

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКА ДЕРЖАВНА ІНЖЕНЕРНА АКАДЕМІЯ

Причиненко Максим Анатолійович

УДК 378:002.1

**ПОРІВНЯЛЬНЕ ДОСЛІДЖЕННЯ МЕТОДІВ СТВОРЕННЯ
СИСТЕМ ДОКУМЕНТООБІГУ ВНЗ**

8.05010301 – програмне забезпечення систем

АВТОРЕФЕРАТ
дипломної роботи магістра

Запоріжжя – 2016

Дипломною роботою магістра є рукопис

Робота виконана на кафедрі програмного забезпечення автоматизованих систем Запорізької державної інженерної академії

Науковий керівник: доктор технічних наук, професор
Шамровський Олександр Дмитрович
завідувач кафедри програмного забезпечення
автоматизованих систем Запорізької державної
інженерної академії

Офіційний рецензент: Ребро Юрій Володимирович,
начальник відділу технічного обслуговування АСУ,
ВАТ “Запоріжсталь”

Захист відбудеться «14» січня 2016 р. о 9⁰⁰ на засіданні Державної екзаменаційної комісії Запорізької державної інженерної академії за адресою: 690006 м. Запоріжжя, пр. Леніна 226, адміністративний корпус, ауд. 40.

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність роботи

У сучасному світі ефективність управління організацією значною мірою залежить від часу, що витрачається на створення та одержання документів, контроль за їх виконанням, організацію збереження, пошук та використання. Оптимізація цих процесів може значно скоротити час та ресурси організації.

Типова система електронного документообігу (СЕД) забезпечує повну підтримку процесів створення, управління, обробки і поширення електронних документів між відповідальними посадовими особами та контроль за потоками документів. Головне призначення СЕД - автоматизувати весь комплекс документаційних потоків, а саме, введення у систему документів, їх реєстрацію, розподіл і розсилання, редагування, оперативне збереження, пошук і перегляд, відтворення, контроль виконання, розмежування доступу до документації, прискорення термінів опрацювання документів, вдосконалення механізмів організації та виконання документів, тощо.

У зв'язку з цим, з метою прискорення обробки документації, підвищення якості цієї роботи, зменшення матеріальних затрат та людських ресурсів впровадження системи документообігу є актуальним.

На поточний момент існує багато СЕД. Здебільшого вони розповсюджуються на платній основі та представляють собою комплексні рішення закритого типу. Правильне налаштування такої системи може бути досить складним завданням, що потребує багато часу. Якщо певні функції комплексного рішення не потрібні організації, то замість спрощення вони можуть ускладнити робочий процес. При цьому доробка готової СЕД та інтеграція з іншими системами може здійснюватися на платній основі або не здійснюватися взагалі. Таким чином, завжди актуальним залишається питання створення такої системи документообігу, що відповідає вимогам конкретної організації.

Створення власної СЕД дозволить інтегрувати це рішення з іншими підсистемами АСУ вищого навчального закладу. Крім того, це полегшить процес подальшої підтримки та модифікації системи у випадку зміни вимог, оскільки ВНЗ не буде залежати від певного виробника програмного забезпечення.

Мета і задачі дослідження

Метою кваліфікаційної роботи магістра є розробка системи електронного документообігу для вищого навчального закладу. Створена система повинна задовольняти вимогам, що висуваються Запорізькою державною інженерною академією.

Досягнення поставленої мети передбачає:

- дослідження предметної області;
- аналіз вимог організації до системи документообігу;

- аналіз існуючих систем електронного документообігу;
- проектування та розробка системи електронного документообігу;
- впровадження системи електронного документообігу;
- перевірку відповідності створеної системи висунутим вимогам;
- аналіз працездатності обраного методу створення СЕД.

Наукова новизна

Перевірена можливість використання сучасних веб-технологій для створення систем електронного документообігу організації.

Практична значимість

На основі результатів досліджень, проведених у рамках даної роботи, розроблено систему електронного документообігу для вищого навчального закладу “ВеeDoc”. Дана система підтримує можливості реєстрації, пошуку та архівування внутрішніх документів організації, сповіщення користувачів про надходження нових документів та контроль їх виконання, здійснення зворотнього зв'язку з автором документа. У разі необхідності система може бути легко інтегрована до складу АСУ вищого навчального закладу. Створена система впроваджується у загальному відділі Запорізької державної інженерної академії.

Апробація роботи

Основні положення й результати досліджень повідомлені й обговорені на XX науково-технічній конференції студентів, магістрантів, аспірантів і викладачів ЗДІА на секції програмного забезпечення автоматизованих систем, 20-24 квітня 2015 р. (м. Запоріжжя).

Публікації

Причиненко М.А., магістрант, Шамровський О.Д., проф., д.ф.-т.н. — науковий керівник. «Порівняльне дослідження та створення системи документообігу у ВНЗ.» Матеріали XX науково-технічної конференції студентів, магістрантів, аспірантів і викладачів ЗДІА «ЕЛЕКТРОНІКА, АВТОМАТИЗОВАНІ СИСТЕМИ ТА СУЧАСНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ» том III, Запоріжжя, ЗДІА, 2015.

Структура й об'єм роботи

Кваліфікаційна дипломна робота магістра складається зі вступу, 4 розділів і висновків, списку використаних джерел.

ЗМІСТ РОБОТИ

У **вступі** показана роль систем електронного документообігу в ефективному управлінні організацією, визначено основні цілі та задачі роботи, оцінено новизну, актуальність та практичне значення розробленої системи.

У **першому розділі** поставлені задачі для реалізації програмної системи. Коротко описано процес внутрішнього документообігу у Запорізькій державній інженерній академії, що існує на поточний момент.

В цьому розділі наведені теоретичні дані про документообіг. Зокрема, розглянуто такі явища, як: документаційне забезпечення управління, діловодство, документообіг та документопотік. Наведено тлумачення електронного документа згідно з Законом України “Про електронні документи та електронний документообіг” та визначено його правовий статус. Стисло описана технологія електронного цифрового підпису.

Детально розглянуто питання автоматизації документообігу організації. Вказані проблеми традиційної технології управління документообігом, зокрема, практична неможливість централізовано відслідковувати рух документів організації в реальному часі. Перелічені можливі напрями використання електронних технологій у традиційному діловодстві (підготовка документів, їх передача та реєстрація в інформаційній системі).

Виділено переваги використання електронного документообігу (можливість використання мультимедійних даних та заздалегідь заготовлених форм, велика швидкість передачі інформації великій кількості адресатів, економія паперу, компактність архівів, висока швидкість пошуку і одержання інформації, можливість захисту документів від несанкціонованого доступу), основні цілі його впровадження (підвищення ефективності управлінської діяльності, прискорення руху документів, зменшення трудомісткості опрацювання документів) та проблеми, що можуть при цьому виникнути (консерватизм персоналу, недостатня освіченість в галузі інформаційних технологій, часті структурні зміни в організації, слабка формалізація бізнес-процесів, паралельне існування традиційних “паперових” та електронних систем документообігу та інші).

Вказані найважливіші характеристики систем електронного документообігу (програмна платформа, можливості масштабування, максимальна кількість користувачів, можливість роботи без “жорсткої” фіксації структур, засоби визначення маршрутних схем проходження документів, можливості контролю за проходженням документів, засоби повідомлення про порушення у регламенті проходження документів, підтримка роботи з кількома версіями документа, можливість інтеграції з іншими додатками, особливості налаштування системи на потреби конкретного замовника, засоби регламентації доступу та криптологічного захисту, інші) та їх функції в інформаційному забезпеченні систем управління (підвищення якості підготовки документів, посилення виконавської дисципліни, прискорення обміну інформацією, спрощення процедури прийняття рішень, скорочення кількості помилок, прискорення

і спрощення інформаційних процесів, підвищення гнучкості й керованості процесів передавання інформації, забезпечення прозорості прийняття рішень). Перелічено основні вимоги до СЕД (масштабованість, розподіленість, модульність, відкритість).

Наведено аналіз існуючих СЕД. На ринку представлений широкий спектр систем (наприклад, WWS Docs, LanDoc, OPTIMA-WorkFlow, ДЕЛО, МОТИВ, Ориент, TerraDocs, NauDoc, 1С:Архив, Логика СЭД, АЛЬФА ДОК, ЕВФРАТ-Документооборот, Magnetico, ТЕЗИС, Доцero та інші). Серед них є як комерційні, так і безкоштовні. З метою ознайомлення з функціональністю та принципами побудови реальних систем електронного документообігу було розглянуто системи Дело, FossDoc, OPTIMA-WorkFlow. Дані продукти обрано випадковим чином серед тих, що надають вільний доступ до демонстраційної версії. Для них вказано компанію-виробника, цінову політику, вимоги до серверної та клієнтської частин продукту, перераховано основний функціонал, показані приклади зовнішнього вигляду, виділені сильні та слабкі сторони. В цілому, системи електронного документообігу — це складні програмні комплекси, що, зазвичай, складаються з багатьох модулів, кожний з яких покликаний вирішувати окрему задачу. Використання правильно налаштованої СЕД може значно спростити та пришвидшити процес документообігу в організації, оскільки вони реалізують весь потрібний для цього функціонал. Проаналізовані системи мають різну цінову політику, різні вимоги до оточення та дещо різні функціональні можливості.

СЕД Дело має можливість за рахунок додаткових підсистем організувати віддалену роботу співробітників та централізований документообіг для територіально віддалених підрозділів компанії.

Система FossDoc має вбудований поштовий сервер та дозволяє дуже гнучко налаштувати список модулів при замовленні.

У системі Optima-WorkFlow СЕД реалізована як окреме типове рішення та має проблеми з адаптивністю інтерфейсу, помилки при відображенні форм.

В основі всіх цих систем лежить однакова функціональність, що пов'язана з реєстрацією, пошуком, контролем виконання документів та контролем доступу до них. Дані системи мають складний інтерфейс з великою кількістю кнопок, полів вводу та вкладок, що негативно впливає на зручність користування програмою. Складність інтерфейсу викликана великою кількістю функцій, що доступні користувачеві в кожний конкретний момент часу. Всі розглянуті системи надають користувачеві доступ до демонстраційної версії продукту та документацію з його використання. Проте, модифікація даних продуктів під потреби конкретної організації, інтеграція з існуючими підсистемами без допомоги виробника може виявитися неможливою.

Аналіз показав доцільність створення власної системи документообігу, що відповідає вимогам конкретної організації. Функціональність такої системи відповідає потребам користувача, що дозволить спростити інтерфейс та полегшить взаємодію з програмою. Дане рішення можна інтегрувати з іншими підсистемами АСУ вищого навчального закладу. Крім того, це полегшить

процес подальшої підтримки та модифікації системи у випадку зміни вимог, оскільки ВНЗ не буде залежати від виробника програмного забезпечення.

У **другому розділі** наведено аналіз способів створення існуючих СЕД. В цілому, всі наведені системи реалізовані з використанням клієнт-серверної архітектури. Вони включають в себе сервер баз даних, сервер застосунків та клієнтський застосунок. Іноді можуть включатися веб-сервер, веб-клієнт та мобільні застосунки. Функціональність системи реалізується за рахунок модулів, які можна комбінувати для досягнення бажаного набору функцій.

Дані СЕД мають схожий принцип роботи з документами: для представлення документа в системі використовується реєстраційна картка документа. З картою пов'язується певний набір полів-атрибутів та завантажені файли. Контроль виконання здійснюється за рахунок створення спеціальних сутностей (наприклад, доручення в системі FossDoc), які пов'язують користувача з картою документа.

Використання клієнт серверної архітектури для вирішення даної задачі виправдане. Адже наявність серверу застосунків забезпечує централізацію процесу роботи з документами. Це значно полегшує вирішення питань пошуку, захисту та контролю доступу до інформації. Також цей підхід надає можливість розподілення та масштабування системи, бо зміна кількості та конфігурації серверів баз даних та серверів застосунків не впливає на функціональність клієнта.

Окремим питанням залишається відкритість системи. Цей елемент (із зрозумілих причин) відсутній у комерційних системах. Адекватне доопрацювання елементів таких систем можливе лише за допомогою компанії-розробника. Відкритість системи можна досягти шляхом використання Open Source рішень або створенням власної СЕД. Слід зазначити, що модифікація стороннього рішення може бути досить складним завданням і потребувати розуміння його архітектури.

Цікавим є питання вибору типу клієнтського застосунока. Він входить до складу всіх наведених систем електронного документообігу у вигляді desktop-застосунка. У такому підході є позитивні та негативні сторони. До позитивних сторін використання desktop-застосунка можна віднести можливість взаємодії з іншими застосунками, що встановлені на комп'ютері (наприклад, для взаємодії з користувачем, використання принтера або сканера та інше). В той же час, він накладає додаткові вимоги на клієнтський комп'ютер (наприклад, на операційну систему або кількість оперативної пам'яті).

Альтернативними варіантами є веб-клієнт та мобільні застосунки, але вони, зазвичай, позиціуються як додаткова функціональність, що може бути включена до системи за додаткову плату. Веб-клієнт не може взаємодіяти з іншими застосунками, але він доступний з будь-якого комп'ютера чи мобільного пристрою, що підключений до мережі. Цей підхід доволі обмежений у функціональному плані, але дуже мобільний і не залежить від платформи.

Мобільний застосунок має можливість взаємодіяти з іншими застосунками. Крім того, зазвичай, люди носять мобільні пристрої із собою, що ро-

бить цей тип застосунків ідеальним для різного роду повідомлень. Даний підхід також дуже мобільний, але обмежений у зв'язку із специфікою використання мобільних пристроїв — малий екран може заважати повноцінній роботі з системою. Слід також зазначити, що на мобільному пристрої може використовуватися веб-клієнт, а мобільний застосунок створює додаткові вимоги для операційної системи пристрою.

Таким чином, в ідеальному випадку система повинна мати всі три типи клієнтів для виконання різних видів задач. Це зробить систему максимально повною та доступною для всіх користувачів. Але тоді може виникнути необхідність одночасно підтримувати декілька версій одного коду, що розраховані на різні платформи.

Якщо вимоги до СЕД не потребують від клієнтського застосунка складної функціональності, то краще створити веб-клієнт. Він буде працювати як на комп'ютерах, так і на мобільних пристроях незалежно від операційної системи та не створюючи великого навантаження на комп'ютер клієнта. Веб-клієнт не потребує підтримки декількох версій програми для різних платформ. Звісно, залишається відкритим питання з функціональної обмеженості такого підходу.

Для вирішення цієї проблеми можна спробувати застосувати суміжні технології нахшталт включення Java-апплетів, Flash-роликів або SilverLight-застосунків. Проте, такий підхід негативно вплине на кроссплатформенність застосунка, бо ці технології можуть не підтримуватися в окремих середовищах виконання (наприклад, браузері Safari в iOS не підтримують можливості включення на сторінку Java-апплетів).

Альтернативним варіантом, що частково може вирішити проблему обмеженості функціональних можливостей клієнта є перенесення логіки обробки даних на сторону сервера. В такому випадку клієнтський застосунок залишається "тонким" і, здебільшого, виконує роль інтерфейса для програми на сервері. Цей підхід також позитивно впливає на можливість централізованого контролю за процесами, що відбуваються у системі.

Обмежень на серверну частину СЕД менше. Для її реалізації потрібна мова програмування, що підтримує роботу з сокетом або інші можливості для комунікації з клієнтом (наприклад, PHP, Java, C#, C++, C, Python, JavaScript та інші). Крім того, при необхідності, серверна частина має можливість спілкуватися з іншими застосунками та здійснювати запити до інших серверів. Чітких обмежень в даному плані (на відміну від клієнтського застосунка) немає, тому технології потрібно обирати на основі конкретних вимог до системи та особистих уподобань розробника.

Висунуті вимоги до СЕД не передбачають великої кількості одночасних звернень до сервера, тому асинхронність технології не відіграє вирішальної ролі при виборі серверної мови програмування. Для реалізації системи можна обрати мову PHP. Близько 20 років тому ця мова була створена спеціально для розробки веб-застосунків. За час свого розвитку вона вдосконалювалася та набирала популярність серед розробників. На поточний момент PHP являється однією з найпопулярніших мов для створення динамічних веб-сайтів та

підтримується більшістю хостинг-провайдерів. Крім того, у випадку необхідності можливості цієї мови можуть розширюватися за рахунок написання розширень на мові С. PHP має С-подібний синтаксис, підтримку ООП, підтримку з боку інтегрованих середовищ розробки, можливість відлагодження та профілювання коду, велике ком'юніті розробників, велику кількість доступних модулів, менеджери пакетів та сучасні фреймворки, що активно розвиваються — все це робить її потенційно дуже зручною для розробки.

Слід відзначити, що для реалізації повідомлення користувачів про події у системі може знадобитися постійне з'єднання між клієнтом та сервером. Однак, специфіка мови PHP передбачає її використання за моделлю запит-відповідь, тому після обслуговування клієнта обробник “помирає”. Виходячи з цього, реалізація постійного з'єднання з клієнтом на PHP не доцільна (хоча і можлива), якщо є можливість використати для цього більш відповідні інструменти.

Наприклад, близько 6 років тому, з'явилася платформа Node.js, що дозволяє використовувати JavaScript в якості мови загального застосування, та, зокрема, для реалізації веб-серверів. В основі цієї технології лежить подієво-орієнтоване та асинхронне програмування з неблокуючим вводом/виводом, що дозволяє серверам добре утримувати навантаження при великій кількості користувачів. Для створення постійного зв'язку між клієнтом та сервером можна використати технологію веб-сокетів. На поточний момент існують реалізації даного протоколу як для клієнтської частини, так і для платформи Node.js.

Для збереження даних в системі потрібна база даних. В даному випадку краще використовувати реляційну базу даних, оскільки структура даних чітко визначена. Серед доступних баз даних можна обрати MySQL. Дана СУБД підтримує роботу з транзакціями, має контроль доступу, добре інтегрується з PHP (це поєднання є “класичним”) та розповсюджується безкоштовно.

Під час створення веб-клієнта розробник обмежений у виборі стеку технологій. Для реалізації застосунка зазвичай використовуються мови HTML, CSS, JS. Останні версії цих технологій дозволяють створювати інтерактивні та зручні інтерфейси, з якими приємно працювати. При необхідності можуть використовуватися суміжні мови, шаблонізатори, системи збірки та інше. При написанні JavaScript кода можна використати бібліотеку jQuery, що значно спрощує взаємодію з DOM-моделлю документа, роботу з асинхронними запитами на сервер, обробку подій та має велику кількість плагінів для реалізації складних елементів управління. Для реалізації постійного з'єднання з сервером обрано бібліотеку socket.io, яка реалізує протокол веб-сокетів одночасно на стороні клієнта та сервері.

Третій розділ містить опис проектування та розробки системи. Зокрема, в ньому вказані вимоги до програми, вимоги до апаратного та програмного забезпечення, вимоги до користувача, організаційні вимоги, засоби реалізації, модулі та алгоритми, структури даних та проект інтерфейсу системи.

Згідно з висунутими вимогами створювана система електронного доку-

ментообігу повинна реалізовувати внутрішній документообіг ВНЗ та надавати наступні функції для роботи з документами:

- реєстрація документів;
- розсилка документів на виконання;
- контроль виконання документів;
- пошук документів.

Крім того, на поточний момент документообіг здійснюється виключно у «паперовій» формі і на документах можуть стояти важливі підписи та печатки. Тому необхідно реалізувати можливість додавання до електронної версії документів відскановані копії паперових документів. Крім копій можна також додавати інші файли у різних форматах.

Стосовно контролю виконання, система електронного документообігу повинна надавати можливість виставляти декілька термінів виконання документів та обирати одразу декількох виконавців.

До моменту інтеграції системи документообігу в загальну АСУ вищого навчального закладу вона повинна мати можливість існувати окремо. Тобто, вона повинна містити модуль керування користувачами та правами доступу. Необхідно врахувати можливість подальшої роботи системи в рамках АСУ.

Діаграма варіантів використання наведена на рисунку 1.

Для реалізації клієнтської частини використовуються наступні технології: мови HTML5, CSS3, JavaScript; бібліотека jQuery та плагіни до неї; бібліотека socket.io; текстовий редактор Sublime Text 3; браузері Mozilla Firefox та Google Chrome; VCS Git та сайт bitbucket.org.

Для реалізації серверної частини використовуються наступні технології: мови PHP 5.4, JavaScript, SQL; фреймворк Yii2; веб-сервер Apache; платформа Node.js; база даних MySQL; текстовий редактор Sublime Text 3; VCS Git та сайт bitbucket.org; MySQL Workbench.

Система складається з наступних частин: веб-клієнт, веб-сервер, сервер веб-сокетів та сервер баз даних. Схема розгортання наведена на рисунку 2.

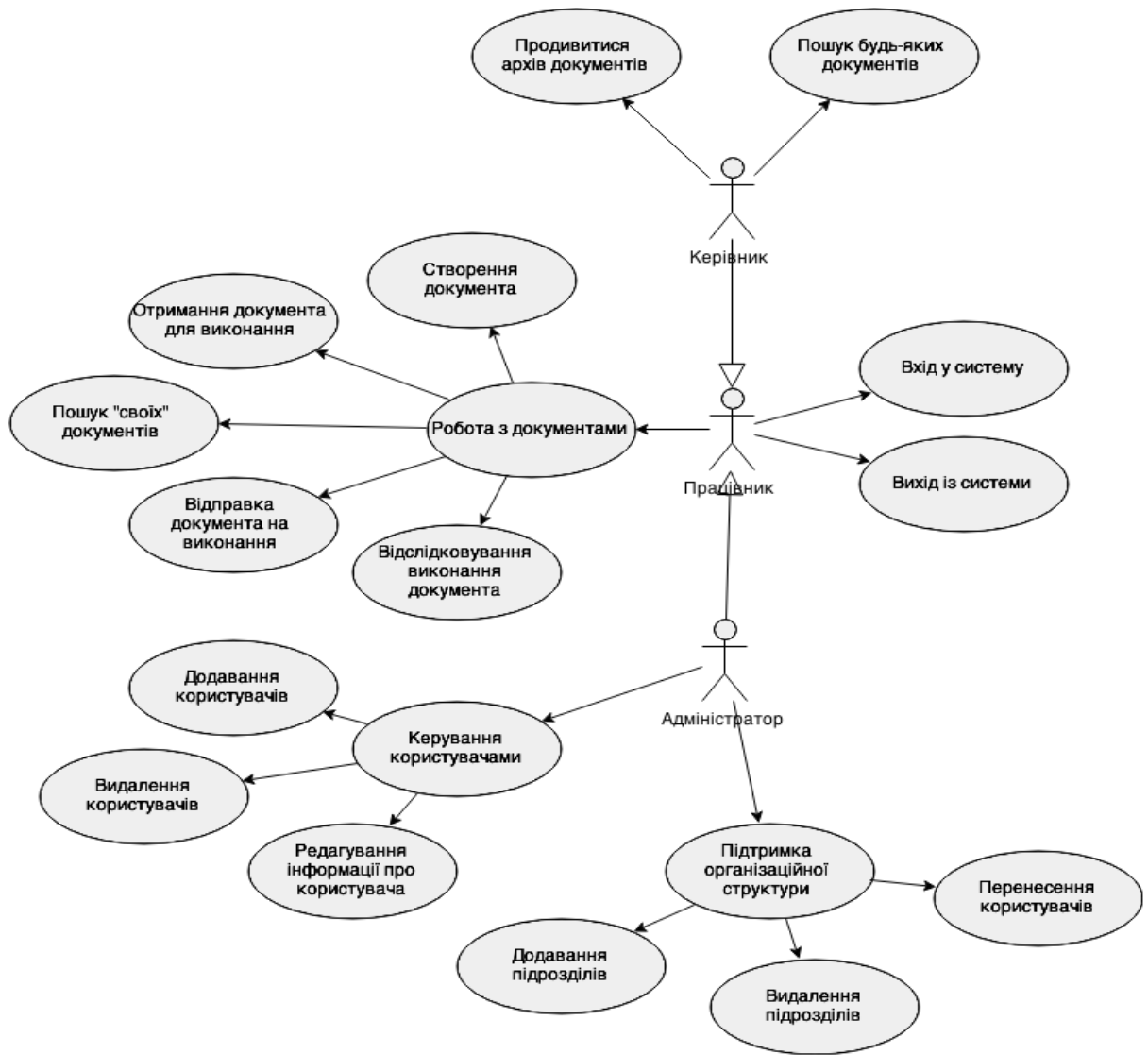


Рис. 1 Діаграма варіантів використання

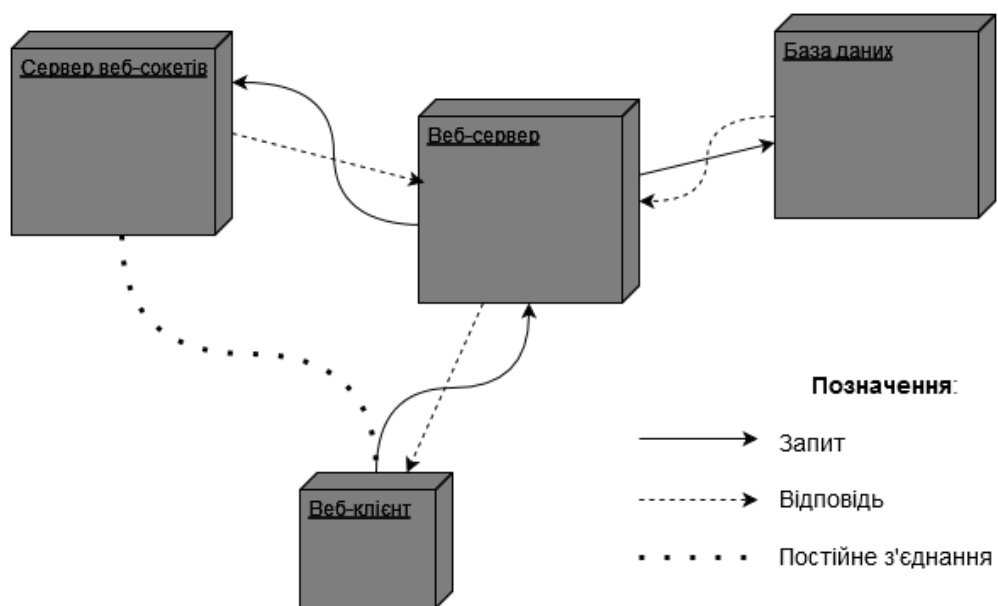


Рис. 2 Схема розгортання

Веб-сервер відповідальний за виконання всіх операцій з користувачами, документами, підрозділами, тощо. Це єдина централізована точка доступу до будь-якої інформації в системі. Тут реалізується механізм перевірки прав доступу до інформації.

Сервер веб-сокетів відповідальний за підтримку постійного з'єднання з веб-клієнтом. Він використовується для розсилки користувачам повідомлень про події, що виникають у системі. Веб-сервер повідомляє сервер веб-сокетів про події за допомогою HTTP запитів використовуючи протокол JSON-RPC 2.0. Після отримання повідомлення про подію сервер веб-сокетів розсилає цю інформацію адресатам, що знаходяться онлайн.

Сервер баз даних використовується для зберігання даних. Тут зберігаються всі дані, крім файлів, що прикріплені до документів. Файли зберігаються окремо. На цьому рівні реалізовані лише базові перевірки прав доступу до таблиць у цілому. Доступ до окремих сутностей реалізується на рівні веб-серверу. Такий підхід краще реалізується з обраним фреймворком та дозволяє у випадку необхідності легко замінити базу даних без необхідності довгого переконфігурування системи.

Клієнтська частина представляє собою “зовнішню” частину веб-сайту та служить у якості інтерфейса користувача. Тут містяться всі графічні елементи управління системи електронного документообігу, до яких має доступ користувач. Клієнтська частина спілкується з веб-сервером за допомогою HTTP запитів та з сервером веб-сокетів за допомогою протоколу веб-сокетів.

Структура бази даних наведена на рисунку 3.

Таблиця **user** зберігає інформацію про профілі користувачів системи. Зберігаються лише дані, що необхідні для розпізнавання користувача. Детальна інформація про користувачів міститься в АСУ.

Таблиці **auth_item**, **auth_item_child**, **auth_rule** та **auth_assignment** зберігають необхідну інформацію для реалізації RBAC системи контролю доступу до елементів системи електронного документообігу. Зокрема, таблиця **auth_item** зберігає основну інформацію про елементи авторизації (таку як тип елемента, його назва, опис, тощо). Таблиця **auth_item_child** дозволяє будувати з елементів **auth_item** ієрархічне дерево прав доступу. Таблиця **auth_rule** містить інформацію про бізнес-правила, що застосовуються до елементів ієрархії для перевірки прав доступу. А таблиця **auth_assignment** надає можливість пов'язувати конкретних користувачів системи з ролями, задачами та операціями.

Таблиця **employee** є частиною АСУ організації. При необхідності користувачі СЕД можуть бути пов'язані з детальною інформацією про співробітників ВУЗу.

Таблиця **department** зберігає базову інформацію про структурні підрозділи організації. Ця таблиця не є частиною безпосередньо системи документообігу, а являється елементом АСУ.

Таблиця **employee_department** показує зв'язок користувачів з різними структурними підрозділами організації.

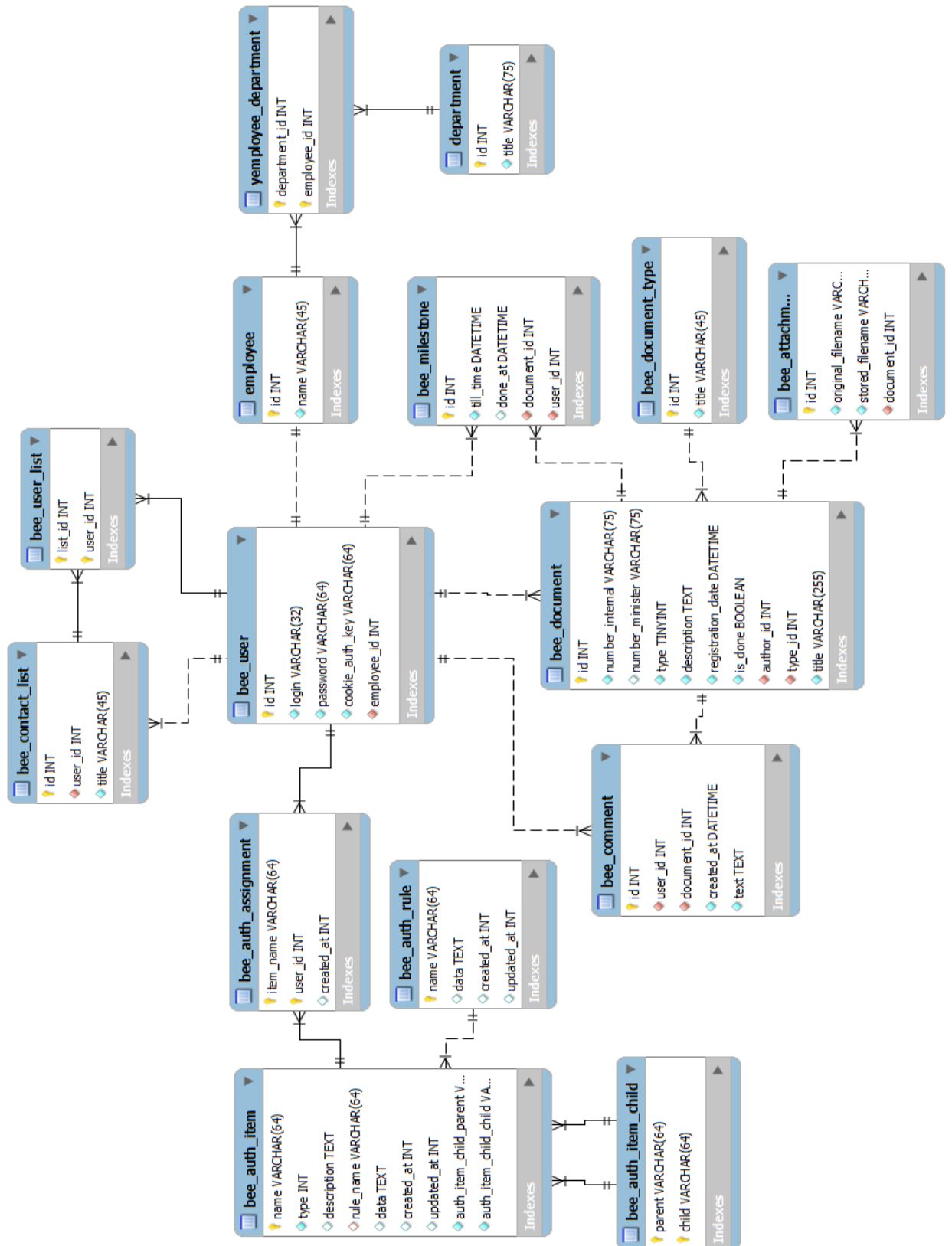


Рис. 3 Структура бази даних

Таблиця **document** є однією з базових таблиць. Вона зберігає інформацію про всі документи, що зареєстровані в системі документообігу. Ця таб-

лиця містить в собі спільні атрибути всіх типів документів, зберігає інформацію про автора та дату реєстрації документу.

Таблиця **attachment** містить інформацію про файли, що були додані до документів. Вона надає можливість додавання одразу багатьох файлів до одного документа та можливість заборони прямого доступу до файлів шляхом відвантаження файлів за допомогою PHP скрипта замість прямої передачі за допомогою веб-сервера.

Таблиця **document_type** містить інформацію про наявні в системі типи документів.

Таблиця **milestone** дозволяє посилати документи на виконання користувачам системи. Така структура дозволяє додавати документам декілька дат виконання та декілька відповідальних осіб. За допомогою збереження дат виконання завдань з'являється можливість контролю за виконанням документів.

Таблиця **comment** містить коментарі користувачів СЕД стосовно тих чи інших документів. За рахунок цієї таблиці реалізується можливість зворотнього зв'язку з автором документа.

Таблиці **contact_list** та **user_list** надають можливість створювати списки масової розсилки документів з метою спрощення цього процесу в майбутньому.

Архітектура веб-сервера базується на основі використання PHP фреймворка Yii2. Виходячи з цього, контролери, моделі, представлення, хелпери, компоненти та інші елементи системи реалізуються у вигляді класів. Більшість цих класів є потомками класів фреймворку та розташовуються у задалегідь відведених для них місцях. У роботі використовується менеджер пакетів Composer, що дозволяє при написанні коду використовувати готові класи для вирішення поставлених задач. Взаємодія з базою даних відбувається за рахунок використання підходу Active Record, що реалізується на рівні моделей.

Сервер веб-сокетів поєднує в собі веб-сервер для отримання команд від основного серверного застосунка та безпосередньо сервер веб-сокетів. Даний елемент реалізується на основі системи модулів Node.js: присутні головний модуль, модуль сервера веб-сокетів, модуль веб-сервера, модулі моделей клієнта та токена, модуль-репозиторій клієнтів та токенів. Веб-сервер реалізується на основі фреймворку Express. Сервер сервер веб-сокетів реалізований з використанням бібліотеки socket.io.

Клієнтський застосунок використовує процедурний, подієво-орієнтований та компонентно-орієнтований підходи до розробки програм. Основними бібліотеками є jQuery, Twitter Bootstrap 3 та socket.io. Для покращення взаємодії з користувачем використовуються різні HTML5 елементи (наприклад, Visibility API, File API, Notification API, тощо).

Для системи електронного документообігу розроблено проект інтерфейсу, що складається з кількох типових екранів.

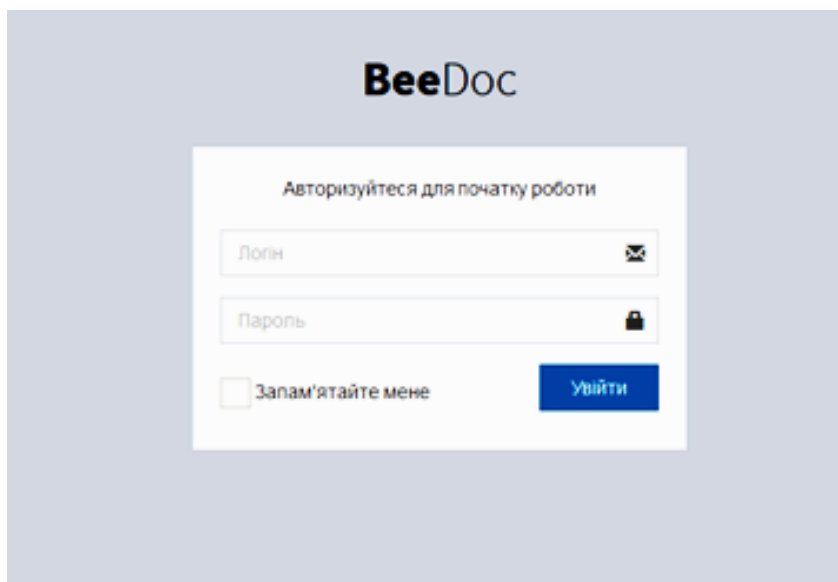


Рис. 4 Екран авторизації

Перший екран, який бачить користувач одразу після завантаження сайту – це екран авторизації (рисунок 4). Користувач має ввести свої логін і пароль для того, щоб система знала хто з нею працює. Екран авторизації містить два текстові поля для введення логіна і пароля. Всі символи пароля маскуються крапками. Нижче знаходиться прапорець, встановивши який користувач може попросити систему запам'ятати його на цьому комп'ютері для того, щоб не вводити логін і пароль кожного разу. В нижньому правому кутку форми авторизації розміщено кнопку входу на сайт.

Після успішної авторизації користувач потрапляє в систему електронного документообігу. Інтерфейс системи побудований з 4 основних блоків: шапка сайту, меню сайту (зліва), блок контенту (посередині) та футер.

Після успішної авторизації користувач потрапляє на головну сторінку. Тут виводяться списки останніх створених документів, останніх отриманих документів та прострочених документів. При необхідності користувач може одразу перейти до перегляду будь-якого документа, що приведений у цих списках скориставшись гіперпосиланням.

За допомогою меню користувач може створити новий документ.

Рис. 5 Перша вкладка екрана реєстрації нового документа

З метою запобігання необхідності постійного прокручування сторінки даний екран розділений на 3 вкладки. Кожна з вкладок містить лише ті поля, що відповідають її тематиці. Таким чином, перша вкладка (рисунок 5) містить набір полів, що відповідають атрибутам документа.

Друга вкладка містить інтерфейс для завантаження файлів (рисунок 6). На цій вкладці відображається список файлів, що прикріплені до документа. Користувач має можливість додати нові файли натиснувши на кнопку Додати файл. В результаті в список буде доданий новий елемент з інформацією про обраний файл. У випадку необхідності доданий файл можна видалити.

Третя вкладка містить інтерфейс для вибору користувачів, що повинні отримати та виконати даний документ (рисунок 7). Користувач може додавати та видаляти етапи виконання документа. Для кожного етапу виконання необхідно вказати його дату завершення та список користувачів, що повинні його виконати до вказаного терміну. Існує можливість вибору дати за допомогою календаря, можна додавати та видаляти користувачів, змінювати статус виконання документа користувачем.

Після внесення даних до картки документа користувач повинен натиснути на кнопку Надіслати, яка відправить введені дані на сервер для подальшої обробки. Після успішного збереження користувач переходить на сторінку зі списком створених документів. У випадку виникнення помилки, користувач побачить повідомлення про помилку і перехід на нову сторінку не відбудеться.

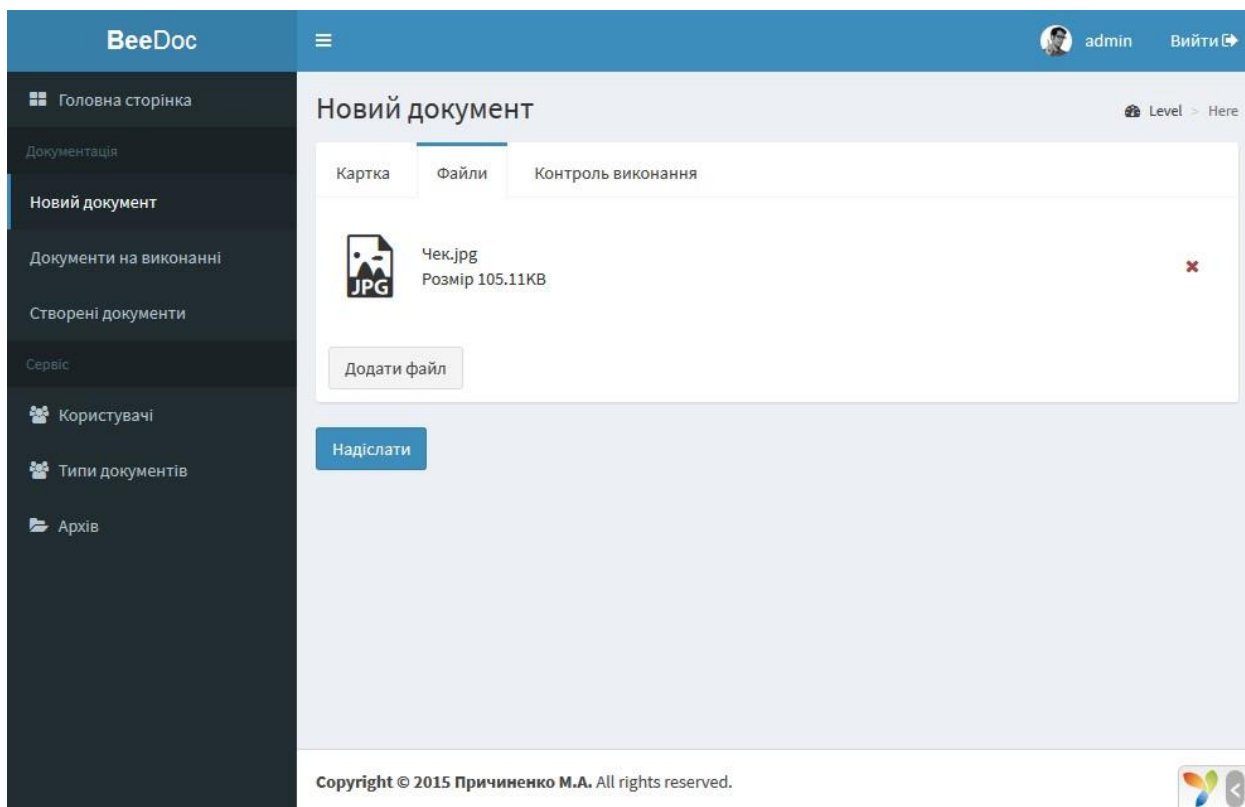


Рис. 6 Друга вкладка екрана реєстрації нового документа

На сторінці зі списком створених документів є можливість перейти до редагування обраного документа. Зовнішній вигляд цієї сторінки копіює сторінку створення нового документа, але знизу додається блок з коментарями до цього документа.

При переході на сторінку Документи на виконанні користувач побачить список всіх документів, що надійшли йому на виконання. Візуально ця сторінка дуже схожа на сторінку зі списком створених документів. З цієї сторінки можна перейти на сторінку перегляду обраного документа.

Сторінка перегляду документа виглядає схожою на сторінку редагування документа, але користувач не може змінювати атрибути документа. Крім того, користувач може змінити тільки свій статус виконання. Функція коментування доступна на цій сторінці.

Пункти меню Користувачі, Типи документів та Архів призначені тільки для адміністраторів та містять списки відповідних типів елементів у системі та можливість їх створення, перегляду, зміни та видалення.

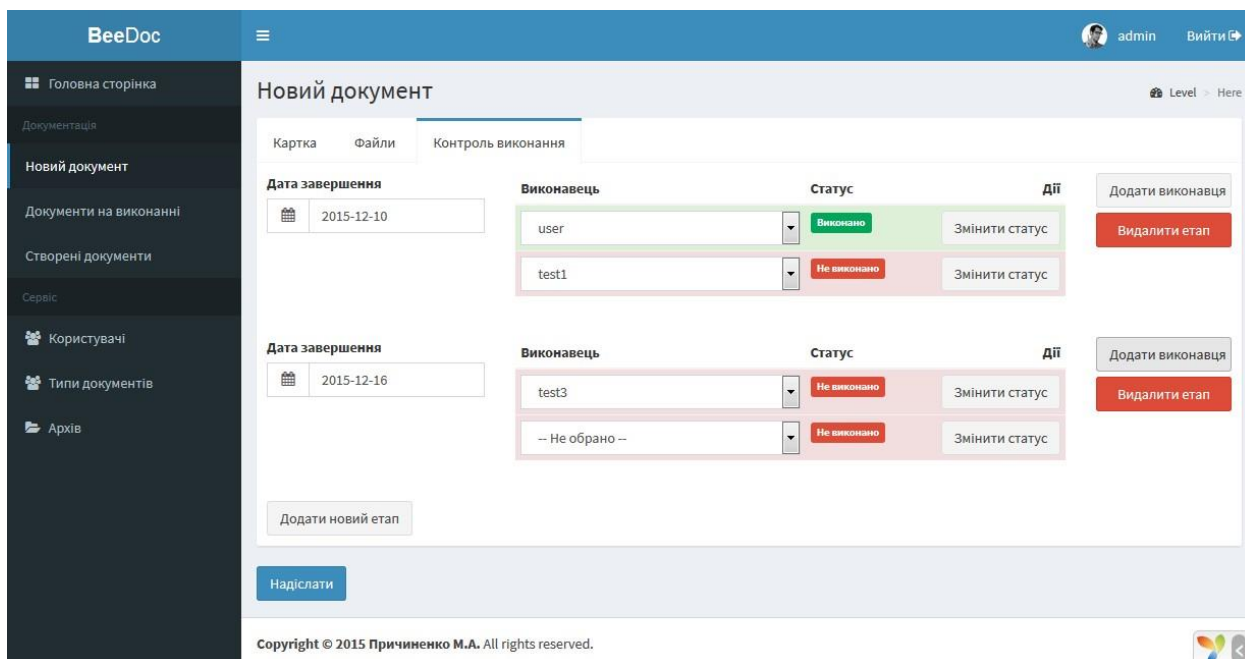


Рис. 7 Третя вкладка екрана реєстрації нового документа

При натисканні на кнопку Вийти користувач виходить із системи та потрапляє на сторінку авторизації. Для продовження роботи з системою він повинен знову увійти в систему.

Четвертий розділ містить оцінку відповідності створеної системи вису- нутим Запорізькою державною академією вимогам до системи. Аналізується відповідність отриманого рішення основним характеристикам СЕД.

Створена система електронного документообігу реалізує функції реєст- рації та редагування документів, додавання файлів до реєстраційних карток документів, забезпечує можливість здійснення контролю виконання докуме- нтів, забезпечує можливість пошуку документів за значенням зв'язаних з ни- ми атрибутів, дозволяє коментувати документи та переміщувати їх в архів, забезпечує контроль доступу до ресурсів системи. Система має можливість керування словником типів документів та список користувачів. Також реалі- зована функція оповіщення користувачів про виникнення подій у системі в режимі реального часу.

Архітектура системи розроблена з урахуванням можливості подальшої її інтеграції до складу АСУ вищого навчального закладу. Обрані технології за- безпечують відкритість системи, а відсутність зайвих функцій значно полег- шує подальший розвиток та модифікацію системи в процесі зміни вимог. Та- ким чином, можна зробити висновок, що розроблена система електронного документообігу повністю задовольняє висунуті Запорізькою державною ін- женерною академією вимоги.

Також, не зайвим буде оцінити ступінь відповідності розробленого про- дукту списку найважливіших характеристик систем електронного докумен- тообігу.

Для роботи системи на сервері необхідна операційна система, для якої доступні СУБД MySQL, платформа Node.js та веб-сервер з можливістю виконання PHP коду. До таких систем відносяться щонайменше Windows, Unix-подібні системи та OS X. Це найбільш популярні операційні системи, частина з яких є безкоштовною. Тому з великою вірогідністю сервер можна буде налаштувати для запуску на ньому даного продукту. Клієнтський веб-застосунок може працювати на будь-якій системі, в якій доступні браузері з підтримкою HTML 5 та JavaScript.

Оскільки система підтримує можливість прикріплення файлів до реєстраційної картки документа, то вона може працювати з будь-якими форматами даних. Але ця робота обмежується операціями збереження та повернення файлу, без його додаткової обробки.

На поточний момент система підтримує можливість вертикального масштабування. Можливе також горизонтальне масштабування, але це потребує проведення додаткових робіт. Зокрема, можуть знадобитися додаткові сервери для розподілу навантаження та налаштування серверів баз даних для коректної синхронізації інформації.

Максимальна кількість користувачів у системі залежить від потужності серверної машини. Оскільки в основі системи лежить веб-сервер, що працює за принципом запит-відповідь, то одночасна робота декількох клієнтів не викликає великого додаткового навантаження на сервер — ресурси використовуються лише на етапі запиту інформації на сервері. Іншим джерелом навантаження може слугувати сервер веб-сокетів, який підтримує постійне з'єднання з клієнтським застосунком. Але більшу частину часу з'єднання знаходяться в пасивному стані та використовують відносно малу кількість ресурсів. Додатковим фактором, який може впливати на максимальну кількість користувачів у системі є фізичне розміщення серверів, оскільки при розміщенні кількох серверів на одному комп'ютері вони ділять між собою доступні ресурси.

Дана система електронного документообігу заснована на принципі вільної схеми розповсюдження документів. Користувач в будь-який момент може додати нові етапи та нових виконавців для документа.

Із засобів для визначення маршрутних схем проходження документів доступні лише можливості маніпулювання етапами та користувачами, що виконують документи. Для полегшення процесу розсилки документів за “популярними” маршрутами існує можливість об'єднувати користувачів у списки користувачів. В цьому випадку відправка документа на виконання списку користувачів призведе до відправлення його кожному з членів даної групи.

В системі присутня можливість контролю за виконанням документа, але відсутні засоби для контролю правильного порядку виконання документа користувачами. Вирішенням цього питання займається автор документу на етапі особистого контролю виконання документа. В СЕД присутня можливість повідомлення користувачів про надходження на виконання нових документів та про зміни в раніше надісланих документах.

Версійність документів повинна підтримуватися користувачами самос-

тійно. Серед доступних засобів для роботи з версійністю є створення кількох копій документа та прикріплення файлів.

Клієнтська частина системи електронного документообігу не має можливості інтеграції з іншими застосунками. В даному випадку необхідності в такій взаємодії немає. Серверна частина системи може вільно інтегруватися з іншими додатками, якщо вони підтримують такі можливості. Серверна частина може інтегруватися з іншими підсистемами АСУ.

Необхідності в налаштуванні даної СЕД на потреби конкретного замовника немає, оскільки вона спеціально проектувалася з врахуванням цих потреб. Налаштування може зводитися до зміни певних “косметичних” елементів та фізичного розташування серверів застосунка.

Продукт має вбудовані засоби для регламентації доступу до ресурсів на базі веб-сервера. Реалізується RBAC принцип перевірки прав доступу. Криптографічний захист даних відсутній у зв'язку з відсутністю таких вимог до цієї системи. При цьому, у випадку необхідності, є можливість додати додатковий рівень захисту.

В цілому у розробленій системі електронного документообігу ВНЗ присутні більшість найважливіших характеристик СЕД. Відсутні характеристики не були вказані у вимогах до продукту і частково можуть бути втілені у життя при необхідності. Таким чином, можна зробити висновок, що, в залежності від вимог конкретної організації, цілком можливо створити СЕД на основі сучасних веб-технологій.

Висновки

1. У сучасному світі ефективність управління значною мірою залежить від часу, що витрачається на роботу з документами. Оптимізація цих процесів може значно скоротити час та ресурси організації.
2. Типова СЕД забезпечує повну підтримку процесів створення, управління, обробки і поширення електронних документів між відповідальними посадовими особами та контроль за потоками документів.
3. На поточний момент існує багато СЕД. Здебільшого вони розповсюджуються на платній основі та представляють собою комплексні рішення закритого типу.
4. Обгрунтовано ряд причин для створення власної СЕД для ВНЗ.
5. В цілому, всі розглянуті системи реалізовані з використанням клієнт-серверної архітектури. Функціональність системи реалізується за рахунок модулів, які можна комбінувати для досягнення бажаного набору функцій.
6. Розроблена система відповідає висунутим вимогам та реалізує таку функціональність: реєстрація документів, розсилка документів на виконання, контроль виконання документів, пошук документів.
7. Для створення СЕД, в залежності від вимог конкретної організації, у якості базових можна використовувати сучасні веб-технології.

ОПУБЛІКОВАНІ ПРАЦІ

Причиненко М.А., магістрант, Шамровський О.Д., проф., д.т.н. — науковий керівник. «Порівняльне дослідження та створення системи документообігу у ВНЗ.» Матеріали XX науково-технічної конференції студентів, магістрантів, аспірантів і викладачів ЗДІА «ЕЛЕКТРОНІКА, АВТОМАТИЗОВАНІ СИСТЕМИ ТА СУЧАСНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ» том III, Запоріжжя, ЗДІА, 2015.

АНОТАЦІЯ

Причиненко М.А. Порівняльне дослідження методів створення систем документообігу ВНЗ – Рукопис.

Мета роботи: аналіз існуючих рішень та розробка системи електронного документообігу для вищого навчального закладу.

Основні результати роботи: в процесі виконання дипломної роботи було досліджено предметну область та зроблено огляд трьох існуючих систем електронного документообігу. За результатами огляду проаналізовано підходи, що використовуються при створенні СЕД. За результатами аналізу розроблено проект та реалізовано систему електронного документообігу на основі сучасних веб-технологій. Проаналізовано отримані результати та зроблено висновки стосовно можливості використання сучасних веб-технологій у якості основи для створення СЕД.

Ключові слова: СЕД, документообіг, документ, контроль виконання, система, підхід, клієнт-серверна архітектура, веб-технології, аналіз, створення.

АНОТАЦІЯ

Причиненко М.А. Сравнительное исследование методов создания систем документооборота ВУЗ – Рукопись.

Цель работы: анализ существующих решений и разработка системы электронного документооборота для высшего учебного заведения.

Основные результаты работы: в процессе выполнения дипломной работы было исследовано предметную область и сделано обзор трёх существующих систем электронного документооборота. По результатам обзора проанализировано подходы, которые используются при создании СЭД. По результатам анализа разработан проект и реализована система электронного документооборота на основе современных веб-технологий. Проанализировано полученные результаты и сделаны выводы относительно возможности использования современных веб-технологий в качестве основы для создания СЭД.

Ключевые слова: СЭД, документооборот, документ, контроль выполнения, система, подход, клиент-серверная архитектура, веб-технологии, анализ, создание.

SUMMARY

Maksym Prychynenko. Comparative research of methods for creating document management systems for university – Manuscript.

Purpose of work: analysis of existing solutions and creation of document management system for university.

Basic job performances: during execution of the thesis a subject area has been researched. An overview of three existing electronic document management systems is done. Approaches that are used for creating document management systems are analyzed based on results of the review. According to the analysis an electronic document management system is drafted and implemented using modern web technologies. The results are analyzed and a conclusion about the possibility of usage modern web technologies as the basis for creation of document management systems is made.

Key words: DMS, document flow, document, accomplishment control, system, approach, client-server architecture, web technologies. Analysis, creation.