

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ
імені адмірала Макарова

Херсонська філія

Кафедра інформаційних технологій та фізико-математичних дисциплін

“Допущений до захисту”
Завідувач кафедри
д.т.н., доцент Гучек П.Й. “_____”
_____ 2020 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи

на здобуття ступеня вищої освіти “магістр”

на тему: «Дослідження технологій створення Web–додатків для
організації та контролю робочого процесу компанії»

Виконав: студент VI курсу, групи 6157м
спеціальності

122 - “Комп’ютерні науки”, ОПП –

(шифр і назва спеціальності та освітньо-професійної програми)

“Інформаційні управляючі системи та технології”

Фокічева Л.В.

(прізвище та ініціали)

Керівник

к.т.н., Гайдаєнко О.В.

(прізвище та ініціали)

Рецензент

(прізвище та ініціали)

30.12 – 2020 року

Анотація

Дипломна робота створена для автоматизації та контролю робочого процесу компанії.

При розробці додатку були проаналізовані аналогічні онлайн-сервіси та реалізовано програмне забезпечення, яке надає змогу:

- 1) створення, редагування та видалення інформації;
- 2) реєстрація, логінізація та делогінізація у системі;
- 3) постановки задач та відстеження процесу їх виконання у реальному часі;
- 4) ведення бази співробітників компанії (додавання, редагування, видалення);
- 5) спілкування через додаток (чат або дзвінки).

Робота викладена українською мовою і виконана на 110 аркушах, містить 22 рисунків, 4 таблиць, 4 додатка. Список використаних джерел містить 9 найменувань.

Summary

The thesis is designed to automate and control the company's workflow.

During the development of the application, similar online services were analyzed and software was implemented, which allows:

- 1) creating, editing and deleting information;
- 2) registration, login and logon in the system;
- 3) creation of tasks and tracking the process of their implementation in real time;
- 4) maintaining the base of the company's employees (adding, editing, deleting);
- 5) communication through the application (chat or calls).

Work outlined in the Ukrainian language and executed on 110 sheets, contains 22 figures, tables 4, 4 appendix. The list of the used sources contains 9 of the names.

Зміст

Вступ.....	7
1. ДОСЛІДЖЕННЯ МЕТОДІВ ВИКОРИСТАННЯ ВЕБ-ДОДАТКІВ ДЛЯ ОРГАНІЗАЦІЇ ТА КОНТРОЛЮ РОБОЧОГО ПРОЦЕСУ КОМПАНІЇ	
1.1 Загальний опис веб-додатку	8
1.2 Структура веб-додатку	9
1.2.1 Функціональні можливості веб-додатку	10
1.2.2 Архітектура веб-додатку	10
1.3 Аналіз існуючих рішень для розробки веб-додатку для організації та контролю робочого процесу компанії	11
1.4 Вітчизняний і зарубіжний досвід застосування веб-додатку для організації та контролю робочого процесу компанії.....	12
1.5 Постановка задачі та опис діяльності організації.....	13
1.6 Вимоги та архітектура веб-додатку для віртуальної організації	15
1.6.1 Дослідження проблем побудови веб-додатку для організації та контролю робочого процесу компанії.....	17
1.6.2 Розробка архітектури для веб-додатку для організації та контролю робочого процесу компанії.....	17
1.6.3 Вимоги до якості розподілення ресурсів в веб-додатку для організації робочого процесу компанії.....	18
Висновки за розділом 1	20
2. ПРОЕКТУВАННЯ РІШЕНЬ ЩОДО ВЕБ-ДОДАТКУ ДЛЯ ОРГАНІЗАЦІЇ ТА КОНТРОЛЮ РОБОЧОГО ПРОЦЕСУ КОМПАНІЇ ВЕБ-ДОДАТКУ	
2.1 Побудова моделі варіантів використання	21
2.2 Побудова концептуальної моделі.....	22
2.3 Рішення з інформаційного забезпечення.....	23
2.3.1 Логічна модель	24
2.3.2 Фізична модель	26

2.3.3	Діаграма класів	27
2.3.4	Діаграма послідовності.....	29
2.4	Рішення з технічного забезпечення.....	31
3.4.1	Опис комплексу технічних засобів.....	31
2.5	Рішення з математичного забезпечення.....	31
2.5.1	Математична модель	31
2.6	Рішення з програмного забезпечення.....	34
2.6.1	Вибір мови проектування.....	34
2.6.2	Модель переходів станів.....	35
	Висновки за розділом 2	37
3.	РОЗРОБКА ПРОЕКТНИХ РІШЕНЬ ЩОДО ВЕБ-ДОДАТКУ ДЛЯ ОРГАНІЗАЦІЇ ТА КОНТРОЛЮ РОБОЧОГО ПРОЦЕСУ КОМПАНІЇ	
3.1	Загальносистемні рішення.....	39
3.1.1	Представлення схеми організаційної структури.....	39
3.1.2	Представлення схеми функціональної структури	40
3.1.3	Описання функцій, що автоматизуються.....	40
3.1.4	Загальний опис системи.....	41
3.2	Рішення з організаційного забезпечення.....	41
	Висновки за розділом 3.....	43
4.	РОЗРАХУНОК І ОЦІНКА ЕКОНОМІЧНОГО ЕФЕКТУ ВІД РОЗРОБКИ ВЕБ-ДОДАТКУ ДЛЯ ОРГАНІЗАЦІЇ ТА КОНТРОЛЮ РОБОЧОГО ПРОЦЕСУ КОМПАНІЇ	
4.1	Загальна характеристика роботи.....	44
4.2	Розрахунок витрат на проектування та впровадження системи.....	44
4.2.1	Техніко-економічний аналіз.....	44
4.2.2	Вартісний аналіз.....	47
4.3	Розрахунок економічної ефективності розробки і впровадження системи.....	51

Висновки за розділом 4	53
------------------------------	----

5. ОХОРОНА ПРАЦІ

5.1 Вступ.....	54
5.2 Аналіз небезпечних і шкідливих факторів.....	55
5.2.1 Освітленість робочого місця.....	55
5.2.2 Вплив випромінювання.....	56
5.2.3 Вплив електричного струму.....	58
5.2.4 Виробничий шум.....	59
5.2.5 Умови мікроклімату.....	60
5.2.6 Організація робочого місця.....	61
5.3 Розрахунок освітленості робочого місця.....	63
5.4 Розробка заходів щодо зменшення впливу небезпечних і шкідливих факторів.....	65
5.4.1. Заходи щодо зниження величини виникаючих зарядів статичної електрики.....	65
5.4.2 Заходи щодо зниження шуму на робочому місці.....	65
Висновки за розділом 5	67

6. ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

6.1 Вступ.....	69
6.2 Загальні відомості про забруднення.....	69
6.3 Заходи щодо запобігання забруднення навколишнього середовища	69
Висновки за розділом 6	71
Висновок.....	72
Список використаних джерел.....	73
Додаток А Інструкція користувача.....	74
Додаток Б Випробовування системи.....	83
Додаток В Опис програми.....	87
Додаток Г Текст програми.....	90

ВСТУП

«Дистанційна робота в останні роки набирає популярність, і в світі значна кількість компаній, які обрали для себе такий формат. Є три основні переваги: розширення зони пошуку співробітників, ефективність, баланс життя і роботи, а також лояльність команди. Для визначення кращого підходу організації роботи компанії, потрібно орієнтуватися як на інтереси керівників так і на інтереси співробітників, чітку постанову та упорядкування задач, визначення строків та надання всім працівникам рівного доступу до проекту, зменшити час на постановку цілей та їх досягнення. Саме цьому за останній час великої популярності набули онлайн–сервіси для організації та контролю робочого процесу компанії.

Онлайн–сервіси дозволяють автоматизувати процес роботи компанії, що виводить якість та швидкість розробки проектів на новий рівень. Постановка та відстеження задач у реальному часі дозволяють швидко та своєчасно вносити корективи, відстежувати процес виконання та досягнення цілей. Відстежувати затрати часу, ресурсів та кадрів на ті чи інші задачі. Регулювати кадрові ресурси та затрати на кожному етапі розробки, аналізувати та вдосконалювати робочий процес...»[1].

Актуальність обраної для дослідження теми полягає в тому, що сотні компаній щодня, переходять на віддалений режим роботи, тобто робочий процес відбувається не виходячи з дому. Організація роботи, як і раніше, є основним завданням, тому надопомогу приходять додатки. За допомогою спеціального програмного забезпечення можна легко вести базу співробітників, оперативно створювати та розподіляти завдання, відстежувати процес їх виконання, проводити ділові дзвінки та бесіди і тп.

1. ДОСЛІДЖЕННЯ МЕТОДІВ ВИКОРИСТАННЯ ВЕБ–ДОДАТКІВ ДЛЯ ОРГАНІЗАЦІЇ ТА КОНТРОЛЮ РОБОЧОГО ПРОЦЕСУ КОМПАНІЇ

1.1 Загальний опис веб–додатків

Веб–додаток – це складний багаторівневий сайт для обслуговування вузької або широкої аудиторії, яка повинна мати можливість виконувати конкретні дії, спрямовані на реалізування задач служби. За допомогою таких сервісів в Інтернеті можна: проводити грошові операції, спілкуватися, шукати, зберігати, редагувати, пересилати та розповсюджувати інформацію та багато іншого.

Веб–додаток на сьогодні є не тільки інформаційним засобом або CRM–системою, а повноцінним маркетинговим інструментом, що привертає нових клієнтів і приносить прибуток.

Незважаючи на всю різноманітність послуг, які вони надають, веб–додатки можна розділити на наступні основні групи, в залежності від їх тематики і призначення:

- 1) Інформаційно–пошукові. Дані ресурси призначені для пошуку необхідної інформації. До них відносяться не тільки пошукові бази, а й онлайн–бібліотеки з книгами, музикою або фільмами, різні каталоги і довідники, а також онлайн–перекладачі.
- 2) Соціальні. До цієї групи відносяться різноманітні соціальні мережі, онлайн–комунікатори, а також електронні поштові служби. За допомогою таких сервісів можна не тільки спілкуватися з людьми, незалежно від їх дислокації, а й вести ділову переписку, пересилати файли і багато іншого.
- 3) Сервіси покупок. Дані ресурси призначені для придбання речей або замовлення послуг. До цієї категорії відносяться не тільки онлайн–

магазини, але і сервіси на замовлення і бронювання квитків (починаючи від театральних і закінчуючи залізничними), готелів і багато чого іншого.

- 4) Банківські сервіси. За допомогою таких ресурсів можна оплатити покупку або послугу, наприклад, замовлення в інтернет–магазині, або навіть заплатити за комунальні послуги, без необхідності особистого відвідування банку.
- 5) Дорожні сервіси дають можливість швидко прокласти маршрут незалежно від способу пересування, оцінити транспортну завантаженість на дорогах, отримати інформацію про тривалість поїздки тощо.

Також, веб–додатки бувають як платними, так і безкоштовними. На платній основі, після проходження процедури реєстрації та оплати членства, користувач отримує доступ до послуг, який даний сервіс надає. Це може бути як доступ до перегляду фільмів онлайн в високій якості, можливість завантаження книг або доступ до певного функціону. Безкоштовні ресурси пропонують свої послуги безоплатно часто навіть не вимагають проходження процедури реєстрації. Однак, і якість цих послуг, часто, набагато нижче, ніж в платних сервісах.

1.2 Структура веб–додатку

При розробці веб–додатку основна увага приділяється його інтерфейсу, оскільки він повинен бути інтуїтивно зрозумілим користувачам, а також швидко завантажуватися з будь–яких пристроїв. Наприклад, дорожніми сервісами користувачі часто користуються вже перебуваючи в дорозі і завантажують їх на смартфонах або планшетах.

Web–сайти фундаментально відрізняються від Web–додатків. Web–додатки, як правило, є інтерактивними. Якщо сайт не є інформаційним, то, швидше за все, він є інтерактивним. Це означає, що користувач є не пасивним споживачем, а є

активним учасником сайту. Користувач шукає, натискає кнопки, заповнює форми, робить певні операції і взагалі постійно працює клавіатурою і мишкою. Сайт є інформаційним або інтерактивним. Більшість інтерактивних сайтів насправді є гібридами. Важливо по-різному представляти для клієнтів сайт, через який потрібно надати користувачам інформацію і сайт, на якому необхідно забезпечити інтерактивність. Ці відмінності в баченні фактично визначатимуть більшість серйозних дизайнерських рішень. Навіть незважаючи на те, що рідко створюються виключно інформаційні або виключно інтерактивні сайти.

1.2.1 Функціональні можливості веб-додатку

Веб-додатки бувають різного призначення, тому їх функціональні можливості залежать насамперед від конкретної сфери для якої вони розробляються. Веб-додаток для організації роботи компанії повинен надавати можливості для:

- 6) створення, редагування та видалення інформації;
- 7) постановки задач та відстеження процесу їх виконання у реальному часі;
- 8) ведення бази співробітників компанії (додавання, редагування, видалення);
- 9) спілкування через додаток (чат або дзвінки).

Веб-додатки дозволяють збирати, зберігати і аналізувати дані, отримані від користувачів. Довгий час використовувався метод, при якому дані, введені в HTML-форми, відсилалися для обробки CGI-додатків або спеціально призначеним працівникам у вигляді повідомлень електронної пошти. Веб-додаток дозволяє зберігати дані безпосередньо в базі даних, а також отримувати дані і формувати звіти на основі отриманих даних для аналізу.

1.2.2 Архітектура веб–додатку

На сьогоднішній день найчастіше використовуються такі технології для реалізації веб–сервісів:

- 1) Стек протоколів TCP/IP–набір протоколів мережі Інтернет, який розуміється практично з будь–яким мережним устаткуванням, від мейнфреймів до портативних пристроїв і PDA.
- 2) HTML–універсальна мова гіпертекстової розмітки, яка використовується для демонстрації контенту пристроями споживачів.
- 3) XML–універсальний засіб для обробки всіх різновидів даних. На його базі можуть працювати і інші протоколи обміну інформацією: SOAP і WSDL. Специфікація XML описує XML–документи і частково описує поведінку XML–процесорів (програм, які читають XML–документи і забезпечують доступ до їх вмісту). XML розроблявся як мова з простим формальним синтаксисом, зручний для створення і обробки документів програмами і одночасно зручний для читання і створення документів людиною, з підкресленням націленості на використання в Інтернеті. Мова називається розширюваним, оскільки ним не фіксується розмітка, яка використовується в документах: розробник вільний створити розмітку відповідно до потреб конкретної області, будучи обмеженим лише синтаксичними правилами мови. Розширення XML – це конкретна граматики, створена на базі XML і представлена словником тегів і їх атрибутів, а також набором правил, що визначають які атрибути і елементи можуть входити до складу інших елементів. Поєднання простого формального синтаксису, зручності для людини, розширюваності, а також базування на кодуваннях Юнікод для подання змісту документів привело до широкого використання як власне XML, так і безлічі похідних спеціалізованих мов на базі XML в найрізноманітніших програмних засобах.

4) UDDI–універсальне джерело розпізнавання, інтеграції та опису. Працює, як правило, в приватних мережах і поки не знайшов достатнього поширення.

Універсальність представлених технологій – основа для розуміння веб-служб. Перераховані протоколи працюють на стандартних технологіях, що не залежать від постачальників додатків і інших ресурсів мережі. Можуть використовуватися в будь-яких операційних системах, серверах додатків, мов програмування і т.д.

1.3 Аналіз існуючих рішень для розробки веб–додатку для організації та контролю робочого процесу компанії

Онлайн–сервіс для організації та контролю роботи є фундаментом побудови успішного робочого процесу команди компанії. Саме цьому все більше ІТ компаній впроваджують такі додатки у свою роботу. Керівництво компаній стоїть перед складним питанням вибору такого сервісу. Тому, розглянемо за якими критеріями обирають.

Перше й найголовніше–це функціонал, який може надати конкретний сервіс. Кожен споживач шукає такий додаток, який надасть йому можливість виконувати свої конкретні задачі. Друге– це звичайно доступність, тому що для когось важлива безкоштовність, а для когось функціональність сервісу, за якість якої зазвичай необхідно платити. Існує багато видів покупки продукту, тому вони поділяються на:

- 1) одноразова покупка;
- 2) покупка певного об'єму (для певної кількості користувачів, наприклад: 10, 50 чи безліміт користувачів);
- 3) підписка на певний період часу (рік, місяць, тиждень);
- 4) підписка на певний пакет послуг (функціональні можливості, які надаються).

Розглянемо веб-додаток Asana – це програмне забезпечення для веб і мобільних пристроїв, призначене для спільної роботи над проектами без електронної пошти. Було створене співзасновником Facebook Дастіном Московіцем і колишнім інженером Джастіном Розенштейном, які працювали над підвищенням продуктивності співробітників на Facebook. Тобто даний додаток надає можливості для організації робочих процесів команди, проте є платний. Один місяць підписки коштує мінімум 11 долларів і може досягати 30,5 долларів за людину, в залежності від виду підписки. Звичайно для невеликих компаній це дороге, тому велика кількість з них відмовляється від користування додатком. Веб-додаток Asana можна використовувати безкоштовно, але зі значно зменшеним функціоналом і обмеженим числом користувачів – 15 чоловік, тому даний додаток не є корисним компаніям з маленьким капіталом.

Додаток Trello – це одна з найпопулярніших систем управління проектами в режимі онлайн, яка користується особливим попитом серед невеликих компаній і стартапів. Вона використовує парадигму керування проектами, відому як канбан. Проекти зображуються дошками, що містять списки. Списки містять картки, якими зображуються задачі. Картки повинні переходити з попереднього списку до наступного (за допомогою перетягування), таким чином зображуючи рух якоїсь функції від ідеї, аж до тестування. Картці може бути присвоєно відповідальних за неї користувачів. Користувачі та дошки можуть об'єднуватись в команди. Трело можна спробувати безкоштовно, користувач має можливість використати 10 дошок, але зі зменшеним функціоналом. Після використання 10 дошок і для використання повного функціоналу додатку необхідно оформити підписку, вартість якої почитається з 12,5 долларів за місяць і збільшується з кількістю функціоналу.

У ході аналізу існуючих додатків стає зрозумілим, що для невеликих компаній не має повноцінних безкоштовних додатків.

1.4 Вітчизняний і зарубіжний досвід застосування веб–додатку для організації та контролю робочого процесу компанії

Веб–додатки для організації та контролю робочого процесу компанії поширені у ІТ індустрії, безпосередньо з якої вони і почали своє існування, а також у великих корпораціях та організаціях.

Вітчизняних веб–додатків для організації та контролю робочого процесу компанії не існує, але українськими компаніями активно використовуються такі додатки, наприклад компанії Danone, Vodafone та Avon використовують додаток Asana. Більше того майже всі ІТ компанії у своїй роботі використовують один із додатків: Trello, Asana, Slack.

Розглянемо зарубіжні додатки. Одним з найвідоміших веб–додатків для організації та контролю робочого процесу компанії є додаток Jira, який дозволяє оперативнo створювати задачі, присвоювати їм конкретного виконавця або групу, задавати час виконання та терміновість, планувати і відстежувати процес їх виконання у реальному часі. Також можна задавати пріоритетність задачам.

Аналогічний додаток Trello, дозволяє створювати «дошки–задачі» надавати їм рівень пріоритетності, задавати час на виконання і назначати виконців. Також у додатку можна вести діалог та залишати повідомлення щодо певної задачі, що дозволяє зменшити затрати часу на обговорювання подробиць у іншому додатку.

У таблиці 1.1 представлена порівняльна характеристика функціоналу існуючих додатків.

Таблиця 1.1 – Порівняльна характеристика функціоналу додатків

Додаток/ Функціонал	Asana	Jira	Trello	Slack
Постановка задач	+	+	+	–
Інтеграція з іншими	+	+	+	+

додатками				
Комунікація	—	—	+	+
Профіль співробітника	+	—	—	+
Безкоштовність	—	—	—	—

У результаті аналізу таблиці можна зробити висновок, що кожен з розглянутих додатків має як свої переваги, так і недоліки, але головний недолік перерахованих додатків, це висока вартість покупки. Тому було вирішено розробити додаток з обмеженим функціоналом, який надаватиме головні можливості, такі як: вести профіль співробітника, створювати задачі, спілкуватися безпосередньо у додатку та буде безкоштовним.

1.5 Постановка задачі та опис діяльності організації

Об'єкт дослідження – веб–додаток для організації та контролю робочого процесу компанії.

Предмет дослідження – процес створення та ведення профілю користувача, процес створення задач, процес комунікації.

Задачею даної роботи є реалізація власного веб–додатку для організації та контролю робочого процесу компанії, ґрунтуючись на перевагах та недоліках уже готових рішень. Проаналізувавши уже готові рішення та способи їх застосування, можна зробити висновок, що актуальною буде розробка з безкоштовного додатку з невеликою кількістю функціоналу, зручним інтерфейсом та максимально безбечним користуванням. Це спричинено бажанням забезпечити користувачам безпечне спілкування та збереження даних у додатку, оскільки такі системи переважно використовуються для організації роботи у ІТ компаніях, для яких конфіденційність та безпечність є

головною запорукою успіху. Отже, розробка повинна задовольняти наступні основні вимоги:

- 1) безкоштовність;
- 2) безпечність;
- 3) функціональність.

Мета роботи – підвищення ефективності та якості роботи додатку у мережі Internet

Для дослідження буде використовуватися один з методів теоретичного аналізу—формалізація.

Формалізація як метод дослідження має ряд достоїнств:

1. забезпечує повноту огляду певної області проблем, узагальненість підходу до їхнього вирішення;
2. базується на використанні спеціальної символіки, що забезпечує стислість і чіткість фіксації знання;
3. пов'язана із приписуванням окремим символам або їхнім системам певних значень, що дозволяє уникнути багатозначності термінів, властивої звичайним мовам;
4. дозволяє формувати знакові моделі об'єктів, а вивчення реальних речей і процесів замінити вивченням цих моделей.

Формалізація є невід'ємною частиною формальної логіки.

Веб–додаток для організації та контролю робочого процесу компанії являтиме собою веб–сайт, на якому будуть розміщені сторінки з сформованими вмістом, вмістом виступатимуть форми, які користувач зможе додавати і надалі працювати з ними, наповнюючи їх вмістом. Остаточний вміст формується після того, як відвідувач сайту заповнює форми необхідною інформацією, тобто створює профіль працівника або створює задачі і тп.. У зв'язку з тим, що остаточний вміст сторінки залежить від запиту, створеного на основі дій користувача, така сторінка називається динамічною.

Працюючи у додатку користувачі можуть отримати всю необхідну інформацію про існуючі задачі та процеси, створювати та видаляти задачі,

переглядати профілі співробітників, вести ділову переписку або проводити дзінки безпосередньо у додатку. За наявністю прав можуть створювати, редагувати та видаляти профілі співробітників або задачі, змінювати їх пріоритетність.

1.6 Вимоги та архітектура веб–додатку для віртуальної організації

1.6.1 Дослідження проблем побудови веб–додатку для організації та контролю робочого процесу компанії

Головною проблемою для побудови безпечного додатку є ослаблений захист інтелектуальної власності. Системи, подібні вебу, RSS і AJAX, схожі тим, що особливих перешкод для їх повторного використання не існує. Велика частина корисного програмного забезпечення має відкриті тексти, а якщо і немає, то існує мало способів захистити свою інтелектуальну власність. Стандартна функція оглядача інтернет "преглянути вихідний код" дозволяє будь–якій людині скопіювати будь–яку веб–сторінку.

Також чималу роль відіграє вибір мови для написання додатку. Кожна мова програмування має як свої переваги так і недоліки. Деякі мови дуже обмежені у виборі бібліотек та СКБД. Інші ж не мають достатньої супровідної документації. Наприклад: на мові PHP дуже багато суперечностей у коді; мова Python це інтерпретована мова програмування, програми на ній працюють повільніше, ніж написані на компільованих мовах.

1.6.2 Розробка архітектури для веб–додатку для організації та контролю робочого процесу компанії

Для розроблюваного веб-додатку для організації та контролю робочого процесу компанії буде використовуватися архітектура клієнт-сервер. Архітектура клієнт-сервер є однією із архітектурних шаблонів програмного забезпечення та є домінуючою концепцією у створенні розподілених мережних застосунків і передбачає взаємодію та обмін даними між ними. Вона передбачає такі основні компоненти:

- 1) набір серверів, які надають інформацію або інші послуги програмам, які звертаються до них;
- 2) набір клієнтів, які використовують сервіси, що надаються серверами;
- 3) мережа, яка забезпечує взаємодію між клієнтами та серверами.

Сервери є незалежними один від одного. Клієнти також функціонують паралельно і незалежно один від одного. Немає жорсткої прив'язки клієнтів до серверів. Більш ніж типовою є ситуація, коли один сервер одночасно обробляє запити від різних клієнтів; з іншого боку, клієнт може звертатися то до одного сервера, то до іншого. Клієнти мають знати про доступні сервери, але можуть не мати жодного уявлення про існування інших клієнтів. Архітектура компонентів.

React не відноситься до фреймворків у чистому вигляді. Це свого роду модифікована бібліотека, «заточена» під MVC (Model-View-Controller, де Модель відповідає за надання даних, Вид – відображає дані Моделі користувачеві, а Контролер інтерпретує дії користувача і змушує Модель вносити зміни).

Отже, для реалізації архітектури знадобиться більше часу. Низькорівневий API (Application Programming Interface – набір готових команд. У разі використання готових фреймворків – Vue.js і Angular, проблем з підбором або налаштуванням бібліотек для різних завдань вже не виникає. Високорівнева API забезпечує зворотну сумісність для всіх бібліотек.

Це дозволить підключитися до проекту сторонньому програмісту без тривалого вивчення архітектури додатку. Саме в уніфікації процесів криється популярність повноцінних фреймворків.

1.6.3 Вимоги до якості розподілення ресурсів в веб– додатку для організації робочого процесу компанії

При розгляді платформ для створення додатків необхідно виділити два основних існуючих підходи:

1. Безпосередня обробка запитів і формування відповідей.
2. Вбудовування програмного коду в шаблони HTML сторінок.

Перший підхід надає найбільші можливості по управлінню обробкою і підвищенню продуктивності. Він передбачає передачу всіх даних про запит безпосередньо виконуваного коду, який може як сформувати відповідь зі сторінкою для користувача, так і відкрити на передачу потік двійкових даних, наприклад для передачі зображення. Однак при такому підході всі дані для передачі формуються програмним шляхом, що уповільнює розробку простих сторінок і ускладнює взаємодію між верстальником і програмістом.

Прикладами цього підходу служать технології CGI, Java Servlets.

Другий підхід використовує шаблони сторінок користувача, оформлені особливим чином, що дозволяє вставляти в них ділянки програмного коду. Цей підхід особливо ефективний при створенні простих додатків, основна інформація в яких статична, а динамічна інформація може бути згенерована простими програмними конструкціями. При розробці складних програмних систем цей варіант ускладнює взаємодію між компонентами і ускладнює реалізацію складної архітектури. Також він менш ефективний за продуктивністю і призводить до обмеження можливостей реалізації складних сторінок. Прикладами цього підходу служать дуже популярні на даний момент технології PHP, ASP, JSP.

Крім різного підходу до генерації сторінок платформи, розробки в різному ступені задовольняють сучасним вимогам, що висуваються при створенні

складних Web систем. Найбільш важливі з цих вимог, наявність яких робить систему привабливою для використання, наведені нижче:

- 1) Платформна незалежність.
- 2) Мова реалізації.
- 3) Продуктивність, масштабованість.
- 4) Можливості розширення і інтеграції.
- 5) Простота використання, наявність засобів розробки.
- 6) Наявність необхідних програмних бібліотек.

Висновки за розділом 1

У даному розділі проаналізовано моделі аналогічних веб–додатків на функціональність, безпечність, зручність та вартість використання. Обрано метод теоретичного аналізу. Виконана постановка задачі та встановлені вимоги до якості розподілення ресурсів на розробку веб–додатку для організації та контролю робочого процесу компанії.

2. ПРОЕКТУВАННЯ РІШЕНЬ ЩОДО ВЕБ–ДОДАТКУ ДЛЯ ОРГАНІЗАЦІЇ ТА КОНТРОЛЮ РОБОЧОГО ПРОЦЕСУ КОМПАНІЇ ВЕБ–ДОДАТКУ

2.1 Побудова моделі варіантів використання

Діаграма прецедентів є графом, що складається з множини акторів, прецедентів (варіантів використання), обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.[2]

Важливо розуміти, що діаграми варіантів використання не призначені для відображення проекту і не можуть описувати внутрішній устрій системи. Діаграми варіантів використання призначені для спрощення взаємодії з майбутніми користувачами системи, з клієнтами, і особливо знадобляться для визначення необхідних характеристик системи. Іншими словами, діаграми варіантів використання говорять про те, що система повинна робити, не вказуючи самі застосовувані методи.

Дана діаграма демонструє основні відносини між акторами та прецедентами, дозволяючи описати та продемонструвати систему на концептуальному рівні.

На (Рис. 2.1) зображена модель варіантів використання веб-додатку для організації та контролю робочого процесу компанії. Актор: користувач (співробітник компанії). Прецеденти: рєстрація/логінізація/делогінізація, робота з персональною сторінкою, робота з дошкою задач, робота зі сторінкою відео-конференцій, робота зі сторінкою компанії.

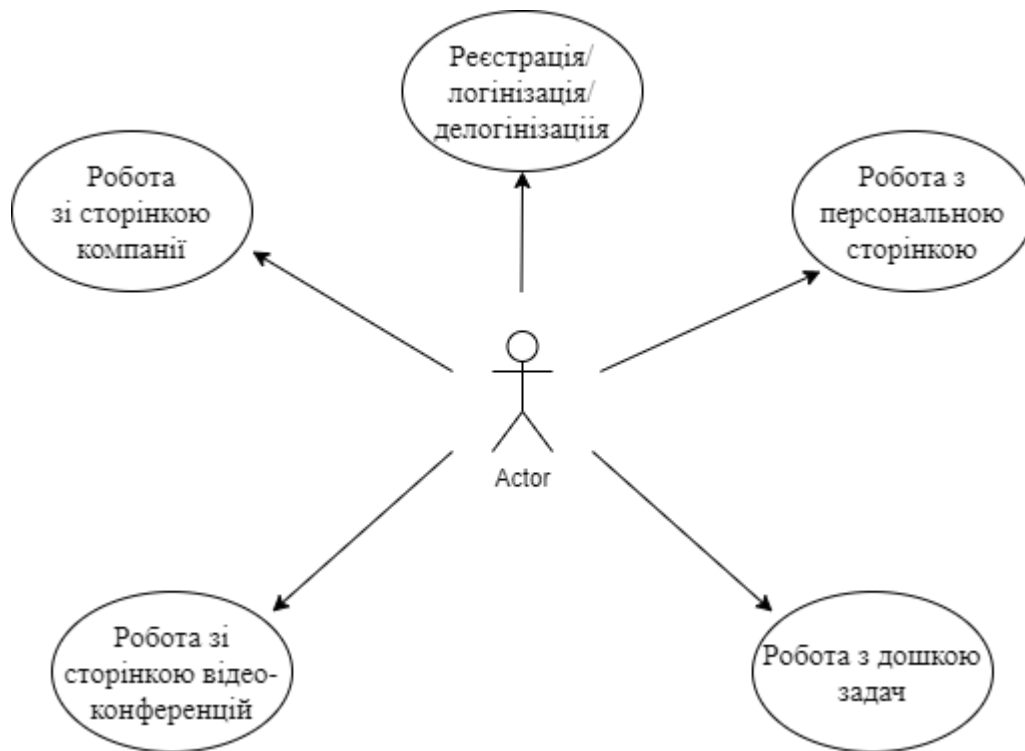


Рисунок 2.1 – Модель варіантів використання у вигляді діаграми прецедентів

2.2 Побудова концептуальної моделі

Концептуальна модель – модель предметної галузі, що складається з переліку взаємопов'язаних понять, що використовуються для опису цієї області, разом з властивостями й характеристиками, класифікацією цих понять, за типами, ситуацій, ознаками в даній галузі, і законів протікання процесів в ній. Концептуальна (змістовна) модель – це абстрактна модель, що визначає структуру модельованої системи, властивості її елементів і причинно–наслідкові зв'язки, властиві системи і суттєві для досягнення мети моделювання.

На (Рис. 2.2) представлена концептуальна модель у вигляді діаграми сутність-зв'язок додатку для організації і контролю робочого процесу компанії:

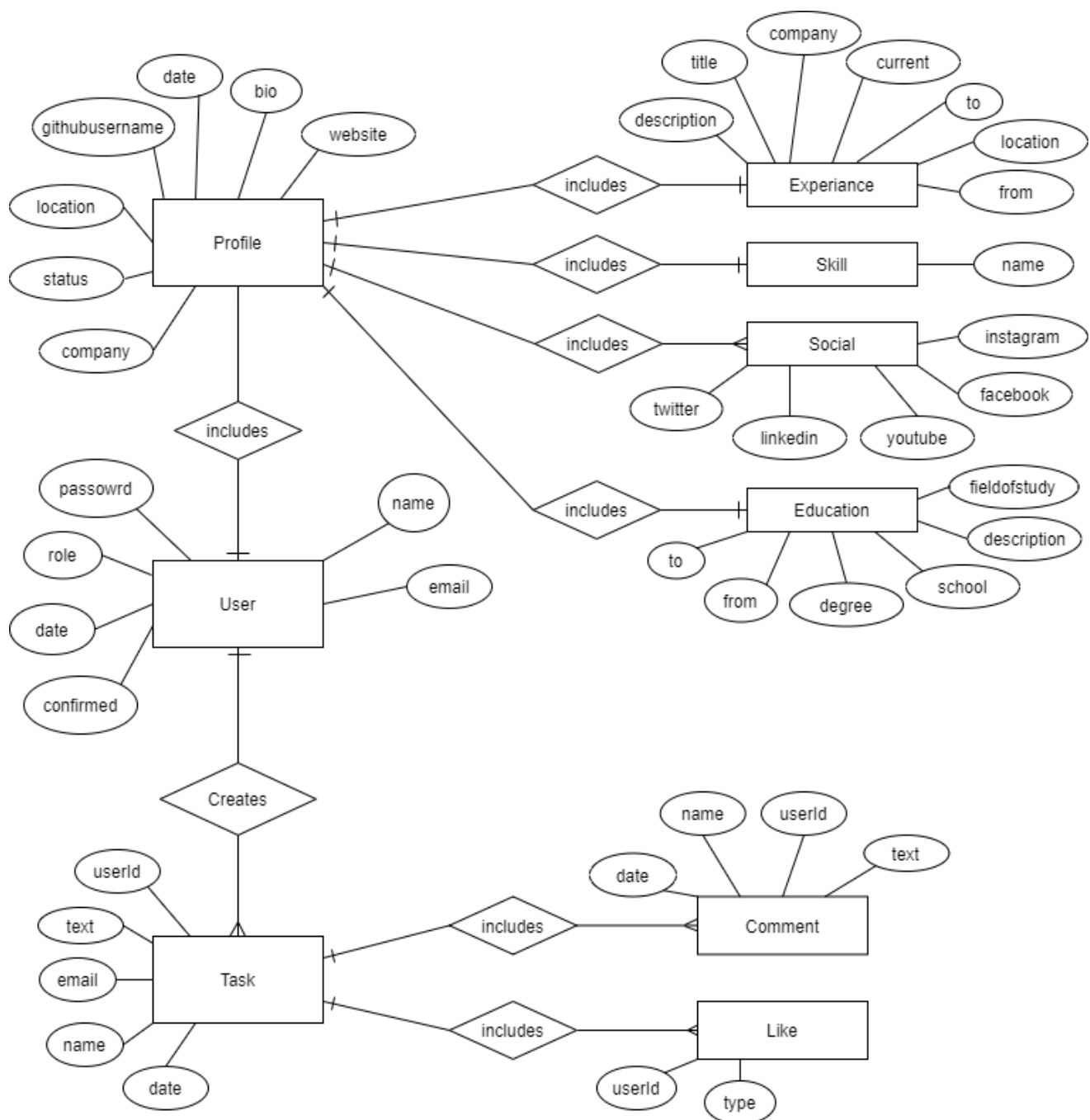


Рисунок 2.2 – Концептуальна модель у вигляді діаграми сутність-зв'язок для розроблюваного веб-додатку

2.3 Рішення з інформаційного забезпечення

2.3.1 Логічна модель

Логічна модель – це загальний погляд на дані, модель даних в певній предметній області. Інколи її називають концептуальною моделлю даних. Такою моделлю і є модель типу сутність–зв'язок.

Програмний пакет ERwin дозволяє створити логічну модель даних сутність–зв'язок у стандарті IDEF1x і на її основі побудувати фізичну базу даних практично для всякої СКБД. Більше того, він дозволяє зробити зворотне перетворення і з фізичної моделі побудувати логічну. Це досить цінна якість. Бази даних у наш час розповсюджені доволі широко і розроблялись вони в різних програмних середовищах. Використовуючи вказану властивість, можна переробити існуючі бази даних, розроблені в середовищі однієї СКБД в логічну модель, а пізніше зробити зворотне перетворення в середовище зовсім іншої СКБД з тих, які підтримує пакет ERwin. Тобто існуючі бази даних можна практично автоматично, не застосовуючи мови програмування, перекодувати в нове програмне середовище.

На логічній моделі можна побачити наступне: сутність, їх атрибути, типи даних, ключові атрибути і зв'язки між сутностями. Для побудови логічної моделі необхідно проаналізувати діаграму сутність–зв'язок та з'ясувати які таблиці та атрибути повинні бути реалізовані в ній. Логічна модель даних додатку представлена на (Рис. 2.3).

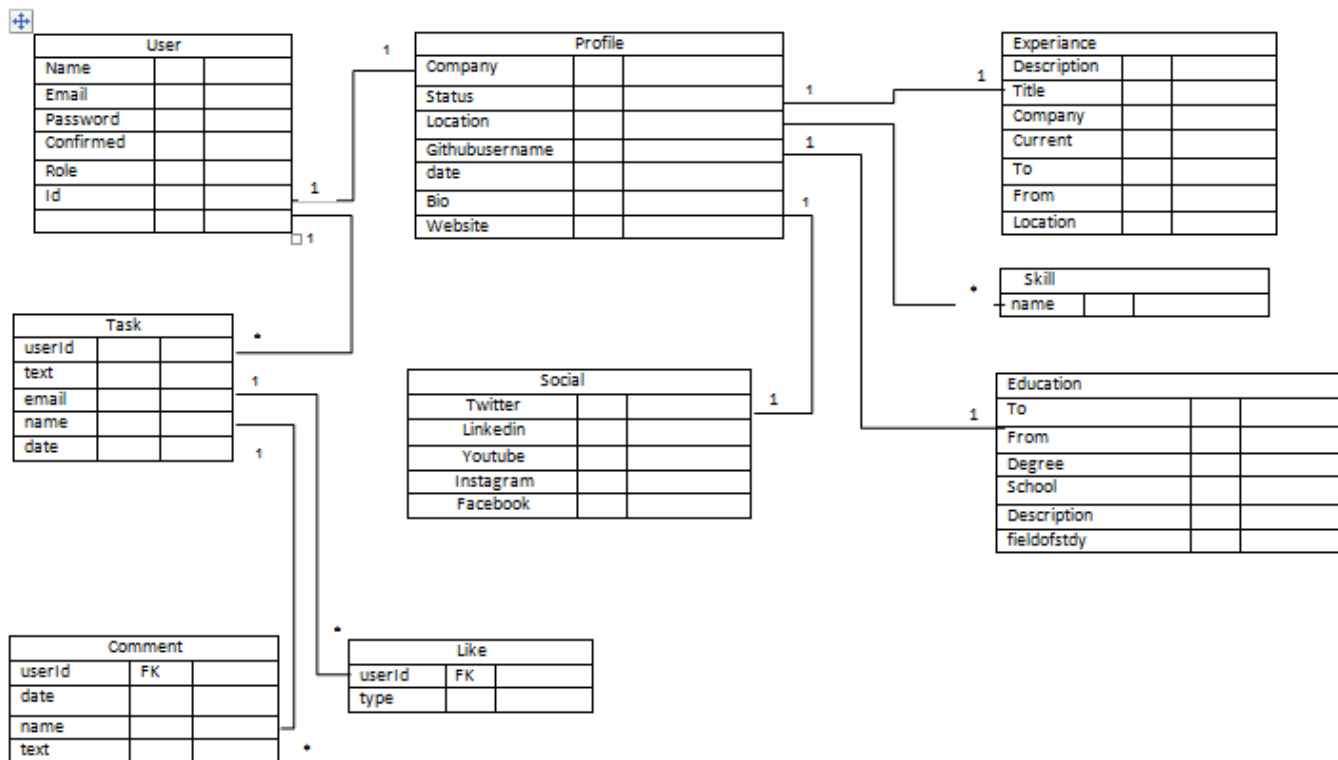


Рисунок 2.3 – Логічна модель даних додатку

2.3.2 Фізична модель

Фізична модель визначає розміщення даних у зовнішній пам'яті. Вона ще називається внутрішньою моделлю системи і форма її представлення залежить від вибраної СКБД. Якщо обрана така СКБД, яка підтримує реляційну модель даних, то треба таблиці разом з атрибутами і зв'язки між таблицями перенести в середовище СКБД з врахуванням вимог до відповідних об'єктів БД. Так, ідентифікатори (імена) таблиць і полів мають задовольняти вимогам СКБД, типи даних, розміри полів, обмеження теж мають бути приведені у відповідність до прийнятих у даній СКБД. Далі визначаються стратегії індексування, а також взаємозв'язки між таблицями, первинні та зовнішні ключі на основі визначених у даталогічній моделі та врахуванням способів їх завдання в обраній СКБД. На етапі фізичного проектування слід приділити особливу увагу забезпеченню цілісності БД. У СКБД цілісність даних забезпечується обмеженнями цілісності,

тобто набором правил, що встановлюють допустимість даних та зв'язків між ними. Наводиться контрольний приклад заповненої БД з урахуванням встановлених обмежень цілісності.

На (Рис. 2.4) наведена фізична модель даних додатку.

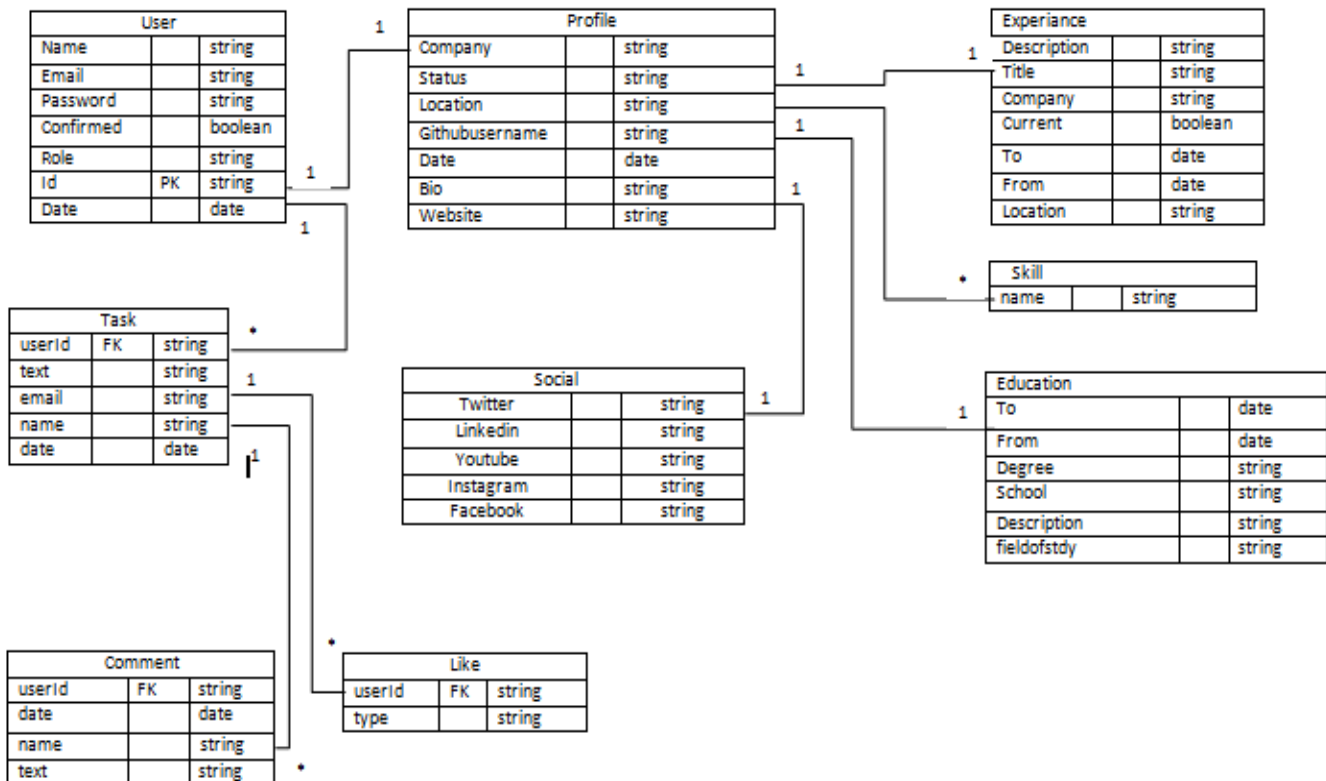


Рисунок 2.4 – Фізична модель даних додатку

2.3.3 Діаграма класів

Діаграма класів – це тип діаграм, які частіше за все використовуються при моделюванні об'єктно-орієнтованих систем. Елементи діаграм класів пов'язані різними структурними зв'язками. Такі діаграми використовують для проектування словника системи чи кооперацій та систем. Діаграми класів мовою UML використовують для того, щоб показати параметри блоків цих діаграм та їх зв'язків. [2]

Діаграма має вигляд символів класів – так званих іконок та зв'язків між ними. Терміном іконка позначають стандартизоване, фіксованої форми,

візуальне зображення (так би мовити, ієрогліф) певного поняття, яке легко розпізнається. Іконка класу має форму прямокутника, який може поділятися на дві або три частини. Верхня його частина обов'язкова, вона містить ім'я класу. Друга й третя частини прямокутника можуть наводитися або пропускатися і містять: друга – список атрибутів класу, третя – список операцій класу. Інакше кажучи, діаграма класу може відображати лише імена класів або імена та відповідні атрибути класів, або імена, атрибути та операції (методи) класів.

Існує два види діаграм класів:

- 1) Статичний вид діаграми розглядає логічні взаємозв'язки класів між собою;
- 2) Аналітичний вид діаграми розглядає загальний вигляд і взаємозв'язку класів, що входять в систему.

Існують різні точки зору на побудову діаграм класів в залежності від цілей їх застосування:

- 3) Концептуальна точка зору – діаграма класів описує модель предметної області, в ній присутні тільки класи прикладних об'єктів;
- 4) Точка зору специфікації – діаграма класів застосовується при проектуванні інформаційних систем;
- 5) Точка зору реалізації – діаграма класів містить класи, використовувані безпосередньо в програмному коді (при використанні об'єктно-орієнтованих мов програмування).

Діаграма класів є однією з форм статичного опису системи з точки зору її проектування, показуючи її структуру. Діаграма класів не відображує динамічну поведінку об'єктів зображених на ній класів. На діаграмах класів показуються класи, інтерфейси і відносини між ними. На (Рис. 2.5) представлена діаграма класів додатку.

Діаграма класів розроблена згідно вимог у середовищі RationalRose.

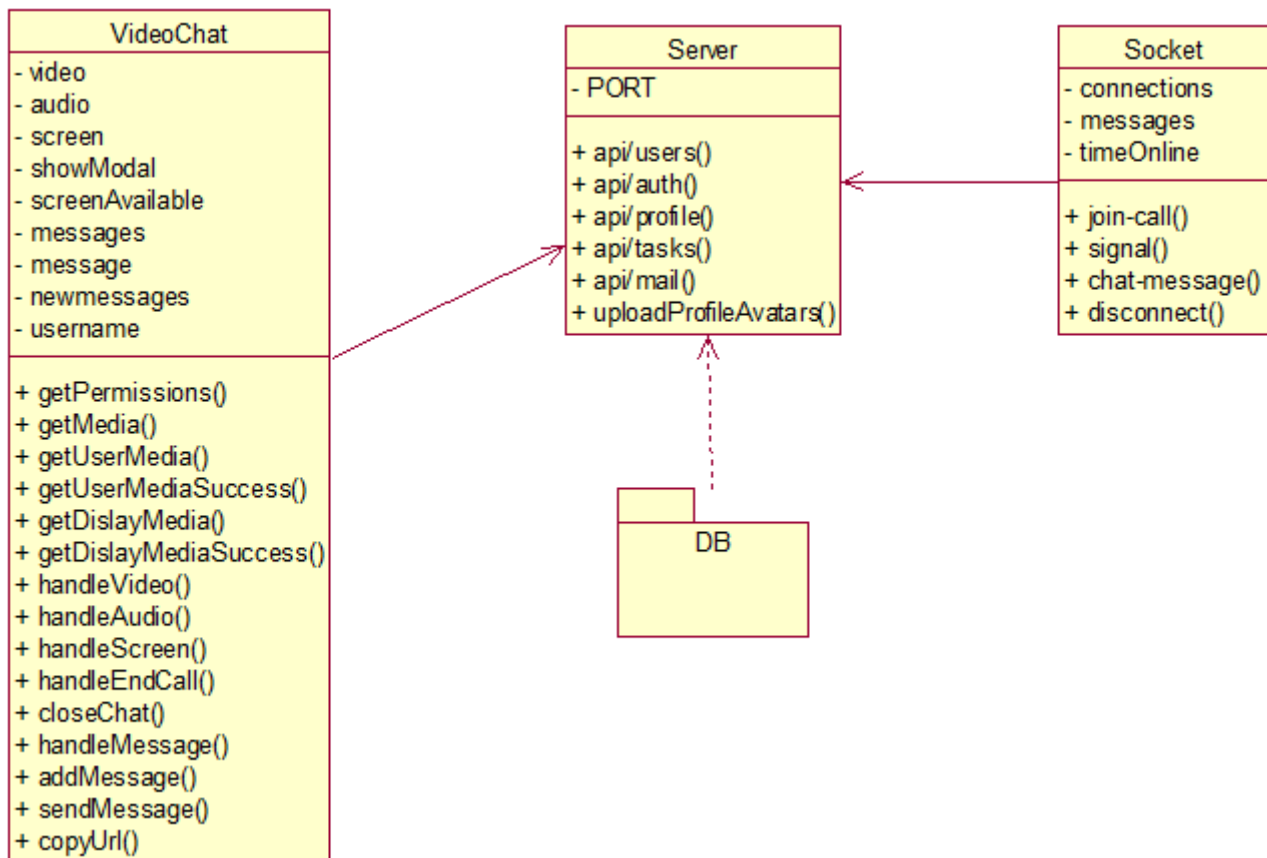


Рисунок 2.5 – Діаграма класів додатку

2.3.4 Діаграма послідовності

Діаграма послідовності відображає взаємодії об'єктів впорядкованих за часом. Зокрема, такі діаграми відображають задіяні об'єкти та послідовність відправлених повідомлень.

Діаграми послідовностей – це відмінний засіб документування поведінки системи, деталізації логіки сценаріїв використання, але є ще один спосіб – використовувати діаграми взаємодії. Діаграма взаємодії показує потік повідомлень між об'єктами системи і основні асоціації між ними і по суті, як вже було сказано вище, є альтернативою діаграми послідовностей. Слід зазначити, що використання діаграми послідовностей або діаграми взаємодії – особистий вибір кожного проектувальника і залежить від індивідуального стилю проектування. На позначеннях, що застосовуються на діаграмі взаємодії,

думаємо, не варто зупинятися детально. Тут все стандартно: об'єкти позначаються прямокутниками з підкресленими іменами (щоб відрізнити їх від класів), асоціації між об'єктами вказуються у вигляді з'єднувальних ліній, над ними може бути зображена стрілка із зазначенням назви повідомлення та його порядкового номера. Необхідність номеру повідомлення пояснюється дуже просто – на відміну від діаграми послідовностей, час на діаграмі взаємодії не показується у вигляді окремого вимірювання.

На (Рис. 2.6) зображено діаграму послідовності для варіанту використання створення профілю співробітника.

Діаграма послідовності розроблена згідно вимог у середовищі RationalRose.

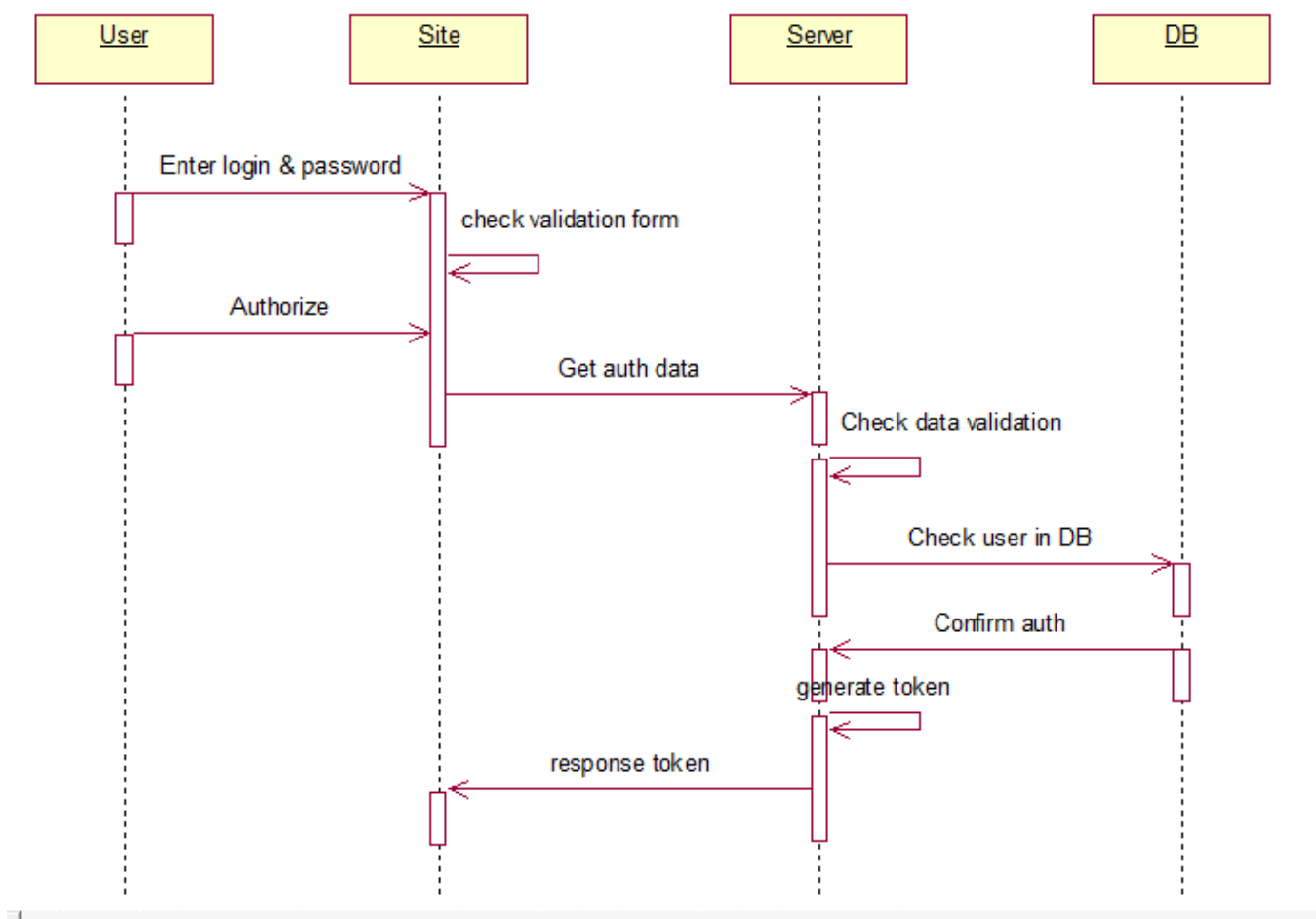


Рисунок 2.6 – Діаграма послідовності для варіанту використання створення профілю співробітника

2.4 Рішення з технічного забезпечення

2.4.1 Опис комплексу технічних засобів

При розробці додатку для організації і контролю робочого процесу компанії використовувались наступні технічні засоби:

- 1) JavaScript – високорівнева мова веб програмування;
- 2) MongoDB – СКБД;
- 3) Visual Studio Code – середа розробки та збору додатків мовою JavaScript;
- 4) XML – мова розмітки що використовується (є офіційним стандартом) для програмування зовнішнього вигляду додатку (дизайну).

2.5 Рішення з математичного забезпечення

2.5.1 Математична модель

Шифрування даних – це метод захисту, в якому інформація кодується і може бути відновлена або розшифрована лише користувачем з правильним ключем дешифрування. Зашифровані дані, також відомі як шифротекст, виглядають зашифрованими або нечитабельними для особи або суб'єкта, які здійснюють доступ без дозволу. Шифрування даних використовується для запобігання доступу до конфіденційних даних зломисниками або небажаними особами. Важлива лінія захисту в архітектурі кібербезпеки – шифрування – робить використання перехоплених даних, настільки складним, настільки можливо. Програмне забезпечення для шифрування даних, також відоме як алгоритм шифрування або шифр, використовується для розробки схеми шифрування, яке теоретично можна обійти тільки з великими обсягами обчислювальної потужності. Шифрування застосовується для зберігання

важливої інформації в ненадійних джерелах та її передачі по незахищеним каналам зв'язку. Така передача даних представляє із себе два взаємно зворотних процеси: перед відправленням даних по лінії зв'язку або перед приміщенням на зберігання вони піддаються зашифровуванню. Хоча дані все ще можуть бути перехоплені, вони будуть незрозумілими і тому не корисними для шпигунів або хакерів. Ключ шифрування/дешифрування можна порівняти з звичайним паролем – наприклад, з тим, що використовується для електронної пошти. Ключ є невід'ємною частиною процесу кодування і декодування даних. Ключі шифрування генеруються таким чином, щоб бути абсолютно унікальними. Ключ шифрування використовується для кодування або декодування даних. Тобто ключ шифрування здатний змішувати дані в нечитабельні символи, і він може повернути ці символи назад до відкритого тексту.

Для даної системи був обраний асиметричний алгоритм RSA через його простоту реалізації та надійність.

На (Рис. 2.7) представлена схема шифрування та дешифрування ключа асиметричним алгоритмом.

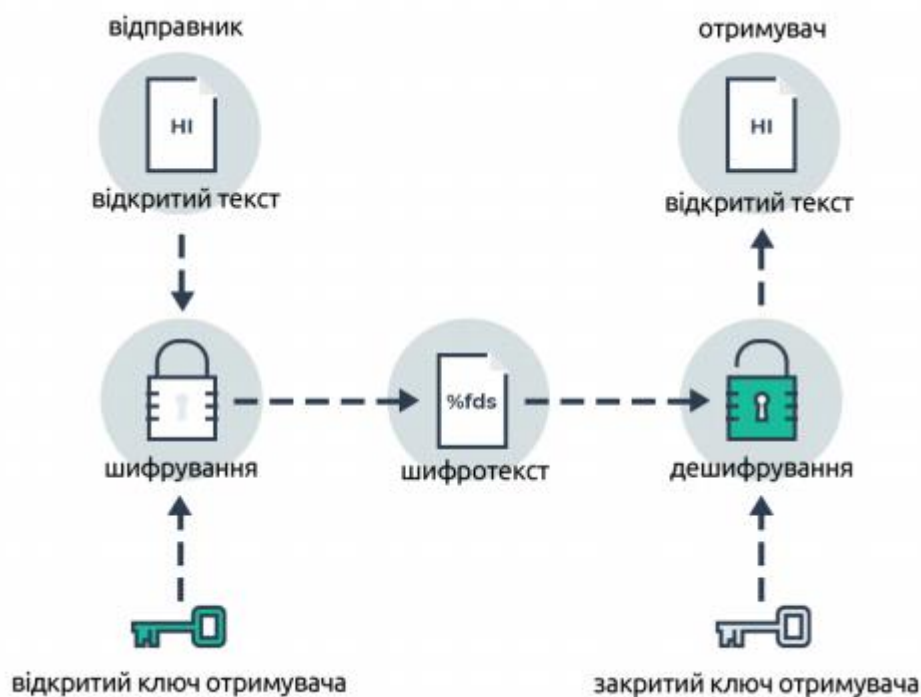


Рисунок 2.7 – Схема шифрування асиметричним алгоритмом

Найбільша перевага використання асиметричних алгоритмів в тому, що ніколи не потрібно надсилати або передавати ключ шифрування або пароль через небезпечний канал. Це знижує можливість взлому.

Для того, щоб згенерувати пари ключів, виконується наступне:

- 1) обираються два великі прості числа p та q кожне приблизно 512б завдовжки;
- 1) обчислюється їх добуток за формулою $n=pq$;
- 2) обчислюється функція Ейлера $\varphi(n) = (p-1)(q-1)$;
- 3) обирається ціле число e , таке, що $1 < e < \varphi(n)$ та e , взаємно просте з $\varphi(n)$;
- 4) за допомогою розширеного алгоритму Евкліда знаходиться число d , таке, що $ed = 1 \pmod{\varphi(n)}$.

Число n називається модулем, а числа e і d – відкритою і секретною експонентами відповідно. Пари чисел (n, e) називаються відкритою частиною ключа, а (n, d) – секретною. Числа p і q після генерації пари ключів не використовуються і можуть бути знищені, але в жодному разі не повинні бути розкриті.

Алгоритм шифрування:

- 1) взяти відкритий ключ (n, e) та повідомлення m ;
- 2) зашифрувати повідомлення $c = E(m) = m^e \pmod n$.

Алгоритм дешифрування:

- 1) взяти закритий ключ (n, d) та повідомлення m ;
- 2) розшифрувати повідомлення $m = D(c) = c^d \pmod n$.

Щоб користувачі додатку могли безпечно спілкуватися у відео–конференціях, не турбуючись про ключ, було прийнято рішення кожного разу генерувати новий ключ користувача в базі в зашифрованому вигляді. Сгенерований ключ копіюється і відправляється іншим співробітникам або співробітнику, який отримуючи ключ та переходячи по ньому потрапляє у приватну конференцію з організатором–відправником ключа. Ключ для шифрування та дешифрування цих ключів зберігатися на сервері не будуть,

також вони не будуть відправлятися на сервер, а шифрування–дешифрування буде відбуватися безпосередню на комп'ютерах користувачів.

2.6 Рішення з програмного забезпечення

2.6.1 Вибір мови проектування

Для розробки програмного забезпечення був обраний фреймворк React з кількох нижче описаних причин.

Компонентно–орієнтований підхід, можливість з легкістю змінювати наявні компоненти і перевикористати код перетворюють React розробку в безперервний процес поліпшення. Компоненти, які були створені під час роботи над тим чи іншим проектом, не мають додаткових залежностей. [3]

Таким чином, ніщо не заважає використовувати їх знову і знову в проектах різного типу. Весь попередній досвід може бути з легкістю застосований при роботі над новим сайтом або навіть при створенні мобільного додатка. Використовуючи передові можливості, такі як Virtual DOM або ізоморфний JavaScript, React розробники можуть з високою швидкістю створювати високопродуктивні додатки, незважаючи на рівень їх складності. [4]

Можливість з легкістю заново використовувати вже наявний код підвищує швидкість розробки, спрощує процес тестування, і, як результат, знижує витрати. Той факт, що ця бібліотека розробляється і підтримується висококваліфікованими розробниками і набирає все більшої популярності з кожним роком, дає підстави сподіватися, що тенденція до подальших поліпшень продовжиться.

Сьогодні JavaScript може виконуватися не тільки в браузері, а й на сервері або на будь-якому іншому пристрої, який має спеціальну програму, що називається «движком» JavaScript. [5]

2.6.2 Модель переходів станів

Модель станів і переходів — схема даних, використовується для дослідження потоку даних конкретних станів. Модель складається із множини станів та переходів між станами.[6]

Діаграма станів відображає скінчений автомат у вигляді графу, вершинами якого є стани об'єкта, поведінка якого моделюється, а переходами – події, які переводять об'єкт, який розглядається, з одного стану в інший. При цьому вважається, що час перебування об'єкта в певному стані набагато більший за час, необхідний для переходу з одного стану в інший, тобто переходи між станами здійснюються миттєво. Кожен стан має ім'я та список внутрішніх дій. В якості імені стану найчастіше використовуються іменники, наприклад: «Введення пароллю».

Список внутрішніх дій містить перелік дій, які виконуються у процесі знаходження системи чи об'єкта в даному стані. Кожна дія відображається у форматі: /, де поле може набувати наступних значень:

- 1) OnEntry – дія виконується під час того, як система входить у даний стан;
- 2) OnExit – дія виконується при виході з даного стану;
- 3) Do – дія виконується під час знаходження в даному стані;
- 4) OnEvent – дія виконується при настанні певної (зовнішньої) події. Графічне представлення стану «Введення пароллю» з трьома внутрішніми діями наведено на рис. . Перехід у даний стан ініціюється при введенні користувачем символів логіну. Для кожного введеного символу система зчитує строку логіну, додає додатковий символ до неї і зберігає її.
- 5) Подія (Event) – це специфікація суттєвого факту, який виникає у часі і просторі. В контексті автоматів подія – це стимул, здатний викликати спрацювання переходу.

- 6) Перехід (Transition) – це відношення між двома станами, яке показує, що об'єкт, який знаходиться у першому стані, повинний виконати деяку подію і перейти у другий стан, як тільки виникне певна подія і будуть виконані задані умови.
- 7) Діяльність (Activity) – це продовження неатомарного обчислення в середині автомату.
- 8) Дія (Action) – це атомарне обчислення, яке приводить до заміни стану або поверненню значення.

У Rational Rose існує можливість створити також один із специфічних станів:

- 1) Вхідний стан – стан, в якому знаходиться система (об'єкт) у початковий момент часу.
- 2) Вихідний стан – стан, в якому знаходиться система (об'єкт) в момент закінчення виконання певної послідовності дій.
- 3) Стан історії – стан, який запам'ятовує дані, що використовувалися при попередньому входженні системи в даний стан. [7]

На (Рис. 2.8) зображено представлення початкового та кінцевого станів на моделі переходів станів у середовищі Rational Rose.



Рисунок 2.8 – Представлення початкового та кінцевого станів

На (Рис. 2.9) представлена модель переходів станів для процесу створення профілю співробітника у додатку.

Діграма переходів станів розроблена згідно вимог у середовищі RationalRose.

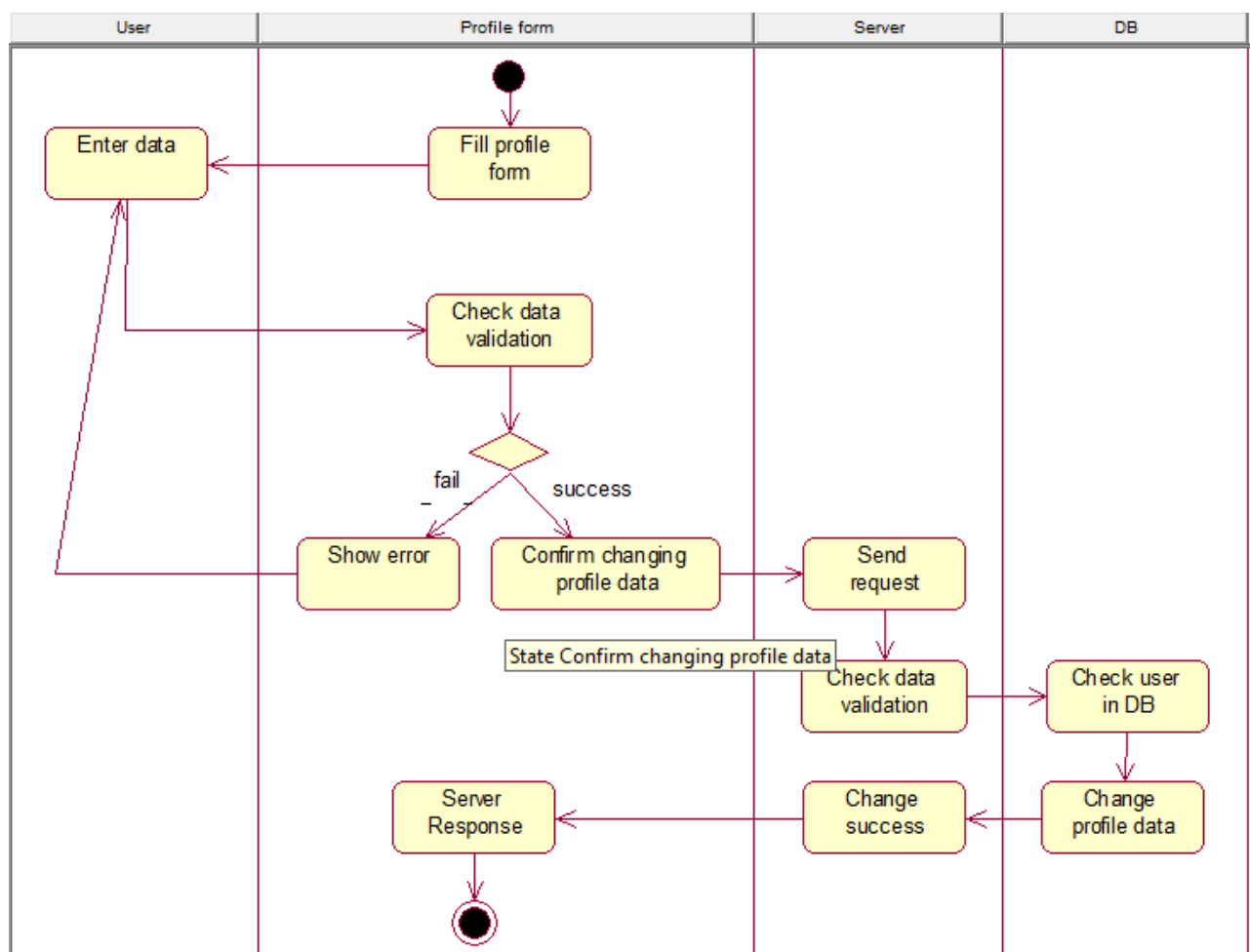


Рисунок 2.9 – Графічне представлення стану створення профілю співробітника в середовищі Rational Rose

Висновки за розділом 2

У даному розділі були спроектовані та побудовані: модель варіантів використання та концептуальна модель, логічна і фізична моделі, модель переходів станів, діаграми послідовності та класів.

3. РОЗРОБКА ПРОЕКТНИХ РІШЕНЬ ЩОДО ВЕБ–ДОДАТКУ ДЛЯ ОРГАНІЗАЦІЇ ТА КОНТРОЛЮ РОБОЧОГО ПРОЦЕСУ КОМПАНІЇ

3.1 Загальносистемні рішення

3.1.1 Представлення схеми організаційної структури

Додаток для організації та контролю робочого процесу компанії не розділяється за правами доступу до функціоналу, у всіх користувачів однакові права. Для того щоб почати роботу у додатку з колегами, новий співробітник повинен спочатку зареєструватися в системі, після чого адміністратор знаходить нового користувача у базі і підтверджує його профіль, цей крок був доданий у цілях безпеки. Адже зареєстрований користувач з непідтвердженим профілем не має доступу до дошок завдань компанії, до профілів співробітників та можливості спілкування з ними. Співробітники компанії виступають головним і основним користувачем додатку, адміністратор додатку не є безпосереднім користувачем, він існує лише для ідентифікації користувачів і надання їм допуску до додатку, що представлено на (Рис. 3.1).

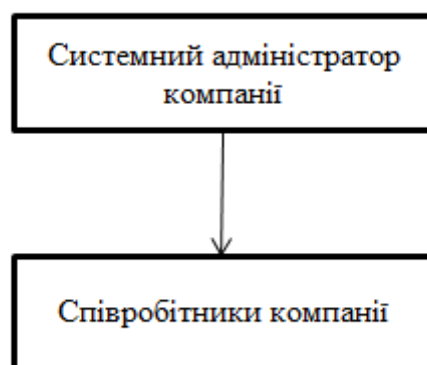


Рисунок 3.1 – Схема організаційної структури

3.1.2 Представлення схеми функціональної структури

Функціональна схема – це схема взаємодії компонентів програмного забезпечення з описом інформаційних потоків, складу даних в потоках. Функціональні схеми більш інформативні, ніж структурні. На (Рис. 3.2) представлена схема функціональної структури для додатку для організації та контролю робочого процесу компанії.[8]



Рисунок 3.2 – Схема функціональної структури

3.1.3 Описання функцій, що автоматизуються

Додаток створений для автоматизації ручної праці та полегшення процесу комунікації в умовах віддаленої роботи співробітників компанії. Автоматизуються наступні процеси:

- 1) процес ідентифікації співробітників;
- 2) процес створення та розподілення задач співробітників;
- 3) процес відстеження прогресу виконання задач;

- 4) процес комунікації співробітників компанії за допомогою доданої можливості проводити дзвінки та ділову переписку у додатку;
- 5) процес ведення бази даних співробітників.

3.1.4 Загальний опис системи

Додаток створений для компаній, переважно ІТ індустрії. Додаток дозволяє:

- 1) додавати, редагувати та видаляти інформацію;
- 2) створювати, переглядати, коментувати та видаляти задачі;
- 3) створювати та вести профіль співробітника, переглядати інформацію та видаляти;
- 4) переглядати інформацію компанії та співробітників;
- 5) проводити онлайн-дзвінки та конференції з високим рівнем шифруванням даних.

Для роботи у додатку створюється профіль системного адміністратора або керуючого безпекою компанії, який є організатором. Спочатку співробітник реєструється, його профіль підтверджується, потім він може виконувати вхід у систему і працювати зі своєю сторінкою, з дошкою задач, сторінкою компанії (на обмежених правах) та заходи у зал конференцій.

3.2 Рішення з організаційного забезпечення

Графічний інтерфейс користувача – тип інтерфейсу, який дозволяє користувачам взаємодіяти з електронними пристроями через графічні зображення та візуальні вказівки, на відміну від текстових інтерфейсів, заснованих на використанні тексту, текстовому наборі команд та текстовій навігації.[9]

Цей термін використовують у багатьох галузях науки й техніки. Його значення належить до будь-якої сполуки взаємочинних сутностей (як природничих, так апаратних і людино–машинних).

Залежно від контексту, поняття застосовне як до окремого елемента (інтерфейс елемента), так і до зв'язка елементів (інтерфейс сполучення елементів). У контексті окремого елемента інтерфейс елемента протилежний до реалізації елемента (внутрішнього устрою та функціонування). Користувачеві елемента немає потреби знати, як реалізовано уживаний елемент, щоб керувати ним, але використовуваний елемент має надати інтерфейс керування.

Інтерфейс додатку повинен бути лаконічним та зрозумілим, щоб користувач міг швидко знайти необхідну інформацію, сторінку чи форму та легко працювати з нею. Інтерфейс повинен бути виконаний у вигляді сторінок з формами та полями.

Висновки за розділом 3

У даному розділі представлені схеми організаційної та функціональної структури. Описані функції, що автоматизувалися у розробці. Виконаний опис системи та розроблене рішення з організаційного забезпечення.

4. РОЗРАХУНОК І ОЦІНКА ЕКОНОМІЧНОГО ЕФЕКТУ ВІД РОЗРОБКИ ВЕБ–ДОДАТКУ ДЛЯ ОРГАНІЗАЦІЇ ТА КОНТРОЛЮ РОБОЧОГО ПРОЦЕСУ КОМПАНІЇ

4.1 Загальна характеристика роботи

Розробка веб–додатку для організації та контролю робочого процесу компанії складається з декількох етапів:

- 1) розробка програмного продукту;
- 2) впровадження програмного продукту.

4.2 Розрахунок витрат на проектування та впровадження системи

4.2.1 Техніко–економічний аналіз

У роботі застосовується метод ФВА для проведення техніко–економічного аналізу розробки. Відповідно цьому варто обирати і систему показників якості програмного продукту.

Технічні вимоги до продукту:

- 1) програмний продукт повинен функціонувати на персональних комп'ютерах із стандартним набором програм;
 - 2) забезпечувати високу швидкість обробки даних у реальному часі;
 - 3) забезпечувати зручність і простоту взаємодії з користувачем або з розробником програмного забезпечення у випадку використання його як модуля;
 - 4) передбачати мінімальні витрати на впровадження програмного продукту.
- Обґрунтування функцій програмного продукту.

Головна функція F0– розробка програмного продукту, який обробляє вхідні

данні та виводить їх для подальшого використання.

Виходячи з конкретної мети, можна виділити наступні основні функції ПП:

F1 – вибір мови програмування;

F2 – вибір оптимальної СКБД;

F3 – інтерфейс користувача.

Кожна з основних функцій може мати декілька варіантів реалізації.

Функція F1:

а) мова програмування Javascript;

б) мова програмування C Sharp;

Функція F2:

а) MSQL Server;

б) Oracle.

Функція F3:

а) інтерфейс користувача, створений за технологією HTML ,CSS, JS;

б) інтерфейс користувача, створений за технологією C++.

Варіанти реалізації основних функцій наведені у морфологічній карті системи (Рис. 2.3). На основі цієї карти побудовано позитивно–негативну матрицю варіантів основних функцій (Таблиця 2.1).

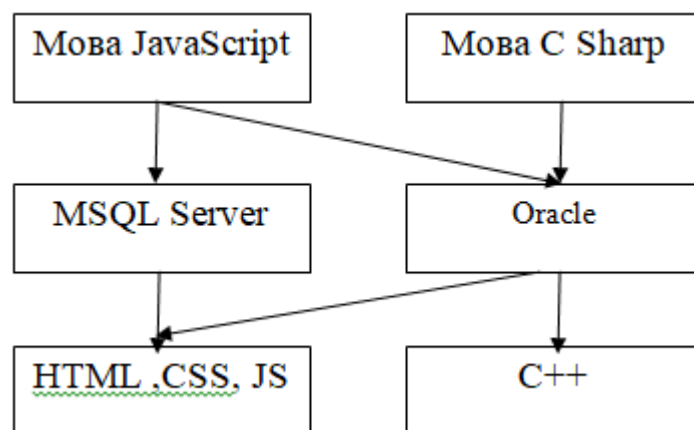


Рисунок 4.1 – Морфологічна карта

Морфологічна карта відображує всі можливі комбінації варіантів реалізації функцій, які складають повну множину варіантів ПП.

Таблиця 4.1 – Позитивно–негативна матриця

Основні функції	Варіанти реалізації	Переваги	Недоліки
F ₁	А	Кроссплатформений	Низька швидкодія
	Б	Займає менше часу при написанні коду	Не кроссплатформений
F ₂	А	Більш дешева вартість корпоративної ліцензії	Необхідність додаткової інсталяції, низький рівень користувацької підтримки
	Б	Подовжений термін користувацької підтримки	Більш висока вартість корпоративної ліцензії. Необхідність додаткової інсталяції
F ₃	А	Простота створення	Необхідність додаткової інсталяції.
	Б	Простота створення, можливість розробки клієнтської і серверної бази на тій-же мові програмування.	Низька швидкодія.

На основі аналізу таблиці робимо висновок, що при розробці програмного продукту деякі варіанти реалізації функцій варто відкинути, тому, що вони не відповідають поставленим перед програмним продуктом задачам. Ці варіанти відзначені у морфологічній карті.

Функція F1: Оскільки розрахунки проводяться з великими об'ємами вхідних даних, то час виконання програмного коду є дуже необхідним, тому

варіант б) має бути відкинтий.

Функція F2: Вибір СКБД не відіграє велику роль у даному програмному продукту, тому вважаємо варіанти а) та б) гідними розгляду.

Функція F3: Оскільки, програмний продукт реалізується на мові Javascript, використовуємо варіант А як єдиний можливий.

4.2.2 Вартісний аналіз

Для визначення вартості розробки ПП спочатку проведемо розрахунок трудомісткості.

Всі варіанти включають в себе два окремих завдання:

- 1) Розробка проекту програмного продукту;
- 2) Розробка програмної оболонки.

Завдання 1 за ступенем новизни відноситься до групи А, завдання 2 – до групи Б. За складністю алгоритми, які використовуються в завданні 1 належать до групи 1; а в завданні 2 – до групи 3.

Для реалізації завдання 1 використовується довідкова інформація, а завдання 2 використовує інформацію у вигляді даних.

Проведемо розрахунок норм часу на розробку та програмування для кожного з завдань. Загальна трудомісткість обчислюється як

$$T_O = T_P \cdot K_P \cdot K_{СК} \cdot K_M \cdot K_{СТ} \cdot K_{СТ.М},$$

де T_P – трудомісткість розробки ПП; K_P – поправочний коефіцієнт; $K_{СК}$ – коефіцієнт на складність вхідної інформації; K_M – коефіцієнт рівня мови програмування; $K_{СТ}$ – коефіцієнт використання стандартних модулів і прикладних програм; $K_{СТ.М}$ – коефіцієнт стандартного математичного забезпечення.

Для першого завдання, виходячи із норм часу для завдань розрахункового характеру ступеню новизни А та групи складності алгоритму 1, трудомісткість дорівнює: $T_P = 60$ людино-днів. Поправочний коефіцієнт, який враховує вид

нормативно–довідкової інформації для першого завдання: $K_{\Pi} = 1.7$.

Поправочний коефіцієнт, який враховує складність контролю вхідної та вихідної інформації для всіх семи завдань рівний 1: $K_{СК} = 1$. Оскільки при розробці першого завдання використовуються стандартні модулі, врахуємо це за допомогою коефіцієнта $K_{СТ} = 0.8$. Тоді, за формулою, загальна трудомісткість програмування першого завдання дорівнює:

$$T_1 = 60 \cdot 1.7 \cdot 0.8 = 81.6 \text{ людино–днів.}$$

Проведемо аналогічні розрахунки для подальших завдань.

Для другого завдання (використовується алгоритм третьої групи складності, степінь новизни Б), тобто $T_P = 27$ людино–днів, $K_{\Pi} = 0.9$, $K_{СК} = 1$, $K_{СТ} = 0.8$:

$$T_2 = 27 \cdot 0.9 \cdot 0.8 = 19.44 \text{ людино–днів.}$$

Складаємо трудомісткість відповідних завдань для кожного з обраних варіантів реалізації програми, щоб отримати їх трудомісткість:

$$T_I = (81.6 + 19.44 + 4.8 + 19.44) \cdot 8 = 1002.24 \text{ людино–годин;}$$

$$T_{II} = (81.6 + 19.44 + 6.91 + 19.44) \cdot 8 = 1019.12 \text{ людино–годин;}$$

Найбільш високу трудомісткість має варіант II.

В розробці бере участь програміст з окладом 20 000 грн. та фінансовий аналітик з окладом 10 500 грн. Визначимо зарплату за годину за формулою:

$$СЧ = M / Tm \cdot t \text{ грн.,}$$

де M – місячний оклад працівників; Tm – кількість робочих днів тиждень; t – кількість робочих годин в день.

$$Сч = 20\,000 + 10\,500/3 \cdot 21 \cdot 8 = 60,51 \text{ грн.}$$

Тоді, розрахуємо заробітну плату за формулою:

$$С_{ЗП} = Сч \cdot Ti \cdot K_d,$$

де $Сч$ – величина погодинної оплати праці програміста; Ti – трудомісткість відповідного завдання; K_d – норматив, який враховує додаткову заробітну плату.

Зарплата розробника за варіантами становить:

$$I. С_{ЗП} = 60,51 \cdot 1002.24 \cdot 1.2 = 72774,65 \text{ грн.}$$

$$II. С_{ЗП} = 60,51 \cdot 1019.12 \cdot 1.2 = 74000,34 \text{ грн.}$$

Відрахування на єдиний соціальний внесок в залежності від групи

професійного ризику (II клас) становить 22%:

$$I. C_{\text{ВІД}} = C_{\text{ЗП}} \cdot 0,22 = 72774,65 \cdot 0,22 = 16003,8 \text{ грн.}$$

$$II. C_{\text{ВІД}} = C_{\text{ЗП}} \cdot 0,22 = 74000,34 \cdot 0,22 = 16280,07 \text{ грн.}$$

Тепер визначимо витрати на оплату однієї машино-години. (C_M)

Так як одна ЕОМ обслуговує програміста з окладом 20 000 грн., з коефіцієнтом зайнятості 0,2 то витрати на обслуговування машини будуть:

$$C_T = 12 \cdot M \cdot K_3 = 12 \cdot 20\,000 \cdot 0,2 = 48000 \text{ грн.}$$

З урахуванням додаткової заробітної плати:

$$C_{\text{ЗП}} = C_T \cdot (1 + K_3) = 48000 \cdot (1 + 0,2) = 57600 \text{ грн.}$$

Відрахування на єдиний соціальний внесок:

$$C_{\text{ВІД}} = C_{\text{ЗП}} \cdot 0,22 = 57600 \cdot 0,22 = 12672 \text{ грн.}$$

Амортизаційні відрахування розраховуємо при амортизації 25% та вартості ЕОМ – 10 000 грн.

$$C_A = K_{\text{ТМ}} \cdot K_A \cdot C_{\text{ПР}} = 1,15 \cdot 0,25 \cdot 10\,000 = 2875 \text{ грн.,}$$

де $K_{\text{ТМ}}$ – коефіцієнт, який враховує витрати на транспортування та монтаж приладу у користувача; K_A – річна норма амортизації; $C_{\text{ПР}}$ – договірна ціна приладу.

Витрати на ремонт та профілактику розраховуємо як:

$$C_P = K_{\text{ТМ}} \cdot C_{\text{ПР}} \cdot K_P = 1,15 \cdot 10\,000 \cdot 0,05 = 575 \text{ грн.,}$$

де K_P – відсоток витрат на поточні ремонти.

Ефективний годинний фонд часу ПК за рік розраховуємо за формулою:

$$T_{\text{ЕФ}} = (D_K - D_B - D_C - D_P) \cdot t_z \cdot K_B = (365 - 104 - 8 - 16) \cdot 8 \cdot 0,9 = 1706,4$$

годин, де D_K – календарна кількість днів у році; D_B , D_C – відповідно кількість вихідних та святкових днів; D_P – кількість днів планових ремонтів устаткування; t_z – кількість робочих годин в день; K_B – коефіцієнт використання приладу у часі протягом зміни.

Витрати на оплату електроенергії розраховуємо за формулою:

$$C_{\text{ЕЛ}} = T_{\text{ЕФ}} \cdot N_C \cdot K_3 \cdot C_{\text{ЕН}} = 1706,4 \cdot 0,156 \cdot 0,9733 \cdot 2,0218 = 404,18 \text{ грн.,}$$

де N_C – середньо-споживча потужність приладу; K_3 – коефіцієнтом зайнятості приладу; $C_{\text{ЕН}}$ – тариф за 1 кВт-годин електроенергії.

Накладні витрати розраховуємо за формулою:

$$C_H = C_{\text{ПР}} \cdot 0.67 = 10\,000 \cdot 0.67 = 6700 \text{ грн.}$$

Тоді, річні експлуатаційні витрати будуть:

$$C_{\text{ЕКС}} = C_{\text{ЗП}} + C_{\text{ВІД}} + C_A + C_P + C_{\text{ЕЛ}} + C_H$$

$$C_{\text{ЕКС}} = 57600 + 12672 + 2875 + 575 + 404.18 + 6700 = 80826.18 \text{ грн.}$$

Собівартість однієї машино–години ЕОМ дорівнюватиме:

$$C_{\text{М-Г}} = C_{\text{ЕКС}} / T_{\text{ЕФ}} = 80826.18 / 1706.4 = 47.37 \text{ грн/час.}$$

Оскільки в даному випадку всі роботи, які пов'язані з розробкою програмного продукту ведуться на ЕОМ, витрати на оплату машинного часу, в залежності від обраного варіанта реалізації, складає:

$$C_M = C_{\text{М-Г}} \cdot T$$

$$\text{I. } C_M = 47.37 \cdot 1002.24 = 47476.1 \text{ грн.};$$

$$\text{II. } C_M = 47.37 \cdot 1019.12 = 48275.7 \text{ грн.};$$

Накладні витрати складають 67% від заробітної плати:

$$C_H = C_{\text{ЗП}} \cdot 0.67$$

$$\text{I. } C_H = 72774.65 \cdot 0.67 = 48738.91 \text{ грн.};$$

$$\text{II. } C_H = 74000.34 \cdot 0.67 = 49580.23 \text{ грн.};$$

Отже, вартість розробки ПП за варіантами становить:

$$C_{\text{ПП}} = C_{\text{ЗП}} + C_{\text{ВІД}} + C_M + C_H$$

$$\text{I. } C_{\text{ПП}} = 72774.65 + 16003.8 + 47476.1 + 48738.91 = 184\,936.46 \text{ грн.};$$

$$\text{II. } C_{\text{ПП}} = 74000.34 + 16280.07 + 48275.7 + 49580.23 = 188136.34 \text{ грн.};$$

Вибір кращого варіанта ПП техніко–економічного рівня.

Розрахуємо коефіцієнт техніко–економічного рівня за формулою:

$$K_{\text{ТЕРj}} = K_{\text{Кj}} / C_{\text{Фj}},$$

$$K_{\text{ТЕР1}} = 2.57 / 184\,936.46 = 1.39 \cdot 10^{-5};$$

$$K_{\text{ТЕР2}} = 1.89 / 188136.34 = 1.005 \cdot 10^{-5}.$$

Як бачимо, найбільш ефективним є перший варіант реалізації програми з коефіцієнтом техніко–економічного рівня $K_{\text{ТЕР1}} = 1.39 \cdot 10^{-5}$.

Витрати на впровадження веб–додатку (B_B) складаються з двох складових:

- витрати на реєстрацію доменного імені (B_{B1}) на 1 рік;
- витрати на реєстрацію в пошукових системах (B_{B2}) (наприклад: Yandex, Google, Bing і т.п.).

Середня вартість реєстрації доменного імені (B_{B1}) з закінченням .COM.UA дорівнює 321 грн./ 1 рік.

Реєстрація в пошукових системах (B_{B2}) безкоштовна, але витрати на оплату SEO працівника за дану процедуру становлять всередньому 200 грн.

Таким чином розрахуємо витрати на впровадження:

$B_B = B_{B1} + B_{B2} = 321 + 200 = 521$ грн. вартість впровадження веб-додатку в системі Internet.

4.3 Розрахунок економічної ефективності розробки і впровадження системи

Економічна ефективність за рік (E_p) визначається як сукупність коштів, вивільнених за рахунок впровадження системи, у данному випадку веб-додатку для організації та контролю робочого процесу компанії, це можуть бути кошти отримані за рахунок продажу продукту. Так як основною метою розробки веб-додатку для організації та контролю робочого процесу компанії була його безкоштовність, за його використання не буде збиратися оптала, тому розрахунок економічної ефективності не є доцільним.

Можна розрахувати приблизну ефективність розробки, якщо уявити що розроблений додаток буде продаватися компаніям за умови покупки доступу до функції продукту на рік, тобто підписка на рік, вартістю 850 долларів (середня вартість аналогів на ринку пропозицій).

Якщо середній курс доллара становитиме 28 грн, тоді вартість підписки (ВП) на рік у гривні складатиме $850 \cdot 28 = 23800$ грн.

Вартість підтримки (ВПП) продукту, тобто обслуговування та вдосконалення, можна розрахувати як погодинну оплату розробника та аналітика, якщо у рік на підтримку продукту відводиться 200 год – 3.84

години на робочий тиждень, то вартість підтримки у рік складатиме:

$$\text{ВПП} = \text{ЗП}_{\text{Год}} \cdot 200 = 60,51 \cdot 200 = 12102 \text{ грн.}$$

Якщо за перший рік підписку придбають 20 компаній, то прибуток від продажу (ПВП) складатиме:

$$\text{ПВП} = \text{ВП} \cdot 50 = 23800 \cdot 20 = 476\,000 \text{ грн.}$$

Звідси чистий або повний прибуток (ПП) з вирахуванням усіх витрат на розробку та впровадження складатиме:

$$\text{ПП} = \text{ПВП} - \text{С}_{\text{ПП}} - \text{ВПП} - \text{В}_{\text{В}} = 476\,000 - 184\,936.46 - 12102 - 521 = 278\,440,54 \text{ грн.}$$

Наведений розрахунок це приклад можливого прогнозування ситуації у випадку успішного продажу продукту. Звісно можливий і негативний прогноз, якщо продаж продукту не буде успішним, то продукт за перший рік може навіть не компенсувати витрати на розробку. Реальне прогнозування можливе лише в умовах виходу продукту на ринок і подальшому аналізі попиту на нього та пропозицій конкурентів.

Висновки за розділом 4

У даному розділі був виконаний розрахунок витрат на проектування та впровадження системи, що розроблялася. Проведений розрахунок економічної ефективності розробки.

5. ОХОРОНА ПРАЦІ

5.1 Вступ

Питання охорони праці та здоров'я наших громадян у процесі їх трудової та будь-якої іншої діяльності повинні стати пріоритетними та увійти до розряду питань найвищого державного рівня, оскільки саме люди, їх життя та здоров'я є найбільшим багатством будь-якої держави.

Окрім соціального, охорона праці має, безперечно, важливе економічне значення – це й висока продуктивність праці, зниження витрат на оплату лікарняних, компенсацій за важкі та шкідливі умови праці тощо.

Охорона праці – це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних та лікувально-профілактичних заходів і засобів, спрямованих на збереження здоров'я та працездатності людини в процесі праці.

Законодавчими актами, що визначають основні положення про охорону праці, є загальні закони України, а також спеціальні законодавчі акти. До загальних законів, що визначають основні положення про охорону праці належать: Конституція України, Закони України «Про охорону праці», «Про охорону здоров'я», «Про пожежну безпеку», «Про використання ядерної енергії та радіаційний захист», «Про забезпечення санітарного та епідемічного благополуччя населення», «Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності», Кодекс законів про працю України (КЗпП). Спеціальними законодавчими актами в галузі охорони праці є Державні нормативні акти про охорону праці, Державні стандарти Системи стандартів безпеки праці, будівельні норми та правила, санітарні норми, правила технічної експлуатації електроустановок споживачів та інші нормативні документи.

В статті 43 Конституції України записано: «Кожен має право на працю, що включає можливість заробляти собі на життя працею, яку він вільно обирає, або на яку вільно погоджується», «Кожен має право на належні, безпечні і здорові умови праці, на заробітну плату, не нижчу від визначеної законом», «Використання праці жінок і неповнолітніх на небезпечних для їхнього здоров'я роботах забороняється».

Основоположним законодавчим документом у галузі охорони праці є Закон України «Про охорону праці», дія якого поширюється на всі підприємства, установи і організації незалежно від форм власності та видів їх діяльності, на усіх громадян, які працюють, а також залучені до праці на цих підприємствах.

5.2 Аналіз небезпечних і шкідливих факторів

У приміщенні по розрахунках з побутовими і промисловими споживачами в обчислювальному центрі виробничого управління теплової енергії можуть мати місце такі фізичні і психологічні шкідливі фактори, як – порушення стану мікроклімату, недостатня освітленість робочої зони, забруднення повітря на робочих місцях, виробничий шум та вібрація, електромагнітні випромінювання, електростатичні поля, іонний склад повітря, відсутність чи недолік природного світла, поразка електричним струмом, загоряння, монотонність праці, перенапруга очей, емоційні перевантаження.

5.2.1 Освітленість робочого місця

Норми освітлення залежать від параметрів, які передбачено роботою. Відстань від очей до предмета праці повинна бути визначена в кожному окремому випадку. Що менше відношення діаметра деталі до відстані від очей, то інтенсивнішим повинно бути освітлення. При цьому необхідно урахувати й

здатність поверхні відбивати світло. Спектр джерел світла повинен максимально наближатися до спектра сонячного випромінювання. Важливо також захистити очі робітники від сліпучого світла. Усі системи освітлення повинні забезпечувати правильне сприйняття відтінків світла, аби в робочих приміщеннях було рівномірне освітлення. Тому слід подбати про загальне та місцеве. Освітлювальні пристрої мають забезпечувати гігієнічні вимоги: освітлення, якого було б достатньо для виконання певної роботи без напруження зору; рівномірність освітлення, без тіней, у межах робочої поверхні, рівень освітлення проходів; захист очей від блиску; виконання вимог безпеки (шляхом обладнання в окремих випадках аварійного освітлення).

Нормативні величини освітленості робочих місць для різних видів робіт та відповідних зорових навантажень визначаються ДБН Б.2.5.–28–2006 «Природне і штучне освітлення». Для роз'яснення зазначимо, що робоча поверхня – головний об'єкт при встановленні регламентованих норм освітлення. Під робочою поверхнею, як об'єкта для нормування рівнів освітленості, розуміють поверхню робочого столу, верстака, частини обладнання, або інструмента, на якій проводиться робота та для якої нормується або на якій вимірюється освітленість.

5.2.2 Вплив випромінювання

Джерела енергії ЕМП радіочастотного діапазону поділяються на технологічні (основні) та додаткові. До технологічних належать плавильні або гартувальні контури, пластини конденсаторів, фідерні лінії. У радіотехнічних пристроях це генератори та ЗВЧ–блоки, антенні системи, елементи хвилеводних трактів. До додаткових джерел належать виносні трансформатори, батареї конденсаторів змінного струму. У радіотехнічних пристроях додатковими джерелами є неякісно екрановані ВЧ–елементи передатчиків і пристроїв

складання потужностей та роздільних фільтрів, неекрановані лінії передачі електромагнітної енергії на антени.

Напруга електричного поля вимірюється у вольтах на метр – В/м, а магнітного поля – в амперах на метр – А/м. Інтенсивність електромагнітного поля з різними хвилями, що діють на працівника, оцінюється за величиною щільності потоку енергії, яка падає на одиницю поверхні, і виражається у ватах на квадратний метр (Вт/м²) або в довільних одиницях: міліватах, мікроватах на квадратний сантиметр (мВт/см², мкВт/см²).

Електромагнітні поля особливо негативно впливають на організм людини, яка безпосередньо працює з джерелом випромінювання. В діапазоні промислових частот більше негативний вплив на біологічний об'єкт має електрична складова поля.

Найчутливішими до ЕМП є нейродинамічні процеси, які прямо чи побічно перемикають хронобіологічні процеси організму на патологічний або стресовий режим функціонування.

При дії ЕМП на людину можливі гострі та хронічні форми порушення фізіологічних функцій організму. Такі порушення виникають в результаті дії електричної складової ЕМП на нервову систему, а також на структуру кори головного та спинного мозку, серцево–судинної системи.

У більшості випадків такі зміни в діяльності нервової та серцево–судинної системи мають зворотній характер, але в результаті тривалої дії вони накопичуються, підсилюються з плином часу, але, як правило, зменшуються та зникають при виключенні впливу та поліпшенні умов праці. Тривалий та інтенсивний вплив ЕМП призводить до стійких порушень в організмі людини та захворювань.

Сумісна дія випромінювань широкого діапазону може викликати окрему радіохвильову хворобу.

Тяжкість її наслідків прямо залежить від напруженості ЕМП, фізичних особливостей різних діапазонів частот, тривалості впливу, умов навколишнього середовища, а також від функціонального стану та стійкості організму до впливу

різних чинників, можливостей адаптації. Збільшується ризик виникнення загальних захворювань, захворювань органів дихання, травлення тощо. Це може відбуватися також і за дуже невеликої інтенсивності ЕМП, яка незначно перевищує гігієнічні нормативи.

Результатом дії на організм людини електромагнітних випромінювань в діапазоні 30 кГц – 300 МГц є: загальна слабкість, підвищена втома, порушення сну, головний біль та біль в ділянці серця. З'являється роздратованість, втрачається увага, сповільнюються рухово–мовні реакції.

5.2.3 Вплив електричного струму

Під час монтажу та експлуатації ліній електромережі необхідно повністю унеможливити виникнення електричного джерела згорання внаслідок короткого замикання та перевантаження проводів, обмежувати застосування проводів з легкозаймистою ізоляцією і, за можливість, перейти на негорючу ізоляцію.

Усі провідники повинні відповідати номінальним параметрам мережі та навантаження, умовам навколишнього середовища, умовам розподілу провідників, температурному режиму та типам апаратури захисту, вимогам ПВЕ.

У приміщенні, де одночасно експлуатується або обслуговується більше п'яти персональних ПК, на помітному та доступному місці встановлюється аварійний резервний вимикач, який може повністю вимкнути електричне живлення приміщення, крім освітлення.

Протікання струму через тіло людини супроводжується термічним, електролітичним та біологічним ефектами.

Термічна дія струму полягає в нагріванні тканини, випаровуванні вологи, що викликає опіки, обвуглювання тканин та їх розриви парою. Тяжкість термічної дії струму залежить від величини струму, опору проходженню струму та часу проходження. За короткочасної дії струму термічна складова може бути визначальною в характері і тяжкості ураження.

Електролітична дія струму проявляється в розкладі органічної речовини (її електролізі), в тому числі і крові, що призводить до зміни їх фізико–хімічних і біохімічних властивостей. Останнє, в свою чергу, призводить до порушення біохімічних процесів у тканинах і органах, які є основою забезпечення життєдіяльності організму.

Біологічна дія струму проявляється у подразненні і збуренні живих тканин організму, в тому числі і на клітинному рівні. При цьому порушуються внутрішні біоелектричні процеси, що протікають в організмі, який нормально функціонує, і пов'язані з його життєвими функціями. Збурення, спричинене подразнюючою дією струму, може проявлятися у вигляді мимовільного непередбачуваного скорочення м'язів.

5.2.4 Виробничий шум

Шум – одна з форм фізичної дії середовища, безпосередньо впливає на працездатність.

Джерелами шуму є: транспорт, насоси, двигуни, пневматичні та електричні інструменти, верстати, будівельна техніка. У нашому випадку також додається шум, який утворює людський натовп, оскільки торгівельні організації підприємства розміщені на ринках.

Для забезпечення нормованих рівнів шуму у виробничих приміщеннях та на робочих місцях застосовуються шумопоглинальні засоби, вибір яких обґрунтовується спеціальними інженерно–акустичними розрахунками.

Як засоби шумопоглинання повинні застосовуватися негорючі або важкогорючі спеціальні перфоровані плити, панелі, мінеральна вата з максимальним коефіцієнтом звукопоглинання в межах частот 31,5 – 8000 Гц, або інші матеріали аналогічного призначення, дозволені для оздоблення приміщень органами державного санітарно–епідеміологічного нагляду. Крім того, необхідно застосовувати підвісні стелі з аналогічними властивостями.

Рівні вібрації під час виконання робіт з ПК у виробничих приміщеннях не повинні перевищувати допустимих значень, визначених в СН 3044–84 та ДСТУ 12.1.012–90 «Вибрационная безопасность. Общие требования».

Для зниження вібрації обладнання, пристрої, пристосування необхідно встановлювати на спеціальні амортизуючі прокладки, передбачені нормативними документами.

5.2.5 Умови мікроклімату

Суттєвий вплив на стан організму працівника, його працездатність здійснює мікроклімат у приміщенні, що визначається діючою на організм людини сукупністю температури, волоДСТУі, руху повітря та теплового випромінювання нагрітих поверхонь.

Приміщення по розрахунках з побутовими і промисловими споживачами в обчислювальному центрі виробничого управління теплової енергії повинно бути обладнане системами опалення, кондиціонування повітря або припливно–витяжною вентиляцією відповідно до СНиП 2.04.05–91 «Вентиляция производственных помещений».

Параметри мікроклімату, іонного складу повітря, вміст шкідливих речовин на робочих місцях, оснащених відеотерміналами, повинні відповідати вимогам пункту 2.4 СН 4088–86 «Санитарные нормы микроклимат производственных помещений», ДСТУ 12.1.005–88 «ССБТ Общие санитарно–гигиенические требования к воздуху рабочей зоны» (таблиця 5.1), СН 2152–80 «Санитарно–гигиенические нормы допустимых уровней ионизации воздуха производственных и общественных помещений» (таблиця 5.2).

Таблиця 5.1 – Нормовані параметри мікроклімату для приміщень з ВДТ та ППК

Пор а	Категорія робіт згідно з	Темпера тур а повітря,	Відносна вологість	Швидкість руху повітря,
----------	-----------------------------	---------------------------	-----------------------	----------------------------

рок у	ДСТУ 12.1–005– 88	°С оптимальна	повітря, % оптимальна	м/с оптимальна
Хол одна	Легка – 1 а	22–24	40–60	0,1
	Легка – 1 б	21–23	40–60	0,1
Теп ла	Легка – 1 а	23–25	40–60	0,1
	Легка – 1 б	22–24	40–60	0,2

Таблиця 5.2 – Рівні іонізації повітря приміщень при роботі на ВДТ та 111К (відповідно до СН 2152–80)

Рівні	Кількість іонів в 1 см ³ повітря	
	n ⁺	n ⁻
Мінімально необхідні	400	600
Оптимальні	1500 – 3000	3000 – 5000
Максимально допустимі	50000	50000

Для підтримки допустимих значень мікроклімату та концентрації позитивних та негативних іонів необхідно передбачати установки або прилади зволоження та/або штучної іонізації, кондиціювання повітря.

5.2.6 Організація робочого місця

Робоче місце – це місце постійного або тимчасового перебування працівника в процесі трудової діяльності. Організація робочого місця користувача приміщення по розрахунках з побутовими і промисловими споживачами в обчислювальному центрі Виробничого управління теплової енергії повинна забезпечувати відповідність усіх елементів робочого місця та їх розташування ергономічним вимогам ДСТУ 12.2.032–78 «ССБТ. Рабочее место

при выполнении работ сидя. Общие эргономические требования».

Площа, виділена для одного робочого місця з відео терміналом або персональною ПК, повинна складати не менше 6 м², а обсяг – не менше 20 м³.

Робочі місця з відео терміналами відносно світлових прорізів повинні розміщуватися так, щоб природне світло падало збоку, переважно зліва.

При розміщенні робочих місць необхідно дотримуватись таких вимог: робочі місця розміщуються на відстані не менше 1 м від стін зі світловими прорізами;

- 1) відстань між бічними поверхнями відео терміналів має бути не меншою за 1,2 м;
- 2) відстань між тильною поверхнею одного відео термінала та екраном іншого не повинна бути меншою 2,5 м;
- 3) прохід між рядами робочих місць має бути не меншим 1 м.

Вимоги щодо відстані між бічними поверхнями відео терміналів та відстані між тильною поверхнею одного відео термінала та екраном іншого враховуються також при розміщенні робочих місць з відео терміналами та персональними комп'ютерів в суміжних приміщеннях, з урахуванням конструктивних особливостей стін та перегородок.

Висота робочої поверхні столу для відео термінала має бути в межах 680–800 мм, а ширина – забезпечувати можливість виконання операцій в зоні досяжності моторного поля.

Робочий стіл для відео термінала, як правило, має бути обладнаний підставкою для ніг шириною не менше 300 мм та глибиною не менше 400 мм, з можливістю регулювання по висоті в межах 150 мм та кута нахилу опорної поверхні – в межах 20°. Підставка повинна мати рифлену поверхню та бортик на передньому краї заввишки 10 мм.

Робоче сидіння (стілець, крісло) користувача відео термінала та персональної ПК повинно мати такі основні елементи: сидіння, спинку та стаціонарні або знімні підлокітники, також повинно бути підйомно–поворотним, таким, що регулюється за висотою, кутом нахилу сидіння та спинки, за

відстанню спинки до переднього краю сидіння, висотою підлокітників. Поверхня сидіння має бути плоскою, передній край – заокругленим. Висота спинки сидіння має становити 300 ± 20 мм, ширина – не менше 380 мм, радіус кривизни в горизонтальній площині – 400 мм. Кут нахилу спинки повинен регулюватися в межах $0-30^\circ$ відносно вертикального положення. Відстань від спинки до переднього краю сидіння повинна регулюватись у межах 260–400 мм.

Екран відео термінала та клавіатура мають розташовуватися на оптимальній відстані від очей користувача, але не ближче 600 мм, з урахуванням розміру алфавітно–цифрових знаків та символів.

Клавіатуру слід розміщувати на поверхні столу або на спеціальній, регульованій за висотою, робочій поверхні окремо від столу на відстані 100–300 мм від краю, ближчого до працівника. Кут нахилу клавіатури має бути в межах $5-15^\circ$.

5.3 Розрахунок освітленості робочого місця

Зробити розрахунок КПО попереднім методом для виробничої лабораторії з боковим одностороннім освітленням. Приміщення розташоване в 4 поясі світлового клімату. Довжина приміщення: 13м, глибина: 8м. Розрахункова точка А знаходиться від зовнішньої стіни на відстані: 7 м. Відстань від робочої поверхні до верха вікон: 4.4 м.

Нормоване значення коефіцієнту бокового природного освітлення для III розряду зорової роботи III світлового поясу $e_n^{III} = 1.5\%$. У зв'язку з тим, що приміщення знаходиться в IV світловому поясі, КПО буде визначено за формулою:

$$1. \quad e_n^{IV} = e_n^{III} \cdot m \cdot c$$

де $m = 0,9$ – коефіцієнт світлового клімату;

2. $c = 0,7$ – коефіцієнт сонячності клімату.

$$3. \quad e_n^{IV} = 1.5 \times 0.9 \times 0.7 = 0.945\%$$

Загальний коефіцієнт світло пропускання визначається за формулою:

$$\tau_0 = \tau_1 \tau_2 \tau_3 \tau_4 \tau_5$$

де $\tau_1 = 0,8$ – засклення подвійне;

$\tau_2 = 0,65$ – Плетення подвійне дерев'яне;

$\tau_3 = 1$ – Несущі конструкції відсутні;

$\tau_4 = 1$ – Сонце захисні конструкції відсутні

$\tau_5 = 1$ – Захисних сіток немає

$$\tau_0 = 0,8 \times 0,65 \times 1 \times 1 \times 1 = 0,52$$

Площа світлових прорізів при боковому освітленні визначається за формулою:

$$S_{\sigma} = \frac{S_n e_n K_z \eta_v K_{\sigma}}{100 r_1 \tau_0}$$

де S_n – площа підлоги приміщення,

$$S_n = 13 \times 8 = 104 \text{ м}^2;$$

$K_z = 1,2$ – коефіцієнт запасу;

$\eta_v = 13$ – світлова характеристика вікон;

$K_{\sigma} = 1$ – коефіцієнт, що враховує затінення вікон будинками, що стоять навпроти;

$r_1 = 2,4$ – коефіцієнт, що враховує підвищення КПО при боковому освітленні завдяки світлу, відбитому від поверхонь приміщення і підстильного шару, що прилягає до будинку.

$$S_{\sigma} = \frac{104 \times 1,26 \times 1,2 \times 13 \times 1}{100 \times 2,4 \times 0,52} = 16,38 \text{ м}^2$$

Загальна ширина вікон знаходиться за допомогою наступної формули:

$$b_v = \frac{S_{\sigma}}{h_v};$$

де $h_v = 4 \text{ м}$ – висота вікна;

$$b_v = \frac{8,24}{4} = 2,06 \text{ м};$$

Приймається 4 вікна, які мають ширину 1 м.

5.4 Розробка заходів щодо зменшення впливу небезпечних і шкідливих факторів

5.4.1. Заходи щодо зниження величини виникаючих зарядів статичної електрики

Для запобігання можливості виникнення небезпечних розрядів з поверхні обладнання, речовин, що перероблюються, а також з тіла людини необхідно передбачати, з урахуванням особливостей виробництва і заходи, які можуть забезпечити відведення заряду:

- 1) зниження інтенсивності генерації заряду статичної електрики;
- 2) відведення заряду шляхом заземлення обладнання та комунікацій, а також забезпечення постійного електричного контакту з заземленням тіла людини;
- 3) відведення заряду шляхом зменшення питомого об'ємного та поверхневого електричного опору;
- 4) нейтралізація заряду шляхом використання різних засобів захисту від статичної електрики по ДСТУ 12.4.124–83.

5.4.2 Заходи щодо зниження шуму на робочому місці

Заходи та засоби захисту від шуму поділяються на колективні та індивідуальні, причому останні застосовуються лише тоді, коли заходами та засобами колективного захисту не вдається знизити рівні шуму на робочих місцях до допустимих значень. Призначення засобів індивідуального захисту від шуму – перекрити найбільш чутливі канали проникнення звуку в організм – вуха. Тим самим різко послаблюються рівні звуків, що діють на барабанну перетинку, а відтак – і коливання чутливих елементів внутрішнього вуха. Такі

засоби дозволяють одночасно попередити розлад нервової системи від дії інтенсивного подразника, яким є шум.

До засобів захисту від шуму належать навушники, протишумові вкладки, шумозаглушувальні шоломи. Вибір обумовлюється видом та характеристикою шуму на робочому місці, зручністю використання засобу при виконанні даної робочої операції та конкретними кліматичними умовами.

Засоби колективного захисту від шуму подібно до віброзахисту поділяються за такими напрямками:

- 1) зменшення шуму в самому джерелі;
- 2) зменшення шуму на шляху його поширення;
- 3) організаційно–технічні заходи;
- 4) лікувально–профілактичні заходи.

Зменшення шуму в самому джерелі – найбільш радикальний засіб боротьби з шумом, що створюється устаткуванням.

Висновки за розділом 5

У даному розділі проведено аналіз потенційно – небезпечних та шкідливих факторів у робочому приміщенні. Розроблені заходи щодо усунення або зниження впливу цих факторів на робітників до нормативних вимог. Дотримання умов, що визначають оптимальну організацію робочого місця програміста, дозволить зберегти гарну працездатність протягом усього робочого дня, що у свою чергу підвищить як у кількісному, так і в якісному відношеннях продуктивність праці.

Список використаних джерел

1. Безпека життєдіяльності. Охорона праці / П.П. Кукін і ін. – М.: Вища школа, 2015. – 336 с.
2. Безпека життєдіяльності. Виробнича безпека та охорона праці. – М.: Вища школа, 2015. – 432 с.
3. Беляков, Г. І. Безпека життєдіяльності. Охорона праці / Г.І. Беляков. – М.: Юрайт, 2016. – 576 с.
4. Білярі Ю. А. Охорона праці в організації. Практичні рекомендації / Ю.А. Білярі, В.В. Ударів. – М.: Книжковий світ, 2016. – 176 с.
5. Вашко І. М. Охорона праці / І. М. Вашко. – М.: ТетраСистемс, 2016. – 208 с.
6. Графкіна М. В. Охорона праці в невиробничій сфері / М.В. Графкіна. – М.: Форум, 2016. – 320 с.
7. Грідін А. Д. Безпека і охорона праці у сфері готельного обслуговування. Практикум / А.Д. Грідін. – М.: Академія, 2015. – 937 с.
8. Девісілов В. А. Охорона праці/В.А. Девісілов. – М.: Форум, 2015.– 512 с.
9. Єфремова О. С. Охорона праці в організації в схемах і таблицях / О.С. Єфремова. – М.: Альфа–прес, 2015. – 120 с.
10. Комплект журналів по охороні праці. – М.: Альфа–прес, 2016. – 704 с.
11. Луковников А. В. Охорона праці / О.В. Луковников. – М.: Колос, 2015. – 368 с.
12. Михайлов Ю. М. Охорона праці в офісі / Ю.М. Михайлов. – М.: Альфа–прес, 2016. – 256 с.
13. Попов Ю. П. Охорона праці / Ю.П. Попов. – М.: КноРус, 2015. – 224 с.
14. Сергєєв А. Г. Менеджмент і сертифікація якості охорони праці на підприємстві / А.Г. Сергєєв, Е.А. Баландіна, В.В. Баландіна. – М.: Логос, 2015. – 216 с.

6. ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

6.1 Вступ

Закон України "Про охорону навколишнього середовища" – визначає правові, економічні, соціальні основи охорони навколишнього середовища. Завдання Закону полягає в регулюванні відносин у галузі охорони праці, використанні та відновленню природних ресурсів, забезпеченні екологічної безпеки, попередженню та ліквідації наслідків негативної дії на навколишнє середовище діяльності людини, збереження природних ресурсів, генетичного фонду нації, ландшафтів й інших природних об'єктів.

6.2 Загальні відомості про забруднення

Під забрудненням навколишнього середовища розуміють надходження в біосферу будь-яких твердих, рідких і газоподібних речовин або видів енергії (теплоти, звуку, радіоактивності і т. п.) у кількостях, що шкідливо впливають на людину, тварин і рослини як безпосередньо, так і непрямим шляхом.

Безпосередньо об'єктами забруднення (акцепторами забруднених речовин) є основні компоненти екотопу (місце існування біотичного угруповання):

- 1) атмосфера,
- 2) вода,
- 3) ґрунт.

6.3 Заходи щодо запобігання забруднення навколишнього середовища

Всі відходи здаються в господарський блок для подальшої утилізації.

Жорсткість вимог до виробництва й матеріалів, а також розробка нових виробничих й утилізаційних технологій дозволяє зменшити антропогенне навантаження на навколишнє середовище.

Заходи щодо запобігання забруднення навколишнього середовища:

- 1) ресурсозберігаючі заходи – збереження і раціональне використання земельних та водних ресурсів;
- 2) повторне використання ресурсів та ін.;
- 3) планувальні заходи – функціональне зонування, організація санітарно-захисних зон та санітарних розривів, озеленення та ін.;
- 4) відновлювальні заходи – технічна і біологічна рекультивація, нормалізація стану окремих компонентів навколишнього середовища тощо;
- 5) захисні заходи.

Висновки за розділом 6

У даному розділі проведено аналіз впливу небезпечних та шкідливих речовин, що потрапляють у середовище. Встановлені заходи щодо їх усунення та утилізації. Встановлені способи вирішення та запобігання впливу цих факторів на оточуюче середовище згідно до нормативних вимог.

Висновок

У результаті виконання дипломної роботи був розроблений веб–додаток для організації та контролю робочого процесу компанії для автоматизації роботи компаній, що відповідає вимогам поставленої задачі. У процесі роботи були розроблені такі основні розділи: дослідження використання веб–додатку для організації та контролю процесу роботи компанії, дослідження проблем та вимог щодо побудови архітектури та розробка концепції веб–додатку, розробка проектних рішень щодо веб–додатку для організації та контролю робочого процесу компанії, розрахунок і оцінка економічного ефекту від розробки веб–додатку для організації та контролю робочого процесу компанії, охорона праці, охорона навколишнього середовища.

Програмне забезпечення було розроблено за допомогою сучасних інструментів, мови програмування та фреймворку.

Програмне забезпечення «веб–додаток для організації та контролю робочого процесу компанії» було протестовано та готове до впровадження у системі Internet.

Список використаних джерел

- 1) Фокічева Л.В. Онлайн-сервіси як перспективні та альтернативні засоби організації роботи компанії [Текст] / Л.В.Фокічева, Н.В.Тендітна, О.В. Гайдаєнко //Інформаційні технології: моделі, алгоритми, системи (ITMAS – 2020): Всеукраїнська науково-практична інтернет конференція (26-28 жовтня 2020 р.) – Миколаїв: НУК ім. адмірала Макарова, 2020 – Режим доступу: <http://itconf.nuos.edu.ua/2020/proceedings/>
- 2) UML. Короткий посібник по UML / М. Фаулер.: Символ Плюс, 2011. – 192 с.
- 3) Зиков С. В. Введення в теорію програмування. Курс лекцій. Учбовий посібник / С.В. Зиков. – М.: Інтернет–університет інформаційних технологій, 2012. – 400 с.
- 4) Опалева Э. А. Мови програмування і методи трансляції / Э.А. Опалева, В.П. Самойленко. – М.: БХВ–Петербург, 2015. – 480 с.
- 5) Матеріали інформаційного сайту «Сучасний підручник JavaScript» [Електронний ресурс]. – Режим доступу: <https://learn.javascript.ru/>
- 6) Матеріали інформаційного сайту «Модель переходів станів» [Електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/>
- 7) Матеріали інформаційного сайту «Вікіпедія» [Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/Rational_rose
- 8) Матеріали інформаційного сайту «Функціональна схема» [Електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/>
- 9) Матеріали інформаційного сайту «Графічний інтерфейс користувача» [Електронний ресурс]. – Режим доступу: <https://sites.google.com/graficnij-interfejs-koristuvaca>

Інструкція користувача

Програмне забезпечення, що було розроблено в ході написання дипломної роботи, складається з екранних форм, які виконують певні функції. Головна екранна форма веб-додатку для організації та контролю робочого процесу компанії демонструє назву веб-додатку та надає стислу інформацію, що розбита по вкладках.

Для початку роботи з програмою необхідно відкрити веб-додаток, який показує головну сторінку та навігаційне меню, що представлено на (Рис. 1).

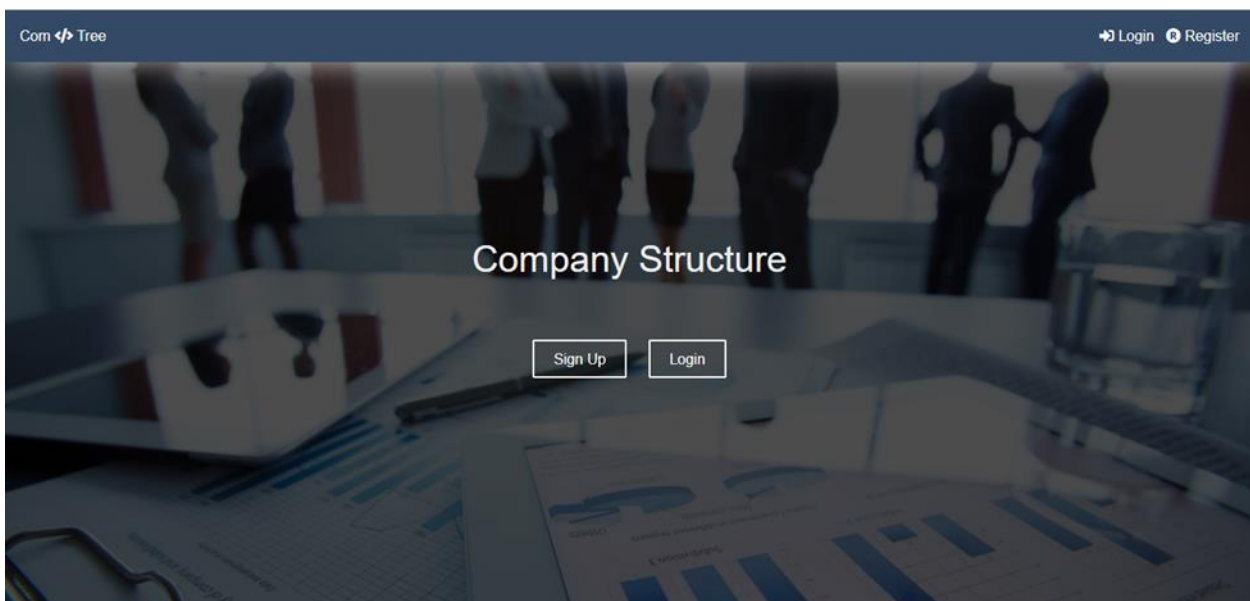
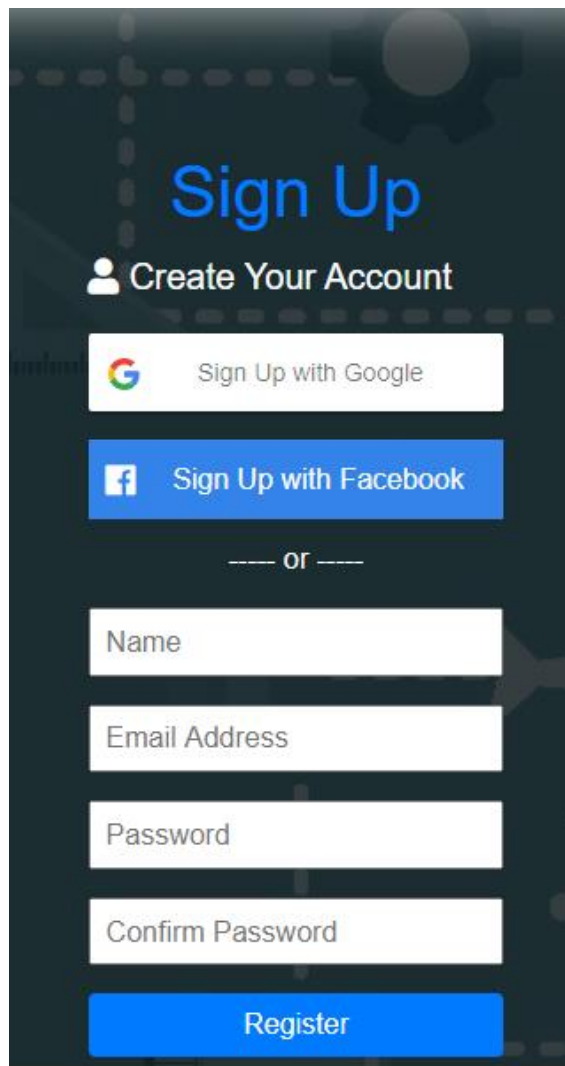


Рисунок 1 – Головна сторінка додатку

Коли користувач заходить до веб-додатку, то він має спочатку зареєструватися, тобто ввести свої дані у вікні реєстрації. Дане вікно представлено на (Рис. 2).



The image shows a 'Sign Up' form on a dark background. At the top, the text 'Sign Up' is in large blue letters, followed by 'Create Your Account' with a user icon. Below this are two social login buttons: 'Sign Up with Google' (with the Google logo) and 'Sign Up with Facebook' (with the Facebook logo). A separator 'or' is centered between these buttons. Below the separator are four white input fields stacked vertically, labeled 'Name', 'Email Address', 'Password', and 'Confirm Password'. At the bottom is a large blue button labeled 'Register'.

Рисунок 2 – Вікно реєстрації користувача

Після реєстрації та підтвердження профілю користувач повинен увійти в систему використовуючи раніше створені дані. Данні потрібно ввести у формі авторизації наведених на (Рис. 3).

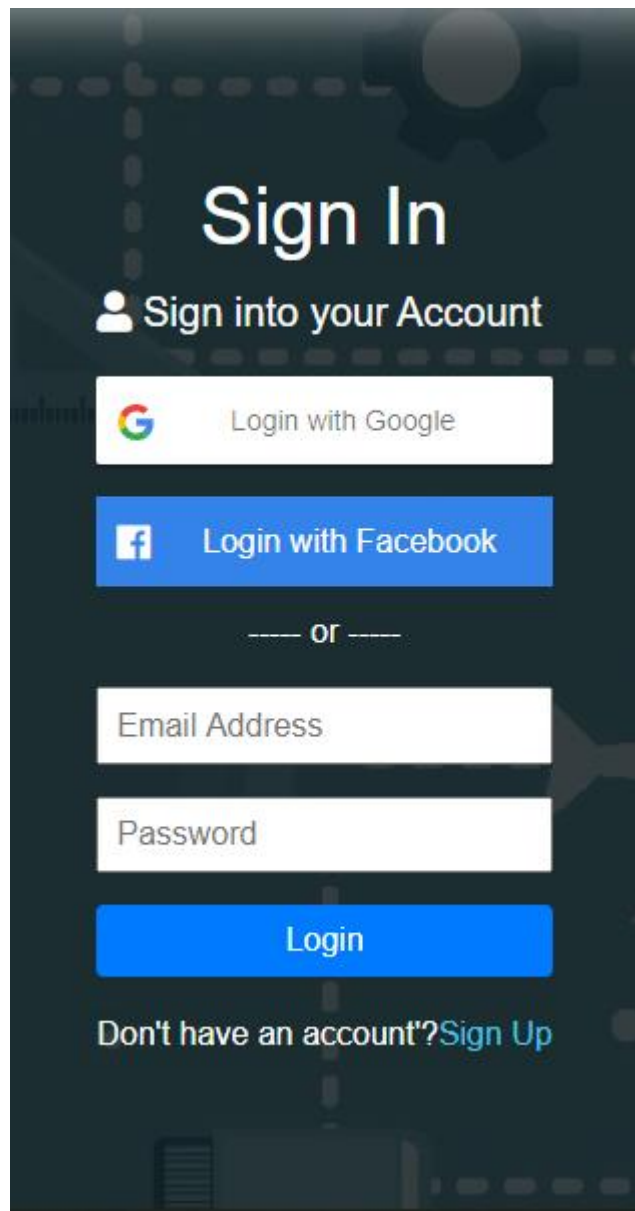


Рисунок 3 –Вікно авторизації

Після успішної авторизації користувач потрапляє на сторінку свого профілю, представлену на (Рис. 4).

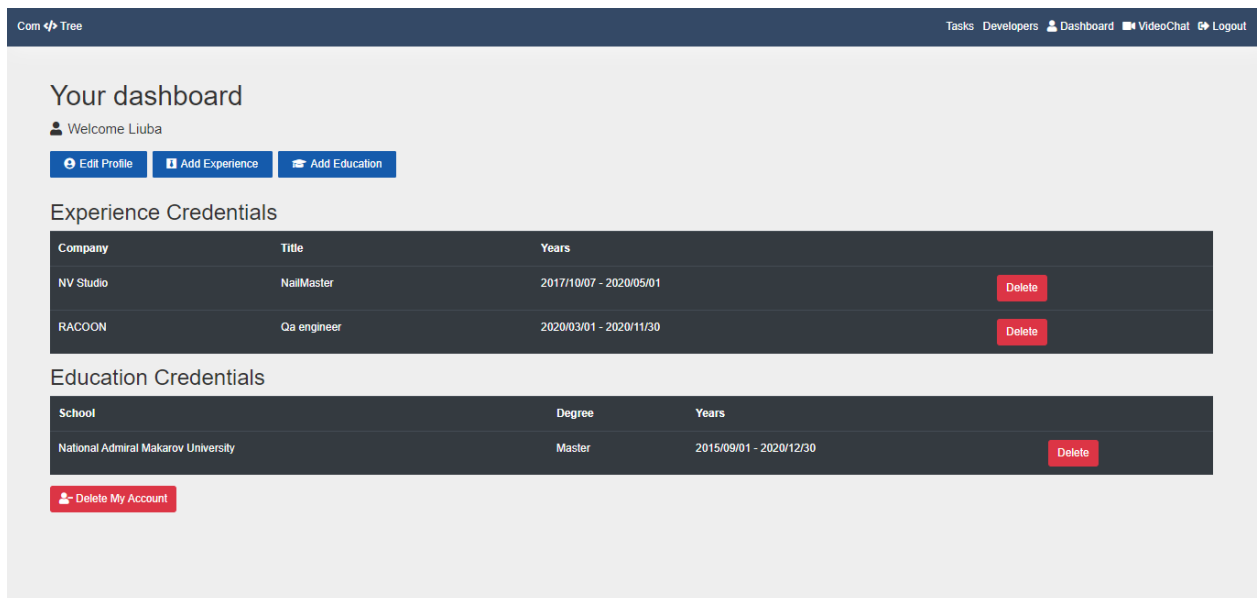


Рисунок 4 – Сторінка користувача

Користувач може додавати, редагувати та видаляти інформацію про себе у формах персональної інформації, досвіду роботи та освіти представлених на (Рис. 5–8).

Edit Your Profile

 Let's get some information to make your profile stand out

* = required field

Junior Developer 

Give us an idea of where you are at in your career

Empire

Could be your own company or one you work for

localhost:8080

Could be your own or a company website

Kherson, UA

City & state suggested (eg. Boston, MA)

Java

Please use comma separated values (eg. HTML,CSS,JavaScript,PHP)

Liubov

If you want your latest repos and a Github link, include your username

A short bio of yourself

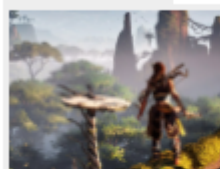
Tell us a little about yourself

Add Social Network Links Optional

Рисунок 5 – Форма персональної інформації

Select a photo, please.

 Profile photo



 preview

Выберите файл hor.jpg

Отправить

Go Back

Рисунок 6 – Форма додання фото

Add An Experience

 Add any developer/programming positions that you have had in the past

* = required field

* Job Title

* Company

Location

From date:

ДД.ММ.ГГГГ 

☐ Current Job

To Date:

ДД.ММ.ГГГГ 

Job Description

Отправить

Go Back

Рисунок 7 – Форма додання інформації про досвід роботи

Add Your Education

 Add any school or university

* = required field

* School or university

* Degree

Field of study

From Date:

дд.мм.гггг



☐ Current School

To Date:

дд.мм.гггг



Program Description

Отправить

Go Back

Рисунок 8 – форма додання інформації про освіту

Користувач може працювати зі сторінкою задач «Tasks», яка представлена на (Рис. 9,10).

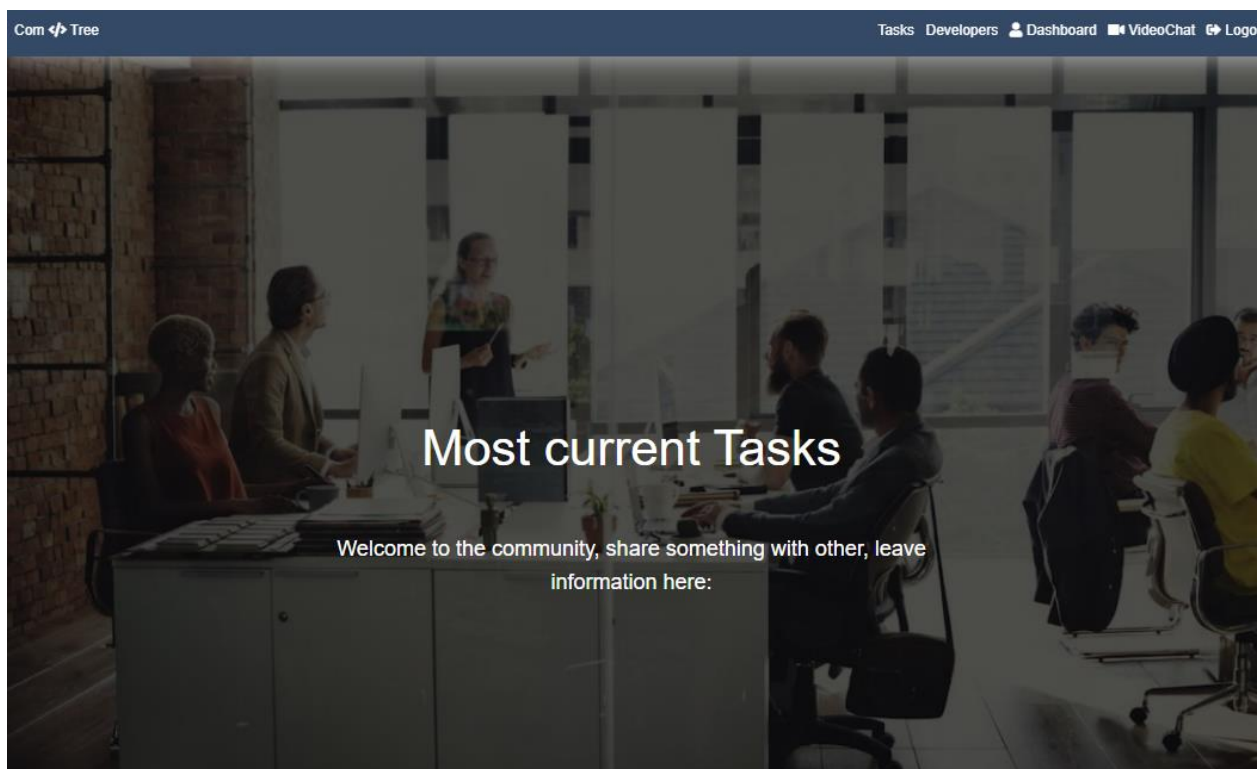


Рисунок 9 – Дошка задач

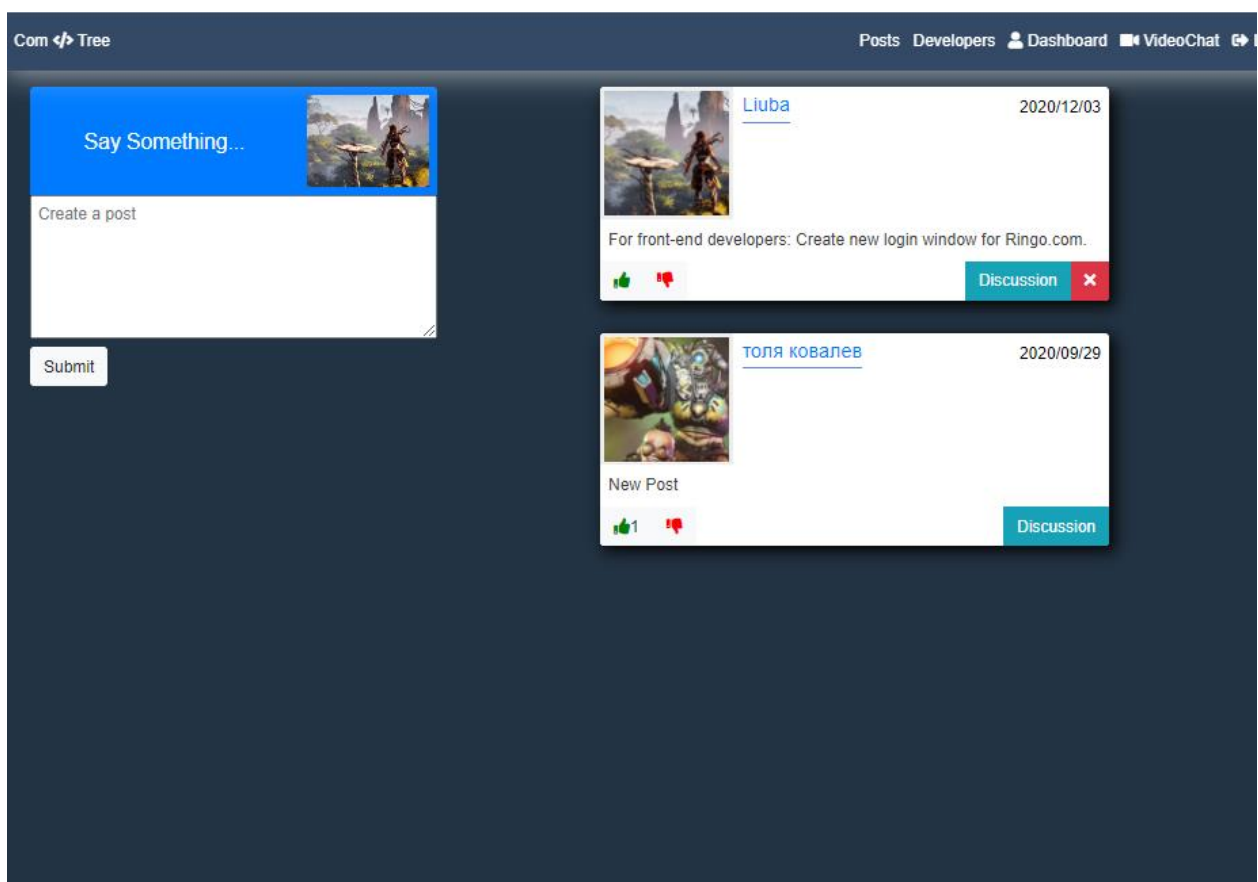


Рисунок 10 – Дошка задач

Також користувач може переглядати сторінку компанії з інформацією про

компанію та співробітників, дана сторінка представлена на (Рис. 11).

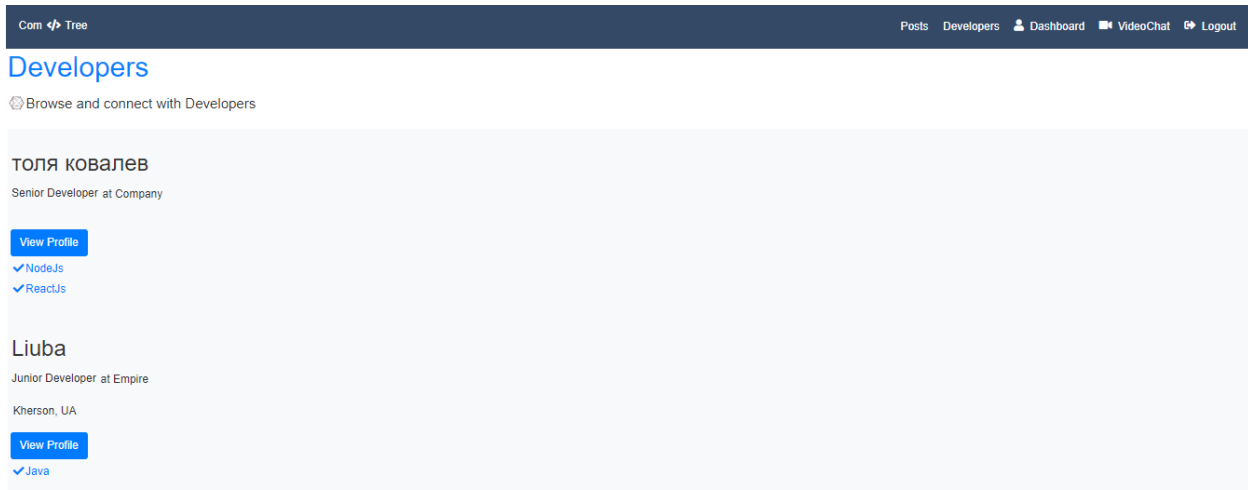


Рисунок 11 – Сторінка компанії

Користувач може заходити у зал відео–конференцій та запрошувати колег у приватну бесіду, що представлено на (Рис. 12).

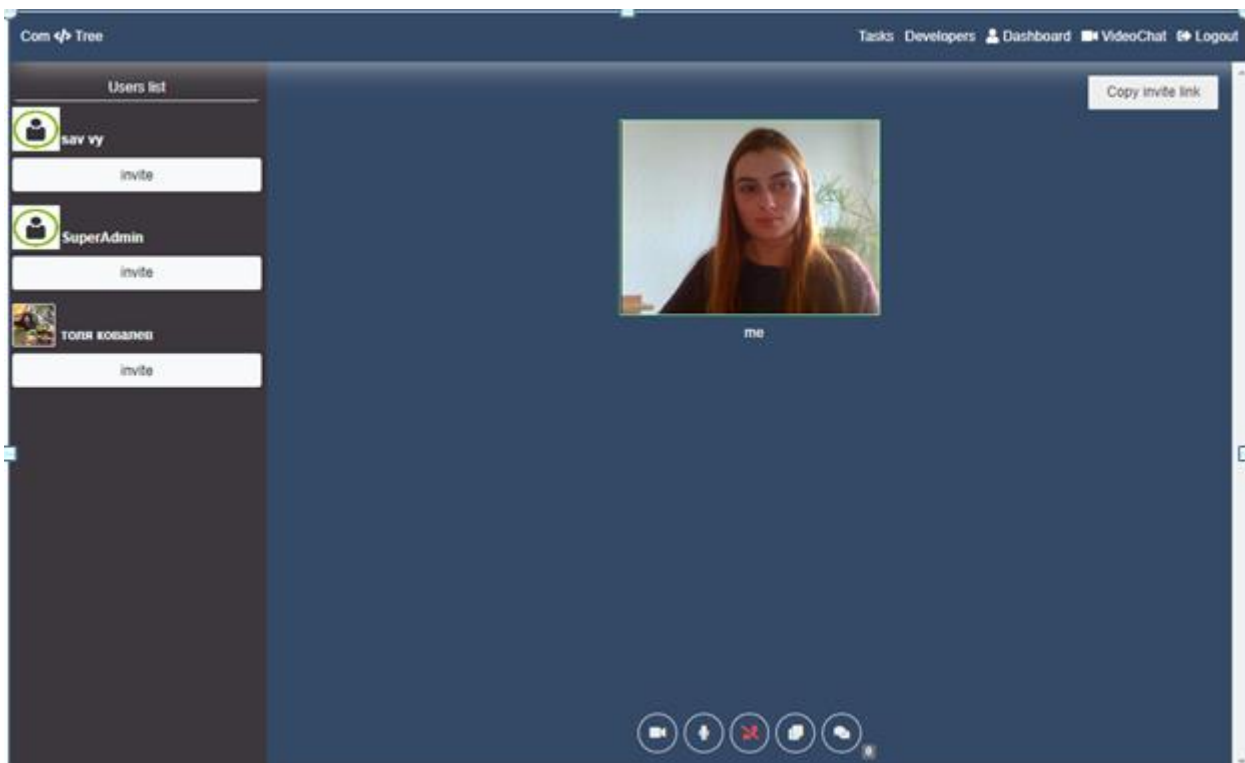


Рисунок 12 – Сторінка для відео–конференцій

Також користувачі можуть виходити зі свого профілю за допомогою кнопки «Logout» та авторизуватися за необхідності у вікні авторизації.

ДОДАТОК Б

Програма і методика випробувань веб–додатку для організації та контролю робочого процесу компанії

Б.1 Об'єкт випробування

Об'єктом випробування є веб–додаток для організації та контролю робочого процесу компанії.

Програмний продукт автоматизує процес роботи співробітників завдяки можливості легко та швидко створювати задачі доступ до яких мають всі співробітники компанії, завдяки чому скорочуються витрати часу на передачу інформації від керівників та тестувальників до дизайнерів та розробників. Автоматизується процес комунікації між співробітниками компанії завдяки розробці можливості онлайн дзвінків та переписки.

Б.2 Мета випробування

Метою випробування є перевірка програмного забезпечення на наявність помилок і відповідність реалізованих функцій функціям, зазначеним у вимогах. Якщо прийнятий рівень відповідності програмного забезпечення вимогам досягнутий, а також відсутні помилки та грубі недоліки, то програмне забезпечення приймається в експлуатацію.

Б.3 Вимоги до програми

При проведенні випробувань функціональні характеристики додатку підлягають перевірці на відповідність вимогам, викладеним у пункті «Вимоги до функціональних характеристик».

Для проведення випробування будемо розглядати всі реалізовані функції додатку:

- 1) Функції «Реєстрації та авторизації».
- 2) Функція «Додання інформації у формах персональної інформації, досвід роботи та освіта».
- 3) Функція «Створення, перегляд, видалення задач».
- 4) Функція «Створення приватного дзінка».

Б.4 Вимоги до програмної документації

Документація, що пропонується до випробування програмного забезпечення, повинна включати наступні документи:

- завдання та вимоги до розробки;
- опис програмного забезпечення;
- текст програмного забезпечення;
- програму та методику випробувань;
- інструкцію користувача.

Б.5 Засоби та порядок випробувань

Для випробувань даного додатку необхідно, щоб на комп'ютері, де буде встановлене програмне забезпечення, була встановлена операційна система Microsoft Windows 7 та вище.

Для роботи додатку необхідна наявність у користувача з'єднання з мережею інтернет, та персонального комп'ютера, що має характеристики, не нижче, ніж:

- IBM–сумісний комп'ютер з тактовою частотою процесора 1.2 МГц і вище;
- обсяг оперативної пам'яті не нижче 512 МБ;
- не менше 200 МБ вільного дискового простору жорсткого диска;
- монітор;

- клавiатура;
- миша.

Випробування додатку будуть проводитися протягом двох тижнів. При наявності відповідної можливості у процесі проведення випробувань можуть додатково брати участь програмісти. Попередньо до початку випробувань визначається кінцевий склад групи, що буде проводити випробування. Кожен член групи одержує необхідну документацію до програми. При випробуванні проводиться:

- 1) контроль складу виконуємих функцій;
- 2) контроль коректності виконання окремих функцій;
- 3) випробування на правильність роботи окремих модулів;
- 4) комплексне випробування.

При виявленні помилок у роботі програмного забезпечення складається перелік помилок і виправляється розробником.

Після цього проводиться повторне випробування у повному обсязі.

Б.6 Методи випробування

Випробування будемо проводити за стратегією «чорного ящика».

Складемо програму випробування:

Випробування функції «Реєстрації та авторизації»

У меню обрали вкладку «Sing Up», після чого потрапили на форму реєстрації, таким чином ми маємо можливість створити нового користувача. Для успішної роботи функції потрібно заповнити всі поля, погодитися з політикою конфіденційності та натиснути кнопку підтвердження. Коли профіль користувача підтверджено потрібно авторизуватися у вікні авторизації, після успішної авторизації потрапляємо на персональну сторінку. При виконанні всіх перерахованих дій та потраплянні на персональну сторінку, дану функцію можна вважати реалізованою.

Випробування функції «Додання інформації у формах персональної інформації, досвід роботи та освіта»

На сторінці персональної інформації обрали відповідну форму, після чого потрапили на форму, таким чином ми мали можливість додати інформацію про користувача, досвід та освіту. При заповненні та підтвердженні дії, знову потрапили на сторінку персональної інформації, на якій при успішній роботі функцій повинна відображатися введена інформація.

Випробування функції «Створення, перегляд, видалення задач»

У додатку натиснули кнопку «Tasks», таким чином ми перейшли до сторінки задач з усіма задачами. Заповнюючи форму створюємо нову задачу. При успішній роботі функції нова задача повинна відображатися на сторінці.

Випробування функція «Створення приватного дзінка»

У додатку натиснули кнопку «VideoChat», таким чином ми перейшли до сторінки відео–конференцій. Натиснувши кнопку «Copy invite link» та відправивши її користувачеві за допомогою кнопки «Invite» потрапляємо у вікно бесіди з запрошеним користувачем. При успішній роботі функції запрошений користувач повинен отримувати запрошення та при натисканні на нього переходить у дзвінок з організатором.

Опис веб–додатку для організації та контролю робочого процесу компанії

В.1 Загальні відомості

Найменування програмного забезпечення: веб–додаток для організації та контролю робочого процесу компанії.

Для роботи з програмним забезпеченням необхідно:

- Мати доступ до мережі інтернет;
- Мати встановлену ОС Windows.

В.2 Функціональне призначення

Функціональне призначення розробки: веб–додаток для організації та контролю робочого процесу компанії повинен зекономити час співробітників компанії у процесі роботи та спілкування. Завдяки автоматизованим функціям додаток допомагає легко та швидко організувати роботу команди та процес комунікації.

В.3 Опис логічної структури

При запуску з'являється головна сторінка додатку, яка містить кнопки для переходу на форми реєстрації та авторизації відповідно, що представлено на (Рис. 1):

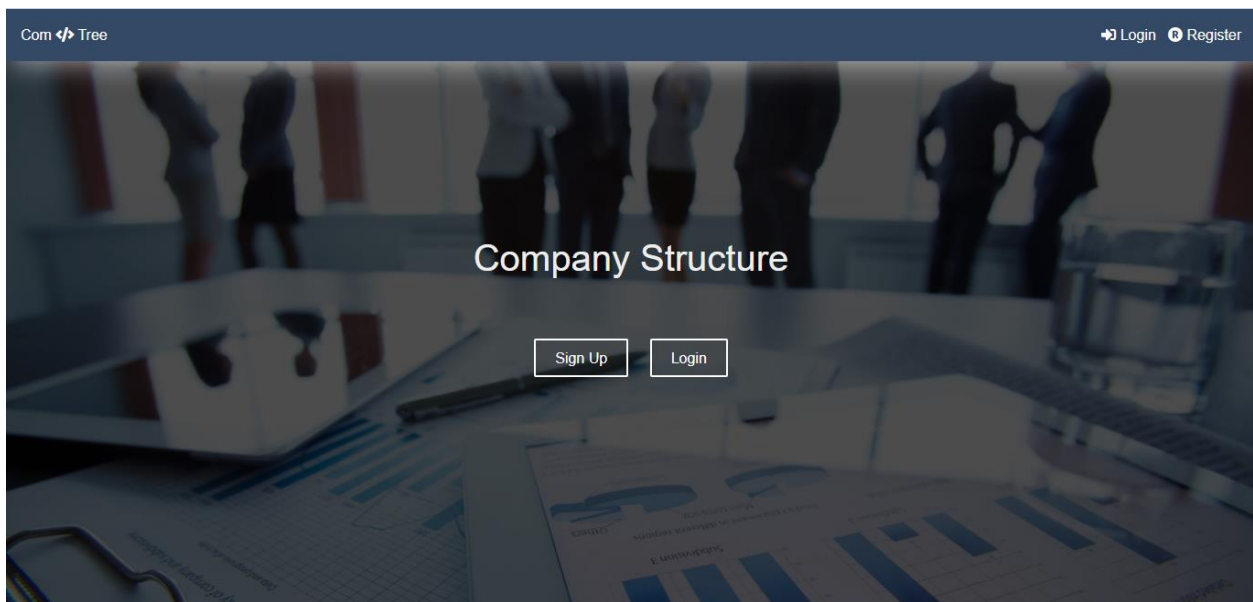


Рисунок 1 – Головна сторінка додатку

В.4 Використані технічні засоби

Для реалізації програмного забезпечення використовувалась мова JavaScript та фреймворк React.

В.5 Виклик та завантаження

Для функціонування програмного забезпечення у користувача повинен бути доступ до мережі інтернет. Для виклику даного програмного забезпечення здійснюється перехід за посиланням з назвою «<https://company-structure.com.ua>».

В.6 Вхідні дані

Вхідною інформацією для даної системи є :

- інформація про користувача, яка заповнюється для подальшого публікування.
- Інформація про завдання, яка заповнюється у формі на сторінці «Tasks» для подальшого виведення всім користувачам.

В.7 Вихідні дані

Вихідною інформацією є перегляд користувачем персональної інформації, також отримання інформації про інших користувачів, інформації про задачі.

Г.1 Програмний код компоненту «профіль користувача» додатку:

```

import React, { useEffect, Fragment } from "react";
import PropTypes from "prop-types";
import { connect } from "react-redux";
import { Link } from "react-router-dom";
import Spinner from "../common/spinner/Spinner";
import { getProfileById } from "../redux/actions/profileActions";
import ProfileTop from "../ProfileTop";
import ProfileAbout from "../ProfileAbout";
import ProfileExperience from "../ProfileExperience";
import ProfileEducation from "../ProfileEducation";
import ProfileGithub from "../ProfileGithub";

const Profile = ({
  auth,
  getProfileById,
  profile: { loading, profile },
  match,
}) => {
  useEffect(() => {
    getProfileById(match.params.id);
  }, [getProfileById]);

  return (
    <Fragment>
      {profile === null || loading ? (
        <Spinner />
      ) : (
        <Fragment>
          <Link to="/profiles" className="btn btn-light">
            Back to Profiles
          </Link>
          {auth.isAuthenticated &&
            auth.loading === false &&
            auth.user._id === profile.user._id && (

```

```

    <Link to="/edit-profile" className="btn btn-dark">
      Edit Profile
    </Link>
  )}
<div className="profile-grid">

```

```

    <ProfileTop profile={profile} />
    <ProfileAbout profile={profile} />
    <div className="profile-exp bg-white p-2">
      <h2 className="text-primary">Experience</h2>
      {profile.experience.length > 0 ? (
        <Fragment>
          {profile.experience.map((experience) => (
            <ProfileExperience
              key={experience._id}
              experience={experience}
            />
          ))}
        </Fragment>
      ) : (
        <h4>No experience</h4>
      )}
    </div>
    <div className="profile-edu bg-white p-2">
      <h2 className="text-primary">Education</h2>
      {profile.education.length > 0 ? (
        <Fragment>
          {profile.education.map((education) => (
            <ProfileEducation
              key={education._id}
              education={education}
            />
          ))}
        </Fragment>
      ) : (
        <h4>No Education</h4>
      )}
    </div>

```

```

    })
  </div>
  {profile.githubusername && (
    <ProfileGithub username={profile.githubusername} />
  )}
</div>
</Fragment>
)}
</Fragment>
);
};

```

```

Profile.propTypes = {
  getProfileById: PropTypes.func.isRequired,
  profile: PropTypes.object.isRequired,

```

```

  auth: PropTypes.object.isRequired,
};

```

```

const mapStateToProps = (state) => ({
  profile: state.profile,
  auth: state.auth,
});
export default connect(mapStateToProps, { getProfileById })(Profile);

```

Г.2 Программний код компоненту «відео-чат» додатку:

```

import React from "react";
import { connect } from "react-redux";
import { isEmpty } from "lodash";
import { Modal, ModalHeader, ModalBody, ModalFooter } from "reactstrap";
const ChatModal = ({
  showModal,

```

```

messages,
handleMessage,
closeChat,
sendMessage,
message,
}) => (
  <Modal isOpen={showModal} className="video-chat-modal">
    <ModalHeader>Chat Room</ModalHeader>
    <ModalBody>
      {isEmpty(messages) ? (
        <p>No messages yet</p>
      ) : (
        messages.map((item, index) => (
          <div key={index} className="video-chat-modal__message">
            <b>{item.sender}</b>: <span>{item.data}</span>
          </div>

          ))
        )}
    </ModalBody>
    <ModalFooter className="div-send-msg">
      <input
        placeholder="Message..."
        value={message}
        onChange={(e) => handleMessage(e)}
      />
      <button onClick={() => sendMessage()}>Send</button>
      <button onClick={() => closeChat()}>Close</button>
    </ModalFooter>

```

```

    </Modal>
  );
  export default ChatModal;

import React, { useState, useEffect, lazy } from "react";
import { isChrome } from "../../utils/videoChatConst";
import { connect } from "react-redux";
import "./VideoChat.scss";
const UsersList = lazy(() => import("./UsersList"));
const VideoChatContent = lazy(() => import("./VideoChatContent"));

const VideoChat = ({ user, profile }) => {
  if (!isChrome())
    <div className="video-not-support">
      <h1>Sorry, your browser doesnt support video chat</h1>
    </div>;
  return (

    <div className="video-chat-wrapper">
      <UsersList />
      <VideoChatContent user={user} profile={profile} />
    </div>

  );
};

const mapStateToProps = (state) => ({
  user: state.auth.user,
  profile: state.profile,
});

```

```
export default connect(mapStateToProps, {})(VideoChat);
```

```
import React, { useEffect, useState } from "react";
import { connect } from "react-redux";
import { getAllUsersList } from "../../redux/actions/profileActions";
import { profilePhotoUrl } from "../../utils/urls";
import "../../UsersList.scss";

const UsersList = ({ auth: { user }, getAllUsersList, profile: { users } }) => {
  useEffect(() => {
    getAllUsersList();
  }, []);
  return (
    <div className="users-list-wrapper">
      <h3>Users list</h3>
      <ul>
        {users.map(
          (userItem, index) =>
            user._id !== userItem._id && (
              <li key={index} className="user-list__item">

                <img
                  className="user-list__photo"
                  alt="photo"
                  src={profilePhotoUrl(userItem.email)}
                />

                <p>{userItem.name}</p>
                <button className="btn btn-light">invite</button>

              </li>
            )
          )
        }
      </ul>
    </div>
  );
}
```

```
        )
      })
    </ul>
  </div>
);
};
const mapStateToProps = (state) => ({
  profile: state.profile,
  auth: state.auth,
});
export default connect(mapStateToProps, { getAllUsersList })(UsersList);
```

```
@import "../variables";
.users-list-wrapper {
  display: flex;
  background: $darkBlueV1;
  color: #fff;
  flex-direction: column;
  width: fit-content;
  min-width: 300px;
  padding: 0.4rem;

  h3 {
    text-align: center;
    font-size: 1rem;
    border-bottom: 2px solid #eee;
    border-radius: 4px;
    margin: 0.5rem 0.2rem;
```



```
white-space: nowrap;
padding: 5px 0.8rem;
}
.user-list__item {
display: flex;
flex-wrap: wrap;
flex-direction: row;
align-items: flex-end;
margin-bottom: 1rem;
font-weight: bold;
&:nth-last-child(1) {
margin-bottom: 0;
}
.btn-light {
margin-top: .5rem ;
width: 100%;
}
.user-list__photo {
overflow: hidden;
border-radius: 5%;
object-fit: cover;
width: 50px;
height: 50px;
border: 2px solid #eee;

}

p {
margin-left: 0.4rem;
```

```
margin-bottom: 0;
height: min-content;
}
}
}
```