Kharkiv National University named after V. N. Karazin. Series: Information Technologies*, 25, 45–55.
11. Adeagbo, M. A., Akinsola, J. E. T., Awoseyi, A. A., & Kasali, F. (2021). Project implementation decision using Software Development Life Cycle models: A comparative approach. *Journal of Computer Science and Its Application*, 28(1). <https://doi.org/10.4314/jcsia.v28i1.10>
12. Bravos, G., Cabrera, A. J., Correa, C., Danilović, D., Evangelidou, N., Ezov, G., ... & Vukobratovic, D. (2022). Cybersecurity for industrial Internet of Things: architecture, models and lessons learned. *IEEE Access*, 10, 124747-124765.
13. Fakhrudeen, H. F., Kadhim, K. A., Shyaa, T. A. & Al-Asady, H. A. J. (2023). Models and Methods of Information and Control System Cyber-security for Smart Buildings. *Malaysian Journal of Fundamental and Applied Sciences*, 19 (3), 313–323.

Посилання на публікацію

- APA Zhovnir, Y. I., Hrybovskyi, O. M., Orlov, M. V., Duda, O. M. & Kunanets, N. E. (2024). Methodology of development and maintenance of information systems based on the technology of the internet of things. *Management of Development of Complex Systems*, 60, 56–70, [dx.doi.org\10.32347/2412-9933.2024.60.56-70](https://doi.org/10.32347/2412-9933.2024.60.56-70).
- ДСТУ Жовнір Ю. І., Грибовський О. М., Орлов М. В., Дуда О. М., Кунанець Н. Е. Методологія розроблення та супроводу інформаційних систем, базованих на технології інтернету речей. *Управління розвитком складних систем*. Київ, 2024. № 60. С. 56 – 70, [dx.doi.org\10.32347/2412-9933.2024.60.56-70](https://doi.org/10.32347/2412-9933.2024.60.56-70).