**Вступ**

1. **ТЕОРЕТИЧНА ЧАСТИНА**
   1. **Формування вимог до контрольного продукту**

Розробити програмну систему, що забезпечує обмін повідомленнями та створення оголошень для персоналу підприємства на основі клієнт-серверної архітектури:

Основним завданням було реалізувати:

* реєстрацію нового користувача;
* авторизацію користувача;
* створення бесіди;
* додавання учасників до бесіди;
* вихід з бесіди;
* відправлення повідомлень у бесіду;
* прикріплення файлів до повідомлення;
* перегляд списку користувачів у бесіді;
* перегляд списку друзів;
* пошук користувачів;
* створення, видалення та редагування оголошень;
* редагування персональної інформації;
* перегляд персональної інформації користувачів;
* використання смайлів;
* завантаження картинки профілю;
* роботу серверу з базою даних;
* захист користувача за допомогою токенів доступу та шифрування паролю;
* адміністрування системи за допомогою окремого додатку з можливістю генерації звітів;
* створення зручного та простого інтерфейсу для користувача;
* ведення журналу подій на сервері.
  1. **Огляд аналогічних додатків**

На даний момент існує багато додатків, які дозволяють підтримувати зв’язок між користувачами шляхом відправки повідомлень. В ході роботи вони були проаналізовані, визначено їхні основні позитивні та негативні сторони, і на основі цього сформовано вимоги до кінцевого продукту. Одними з таких додатків є Skype та Telegram.

Додаток Skype являє собою безплатне програмне забезпечення з закритим кодом, яке забезпечує текстовий, голосовий та відеозв’язок через інтернет. Перша версія додатку з’явилась в 2003 році і цей продукт досі залишається популярним як серед компаній, так і серед звичайних користувачів для підтримки зв’язку через інтернет. Перевагами програми є: висока надійність, яка вдосконалювалась роками, голосовий та відеозв’язок, відправка файлів, групові дзвінки, демонстрація екрану.

Також додаток Skype передбачає забезпечення безпеки та захисту даних. Доступність розшифрованих даних дозволена власнику мережі (Microsoft), працівникам Microsoft, а також провайдерам мережі інтернет.

Telegram – це програмне забезпечення для смартфонів та інших пристроїв, яке дозволяє обмінюватись текстовими повідомленнями та різноманітними файлами, зокрема графічними та відеофайлами, а також безкоштовно телефонувати іншим користувачам програми. Перша версія клієнтського додатку вийшла 14 серпня 2013 року.

Для цього продукту було створено протокол MTProto, що передбачає використання декількох протоколів шифрування. Доступна функція створення чату в захищеному від перехоплень повідомлень режимі. При авторизації та аутентифікації використовуються алгоритми RSA-2048, DH-2048 для шифрування, а також при передачі повідомлень використовується шифрування з секретним ключем.

Розмір завантажуваних файлів обмежений 1.5 Гб. Максимальна кількість учасників в одному чаті становить 5000 користувачів. За допомогою прикладного програмного інтерфейсу від розробників доступна можливість створення профілю, керованого програмним кодом.

* 1. **Мова програмування Java**

Java – об’єктно-орієнтована мова програмування, заснована в 1995 році компанією «Sun Microsystems» як основний компонент платформи Java. з 2009 року компанія належить «Oracle».

Дана мова багато запозичила з мов С та С++. Зокрема взято об’єктну модель С++ та модифіковано її. Усунуто можливість появи деяких конфліктних ситуацій, що могли виникнути через помилки програміста та полегшено сам процес розробки об’єктно-орієнтованих програм. Ряд дій, таких як керування пам’яттю, доручено віртуальній машині.

При розробці мови Java керувались такими принципами, як: простий, об’єктно-орієнтований синтаксис, безвідмовна та безпечна реалізація, збереження незалежності від архітектури та переносимість, висока продуктивність виконання.

На сьогоднішній день Java займає лідерські позиції в рейтингах. Її використовують для побудови великих систем для підприємств завдяки її надійності та захищеності.

Програми, написані на мові Java, виконуються віртуальною машиною, що призводить до зниження швидкодії. Проте розробники віртуальної машини постійно вдосконалюють її, тож на даний момент швидкість практично не поступається аналогам на С++. Іншою проблемою є те, що для низькорівневого програмування необхідно використовувати модулі, написані на інших мовах програмування.

* 1. **Платформа JavaFX**

JavaFX – платформа та набір інструментів для створення насичених інтернет-додатків. Вперше продемонстровано Sun Microsystems на Міжнародній конференції Java-розробників JavaOne 2017 року. JavaFX включає в себе набір утиліт, за допомогою яких веб-розробники можуть швидко створювати інтернет-додатки для настільних комп’ютерів, мобільних пристроїв, телебачення та інших платформ.

Платформа складається з JavaFX Script і JavaFX Mobile. Починаючи з випуску JavaFX 2.0 забезпечено можливість створення JavaFX-додатків, написаних цілком мовою Java.

* 1. **Середовище розробки Eclipse**

Eclipse – вільне модульне інтегроване середовище розробки програмного забезпечення. Розробляється і підтримується Eclipse Foundation і включає проекти, такі як платформа Eclipse, набір інструментів для розробників на мові Java, засоби для управління кодом, візуальні системи для графічного інтерфейсу та інші.

Спочатку проект розроблявся в IBM як корпоративний стандарт інтегрованого середовища розробки на багатьох мовах. Потім проект було перейменовано на Eclipse і надано для подальшого розвитку спільноті.

Насамперед це повноцінне середовище, націлене на групову розробку, має засоби роботи з системами контролю версій. З огляду на безкоштовність, у багатьох організаціях є корпоративним стандартом для розробки програмного забезпечення на Java.

Eclipse написаний на Java, тому є платформо-незалежним продуктом, крім бібліотеки графічного інтерфейсу SWT, яка розробляється окремо для більшості поширених платформ. Бібліотека використовує графічні засоби платформи, що забезпечує швидкість і звичний зовнішній вигляд інтерфейсу користувача.

Для середовища існує цілий ряд вільних і комерційних модулів. Спочатку середовище було розроблене для мови Java, але сьогодні існують числення розширення для підтримки інших мов, наприклад як C/C++, Perl, PHP, JavaScript, Python, Ruby та інші. Існують також модулі для створення графічних інтерфейсів.

* 1. **Засіб автоматизації Apache Maven**

Apache Maven – засіб автоматизації збірки проектів на основі описання їх структури в файлах на мові POM (Project Object Model), яка є підмножиною XML. Проект Maven видається спільнотою Apache Software Foundation.

Технологія забезпечує декларативну, а не імперативну збірку проекту. В файлах опису проекту міститься його специфікація, а не окремі команди виконання. Всі задачі по обробці файлів, описані в специфікації, Maven виконує за допомогою їх обробки послідовністю вбудованих і зовнішніх плагінів.

Середовище розробки Eclipse включає в себе підтримку виконання команд системи Maven за допомогою графічного інтерфейсу.

Інформація для збірки проекту зберігається в XML-файлі з назвою pom.xml. при запуску Maven перевіряє, чи містить конфігураційний файл всі необхідні дані і чи всі дані синтаксично правильно записані.

Мінімальна конфігурація включає в себе версію конфігураційного файлу, ім’я проекту, його автора та версію. За допомогою pom.xml конфігуруються залежності від інших проектів, індивідуальні фази процесу побудови проекту, список плагінів, які реалізовують порядок збірки.

Великі проекти можуть бути поділені на декілька модулей або підпроектів, кожен зі своїм pom-файлом. Операції над модулями можуть виконуватись через загальний кореневий pom.xml однією командою.

* 1. **Мова програмування C#**

C# - об’єктно-орієнтована мова програмування, розроблена в 2001 році групою інженерів під керівництвом Андерса Хайлсберга в компанії Microsoft як мова розробки додатків для платформи Microsoft .NET Framework.

C# відноситься до сім’ї С-подібних мов. Мова має статичну типізацію, підтримує поліморфізм, перевантаження операторів, делегати, атрибути, події, узагальнені типи і методи, ітератори, анонімні функції з підтримкою замикань, LINQ, виключення, коментарі в форматі XML.

C# розроблено як мову програмування прикладного рівня для CLR (Common Language Runtime) і залежить, перш за все, від можливостей самої CLR. Це стосується, насамперед, системи типів C#, яка відображає бібліотеку базових класів. Присутність чи відсутність тих чи інших особливостей мови диктується тим, чи може конкретна особливість мови транслюватись в відповідності конструкції CLR.

CLR надає С#, як і всім іншим .NET-орієнтованим мовам, додаткові можливості, яких позбавлені класичні мови програмування. Наприклад, збірка сміття не реалізована в самому C#, а виконується CLR для програм, написаних для програм на платформі .NET.

* 1. **Технологія ASP.NET Core MVC**

ASP.NET MVC Framework – технологія для створення веб-додатків, яка реалізовує шаблон Model-View-Controller. В квітні 2009 року вихідний код був опублікований під ліцензією Microsoft Public License. Наразі розробляється ASP.NET MVC 6 як частина ASP.NET Core, 27 червня відбувся вихід версії 1.0.0.

Платформа ASP.NET MVC базується на взаємодії трьох компонентів: контролеру, моделі та представлення. Контролер приймає запити, обробляє користувацький ввід, взаємодіє з моделлю та представленням і повертає результат обробки запиту. Модель представляє собою шар, який описує логіку організації даних в додатку. Представлення отримує дані з контролеру та генерує елементи користувацького інтерфейсу для відображення інформації.

.NET Core являється багатоплатформенним аналогом .NET Framework з відкритим вихідним кодом. Він містить середовище CoreCLR – багатоплатформенну реалізацію CLR, в яку входить JIT-компілятор – RyuJIT. Також .NET Core включає в себе CoreFX – набір бібліотек, функціональність яких частково співпадає з бібліотекою базових класів. API .NET Core представляє підмножину API .NET Framework, але також надає ряд власних API.

16 листопада 2016 року було випущено .NET Core версії 1.1.

* 1. **Середовище розробки Microsoft Visual Studio**

Microsoft Visual Studio – серія продуктів компанії Microsoft, які включають інтегроване середовище розробки програмного забезпечення і ряд інших інструментальних засобів. Дані продукти дозволяють розробляти як консольні додатки, так і додатки з графічним інтерфейсом, в тому числі з підтримкою технологій Windows Forms та WPF, а також веб-сайти, веб-додатки, веб-служби та інше.

Середовище також включає в себе редактор вихідного коду з підтримкою технології IntelliSense та можливістю найпростішого рефакторингу коду. Вбудований відладчик може працювати як відладчик вихідного коду, так і відладчик машинного рівня. Visual Studio дозволяє створювати та підключати сторонні доповнення для розширення функціональності практично на кожному рівня, включаючи додавання підтримки систем контроля версій вихідного коду.

* 1. **Система керування базами даних MS SQL**

Microsoft SQL Server – система управління реляційними базами даних, розроблена корпорацією Microsoft. Основною мовою запитів є Transact-SQL, створений спільно Microsoft та Sybase. Transact-SQL являє собою реалізацію стандарту ANSI/ISO по структурованій мові запитів з розширеннями. Використовується для роботи з базами даних розміром від персональних до великих баз даних масштабів підприємств.

13-го січня 1988 року в Нью-Йорку відбулась прес-конференція, на якій було об’явлено про союз Ashton-Tale і Microsoft для розробки нового продукту, який отримав назву Ashton-Tale/Microsoft SQL Server. 29-го травня 1989 року розпочався офіційний продаж Ashton-Tale/Microsoft SQL Server 1.0.

Реліз SQL Server 2014 став доступний 1 квітня 2014 року.

MS SQL Server 2014 містить наступні нововведення:

* резервне копіювання в SQL Server по URL-адресу;
* шифрування для резервного копіювання;
* розміщення бази даних SQL Server в віртуальній машині Windows Azure;
* нова конструкція для оцінки кількості елементів;
* переключення секцій та індексування;
* управління пріорітетом блокування операцій в мережі;
* розширення буферного пулу;
* додаткові статистики;
* розширення Transact-SQL;
* покращення системи безпеки.

З 1-го червня 2016 року доступний SQL Server 2016.

Для роботи з базою даних використовується середовище SQL Server 2014 Management Studio.

1. **КОНСТРУКТОРСЬКА ЧАСТИНА**
   1. **Аналіз та виділення об’єктів предметної області**

В ході розробки проекту було досліджено предметної області. На основі цього визначено об’єкти предметної області, якими маніпулює система. Кожен з цих об’єктів характеризується набором атрибутів та знаходиться у зв’язку з іншими об’єктами. Реалізовано такі зв’язки, як: один до одного, один до багатьох, багато до багатьох.

В ході дослідження та аналізу предметної області виділено 6 об’єктів:

* User;
* PersonalInfo;
* Announcement;
* Conversation;
* Message;
* EventLog.

Об’єкт типу «User» – користувачі додатку, характеризується наступними атрибутами:

* Id – ключове поле;
* Email – електронна пошта;
* IsAdmin – статус адміністратора;
* IsBanned – статус блоку;
* IsOnline – присутність користувача в мережі;
* Login – назва профілю;
* Password – пароль профілю;
* RegistrationDate – дата реєстрації.

Об’єкт типу «PersonalInfo» - персональна інформація користувача, характеризується наступними атрибутами:

* Id – ключове поле;
* BirthDate – дата народження;
* FirstName – ім’я;
* LastName – прізвище;
* PhoneNumber – номер телефону;
* Picture – посилання на файл з картинкою;
* UserId – код користувача, вторинний ключ, використовується для зв’язку з об’єктом «User».

Об’єкт типу «Announcement» - оголошення для всіх користувачів додатку, характеризується наступними атрибутами:

* Id – ключове поле;
* ClosingDate – дата закриття;
* CreationDate – дата створення;
* Description – опис;
* IsActive – стан активності;
* UserId – код користувача, вторинний ключ, використовується для зв’язку з об’єктом «User».

Об’єкт типу «Conversation» - чат, характеризується наступними полями:

* Id – ключове поле;
* CreationDate – дата створення;
* Name – назва.

Об’єкт типу «Message» - повідомлення, характеризується наступними полями:

* Id – ключове поле;
* Attachment – посилання на прикріплений до повідомлення файл;
* ConversationId – код чату, вторинний ключ, використовується для зв’язку з об’єктом «Conversation»;
* SendDate – дата відправлення;
* Text – текст;
* UserId – код користувача, вторинний ключ, використовується для зв’язку з об’єктом «User».

Об’єкт «EventLog» - журнал подій, характеризується наступними полями:

* Id – ключове поле;
* CreatedTime – дата;
* EntityId – сутність, з якою пов’язана подія;
* EventId – тип події;
* Message – коментар.
  1. **Побудова ER-діаграми бази даних**

ER-модель – модель даних, яка дозволяє описувати концептуальні схеми за допомогою узагальнених конструкцій та блоків. ER-модель – це засіб опису моделей даних.

Модель сутність-зв’язок є результатом систематичного процесу, який описує деяку предметну область. Вона не визначає сам процес, а лише візуалізовує його. Дані представлені у вигляді сутностей, які пов’язані між собою певними зв’язками, які виражають залежності і вимоги між ними. Сутності можуть мати різні властивості (атрибути), які характеризують їх.

На основі досліджень предметної області розроблено ER-модель бази даних за допомогою інструмента для побудови діаграми в середовищі SQL Server 2014 Management Studio.

ER-діаграму представлено в додатку А.

* 1. **Розробка структури програми**

Для зручного зв’язку користувача з сервером створено додаток на мові Java з використанням графічної бібліотеки JavaFX.

При проектуванні структури клієнтського додатку головною метою було створити інтуїтивно зрозумілий інтерфейс та швидкий доступ до всіх важливих функцій.

Після запуску програми відкривається вікно вибору мови. Наразі доступні дві: англійська та українська. Вікно вибору мови представлено на рисунку 2.1.

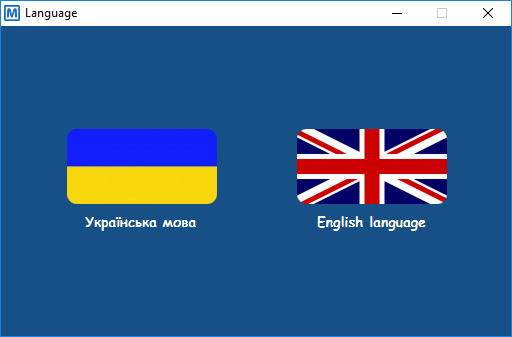


Рисунок 2.1 – Структура вікна вибору мови

Після вибору мови відкривається вікно авторизації, де користувач може ввести логін і пароль та перейти на головне вікно програми. Вікно авторизації представлено на рисунку 2.2.

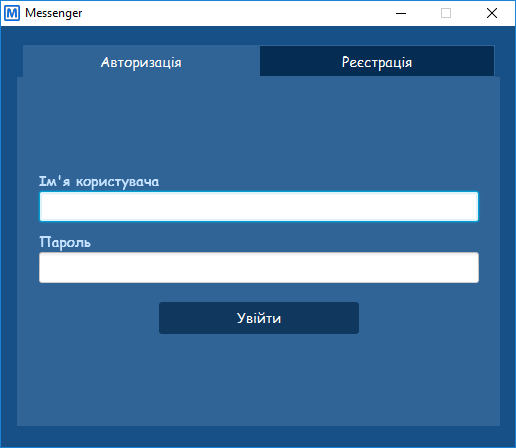


Рисунок 2.2 – Структура вікна авторизації

Також є можливість перейти у режим авторизації за допомогою перемикання вкладки. Режим реєстрації представлено на рисунку 2.3.

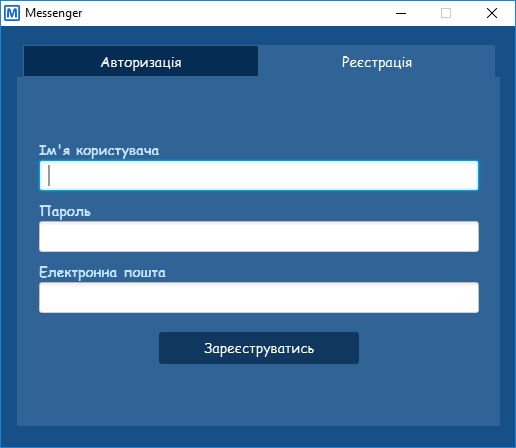


Рисунок 2.3 – Структура вікна реєстрації

При неправильному вводі при реєстрації або авторизації користувач отримає повідомлення про помилку. Приклад повідомлення про помилку представлено на рисунку 2.4.

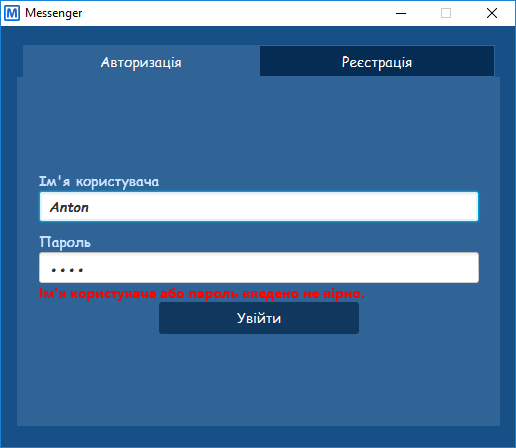


Рисунок 2.4 – Помилка при неправильному вводі

Після успішної авторизації відкривається головне вікно програми, за допомогою якого можна використовувати весь функціонал додатку. Головне вікно програми представлено на рисунку 2.5.

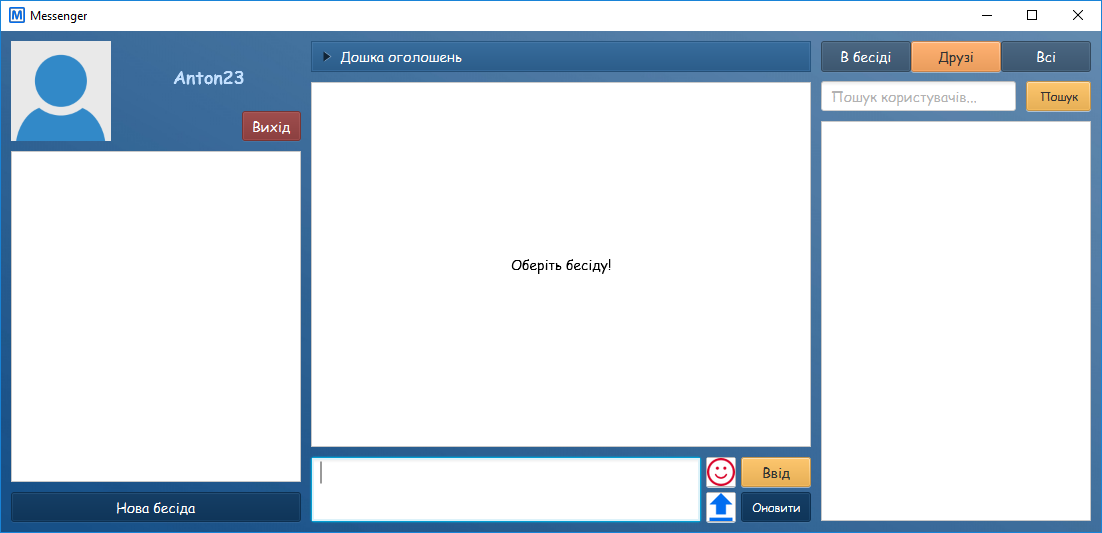


Рисунок 2.5 – Структура головного вікна програми

Користувач може переглянути список користувачів та створити нові чати. Запросивши інших користувачів до чату з’являється можливість обміну повідомленнями. Приклад обміну повідомленнями представлено на рисунку 2.6.

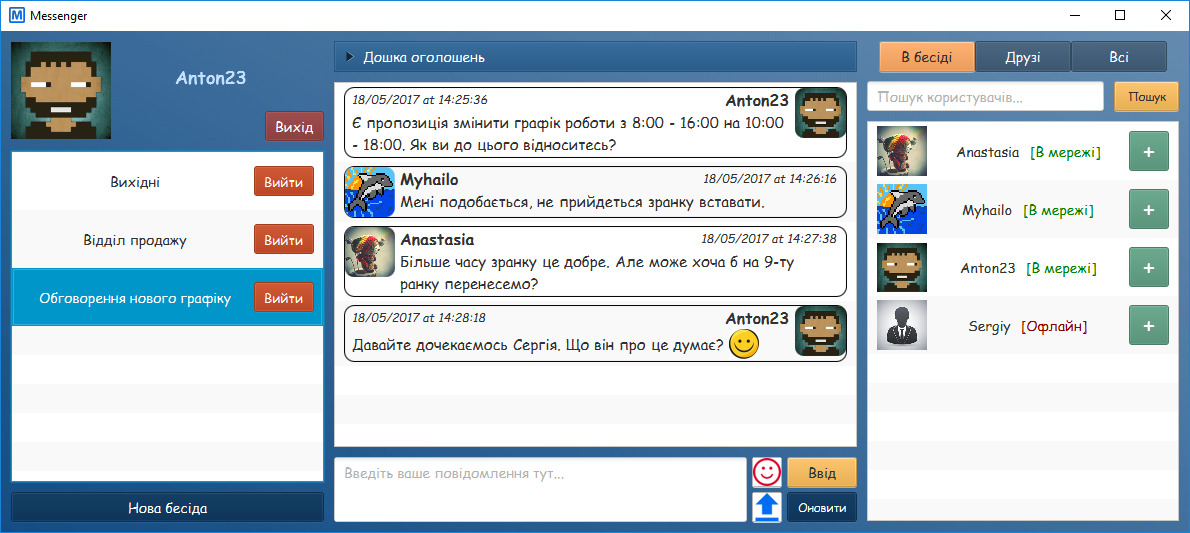


Рисунок 2.6 – Обмін повідомленнями у чаті

Кожен користувач може створити оголошення, яке будуть бачити всі інші користувачі мережі на дошці оголошень. Приклад перегляду дошки оголошень представлено на рисунку 2.7.

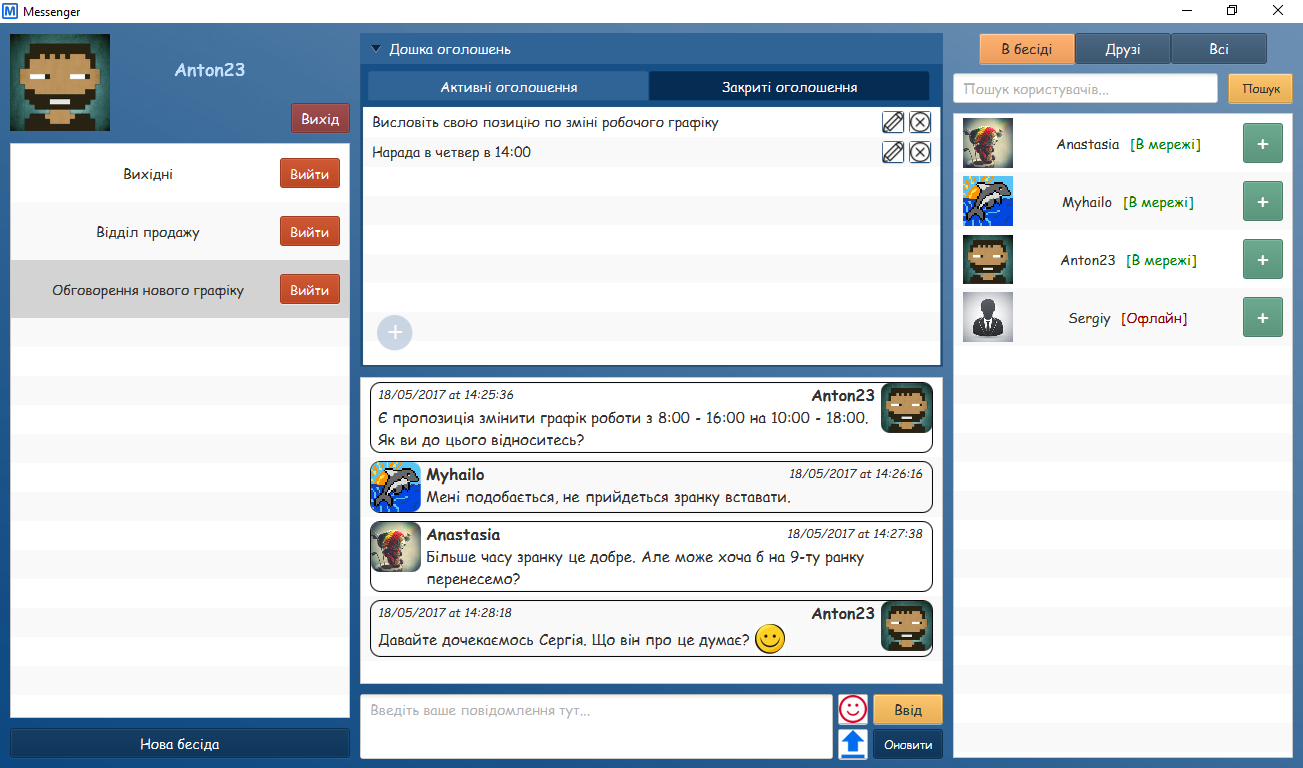


Рисунок 2.7 – Перегляд дошки оголошень

У кожного користувача є персональна інформація. Що переглянути її достатньо натиснути по зображенню картинки профілю. Власну персональну інформацію можна редагувати. Приклад вікна з персональною інформацією представлено на рисунку 2.8.

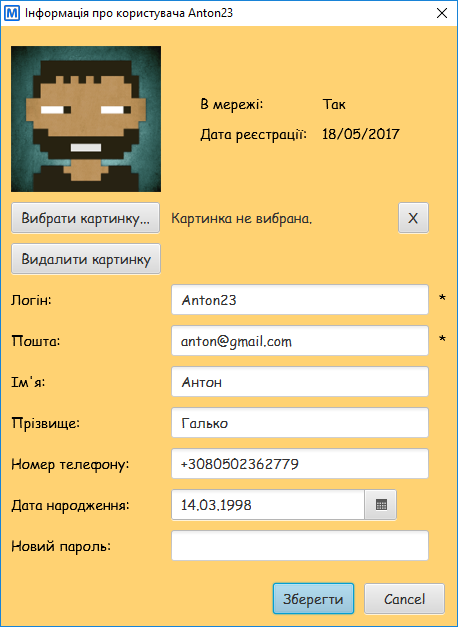


Рисунок 2.9 – Структура вікна з персональною інформацією

Після запуску додатку для адміністратора користувачу пропонується авторизуватись. Зайти в цей додаток може тільки користувач з правами адміністратора. Приклад вікна аутентифікації та авторизації представлено на рисунку 2.10.

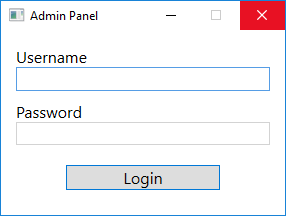


Рисунок 2.10 – Структура вікна авторизації та аутентифікації

Після успішної аутентифікації та авторизації користувач отримує доступ до головного вікна програми, де він може виконувати операції над даними, переглядати журнал подій, а також формувати весь або частини журналу у звіт. Приклад головного вікна програми представлено на рисунку 2.11.

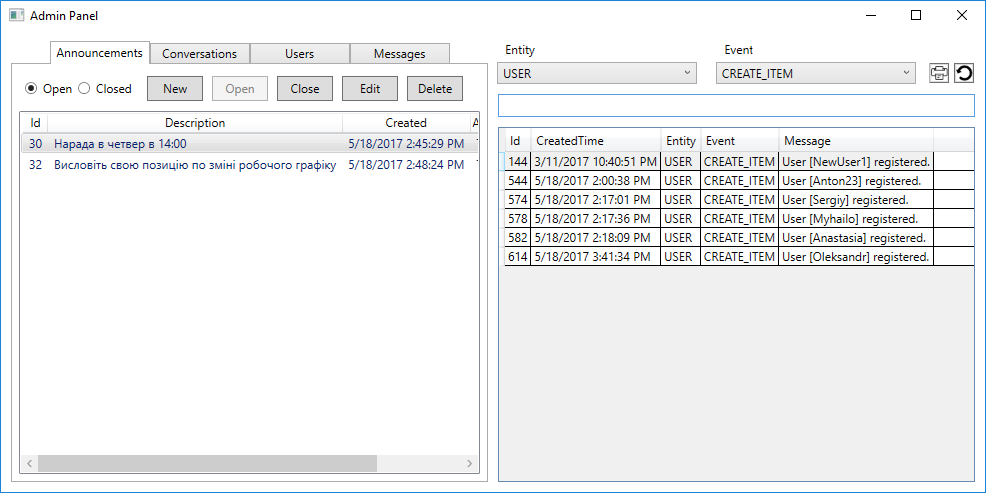


Рисунок 2.11 – Структура головного вікна програми