

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
Факультет електроніки та інформаційних технологій  
Кафедра комп'ютерних наук

Кваліфікаційна робота магістра  
**ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ПРОЕКТУВАННЯ СИСТЕМИ  
ЕЛЕКТРОННОГО ДОКУМЕНТООБИГУ У ПІДПРИЄМСТВІ**

Здобувач освіти гр. ІН.мдн-11С

Альона ДРОБОНОГ

Науковий керівник,  
кандидат ф.-м. наук

Галина ОЛЕКСІЄНКО

В.о. завідувач кафедри,  
кандидат технічних наук,  
доцент

Ігор ШЕЛІХОВ

СУМИ 2022

Сумський державний університет

(назва вузу)

Факультет \_\_\_\_\_ ЦЗДФН Кафедра \_\_\_\_\_ Комп'ютерних наук

Спеціальність 122 «Комп'ютерні науки»

Затверджую:

зав. кафедри \_\_\_\_\_

“ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ**

**НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТОВІ**

Дробоног Альона Михайлівна

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Інформаційна технологія проектування системи електронного документообігу у підприємстві

затверджую наказом по інституту від “ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_ р. № \_\_\_\_\_

2. Термін здачі студентом закінченого проекту (роботи) \_\_\_\_\_

3. Вхідні данні до проекту (роботи) \_\_\_\_\_

\_\_\_\_\_

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити)

1) Інформаційний огляд. 2) Вибір програмних засобів. 3) Практична реалізація

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) \_\_\_\_\_

\_\_\_\_\_

6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання

\_\_\_\_\_

Керівник

\_\_\_\_\_

(підпис)

Завдання прийняв до виконання

\_\_\_\_\_

(підпис)

### КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів дипломного проекту (роботи)	Термін виконання проекту (роботи)	Примітка
1	<i>Інформаційний огляд</i>		
2	<i>Вибір програмних засобів</i>		
3	<i>Практична реалізація</i>		
4	<i>Оформлення кваліфікаційної магістерської роботи</i>		

Студент – дипломник \_\_\_\_\_

(підпис)

Керівник проекту \_\_\_\_\_

(підпис)

## РЕФЕРАТ

**Записка:** 56 ст., 14 рис., 18 джерела, 1 додатки

**Мета роботи** — створення веб додатку для спрощення і аргументації прийняття рішень на користь переходу на електронний документообіг.

**Об'єкт дослідження** — аналіз доцільного використання електронного документообігу в малих підприємствах.

**Предмет дослідження** — швидкість обробки даних згідно з усередненим алгоритмом прийняття рішення.

**Методи дослідження** — аналіз даних за допомогою комп'ютерного програмування.

**Результати** — був створений веб додаток, який дозволяє змоделювати процес документообігу за допомогою мови програмування JavaScript.

ЕЛЕКТРОННИЙ ДОКУМЕНТООБІГ, ЕЛЕКТРОННИЙ ПІДПИС, ВЕБ-ОРІЄНТОВАНА СИСТЕМА, JAVASCRIPT, HTML, CSS

## ЗМІСТ

ЗМІСТ .....	4
ВСТУП .....	5
1. ЛІТЕРАТУРНИЙ ОГЛЯД .....	6
1.1 Електронний документообіг .....	6
1.2 Електронний документообіг в Україні .....	12
1.3 Постановка задачі .....	15
2. ВИБІР ПРОГРАМНИХ ЗАСОБІВ .....	17
2.1 HTML для створення графічного інтерфейсу .....	17
2.2 CSS мова стилів HTML .....	18
2.3 Мова програмування JavaScript .....	19
2.4 Аналіз та створення алгоритма роботи веб-додатку .....	20
3. ПРАКТИЧНА РЕАЛІЗАЦІЯ .....	22
3.1 Створення веб додатку з використанням мови програмування JavaScript. ....	22
3.2 Тестування веб додатку .....	24
ВИСНОВКИ .....	29
СПИСОК ЛІТЕРАТУРИ .....	30
ДОДАТОК .....	32

## ВСТУП

В кожному із підприємств настає момент коли потрібно вдосконалювати та розвивати бізнес, щоб бути конкурентоспроможним. Тому підприємці шукають рішення для збільшення швидкості обробки документів в компанії, що в свою чергу збільшить прибутки. Одним з рішенням по розвитку компанії є електронний документообіг. Його впровадження надає суттєву перевагу звичайним паперовим документообігом. Він пришвидшує обробку, коригування, доповнення, зберігання документів. Але його впровадження є затратним та складним. Тому було прийнято рішення про створення інтерфейсу, за допомогою якого можливо змодельовати скільки часу буде потребувати прийняття документа в компанії за допомогою електронних ресурсів.

Порівняно з паперовим документообігом електронний документообіг дає можливість планувати корпоративну діяльність, облік та контроль на всіх етапах управління, а також новий рівень складності підходу, системного аналізу та прогнозування. І це підвищує стійкість фірми у частих кризових ситуаціях і знижує ризик у невизначених ситуаціях.

Використання електронного документообігу в українських компаніях є актуальним завданням. Адже важливість підкреслено на законодавчому рівні з 2003 року Законом «Про електронні документи та електронний документообіг». Держава привчала до електронного документообігу не тільки підприємства і установи, а й приватний бізнес. Проаналізувавши ринок, було виявлено, що у малих підприємствах в середині компанії не використовують електронний документообіг, тому що його впровадження, навчання співробітників є затратним не тільки в час, а й фінансовому. Електронна документація не тільки економить час в створенні нових документів.[1]

Цей веб додаток дозволяє порівняти швидкість обробки даних при застосуванні електронного обігу порівняно з традиційним документообігом.

# 1. ЛІТЕРАТУРНИЙ ОГЛЯД

## 1.1 Електронний документообіг

Електронний документообіг – це обмін документами в електронному вигляді (на відміну від паперового) захищеним каналом зв'язку. Такі документи не потрібно роздруковувати, оскільки вони мають юридичну силу та прирівняні до оригіналу - відповідно до Закону України "Про електронні документи та звернення електронних документів".[1]

Електронний підпис (ЕП) — це еквівалент звичного підпису від руки, який підтверджує авторство електронного документа та має повну юридичну силу.

Перевівши свій бізнес на електронний документообіг, ви зможете замінити основні фінансові документи, такі як контракти, рахунки-фактури та звіти про виконані роботи.

Електронний документообіг відноситься до процесу управління документами в організації. Цей процес включає отримання файлів, індексування та зберігання, зберігання інформації, доступ до файлів і розповсюдження.

Із зростаючою дематеріалізацією документів і демократизацією інструментів ІТ-комунікації компанії, спільноти та представники вільних професій виробляють і керують все більшою кількістю електронних документів.[2]

Навіщо компаніям потрібен електронний документообіг

Ось п'ять основних причин:

- Зменште кількість паперових документів в офісі.
- Зниження фінансових та трудових витрат на зберігання документів.

- Скоротить витрати на друк документів, канцелярські товари та доставку.
- Скоротити трудовитрати за рахунок виключення ручних операцій та автоматизації процесів.
- Позбавтеся необхідності втрачати або видавати нові документи.

Як виглядає електронний документообіг під час обміну документами між компаніями

1. Співробітник однієї з компаній готує документ, наприклад заяву або офіційний лист.
2. Потім підпишіть його своїм електронним підписом та надішліть до пункту призначення через систему ЕДО.
3. Співробітник приймаючої компанії читає документ, а також підписує його електронним підписом.
4. Незабаром після цього перша організація отримає повідомлення, після чого можна буде відвантажити товар, перевести платіж або виконати будь-яку іншу дію, зазначену у документі.[3]

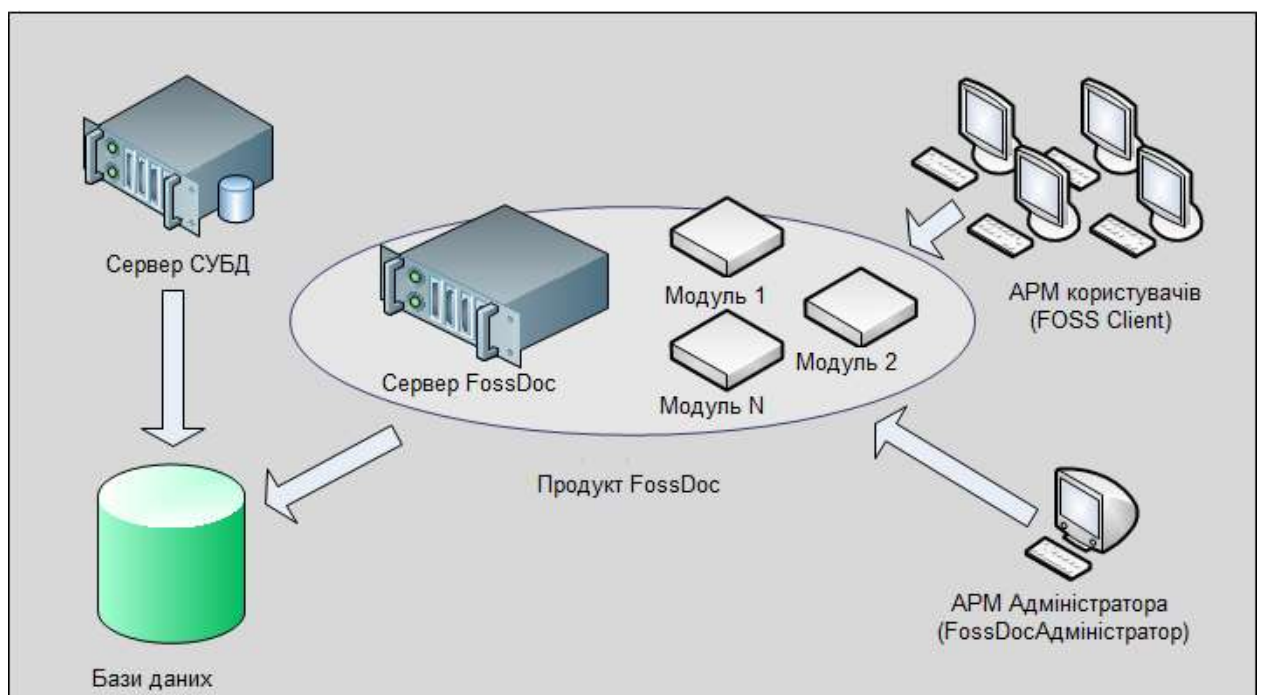


Рис.1.1.1 - Передавання даних між компаніями та держустановами



Можна виділити два основних види ЕДО: внутрішній (у межах однієї організації) та зовнішній (див. рис.1.1.1) (передавання даних між компаніями та держустановами). Існує детальніша класифікація за типами документів. Ось вона:

- кадровий ЕДО;
- управлінський;
- складський;
- бухгалтерський;
- архівний тощо.

Багато великих послуг підтримують роботу з усіма видами документів. Але найчастіше принципи електронного документообігу використовуються створення великих фінансових документів і рахунків-фактур.[4]

Електронна пошта може вважатися найпростішим прикладом ЕДО. Він також пропонує високошвидкісну передачу даних лише у тому випадку, якщо обидві сторони згодні ділитися даними. Однак поштова пошта є небезпечним каналом для надсилання важливих документів, і деякі державні відомства просто відмовляються приймати документи, надіслані звичайною електронною поштою.[3, 5]

Щоб уникнути неприємностей та безпечно обмінюватися важливими документами, слід працювати через систему оператора. Операторами ЕДО є організації, які мають необхідні програмні комплекси для обміну інформацією, відкритою та конфіденційною. В основному це послуги із захищеними каналами зв'язку з використанням спеціального шифрування.

При виборі системи ЕДО враховуйте таке:

- Швидкість технічної підтримки клієнтів.
- Будьте готові допомогти відповідати іншій людині.
- Додаткові можливості для інтеграції.
- Наявність консультантів для допомоги клієнтам у реалізації ЕДО

Окрім обміну документами, оператор пропонує клієнту повний сервіс.

- Використовуйте для обміну лише високозахищені канали для конфіденційності.
- Дотримуйтесь правил та вимог законодавства.
- Перевірити автентифікацію електронних підписів.



Рис.1.1.2 - Переваги ЕДО

### Переваги ЕДО

Впровадження електронного документообігу має багато переваг, але ось основні з них(див. рис. 1.1.2) :

- Зберігати гроші. Заощаджуйте десятки тисяч гривень на друку паперових документів, конвертів та доставці.
- Економити час. Обмінюйтеся електронними документами так само швидко, як і онлайн. Не потрібно витратити час на пошук, розповсюдження та обробку документів.
- Доступ 24/7. Підписуйте та надсилайте документи будь-коли з будь-якого пристрою, який має доступ до електронних підписів.

- Гарантії доставки та конфіденційності. Листи зберігаються в захищеному особистому кабінеті, тому їх не можна втратити або потрапити в руки злочинців.
- Зручність зберігання. Не потрібно будувати приміщення для зберігання документів.

ЕДО працює завдяки системі електронного документообігу. Вони мають свої типи та функції. Ми пояснимо, що таке, і розглянемо нові системи електронного документообігу.

Система електронного документообігу - це програмне забезпечення для управління електронними документами на всіх етапах їх життєвого циклу, включаючи створення, редагування та зберігання.

Сучасні системи підтримують маршрутизацію документів і, звичайно, базові функції, такі як пошук і сортування. Документи та бізнес-процеси тісно пов'язані між собою, тому сучасні системи автоматизації повинні включати інструменти обробки як процесів, так і інформації. електронний архів.



Рис.1.1.3 - Способи захисту електронних документів

Традиційно можна виділити кілька видів систем електронного документообігу:

- **Workflow.** У перекладі з англійської це «робочий процес», що означає систему, що координує виконання завдань у рамках бізнес-процесів підприємства. Більшість бізнес-процесів супроводжуються такими документами, як заяви працівників на відпустку, пояснювальні відомості, внутрішні накази, запровадження нових нормативних актів.

Завдяки внутрішній системі електронного документообігу ви зможете позбутися паперової тяганини та тривалих погоджень документів. Система вчить правильно писати внутрішні документи, а терміни виконання завдань відстежуються за допомогою нагадувань.

- **ЕСМ-система (Enterprise Content Management).** Документообіг є гібридом системи документообігу та електронного архіву. Він не лише дбає про маршрутизацію процесів, які раніше виконувались з використанням паперових документів, а й зберігає важливі файли та організує віддалений обмін знаннями для співробітників. Він також може містити CRM-систему. Історія ваших взаємодій з клієнтом зберігається і ви можете нею керувати.

- Це окремий випадок системи управління документообігом — етап переходу від паперового до електронного документообігу. Паперовий архів - це купа папок і ящиків, заповнених документами, в яких потрібно вручну розмістити необхідні документи. Самі паперові документи зберігаються в електронних архівах, але в цифровому вигляді. Таким чином, ви можете сортувати їх за різними ознаками та здійснювати зручний пошук за даними самих документів, а не лише за їх символами. Такий архів розв'язує проблему фізичного доступу до документів для великих організацій із філіями в різних містах.

Основними можливостями системи електронного документообігу є:

- Створення, реєстрація та зберігання електронних документів, створення карток документів.

- Переміщення документів за заданими маршрутами, можливість їх відстеження.
- Зберігайте історію переглядів документів.
- Створення та налаштування сповіщень про хід виконання документа та його статус.
- Ведення журналів, керування термінами та створення звітів.
- Можливості аналітики.
- Імпорт та експорт документів, включаючи сканування та друк паперових документів.
- Забезпечення інформаційної безпеки та конфіденційності, особливо контролю доступу та розмежування, використання цифрових підписів.[1, 6]

## **1.2 Електронний документообіг в Україні**

Питання електронного документообігу та його впровадження в організаціях і установах України є загальнодержавним. З 2003 року законодавство «Про електронні документи та електронний документообіг» на законодавчому рівні підкреслило важливість електронного документообігу в Україні.[1, 7]

З того часу було запроваджено багато інших правових принципів. Зокрема, Закон України «Про електронні трастові послуги» встановлює юридичну силу електронного документообігу, а Постанова Кабінету Міністрів № 55 від 17 січня 2018 року «Деякі питання документообігу» встановлює адміністративні повноваження в електронній формі, рекомендуємо перейти на документообіг. Формат електронних файлів державні органи.

Водночас держава також запровадила електронний документообіг на приватних підприємствах. Зокрема, потрібна електронна звітність до податкових органів. Здебільшого бізнес обмежувався лише цим до початку

пандемії коронавірусу. Карантин значно прискорив розвиток електронного документообігу, і все більше компаній обмінюються один з одним оригіналами документів в електронному вигляді. [7]

З правової основи поширення електронних документів, можна виділити правила. Ці правила зведено до наступної таблиці (таблиця 1.2). Так, загалом немає потреби роздруковувати електронні документи, але є винятки для зберігання (див. таблицю 1.2, пункти 5 та 6) та податкових перевірок (див. таблицю 1.2, пункт 7).

ПРАВИЛА ЕЛЕКТРОННОГО ДОКУМЕНТООБОРОТУ 2022			
№	Правило	Пояснення	Норма законодавства
<b>Реквізити електронних документів</b>			
1	Електронні документи повинні мати всі обов'язкові реквізити, які є в їхніх паперових аналогів	<p>Обов'язкові реквізити електронних документів такі ж, як і у паперових, а саме:</p> <ul style="list-style-type: none"> <li>• назва документа (форми);</li> <li>• дата складання;</li> <li>• назва підприємства, від імені якого складений документ;</li> <li>• зміст, обсяг та одиниця виміру господарської операції;</li> <li>• посади осіб, які відповідальні за впровадження господарської операції та правильність її оформлення;</li> <li>• особистий підпис або інші дані, які дозволяють ідентифікувати особу, яка брала участь у господарській операції</li> </ul> <p>Крім того, саме електронні документи повинні мати дані, які дозволяють встановити:</p> <ul style="list-style-type: none"> <li>• походження;</li> <li>• призначення документа;</li> <li>• дату і час його отримання.</li> </ul>	ч. 2 ст. 9 Закону про бухоблік, ст. 13 Закону № 851, п. 2 розд. II Порядку № 188/6

<b>Електронний документ проти паперового документу</b>			
2	Оригінал електронного документа – це електронний примірник документа, а не його паперова форма. Роздрукований електронний документ вважається копією	Для того, щоб мати статус документа до нього має бути застосований електронний підпис або кваліфікований електронний підпис (КЕП). Останній прирівнюється до власноручного.  При цьому електронні документи можуть зберігатися на декількох носіях і бути розіслані декільком суб'єктам. Усі вони вважаються оригіналом	ч. 4 ст. 18 Закону № 2155, ст. 7 Закону № 851
3	Ідентичні за змістом електронний і паперовий документ мають однакову юридичну силу	Якщо електронний документ роздрукувати та підписати паперовий варіант, то вони обидва вважаються оригіналами, якщо не підписати – паперовий вважається копією електронного (див. правило 2)	ст. 7 Закону № 851
4	До електронних і паперових документів застосовується однаковий підхід щодо діловодства та їхнього зберігання	Це значить, що електронні документи слід зберігати не менше того ж часу, який визначений і для їхніх паперових аналогів	ст. 13 Закону № 851, Порядок № 188/6
<b>Зберігання електронних документів</b>			
5	Електронні документи тривалого зберігання (понад 10 років) повинні мати свою паперову копію	Це значить, що такі документи мають зберігатися у двох формах – електронній та паперовій	п. 4 розд. I Порядку № 188/6
6	Електронні документи щодо яких є ймовірність незбереження впродовж визначеного для них мінімального часу треба: <ul style="list-style-type: none"> <li>• продублювати на декількох електронних носіях;</li> <li>• або мати їхні паперові копії</li> </ul>	Варіантами зберігання електронних документів можуть бути зовнішні накопичувачі (флешки, диски тощо), внутрішні накопичувачі на комп'ютерах підприємства або ж хмарні сервіси. Електронний документ як файл можна продублювати одразу на кількох носіях, які можна захистити паролем	ст. 13 Закону № 851
<b>Друк електронних документів на вимогу</b>			
7	Підприємство зобов'язане роздрукувати електронний документ на вимогу: <ul style="list-style-type: none"> <li>• учасника господарської операції;</li> <li>• правоохоронних органів;</li> <li>• податківців;</li> <li>• судових органів;</li> <li>• інших органів у межах їхніх повноважень</li> </ul>	У разі зберігання первинних документів з використанням електронних засобів підприємство зобов'язане за свій рахунок виготовити їхні копії на паперових носіях на вимогу таких сторін.  Щодо податківців, то вони мають право ще й вимагати копії документів (електронні чи паперові – не уточнюється) засвідчені підписом платника або його посадовою особою та скріплені печаткою (якщо така є)	ч. 6 ст. 9 Закону про бухоблік, пп. 20.1.2 ПК кодексу, п. 2.3 Положення № 88

Таблиця 1.2.1 – Правила електронного документообороту 2022. [8]

В Україні вже діє система електронного документообігу. Наприклад, програма електронного документообігу Deals. Завдання системи – збільшити кількість документів, затверджених у строк. Іншими перевагами програми є:

- Скорочення часу підготовки та погодження документів у кілька разів.
- Економія на адміністративних та логістичних витратах.
- Забезпечує постійний доступ до документів.
- Підтримка всіх українських ACSC.

Щоб почати використовувати, вам потрібно лише підключитися до системи. Потім додайте власного контрагента і все – можете підписувати та обмінюватись документами онлайн.

Ще одна програма електронного документообігу – Megapolis.DocNet. Це дає можливість організувати систему електронного документообігу в компаніях будь-якого рівня та сфери діяльності.

Його також можуть використовувати державні установи, підприємства та установи. Система автоматизує документоорієнтовані бізнес-процеси, пов'язані із загальною та операційною діяльністю організації.[9]

### **1.3 Постановка задачі**

Необхідно створити веб-орієнтований додаток в якому є можливість автоматично зробити обчислення кількості затраченого часу при електронном документообігу. Користувач зможе надати дані при зіставленні з маючими даними в компанії, дозволить прийняти рішення про потреби впровадження технології електронного документообігу. Програма повинна дозволити малому бізнесу прийняти рішення без складних економічних і статистичних обчислень. Для реалізації інтерфейсу було створено веб-сторінка де користувач повинен буде заповнити такі поля:



- Кількість підрозділів
- Кількість підписантів
- Кількість повернень
- Пріоритетність документа

В результаті отримає мінімальне, середнє і максимальне кількість часу витраченого на прийняття рішення при електронному документообігу.

Постановка задачі:

1. Створення універсального алгоритму процесу для обробки документів
2. Розробка веб додатку «Аналіз доцільності електронного документообігу»
3. Порівняння результатів з веб додатком і функціонуваним програмним забезпеченням

## 2. ВИБІР ПРОГРАМНИХ ЗАСОБІВ

### 2.1 HTML для створення графічного інтерфейсу

HTML (мова гіпертекстової розмітки) — це код, який використовується для створення та відображення веб-сторінок та їх вмісту. Наприклад, ви можете організувати свій контент у кілька абзаців, маркованих списків або таблиць, використовуючи зображення та дані.[10]

HTML не є мовою програмування; це мова розмітки, яка повідомляє вашому браузеру, як відображати веб-сторінки, які ви відвідуєте. Він може бути складним або простим, залежно від потреб веб-дизайнера. HTML складається з серії елементів, які можна використовувати для розміщення або оточення різних частин вмісту, щоб вміст виглядав або поведився певним чином.[10]

Мова гіпертекстової розмітки HTML була запропонована Тімом Бернерсом-Лі в 1989 році як один із компонентів технології розробки розподіленої гіпертекстової системи Всесвітньої павутини. Ідея гіпертекстової інформаційної системи полягає в тому, що користувачі мають можливість переглядати документи (сторінки тексту) у найбільш зручному для них порядку, а не в тому порядку, в якому вони звикли читати книги. Це досягається шляхом створення спеціального механізму, який використовує гіпертекстові посилання для зв'язування різних сторінок тексту.[11]

Мова HTML дозволяє визначати структуру електронних документів на рівні друку. Отриманий документ може містити різноманітні елементи: ілюстрації, аудіо- та відеофрагменти. Мова HTML містить розширені інструменти для визначення багаторівневих заголовків, вибору шрифту, груп різних об'єктів і багато інших функцій.

Ключовим фактором, що вплинуло на розвиток мови HTML, було те, що мова HTML була обрана як основа для гіпертекстової бази даних простих

текстових файлів, які можна було створити в будь-якому текстовому редакторі на будь-якій апаратній платформі в будь-якій операційній системі.

Таким чином, гіпертекстова база даних у понятті WWW є набором текстових файлів, розмічених на мові HTML, що визначає формат подання інформації (розмітку) і структуру посилань (гіпертекстових посилань) цих файлів.

Модель розмітки документа HTML базується на моделі розмітки. Модель з тегами описує документи як набір контейнерів, кожен контейнер починається і закінчується тегом. Тобто документи HTML — це не що інше, як звичайні файли ASCII з доданими керуючими кодами HTML (тегами).

Теги документів HTML здебільшого прості та легкі у використанні, оскільки вони створюються з використанням загальноприйнятих англійських слів, чітких скорочень і символів.[11]

## **2.2 CSS мова стилів HTML**

CSS (скорочення від Cascading Style Sheets, перекладається як Каскадні таблиці стилів) — це спеціальна мова (мова стилів), написана мовою розмітки даних для опису зовнішнього вигляду документа (як і де відобразити елементи веб-сторінки). Здебільшого CSS використовується для документів, розмічених у HTML, XHTML та XML.

Якийсь час після того, як Tim Berners Lee винайшов World Wide Web, мова HTML використовувалася тільки для визначення структури вмісту сторінки. Розробники могли використовувати теги HTML, такі як `<h1>` і `<p>` лише для того, щоб позначити текст як «це заголовок» або «це абзац». Трохи пізніше став з'являтися особливий тег. Ви, напевно, пам'ятаєте, що були такі теги, як `<font>` та `<bgcolor>`. Дизайн веб-сторінки було виконано безпосередньо у змісті документа. Ці теги відрізнялися від інших тим, що вони визначали дизайн, а не структуру, більшість інших тегів мали такі властивості, як `color` і `border`.

Це бажання збільшити стилістичні можливості призвело до того, що необроблені структуровані теги, такі як `<table>`, все частіше використовуються в дизайні сторінок, а не структурований текст. Багато нових тегів, наприклад `<blink>`, підтримуються лише одним браузером. «Для перегляду цього ресурсу вам потрібен браузер X» — такі відмови стали звичним явищем на веб-сайтах.

При цьому в якийсь момент код сторінки став настільки громіздким і нечитаним, що стало очевидним, що це сміття, і згодом ним уже ніхто не користувався (HTML) та його візуальне оформлення (CSS). Саме після цього поділу необхідність у елементах стилю відпала і такі теги та атрибути поступово застаріли та розробники їх прибрали.[12]

### **2.3 Мова програмування JavaScript**

JavaScript було створено для того, щоб “оживити веб-сторінки”.

Програми на цій мові називаються скриптами. Їх можна писати прямо на сторінці в коді HTML і вони автоматично виконуються при завантаженні сторінки.

Скрипти надаються та виконуються як простий текст. Для запуску їм не потрібна спеціальна підготовка чи компілятор.[13]

JavaScript - єдина браузерна технологія, що поєднує у собі всі три. Це робить його унікальним. Ось чому JavaScript є найпопулярнішим інструментом створення інтерфейсів браузера.[13–15]

JavaScript («JS» для короткості) — це повноцінний динамічний мову програмування, який застосовується до HTML-документа та може забезпечити динамічну інтерактивність на веб-сайтах. Його розробив Брендан Айх, співоснователь проекту Mozilla, Mozilla Foundation і Mozilla Corporation.

JavaScript - неймовірно універсальний і зручний для початківців мову. Наш великий досвід дозволяє нам створювати ігри, анімовану 2D- та 3D-графіку, повноцінні програми для роботи з базами даних та багато іншого.

JavaScript сам по собі дуже компактний, але дуже гнучкий. Розробники створили велику кількість інструментів поверх основної мови JavaScript, зробивши доступними безліч додаткових функцій з мінімальними зусиллями.

Програмні інтерфейси додатків (API), вбудовані в браузері, надають різні функціональні можливості, такі як динамічне створення HTML і встановлення стилів CSS, захоплення та маніпулювання відеопотоком, робота з веб-камерою користувача або генерація 3D-графіки та аудіо-зразків.

Сторонні API від розробників забезпечують функціональність на своїх сайтах інших розробників, таких як Twitter або Facebook.[16]

Використання JavaScript робить інтерфейс більш зручним та інтуїтивно зрозумілим для користувачів. Ця програма може бути запущена з будь-якого пристрою з браузером та не потребує підключення до Інтернету.

Мова JavaScript спочатку створювалася тільки як мова для браузерів, але тепер вона використовується і в інших середовищах.

Сьогодні JavaScript вважається найбільш широко використовуваною мовою браузера, повністю інтегрованою з HTML/CSS.

Є багато мов які “транспілюються” в JavaScript і надають певні функції. Рекомендується переглянути їх, принаймні мигцем, після освоєння JavaScript.[17]

## **2.4 Аналіз та створення алгоритма роботи веб-додатку**

Алгоритм має на увазі під собою програмний аналіз швидкості обробки документів. Для роботи програми користувачеві потрібно ввести дані своєї організації які в себе включають :

- Кількість підрозділів
- Кількість робітників
- Кількість підписантів
- Пріоритетність
- Кількість повернень

Ці дані потрібні з урахуванням аналізу конкретного документу.

Для з поля є діапазон часу мінімальне, середнє і максимальний час який документ буде перебувати на різних етапах. На етапі «Підрозділ» затрачується мінімально 30 хвилин ,середнє -50 хвилин і максимальнo 80 хвилин. Наступним кроком переходить на обробку робітникам мінімально – 40 хвилин, середнє – 60 хвилин, максимальнo – 90 хвилин. Далі треба підписати документ відповідальними особами мінімально затрачується часу 10 хвилин, середнє – 30 хвилин, максимальнo 40 хвилин.

Після заповнення полів починається підрахунок часу затраченого на підписання документу. Якщо на полі «Пріоритетність» стоїть галочка, то час обробки і підписання скорочується 1,5 рази.

Змінна «Кількість повернень» задається числом. При підрахунку враховується, що документ повертається на перший етап рандомно з інших етапів і час обробки документу збільшується.

В кінці алгоритму секунди переводяться в дні, години, хвилини залежно від підрахунків за заданими даними.

### 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ

#### 3.1 Створення веб додатку з використанням мови програмування JavaScript.

Постійне збільшення обсягу документів, збільшення інформаційних потоків і швидкий розвиток цифрових технологій порушили функціонування організацій, довгий час обмежуваних традиційним документообігом, заснованим на масовому використанні паперу.

У цьому контексті електронний документообіг швидко зарекомендував себе як ідеальне рішення для будь-якої компанії, яка шукає документообіг. Однак функціональні можливості цього типу інструментів не завжди добре відомі, і сьогодні залишається поширеною плутанина між рішеннями, які сприймаються як подібні.[18]

Веб додаток має такий вигляд (див. рис. 3.1.2). Кожне поле відповідає за своє значення.

Аналіз доцільності електронного документообігу

Будь-ласка, заповніть всі поля форми!

Кількість підрозділів:

Кількість робітників:

Кількість підписів:

Пріоритетність документу

Кількість повернень:

ЗАПОВНИТИ ПОЛЯ    РОЗРАХУНОК    ЗАТЕРТИ ВВЕДЕНІ ДАНІ    ЗАТЕРТИ РЕЗУЛЬТАТИ

Мінімально витрачений час    Медіана витраченого часу    Максимально витрачений час

Рис.3.1.1 – Вигляд веб додаток

Поле «Кількість підрозділів» вносяться дані кількості відділів або департаментів або підрозділів для підписання документації.

Поле «Кількість робітників» задається значення загальну кількість робітників у кожному відділу, щоб опрацювати документ.

Поле «Кількість підписів» задається число. Підписувати можуть лише ті користувачі в яких є відповідна посада і електронний цифровий підпис.

«Пріоритетність» тобто важливість документу, потрібно поставити галочку якщо важливий або залишити пусте віконце. Так як швидкість оброки документу скорочується в 2 рази порівняно зі звичайним.

Останнє поле відповідає за «Кількість повернень» . Дуже часто документ не проходить перевірку різних підрозділів з першого разу. На це все грає фактор неувважності людини, яка заповнює його і відволікання на інші справи під час оброки.

В поле можна вписати від 1 до 99 числа. Чим більші числа тим довше буде робитися підрахунки.

Кожна кнопка відповідає за свій функціонал (див. рис. 3.1.2).

Аналіз доцільності електронного документообігу

Будь-ласка, заповніть всі поля форми!

Кількість підрозділів:  
Введіть число департаментів від 1 до 99

Кількість робітників:  
Введіть число робітників від 1 до 99

Кількість підписів:  
Введіть число підписів від 1 до 99

Пріоритетність документу

Кількість повернень:  
Введіть число повернень документа від 0 до кількості робітників

ЗАПОВНИТИ ПОЛЯ    РОЗРАХУНОК    ЗАТЕРТИ ВВЕДЕНІ ДАНІ    ЗАТЕРТИ РЕЗУЛЬТАТИ

Рис.3.1.2 – Вигляд веб додатку



Кнопка «Заповнити полі», дозволяє автоматично заповнити поля як показано на рисунку 3.1.1.

«Розрахунок» при вводу даних виконує підрахунок затраченого часу.

«Затерти введені дані», дозволяє обнулити всі введені дані раніше для подальшого вводу нових параметрів.

«Затерти результат» - прибрати виведені результати з екрану.

Якщо поглянути на рис.3.1.3 показано поле для виводу результатів програми. Кожне поле відповідає за своє значення : мінімально, медіана (тобто середнє) і максимально витраченого часу.

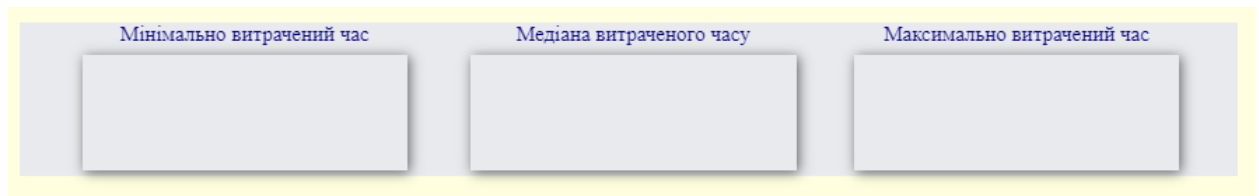


Рис.3.1.3 – Поле для виведення результатів програми

### 3.2 Тестування веб додатку

Розглянемо компанію «Х», для того щоб оформити звітність за минулий місяць, документ проходить 5 підрозділів, 8 робітників які будуть опрацьовувати цей документ і повинно бути 4 підписи від начальників відділів.

Відтворимо схему процесу для сприйняття шляху підписання документу для даного випадку. Також паралельно з цим врахуємо скільки часу затрачено при традиційному тобто паперовій формі документу.

За допомогою програми для електронного документообігу, відтворимо схематично шлях документу по підрозділах (див. рис. 3.2.1). Процес складається з таких етапів як :

1. Виконавець – це користувач, який додає в програму документ ( звіт, договір і т. д.)

2. Вибір підписантів, тобто відділи через які повинен пройти документ
3. Підпис начальника бюро, де виконавець працює
4. Підрозділ 2
5. Підрозділ 3
6. Підпис нач.бюро підрозділу 3
7. Підрозділ 4
8. Підпис нач.бюро підрозділу 4
9. Підрозділ 5
10. Підпис директора компанії
11. Переміщення документу в архів

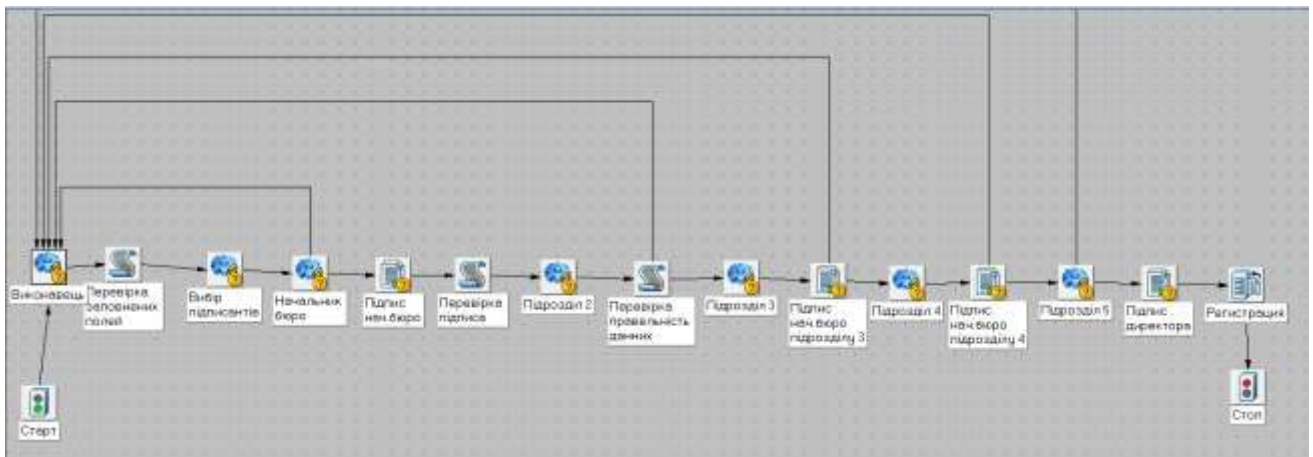


Рисунок 3.2.1- Схематичне відображення шляху документу(процес)

Розглянемо перший випадок, коли його повертають до виконавцю 3 рази (див. рис. 3.2.2).

Кількість підрозділів:  
5

Кількість робітників:  
8

Кількість підписів:  
4

Пріоритетність документу

Кількість повернень:  
3

Рисунок 3.2.2 – Випадок перший коли кількість повернень дорівнює 3

Затрачено часу (див. рис. 3.2.3), ми бачимо, що мінімально витрачено дорівнює 12 годин 30 хвилин, а максимальний 1 день 8 годин 40 хвилин.

Мінімально витрачений час	Медіана витраченого часу	Максимально витрачений час
Замер номер: 4, Результат: 12 годин : 30 хвилин : 0 секунд.	Замер номер: 4, Результат: 21 годин : 30 хвилин : 0 секунд.	Замер номер: 4, Результат: 1 день : 8 годин : 40 хвилин : 0 секунд.

Рисунок 3.2.3 – Результат витраченого часу при випадку 1

Другий випадок, коли його повертають на початок 3 рази і пріоритетність «важлива» (див. рис. 3.2.4).

Кількість підрозділів:  
5

Кількість робітників:  
8

Кількість підписів:  
4

Пріоритетність документу

Кількість повернень:  
3

Рисунок 3.2.4 – Випадок другий

Затрачено часу (див. рис. 3.2.5), ми бачимо, що мінімально затрачено дорівнює 8 годин 13 хвилин, а максимальний 16 годин.

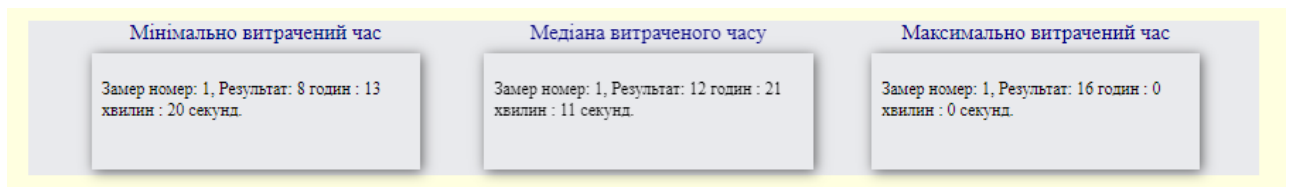


Рисунок 3.2.5 – Результат затраченого часу при випадку 2

Третій випадок, коли 0 повернень і пріоритетність «важлива» (див.рис.3.2.6).

Кількість підрозділів:  
5

Кількість робітників:  
8

Кількість підписів:  
4

Пріоритетність документу

Кількість повернень:  
0

Рисунок 3.2.6 – Випадок третій

Затрачено часу (див. рис. 3.2.7), ми бачимо, що мінімально витрачено дорівнює 5 годин 40 хвилин, а максимальний 10 годин 40 хвилин. Отже, пріоритетність документу впливає на час опрацювання самого файлу.

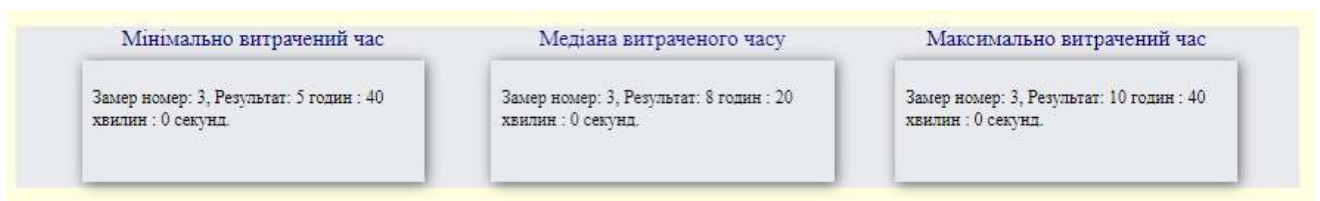


Рисунок 3.2.7 – Результат витраченого часу при випадку 3

Застосувавши ті самі дані для підписання паперового документу. Час який в середньому було витрачено за цими умовами склав 3 дні. Отже, можна зробити висновок, що для малих підприємств, які складають з двох і більше підрозділів потрібен електронний документообіг, а для тих в кого максимум 2 підрозділи за власним бажанням. Але треба враховувати, що зараз 21 століття і електронні документообіг зараз актуальні.

## ВИСНОВКИ

В даній магістерській роботі був створений веб додаток за допомогою якого зроблений аналіз доцільного переходу підприємств на електронний документообіг для даної компанії.

В ході роботи було проаналізовано усі можливі варіанти документообігу для компанії. Було виявлено, що електронний документообіг в рази скорочує час підписання, обробки і затвердження документу. Даний веб додаток допоможе компаніям оціни чи потрібно їм програми які пропонують ІТ компанії, для відтворення електронного документообіг.

За допомогою розрахунку затраченого часу витраченого на один файл, було показано настільки впливає пріоритетність документу, кількість повернень його на початок процесу і обробка співробітниками компанії. Чим частіше повертається документ тим довше він доходить до кінцевого шляху.

## СПИСОК ЛІТЕРАТУРИ

1. Про електронні документи та е... | від 22.05.2003 № 851-IV [Electronic resource]- Режим доступу: <https://zakon.rada.gov.ua/laws/show/851-15#Text>.
2. La GED - Gestion électronique des documents - Infos et devis [Electronic resource]- Режим доступу: <https://www.archivage-numerique.fr/ged>.
3. Obukhov A., Krasnyanskiy M., Nikolyukin M. Algorithm of adaptation of electronic document management system based on machine learning technology // Prog. Artif. Intell. 2020 94. Springer, 2020. Vol. 9, № 4. P. 287–303.
4. Haryawan C., Iswanti S. IMPLEMENTASI PIECES FRAMEWORK DALAM PERANCANGAN E-DOCUMENT MANAGEMENT SYSTEM WEBSITE PUSAT JAMINAN MUTU // J. Tekinkom (Teknik Inf. dan Komputer). Jurnal Ekonomi Dan Bisnis dan Jurnal Teknik Informasi Dan Komputer, Politeknik Bisnis Indonesia, 2021. Vol. 4, № 2. P. 137–143.
5. Baratov D.K., Aripov N.M., Ruziev D.K. Formalized Methods of Analysis and Synthesis of Electronic Document Management of Technical Documentation // 2019 IEEE East-West Des. Test Symp. EWDTs 2019. Institute of Electrical and Electronics Engineers Inc., 2019.
6. Mahmood A., Okumus I.T., Makalesi A. Design and Implementation of an Electronic Document Management System // Mehmet Akif Ersoy Üniversitesi Uygulamalı Bilim. Derg. Mehmet Akif Ersoy Üniversitesi Uygulamalı Bilimler Dergisi, 2017. Vol. 1, № 1. P. 9–17.
7. Електронний документообіг в Україні: як він працює | [buduysvoe.com](http://buduysvoe.com) [Electronic resource]- Режим доступу: <https://buduysvoe.com/publications/elektronnyy-dokumentoobig-v-ukrayini-yak-vin-prasyuye>.
8. Електронний документообіг замість паперового [Electronic resource]-

- Режим доступу: <https://buhplatforma.com.ua/article/9413-elektronniy-dokumentooborot-zamst-paperovogo>.
9. Електронний документообіг: що це таке та як його впровадити | Deals — майбутнє документообігу [Electronic resource]- Режим доступу: <https://dealssign.com/blog/elektronnij-dokumentoobig-shho-ce-take-ta-yak-jogo-vprovaditi/>.
  10. HTML basics - Learn web development | MDN [Electronic resource]- Режим доступу: [https://developer.mozilla.org/en-US/docs/Learn/Getting\\_started\\_with\\_the\\_web/HTML\\_basics](https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/HTML_basics).
  11. Лекція 3. Основні поняття мови HTML та структура документів - Вивчаємо WEB-дизайн дистанційно [Electronic resource]- Режим доступу: <https://sites.google.com/site/vivcaemowebdizajndistancijno/html/lekcia-3-osnovni-ponatta-movi-html-ta-struktura-dokumentiv>.
  12. Що таке CSS [Electronic resource]- Режим доступу: [https://css.in.ua/article/shcho-take-html\\_10](https://css.in.ua/article/shcho-take-html_10).
  13. Сучасний підручник з JavaScript [Electronic resource]- Режим доступу: <https://uk.javascript.info/>.
  14. Satyapriya Mishra. Most Essential Concepts of JavaScript.
  15. Bae S. JavaScript Data Structures and Algorithms // JavaScript Data Struct. Algorithms. Apress, 2019.
  16. JavaScript basics [Electronic resource]. 2020- Режим доступу: [https://developer.mozilla.org/en-US/docs/Learn/Getting\\_started\\_with\\_the\\_web/JavaScript\\_basics](https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/JavaScript_basics).
  17. Вступ до JavaScript [Electronic resource]- Режим доступу: <https://uk.javascript.info/intro>.
  18. Gestion Électronique des Documents (GED) : le guide complet [Electronic resource]- Режим доступу: <https://www.elo.com/fr-fr/gestion-electronique-de-documents-guide-complet.html>.



## ДОДАТОК

### Index.html

```

<!DOCTYPE html>

<html>
<head>
<meta charset="utf-8">
<script
src="http://ajax.googleapis.com/ajax/libs/jquery/1.5/jquery.min.js"></script>
<script
src="https://cdn.rawgit.com/zenorocha/clipboard.js/master/dist/clipboard.min.js">
</script>
<script src="js/clipboard.js"></script>
<script src="js/jquery.min.js"></script>
<script src="js/script.js"></script>
<script type='text/javascript'>
function validate(value) {
    if (/^[1-9][0-9]?$/i.test(value)) {

        document.getElementById('information').className =
"povidomlenna1";
        document.getElementById('information').innerHTML =
"Коректні значення, все добре";
        return(true);
    }
    alert("Перевірте будь ласка введені вами данні");
    document.getElementById('information').className =
"povidomlenna2";

```

```

        document.getElementById('information').innerHTML      ="Error!!
Змініть написані початкові дані";
        var          Past_in_dlock_bos                      =
document.getElementById('GOLOVN_ITOG1');
        Past_in_dlock_bos.innerHTML = " ";
        var Past_in_sky1 = document.getElementById('field_1');
        Past_in_sky1.innerHTML = " ";
        var field_2 = document.getElementById('field_2');
        field_2.innerHTML = " ";
        return (false);
    }

    function validate2(value) {
        if (/^[0-9]?$/i.test(value)) {
            document.getElementById('information').className      =
"povidomlenna1";
            document.getElementById('information').innerHTML      =
"Корректні значення, все добре";
            return(true);
        }
        alert("Будь-ласка, перевірте введені значення");
        document.getElementById('information').className          =
"povidomlenna2";
        document.getElementById('information').innerHTML          ="Помилка!!
Змініть написані початкові дані";
    }

    function validate3(value) {

```

```

var Department = document.getElementById("Department").value;
var Signatures = document.getElementById("Signatures").value;
var Refund = document.getElementById("Refund").value;
var Summ = +Department + +Signatures; // ToNumber();

    if( Refund <= Summ){
        document.getElementById('information').className =
"povidomlenna1";
        document.getElementById('information').innerHTML =
"Введені дані - ОК!";
        return(true);
    }
    alert("Кількість повернень не може бути більше ніж сума
підрозділів та підписів");
    document.getElementById('information').className =
"povidomlenna2";
    document.getElementById('information').innerHTML ="ПОМИЛКА!
ПЕРЕВІРТЕ ВВЕДЕНІ ЗНАЧЕННЯ";
    return (false);
}

</script>
<link rel="stylesheet" href="css/style.css">
</head>
<body>
<div class="title_prog">Аналіз доцільності електронного документообігу
</div>
<div class="page0">

```

```
<p class='povidomlenna' id="information" >Заповніть будь ласка форму!</p>
```

```
<div class="sky2">
```

```
<div class="label">Кількість підрозділів: </div><input type="number" min="1" max="99" id="Department" placeholder='Введіть число департаментів від 1 до 99' autocomplete="off" onchange='validate(this.value)' required > <br>
```

```
<div class="label">Кількість робітників: </div><input type="number" min="1" max="99" id="Workers" placeholder='Введіть число робітників від 1 до 99' autocomplete="off" onchange='validate(this.value)' required > <br>
```

```
<div class="label">Кількість підписів: </div><input type="number" min="1" max="99" id="Signatures" placeholder='Введіть число підписів від 1 до 99' onchange='validate(this.value)' name="" required > <br>
```

```
<div class="label">Пріоритетність документу</div>
```

```
<label class="container">
```

```
<input type="checkbox" checked="checked" id="Important" name="" > <br>
```

```
<span class="checkmark"></span>
```

```
</label>
```

```
<!-- <div class="label">Пріоритетність документу </div><input type="checkbox" checked="checked" id="Important" name="" ><br> -->
```

```
<div class="label">Кількість повернень: </div><input type="number" min="0" max="99" id="Refund" placeholder='Введіть число повернень документа від 0 до кількості робітників+3 (не більше 9) ' onchange='validate2(this.value)' name="" >
```

```
</div>
```

```

<div class="str0"></div>
<div class="str1"></div>
</div>
<div class="bg_block_kontr">
  <div class="block_kontr">
    <div class="block_button_ger">
      <div class="block_Кнопка_result"><input
class="Кнопка_zapolnit" type="button" value="Заповнити поля" ></div>
      <div class="block_Кнопка_result"><input
class="Кнопка_result" type="button"
onclick="validate(Department.value)&&validate(Signatures.value)&&validate(Wo
rkers.value)&&validate3(Refund.value)" value="Розрахунок" ></div>

      <div class="block_Кнопка_result"><input
class="Кнопка_erase_introduction_denmark" type="button" value="Затерти
введені дані" ></div>

      <div class="block_Кнопка_result"><input
class="Кнопка_erase_denmark" type="button" value="Затерти
результати"></div>
    </div>
    <div class="left_pale">
      <div class="title_GOLOVN_ITOG1">Мінімально
витрачений час</div>
      <div id="GOLOVN_ITOG1"
class="GOLOVN_ITOG1"></div>

    </div>
    <div class="right_pal">

```

```

        <div class="title_field_1">Медіана витраченого
часу</div>

        <div id="field_1" class="field_1"></div>

    </div>
    <div class="fin_sky2">
        <div class="title_field_1">Максимально витрачений
час</div>

        <div id="field_2" class="field_2"></div>

    </div>
</div>
</body>
</html>

```

### Script.js

```

$(document).ready(function(){
let zamer = 1;
new Clipboard('.bos_copy_but');
new Clipboard('.copy_but1');
new Clipboard('.copy_but2');
var Past_in_dlock_bos = document.getElementById('GOLOVN_ITOG1');
var Past_in_sky1 = document.getElementById('field_1');
var Past_in_sky2 = document.getElementById('field_2');
$(".Knopka_result").click(function(){
var Department = document.getElementById("Department").value;
var Workers = document.getElementById("Workers").value;
var Signatures = document.getElementById("Signatures").value;

```

```
var Important = document.getElementById('Important');  
var Refund = document.getElementById("Refund").value;  
var test = 0;
```

```
var maxTime = 0;  
var minTime = 0;  
var approxTime = 0;
```

```
for ( var i = 0; i < Department; i++) {  
  
    minTime    = minTime + 60*30;  
    approxTime = approxTime + 60*50;  
    maxTime    = maxTime + 60*80;  
  
};
```

```
for ( var i = 0; i < Signatures; i++) {  
  
    minTime    = minTime + 60*10;  
    approxTime = approxTime + 60*30;  
    maxTime    = maxTime + 60*40;  
  
};
```

```
for (var i = 0; i < Workers; i++) {  
  
    minTime    = minTime + 60*40;  
    approxTime = approxTime + 60*60;  
    maxTime    = maxTime + 60*90;
```

```

};

if (Refund >0 ){

    for (var i = 0; i < Refund; i++) {

        ComebackDepartment = Math.round((Math.random() *
Department) + 0.5);
        for ( var j = 0; j < ComebackDepartment; j++) {

            minTime = minTime + 60*30;
            approxTime = approxTime + 60*50;
            maxTime = maxTime + 60*80;
        };
        ComebackSignatures = Math.floor(Math.random()+(Signatures
/ Department));

        for ( var k = 0; k < ComebackSignatures; k++) {

            minTime = minTime + 60*10;
            approxTime = approxTime + 60*30;
            maxTime = maxTime + 60*40;
        };

    };
};

```



```
if (document.getElementById('Important').checked) {
```

```
    minTime          = minTime / 1.5;
    approxTime = Math.round(approxTime / 1.7);
    maxTime         = maxTime / 2;
```

```
};
```

```
minTime    = SecondsToDate(minTime);
approxTime = SecondsToDate(approxTime);
maxTime    = SecondsToDate(maxTime);
```

```
Past_in_dlock_bos.innerHTML = Past_in_dlock_bos.innerHTML +
"<br>" + "№" + zamer + ", Результат: "+ minTime;
```

```
Past_in_sky1.innerHTML = Past_in_sky1.innerHTML + "<br>" + "№"
+ zamer + ", Результат: "+ approxTime;
```

```
Past_in_sky2.innerHTML = Past_in_sky2.innerHTML + "<br>" + "№"
+ zamer + ", Результат: "+ maxTime;
```

```
zamer = zamer + 1;
```

```
});
```

```
function SecondsToDate(Time) {
```

```
    var hours = Math.floor(Time / 60 / 60);
```

```
    var minutes = Math.floor(Time / 60) - (hours * 60);
```

```

var seconds = Time % 60;
if (hours >= 24) { // 50 /24 =2/2561521512 = 2
    var days = Math.floor(hours / 24);
    hours = hours % 24
    if (days == 1){
        var formatted = days + ' день : '+ hours + ' годин : ' +
minutes + ' хвилин : ' + seconds + ' секунд.' ;
    }
    else{
        var formatted = days + ' днів : '+ hours + ' годин : ' +
minutes + ' хвилин : ' + seconds + ' секунд.' ;
    }

}
else{
    var formatted = hours + ' годин : ' + minutes + ' хвилин : ' +
seconds + ' секунд.' ;
}

return    formatted;
}

/*Вводимо значення за замовчуванням*/
$("#Кнопка_zapolnit").click(function(){

    document.getElementById("Department").value = 5;
    document.getElementById("Signatures").value = 2;
    document.getElementById("Workers").value = 10;
    document.getElementById("Important").checked = true;
    document.getElementById("Refund").value = 0;

```

```

        document.getElementById('information').className =
"povidomlenna1";
        document.getElementById('information').innerHTML = "Поля
Заповнені!";
    });

    /*Очищення заданих значень*/
    $(".Кнопка_erase_introduction_denmark").click(function(){

        document.getElementById("Department").value = "";
        document.getElementById("Signatures").value = "";
        document.getElementById("Workers").value = "";
        document.getElementById("Important").checked = false;
        document.getElementById("Refund").value = "";
        document.getElementById('information').className =
"povidomlenna";
        document.getElementById('information').innerHTML = "Заповніть
будь ласка форму!";
    });

    $(".Кнопка_erase_denmark").click(function(){

        var Past_in_dlock_bos =
document.getElementById('GOLOVN_ITOG1');
        Past_in_dlock_bos.innerHTML = "";
        var Past_in_sky1 = document.getElementById('field_1');
        Past_in_sky1.innerHTML = "";
        var Past_in_sky2 = document.getElementById('field_2');
        Past_in_sky2.innerHTML = "";
    });

```

```
});  
});
```

### Stayl.css

```
.povidomlenna{  
    text-align: center;  
    font-size: 20px;  
    color: blue;  
  
}  
.povidomlenna1{  
    text-align: center;  
    font-size: 20px;  
    color: green;  
  
}  
.povidomlenna2{  
    text-align: center;  
    font-size: 20px;  
    color: red;  
  
}  
.title_prog {  
    text-align: center;  
    font-size: 25px;  
    color: #2A3541;  
}  
.information {  
    text-align: center;
```

```
        font-size: 25px;
        color:blue;
    }
.GOLOVN_ITOG1 {
    width: 300px;
    height: 100px;
    /* border: 1px solid red; */
    overflow: auto;
    padding: 5px 0 5px 10px;
    box-shadow: 2px 2px 12px 0px rgba(50, 50, 50, 0.75);
}
.field_1 {
    width: 300px;
    height: 100px;
    /* border: 1px solid red; */
    overflow: auto;
    padding: 5px 0 5px 10px;
    box-shadow: 2px 2px 12px 0px rgba(50, 50, 50, 0.75);
}
.field_2 {
    width: 300px;
    height: 100px;
    /* border: 1px solid red; */
    overflow: auto;
    padding: 5px 0 5px 10px;
    box-shadow: 2px 2px 12px 0px rgba(50, 50, 50, 0.75);
}
.label{
    width: 350px;
```

```
        margin: 5px 0;
        display: inline-block;
        font-size: 25px;
    }
    .sky2 input,
    .PALATE1 input,
    .sky1 input{
        width: 600px;
        height: auto;
        padding: 0 0,5px;
        background-color: #fcfcfc;
        border: 5px solid #b3b2b2;
        color: #000;
        font-size: 20px;
        border-radius: 30px;
        box-shadow: inset 1px 3px 10px 0 #eeeeef;
    }

    .PALATE1 input{
        margin: 3px 0;
    }

    .page0{
        background: url("../image/1.jpg") no-repeat;
        background-size: cover;
        width: 1160px;
        height: 500px;
        position: relative;
        margin: 0 auto;
    }
```

```
.PALATE1 {
    position: absolute;
    top: 28%;
    left: 1%;
        z-index: 10;
}
.sky1 {
    position: absolute;
    top: 16%;
    left: 66%;
        z-index: 10;
}
.sky2 {
    position: absolute;
    top: 106px;
    left: 43px;
        z-index: 10;
    width: 181px;
    height: 220px;
}
.block_Knopka_result{
    margin: 0 auto;
    position: relative;
    display: inline-block;
}
.right_pal {
    display: inline-block;
    float: center;
```

```
    margin: 0 55px 0 60px ;
}
.fin_sky2 {
    display: inline-block;
    float: right;
    margin: 0 55px 0 0 ;
}
.left_pale {
    display: inline-block;
    float: left;
    margin: 0 0 0 60px;
}
.PALATE1 .int_et_0_0 {
    display: inline-block;
    margin: 0 0 0 41px;
}
.PALATE1 .int_et_0_1 {
    display: inline-block;
}

.sky1 .int_SP1_0_0 {
    display: inline-block;
}
.sky1 .int_et_0_1 {
    display: inline-block;
    margin: 0 0 0 40px;
}
.sky2 .int_SP2_0_0 {
    display: inline-block;
```



```
}  
.sky2 .int_SP2_0_1 {  
    display: inline-block;  
    margin: 0 0 0 40px;  
}  
  
.bg_block_kontr {  
    background: #FFFFE0;  
    border-top: 10px solid #FFFFE0;  
    padding: 0 0 100px 0;  
  
}  
input.copy_but1 {  
    margin: 10px 0 0 0;  
}  
input.bos_copy_but {  
    margin: 10px 0 0 0;  
}  
input.copy_but2 {  
    margin: 10px 0 0 0;  
}  
  
input.Knopka_erase_introduction_denmark,  
input.Knopka_zapolnit,  
input.Knopka_result,  
input.Knopka_erase_denmark {  
    padding: 6px 15px;  
    background-color: #B22222;  
    border: 40;
```

```
border-radius: 40px;
color: #fff;
font-size: 20px;
text-transform: uppercase;
font-weight: 200;
margin: 0 0 0 0;
cursor: pointer;
}
input.Knopka_erase_introduction_denmark:hover,
input.Knopka_zapolnit:hover,
input.Knopka_result:hover,
input.Knopka_erase_denmark:hover{
    background: #B22222;
}
.title_GOLOVN_ITOG1,
.title_field_1,
.title_field_2 {
    text-align: center;
    font-size: 20px;
    margin: 0 0 7px 0;
    color: #000080;
}
input#copy_but2,
input#bos_copy_but,
input#copy_but1 {
    background: #0B5A6F;
    padding: 6px 15px;
    border: 0;
    border-radius: 3px;
```

```
color: #fff;
font-size: 15.4px;
font-weight: 600;
margin: 8px 0 0 0;
cursor: pointer;
}
input#copy_but2:hover,
input#bos_copy_but: hover,
input#copy_but1: hover {
    background: #047390;
}
.block_user input: focus,
.block_router input: focus,
.PALATE1 input: focus,
.path_blocking1 input: focus,
.path_blocking2 input: focus,
.sky2 input: focus,
.sky1 input: focus {
    border-color: #74d36b;
}
.block_user {
    position: absolute;
    top: 13px;
    left: 622px;
}
.block_user .label {
    width: 67px;
}
.block_router {
```

```
    position: absolute;
    top: 195px;
    left: 254px;
        z-index: 10;
}

.block_router .label{
    width: 150px;

}

.path_blocking1 {
    position: absolute;
    top: 123px;
    left: 520px;
        z-index: 10;
}

.path_blocking1 .label{
    width: 150px;

}

.path_blocking2 {
    position: absolute;
    top: 270px;
    left: 520px;
        z-index: 10;
}

.path_blocking2 .label{
```

```
    width: 150px;

}

.block_kontr {
    width: 1160px;
    height: auto;
    margin: 0 auto;
    background: #E9EAED;
}

.dock{
    margin: -20px 0 0 -155px;
    text-align: center;
}

.custom-checkbox {
    position: absolute;
    z-index: -1;
    opacity: 0;
}

.custom-checkbox+label {
    display: inline-flex;
    align-items: center;
    user-select: none;
}

.container {
    display: block;
    position: relative;
    padding-left: 35px;
    margin-bottom: 12px;
```

```
cursor: pointer;
font-size: 22px;
-webkit-user-select: none;
-moz-user-select: none;
-ms-user-select: none;
user-select: none;
}

.container input {
  position: absolute;
  opacity: 0;
  cursor: pointer;
  height: 0;
  width: 0;
}

/* Create a custom checkbox */
.checkmark {
  position: absolute;
  top: 0;
  left: 0;
  height: 25px;
  width: 25px;
  background-color: #eee;
}

.container:hover input ~ .checkmark {
  background-color: #ccc;
}
```

```
.container input:checked ~ .checkmark {  
  background-color: #2196F3;  
}  
  
.checkmark:after {  
  content: "";  
  position: absolute;  
  display: none;  
}  
  
.container input:checked ~ .checkmark:after {  
  display: block;  
}  
  
.container .checkmark:after {  
  left: 9px;  
  top: 5px;  
  width: 5px;  
  height: 10px;  
  border: solid white;  
  border-width: 0 3px 3px 0;  
  -webkit-transform: rotate(45deg);  
  -ms-transform: rotate(45deg);  
  transform: rotate(45deg);  
}  
  
.custom-checkbox+label::before {  
  content: " ";  
  display: inline-block;
```

```

width: 1em;
height: 1em;
flex-shrink: 0;
flex-grow: 0;
border: 1px solid #adb5bd;
border-radius: 0.25em;
margin-right: 0.5em;
background-repeat: no-repeat;
background-position: center center;
background-size: 50% 50%;
}
.custom-checkbox:checked+label::before {
border-color: #0b76ef;
background-color: #0b76ef;
background-image: url("data:image/svg+xml,%3csvg
xmlns='http://www.w3.org/2000/svg' viewBox='0 0 8 8'%3e%3cpath fill='%23fff'
d='M6.564 7.51-3.59 3.612-1.538-1.55L0 4.26 2.974 7.25 8
2.193z'/%3e%3c/svg%3e");
}
p
{
text-indent: 1.5em;
text-align: justify;
}
.block_button_ger{
background: #FFFFFFE0;
text-align: center;
height: 65px;
}

```