

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МАРІУПОЛЬСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ЕКОНОМІКО-ПРАВОВИЙ ФАКУЛЬТЕТ
КАФЕДРА СИСТЕМНОГО АНАЛІЗУ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

**До захисту допустити:
В.о. зав. кафедри**



Ганна МАРТИНЮК

«04» червня 2025 р.

**СТВОРЕННЯ ВЕБ – ІНТЕРФЕЙСУ ДЛЯ РОБОТИ З МЕРЕЖЕВИМИ
КАТАЛОГАМИ»**

Кваліфікаційна робота
здобувача вищої освіти першого
(бакалаврського) рівня вищої освіти
освітньо-професійної програми
«Комп'ютерні науки»
Акрітова Матвія Вадимовича
Науковий керівник:
Мнацаканян Марія Сергіївна,
кандидат технічних наук,
доцент кафедри системного аналізу та
інформаційних технологій
Рецензент:
Супрун Ольга Миколаївна,
кандидат фізико-математичних наук, доцент,
професор кафедри інтелектуальних
кібернетичних систем Державного
університету «Київського авіаційного
університету»

Кваліфікаційна робота захищена
з оцінкою задовільно 63 (E)
Секретар ЕК



«11» червня 2025 р.

ЗМІСТ

ВСТУП	3
РОЗДІЛ 1. ДОСЛІДЖЕННЯ ТЕХНОЛОГІЇ МЕРЕЖЕВИХ КАТАЛОГІВ.....	5
1.1. Принцип праці мережеских каталогів	5
1.2. Технології, побудовані на основі мережевого каталогу	6
Висновки до першого розділу	7
РОЗДІЛ 2. ОГЛЯД ВЕБ-ІНТЕРФЕЙСІВ ДЛЯ РОБОТИ З МЕРЕЖЕВИМИ КАТАЛОГАМИ.....	9
2.1. phpLDAPadmin	9
2.2. ldap-ui	14
Висновки до розділу 2	18
РОЗДІЛ 3. ОПИС ВИМОГ ДЛЯ СТВОРЕННЯ ВЕБ-ІНТЕРФЕЙСУ ДЛЯ РОБОТИ З МЕРЕЖОВИМИ КАТАЛОГАМИ.....	20
3.1. Функціональні вимоги.....	20
3.2. Вибір технологій.....	21
Висновки до розділу 3	22
ВИСНОВКИ.....	28
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	29

ВСТУП

Актуальність теми. У сучасному світі інформаційних технологій, обсяг інформації, що зберігається, обробляється й передається в локальних мережах і хмарних сервісах, постійно збільшується. Одним із головних питань у цій галузі є ефективне керування мережевими каталогами — структурованими сховищами об'єктів, які доступні в межах якоїсь мережі. Способи роботи з такими каталогами, які були сформувані за велику кількість років і зазвичай передбачають використання додатків або командного рядка, можуть бути незручними, особливо для користувачів без технічної підготовки.

У зв'язку з цим зростає необхідність у створенні інтуїтивно зрозумілих, функціональних та безпечних веб-інтерфейсів, котрі дозволяють здійснювати доступ, перегляд, редагування та керування вмістом мережеских каталогів за допомогою веб-браузера. Такі інтерфейси полегшують роботу користувачів, забезпечують незалежність від платформ та операційних систем та дозволяють інтегрувати додаткові сервіси.

Особливо актуальним це стає в умовах віддаленої роботи, розвитку хмарних сервісів і зростаючих вимог до кібербезпеки. Розробка веб-інтерфейсів для роботи з мережевими каталогами сприяє підвищенню ефективності організації доступу до інформаційних ресурсів, покращенню взаємодії між користувачами та ІТ-інфраструктурою, а також оптимізації внутрішніх бізнес-процесів.

Отже, тема дослідження є актуальною з огляду як технічного розвитку, так і практичного впровадження інформаційних технологій у різних сферах діяльності.

Метою роботи є створення можливості керування і взаємодії мережевого каталогу за допомогою веб – інтерфейсу.

Для досягнення поставленої мети в роботі були поставлені наступні завдання:

- Дослідити технології мережеских каталогів

- Визначити основні поняття
- Провести огляд доступних рішень в Інтернеті
- Сформулювати вимоги для нових рішень
- Розробити прототип згідно з вимогами

РОЗДІЛ 1. ДОСЛІДЖЕННЯ ТЕХНОЛОГІЇ МЕРЕЖЕВИХ КАТАЛОГІВ

1.1. Принцип праці мережесих каталогів

Відповідно до енциклопедії ресурсу PCmag (1), мережесий каталог (network directory) – це система для зберігання інформації у виді структурованого списку ресурсів в мережі про об’єкти, наприклад, користувачів, групи, пристрої, дозволи. Також можна зустріти назву «сервіс каталогів», коли на запит імені, мережа повертає профіль особистості, що може включати в себе дозволи на доступ до даних і інформацію про особистість.

В стандарті RFC 4512 (2) написано, що дана система організована у формі ієрархічної структури (дерева), також названою Directory Information Tree (DIT), де кожен елемент має унікальне ім’я (distinguished name, DN), котрий вказує на позицію елемента в дереві. Кожен елемент обов’язково має мати хоча б один атрибутивного словника (назва атрибуту - значення). Атрибути, які найчастіше зустрічаються, описані в таблиці 1.1.

Таблиця 1.1.

Основні атрибути у DN

Атрибут	Опис	Приклад значення
cn	Common Name (звичайне ім’я)	Тарас Степаненко
ou	Organizational Unit (організаційна одиниця, департамент)	Бухгалтерія
o	Organization (організація, підприємство)	ТОВ «Стара Пошта»

Також кожен елемент в дереві має як мінімум один об'єкт класу (object class), який, власне, визначає тип цього елемента. Список класів визначений в схемі мережевого каталогу.

1.2. Технології, побудовані на основі мережевого каталогу

1.2.1 LDAP (Lightweight Directory Access Protocol)

LDAP — це стандартний мережевий протокол, призначений для доступу до інформації у мережевому каталозі. LDAP підтримує ряд основних операцій: пошук, додавання, оновлення та видалення записів. Він також використовується для аутентифікації та авторизації користувачів — система перевіряє дані у каталозі, щоб підтвердити правомірність вхідного імені та пароля. Це дозволяє створювати єдину систему аутентифікації для різних сервісів, що значно зручніше для адміністрування.

LDAP має широкий спектр застосувань. Він використовується у корпоративних мережах для управління користувачами, групами, дозволами та іншими об'єктами. Також його можна знайти в системах Single Sign-On (SSO), де виконується функція авторизації користувача на різних сервісах за один раз. LDAP дозволяє також зберігати інформацію про електронну пошту, мобільні пристрої та інші дані.

LDAP має наступні переваги:

- Висока ефективність при пошуку інформації.
- Підтримка розподілених систем.

Не дивлячись на описаний функціонал і переваги, LDAP також має недоліки:

- Невисока ефективність при виконанні складних операцій
- Відсутність підтримки транзакцій.

1.2.2 Active Directory (AD)

Active Directory — це система каталогів, створена компанією Microsoft, що використовується для управління користувачами, групами, комп'ютерами

та іншими мережевими об'єктами у середовищі Windows. Це одна з найпоширеніших систем в корпоративному середовищі, забезпечуючи централізоване управління активами та безпекою.

У Active Directory реалізовані основні функції:

- Аутентифікація і авторизація користувачів — перевірка прав облікового запису та можливості надання доступу до ресурсів.
- Керування групами — дозволяє створювати групи для користувачів, що дозволяє спростити керування правами.
- Політики безпеки (GPO) — це механізм, який дозволяє адміністраторам застосовувати конфігурації до користувачів і комп'ютерів.
- Архітектура дерев — можливість створення структурованого розташування об'єктів, що сприяє ефективному управлінню.

Особливостями Active Directory є:

- Висока стабільність і надійність у корпоративних середовищах.
- Підтримка масштабування.
- Глибока інтеграція з Windows-системами.

Однак у нього також є певні недоліки:

- Складна налаштування для початківців.
- Високі вимоги до апаратних ресурсів.

Висновки до першого розділу

У цьому розділі було розглянуто теоретичні відомості про мережеві каталоги, що включає детальний аналіз їх принципу дії, обговорення ключових термінів та понять, а також огляд поточних технологій, які побудовані на основі мережевого каталогу. Було вивчено різні типи мережевих каталогів, серед яких особливого уваги заслуговують LDAP (Lightweight Directory Access Protocol), Active Directory, Kerberos та інші.

Викладені теоретичні основи дозволили краще зрозуміти, як працюють системи каталогів, як вони організовані, що забезпечує їх ефективність у

корпоративному середовищі. Було проаналізовано, як мережеві каталоги здійснюють управління користувачами, групами, правами та іншими об'єктами, а також як вони забезпечують безпеку у мережі. Також розглянуто роль кожної з технологій у сучасних інфраструктурах: наприклад, Active Directory як централізована система управління для Windows-середовищ, LDAP — як стандартний протокол для взаємодії з каталогами

РОЗДІЛ 2. ОГЛЯД ВЕБ-ІНТЕРФЕЙСІВ ДЛЯ РОБОТИ З МЕРЕЖЕВИМИ КАТАЛОГАМИ

Перед тим, як почати розробку веб застосунку для роботи з мережевими каталогами, потрібно зробити огляд існуючих рішень. Це дозволить зрозуміти, які переваги й недоліки має кожне з рішень, і на що варто звернути увагу, при розробці нового рішення.

При огляді не буде розглянуто таких рішень, як, наприклад, Apache Directory Studio, який являється класичними інструментом для керування мережевими каталогами, так як додаток є десктопним, а не веб застосунком.

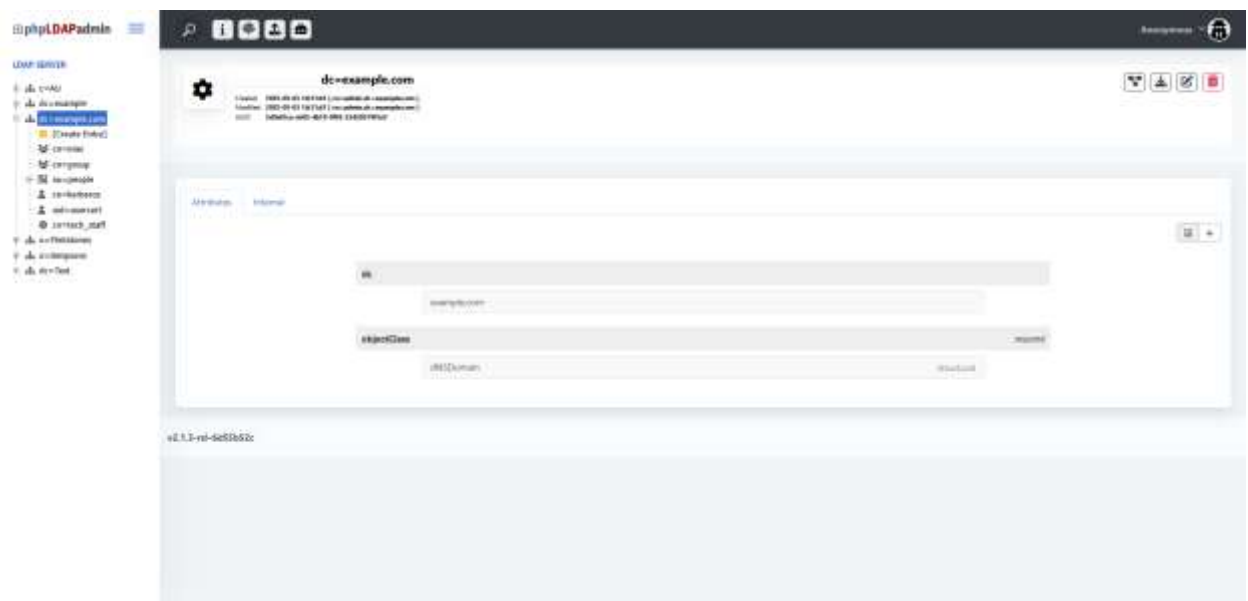
Огляд кожного веб-застосунку є побудованим на наступних критеріях:

1. Функціональність
 - a. Підтримка основних операцій, таких як:
 - i. Зчитування об'єктів
 - ii. Додавання об'єктів
 - iii. Видалення об'єктів
 - iv. Редагування даних
 - v. Пошук
 - vi. Керування групами
 - vii. Керування правами доступу
 - b. Підтримка складних запитів.
2. Інтуїтивність та зручність інтерфейсу
3. Безпека
4. Сумісність та інтеграція
5. Вимоги до розгортання
6. Наявність документації та підтримки
7. Ліцензія та доступність

2.1. phpLDAPadmin

phpLDAPadmin (PLA) — це веб - інтерфейс для адміністрування LDAP-серверів (протоколу Lightweight Directory Access Protocol), створений з використанням мови програмування PHP. (3) Інструмент призначений для спрощення управління структурованими даними, які зберігаються в LDAP-каталогах, та надає адміністраторам інтуїтивно зрозумілий спосіб керувати обліковими записами, групами, організаційними одиницями, атрибутами користувачів та іншими об'єктами каталогу.

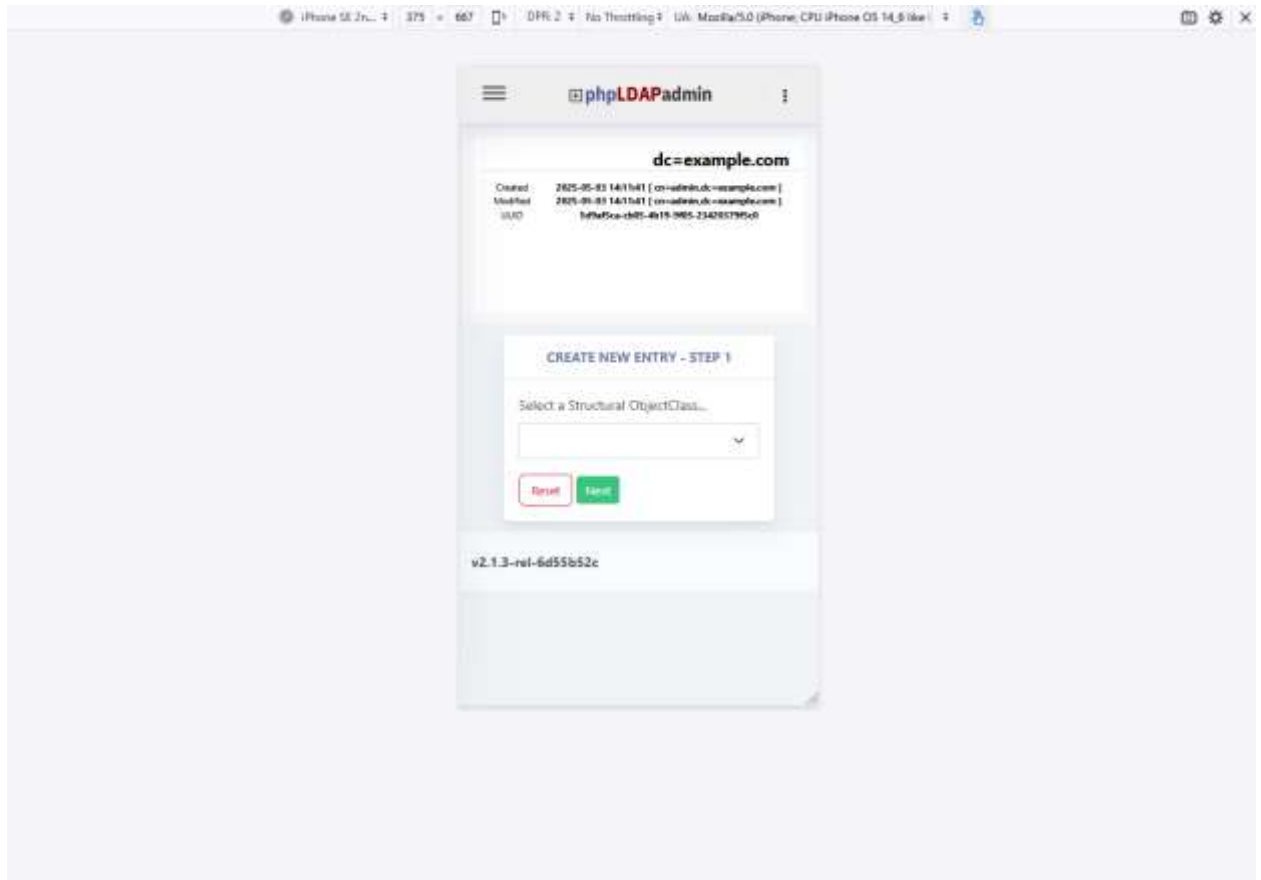
Основна мета phpLDAPadmin — зробити управління LDAP-серверами більш доступним і зручним за допомогою візуального середовища, без необхідності вручну редагувати LDIF-файли або виконувати команди через термінал. Це значно знижує ризик помилок та пришвидшує процес обслуговування каталогів.



Зображення 2.1 – вигляд сторінки phpLDAPadmin.

На зображенні 2.1 можна побачити типове відображення сторінки phpLDAPadmin. Це зображення демонструє функціонал зчитування ієрархічної структури (з лівою частини зображення) та атрибутів об'єктів, котрі є в цьому дереві (основна частина зображення). Дерево можна сховати, і це є стан за замовчанням якщо входити на сайт зі смартфона. Сайт підтримує формат смартфонів (зображення 2.2), але звичайно можна помітити, що десь елементи не повністю вміщуються в div-контейнер, десь навпаки не

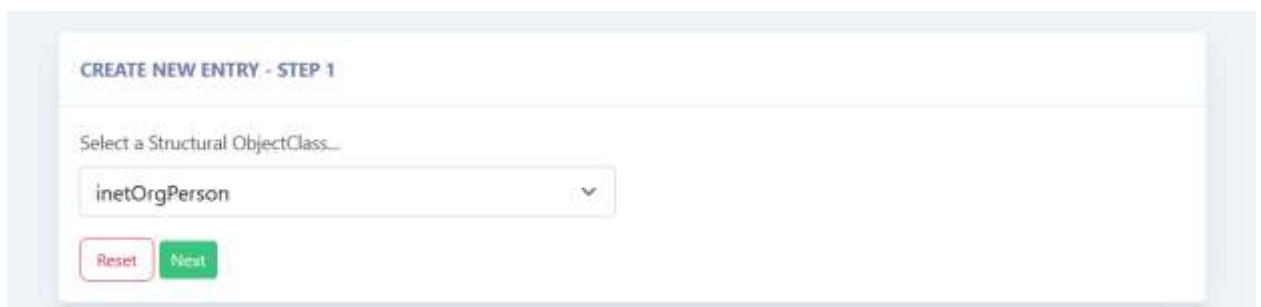
використовують усю ширину сторінки, що може змусити користувачів перейти або на планшети, або на персональні комп'ютери та ноутбуки, що зменшує мобільність, яка інколи так важлива для системних адміністраторів.



Зображення 2.2 – вигляд сайту в мобільному браузері (за допомогою емуляції).

Щоб створити об'єкт в мережевому каталозі в phpLDAPadmin, потрібно заповнити форму, яка складається з декілька кроків.

В першому кроку, який можна побачити на зображенні 2.3, потрібно вибрати батьківський елемент, для котрого буде створена нове значення, та обрати значення object class зі списку.



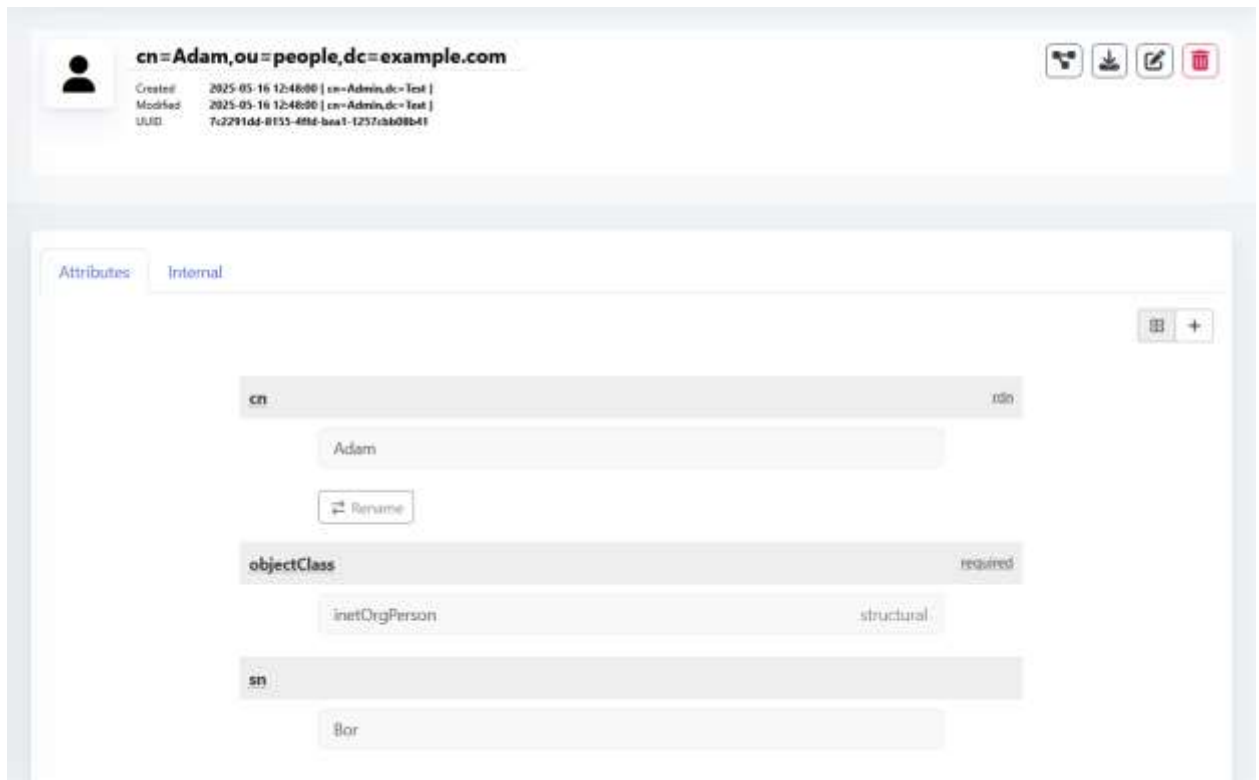
Зображення 2.3 – перший крок при створенні нового елементу, обирання значення ObjectClass

В другому кроку (зображення 2.4) потрібно визначити значення атрибутів, які є за замовчанням, а так, опціонально, можна додати додаткові object classes та атрибути.

Зображення 2.4 – ініціалізація атрибутів.

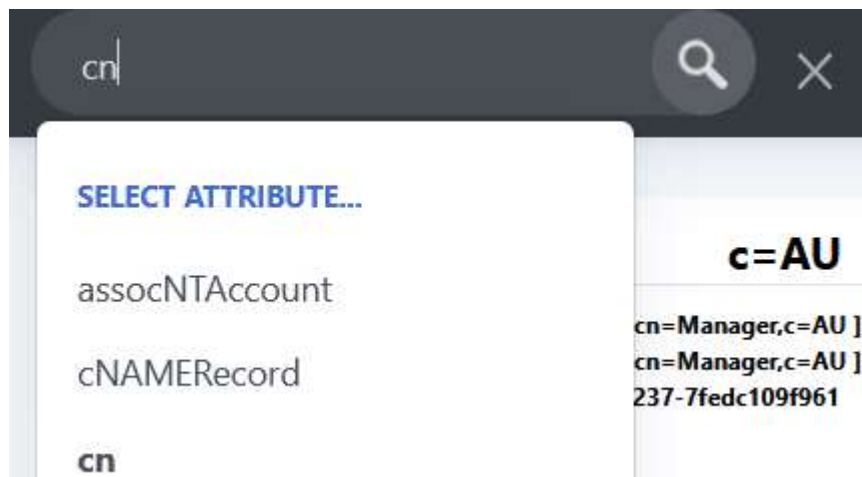
При спробі додавання фотографії за допомогою атрибуту jpegPhoto (дозволяє додати фотографію до користувача), phpLDAPadmin не відповідає на натискання кнопки і ніякі повідомлення не відображаються.

Після заповнення форми, веб-додаток перенаправляє користувача на створений елемент (зображення 2.5), де можна поміняти значення атрибутів (як стандартних, так і користувацьких), або подивитись більш детальнішу інформацію (наприклад, коли саме цей елемент був створений).



Зображення 2.5 – створений елемент

phpLDAPadmin присутній функціонал пошуку. Пошук розташований в голові сайту, в лівій частині. Пошук відбувається за вибором атрибуту та подання значення цього атрибуту (зображення 2.6). Потрібно зауважити, що у веб – додатку відсутня підтримка складних фільтрів пошуку (наприклад, (&(objectClass=person)((cn=*Bamm*))).



Зображення 2.6 – пошук в phpLDAPadmin

При успішному пошуку, буде відображено один або декілька результатів пошуку. При обранні одного з елементів, користувач буде направлений на цей елемент, а в провіднику буде показано розташування цього елементу в дереві.

На жаль, додаток не має підтримки інших мов, що ускладнює користуванням ним користувачами, які не знають англійську мову.

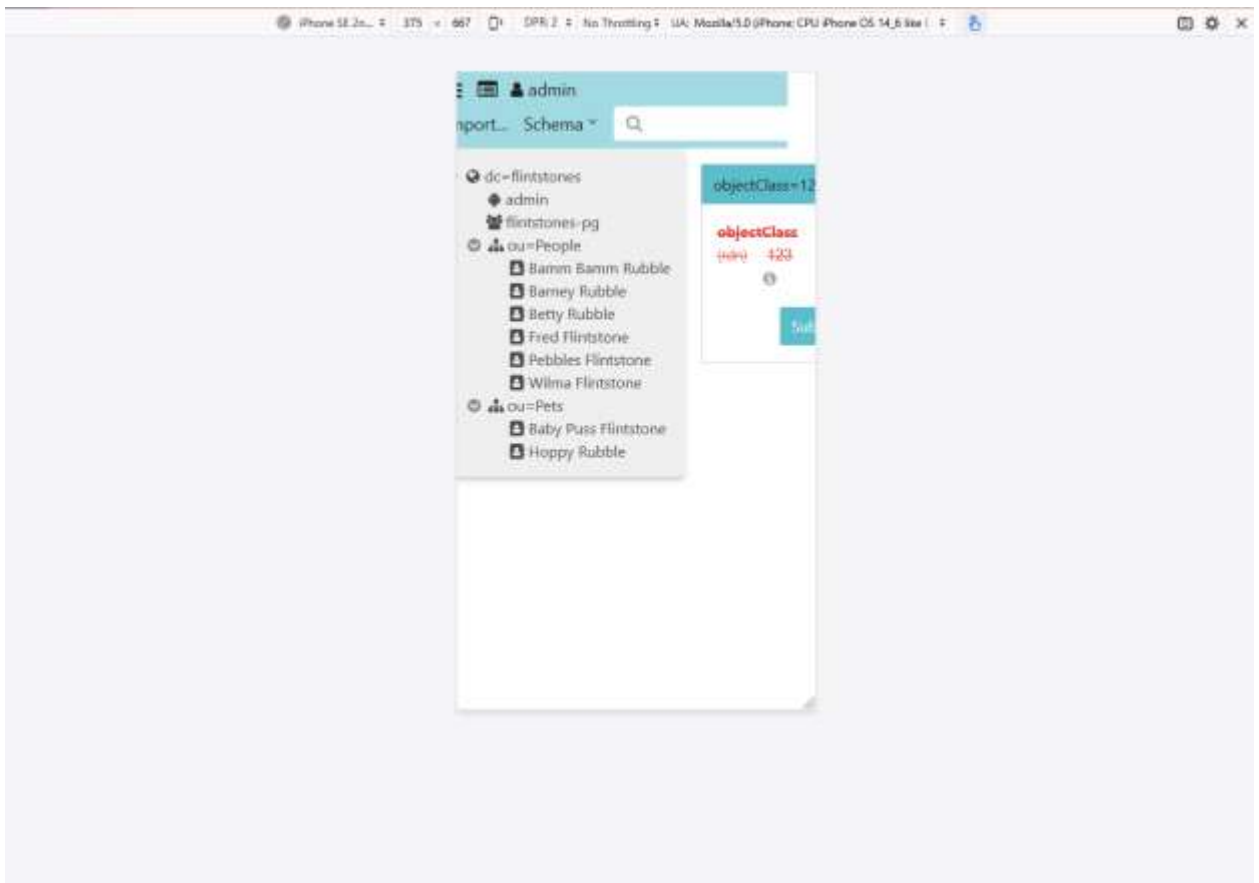
2.2. ldap-ui

ldap-ui – це інший веб -додаток, який побудований з використанням Python в якості серверної частини та Vue.js фреймворку для клієнтської частини. Як і phpLDAPadmin, він був розроблений для адміністрування мережевих каталогів, на момент 2025 року, офіційно підтримується тільки OpenLDAP, хоча і не виключається частична підтримка усіх LDAP рішень, які можуть працювати на 389 порті.



Зображення 2.7 – вигляд сторінки ldap-ui

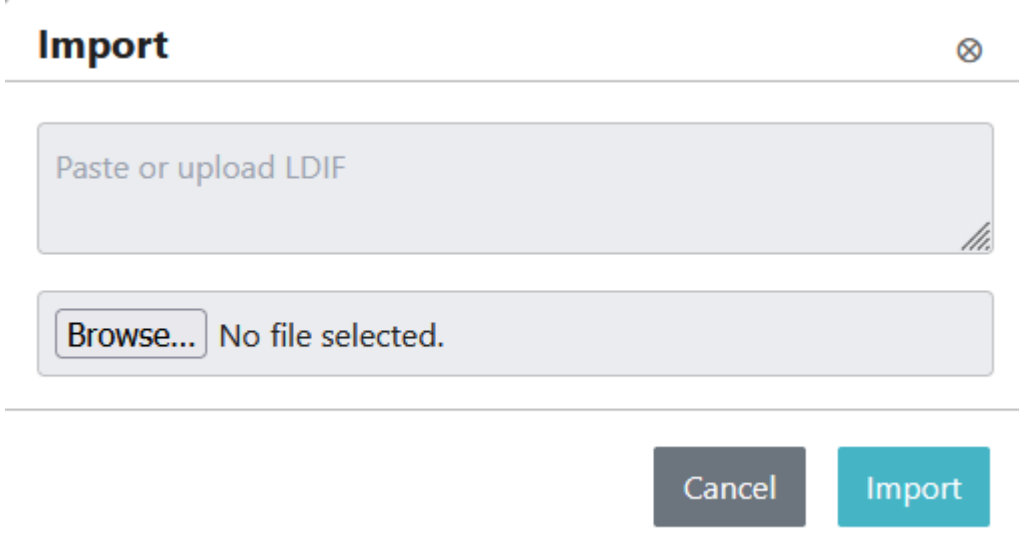
На зображенні 2.7 можна побачити типове відображення сторінки ldap-ui, яке має подібний вигляд, як і phpLDAPadmin. Це зображення демонструє функціонал зчитування ієрархічної структури (з лівою частини зображення) та атрибутів об'єктів, котрі є в цьому дереві (основна частина зображення). Сайт не підтримує формат мобільних пристроїв (зображення 2.8), так як схоже на те, що під час розробки увага приділялась тільки персональним комп'ютерам. При відкриття дерева на мобільному пристрої, форма, в котрих можна побачити дані, сильно зміщується вправо, тим самим, виходячи за кордони фізичного екрану і примушує користувачів робити додаткові дії.



Зображення 2.8 – відображення додатку на мобільних пристроях.

У веб – додатку, присутня можливість імпортувати існуючі записи у форматі .LDIF до підключеного мережевого каталогу. Щоб зробити це, у шапці сайту потрібно натиснути на підписану кнопку «Import...» та обрати .LDIF файл, який потрібно завантажити (зображення 2.9). На жаль, під час тестування цього функціоналу, нічого не відбувалось, що вказує на те, що відбувається десь помилка (на стороні клієнту або сервера), яка не показується звичайним користувачам.

Також присутня підтримка експорту, яка дозволяє вивантажити усі записи з мережевого каталогу на локальний пристрій.



Import ⊗

Paste or upload LDIF

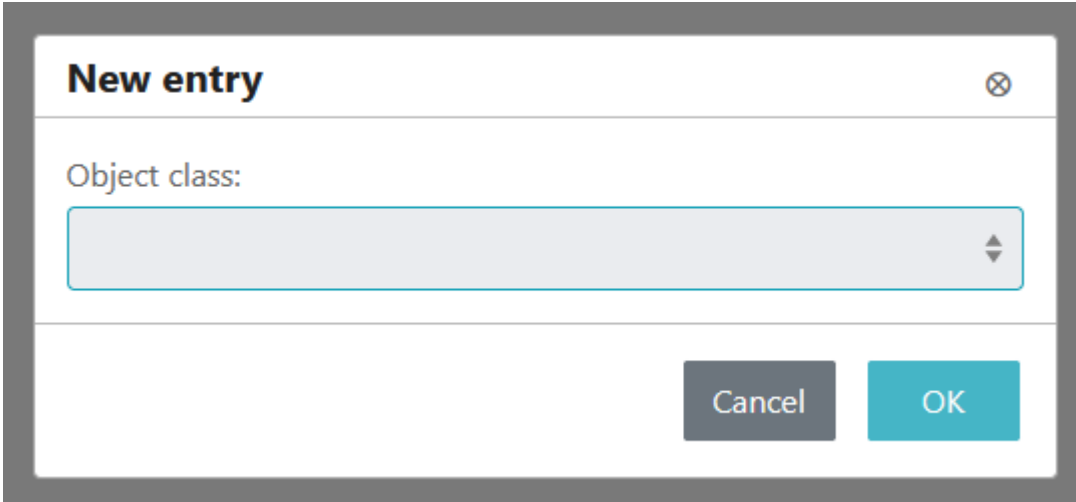
Browse... No file selected.

Cancel Import

Зображення 2.9 – форма для завантаження існуючих об'єктів.

Щоб створити новий об'єкт в каталозі, потрібно натиснути на батьківський елемент (в котрому буде створення), після цього натиснути на назву цього елемента і обрати «Add child...».

Після цього, з'явиться перша форма в модальному вікні (зображення 2.10), в якому потрібно обрати необхідний тип об'єкту.



New entry ⊗

Object class:

Cancel OK

Зображення 2.10 – обирання типу об'єкта.

Після обрання об'єкту, появляється друга форма в модальному вікні (зображення 2.11), в якому потрібно заповнити значення атрибутів.

Зображення 2.11 – заповнення значення атрибутів об'єкта

Після успішного створення об'єкту, він буде автоматичний відкритий у веб – додатку, а також буде показано його розташування у дереві.

У веб – додатку, є можливість подивитись усі існуючі типи об'єктів та які атрибути вони мають мати і можуть мати. Це можна зробити, натиснувши на підписану кнопку «Schema...», і обравши необхідний атрибут. Після цього, інформація про необхідний атрибут з'явиться під HTML елементом, який містить в собі ієрархічне дерево (зображення 2.12)

Зображення 2.12 – відомості про обраний тип об'єкту

Щоб видалити об'єкт, потрібно знайти його в дереві, натиснути на нього, натиснути на ім'я об'єкту та обрати опцію «Delete...». Після підтвердження, з'явиться повідомлення, що запис був успішно видалений (зображення 2.13).

Deleted: cn=Matviy,ou=People,dc=flintstones,dc=com

Зображення 2.13 – повідомлення про успішне видалення запису.

ldap-ui дозволяє проводити пошуки елементів в ієрархічному дереві (зображення 2.14). Підтримується як простий пошук (який намагається знайти усі відповідності у distinguished name), так і складний (наприклад, (&(objectClass=person)((cn=*Bamm*))).

class=person)((cn=*Bamm

Зображення 2.14 – вигляд форми пошуку.

Якщо результат пошуку успішний, користувач буде використана автоматична навігація до елемента в дереві. Якщо результат пошуку нічого не дав, користувач отримає відповідне повідомлення (зображення 2.15).

No search results

Зображення 2.15 – повідомлення про невдалий пошук

Висновки до розділу 2

У ході дослідження було здійснено порівняльний аналіз двох open-source веб-інтерфейсів для адміністрування мережевих каталогів — «phpLDAPadmin» та «ldap-ui». Оцінювання проводилося на основі десяти критеріїв: функціональність, інтуїтивність інтерфейсу, безпека, сумісність та інтеграційні можливості, вимоги до встановлення, документаційне забезпечення, ліцензія, масштабованість, адаптивність до мобільних пристроїв, а також підтримка локалізації.

Обидва веб - застосунки забезпечують повноцінну роботу з базовими об'єктами каталогу: створення, редагування, видалення та перегляд LDAP-записів. Підтримується навігація по ієрархічному дереву, пошук за атрибутами та обмін даними у форматі LDIF. В обох реалізовано базову підтримку безпечного з'єднання за допомогою SSL/TLS (при відповідному налаштуванні

веб-сервера). Суттєвою перевагою є наявність відкритих ліцензій (GPL у «phpLDAPadmin» та MIT у «ldap-ui»), що дозволяє використовувати, модифікувати та інтегрувати ці рішення без необхідності придбання ліцензії на них.

Попри наявність таких переваг, обидва додатки мають наступні недоліки:

- Відсутність мобільної адаптації: інтерфейси не є адаптивними, що ускладнює використання на мобільних пристроях;
- Проблеми з локалізацією: phpLDAPadmin та ldap-ui мають лише англomовний інтерфейс;
- Обмежені можливості інтеграції з іншими сервісами (наприклад, немає підтримки SSO, Kerberos або зовнішніх API);
- Сумнівна масштабованість: phpLDAPadmin демонструє зниження продуктивності при роботі з великими базами, тоді як ldap-ui хоч і використовує асинхронний бекенд, однак не має вбудованих механізмів кешування чи оптимізації запитів;
- Інтерфейсна застарілість: phpLDAPadmin має класичний, морально застарілий дизайн, тоді як ldap-ui характеризується надмірним мінімалізмом. Обидва є потенційно непридатні для користувачів без попереднього досвіду роботи з LDAP.

Таким чином, на основі отриманих під час аналізу веб - додатків позитивних та негативних результатів, можна почати формувати вимоги до нового веб – додатку.

РОЗДІЛ 3. ОПИС ВИМОГ ДЛЯ СТВОРЕННЯ ВЕБ-ІНТЕРФЕЙСУ ДЛЯ РОБОТИ З МЕРЕЖОВИМИ КАТАЛОГАМИ

Розробка ефективного веб-інтерфейсу для роботи з мережевими каталогами є важливим етапом у забезпеченні доступу до структурованої інформації. На основі проведеного аналізу аналогічних систем та загального опису функціональних особливостей мережеских каталогів були сформульовані вимоги до створення такого веб – додатку.

3.1. Функціональні вимоги

Для того, щоб відповідати сучасним вимогам веб – додаткам, що працюють з мережевими каталогами, проект має реалізовувати набір основних функцій.

Без підтримки запитів до LDAP – сервера немає сенсу робити іншій функціонал. Підтримка запитів потрібна для вибірок усіх доступних записів в каталозі. Це включає в себе користувачів, групи, організації і так далі. Імплементация цього забезпечить можливість мати початку синхронізацію даних і закладе основу для подальшої побудови візуального представлення каталогу.

На основі отриманих результатів з запитів, веб – інтерфейс має візуально формувати ієрархічну структуру, де кожен об’єкт має унікальний Distinguished Name. В цьому функціоналі, важливо мати реалізацію підходу лінивого завантаження (lazy loading) дочірніх елементів, так як це дозволяє зекономити ресурси та пришвидшити початкову ініціалізацію веб – додатку.

Завдяки наявності побудованого структурованого дерева, користувачі мають мати можливість:

- Створювати нові записи з вибором необхідними даними і атрибутів, включаючи підтримку обов’язкових та додаткових полів.
- Редагувати атрибути існуючих об’єктів із перевіркою введених даних згідно з вимогами схеми ієрархічного дерева.

- Видаляти непотрібні записи з підтвердженням дії та обробкою залежностей.

Інтегрований пошук має підтримувати як простий пошук за ключовими атрибутами, так і побудову складних фільтрів, згідно зі специфікаціями LDAP (наприклад, (&(objectClass=person)((cn=*Іван*)(sn=*Петренко*))). Модуль, в залежності від прав користувача та налаштувань, може мати можливість зберігати історію пошуку. Також, в сучасному світі, потрібно мати розроблене автодоповнення, на основі введених даних та перехід до знайденого елемента у дереві.

Для того, щоб мати можливість гнучкого налаштування веб – додатку, необхідно мати підтримку використання конфігураційних файлів, або зчитування змін середовища, або зчитування аргументів для додатка.

Усі операції, вдалі або невдалі (помилка при з'єднанні з сервером, неправильно заповнена форма, тощо) мають супроводжуватися повідомленнями на рівні інтерфейсу, а також мають бути записані у файл з логами. Це дозволить швидше виявляти та усувати проблеми як під час адміністрування каталогу, так і під час покращення додатку та усунення помилок в програмному коді.

3.2. Вибір технологій

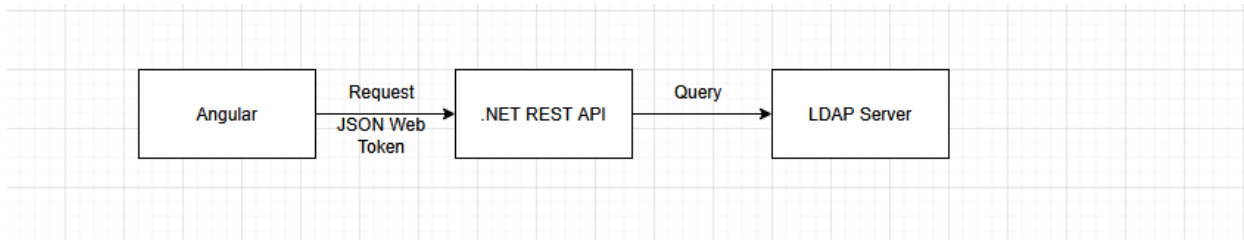
Для того, щоб реалізувати описані вище вимоги, було вирішено використовувати наступний набір технологій.

ASP.NET в якості веб - серверу буде виступати в якості моста між користувачем та мережевим каталогом. Саме веб – проект .NET, розроблений Microsoft, буде обробляти усі запити, зроблені користувачами за допомогою клієнту, та відправляти запити до LDAP. Після отримання результатів, сервер буде опрацювати результати та повертати необхідні дані клієнту. Також, технології ASP.NET дозволяють дуже просто працювати як з файлами конфігурацій (за допомогою appsettings.json), так і зчитувати необхідні змінні середовища та аргументи.

В якості клієнту буде використаний JavaScript фреймворк від Google під назвою Angular. Даний фреймворк має можливість використовувати типі для змінних за допомогою використання TypeScript в проєкті за замовчанням. Також, Angular відомий тим, що дає можливість розробки проєктів зі строгою структуризацією, що дозволяє простіше підтримувати розробку та якість коду. Також, фреймворк підтримує безліч функціональних бібліотек та бібліотек інтерфейсу з різними стилями (Material UI, Fluent Design, тощо), що дозволяє розробити унікальний дизайн, який дозволить користувачам простіше користуватися даним проєктом.

3.3 Проектування архітектури додатку

Архітектура веб-додатку для праці з мережевими каталогами базується на мікросервісній архітектурі, яка забезпечує гнучкість, масштабованість та ефективність системи (зображення 3.1).



Зображення 3.1 – архітектура додатку

Цей підхід полягає у розподілі функціональності додатку на незалежні, спеціалізовані сервіси, кожен з яких виконує конкретну задачу (наприклад, обробка запитів до каталогів). Такий підхід дозволяє забезпечити:

- Гнучкість: кожен сервіс може бути розроблений, випущений та оновлений незалежно, що дозволяє адаптувати систему до змін у потребах користувачів чи технологіях.
- Масштабованість: сервіси можуть масштабуватися окремо, залежно від навантаження (наприклад, підвищення продуктивності обробки запитів до каталогів під час пікових годин).

- Ефективність: кожен сервіс використовує стільки ресурсів, скільки необхідно, що зменшує витрати на інфраструктуру.

3.4 Дизайн інтерфейсу для користувачів

Кожен новий проект має базуватися на основі інтуїтивного та адаптивного дизайну, який дозволяє веб сторінка та веб застосункам виглядати гарно на усіх можливих девайсах. (4) Хоча і стверджується, що адаптивний дизайн використовує тільки HTML та CSS, але під час вирішення задач у реальному світі, JavaScript часто застосовується, щоб виконати нестандартні вимоги до адаптивності.

Створення прототипу є важливим етапом для кожного додатку, так як дозволяє побачити приблизний результат, а також, у разі чого, внести необхідні зміни у разі нових вимог. Основні частини прототипу на даний момент включають в себе:

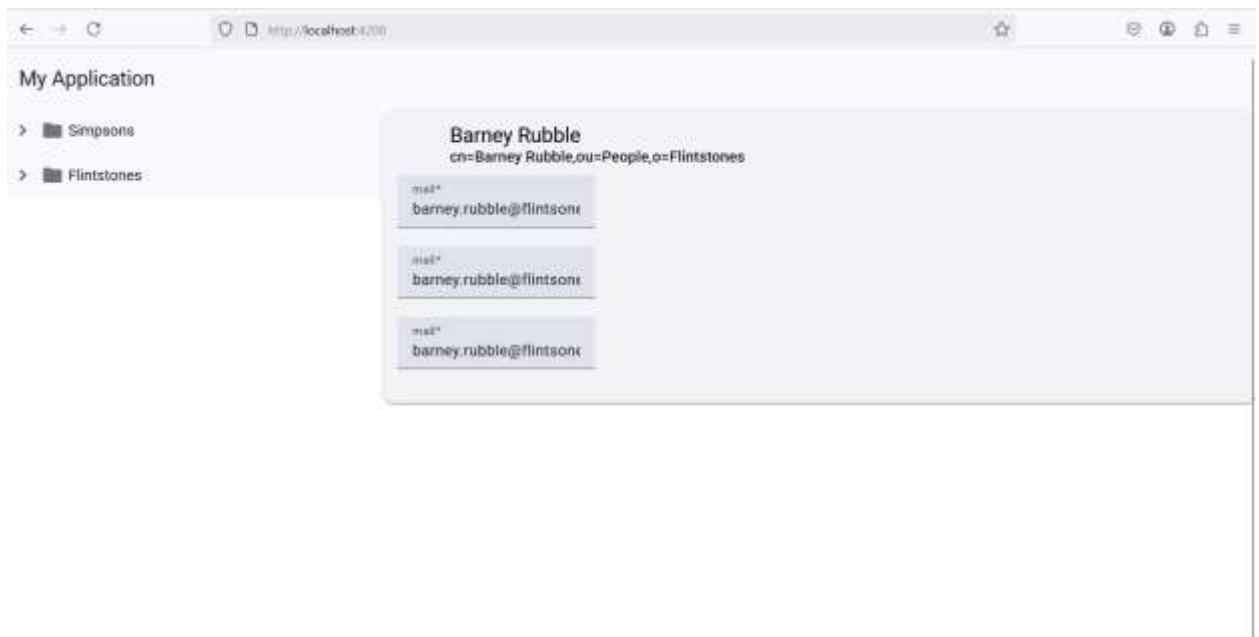
Головна сторінка для великих екранів: сторінка, де користувачі можуть робити навігацію по дереву мережевого каталогу, дивитися на дані об'єктів, вносити зміни в них та видаляти їх. Навігація по дереву має знаходитись з лівої сторони, а форма даних про об'єкти має займати все залишене місце праворуч. (зображення 3.2). Шапка сторінки буде дозволяти виконувати пошукові операції, а також відображення модальних вікон з формами створення нових об'єктів. Головна вимога для відображення цієї версії сторінки: мати ширину та висоту екрана мінімум 768 пікселів.

Головна сторінка для маленьких екранів: для того, щоб відповідати адаптивному дизайну, прототип мусить мати другу версію головної сторінки, яка має функціональність першої, але абсолютних інший вигляд (зображення 3.3). Хоча шапка сторінки і залишиться незмінною, але дерево буде займати усю сторінку, для зручності навігації на маленьких пристроях (наприклад, мобільний телефон). Елемент в дереві має дві функції: перехід на дитячий елемент (при натисканні на батьківський елемент) або зміна цього елемента (за допомогою натискання відповідної кнопки поруч). При зміні цього

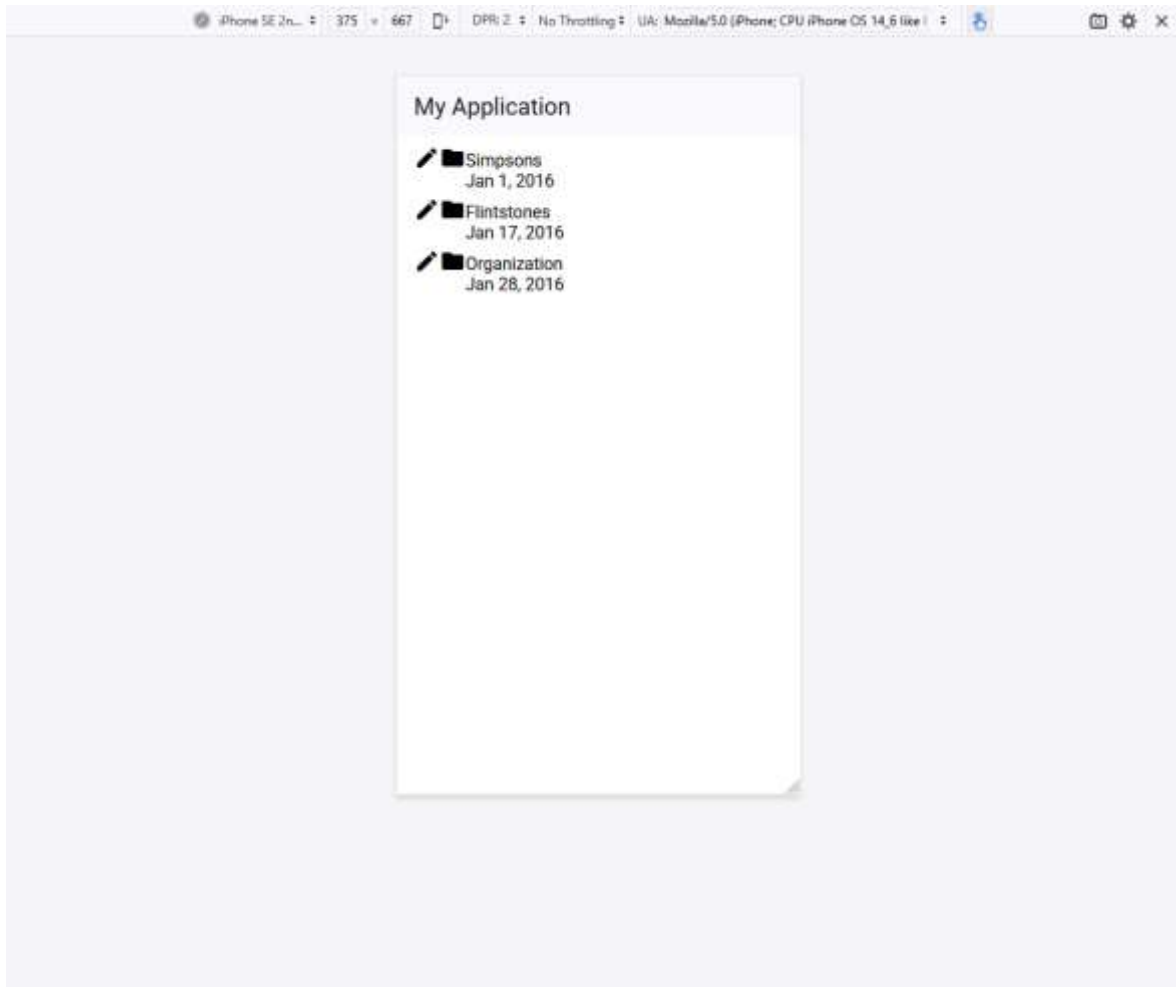
елементу, користувач буде переведений на нову сторінку, де можна подивитись значення атрибут та виконати необхідні операції (зображення 3.4)

Сторінка аутентифікації: дана сторінка потрібна для того, щоб захистити мережевий каталог від несанкціонованих доступу та змін від непривілейованих користувачів. (зображення 3.5)

На даному етапі створення веб – застосунку, не має необхідності мати високий пріоритет створювання кольорової палітри для прототипу, обирати шрифт та аналізувати якусь статистику, так як веб – застосунок використовується виключно вузьким колом спеціалістів, які будуть адмініструвати LDAP сервера та можна робити тільки тоді, коли не буде важливих додаткових вимог до застосунку або помилок на виправлення.



Зображення 3.2 – вигляд головної сторінки для великих екранів.



Зображення 3.3 – вигляд дерева в мережевому каталозі при користуванні смартфоном.

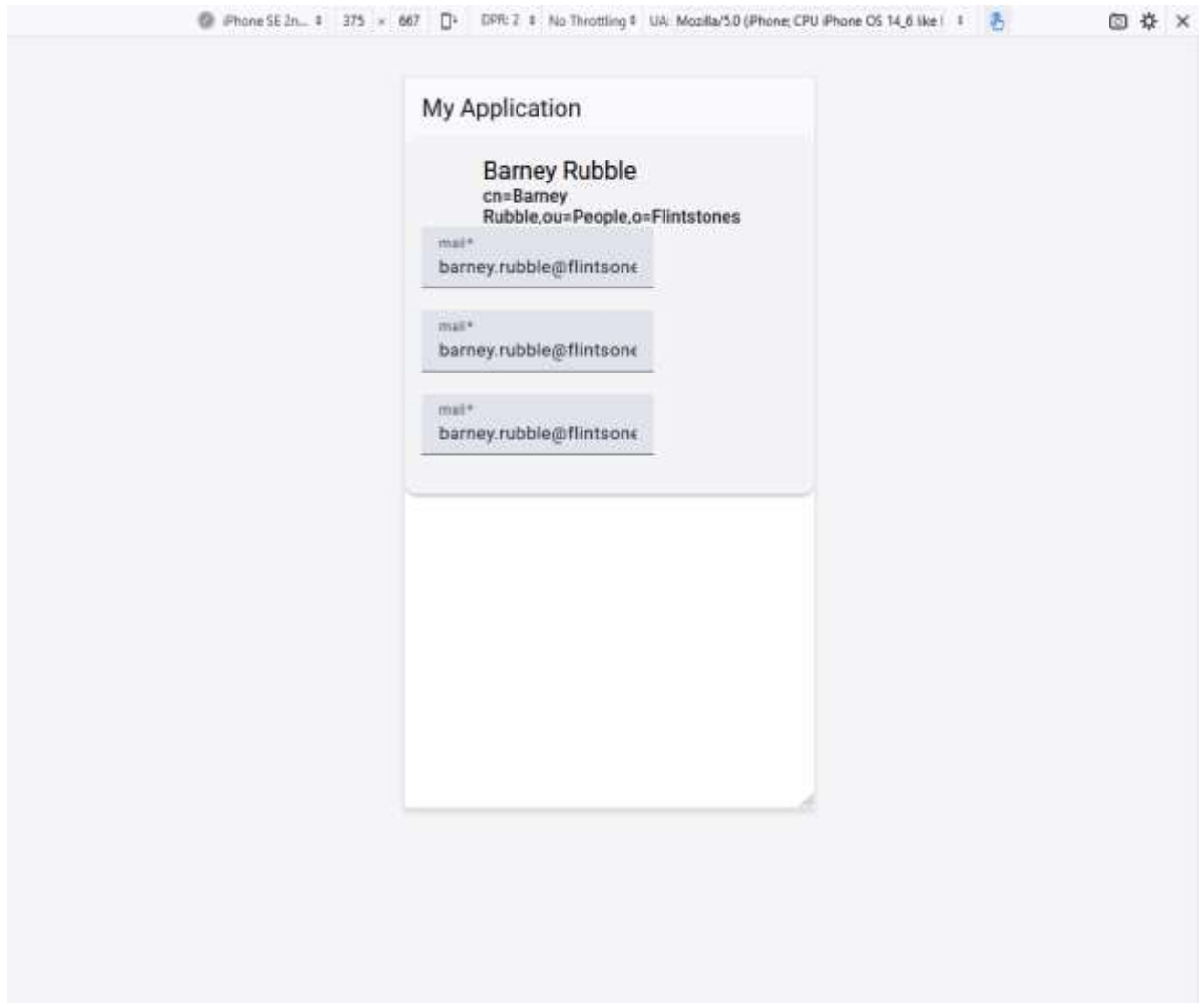
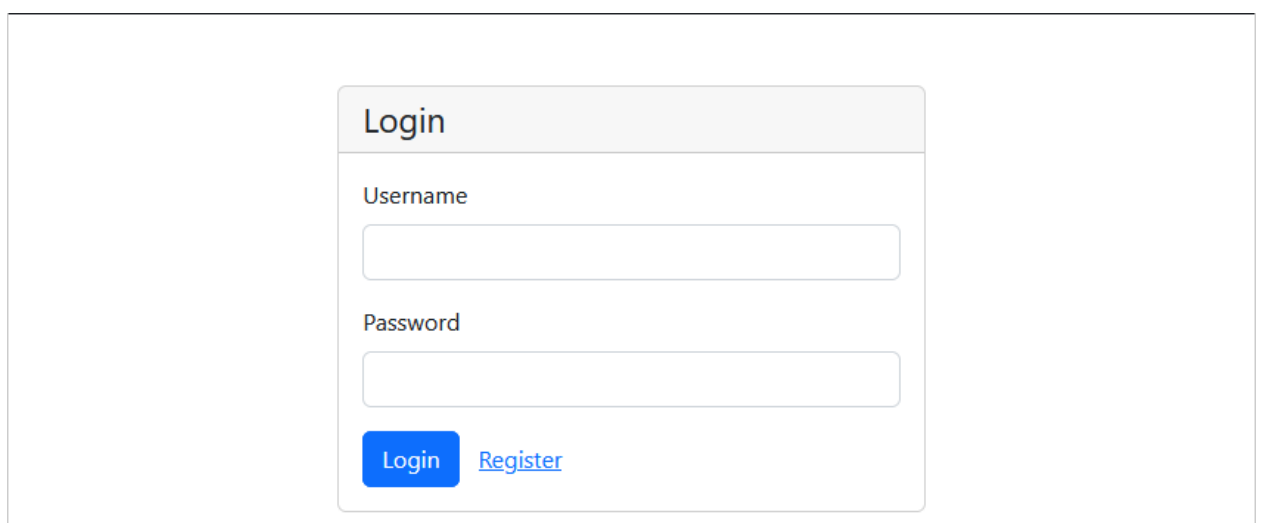


Рисунок 3.4 – вигляд об’єкту дерева при перегляді зі смартфона.



Зображення 3.5 – вигляд сторінки аутентифікації.

Висновки до розділу 3

У підсумку проведеного аналізу та формалізації вимог до веб-інтерфейсу для роботи з мережевими каталогами можна констатувати таке. По-перше, ключовою передумовою ефективного функціонування системи є повноцінна підтримка LDAP-запитів для вибірки всіх записів каталогу, що створює основу для синхронізації даних та їх візуального відтворення у вигляді ієрархічного DIT-дерева з механізмом lazy loading. По-друге, реалізація повного циклу CRUD-операцій (створення, редагування та видалення записів) за допомогою чіткого контролю атрибутів і логіки залежностей дозволяє гарантувати коректність та цілісність схеми каталогу. По-третє, розширений пошуковий модуль із підтримкою складних LDAP-фільтрів, автодоповненням та збереженням історії запитів значно спрощує навігацію й підвищує продуктивність адміністративних операцій.

Не менш важливою є здатність системи до гнучкого налаштування — через конфігураційні файли, змінні середовища чи аргументи запуску — що забезпечує її адаптивність до різноманітних середовищ експлуатації без необхідності перекомпіляції. Адаптовані повідомлення про помилки, як в інтерфейсі, так і в логах, сприятимуть оперативному виявленню та усуненню проблем як у продуктивному, так і в процесі розробки.

Обраний стек технологій ASP.NET Core (бекенд) та Angular (фронтенд) відповідає сформованим вимогам: перший забезпечує надійний міст між клієнтом та LDAP-сервером із зручним робочим середовищем для конфігурації й безпеки, а другий — гнучкий, типізований та структурований підхід до реалізації складного інтерфейсу користувача. Така комбінація дозволить створити масштабовану, безпечну та зручну у підтримці систему для адміністрування мережеских каталогів, що відповідає сучасним стандартам розробки та потребам кінцевих користувачів.

ВИСНОВКИ

В результаті проведеної роботи було досягнуто основної мети: створити веб інтерфейс для керування і взаємодії мережевого каталогу.

Під час розробки було вивчено як працюють мережеві каталоги, визначені їх термінології та основні атрибути.

Також, був проведений аналіз двох веб – додатків, що дозволяють працювати з мережевими каталогами: phpLDAPadmin та ldap-ui. Аналіз показав, що хоча ці додатки і мають основний функціонал, вони вимагають значних знань та досвіду праці з мережевими каталогами.

На додачу до цього, були сформувані основні вимоги до веб – додатків, що працюють з мережевими каталогами та визначений набір технологій, що дозволяють відповідати даними вимогам.

Відповідно до цих вимог, була спроектована мікросервісна архітектура та був показаний прототип, який відповідає до цих вимог.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Американський комп'ютерний журнал PCMag. Network Directory. *Веб-сайт журналу PCMag*. [Онлайновий] [Цитовано: 1 Квітень 2025 р.] <https://www.pcmag.com/encyclopedia/term/network-directory>.
2. Зейленга, Курт. Lightweight Directory Access Protocol (LDAP): Directory Information Models. *RFC Editor*. [Онлайновий] 1 Червень 2006 р. [Цитовано: 2 Квітень 2025 р.] <https://www.rfc-editor.org/rfc/rfc4512>. RFC.
3. George, Deon. Wiki page phpLDAPadmin. *GitHub*. [Online] 27 Квітень 2025. [Cited: 1 Травень 2025.] <https://github.com/leenooks/phpLDAPadmin/wiki>.
4. Компанія W3Schools Network AS. Responsive Web Design - Introduction. *W3Schools*. [Online] [Cited: Травень 28, 2025.] https://www.w3schools.com/Css/css_rwd_intro.asp.