

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	7
ВСТУП	10
1 ОХОРОНА ПРАЦІ	12
1.1 Значення охорони праці для забезпечення безпечних і здорових умов праці	12
1.2 Аналіз потенційних небезпек та шкідливих факторів виробничого середовища	13
1.3 Забезпечення нормальних умов праці при роботі з ЕОМ	14
1.4 Забезпечення безпеки монтажу, пусконаладжувальних, ремонтних робіт та експлуатації ЕОМ і комп'ютерних мереж	16
1.5 Пожежна безпека та безпека в надзвичайних ситуаціях	18
1.6 Висновки	22
ВИСНОВКИ	23
СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА	24
ДОДАТКИ	26
БІБЛІОГРАФІЧНА ДОВІДКА	54

					ДП.ПЗ - 04.00.00.000 ПЗ			
Зм.	Арк.	№ докум.	Підпис	Дата				
Розробив	Бойчук Я.В.				Розробка програмного алгоритмічного забезпечення багатофункціональної корпоративної системи для сумісної роботи, управління документами і проектами ПОЯСНЮВАЛЬНА ЗАПИСКА	Літ.	Аркуш	Аркушів
Перевірив	Випасняк Л.І.						6	57
Н.контр	Вовк Р.Б.					ІНФТУНГ ПЗ-07-1		
Затв.	Юрчишин В.М.							

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ПЗ – програмне забезпечення
БД – база даних
Java EE – Java Enterprise Edition
Java ME – Java Mobile Edition
Java SE – Java Second Edition
J2EE – Java 2 Enterprise Edition
J2SE – Java 2 Standard Edition
J2ME – Java 2 Mobile Edition
JVM – Java Virtual Machine
JIT – Just In Time
AOT – Ahead of Time
JCP – Java Community Process
SDK – Software development kit
ORM – Object-relational mapping
SQL – Structured Query Language
UI – User interface
GNU – GNU's Not Unix (акронім)
GPL – GNU General Public License
MIT – Massachusetts Institute of Technology
DOM – Document Object Model
HTML – HyperText Markup Language
CSS – Cascading Style Sheets
XML – eXtensible Markup Language
XPath – XML Path Language
CDN – Content delivery network
AJAX – Asynchronous JavaScript and XML
ASF – Apache Software Foundation
СКБД – Система керування базою даних

					ДП.ПЗ - 04.00.00.000 ПЗ	Аркуш
						7
Зм.	Арк.	№ докум.	Підпис	Дата		

API – Application programming interface
 JPA – Java Persistence API
 GUI – Graphical User Interface
 UML – Unified Modelling Language
 MCC(MVCC) – Multiversion concurrency control
 EIP – Enterprise Information Portal
 IBM – International Business Machines
 JSR – Java Specification Request
 WSRP – Web Services for Remote Portlets
 CMS – Content Management System
 ECM – Enterprise Content Management
 WCM – Web Content Management
 DAM – Digital Asset Management
 DMS – Document Management System
 ODMA – Open Document Management API
 LDAP – Lightweight Directory Access Protocol
 WebDAV – Web Distributed Authoring and Versioning
 SOAP – Simple Object Access Protocol
 CSCW – Computer Supported Cooperative Work
 HTTP – Hypertext Transfer Protocol
 HTTPS – Hypertext Transfer Protocol Secure
 VPN – Virtual Private Network
 CGI – Common Gateway Interface
 TCP/IP – Transmission Control Protocol/Internet Protocol
 SMTP – Simple Mail Transfer Protocol
 FTP – File Transfer Protocol
 MVC – Model-View-Controller
 РСКБД – Реляційна система керування базами даних
 ACID – Atomicity, Consistency, Isolation, Durability

					ДП.ПЗ - 04.00.00.000 ПЗ	Архив
						8
Зм.	Арк.	№ докум.	Підпис	Дата		

EAV – Entity Attribute Value Model

UI – User Interface

JDBC – Java Database Connectivity

JMX – Java Management Extensions

JMS – Java Message Service

RPC – Remote procedure call

RMI – Remote Method Invocation

CORBA – Common Object Request Broker Architecture

UML – Unified Modeling Language

SHA – Secure Hash Algorithm

JSP – JavaServer Pages

WYSIWYG – What You See Is What You Get

					ДП.ПЗ - 04.00.00.000 ПЗ	Архив
						9
Зм.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

В нас час важливою економічною точкою опори для будь-якої комерційної організації є наявність певних факторів, які визнають чітку позицію компанії на ринку. До всіх цих чинників можна віднести багато варіантів, зокрема:

- капітал підприємства;
- матеріальна база;
- технічна база;
- кваліфікований персонал;
- місце підприємства на внутрішньому і зовнішньому ринку;
- наявність сучасних засобів виробництва та ведення бізнесу;
- та інші.

Тому кожний розділ ведення бізнесу повинний бути детально розглянутий та впроваджений у життя. Але якщо подивитися із точки зору програмного забезпечення, то на даному етапі розвитку цивілізації, якісне ПЗ відіграє напевно найбільш важливу роль. Адже не можливо зараз утримувати всі дані в паперовому вигляді, не можливо відсилати друковані листи, чи спілкуватися тільки по телефоні і взнавати новини компанії тільки при зустрічі. У наш стрімкий час розвитку, новини міняються із колосальною швидкістю, тому встигнути за всім просто не можливо без певного програмного продукту.

Уявіть собі інформатор, який сповіщає будь-які для Вас новини чи корисну інформацію в зручний для Вас час, при цьому вміє фільтрувати і аналізувати дані із попередніх запитів. Також на даний момент важко уявити не можливість спільної роботи над документами, над електронними таблицями. Дані технології вже давно використовуються людьми і підприємствами, починаючи від найменших де працює двоє людей, до величезних із кількістю працівників більше ста тисяч. Але для цього всього використовуються дуже багато технологій, які важко налаштувати і потребують великих витрат на підтримку. Тому було розроблено багато сервісів і додатків, які полегшують роботу в мережі для підприємств.

					ДП.ПЗ - 04.00.00.000 ПЗ	Аркуш
						10
Зм.	Арк.	№ докум.	Підпис	Дата		

Дане ПЗ використовується у всіх нішах нашого життя, починаючи від шкіл і лікарень, закінчуючи величезними корпораціями з будівництва космічних кораблів. Тому розробити універсальний продукт, який забезпечить всі вимоги, просто не можливо. Для кожної сфери існують свої нюанси.

Цікавою нішею для дослідження стало корпоративне програмне забезпечення для малого і середнього бізнесу. На даний момент існує багато програмних продуктів для комерційних цілей, проте вони здебільшого розраховані на великі корпорації і підприємства. Тому використання їх для менших фірм просто не доцільно, або дуже складно із фінансової сторони (витрати на підтримку передують вигоді). Як відомо, на ринку до цих пір зберігається тенденція на попит на корпоративне програмне забезпечення, яке б відповідало вимогам малого і середнього бізнесу, і в той же час було практично придатним для використання у великих корпоративних цілях

					ДП.ПЗ - 04.00.00.000 ПЗ	Аркуш
						11
Зм.	Арк.	№ докум.	Підпис	Дата		

1 ОХОРОНА ПРАЦІ

1.1 Значення охорони праці для забезпечення безпечних і здорових умов праці

Значення охорони праці в будь-якій галузі України є дуже вагоме, адже саме охорона праці напряду пов'язана з вивченням та вирішенням питань безпеки праці на виробництві, попередженні виробничого травматизму і професійних захворювань, пожеж та вибухів, а також охорони навколишнього середовища.

При проведенні робіт нерідко порушуються діючі правила й інструкції з техніки безпеки. Це відбувається по причині незадовільного інструктажу й навчання робітників, внаслідок неправильної організації робіт, недостатнього технічного нагляду зі сторони інженерно-технічних працівників.

Охорона праці та навколишнього середовища здійснюється на основі правових норм України і розглядає основи наукової організації праці робітників системи буріння нафтових і газових свердловин, питання виробничої санітарії, основи електробезпеки і техніки безпеки при монтажі і експлуатації бурового обладнання, моніторингу за роботою електростанцій на бурових, збором інформації з вимірювальних приладів бурових, що є важливим.

Для того щоб максимально знизити травматизм, необхідна висока кваліфікація робітників, знання технологічних особливостей процесу буріння свердловин, призначення, конструкції та правила експлуатації устаткування й механізмів, правильних й безпечних прийомів виконання робіт, а також високий рівень технічного нагляду зі сторони керівників робіт.

Покращення організації праці, механізації тяжких й трудомістких робіт, раціоналізація технологічних процесів, впровадження нових, більш сучасних видів устаткування, механізмів та інструменту — основний напрям підвищення продуктивності праці й створення здорової і безпечної виробничої обстановки на бурових підприємствах.

					ДП.ПЗ - 04.00.00.000 ПЗ	Аркуш
						12
Зм.	Арк.	№ докум.	Підпис	Дата		

1.2 Аналіз потенційних небезпек та шкідливих факторів виробничого середовища

Важливою умовою функціонування будь-якого сучасного підприємства є робота із ЕОМ(електронно-обчислювальними машинами). Проте ця робота супроводжується впливом багатьох чиннів на організм оператора по роботі з ЕОМ. Було проведено аналіз і зроблено характеристику несприятливих виробничих факторів, які здатні впливати на здоров'я та самопочуття працівника. Всі дані наведено в таблиці 1.1.

Таблиця 1.1 – Аналіз потенційних небезпек виробничих факторів при роботі з ЕОМ

Джерело небезпек	Характеристика потенційно-небезпечних виробничих факторів та їх допустимі значення
ренгенівське випромінювання	Фактичні (середні) дані вимірів 9-12мкР/год (в діапазоні 1.2КеВ). Гранично допустима експозиційна доза: 100мкР/год.
ультрафіолетове випромінювання	Фактичні дані вимірів: 0.001 Вт/м ² в діапазоні 280-315 нм - УФ-Ф). Допустима інтенсивність: 0.01 Вт ² – УФ-В
ІЧ-випромінювання	Фактичні дані вимірів інтенсивності теплового випромінювання 0,05-4 Вт/м ² (в діапазоні 700 нм-1мм). Допустима інтенсивність: 35-70 Вт/м ² .
видимий діапазон	Фактичні дані: 0,1-2 Вт/м ² (в діапазоні 320-400 нм) і 2,5-4 Вт/м ² (в діапазоні 400-700 нм). Допустима інтенсивність потоку енергії: 10 Вт/м ² .
яскравість	Фактичні дані: 306 кд/м ² . Допустиме значення: 35 кд/м ² .
електростатичне поле	Фактичні дані: 15 кВ/м (0 Гц) Допустима напруженість поля 20-60 кВ/м.
шум	Діюче значення звукового тиску: 28.6-44 дБА. Допустиме значення: 55дБА

Робота оператора з ЕОМ вимагає максимальної концентрації протягом цілого робочого дня, що в свою чергу призводить до значних навантажень на організм. Також через постійне розумове навантаження зростає ризик перевантаження аналізаторів та ймовірність психічного розладу.

1.3 Забезпечення нормальних умов праці при роботі з ЕОМ

Для забезпечення нормальних умов праці оператора з ЕОМ потрібно створити максимально комфортні умови праці. Для цього було обрано робочий кабінет, який відповідає всім вимогам та стандартам праці.

В робочому кабінеті будуть знаходитися 11 робочих місць (рисунок 1.1). На кожного працівника розраховано одна одиниця техніки (сюди входить монітор, системний блок, клавіатура, мишка). Проте деякі працівники, у зв'язку зі специфічним видом роботи, потребуються два монітори, які розташовані поряд один з один. Кожний працівник відгороджений від інших дерев'яною перегородкою. Кожний працівник забезпечений робочим місцем загальною площею 8м².

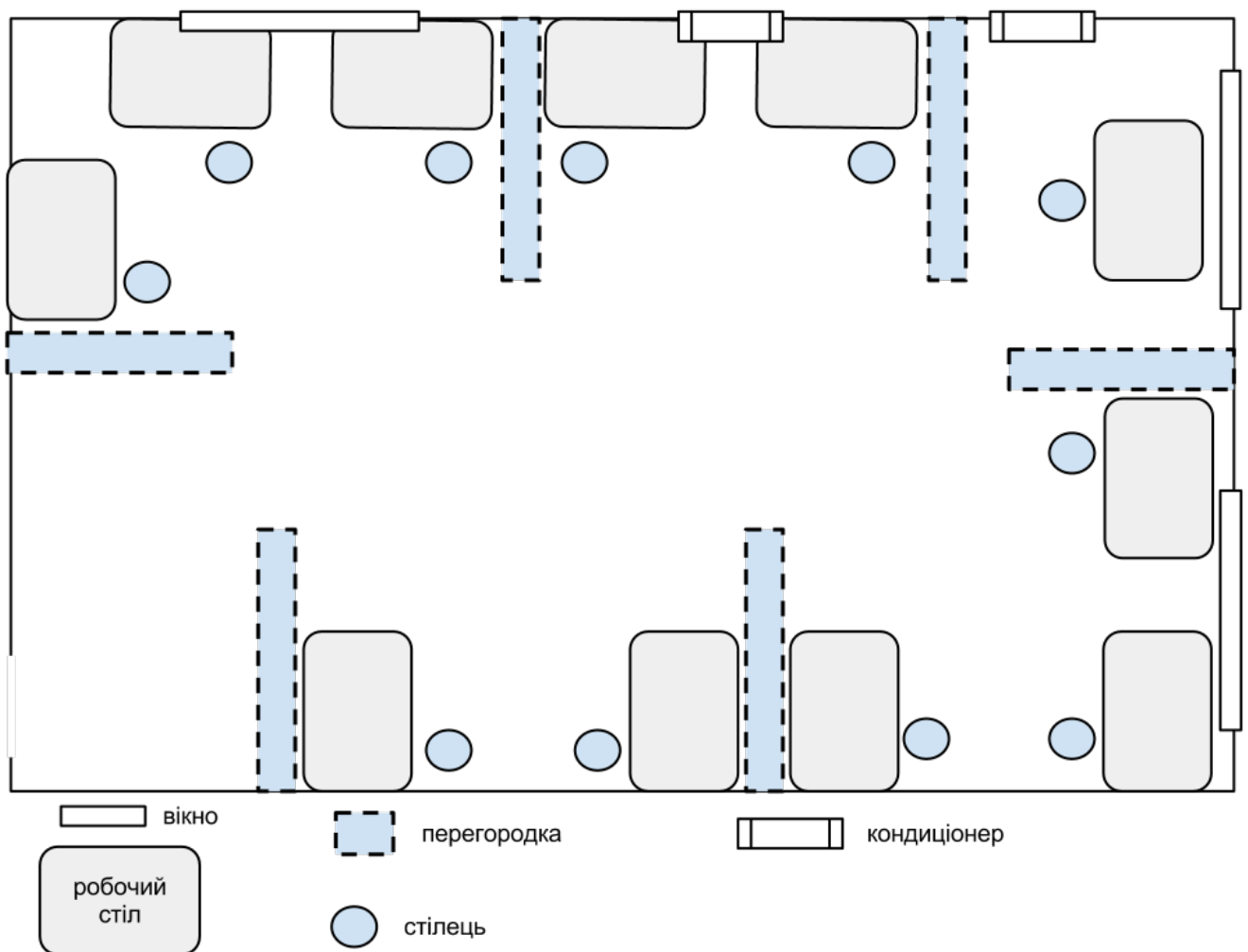


Рисунок 1.1 – Схема робочої кімнати - офісу

Також для кожного працівника, для того щоб забезпечити комфортні умови

					ДП.ПЗ - 04.00.00.000 ПЗ	Аркуш
Зм.	Арк.	№ докум.	Підпис	Дата		14

праці розраховано три висувні ящики та тумбочку для власних речей, та шафки зверху робочого місця, для швидкого доступу до документації та канцелярських речей, що в свою чергу становить загальний об'ємом 22м³. Кожний працівник забезпечується зручним та комфортним кріслом, яке регулюється згідно вимог. Все це зроблено із урахуванням ГОСТ 12.2.032-78[18], ДНАОП 0.00-1.31-99[19] та ДСан ПіН 3.3.2.007-98[20].

На кожному вікні є жалюзі, що дає змогу зменшити вплив сонячних променів на працюючого (засліплення монітору для прикладу).

Комп'ютерна-та електромережа прокладена із врахуванням всіх вимог техніки безпеки.

Крім обчислювальних одиниць, більше технічних засобів в кімнаті не передбачається, адже всі периферійні пристрої розташовані на коридорі.

Для забезпечення мікрокліматичних умов праці передбачено два кондиціонери, які можуть працювати в автоматичному режимі і підтримувати сталу температуру приміщення. Також налагоджено роботу постійної вентиляція, яка працює за наступним принципом: вентиляційні витяжки розміщені в 4 місцях (по кутах) кімнати. Дві із них працюють як витяжки, тобто втягують повітря, інші дві забезпечують свіжим повітрям кімнату – в кінцевому результаті постійна циркуляція повітря. Вентиляційні труби виведені на вулицю. В таблиці 1.2 наведено оптимальні значення метеорологічних умов згідно ГОСТ 12.1.005-88 [17] в робочому кабінеті.

Таблиця 1.2 – Оптимальні значення метеорологічних умов в робочому кабінеті для легкої категорії робіт в офісному приміщенні

Категорія робіт	Період року	Температура, °С	Відносна вологість, %	Швидкість руху повітря, м/с
Легка Іа	Холодний	22-24	40-60	0,1
Легка Іа	Теплий	23-25	40-60	0,1

Як згадувалося вище про вентиляцію – саме вентиляція є основою забез-

печення комфортних умов праці, тобто регуляцію мікроклімату. Характеристики штучної вентиляції наведено в таблиці 1.3

Таблиця 1.3 – Характеристика штучної вентиляції

Приміщення	Тип вентиляції	Вентиляційне обладнання	Кратність повітряного обміну, 1/год
Офіс	Механічна	Кондиціонер (700-1000 м ³ /год, 4 кВт), 2шт	4.4

Так як робота людини з ЕОМ більше як на 90% складається із зорової роботи – тому правильне і раціональне освітлення становить основу для створення сприятливих умов праці та унеможливлення розвитку професійних захворювань. Освітлення повинне забезпечувати комфортну роботу працівника в будь-яку пору дня, чи то ранок, чи обід чи вечір. Характеристика штучної освітленості робочих місць наводиться у таблиці 1.4 згідно ДБН В.2.5-28-2006 [14].

Таблиця 1.4 – Оптимальні значення метеорологічних умов в робочому кабінеті для легкої категорії робіт

Приміщення	Розряд зорової роботи	Загальне освітлення, лК	Комбіноване освітлення, лК	Аварійне освітлення для евакуації, лК
Офіс	IVв	200	400	0.5

Засоби індивідуального захисту не передбачаються, так як монітори побудовані на основі рідких кристалів, тому тут немає регенерації картини, потім завжди сталий.

1.4 Забезпечення безпеки монтажу, пусконаладжувальних, ремонтних робіт та експлуатації ЕОМ і комп'ютерних мереж

Так як ЕОМ завжди підключені до мережі і створюють значне навантаження неї, правильне проектування мережі визначає безпеку роботи як працівників так і самого підприємства.

					ДП.ПЗ - 04.00.00.000 ПЗ	Архив
						16
Зм.	Арк.	№ докум.	Підпис	Дата		

Для забезпечення захисту людей від ураження електричним струмом використовуються окремо або в поєднанні один з одним такі технічні способи та засоби як: захисне заземлення, занулення, вирівнювання потенціалів, мала напруга, захисне відімкнення, ізоляція провідників із струмом, огорожувальні пристрої, попереджувальна сигналізація, блокування, знаки безпеки, засоби захисту та запобіжні пристрої.

Для захисту від дотику до частин, що знаходяться під напругою, використовується ізоляція. Для захисту від дотику до частин, що знаходяться під напругою, використовується також подвійна ізоляція - електрична ізоляція, що складається з робочої та додаткової ізоляції.

Також передбачене використання звукової та світлової сигналізації, надписів, плакатів та інших засобів інформації, що попереджують про небезпеку.

Вимоги електричної і механічної безпеки для ЕОМ і систем обробки даних встановлені ГОСТ 25861 - 83 [15]. Додаткові або особливі заходи безпеки, яких необхідно дотримуватися при експлуатації і технічному обслуговуванні ЕОМ і їх пристроїв, вказані в ЕД (експлуатаційна документація).

Особи, що допускаються до експлуатації і технічного обслуговування ЕОМ, проходять цільове навчання по вивченню правил роботи і вимог безпеки при роботі з ЕОМ, а також експлуатаційну документацію на конкретні види ЕОМ, до роботи з якими вони одержують допуск. До експлуатації ЕОМ допускаються особи, що мають групу по електробезпеці не нижче II, до технічного обслуговування – групу III.

Для безпечної експлуатації ЕОМ в приміщенні, де вона встановлена, забезпечуються кліматичні умови, встановлені експлуатаційною документацією.

Всі пристрої ЕОМ підлягають захисному заземленню, за винятком переносних і переносних, в конструкціях яких заземлення не передбачено.

					ДП.ПЗ - 04.00.00.000 ПЗ	Аркуш
						17
Зм.	Арк.	№ докум.	Підпис	Дата		

1.5 Пожежна безпека та безпека в надзвичайних ситуаціях

Згідно з ПУЕ приміщення, де експлуатуються ЕОМ і ПЕОМ, належать до приміщень без підвищеної небезпеки ураження людини електричним струмом.

Вимоги електробезпеки і пожежної безпеки у приміщеннях, де встановлені ЕОМ і ПЕОМ, подані у НПАОП 0.00-1.28-10 [16]: ЕОМ і все устаткування для обслуговування, ремонту та налагодження їх роботи, електропроводи і кабелі мають відповідати вимогам електробезпеки зони за ПВЕ та мати апаратуру захисту від струму короткого замикання.

Забезпечено неможливість виникнення джерела загорання внаслідок короткого замикання та перевантаження проводів використанням негорючої ізоляції.

Всі приміщення обладнані системою автоматичної пожежної сигналізації з димовими пожежними сповіщувачами та двома переносними вуглекислотними вогнегасниками з зарядом вогнегасної речовини 5л. [21] (таблиця 1.5).

Всюди, включаючи коридори, передбачені засоби пожежної сигналізації на випадок НС.

Дані про первинні засоби пожежогасіння в офісі приводяться в таблиці 1.5

Таблиця 1.5 – первинні засоби пожежогасіння

Категорія	Захищена площа, м ²	Вуглекислотний вогнегасник	Хімічно-пінний вогнегасник	Порошковий вогнегасник
Д	90	2	-	-

З таблиці 1.5 категорія «Д» визначається як: «Негорючі речовини та матеріали в холодному стані».

Для запобігання впливу на людей небезпечних чинників пожежі повинна бути передбачена можливість швидкої евакуації людей з будівлі.

Необхідний час для евакуації для приміщень категорії «Д» не обмежується.

Розрахунковий час евакуації визначається по розрахунку часу руху одного

або декількох людських потоків через евакуаційні виходи з найбільш віддалених місць розміщення людей.

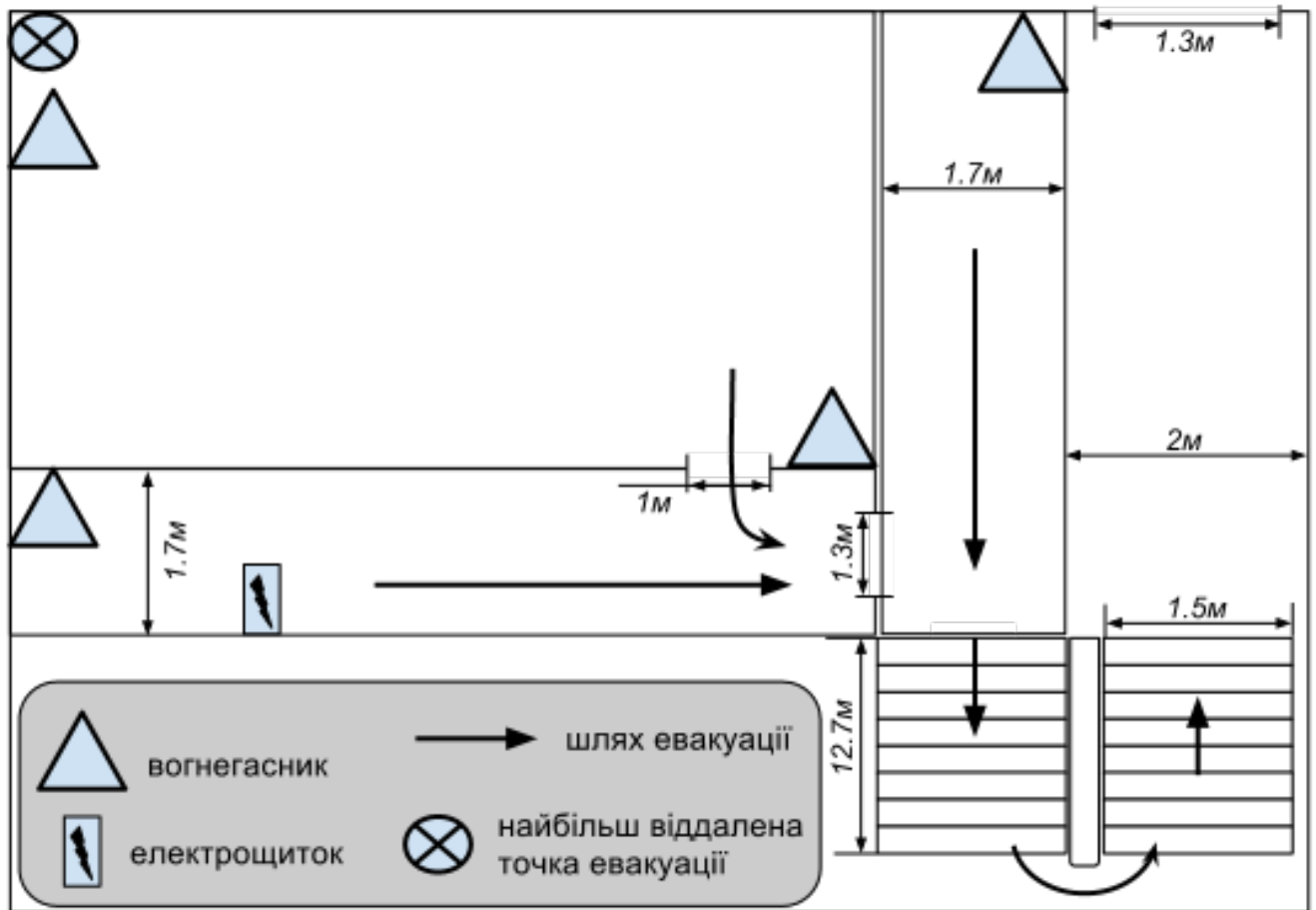


Рисунок 1.2 – Схема евакуації з офісу при пожежі

При розрахунку весь шлях руху людського потоку поділяється на ділянки.

Розрахунковий час евакуації людей потрібно визначити як суму часу руху людського потоку по окремих ділянках шляху:

$$T_p = T_1 + T_2 + \dots + T_i, \quad (1.1)$$

де T_1 – час руху людського потоку на першій ділянці;

T_1, T_1, \dots, T_i – час руху людського потоку на кожному з наступних ділянок шляху.

Швидкість руху людського потоку на ділянках шляху визначається в за-

лежності від щільності людського потоку D_n :

$$D_n = \frac{N_n \cdot f}{L_n \cdot S_n}, \quad (1.2)$$

де N_n – число людей в відділі;

f – середня площа горизонтальної проекції людини, m^2 (в зимовий час $f = 0.125 m^2$, в літній час $f = 0.1 m^2$), приймаємо $f = 0.125 m^2$;

L_n – довжина ділянки шляху, м;

S – ширина ділянки шляху, м.

Далі підраховуємо час руху людського потоку на ділянці шляху:

$$T_n = \frac{L_n}{V_n}. \quad (1.3)$$

Інтенсивність руху людського потоку по кожному з ділянок визначаються по формулі:

$$q_i = \frac{q_{i-1} \cdot S_{i-1}}{S_i}, \quad (1.4)$$

де S_i, S_{i-1} – ширина i -го що розглядається і попереднього йому ($i - 1$)-го ділянок шляху в метрах;

q_i, q_{i-1} – інтенсивність руху людського потоку по i -му, що розглядається попередньому йому ($i - 1$)-му ділянцям шляху в м/хв.

Тепер розрахуємо час евакуації для другого поверху приміщення з ЕОМ. Зведені початкові дані зображено в таблиці 1.7

Таблиця 1.6 – Умови розрахунку часу і шляхи евакуації з будівлі

№ ділянки	Ділянка руху людського потоку, m^2	Довжина, L , м	Ширина σ_i , м
1	Прохід між робочими місцями в приміщенні	7,2	1,0
2	Дверний отвір	1,0	1,0
3	Коридор (2 поверх)	6,5	1,7
4	Дверний отвір	1,0	1,3
5	Сходи (вниз)	12,7	1,5
6	Дверний отвір	1,0	1,3

№ ділянки	Ділянка руху людського потоку, м ²	Довжина, L, м	Ширина σ_i , м
7	Коридор (1 поверх)	9,8	1,7
8	Коридор-хол	4,0	2,0
9	Дверний отвір (евакуаційний вихід)	1,0	1,3

Максимальна відстань від найбільш видаленого місця у відділі до евакуаційного виходу становить 44,2 м, що відповідає встановленій нормі.

Визначаємо щільність людського потоку на першій ділянці D_1 :

$$D_1 = \frac{7 \cdot 0.125}{7.2 \cdot 1} = 0.12, \quad (1.5)$$

Отже $D_1 = 0.12$ люд./м²

По отриманому значенню D_n приймаємо швидкість руху людського потоку $V_n = 80$ (м/хв) і інтенсивність руху людського потоку $q_n = 8$ (м/хв).

Підраховуємо час руху людського потоку на першій ділянці:

$$T_1 = \frac{7.2}{80} = 0.09 \quad (1.6)$$

Отже час руху на першій ділянці $T_1 = 0.09$ хв.

Швидкість руху людського потоку на ділянках шляху, наступних після першого, приймаємо в залежності від інтенсивності руху людського потоку по кожному з ділянок шляху.

$$q_2 = \frac{8 \cdot 1}{1} = 8 \quad (1.7)$$

Отже швидкість руху на другій ділянці $q_2 = 8$ м/хв.

Беремо $V_2 = 80$ м/хв. Отже час руху по другій ділянці:

$$T_2 = \frac{1}{80} = 0.01 \quad (1.8)$$

Інші дані розрахунків виконуються аналогічно. Для наглядності ці дані по-

					ДП.ПЗ - 04.00.00.000 ПЗ	Аркуш
Зм.	Арк.	№ докум.	Підпис	Дата		21

дані в таблиці.

Таблиця 1.7 – Умови розрахунку часу і шляхи евакуації з будівлі

№ ділянки	Інтенсивність руху q_i , м/хв	Швидкість руху V_i , м/хв	Час руху T_i , хв
1	8	80	0,09
2	8	84	0,01
3	4,7	100	0,07
4	6,2	95	0,01
5	5	100	0,13
6	6,2	95	0,01
7	4,7	100	0,1
8	4	100	0,04
9	6,2	95	0,01

Розрахунковий час евакуації людей визначається як сума часу руху людського потоку по кожній ділянці шляху: $T_z = \sum T_i = 0.47$ хв.

ВИСНОВКИ

Завдяки сучасним технологіям і корпоративним стандартам, розвиток розробки комерційних продуктів виріс дуже стрімко. Зокрема сюди і відноситься відносно молодий напрямок – це розробка корпоративних порталів. Було встановлено стандарти щодо розробки додатків і аплікацій – це допомогло добитися легкої інтеграції і взаємодії.

Було також ро

В розділі охорона праці розглянуто основні небезпечні та шкідливі фактори, які мають або можуть мати місце в процесі роботи. Також значну увагу зосереджено на організаційно-технічних заходах, які в свою чергу направлені на усунення чи зменшення впливу шкідливих факторів. Визначено оптимальні метеорологічні умови, які будуть підтримуватися в приміщенні. Встановлено загальні правила, які необхідно виконувати при роботі з ЕОМ. Запропоновані заходи з охорони праці, направлені на профілактику травматизму та професійних захворювань, а також на покращення умов праці.

					ДП.ПЗ - 04.00.00.000 ПЗ	Аркуш
						23
Зм.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА

1. <http://www.jcp.org/en/jsr/detail?id=286> - стандарт портлетів Java Portlet 2.0
2. <http://www.jcp.org/en/jsr/detail?id=168> - стандарт портлетів Java Portlet 1.0
3. <http://google.com> - пошук доступної в інтернеті інформації
4. <http://tomcat.apache.org/tomcat-5.5-doc/> – специфікація серлетів
5. http://w3techs.com/technologies/overview/javascript_library/all – використання jQuery
6. <http://pz.nung.edu.ua/> - сайт кафедри ПЗАС
7. <http://www.intranetno.ru/> - бізнес рішення на базі SaaS, PaaS
8. http://en.wikipedia.org/wiki/Entity-attribute-value_model - EAV модель
9. <http://www.springsource.org/> - Spring framework
10. Макро-Економіка / під. ред. Базилевич В.Д. – К. : Знання 2007р. – 688с.
11. Загальна економіка / під. ред. Радіонова І. Ф. – К. : Абетка-НОВА 2007р. – 367с.
12. Вовк Р.Б., Шекета В.І. Методичні вказівки до дипломного проектування – Івано-Франківськ: ІФНТУНГ, 2010. - 44с.
13. Калічак О.В., Стеліга І.І., Кулик М.П, Воевідко І.В. – Охорона праці. Методичні вказівки для дипломного проектування. Івано - Франківськ. Факел.2008- 82 с.
14. ДБН В.2.5-28-2006 – природне і штучне освітлення
15. ГОСТ 25861-83 – машины вычислительные и системы обработки данных. Требования по электрической и механической безопасности и методы испытаний
16. НПАОП 0.00-1.28-10 – правила охорони праці під час експлуатації ЕОМ.

					ДП.ПЗ - 04.00.00.000 ПЗ	Аркуш
Зм.	Арк.	№ докум.	Підпис	Дата		24

17. ГОСТ 12.1.005-88 – Общие санитарно-гигиенические требования к воздуху рабочей
18. ГОСТ 12.2.032-78 – ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования.
19. ДНАОП 0.00-1.31-99 – правила охорони праці під час експлуатації електронно-обчислювальних машин
20. ДСан Пін 3.3.2.007-98 – державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин
21. <http://zakon2.rada.gov.ua/laws/show/z0554-04> – наказ №151 від 02.04.2004 – про затвердження типових норм належності вогнегасників

					ДП.ПЗ - 04.00.00.000 ПЗ	Аркуш
						25
Зм.	Арк.	№ докум.	Підпис	Дата		

ДОДАТКИ

ДОДАТОК А

Домен для мапінгу даних

```
package com.diploma.ccms.domain;

import java.util.Collection;
import java.util.Date;
import java.util.HashSet;
import java.util.List;
import java.util.Set;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.EntityManager;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Lob;
import javax.persistence.ManyToOne;
import javax.persistence.PersistenceContext;
import javax.persistence.Temporal;
import javax.persistence.TemporalType;
import javax.persistence.TypedQuery;
import javax.persistence.Version;
import javax.validation.constraints.NotNull;
import org.apache.commons.lang3.builder.ReflectionToStringBuilder
    ;
import org.apache.commons.lang3.builder.ToStringStyle;
import org.springframework.beans.factory.annotation.Configurable;
import org.springframework.format.annotation.DateTimeFormat;
import org.springframework.security.core.GrantedAuthority;
import org.springframework.security.core.context.
    SecurityContextHolder;
import org.springframework.security.core.userdetails.UserDetails;
import org.springframework.transaction.annotation.Transactional;
import com.diploma.ccms.domain.service.DomainUserIntarface;

@Entity
@Configurable

public class Worker implements UserDetails, DomainUserIntarface {
```


Продовження додатку А

Продовження додатку А

```
private static final long serialVersionUID = 1148968754347580432L
    ;
@NotNull
@Column(unique = true)
private String login;
@NotNull
private String pass;
@NotNull
private String name;
@NotNull
private String surname;
private String phone;
private String mobile;
private String privateMail;
private String street;
@Temporal(TemporalType.DATE)
@DateTimeFormat(pattern = dd/MM/yyyy)
private Date birthday;
@Temporal(TemporalType.DATE)
@DateTimeFormat(pattern = dd/MM/yyyy)
private Date dateHire;
@ManyToOne
private WorkerRole roleName;
@NotNull
@ManyToOne
private Region regionName;
@NotNull
@ManyToOne
private Team teamName;
@NotNull
@ManyToOne
private WorkerJobType jobTypeName;
@Lob
private byte[] photo;
public static Worker getPrincipal(){
```



```
return (Worker) SecurityContextHolder.getContext().  
    getAuthentication().getPrincipal();  
}  
public String getLogin() {  
return this.login;  
}  
public void setLogin(String login) {  
this.login = login;  
}  
public String getPass() {  
return this.pass;  
}  
public void setPass(String pass) {  
this.pass = pass;  
}  
public String getName() {  
return this.name;  
}  
public void setName(String name) {  
this.name = name;  
}  
public String getSurname() {  
return this.surname;  
}  
public void setSurname(String surname) {  
this.surname = surname;  
}  
public String getPhone() {  
return this.phone;  
}  
public void setPhone(String phone) {  
this.phone = phone;  
}  
public String getPrivateMail() {  
return this.privateMail;  
}
```

```

public void setPrivateMail(String privateMail) {

this.privateMail = privateMail;

}

public String getStreet() {

return this.street;

}

public void setStreet(String street) {

this.street = street;

}

public Date getBirthday() {

return this.birthday;

}

public void setBirthday(Date birthday) {

this.birthday = birthday;

}

public Date getDateHire() {

return this.dateHire;

}

public void setDateHire(Date dateHire) {

this.dateHire = dateHire;

}

public WorkerRole getRoleName() {

return this.roleName;

}

public void setRoleName(WorkerRole roleName) {

this.roleName = roleName;

}

public Region getRegionName() {

return this.regionName;

}

public void setRegionName(Region regionName) {

this.regionName = regionName;

}

public Team getTeamName() {

return this.teamName;

}

```

```

}

public void setTeamName(Team teamName) {
this.teamName = teamName;
}

public WorkerJobType getJobTypeName() {
return this.jobTypeName;
}

public void setJobTypeName(WorkerJobType jobTypeName) {
this.jobTypeName = jobTypeName;
}

public byte[] getPhoto() {
return this.photo;
}

public void setPhoto(byte[] photo) {
this.photo = photo;
}

@PersistenceContext
transient EntityManager entityManager;

public static final EntityManager entityManager() {
EntityManager em = new Worker().entityManager;
if (em == null)
throw new IllegalStateException(Entity manager has not been
    injected (is the Spring Aspects JAR configured as an AJC/AJDT
    aspects library?));
return em;
}

public static long countWorkers() {
return entityManager().createQuery(SELECT COUNT(o) FROM Worker o,
    Long.class).getSingleResult();
}

public static List<Worker> findAllWorkers() {
return entityManager().createQuery(SELECT o FROM Worker o, Worker
    .class).getResultList();
}

public static Worker findWorker(Long id) {
if (id == null)

```

```

return null;

return entityManager().find(Worker.class, id);
}

public static List<Worker> findWorkerEntries(int firstResult, int
    maxResults) {
return entityManager().createQuery(SELECT o FROM Worker o, Worker
    .class).setFirstResult(firstResult).setMaxResults(maxResults).
    getResultList();
}

@Transactional
public void persist() {
if (this.entityManager == null)
this.entityManager = entityManager();
this.entityManager.persist(this);
}

@Transactional
public void remove() {
if (this.entityManager == null)
this.entityManager = entityManager();
if (this.entityManager.contains(this)) {
this.entityManager.remove(this);
} else {
Worker attached = Worker.findWorker(this.id);
this.entityManager.remove(attached);
}
}

@Transactional
public void flush() {
if (this.entityManager == null)
this.entityManager = entityManager();
this.entityManager.flush();
}

@Transactional
public void clear() {
if (this.entityManager == null)
this.entityManager = entityManager();

```

```

this.entityManager.clear();

}

@Transactional
public Worker merge() {
if (this.entityManager == null)
this.entityManager = entityManager();
Worker merged = this.entityManager.merge(this);
this.entityManager.flush();
return merged;
}

public String toString() {
return ReflectionToStringBuilder.toString(this, ToStringStyle.
    SHORT_PREFIX_STYLE);
}

public static TypedQuery<Worker> findWorkersByLoginEquals(String
    login) {
if (login == null || login.length() == 0)
throw new IllegalArgumentException(The login argument is required
    );
EntityManager em = Worker.entityManager();
TypedQuery<Worker> q = em.createQuery(SELECT o FROM Worker AS o
    WHERE o.login = :login, Worker.class);
q.setParameter(login, login);
return q;
}

@Id
@GeneratedValue(strategy = GenerationType.AUTO)
@Column(name = id)
private Long id;

@Version
@Column(name = version)
private Integer version;
public Long getId() {
return this.id;
}

public void setId(Long id) {

```

```
this.id = id;

}

public Integer getVersion() {
return this.version;
}

public void setVersion(Integer version) {
this.version = version;
}

@Override
public Collection<? extends GrantedAuthority> getAuthorities() {
Set<GrantedAuthority> roles = new HashSet<GrantedAuthority>();
roles.add(roleName);
return roles;
}

@Override
public String getPassword() {
return pass;
}

@Override
public String getUsername() {
return login;
}

@Override
public boolean isAccountNonExpired() {
return true;
}

@Override
public boolean isAccountNonLocked() {
return true;
}

@Override
public boolean isCredentialsNonExpired() {
return true;
}

@Override
public boolean isEnabled() {
```

```

return true;

}

public String getUiValue() {
return name + " " + surname;
}

public String getMobile() {
return mobile;
}

public void setMobile(String mobile) {
this.mobile = mobile;
}
}

```

Контроллер даних

```

package com.diploma.ccms.web.controller;

import java.io.UnsupportedEncodingException;
import java.util.ArrayList;
import java.util.List;

import javax.servlet.http.HttpServletRequest;
import javax.validation.Valid;

import org.joda.time.format.DateTimeFormat;
import org.springframework.context.i18n.LocaleContextHolder;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.validation.BindingResult;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.ResponseBody;
import org.springframework.web.util.UriUtils;
import org.springframework.web.util.WebUtils;

```



```
import com.diploma.ccms.domain.Region;

import com.diploma.ccms.domain.Team;
import com.diploma.ccms.domain.Worker;
import com.diploma.ccms.domain.WorkerJobType;
import com.diploma.ccms.domain.WorkerRole;

@RequestMapping("/workers")
@Controller
public class WorkerController {

    @RequestMapping(method = RequestMethod.POST, produces = "text/
        html")
    public String create(@Valid Worker worker, BindingResult
        bindingResult, Model uiModel, HttpServletRequest
        httpServletRequest) {
        if (bindingResult.hasErrors()) {
            populateEditForm(uiModel, worker);
            return "workers/create";
        }
        uiModel.asMap().clear();
        worker.persist();
        uiModel.addAttribute("menu", "WORKER");
        return "redirect:/workers/" + encodeUrlPathSegment(worker.getId()
            .toString(), httpServletRequest);
    }

    @RequestMapping(params = "form", produces = "text/html")
    public String createForm(Model uiModel) {
        populateEditForm(uiModel, new Worker());
        List<String[]> dependencies = new ArrayList<String[]>();
        if (Region.countRegions() == 0) {
            dependencies.add(new String[] { "region", "regions" });
        }
        if (Team.countTeams() == 0) {
            dependencies.add(new String[] { "team", "teams" });
        }
    }
}
```

```

}

if (WorkerJobType.countWorkerJobTypes() == 0) {
dependencies.add(new String[] { "workerjobtype", "workerjobtypes"
    });
}
uiModel.addAttribute("dependencies", dependencies);
uiModel.addAttribute("menu", "WORKER");
return "workers/create";
}

@RequestMapping(value =("/{id}", produces = "text/html")
    public String show(@PathVariable("id") Long id, Model uiModel) {
addDateTimeFormatPatterns(uiModel);
uiModel.addAttribute("worker", Worker.findWorker(id));
uiModel.addAttribute("itemId", id);
uiModel.addAttribute("menu", "WORKER");
return "workers/show";
}

@RequestMapping(produces = "text/html")
    public String list(@RequestParam(value = "page", required = false
        ) Integer page, @RequestParam(value = "size", required = false
        ) Integer size, Model uiModel) {
uiModel.addAttribute("workers", Worker.findAllWorkers());
//
uiModel.addAttribute("menu", "WORKER");
addDateTimeFormatPatterns(uiModel);
return "workers/list";
}

@RequestMapping(method = RequestMethod.PUT, produces = "text/html
    ")
    public String update(@Valid Worker worker, BindingResult
        bindingResult, Model uiModel, HttpServletRequest
        httpRequest) {
if (bindingResult.hasErrors()) {

```

```

populateEditForm(uiModel, worker);

uiModel.addAttribute("menu", "WORKER");
return "workers/update";
}

uiModel.asMap().clear();
worker.merge();
uiModel.addAttribute("menu", "WORKER");
return "redirect:/workers/" + encodeUrlPathSegment(worker.getId()
    .toString(), httpRequest);
}

@RequestMapping(value =("/{id}", params = "form", produces = "
    text/html")
public String updateForm(@PathVariable("id") Long id, Model
    uiModel) {
populateEditForm(uiModel, Worker.findWorker(id));
uiModel.addAttribute("menu", "WORKER");
return "workers/update";
}

@RequestMapping(value =("/{id}", method = RequestMethod.DELETE,
    produces = "text/html")
public String delete(@PathVariable("id") Long id, @RequestParam(
    value = "page", required = false) Integer page,
    @RequestParam(value = "size", required = false) Integer size,
    Model uiModel) {
Worker worker = Worker.findWorker(id);
worker.remove();
uiModel.asMap().clear();
uiModel.addAttribute("page", (page == null) ? "1" : page.toString
    ());
uiModel.addAttribute("size", (size == null) ? "10" : size.
    toString());
uiModel.addAttribute("menu", "WORKER");
return "redirect:/workers";
}

```

```

void addDateTimeFormatPatterns(Model uiModel) {
uiModel.addAttribute("worker_birthdate_date_format",
    DateTimeFormat.patternForStyle("M-", LocaleContextHolder.
        getLocale()));
uiModel.addAttribute("worker_datehire_date_format",
    DateTimeFormat.patternForStyle("M-", LocaleContextHolder.
        getLocale()));
}

```

```

void populateEditForm(Model uiModel, Worker worker) {
uiModel.addAttribute("worker", worker);
addDateTimeFormatPatterns(uiModel);
uiModel.addAttribute("regions", Region.findAllRegions());
uiModel.addAttribute("teams", Team.findAllTeams());
uiModel.addAttribute("workerjobtypes", WorkerJobType.
    findAllWorkerJobTypes());
uiModel.addAttribute("workerroles", WorkerRole.findAllWorkerRoles
    ());
uiModel.addAttribute("menu", "WORKER");
}

```

```

String encodeUrlPathSegment(String pathSegment,
    HttpServletRequest httpRequest) {
String enc = httpRequest.getCharacterEncoding();
if (enc == null) {
enc = WebUtils.DEFAULT_CHARACTER_ENCODING;
}
try {
pathSegment = UriUtils.encodePathSegment(pathSegment, enc);
} catch (UnsupportedEncodingException uee) {
}
return pathSegment;
}

```

```

@RequestMapping(params = { "find=ByLoginEquals", "form" }, method

```

```

        = RequestMethod.GET)

    public String findWorkersByLoginEqualsForm(Model uiModel) {
        uiModel.addAttribute("menu", "WORKER");
        return "workers/findWorkersByLoginEquals";
    }

    @RequestMapping(params = "find=ByLoginEquals", method =
        RequestMethod.GET)
    public String findWorkersByLoginEquals(@RequestParam("login")
        String login, Model uiModel) {
        uiModel.addAttribute("workers", Worker.findWorkersByLoginEquals(
            login).getResultList());
        uiModel.addAttribute("menu", "WORKER");
        return "workers/list";
    }

    /**
     * Generate preview for e-mail
     *
     * @param petId
     * @param model
     * @return
     */
    @RequestMapping(value = "/preview_email", method = RequestMethod.
        POST)
    @ResponseBody
    public String getPreview(@RequestParam("data") String data) {
        return data;
    }
}

```

Сервіс для збереження даних

```
package com.diploma.ccms.service;
```


Продовження додатку А

Продовження додатку А

```
import org.springframework.security.core.userdetails.UserDetails;

import org.springframework.security.core.userdetails.
    UserDetailsService;
import org.springframework.security.core.userdetails.
    UsernameNotFoundException;

import com.diploma.ccms.domain.Worker;

public class UserService implements UserDetailsService {

    @Override
    public UserDetails loadUserByUsername(String username) throws
        UsernameNotFoundException {
        return Worker.findWorkersByLoginEquals(username).getSingleResult
            ();
    }

}
```

Інтерфейс для збереження даних користувача

```
package com.diploma.ccms.domain.service;

public interface DomainUserInterface {
    /**
     * used for getting value for showing on UI form
     */
    public String getUiValue();
}
```

Вигляд створення для користувача

```
<?xml version=1.0 encoding=UTF-8 standalone=no?>
<div xmlns:c=http://java.sun.com/jsp/jstl/core xmlns:field=urn:
    jsptagdir:/WEB-INF/tags/form/fields xmlns:form=urn:jsptagdir:/
    WEB-INF/tags/form
```



```
xmlns:jsp=http://java.sun.com/JSP/Page xmlns:spring=http://www.
springframework.org/tags version=2.0>
<jsp:directive.page contentType=text/html;charset=UTF-8 />
<jsp:output omit-xml-declaration=yes />

<section class=main-section grid_7>

<div class=main-content>
<header>
<h2>Employees</h2>
</header>
<section class=container_6 clearfix>
<field:input field=login id=
    c_com_diploma_ccms_domain_Worker_login required=true label=
    Login smallLabel=Enter login type=text/>
<field:input field=pass id=c_com_diploma_ccms_domain_Worker_pass
    required=true label=Password smallLabel=Enter password type=
    password/>
<field:input field=name id=c_com_diploma_ccms_domain_Worker_name
    required=true label=Name smallLabel=Enter name type=text/>
<field:input field=surname id=
    c_com_diploma_ccms_domain_Worker_surname required=true label=
    Surname smallLabel=Enter surname type=text/>
<field:input field=mobile id=
    c_com_diploma_ccms_domain_Worker_mobile required=false label=
    Mobile phone number smallLabel=Private phone number type=text
    />
<field:input field=privateMail id=
    c_com_diploma_ccms_domain_Worker_privateMail required=false
    label=Email smallLabel=A valid email address type=email/>
<field:select field=regionName id=
    c_com_diploma_ccms_domain_Worker_regionName items=${regions}
    required=true label=Region smallLabel=Please select/>
<field:input field=street id=
    c_com_diploma_ccms_domain_Worker_street required=false label=
```

```

    Address smallLabel=Enter full home address type=text/>

<field:input field=birthday id=
    c_com_diploma_ccms_domain_Worker_birthday required=false label
    =Birthdate smallLabel=mm/dd/yyyy type=date/>
<field:input field=dateHire id=
    c_com_diploma_ccms_domain_Worker_dateHire required=false label
    =Hire Date smallLabel=mm/dd/yyyy type=date/>
<field:select field=roleName id=
    c_com_diploma_ccms_domain_Worker_roleName items=${workerroles}
    required=true label=Employee role smallLabel=Please select
    role/>
<field:select field=teamName id=
    c_com_diploma_ccms_domain_Worker_teamName items=${teams}
    required=true label=Employee related team smallLabel=Please
    select working team/>
<field:select field=jobTypeName id=
    c_com_diploma_ccms_domain_Worker_jobTypeName items=${
    workerjobtypes} required=true label=Job type smallLabel=Like
    QA, Developer, etc../>
<field:input field=phone id=
    c_com_diploma_ccms_domain_Worker_phone required=false label=
    Phone number smallLabel=Working office phone number type=text
    />
<field:textarea field=photo id=
    c_com_diploma_ccms_domain_Worker_photo />
</form:create>
</section>
</div>
</section>
</div>

```

Вигляд для показу даних про

```

<?xml version=1.0 encoding=UTF-8 standalone=no?>
<div xmlns:field=urn:jsptagdir:/WEB-INF/tags/form/fields
xmlns:c=http://java.sun.com/jsp/jstl/core
xmlns:sec=http://www.springframework.org/security/tags

```


Продовження додатку А

Продовження додатку А

```
xmlns:jsp=http://java.sun.com/JSP/Page xmlns:page=urn:jsptagdir:/
    WEB-INF/tags/form version=2.0>
<jsp:directive.page contentType=text/html;charset=UTF-8 />
<jsp:output omit-xml-declaration=yes />
<!-- Main Section -->
<section class=main-section grid_7>
<div class=main-content grid_4 alpha>
<header class=clearfix>
<span class=avatar> <!-- -->
</span>
<hgroup>
<h2>${worker.name} ${worker.surname}</h2>
<h4>
<a href=#>${worker.jobTypeName.jobTypeName}</a>
</h4>
</hgroup>
<p class=tags>
<!-- <a href=#>Add To Contacts</a> -->
</p>
</header>
<section>
<script type=text/javascript>
    $(document).ready(function() {
        // Add markItUp! to your textarea in one line
        $('.markItUpTextarea').markItUp(mySettings, {
            root : 'markitup/skins/simple/'
        });
    });
</script>
<style>
.markItUp {
width: 390px;
}
</style>
<sec:authentication var=principal_login property=principal.login
```

```

/>

<c:choose>
<c:when test=${worker.login eq principal_login}>
<h3> Have a nice day ;) </h3>
</c:when>
<c:otherwise>
<h3>Send private message to ${worker.name} ${worker.surname}</h3>
<form method=post action=/ccms/messages?id=${worker.id} style=
width: 400px;>
<field:input id=c_com_diploma_ccms_domain_Message_title field=
title required=true label=Title smallLabel= type=text/>
<textarea id=c_com_diploma_ccms_domain_Message_text name=text
class=markItUpTextarea style=height: 80px; width: 100%;
required=required><!-- --></textarea>
<button class=fr button button-gray type=submit>Send message</
button>
</form>
</c:otherwise>
</c:choose>
<div class=clear>
<!-- -->
</div>
<!-- <h3>History</h3>
<ul class=listing list-view>
<li class=note><a href=editnote.html class=more></a> <span class=
timestamp>Dec 28, 2010</span>
<p>Vestibulum ultrices vehicula leo ac tristique. Mauris id nisl
nibh.</p>
<div class=entry-meta>Posted by Administrator</div></li>
<li class=note><a href=editnote.html class=more></a> <span class=
timestamp>Dec 28, 2010</span>
<p>Vestibulum ultrices vehicula leo ac tristique. Mauris id nisl
nibh.</p>
<div class=entry-meta>Posted by Administrator</div></li>
<li class=note><a href=editnote.html class=more></a> <span class=
timestamp>Dec 28, 2010</span>

```

```

<p>Vestibulum ultrices vehicula leo ac tristique. Mauris id nisl
  nibh.</p>
<div class=entry-meta>Posted by Administrator</div></li>
</ul> -->
</section>
</div>
<div class=preview-pane grid_3 omega>
<div class=content>
<h3>${worker.name}s contact information</h3>
<ul class=profile-info>
<li class=email><c:if test=${empty worker.privateMail}>:: none
  ::</c:if> ${worker.privateMail}<span>email</span></li>
<li class=mobile><c:if test=${empty worker.mobile}>:: none ::</c:
  if>${worker.mobile}<span>mobile</span></li>
<li class=phone><c:if test=${empty worker.phone}>:: none ::</c:if
  >${worker.phone}<span>work</span></li>
</ul>
<!-- <h3>Tasks About John</h3>
None so far. <a href=#>Add a task now</a>  -->
<h3>Additional info</h3>
<ul class=profile-info>
<li class=calendar-day><c:if test=${empty worker.birthday}>::
  none ::</c:if>${worker.birthday}<span>birthday</span></li>
<li class=calendar-day><c:if test=${empty worker.dateHire}>::
  none ::</c:if>${worker.dateHire}<span>hire date</span></li>
<li class=building><c:if test=${empty worker.regionName.
  regionName}>:: none ::</c:if>${worker.regionName.regionName}<
  span>living region</span></li>
<li class=house><c:if test=${empty worker.street}>:: none ::</c:
  if>${worker.street}<span>home address</span></li>
</ul>
</div>
<div class=preview>
<!-- -->
</div>
</div>

```

```

</section>

<!-- Main Section End -->

<!-- <page:show id=ps_com_diploma_ccms_domain_Worker object=${
    worker}
path=/workers z=iQamgfMz7tfHSdGgWB8YCdDd83Q=>
<field:display field=login
id=s_com_diploma_ccms_domain_Worker_login object=${worker}
z=4HjNYZAHZPDLTqADXwCdyd4GRAk= />
<field:display field=pass id=
    s_com_diploma_ccms_domain_Worker_pass
object=${worker} z=HYhzipM8Xx937WviE05t1DOTtIbU= />
<field:display field=name id=
    s_com_diploma_ccms_domain_Worker_name
object=${worker} z=egBK4f8HsCgN5Ynj/jGjRUelAo= />
<field:display field=surname
id=s_com_diploma_ccms_domain_Worker_surname object=${worker}
z=ES6rDyTeRfbtJ4oZ9zGg6RpkdIO= />
<field:display field=phone
id=s_com_diploma_ccms_domain_Worker_phone object=${worker}
z=1VohSBGktVWKME/kgylZA99MluU= />
<field:display field=privateMail
id=s_com_diploma_ccms_domain_Worker_privateMail object=${worker}
z=JZug5Vfq4gKYkQsNndLyunI+pLo= />
<field:display field=street
id=s_com_diploma_ccms_domain_Worker_street object=${worker}
z=e/Rkpgq+iZVtm51YsnThOISuQXRA= />
<field:display date=true
dateTimePattern=${worker_birthday_date_format} field=birthday
id=s_com_diploma_ccms_domain_Worker_birthday object=${worker}
z=EmRENzOD8cefRn6vszpdLPB3+04= />
<field:display date=true
dateTimePattern=${worker_datehire_date_format} field=dateHire
id=s_com_diploma_ccms_domain_Worker_dateHire object=${worker}
z=oQ0LtnMFCgHhLh6m2nvl4LtMrJU= />
<field:display field=roleName
id=s_com_diploma_ccms_domain_Worker_roleName object=${worker}

```

```
z=wJnkS5H0d6iZ3AnuqTU07TI2wX8= />

<field:display field=regionName
id=s_com_diploma_ccms_domain_Worker_regionName object=${worker}
z=obgn4zC59pMlQ8GaXlDJYjUcoGs= />
<field:display field=teamName
id=s_com_diploma_ccms_domain_Worker_teamName object=${worker}
z=iRDO/HwFc7JQ+f+sLvRXQwEzBOw= />
<field:display field=jobTypeName
id=s_com_diploma_ccms_domain_Worker_jobTypeName object=${worker}
z=a43X3qenfQB/KwX1hCUeDmDGIYM= />
<field:display field=photo
id=s_com_diploma_ccms_domain_Worker_photo object=${worker}
z=lcI4umkwCdImQFaIzB4gefboNPI= />
</page:show>
-->
</div>
```


БІБЛІОГРАФІЧНА ДОВІДКА

ТЕМА ДИПЛОМНОГО ПРОЕКТУ: «Розробка програмного та алгоритмічного забезпечення багатofункціональної корпоративної системи для спільної роботи, управління документами і проектами засобами Java EE»

Обсяг пояснювальної записки: 57 аркушів

Дата закінчення проекту 4 травня 2012 р.

Підпис студента-дипломника _____

АНОТАЦІЯ

В дипломному проекті розроблено корпоративну систему для управління документами, користувачами, проектами з можливістю спільної роботи. Визначено основні аспекти, що впливають на створення програмного продукту. Розроблено серверну і користувацькі частини порталу.

КЛЮЧОВІ СЛОВА: корпоративний портал, сумісна робота, документація, Java EE, Spring Framework.

АНОТАЦИЯ

В дипломном проекте разработана корпоративная система для управления документами, пользователями, проектами из возможностью совместной работы. Определены основные аспекты, влияющие на создание программного продукта. Разработана серверная и пользовательские части портала.

КЛЮЧЕВЫЕ СЛОВА: корпоративный портал, совместная работа, документация, Java EE, Spring Framework.

ANNOTATION

In a diploma project there was developed a corporate system for document management, users, projects with collaboration ability. Provided main aspects that influence on creation of software. Developed server and user part of portal.

KEY WORDS: corporate portal, collaboration, documentation, Java EE, Spring Framework.