

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проєктами
Кафедра інформаційних управляючих систем та технологій

"Допущений до захисту"

Завідувач кафедри ІУСТ



к.т.н, доц. Михелєв І.Л.

" ____ " _____ 2026 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти "бакалавр"

на тему: Розробка автоматизованої системи технологічної підготовки
токарних операцій

Виконав: студент групи

 *Журжій В.В.*
(підпис)

Керівник роботи:

PhD
(посада, науковий ступінь вчене звання)
 *Ворона М.В.*
(підпис)

м. Миколаїв – 2026 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій
Спеціальність 122 "Комп'ютерні науки"
Освітня програма "Комп'ютерні науки"

"ЗАТВЕРДЖУЮ"

Гарант освітньої програми


к.т.н, доц. Гайда А.Ю.

" ____ " ____ 2026 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття ступеня вищої освіти "бакалавр"

Студенту Журжію Володимирі Володимировичу

(прізвище, ім'я, по батькові)

1. Тема роботи: Розробка автоматизованої системи технологічної підготовки токарних операцій

Керівник роботи Ворона Михайло Владиславович

Затверджені наказом ректора № 270-уч від "31" березня 2026 року.

2. Термін подання роботи: 22 червня 2026 р.

3. Вихідні дані до проекту (роботи) ДСТУ щодо обробки інформації, літературні джерела, технічна документація на існуючі аналоги систем,

4. Перелік питань, що належать до розробки (найменування розділів):

1. Розробка концепції інформаційної системи 2. Розробка проекту інформаційної системи. 3. Реалізація проекту ІС 4. Охорона праці

5. Перелік презентаційних матеріалів Титульний слайд, Мета, об'єкт, предмет та актуальність дослідження, Організаційна структура, Концепція інформаційної системи, DFD-діаграми бізнес-процесів інформаційної системи, Проектування бази даних та UML-модель системи, Алгоритм формування рейтингових списків, Використані технології розробки, Результати впровадження

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Ворона М.В.	02.02.2026	23.02.2026
2	Ворона М.В.	23.02.2026	20.03.2026
3	Ворона М.В.	20.03.2026	24.04.2026
4		24.04.2026	01.06.2026

7. Дата видачі завдання 02 лютого 2026 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів роботи	Примітка
1	Аналіз предметної галузі	02.02.2026	
2	Постановка задачі	16.02.2026	
3	Розробка концепції	23.02.2026	
4	Розробка проекту ІС	23.03.2026	
5	Реалізація проєкту	27.04.2026	
6	Розробка питань з охорони праці	01.06.2026	
7	Розробка графічних матеріалів	05.06.2026	
8	Подання роботи на перевірку на плагіат	08.06.2026	
9	Подання роботи рецензенту	15.06.2026	
10	Подання роботи на захист	22.06.2026	

Студент


(підпис)

Журжій В.В.
(ПІБ)

Керівник роботи


(підпис)

Ворона М.В.
(ПІБ)

Анотація

Представлена робота присвячена розробці підсистеми технологічної підготовки токарних операцій машинобудівного підприємства з метою підвищення продуктивності роботи технологів.

Під час роботи формуються вимоги до системи, розроблюються концепції системи і проводиться вибір найкращої, розробляється технічне завдання.

Проводиться розробка основних проєктних рішень за технічним, інформаційним та програмним забезпеченням, формується робоча документація за програмним забезпеченням.

Дипломна робота виконана на 107 аркушах машинописного тексту, містить 11 рисунків, 22 таблиці, 4 додатки, список використаних джерел містить 19 найменувань.

Ключові слова: інформаційна система, токарні операції, база даних, Java.

Abstract

The presented work is devoted to the development of a subsystem of technological preparation of turning operations of a machine-building enterprise in order to increase the productivity of technologists.

During the work, the requirements for the system are formed, system concepts are developed and the best one is selected, and the terms of reference are developed. The development of the main design solutions for technical, informational and software is carried out, working documentation for the software is formed.

The diploma work was made on 107 sheets of typewritten text, contains 11 figures, 22 tables, 4 appendices, the list of sources used contains 19 titles.

Keywords: information system, turning operations, database, Java.

ЗМІСТ

Вступ	
1. Опис предметної області, розробка концепції та формування вимог до підсистеми технологічної підготовки токарних робіт	
1.1. Формування вимог до підсистеми.....	
1.2. Розробка концепції інформаційної системи.....	
2. Розробка проектних рішень щодо системи та її частин.....	
2.1. Загальносистемні рішення.....	
2.2. Рішення з організаційного забезпечення.....	
2.3. Опис організації інформаційної бази	
2.4. Рішення по технічному забезпеченню.....	
2.5. Рішення по програмному забезпеченню.....	
3. Розробка робочої документації підсистеми технологічної підготовки токарних операцій	
4. Охорона праці на машинобудівному підприємстві	
Висновки.....	
Список використаної літератури.....	
Додатки	

Вступ

Завданням дипломної роботи являється розробка підсистеми технологічної підготовки токарних операцій та АРМ технолога токарної дільниці машинобудівного підприємства, призначеної для технологічної підготовки виробництва, розробки технологічного процесу виробництва та створення технологічної документації для розробленого ТП. Дана система розробляється для впровадження на ПАТ «КВАДР», і тому розробка ведеться виходячи з вимог, висунутих даним підприємством. На поточний момент на підприємстві весь процес створення технологічної документації робиться технологами власноруч, що займає забагато часу та зусиль, отож впровадження сучасних інформаційних технологій просто необхідний. Росіяни вже розробили декілька автоматизованих систем технологічної підготовки виробництва («ДиаМан», КОМПАС-Автопроект, ЧПУ ГЕММА-3D), але ці системи знаходяться в опитній експлуатації, та для їх впровадження в дане підприємство необхідно зробити багато змін, як в самій програмі, так і структурі підприємства. І головною причиною неможливості їх впровадження – це їх велика вартість, яку дане підприємство просто не спроможне собі дозволити.

Впровадження розробленої інформаційної системи дозволить зменшити витрати часу на розробку технологічної документації, полегшити та зменшити витрати часу на підбір обладнання, інструменту, матеріалу та інших ресурсів, полегшити керівникам контролювати розробку технологічного процесу, що дозволить їм більш ефективно виконувати свої функції, знизити ризик помилок у оформленні документації та у коректності вибору певних ресурсів, підвищити швидкість процесу обробки інформації і знизити витрати, пов'язані з ним.

Цілями розробки автоматизованої системи є:

- підвищення швидкості та спрощення процесу створення технологічної документації за рахунок автоматизації даного процесу;
- зменшення ризику помилок при прийнятті рішень через невірогідність інформації про функціонування підприємства;
- зменшення витрат на документообіг.

Для досягнення поставленої мети повинні бути вирішені наступні основні задачі:

- проведення аналізу процесу створення технологічної документації та документообігу при розробці ТП;
- формування вимог до системи;
- розроблення загальносистемних рішень,
- розробка рішень з організаційного, інформаційного, технічного та програмного забезпечень;
- проведення дослідження з охорони праці;
- розробка відповідної документації.

1 Аналіз предметної області та постановка задачі розроблення системи технологічної підготовки токарних операцій машинобудівного заводу

1.1. Аналіз предметної області та формування вимог до системи технологічної підготовки токарних операцій

1.1.1. Характеристики об'єкта та результати його діяльності

Об'єктом дослідження, для якого розробляється система, є інформаційна інфраструктура ПАТ «КВАДР» — підприємства, що спеціалізується на виробництві механізмів і вузлів для потреб сільського господарства, а також виконує ремонтні роботи за заявками цукрових заводів тощо.

Дане підприємство складається з наступних підрозділів, структурна схема яких зображена на рисунку 1.1.

Директор:

- керує та контролює фінансові і робочі процеси підприємства;
- відслідковує графік виробництва об'єктів для загального розвитку підприємства, удосконалення технології виробництва;
- одержує від Замовника технічну документацію на виробництво;
- укладає Договір на виробництво;
- дає розпорядження Технічному директору на підготовку робочої конструкторської та технологічної документації, підготовку виробництва, організацію робіт у виробничих цехах;

Технічний директор(ТД):

- підготовляє, організує та управляє процесом виробництва;
- веде безпосередній контроль за виконанням робіт у всіх підрозділах підприємства згідно до графіку виробництва.

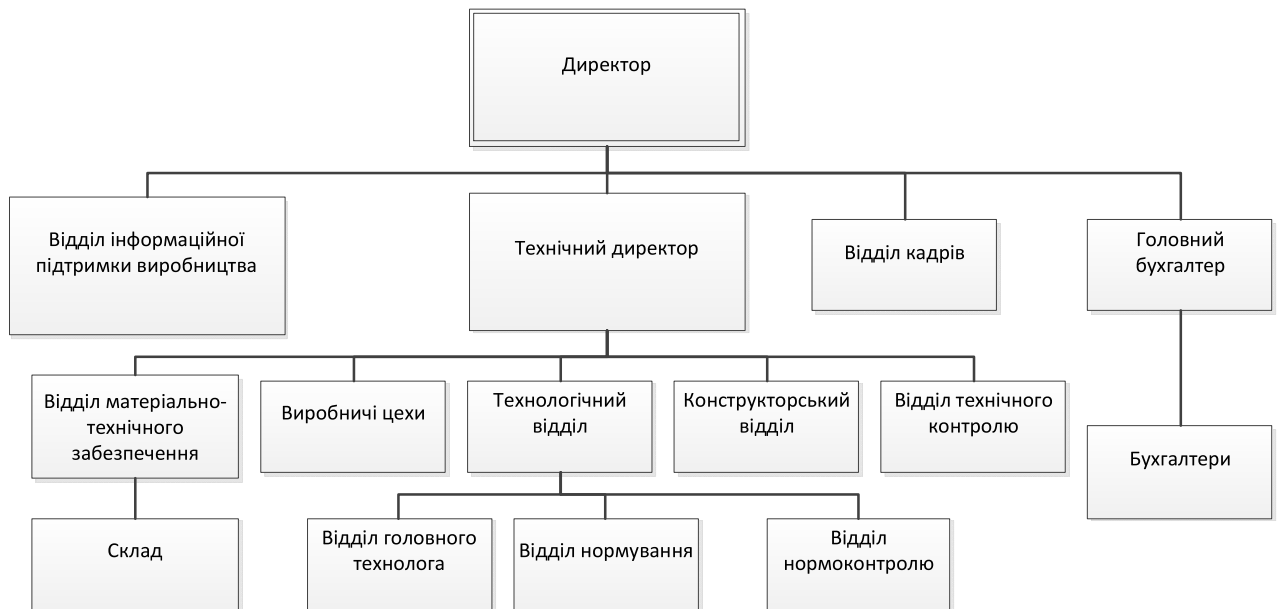


Рисунок 1.1 – Структурна схема підрозділів ПАТ „Миколаївський машинобудівний завод”

Бухгалтер:

- веде облік і контролює фінансові операції на підприємстві;
- своєчасно звітує у державних інспекціях;
- надає Директору звіт про фінансовий стан підприємства і складає економічний прогноз роботи підприємства.

Відділ інформаційної підтримки виробництва(ВІПВ):

- проводить установку та налагодження програмних і апаратних засобів на підприємстві;
- відповідає за коректну роботу локальних мереж та програмного забезпечення.

Відділ кадрів приймає на роботу та веде облік існуючих на підприємстві працівників.

Виробничі цеха займаються безпосередньо виробництвом згідно до технологічного процесу. Серед усіх виробничих цехів провідне місце займає

механозбиральний цех, як за наявністю обладнання, так і по продукції, яку він виробляє. Цех випускає шпилі (якірні й швартові), лебідки, підйомники, валопроводи й дейдвудні пристрої, шлюпбалки, ахтерштевні, форштевні та інші механізми для суднобудування. Перераховані механізми й вузли складаються з багатьох деталей і підвузлів. Характерними деталями, виготовленими на ділянках цеха є зубчасті колеса, вали, вали-шестерні, корпуси редукторів, глечика, кронштейни та ін. Механозбиральний цех належить до цехів, у яких відбувається одиничне виробництво продукції, або виробництво продукції невеликими серіями. Тому все обладнання цеху розставлено по групах, відмінно від цехів із серійним і масовим виробництвом, де верстати встановлюються по ходу технологічного процесу. Цех розподілено на виробничі ділянки, виходячи зі специфіки робіт і встановленого обладнання. Крім основного специфічного обладнання ділянки, на кожній ділянці є ряд інших верстатів (у невеликих кількостях) для більшого замикання виробничого циклу.

Відділ технічного контролю(ВТК):

- здійснює передбачений технологічним процесом контроль на стадіях виробничого процесу.

Відділ матеріально технічного забезпечення(ВМТЗ):

- обирає постачальників та узгоджує ціни на матеріали та обладнання;
- закупляє за специфікацією, контролем якості та термінів поставки узгоджене обладнання та матеріали згідно до графіку закупки;
- передає закуплені матеріали та обладнання на склад і звітує про це перед ТД.

Склад:

- приймає, веде облік, розміщує, зберігає і відпускає матеріально-технічних ресурси;
- звітує перед бухгалтером і Технічним директором про надходження, видачу та залишки матеріалів.

Обладнання і матеріали на склад надходять від ВМТЗ, приймаються по накладних, видаються у виробничі цеха згідно розпорядження ТД.

Конструкторський відділ(КВ):

- розробляє та видає робочу конструкторську документацію на підприємстві (креслення деталі, складальної одиниці, креслення або ескіз заготівки);
- розробляє специфікації на матеріали.

Відділ головного технолога (ВГТ):

- розробляє технологічний процес на підприємстві;
- підготовлює та видає робочу технологічну документацію.

Відділ нормування(ВН):

- розробляє норми на виробництво;
- визначає професію та кваліфікацію працівників, необхідних для виконання робіт згідно з потребами технологічного процесу.

Відділ нормоконтролю:

- перевіряє дотримання норм і вимог встановлених стандартами в документах, що розробляються ;
- перевіряє правильність оформлення документів згідно з вимогами діючих систем стандартів;

- перевіряє раціональне використання встановлених обмежувальних номенклатур устаткування оснащення, матеріалів, профілів і розмірів прокату.

Нормоконтроль проводиться з метою підвищення рівня типізації технологічного процесу, уніфікації технологічної документації, обладнання, скорочення строків підготовки виробництва, зниження собівартості. Технологічна документація без підпису нормоконтролера прийому в бюро нормативно технічної документації або в підрозділи, що його замінюють, розмноженню й використанню для підготовки виробництва не підлягає.

Процес виробництва відбувається таким чином. Початковою стадією усієї роботи є отримання від Замовника технічного завдання(ТЗ) та, якщо є, проектно-конструкторської документації. Після узгодження ТЗ підписується Договір на виробництво. Виконавець, тобто ПАТ „КВАДР”, отримує необхідні платежі для підготовки та початку виробництва. Наступним етапом є проектування, розробка технологічного процесу та послідовне видання пакету робочої документації на підприємстві. Далі складається графік виробництва узгоджений з поставками матеріалів та заготовок. Підписуються Договори з фірмами постачальниками і розпочинається виробництво. У процесі виробництва Виконавець отримує від Замовника відповідні платежі згідно Договору на виробництво та загальної специфікації придбання матеріалів. По закінченню виробництва проводиться остаточний контроль та іспити, та після підписання акту прийому-передачі Виконавець передає пакет необхідних документів Замовнику.

Великий обсяг часу усього виробничого процесу займає розробка технологічного процесу та технологічної документації до нього. Технологічний процес (ТП) - частина виробничого процесу, безпосередньо пов'язана зі зміною форми, розмірів або властивостей оброблюваної заготовки, виконується в певній послідовності. Розробка ТП виконується для виготовлення виробів, конструкція яких відпрацьована на технологічність. ТП повинен відповідати

вимогам безпеки й промислової санітарії, викладеним у системі стандартів безпеки праці. Технологічний процес розробляється для виконання операцій обробки різанням, зборки, випробування й пов'язаних з ними операцій переміщення, розмітки, слюсарної обробки, зварювання, покриття, промивання, зважування готових деталей, маркування, консервації фарбування, технічного контролю, упакування й т.д. Технологічний процес виконується робітниками за допомогою технологічного обладнання, інструментів, пристосувань, розміщених в на робочому місці. Технологічний процес звичайно ділять на частини, які називаються операціями (закінчена частина технологічного процесу, виконувана на одному робочому місці). Операція є найменшою частиною технологічного процесу, для якої розробляється технологічна документація, по якій ведеться планування й облік. Необхідність розподілу технологічного процесу на операції породжена двома причинами:

1. фізичними (неможливо обробити деталь із шести сторін; необхідно різне обладнання для чистової й чорнової обробки);
2. економічними (доцільність створення спеціального верстата).

Вихідні дані для розробки технологічного процесу:

- креслення деталі, складальної одиниці й специфікація (основні документа);
- креслення або ескіз заготовки;
- відомість подетальних норм або документ що її заміняє (відомість технологічних маршрутів...);
- обсяг випуску деталей (виробів);
- тип виробництва;
- склад технологічного встаткування;
- нормативно-технічна документація;
- графік розробки, і випуску технологічних процесів.

В обґрунтованих випадках вихідні дані повинні бути доповнені картою термообробки й ескізом до неї.

Встановлено три види ТП: одиничний, типовий та груповий, котрі по ступені деталізації операцій можуть бути маршрутними, маршрутно-операційними та операційними.

Розробка технологічних процесів у загальному випадку включає комплекс взаємозалежних робіт:

- вибір заготовок;
- вибір технологічних баз;
- підбор типового технологічного процесу;
- визначення послідовності й змісту технологічних операцій;
- визначення, вибір і замовлення нових засобів технологічного оснащення (у тому числі засобів контролю й випробування);
- призначення розрахункових даних, установлення режимів обробки (при розробці операційного технологічного процесу);
 - нормування, технологічного процесу;
 - визначення професій і кваліфікації виконавців;
- вибір засобів механізації й автоматизації елементів технологічних процесів і внутрицехових засобів транспортування;
- оформлення робочої документації на технологічні процеси.

1.1.2. Опис діючої системи розробки технологічної документації підприємства

На поточний момент на підприємстві не має єдиної, взаємозалежної і повноцінно функціонуючої автоматизованої інформаційної системи підготовки технологічної документації. Всі розробки виконуються технологами власноруч, використовуючи довідники та власний досвід. Для більш повного розуміння усього процесу розробки ТП розглянемо його за допомогою діаграми IDEF 0, що дасть можливість побачити як потоки даних, так і керуючі потоки і механізми.

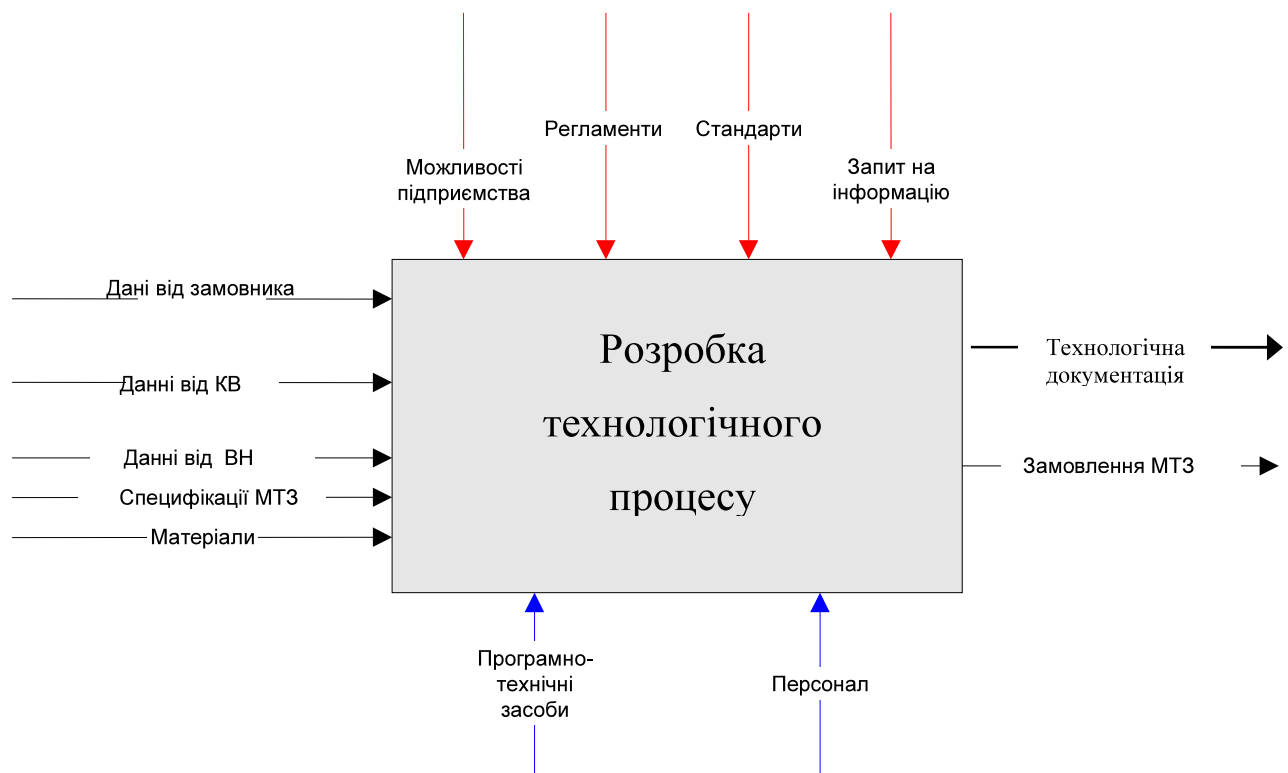


Рисунок 1.2 – Контекстна діаграма процесу розробки технологічної документації підприємства ПАТ „КВАДР”

Функції або роботи зображуються у вигляді прямокутника, у даному випадку **«Розробка технологічного процесу»**. Дані й об'єкти зображуються у виді стрілок, як зв'язок між об'єктами.

Стрілки входу – зображують дані, змінювані в ході роботи. Вхідними потоками є:

«**Дані замовника**» - являє собою технічне завдання (призначення деталі, її властивості, вимоги і т.д.);

«**Дані від КВ**» - являє собою робочу технічну документацію. (креслення деталі, креслення заготовки);

«**Дані від ВН**» - являє собою документ, в якому визначається професія, розряд роботи та ін;

«Специфікації матеріалів» - являє собою потік несучий різномірну інформацію, що, так чи інакше, може вплинути на діяльність підприємства;

Стрілки управління (входять у верхню грань) – зображують правила й обмеження, згідно яким виконується робота.

«Можливості підприємства» – потік управлінського класу, свого роду обмеження, що накладається на якість послуг, швидкість обслуговування й інші функції аналізованої системи;

«Регламенти» – потік управлінського класу, спрямований на поліпшення роботи підприємства. (Команди керівника, рішення ради директорів);

«Стандарти» – потік управлінського класу, що визначає правила та вимоги при проектуванні і виробництві. Даний керуючий процес є дуже важливим, тому що вироблена деталь повинна відповідати певним стандартам;

«Запит на інформацію» – потік управлінського класу, що представляє собою бажання замовника одержати інформацію про товари і послуги наданих підприємством.

Стрілки виходу зображують дані об'єкти, що з'являються в результаті виконання роботи.

«Технологічна документація» – являє собою розроблену, оформлену певним чином технологічну документацію ;

«Замовлення матеріалів» – являє собою потік, що несе інформацію про кількість і номенклатуру матеріалів, що замовляються.

Стрілки механізму (входять у нижню грань роботи) – зображують ресурси, необхідні для виконання роботи, але не змінюються в процесі роботи (обладнання, людські ресурси і т.д.).

На підприємстві ПАТ «КВАДР» розробка технологічного процесу відбувається в наступній послідовності:

1. Замовник передає пакет робочою документації (креслення деталі, обсяг випуску, основні характеристики і т.д.);
2. якщо немає креслень, то КВ згідно з вихідними даними розробляє робоче креслення готової деталі та креслення або ескіз заготовки і передає пакет робочої документації в ВГТ;
3. отримавши вихідні дані та креслення технолог починає розробку технологічного процесу:
 - а) обирається заготовка;
 - б) обирається типовий технологічний процес, якщо такий існує;
 - в) обирається послідовність зміни розмірів, форм та властивостей заготовки;
 - г) обирається обладнання та інструменти, необхідні для даних змін;
 - д) обираються методи та засоби транспортування;
 - е) обирають людські ресурси для виконання даних робіт;
 - е) проводить процес узгодження та затвердження.

При виборі заготовки подається запит на склад про існування необхідної заготовки, якщо немає підходящої заготовки, проводиться вартісний аналіз і обирається або закупка готових заготовок, або виробництво нових. Заготовки підбираються таким чином, щоб забезпечити найбільш раціональне використання матеріалу, мінімальну трудомісткість одержання заготовок і можливість зниження трудомісткості виготовлення самої деталі. При закупці нових заготовок подається розпорядження в ВМТЗ на придбання партії необхідних заготовок. Вразі виробництва заготовок розробляється технологічний процес на виробництво. Коли заготовка вже обрана перевіряється чи існує підходящий типовий процес. *Типовий технологічний процес* є уніфікованим для найбільш типових деталей, що володіють подібними техніко-конструктивними параметрами. Інженерами високого класу розробляється технологічний процес для типових деталей, а потім,

з їхньою допомогою, становлять робочі технологічні процеси для конкретної деталі. Використання типового технологічного процесу дозволяє спростити розробку тех. процесів, підвищити якість цих розробок, заощадити час і скоротити витрати на технологічну підготовку виробництва. Якщо не існує типового ТП, то розробляється власний ТП. Спочатку обирається послідовність зміни розмірів, форм та властивостей заготовки і яке обладнання для цього потрібне. Обладнання обирається з урахуванням наступних факторів:

- вид обробки;
- точність оброблюваної поверхні;
- розташування оброблюваної поверхні щодо технологічних баз;
- габаритні розміри й маса заготовки;
- продуктивність операції;
- тип виробництва.

Вибір інструменту здійснюється залежно від методу обробки, форми й розмірів оброблюваної поверхні, її точності й шорсткості, оброблюваного матеріалу, заданої продуктивності й періоду стійкості (заміни) інструмента. По можливості використовуються стандартні інструменти. Всі ці дані технолог обирає з існуючих довідників. Також технолог постійно робить запити до виробничих цехів про справність та завантаженість даного обладнання, а також до комірника про наявність необхідного інструменту. Щоб обрати професію та кваліфікацію працівників, необхідних для виконання даних робіт, технолог робить запит до відділу нормування. Також відділ нормування розраховує величину штучного часу для кожної операції та штучно-калькуляційного часу. По закінченню розробки технологічний процес проходить процес узгодження та затвердження. Перевіряється наявність всієї документації та всіх підписів.

На підприємстві використовується наступне апаратне забезпечення:

Комп'ютери – 8 шт.(у ВГТ, ТД, на складі і в виробничих цехах)

наступної конфігурації:

- CPU Intel Core i3 2400 MHz;
- RAM 2GB;
- HDD 320GB;
- MONITOR 19” LCD;

Комп'ютери – 2 шт.(у КВ) наступної конфігурації:

- CPU Intel Core i5 2500 MHz;
- RAM 4GB;
- HDD 1000 GB;
- MONITOR 23” LCD;

Використовуване ПЗ включає програми для вирішення наступних задач:

- a) Функціонування прикладного ПЗ;
- b) Розробка креслень;
- c) Підбор матеріалів та обладнання;
- d) Складський облік.

Розглянемо ці види більш докладно:

Операційна система - на підприємстві у відділах: ТД, ВГТ,КВ, ВМТЗ та на складі, виробничих підрозділах, в бухгалтерії використовується ОС Windows 10 та Windows 7 і при даній організації інформаційної структури можливостей цієї ОС досить для рішення виконуваних задач.

Розробка креслень – виконується у КБ в програмі AutoCad і являє собою сукупність креслень деталей, вузлів, заготовок та ін.

Підбір обладнання та матеріалів – виконує ВМТЗ, який використовує ПЗ MS Excel 2019 і являє собою сукупність таблиць з різноманітними видами матеріалів та обладнання.

Складський облік – виконується на складі комірником з використанням ПЗ MS Excel 2019 і являє собою таблицю наступного змісту:

Лист, що містить інформацію про найменування, кількість, вид і наявності матеріалів, обладнання та інструментів використовуваних у процесі виробництва;

1.1.3. Опис недоліків існуючої системи

Якщо ретельно вивчити існуючий процес створення технологічної документації, то з неї можна виділити наступні недоліки:

- низька швидкість обміну інформацією;
- багато інформації технолог повинен тримати в пам'яті, що робить його важко замінимим;
- в багатьох ситуаціях технологу необхідно звертатись до довідників, що також забирає багато часу;
- технолог невчасно дізнається про поломки існуючого обладнання, що може призвести до збою усього графіка виробництва;
- не враховується навантаження на обладнання;
- не існує єдиної бази типових технологічних процесів, що могло б значно прискорити процес розробки ТП для схожих між собою ТП;
- не існує шаблонів для однотипних операцій;
- технолог не має доступу до баз відділу кадрів;
- розробити ТП здатні тільки висококваліфіковані спеціалісти з великим досвідом роботи.

1.1.4. Обґрунтування необхідності удосконалення

У результаті ґрунтовного аналізу наявних етапів розроблення технологічного процесу та формування технологічної документації встановлено, що доцільним напрямом інформатизації підприємства є розроблення і впровадження системи технологічної підготовки операцій у складі інформаційної системи підприємства, що забезпечить:

- зменшити витрати часу на розробку технологічної документації;
- полегшити та зменшити витрати часу на підбір необхідного обладнання, інструменту, матеріалів та інших ресурсів;
- полегшити керівникам контролювати розробку технологічного процесу, що дозволить їм більш ефективно виконувати свої функції;

- знизити ризик помилок у оформленні документації та у коректності вибору певних ресурсів;
- підвищити швидкість процесу обробки інформації і знизити витрати, пов'язані з ним;
- підвищити рівень організації і досвіду робітників в використанні інформаційних технологій, що підготує умови для наступного етапу інформатизації підприємства.

(а) 1.1.5. Цілі, критерії та обмеження створення системи

Метою створення системи технологічної підготовки токарних операцій інформаційної системи машинобудівного заводу є:

- підвищення швидкості та спрощення процесу створення технологічної документації за рахунок автоматизації даного процесу;
- зменшення ризику помилок при прийнятті рішень через невірогідність інформації про функціонування підприємства;
- зменшення витрат на документообіг.

Критеріями для якісної оцінки проекрованої підсистеми обираємо:

- Надійність функціонування підсистеми.
- Можливість підтримки інформації в актуальному стані.
- Швидкість обробки даних.
- Простота використання системи.
- Припустимі фінансові витрати на створення і функціонування системи.

На систему накладаються такі обмеження:

- для мінімізації витрат на переучування персоналу, програмна частина повинна функціонувати у середовищі ОС Windows;
- потрібно використати наявні апаратні засоби.

(b) 1.1.6. Функції та задачі створюваної системи

Основною задачею системи технологічної підготовки операцій є автоматизація та раціоналізація процесу створення технологічної документації. Крім того, система повинна забезпечувати виконання таких задач:

- автоматизувати розробку технологічного процесу;
- автоматизувати випуск технологічної документації;
- автоматизувати контроль коректності вибору існуючих ресурсів;
- автоматизувати побудову та контроль графіку виробництва, з можливістю швидкого реагування з боку керівництва;

Функції створюваної системи:

- розробка технологічного процесу;
- підготовка технологічної документації для розробленого ТП;
- вибір засобів, обладнання, інструменту, що використовуються в даному ТП;
- вибір рівня та кваліфікації працівників для виконання робіт згідно даного ТП;
- обмін інформацією з виробничими підрозділами;
- відображення графіку виробництва;
- відображення графіку випуску робочої документації;

1.1.7. Очікуваний техніко-економічний результат

При введенні в дію створюваної підсистеми інформаційної системи машинобудівного заводу та усуненні вище розглянутих недоліків, очікуються наступні результати:

- підвищення швидкості розробки ТП та створення технологічної документації;
- підвищення ефективності контролюючих дій;

- підвищення ефективності функціонування підрозділів;
- значне зниження вірогідності помилок.

У результаті передбачається істотний техніко-економічний результат: зниження часу обробки інформації, що підвищить пропускну здатність системи і дозволить вивільнити час, що витрачається на формування документації, зниження можливості помилок. Це дозволить підвищити продуктивність та взаємодію відділів без упровадження додаткових потужностей, а отже, приведе до підвищення рентабельності, підвищення технологічного рівня та конкурентноспроможності даного підприємства.

1.1.8. Висновки і пропозиції.

У результаті вивчення структури ПАТ «КВАДР» та результатів функціонування його підрозділів були розглянуті та сформульовані проблеми, розв'язання яких потребує розробки інформаційної системи, що буде реалізовувати поставлені цілі і задачі.

1.2 Розробка концепції системи технологічної підготовки токарних операцій

1.2.1. Опис результатів вивчення об'єкта автоматизації

У результаті аналізу діючої системи розробки технологічного процесу та технологічної документації були відзначені істотні недоліки, які знижують ефективність роботи як відділу головного технолога так і всього підприємства в цілому, сформовані вимоги до системи, а також до виконуваних нею функцій і покладених на неї задач. Система повинна:

- розробляти технологічний процес;
- формувати технологічну документацію для розробленого ТП;
- раціонально обирати існуюче обладнання, що буде використовуватися в виробничому процесі;
- мати доступ до баз даних складу та відділу кадрів;
- відображати графік випуску робочої документації;
- робити замовлення в ВМТЗ на закупівлю матеріалів, механізмів та обладнання.

Грунтуючись на результатах вивчення об'єкта автоматизації необхідно провести порівняльний аналіз можливих концепцій системи й оцінити переваги і недоліки кожної з них на підставі різних критеріїв і вимог користувача.

1.2.2. Опис і оцінка переваг і недоліків розроблених альтернативних варіантів концепції створення інформаційної системи

Виходячи з аналізу функцій системи і вимог до системи, можна розробити концепцію.

Основними компонентами системи є:

- *система управління базами даних* – ядро будь-якої інформаційної системи;
- *серверна платформа і серверне програмне забезпечення* – один з основних факторів ідеології організації інформаційної системи;
- *клієнтська платформа і клієнтське програмне забезпечення*;
- *мережеве середовище*;
- *засоби розробки*.

Від вибору перерахованих компонентів багато в чому залежить ефективність системи. Очевидно, що для рішення подібної задачі може існувати кілька варіантів концепції.

Таким чином, ґрунтуючись на ступені поширеності і необхідних функціональних можливостях, визначаємо варіанти використовуваної операційної системи (таблиця 1.1).

Таблиця 1.1 – Варіанти операційних систем

ОС (сервер)	ОС (клієнт)
MS Windows 2018 Server	MS Windows 7 Professional
MS Windows 2018 Server	MS Windows 10
Ubuntu Linux 20.04	Ubuntu Linux 20.04
Ubuntu Linux 20.04	MS Windows 10

Існує кілька варіантів серверів БД, виходячи з обраних операційних систем і функціональних можливостей сервера БД.

Їх варіанти представлені в таблиці 1.2:

Таблиця 1.2 - Варіанти серверів БД

Сервер БД	Платформа	Використання, що рекомендується
InterBase	NT, Windows	Середні підприємства
FireBird	Unix, Windows	Середні підприємства
MySQL	Unix, Windows	Середні і дрібні підприємства

З огляду на варіанти ОС і серверів БД, найбільш доцільно використовувати наступні засоби розробки :

- Embarcadero Delphi;
- IntelliJ IDEA + Java (JDK21).

З огляду на сумісність перерахованих вище варіантів для різних компонентів системи можна скласти варіанти загального рішення для проектованої системи:

- 1) MS Windows 2018 Server, InterBase, MS Windows 10 Professional, Embarcadero Delphi.
- 2) MS Windows 2018 Server, FireBird, MS Windows 10, IntelliJ IDEA + Java.
- 3) Ubuntu Linux 20.04, MySQL 8, Ubuntu Linux 20.04, IntelliJ IDEA + Java.

Крім того, кожен варіант концепції системи враховує технічне забезпечення.

Технічне забезпечення, при якому система буде ефективно працювати, складається з:

CPU Intel Core i5, RAM 32 Гб, HDD 1000 Гб у якості сервера;
 CPU Intel Core i3, RAM 16 Гб, HDD 500 Гб у якості робочої станції;
 мережної карти (100 Мбіт/сек) для локальної мережі Ethernet.

З урахуванням існуючих у даний момент на підприємстві технічного забезпечення пропонується два варіанти технічного забезпечення:

- 1) модернізація наявного технічного забезпечення до необхідного рівня;
- 2) встановлення нового технічного забезпечення необхідного рівня.

Таким чином, отримано 6 варіантів концепції системи з урахуванням програмного і технічного забезпечення.

Варіанти концепції інформаційної системи наведені в таблиці 1.3:

Таблиця 1.3 - Варіанти концепції системи

	Програмне забезпечення	Технічне забезпечення
1	MS Windows 2018 Server, InterBase, MS Windows 10 Professional, Embarcadero Delphi.	Модернізація наявного технічного забезпечення до необхідного рівня (CPU Intel Core i5, RAM 32 Гб, HDD 1000 Гб у якості сервера; CPU Intel Core i3, RAM 16 Гб, HDD 500 Гб у якості робочої станції;)
2	MS Windows 2018 Server, FireBird, Windows 10 Professional, IntelliJ IDEA + Java (JDK21).	
3	Ubuntu Linux 20.04, MySQL 8, Ubuntu Linux 20.04, IntelliJ IDEA + Java (JDK21).	
4	MS Windows 2018 Server, InterBase, MS Windows 10 Professional, Embarcadero Delphi.	Встановлення нового технічного забезпечення необхідного рівня (CPU Intel Core i5, RAM 32 Гб, HDD 1000 Гб у якості сервера; CPU Intel Core i3, RAM 16 Гб, HDD 500 Гб у якості робочої станції;)
5	MS Windows 2018 Server, FireBird, Windows 10 Professional, IntelliJ IDEA + Java (JDK21).	
6	Ubuntu Linux 20.04, MySQL 8, Ubuntu Linux 20.04, IntelliJ IDEA + Java (JDK21).	

Тепер необхідно провести аналіз переваг і недоліків запропонованих варіантів концепцій і оцінити кожний з них з погляду досягнення поставлених перед системою цілей, задоволення висунутим критеріям і обмеженням.

Після консультацій з експертами та аналізу корпоративних вимог до програмного забезпечення, а також враховуючи фінансовий стан підприємства,

було прийняте рішення обрати Варіант №2, як такий що задовольняє вимоги до функціональності системи при використанні наявного програмного забезпечення

1.2.3. Очікувані результати й ефективність реалізації обраного варіанта концепції системи

Тому що обраний варіант реалізації задовольняє всім основним вимогам до системи, очікуються ефективні і високі результати при рішенні поставлених задач:

- створення єдиної комп'ютерної мережі підприємства;
- забезпечення надійного контролю цілісності даних;
- підвищення ефективності виконання програм за рахунок використання потужності сервера з застосуванням технології “клієнт-сервер”;
- використання існуючого обладнання у якості клієнтських місць.

1.2.4. Орієнтовний план реалізації обраного варіанта концепції системи

Орієнтований план реалізації обраного варіанта концепції створення системи зображено в таблиці 1.4:

Таблиця 1.4 - Орієнтований план реалізації обраного варіанта концепції системи

Етапи розробки	Терміни виконання
Технічне завдання (визначення й аналіз вимог)	01.03.2026 – 21.03.2026
Ескізне проектування (розробка специфікацій)	22.03.2026 – 15.04.2026
Технічне проектування	16.04.2026 – 15.05.2026
Робоче проектування (кодування, налагодження, тестування)	16.05.2026 – 31.05.2026
Впровадження	1.06.2026 – 1.17.2026

1.2.5. Умови приймання системи

Приймання створеної системи здійснюється представниками виконавця і замовника відповідно до технічного завдання. Складається двосторонній акт, що стає підставою для розрахунків.

1.3 Розробка технічного завдання

1.3.1 Загальні відомості

Необхідно розробити систему технолога інформаційної системи, призначену для використання технологом під час розроблення технологічного процесу, зокрема для вибору обладнання, інструментів, засобів транспортування та інших ресурсів, а також для формування робочої технологічної документації.

Назва розробки: «Система технологічної підготовки токарних операцій на машинобудівному підприємстві».

1.3.2 Призначення і мета створення системи

Призначенням системи є автоматизація функцій, які виконують технологи підприємства під час розроблення технологічного процесу.

Метою створення системи є підвищення ефективності та якості роботи технологів, а також зниження ймовірності помилок під час розроблення технологічного процесу.

1.3.3 Вимоги до інформаційного забезпечення

Створювана система повинна бути відкритою, тобто мати властивості мобільності та інтероперабельності.

Система технолога повинна підтримувати розподілену дворівневу архітектуру «клієнт-сервер» і мати зручний інтерфейс, що забезпечує доступ до інформації через локальну мережу підприємства.

Система, що розробляється як складова інформаційної системи підприємства, повинна забезпечувати захист даних шляхом автентифікації та

ідентифікації користувачів, захисту каналів зв'язку, зокрема шифрування даних, а також фізичного захисту обладнання.

Необхідно використовувати стандартні можливості СУБД із метою підтримки переносимості.

1.3.4 Вимоги до програмного забезпечення

Інформаційна система включає сервер, СУБД та клієнтські комп'ютери.

Вимоги до програмного забезпечення сервера:

- Операційна система: MS Windows 2018 Server

Вимоги до програмного забезпечення СУБД:

- СУБД FireBird

Вимоги до програмного забезпечення клієнтських комп'ютерів АРМ:

- Будь-яка операційна система з можливістю установки віддаленого мережного з'єднання і графічним інтерфейсом.

1.3.5 Вимоги до технічного забезпечення

Вимоги до технічного забезпечення сервера:

- мікропроцесор класу Intel Core i5 2.5Гц вище;
- оперативна пам'ять: RAM 32 Гб і більше;
- жорсткий диск: 1000 Гб і більше;
- мережна карта Ethernet, що підтримує швидкості передачі 100 Мбіт/сек.

Вимоги до технічного забезпечення клієнтських комп'ютерів:

- мікропроцесор класу Intel Core i3 і вище;
- оперативна пам'ять: RAM 16 Гб і більше;
- жорсткий диск: 500 Гб і більше;
- мережна карта Ethernet .

В якості мережного середовища необхідно використовувати дроти типу «кручена пара».

1.3.6 Вимоги до документації

Необхідна наявність наступних документів:

- Загальний опис системи.
- Посібник користувача.
- Посібник адміністратора з установки програмного забезпечення.
- Програма і методика випробувань.
- Текст програми.

1.3.7 Порядок контролю і прийому системи

Перевірка документації системи технолога інформаційної системи здійснюється замовником з метою встановлення відповідності або невідповідності системи технічному завданню.

Тестування системи технолога виконує розробник у процесі її створення без безпосередньої взаємодії із замовником. Процес тестування передбачає експлуатацію системи в контрольованих умовах та аналіз отриманих результатів. При цьому перевіряється робота системи як із коректними, так і з помилковими даними та подіями. Тестування розподілених систем ускладнюється значною кількістю взаємозв'язків. Кожна залежність, компонент, база даних, код графічного інтерфейсу, програмне забезпечення середнього рівня та мережева інфраструктура повинні бути перевірені не лише на коректність роботи, а й на сумісність між собою в різних конфігураціях.

Найкраще для тестування розподілених систем застосовувати метод «знизу-нагору». Спочатку розроблювач повинний перевірити кожен компонент (елемент системи) по окремості; потім – протестувати взаємодію компонента з іншими компонентами системи на автономному комп'ютері; і тільки потім – перевірити роботу всієї системи в розподіленому середовищі. Усі ці види тестування виконуються розроблювачем без участі замовника.

Приймання системи виконується замовником. Система вважається придатною, якщо вона задовольняє всім пунктам даного технічного завдання (постановки задачі). Замовник і/або його представники самостійно проводять тестування усіх функцій системи методом «чорної шухляди» у присутності розроблювача. При виявленні помилок у роботі системи складається їхній перелік і обговорюється термін їхнього виправлення розроблювачем. Після цього замовник проводить повторне тестування.

Після виправлення помилок або у випадку, якщо помилки не були знайдені, а також при відповідності системи встановленим у технічному завданні вимогам, підписується акт приймання системи.

Висновок до розділу 1

У першому розділі було проаналізовано предметну область, організаційну структуру підприємства та чинний порядок розроблення технологічного процесу і технологічної документації. Визначено основні недоліки наявної системи, зокрема значні витрати часу, недостатню оперативність обміну інформацією, відсутність єдиної бази типових технологічних процесів і підвищений ризик помилок під час підготовки документації. На підставі проведеного аналізу обґрунтовано необхідність створення системи технологічної підготовки токарних операцій, сформульовано її мету, функції, обмеження та очікуваний техніко-економічний результат. Отримані висновки є основою для подальшого розроблення концепції, технічного завдання та проєктних рішень системи.

2. Розробка проєктних рішень щодо системи технологічної підготовки токарних операцій

2.1. Загальносистемні рішення

Схема організаційної структури

Схема організаційної структури підприємства зображена на рисунку 1.1.

Користувачі системи та потоки інформації, якою вони будуть обмінюватись зображено на рисунку 2.1.

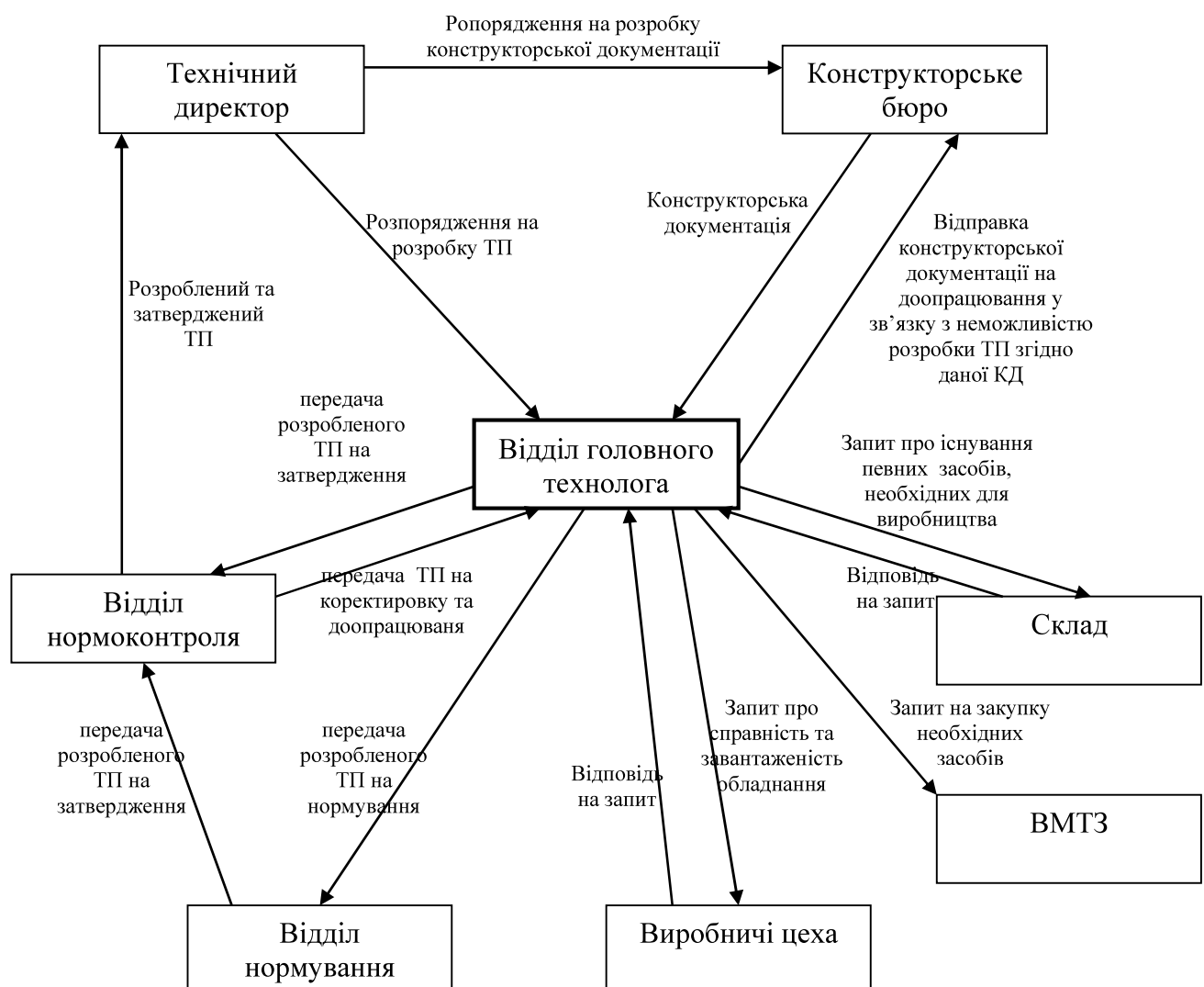


Рисунок 2.1 – Організаційно-функціональна структура

Схема функціональної структури

Для уточнення функціональності інформаційної системи, виявлення схеми функціональної структури, визначення входних у її склад підсистем (модулів) і вивчення динаміки роботи системи в цілому необхідно побудувати функціональну модель.

Функціональна модель описує обчислення в системі. Вона показує, яким чином вихідні дані обчислюються по входнім даним, не розглядаючи порядок чи спосіб реалізації обчислень. Функціональна модель складається з набору діаграм потоків даних, які показують потоки значень від зовнішніх входів через операції і внутрішні сховища даних до зовнішніх виходів.

Основним засобом моделювання функціональних вимог до системи, що проектується є діаграми потоків даних (Data Flow Diagrams). З їхньою допомогою ці вимоги розбиваються на функціональні компоненти і представляються у вигляді мережі, зв'язаної потоками даних. Головна мета таких засобів – продемонструвати, як кожен процес перетворить свої входні дані у вихідні, а також, виявити відносини між цими процесами.

DFD використовує чотири важливих елементи:

- *Роботи.* Роботи в DFD позначають функції або процеси, що обробляють і змінюють інформацію. Роботи представлені на діаграмах у вигляді прямокутників.
- *Стрілки.* Стрілки йдуть від об'єкта джерела до об'єкта приймача, позначаючи інформаційні потоки в системі документообігу.
- *Зовнішні сутності.* Зовнішні сутності вказують на місце, організацію або людину, що беруть участь у процесі обміну інформацією із системою, але розташовуються за рамками цієї діаграми.
- *Сховища даних.* Сховища даних являють собою власне дані, до яких здійснюється доступ, ці дані також можуть бути створені або змінені роботами. На одній діаграмі може бути присутнім кілька копій того самого сховища даних.

Модель системи визначається як ієрархія діаграм потоків даних (DFD), яка описує асинхронний процес перетворення інформації від її вводу у систему до видачі користувачу. Діаграми верхніх рівнів ієрархії (контекстні діаграми) визначають основні процеси або підсистеми з зовнішніми входами та виходами. Вони деталізуються за допомогою діаграм нижнього рівня. Така декомпозиція продовжується, створюючи багаторівневу ієрархію діаграм, до того часу, поки не буде досягнутий такий рівень декомпозиції, на якому процес стає елементарним і деталізувати його далі неможливо. Перед побудовою контекстної DFD необхідно проаналізувати зовнішні події (об'єкти), які впливають на функціонування даної системи. Ці об'єкти взаємодіють з інформаційною системою шляхом інформаційного обміну з нею.

Наша система складається з двох частин: для користувачів та для адміністратора системи.

Частина, в якій працює користувач, дозволяє йому заносити дані по задачах, що стоять перед ним; продивлятися дані та результати попередніх задач по проекту.

Адміністратор має право вносити будь які зміни в базу даних, а саме відкривати або видаляти нові проекти, вносити та видаляти задачі, призначати або змінювати учасників проекту та координувати їх роботу.

Усі необхідні дані будуть зберігатися у файлах бази даних.

Розглянуту програму можна представити у виді процесу „Робота системи” взаємодіючого із зовнішніми сутностями („Користувач” і „Адміністратор”).

Взаємодія елементів відображена на контекстній діаграмі (Рисунок 2.2). Опис потоків даних приведено нижче.

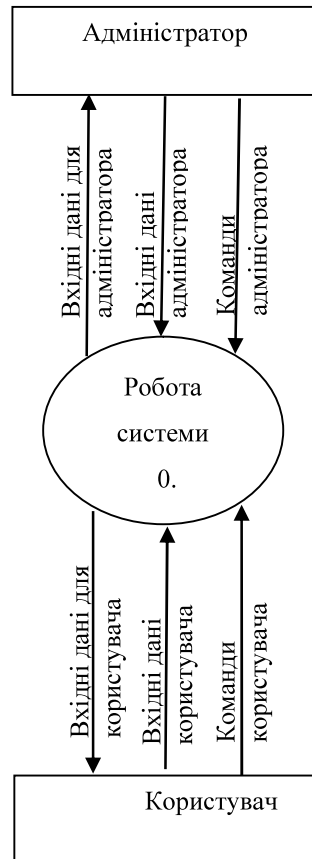


Рисунок 2.2 – Контекстна діаграма потоків даних системи (DFD)

Потік „Вхідні дані користувача” являє собою набір даних, які містять ім’я та пароль. Дані зберігаються у вигляді записів в серверній частині програмного комплексу, побудованій на базі FireBird, які викликаються за допомогою запитів.

Потік „Вхідні дані для користувача” являє собою набір даних, що виводяться на екран монітора і призначені для перегляду клієнтом. Цей набір даних може містити в собі: список всіх проектів, на які назначений користувач, список задач, що виконуються в даному проекті.

Потік „Команди користувача” являє собою набір команд, що вводяться з клавіатури клієнтом. Цей набір команд може містити в собі: занести дані, очистити форму, продивитися дані та інші.

Потік „Вхідні дані адміністратора” являє собою набір даних, що вводяться з клавіатури клієнтом. Цей набір даних може містити в собі: інформацію о користувачах, інформацію про задачі, інформацію про проекти тощо.

Потік „Вхідні дані для адміністратора” являє собою набір даних, що виводяться на екран монітора і призначені для його перегляду. Цей набір даних може містити в собі: список всіх проектів, на які назначений користувач, список задач, що виконуються в даному проекті.

Потік „Команди адміністратора” являє собою набір команд, що вводяться з клавіатури адміністратором. Адміністратор має права доступу до користувачів, а саме він може заносити нових користувачів, давати їм права технолога або іншого користувача проекту та пароль кожному з них. В свою чергу технолог вже може створити проект, занести задачу, змінити задачу, занести проект, змінити проект, призначити користувачів на проект та інші.

Даний процес може бути деталізований за допомогою контекстної діаграми першого рівня. Як зображено на рисунку 2.3 ця діаграма містить 2 процеси:

- Обслуговування користувача;
- Обслуговування адміністратора.

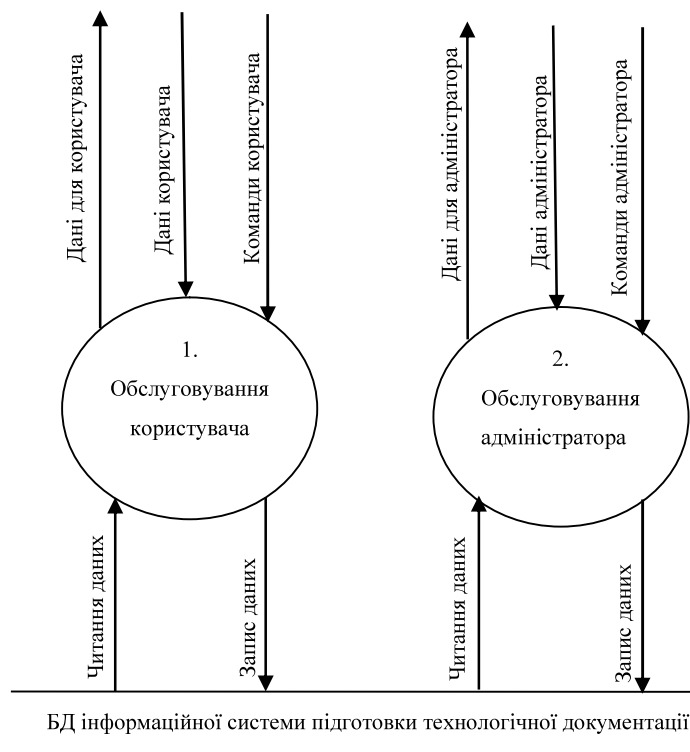


Рисунок 2.3 – Деталізація процесу 0 „Робота з системою”

Процес 1 „Обслуговування користувача” виконує дії, зв’язані з перевіркою правильності введення ім’я та паролю користувача, внесенням змін до записів (результатів задач), затвердженням даних, збереженням даних, поданням користувачу необхідної інформації.

Вхідні потоки:

- Команди користувача;
- Дані користувача;
- Читання даних – дані зчитуються з БД.

Вихідні потоки:

- Дані для користувача;
- Запис даних – дані записуються в БД.

Процес 2 „Обслуговування адміністратора” виконує дії, зв’язані з обслуговуванням і налаштуванням бази даних, внесення змін до списку задач, внесення змін до списку користувачів, внесення змін до списку проектів, переглядом та затвердженням необхідної інформації.

Вхідні потоки:

- Команди адміністратора;
- Дані адміністратора;
- Читання даних – дані зчитуються з БД.

Вихідні потоки:

- Дані для адміністратора;
- Запис даних – дані записуються в БД.

Зробимо декомпозицію процесу 1 „Обслуговування користувача”.

Процес 1 складається з трьох процесів:

- Реєстрація в системі (вхід до системи);
- Отримання списку задач;
- Занесення результатів виконання задач.

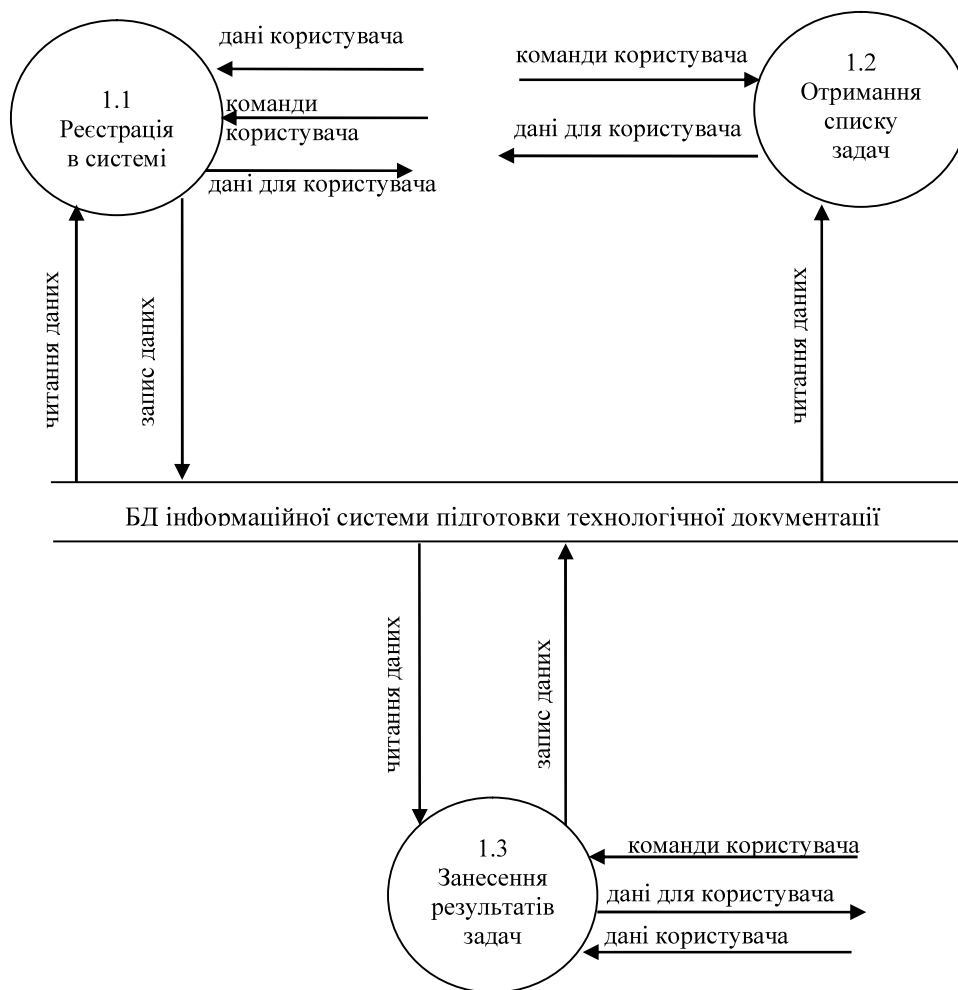


Рисунок 2.4 – Деталізація процесу 1 „Обслуговування користувача”

Процес 1.1 „Реєстрація в системі” виконує дії, зв’язані з отриманням, введенням і передачею даних необхідних для реєстрації в підсистемі.

Вхідні потоки:

- Команди користувача (ввести, відмінити);
- Дані від користувача – інформація, яка буде перетворена системою;
- Читання даних – зчитування даних з БД

Вихідні потоки:

- Дані для користувача (інтерфейс);
- Запис даних – дані передаються для порівняння з записами БД.

Процес 1.2 „Отримання списку задач” виконує дії по отриманню та виведенню для користувача списку задач, що стоять перед ним.

Вхідні потоки:

- Команди користувача;
- Читання даних – зчитування даних з БД

Вихідні потоки:

- Дані для користувача (інформація про задачі).

Процес 1.3 „Занесення результатів виконання задач” здійснює дії по відображенню результатів, що вводяться користувачем та запису їх до бази даних.

Вхідні потоки:

- Команди користувача (занести, очистити і т.п.);
- Дані від користувача (дані про результат виконання задач);
- Читання даних – зчитування даних з БД

Вихідні потоки:

- Дані для користувача;
- Запис даних – дані записуються до БД.

Процес 2.4.1 „Завантаження користувача” виконує дії по завантаженню користувача з бази даних для роботи з ним і передає керування на процес 2.4.2 „Робота з користувачем”.

Вхідні потоки:

- Команди адміністратора;
- Команди процесу 2.4.2 „Робота з користувачем”;
- Читання даних з БД

Вихідні потоки:

- Дані для адміністратора;
- Дані для процесу 2.4.2 „робота з користувачем”

Процес 2.4.2 „Робота з користувачем” виконує дії по зміні, додаванню, перегляду і т.п. задачі, що завантажений, а також передає команди на завантаження проекту процесу 2.4.1 „Завантаження користувача”.

Вхідні потоки:

- Дані від процесу 2.4.1 „Завантаження користувача”;
- Команди адміністратора;
- Дані адміністратора;
- Читання даних з БД

Вихідні потоки:

- Команди процесу 2.4.1;
- Дані для адміністратора;
- Запис даних в БД

Опис функцій, що автоматизуються

У ході розробки АРМ технолога буде автоматизовані наступні функції:

- автоматизація роботи працівників відділу головного технолога;

Розглянемо цю функцію більш детально.

Функцій ВГТ:

- розробка технологічних маршрутів виготовлення виробу на основі його конструкторської документації, їхнє узгодження й твердження;
- підготовка заявок на виготовлення відсутніх заготівель;
- підготовка замовлень на розробку (зміна) оснащення і їх затвердження головного технолога;
- підготовка даних про потребі в основних і допоміжних матеріалах для виготовлення виробу;
- розробка маршрутно-операційних й операційних техпроцесів виготовлення (зборки) елементів виробу, при цьому вибір:
 - устаткування для обробки поверхонь деталей, що виготовляють;
 - інструмента для обробки розглянутих поверхонь.
- узгодження розроблених техпроцесів з технологами-суміжниками, що беруть участь у розробці технологічної документації на даний виріб;
- передача розроблених техпроцесів у підрозділ нормування;
- узгодження, затвердження розроблених техпроцесів й їхня передача в архів технологічної документації;
- підготовка повідомлень про зміну техпроцесів, узгодження їх з підрозділом нормоконтроля, затвердження й передача в архів технологічної документації.

Опис постановки задачі

На підставі аналізу вимог і обмежень, а також попередніх концепцій реалізації підсистеми, сформулюємо опис постановки задачі. Необхідно розробити АРМ технолога, що займається підготовкою токарних і фрезерних робіт інформаційної системи підготовки технологічної документації ПАТ «КВАДР», що буде задовольняти перерахованим вище функціональним вимогам. Необхідно також забезпечити надійність збереження і безпеку доступу до АРМ. Підсистема повинна обслуговувати одночасну безперебійну роботу 10 користувачів, причому зміни, внесені керівниками підрозділів і директорами повинні бути доступні всім іншим користувачам негайно.

Програмне забезпечення повинне бути побудоване на платформі Microsoft Windows 2018 Server / Windows 10 / FireBird / Java + IntelliJ IDEA і реалізовувати графічний інтуїтивно зрозумілий інтерфейс користувача. Апаратне забезпечення повинне максимально повно охоплювати існуючу комп'ютерну техніку на підприємстві.

Загальний опис системи

Система, що розробляється, являє собою інструмент розробки технологічного процесу та створення технологічної документації, побудований на основі інформаційної системи ПАТ «КВАДР». Основними функціональними характеристиками системи є:

- ведення загальної інформаційної бази з розподіленням доступом;
- авторизований доступ до даних на основі прав доступу;
- негайне відображення змін, внесених з будь-якого робочого місця;
- можливість введення і редагування планів робіт і оперативного контролю ходу їхнього виконання;
- формування необхідних вихідних документів.

Система реалізує наступні функції:

- розробка конструкторської документації;
- розробка технологічного процесу;

- підготовка технологічної документації для розробленого ТП;
- розробка технологічних процесів для окремих технологічних операцій;
- вибір необхідних матеріалів і устаткування для даного ТП;
- прив'язка необхідних матеріалів і устаткування до виконуваним підрозділом роботам;
- вибір професії та кваліфікації для виконання робіт згідно ТП;
- формування замовлень на закупку необхідних матеріалів, устаткування та ін.;

Система має звичний графічний інтерфейс користувача, що знижує витрати на навчання персоналу, і побудована з максимальним використанням існуючого на підприємстві в даний момент устаткування і програмного забезпечення.

Система складається з наступних підсистем:

- підсистема розробки конструкторської документації;
- підсистема розробки технологічних маршрутів;
- підсистема оформлення заявок та запитів;
- підсистема передачі повідомлень між користувачами системи;
- підсистема розробки ТП для токарних операцій;
- підсистема розробки ТП для фрезерних операцій;
- підсистема розробки ТП для термічних операцій;
- підсистема розробки ТП для складальних операцій;
- підсистема нормування ТП;
- підсистема проходження процесу затвердження розробленого ТП.

Впровадження автоматизованої підсистеми підготовки токарних і фрезерних робіт інформаційної системи підготовки технологічної документації ПАТ «КВАДР» дозволить:

- підвисити оперативність надання інформації;
- ліквідувати зайве дублювання інформації;
- підвищити рівень контролю за виконанням робіт.

З рисунка 2.12 ми бачимо, що значна частина робіт по технологічній документації для нової деталі є стандартною і може бути автоматизованою, бо зводиться до визначення учасників проекту, та своєчасного надсилання їм запитів про виконання тих чи інших (і, як правило, однотипних) задач. Таким чином, автоматизація робіт по створенню технологічної документації для нового виду деталі у відділі головного технолога приведе до зниження термінів розробки проекту.

Захист від несанкціонованого доступу

Захист системи від несанкціонованого доступу реалізовано на двох рівнях – системному рівні і рівні додатка. Системний рівень передбачає можливість входу користувача в систему і запуску програм. З достатнім ступенем надійності такий захист реалізований на сервері системи, який базується на операційній системі Windows 2018 Server. При запуску програм відбувається повторна авторизація користувача, причому його ім'я може не збігатися з ім'ям в операційній системі. Доступ до даних і функцій дозволяється тільки при правильному введенні користувальницького імені і пароля. При будь-яких змінах даних по протоколі зроблених змін можна визначити ким і коли вони були зроблені.

2.2 Опис організації інформаційної бази

Опис концептуальної моделі підсистеми технологічної підготовки токарних операцій

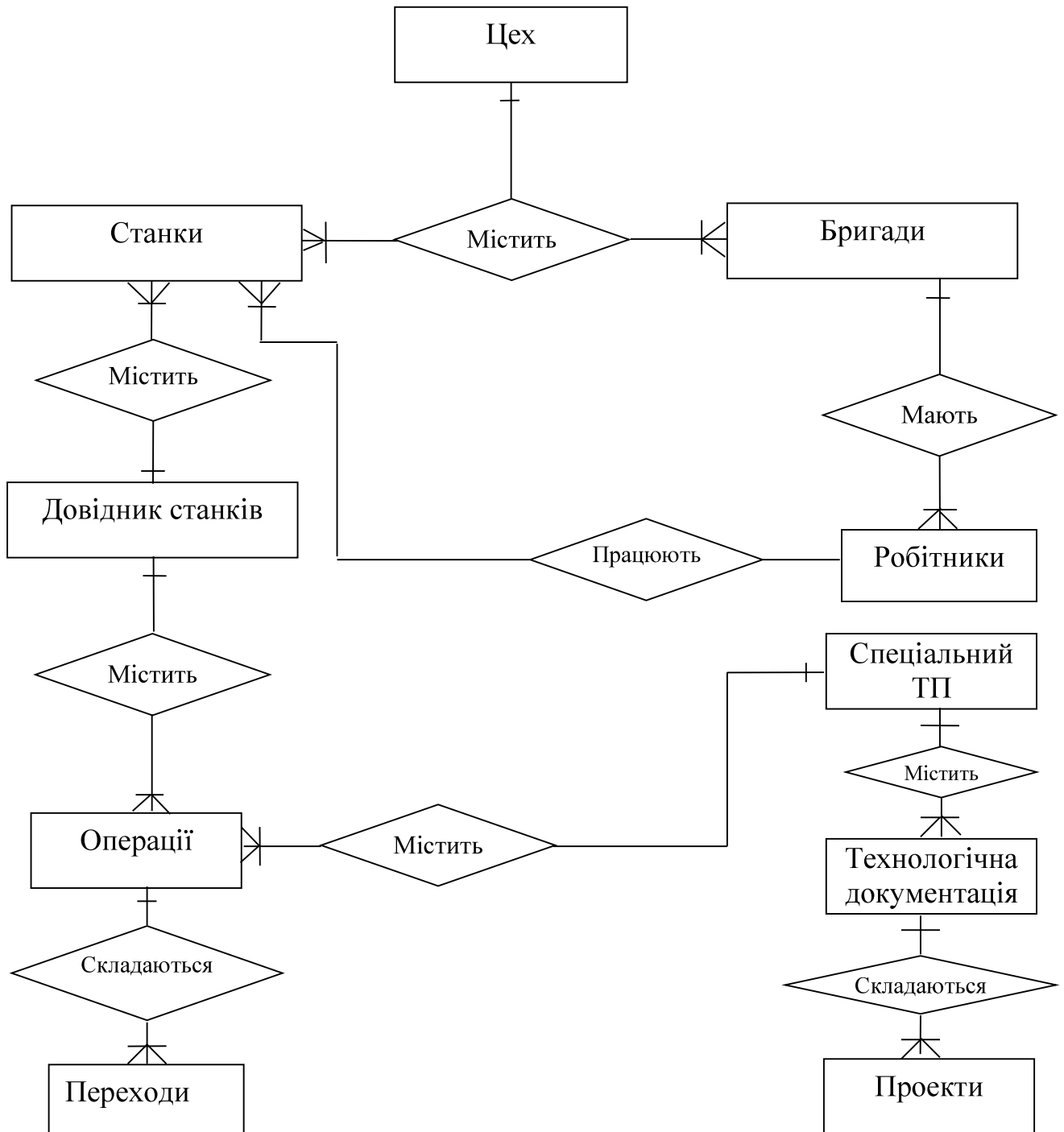


Рисунок 2.5 – Концептуальна модель підсистеми технологічної підготовки токарних операцій

На діаграмі „сутність-зв'язок” зображені наступні типи сутностей і зв'язку між ними:

- Цех містить станки і бригади;
- Бригади мають робітників;
- Робітники працюють на станках;
- Довідник станків містить інформацію про станки;
- Довідник станків містить операції;
- Операції складаються з переходів.

Розробка логічної моделі підсистеми технологічної підготовки токарних операцій

Розглянувши процеси і потоки даних, логічну модель бази даних можна представити з наступних сутностей, що є таблицями (рисунок 2.6).

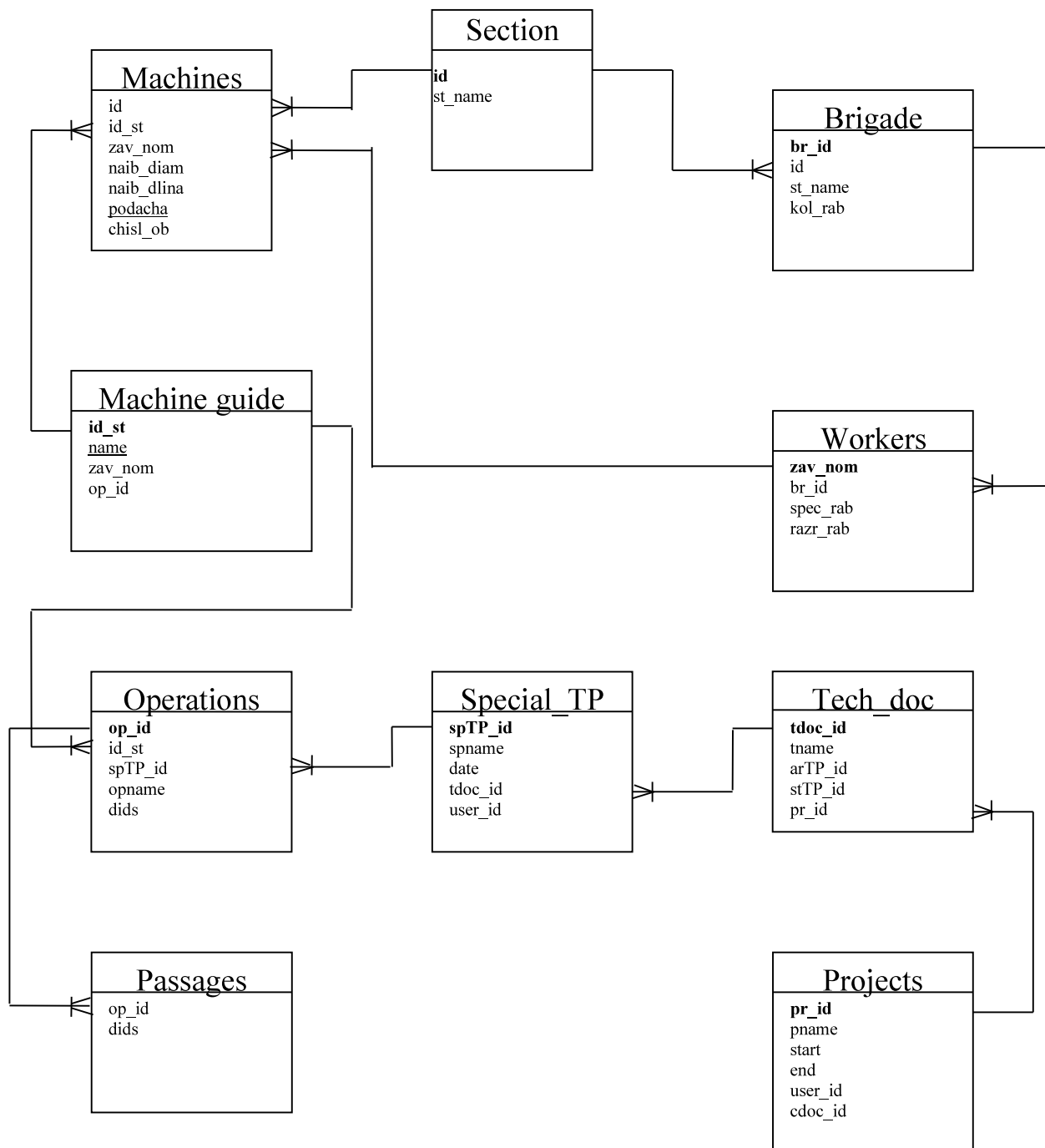


Рисунок 2.6 – Логічна модель даних системи

Таблиця 2.1 - Дані про станки „Machine”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	Номер цеху <u>id</u>	(обов'язкове)
2	Номер станка <u>id st</u>	Номер станка для користувачів
3	Заводський номер станка <u>zav_nom</u>	Заводський номер станка
4	Найбільший діаметр <u>naib_diam</u>	Найбільший діаметр деталі, що оброблюється.
5	Найбільша довжина <u>naib_dlina</u>	Найбільша довжина деталі, що оброблюється
6	Подача <u>podacha</u>	Подача станка
7	Число обертів <u>chisl_ob</u>	Число обертів станка

Таблиця 2.2 - Дані про робочих „Workers”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	Номер цеху <u>id</u>	(обов'язкове)
2	Номер бригади <u>br id</u>	Номер бригади
3	Заводський номер станка <u>zav_nom</u>	Заводський номер станка, на якому працює робочий
4	Спеціальність робочого <u>spec_rab</u>	Спеціальність робочого (професія)
5	Розряд робочого <u>razr_rab</u>	Розряд робочого

Таблиця 2.3 - Дані про цех „Section”

	<i>Поле</i>	<i>Тип</i>
	Номер цеху <u>id</u>	(обов'язкове)
	Назва цеху <u>st_name</u>	Назва цеху

Таблиця 2.4 - Дані про бригади „Brigade”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	Номер бригади <u>br_id</u>	(обов'язкове)
2	Назва бригади <u>st_name</u>	Назва бригади
3	Кількість робочих <u>kol_rab</u>	Кількість робочих у бригаді

Таблиця 2.5 - Дані, що містить „Machine guide”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	Номер станка <u>id_st</u>	(обов'язкове)
2	Назва станка <u>name</u>	Назва станка
3	Заводський номер станка <u>zav_nom</u>	Заводський номер станка

Таблиця 2.6 - Дані про операції „Operations”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	Номер операції <u>op_id</u>	(обов'язкове)
2	Номер станка <u>id_st</u>	Номер станка
3	Номер спец.ТП <u>spTP_id</u>	Номер спец.ТП
5	Назва операції <u>opname</u>	Назва операції
6	Дії <u>dids</u>	Дії, що містить операція

Таблиця 2.7 - Дані про переходи „Passages”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	Номер операції <u>op_id</u>	Номер операції
2	Дії <u>dids</u>	Дії, що містить перехід

Таблиця 2.8 - Структура таблиці „Projects”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	Номер проекту pr_id	(обов'язкове)
2	Назва проекту pname	Назва проекту
3	Початок start	Дата
4	Кінець end	Дата
5	Номер користувача user_id	Номер користувача
7	Номер креслення cdoc_id	Номер креслення

Таблиця 2.9 - Структура таблиці „Tech_doc”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	Номер техн.докум tdoc_id	(обов'язкове)
2	Назва техн.докум tname	Назва
3	Номер ст.ТП stTP_id	Номер
4	Номер ар.ТП arTP_id	Номер
5	Номер проекту pr_id	Номер

Таблиця 2.10 - Структура таблиці „Special_TP”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	Номер сп.ТП spTP_id	(обов'язкове)
2	Назва сп.ТП spname	Назва
3	Дата date	Дата
4	Номер техн.докум tdoc_id	Номер
5	Номер користувача user_id	Номер

Розробка фізичної моделі підсистеми підготовки токарних і фрезерних робіт інформаційної системи підготовки технологічної документації

На наступному етапі, на підставі логічної моделі бази даних, виконується фізична модель (рисунок 2.7).

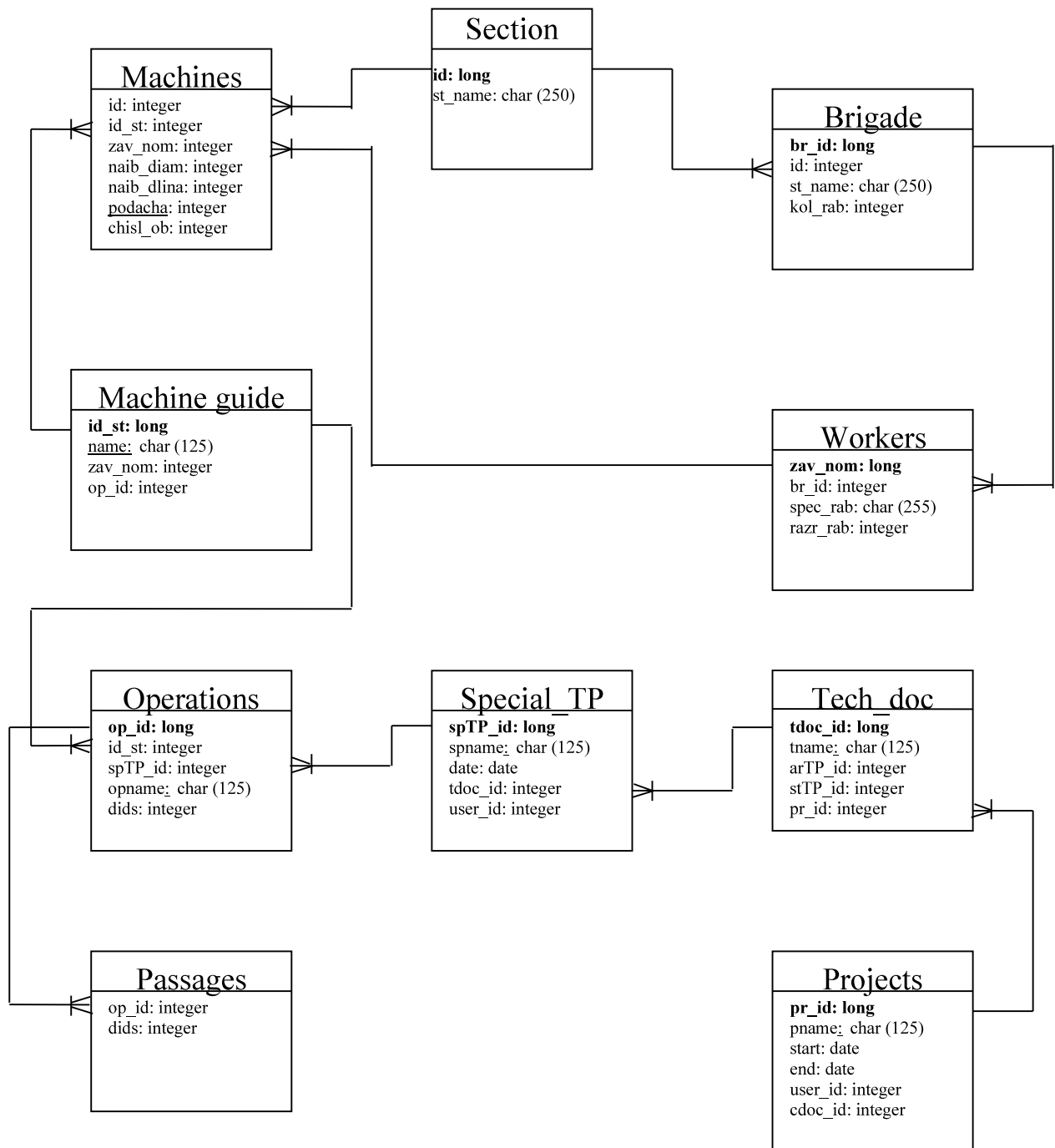


Рисунок 2.7 – Фізична модель даних АРМ технолога машинобудівного заводу

У таблиці 2.11 наведений список таблиць бази даних з описом кожної таблиці.

Таблиця 2.11 - Таблиці бази даних

<i>№</i>	<i>Назва таблиці</i>	<i>Опис таблиці</i>
1	Section	Інформація про цеха
2	Machines	Інформація про станки
3	Machine guide	Довідник станків
4	Operations	Операції
5	Passages	Інформація про переходи
6	Brigade	Інформація про бригади
7	Workers	Інформація про робочих
8	Special_TP	Спеціальний ТП
9	Tech_doc	Технологічна документація
10	Projects	Інформація про проекти

Визначимося з атрибутами таблиць. Кожна таблиця повинна містити поля, що відбивають корисну інформацію, що буде оброблятися. Для гарантованого забезпечення унікальності записів введемо для кожної таблиці штучний первинний ключ (primary key). Обмеження первинного ключа гарантує, що значення, що встановлюються в поле первинного ключа, унікальні усередині таблиці.

Обмеження зовнішнього ключа (foreign key) застосовується для визначення зв'язків між таблицями.

Розглянемо структуру кожної з таблиць бази даних. У стовпці „поле” відображається назва поля в базі даних, у стовпці „тип” відображається тип даних. Використовуються наступні типи даних:

- Integer – цілочисельне значення;
- Long – довге ціле

- Char – символний тип
- Date – поле дати
- Boolean – істина/крив

В таблицях 2.12 - 2.21 наведені структури таблиць бази даних (Section, Machines, Machine guide, Operations, Passages, Brigade, Workers)

Таблиця 2.12 - Структура таблиці „Machine”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	Id	long
2	id_st	integer
3	zav_nom	integer
4	naib_diam	integer
5	naib_dlina	integer
6	podacha	integer
7	chisl_ob	integer

Таблиця 2.13 - Структура таблиці „Workers”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	br_id	integer
2	zav_nom	integer
3	spec_rab	char (255)
4	razr_rab	integer

Таблиця 2.14 - Структура таблиці „Section”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	Id	long
2	st_name	char (250)

Таблиця 2.15 - Структура таблиці „Brigade”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	Id	integer
2	br_id	long
3	st_name	char (250)
4	kol_rab	integer

Таблиця 2.16 - Структура таблиці „Machine guide”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	id_st	long
2	Name	char (125)
3	zav_nom	integer
4	op_id	integer

Таблиця 2.17 - Структура таблиці „Operations”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	op_id	long
2	id_st	integer
3	spTP_id	integer
4	opname	char (125)
5	Dids	integer

Таблиця 2.18 - Структура таблиці „Passages”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	op_id	integer
2	Dids	integer

Таблиця 2.19 - Структура таблиці „ Projects”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	pr_id	long
2	Pname	char (125)
3	Start	date
4	End	date
5	user_id	integer
7	cdoc_id	integer

Таблиця 2.20 - Структура таблиці „ Tech_doc”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	tdoc_id	long
2	Tname	char (125)
3	stTP_id	integer
4	arTP_id	integer
5	pr_id	integer

Таблиця 2.21 - Структура таблиці „Special_TP”

<i>№</i>	<i>Поле</i>	<i>Тип</i>
1	spTP_id	long
2	spname	char (125)
3	Date	date
4	tdoc_id	integer
5	user_id	integer

2.3. Рішення по технічному забезпеченню

Підсистема технологічної підготовки токарних та фрезерних операцій інформаційної системи машинобудівного заводу призначена для спільної роботи співробітників відділу головного технолога, відділу нормоконтроля,

відділу нормування та конструкторського бюро, комп'ютери яких знаходяться в єдиній мережі.

Для передачі даних по мережі використовується протокол TCP/IP.

При проектуванні комп'ютерної мережі розглядається велика кількість питань, не правильне рішення яких приведе до проблеми надалі.

Першим етапом проектування є вибір технології, по якій буде будуватися мережа. У зв'язку з тим, що підприємство за своєю мережною структурою є дуже компактним, то передбачається, що в її складі буде працювати до 10 комп'ютерів. Трафік при роботі з підсистемою – середньоскоросний. Виходячи з цього, вибір пав на технологію Fast Ethernet, пропускна здатність якої є достатньою для роботи розроблюваної підсистеми. Фізична передача даних здійснюється на основі кабелю одного типу, а саме 100BaseT - "кручена пара" 5-ї категорії. Застосування цього кабелю по всій мережі пояснюється низькою його вартістю.

Архітектура Ethernet у даному випадку орієнтована на зіркову топологію. Зіркова топологія складається з множини кабелів (по одному на кожний підключаємий сітьовий пристрій), які підключаються до одного, центрального пристрою. Цей центральний пристрій називається комутатором, який і виконує ширококомовну передачу даних на всі підключені вузли.

Топологія зірки є фізичною топологією, у якій численні вузли приєднані до центрального компонента. Незважаючи на зіркоподібний зовнішній вигляд, така модель монтажного з'єднання, звичайно, реалізується у виді логічної топології загальної шини. Одною з переваг топології зірки є легка локалізація несправностей, а саме при обриві або пошкодженні перестає робити тільки один з вузлів мережі, тому пошук пошкодження значно зменшується. Крім того досить просто додавати і видаляти вузли, змінювати схеми кабельних з'єднань. До недоліків можна віднести необхідність великої кількості кабелів.

У зв'язку з тим, що в даній роботі розроблюється підсистема, а не система, тому не має сенсу змінювати існуючу мережу, яка повністю задовольняє усім вимогам розроблюваній підсистеми. Її також не треба змінювати через додаткові фінансові витрати, які не є необхідними та доцільними у даному випадку для підприємства.

Згідно вищевикладеного було вирішено використовувати вже існуючу мережу на підприємстві.

На рисунку 2.8 представлена існуюча схема комп'ютерної мережі підприємства, реалізована на основі технології Fast Ethernet.

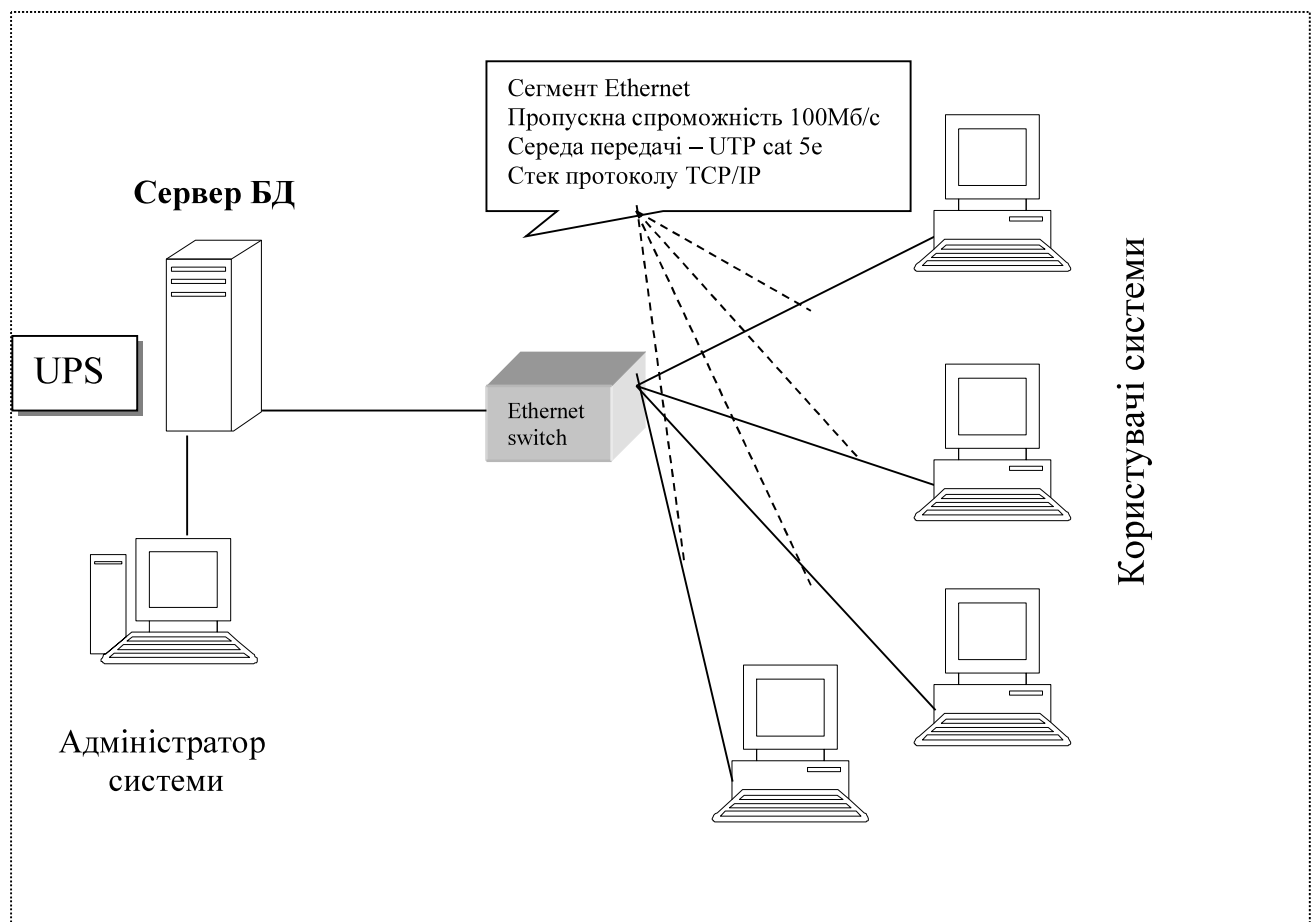


Рисунок 2.8 - схема комп'ютерної мережі ПАТ «КВАДР».

Фізична топологія визначає розташування монтажних з'єднань мережі. За допомогою такої схеми визначається, яким чином з'єднані між собою елементи мережі.

Щоб визначити місце розташування комутуючого обладнання необхідно враховувати планування і розміри будинку. Однак, якщо все комутаційне устаткування – сервери, комутатори і блоки безперебійного харчування для них – буде знаходитися в одному спеціально відведеному приміщенні, для адміністрування мережі це набагато краще.

Вибір устаткування досить суб'єктивна річ, але можна виділити ряд критеріїв, яким повинне відповідати активне мережне устаткування.

- *Пропускна спроможність*

При визначенні необхідної пропускнуєї спроможності варто враховувати не тільки обсяг даних, але і спосіб їхнього використання, частоту і характер запитів, структуру в цілому.

- *Надійність*

Устаткування одного виробника може не працювати з устаткуванням іншого. Так що повніше виправдано вибір пристроїв одного або групи виробників, сумісність яких уже перевірена.

- *Відповідність вимогам підприємства і логічній структурі мережі*

Устаткування повинне як можна точніше проектуватися на структуру мережі підприємства і найбільше повно забезпечувати його потреби.

Один з найскладніших етапів побудови мережі – прокладка кабелю. Від якості кабельної системи залежить ефективність мережі в цілому. Створюючи кабельну систему, треба не тільки свердлити отвори в стінах і перекриттях, але й монтувати коробки по всьому підприємству. При зростанні навантаження на мережу відбувається зниження її продуктивності, тому що будь-яка недостатньо якісна лінія може викликати багаторазову повторну передачу пакетів.

Кожний етап проектування необхідно описувати і постійно підтримувати документацію в актуальному виді. У документації варто описувати наступне:

- логічну схему мережі;
- схему прокладки кабельної підсистеми щодо плану будинку;
- інформацію про підключення користувачів (відповідність мережних

розеток робочим місцям);

- параметри активного устаткування і серверів (паролі, мережні адреси і т.д.).

2.4. Рішення по програмному забезпеченню

Вибір архітектури розробки програмного забезпечення

При проектуванні програмного забезпечення необхідно прийняти рішення про вибір архітектури розробки. Як вже було визначено в аналізі поточного стану проблеми для розроблюваної підсистеми бажано використовувати клієнт-серверну архітектуру. Визначимо її переваги перед іншими архітектурами.



Рисунок 2.9 - Порівняння файл-серверної та клієнт-серверної моделей

Клієнт-серверна архітектура характеризується наявністю двох взаємодіючих самостійних процесів – клієнта та сервера, які виконуються на різних комп'ютерах, обмінюючись даними по мережі. За такою схемою можуть бути побудовані системи обробки даних на основі СУБД, поштові та інші системи. Ми будемо розглядати бази даних. Тому у даному випадку зручніше буде не просто розглянути клієнт-серверну архітектуру, а й порівняти її з іншою – файл-серверною.

В файл-серверній системі дані зберігаються на файловому серверу (наприклад, Novell NetWare или Windows NT Server), а їх обробка здійснюється на робочих станціях, на яких, як правило, функціонує одна з них, так званих „настільних СУБД” - Access, FoxPro, Paradox и т.п.

Додаток на робочій станції „відповідає за все” – за формування користувального інтерфейсу, логічну обробку даних і за безпосереднє маніпулювання даними. Файловий сервер надає послуги тільки самого низького рівня – відкриття, закриття та модифікацію файлів. Основною функцією файлового серверу є збереження, управління та передача файлів даних. Він не обробляє та не змінює збережені та передавані файли. Всі зміни в файлах даних здійснюються з клієнтських робочих станцій. Для цього клієнти зчитують файли даних з файлового сервера, здійснюють необхідні зміни даних та повертають їх назад до файлового сервера. При такій взаємодії відбувається висока напруга на мережу та низька безпека цілісності даних. Тобто така архітектура не є надійною для роботи підсистеми з великою кількістю користувачів та ймовірність цілісності даних є низькою.

В клієнт-серверній системі функціонують (як мінімум) два додатка - клієнт та сервер, які розподіляють між собою ті функції, які в файл-серверній архітектурі цілком виконує додаток на робочій станції. Збереженням та

безпосереднім маніпулюванням даними займається сервер баз даних, в якості якого може виступати FireBird, Microsoft SQL Server, Sybase и т.п..

Формуванням користувацького інтерфейсу займається клієнт, для побудови якого можна використовувати цілий ряд спеціальних інструментів, а також більшість настільних СУБД.

Клієнтом є комп'ютер, який запускає додаток, який знаходиться на іншому комп'ютері - серверу.

Сервер представляє собою спеціалізований комп'ютер, орієнтований на виконання основного об'єму обчислювальних робіт і функцій по керуванню комп'ютерною мережею. Мережний сервер має достатньо високу швидкість та великий об'єм пам'яті. При подібній мережній організації функції робочих станцій зводяться до вводу-виводу інформації і обміну її з мережним сервером.

Таким чином, безпосереднім маніпулюванням даними займається один процес. При цьому, обробка даних відбувається там же, де дані зберігаються – на сервері, що виключає необхідність передачі великих об'ємів по мережі.

З точки зору потреб нашого підприємства розглянемо якості, які привносить клієнт-сервер в розроблювану підсистему підготовки токарних і фрезерних робіт інформаційної системи розробки технологічної документації:

Надійність

Сервер баз даних здійснює модифікацію даних на основі механізму транзакцій, який надає будь-якій сукупності операцій, об'явлених як транзакція, наступні властивості:

атомарність – при будь-яких обставинах будуть виконані або всі операції транзакції, або не виконана ні одна; цілісність даних при завершенні транзакції;

незалежність – транзакції, ініційовані різними користувачами, не втручаються в справи один одного;

стійкість до збою – після завершення транзакцій, її результати вже не зникнуть.

Механізм транзакції, що підтримується сервером баз даних, найбільш ефективний, ніж аналогічний механізм в настільних СУБД. Це пояснюється тим, що сервер централізовано контролює роботу транзакцій. Крім того, в файл-серверній системі збій на одній з робочих станцій може привести до втрати даних та їх недоступності для других робочих станцій, у той час, як в клієнт-серверній системі збій на клієнті, практично, ніколи не виявляється на цілісності даних та їх недоступності для інших клієнтів.

Масштабіруемість

Масштабіруемість – здатність системи адаптуватися до росту кількості користувачів і об'єму баз даних при адекватному зростанні продуктивності апаратної платформи, без зміни програмного забезпечення.

Відомо, що можливості настільних СУБД серйозно обмежені – це п'ять – сім користувачів і 30-50 Мб, відповідно. Цифри представляють собою середні значення, в конкретних випадках вони можуть відхилятися як в ту, так і в іншу сторону. І що найбільш суттєво, ці бар'єри неможливо подолати через нарощування можливостей апаратури.

Системи ж на основі серверів баз даних можуть підтримувати тисячі користувачів та сотні Гб інформації, якщо дати їм відповідну апаратну платформу.

Безпека

Сервер баз даних надає могутні засоби захисту даних від несанкціонованого доступу, які неможливі в настільних СУБД. При цьому, права доступу адмініструються дуже гнучко – до рівня полів таблиць.

Тому адміністратор може бути впевненим, що ніякий дуже розумний користувач не прочитає те, що йому читати не потрібно.

Гнучкість

В додатку, який працює з даними, можна виділити три логічних слою:
- інтерфейсу користувача;

- правил логічної обробки;
- управління даними.

Як вже було відмічено, в файл-серверній архітектурі всі три слою реалізуються в одному монолітному додатку, функціонуючому на робочій станції. Тому зміни в будь-якому із слоїв приводять однозначно до модифікації додатку і наступному поновленню його версій на робочих станції.

У дворівневому клієнт-серверном доданку, який показаний на рисунку 2.19, як правило всі функції по формуванню користувального інтерфейсу реалізуються на клієнті, всі функції по управлінню даними – на серверу, а бізнес-правила можна реалізувати як на серверу так і на клієнті.

Операційна система.

Як показано вище, оптимальним варіантом серверної ОС для реалізації системи є ОС Microsoft Windows 2018 Server – подальший розвиток сімейства серверних операційних систем від Microsoft. Для одержання більш надійної архітектури операційної системи в неї були внесені такі поліпшення, як посилені захист пам'яті ядра ОС, реєстрація драйверів і інші. Windows 2018 сертифікований по стандарті Common Criteria (міжнародний стандарт безпеки).

Використання цієї операційної системи дозволяє максимально реалізувати можливості СУБД FireBird. Але слід зазначити, що СУБД FireBird ефективно працює і з операційною системою Linux.

Вибір мови програмування

Вибір мови програмування цілком залежить від розроблювача. Основні вимоги до вибору мови програмування:

- незалежність від архітектури;

- робота в мережному середовищу;
- гнучкість щодо внесення майбутніх змін.

Приймаючи до уваги наведені вимоги, у якості мови програмування розроблювачем була обрана мова Java SE 21, яка повністю задовольняє вирішенню поставленого завдання.

Основним засобом розробки програмного забезпечення підсистеми є програмне середовище для розробки Java-додатків IntelliJ IDEA. Її характерними рисами є:

- створення додатків для цільової операційної системи (OC Windows 2018 Server / Windows 10);
- ефективний компілятор Java;
- інтегроване середовище розробки, що включає редактор коду, редактор екранних форм, отладчик і засоби управління проектом;

(с) 2.4.4. Інструментальні програмні засоби.

При побудові всієї системи взагалі, та саме АРМ технолога використовуються також наступні програмні засоби:

- СУБД FireBird – система управління базами даних, для збереження і забезпечення доступу до даних системи. Сервер підтримує всі сучасні технології роботи з даними, містить вбудовані засоби обробки транзакцій, а також реалізації збережених процедур, тригерів і інших додаткових можливостей. Також реалізований захищений доступ до даних, засобу їхнього резервування і збереження цілісності. Необхідно також відзначити гарну масштабовуємість сервера, що забезпечує захист інвестицій у програмне забезпечення.

2.5. Опис алгоритмів функціонування системи

Алгоритм роботи основної системи зображений на рисунку 2.10. Система працює з користувачем в діалоговому режимі. Можливість використання функцій системи визначається правами доступу користувача.

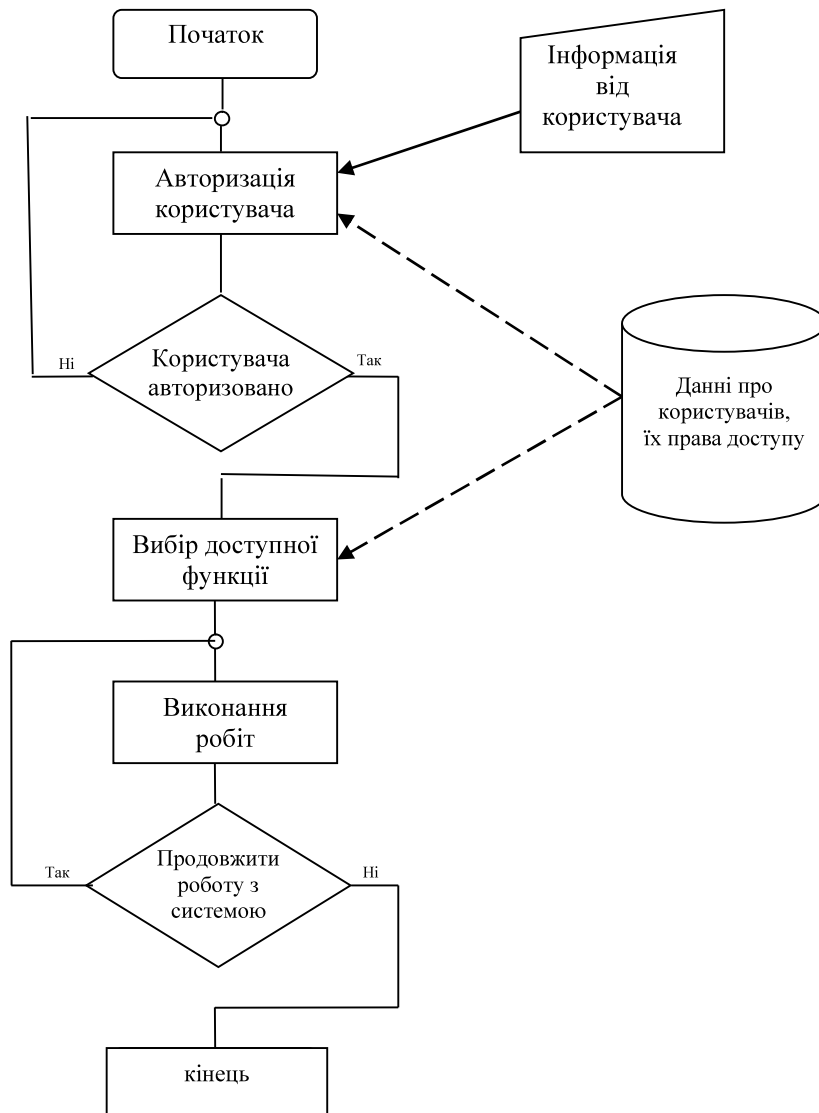


Рисунок 2.10. – Алгоритм роботи системи.

Розробка технологічного процесу є однією з головних функцій АРМ технолога, що є складовою частиною цієї системи. Вона викликається відповідним меню в інтерфейсі користувача-технолога. Вхідними даними є креслення виробу та його основні параметри. Вихідними даними є розроблений технологічний процес. В процесі роботи функція використовує інформацію з бази даних. Алгоритм роботи функції розробки ТП зображено на рисунку 2.18.

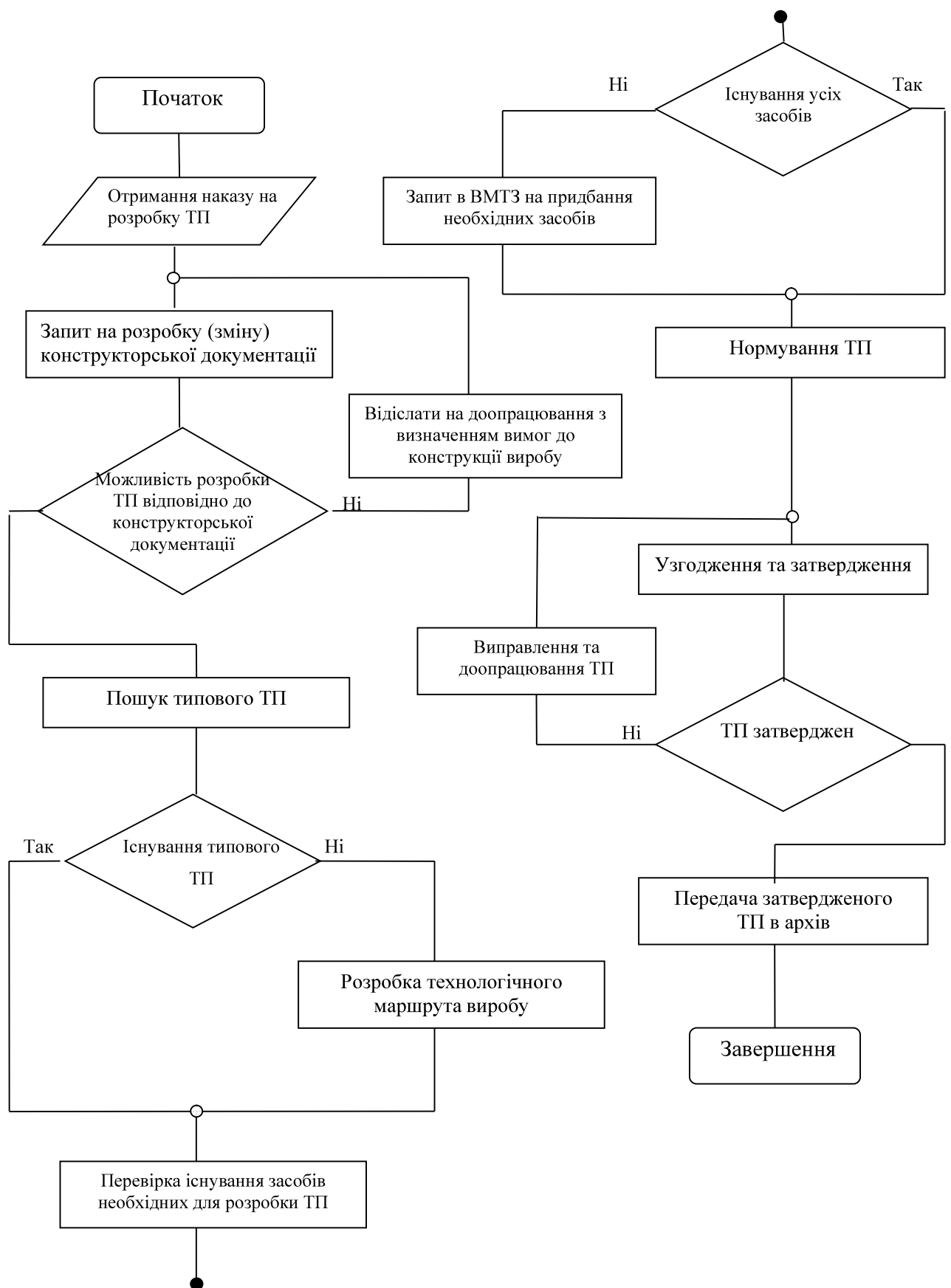


Рисунок 2.11. – Алгоритм роботи функції «Розробка ТП».

2.6. Рішення з програмного забезпечення

Структура програмного забезпечення

Структура програмного забезпечення системи складається з наступних модулів:

- модуль аутентифікації;
- модуль інтерфейсу користувача;
- модуль інтерфейсу адміністратора;
- модуль ведення журналу операцій;
- модуль робочих форм;
- модуль взаємодії підсистем.

Функції частин програмного забезпечення

Модуль аутентифікації взаємодіє з базою даних, отримуючи звідти данні для аутентифікації користувачів. При успішному проходженні аутентифікації користувач отримує признак вірно пройденої аутентифікації, який не можливо підробити. Модуль призначений для організації входу користувача до системи. Модуль викликається на початку роботи користувача з системою. При перевірці імені входу до системи і пароль користувача модуль зберігає всі дані про ці дії, і у разі певної кількості невдалих спроб введення паролю IP адреса, з якої відбувалися спроби, буде заблоковано на деякий час.

Модуль інтерфейсу користувача отримує управління від модуля аутентифікації після успішного введення імені та паролю користувача. Дає змогу виконувати функції, що визначені правами доступу.

Модуль інтерфейсу адміністратора отримує управління від модуля аутентифікації після успішного введення імені та паролю адміністратора. Дає змогу редагувати роботу системи, надавати додаткові можливості або забороняти доступ користувачу.

Модуль ведення журналу операцій викликається всіма модулями при виконанні критичних або просто важливих дій для запису стану результату та параметру виконуваних дій. Забезпечує збереження в журналі операцій всіх важливих дій, що трапляються. Використовується для запобігання та розслідування спроб і випадків несанкціонованого доступу.

Модуль робочих форм викликається на початку роботи користувача з системою і визначає дизайн робочих форм, розроблений для зручної взаємодії користувача з системою.

Модуль взаємодії підсистем викликається при використанні користувачем певних функцій.

Методи і засоби розробки програмного забезпечення

При аналізі і проектуванні системи застосовувався об'єктно-орієнтований підхід (ООП), виходячи з чого програмне забезпечення системи повинне бути розроблене з його використанням. Основними принципами ООП є спадкування, інкапсуляція і поліморфізм.

Використання ООП у розробці дозволяє:

- використовувати отримані в результаті аналізу і проектування логічні моделі для первинної кодогенерації;
- скоротити час розробки за рахунок повторного використання коду програмних компонентів;
- проводити налагодження і тестування ґрунтуючись на поводженні об'єктів, без знання їхньої внутрішньої структури;
- цілком використовувати потенціал обраних засобів розробки ПО.

Основним засобом розробки програмного забезпечення системи є програмне середовище для розробки Java-додатків NetBeans. Її характерними рисами є:

- створення додатків для цільової операційної системи (ОС Windows 2008 Server / Windows 7);
- ефективний компілятор Java;

- інтегроване середовище розробки, що включає редактор коду, редактор екранних форм, отладчик і засобу управління проектом;
- сумісність з використовуваним програмним забезпеченням розробки й аналізу.

Для реалізації програмного забезпечення інформаційної системи, з урахуванням проектних рішень, було обрано архітектуру “клієнт-сервер”, що володіє наступними якостями:

Надійність

Сервер баз даних здійснює модифікацію даних на основі механізму транзакцій, який надає будь-якій сукупності операцій, об’явлених як транзакція, наступні властивості:

атомарність – при будь-яких обставинах будуть виконані або всі операції транзакції, або не виконана ні одна; цілісність даних при завершенні транзакції;

незалежність – транзакції, ініційовані різними користувачами, не втручаються в справи один одного;

стійкість до збоїв – після завершення транзакцій, її результати вже не зникнуть.

Масштабіруемість

Масштабіруемість – здатність системи адаптуватися до росту кількості користувачів і об’єму баз даних при адекватному зростанні продуктивності апаратної платформи, без зміни програмного забезпечення.

Безпека

Сервер баз даних надає могутні засоби захисту даних від несанкціонованого доступу, які неможливі в настільних СУБД. При цьому, права доступу адмініструються дуже гнучко – до рівня полів таблиць.

Тому адміністратор може бути впевненим, що ніякий дуже розумний користувач не прочитає те, що йому читати не потрібно.

Гнучкість

В додатку, який працює з даними, можна виділити три логічних слою:

- користувального інтерфейсу;
- правил логічної обробки (бізнес-правил);
- управління даними.

Всі три слою реалізуються в одному монолітному додатку, функціонуючому на робочій станції. Тому зміни в будь-якому із слоїв приводять однозначно до модифікації додатку і наступному поновленню його версій на робочих станції.

Висновок до розділу 2

У другому розділі було розроблено основні проектні рішення щодо системи технологічної підготовки токарних операцій, зокрема визначено її організаційну та функціональну структуру, описано інформаційні потоки, логічну і фізичну моделі бази даних. Обґрунтовано використання клієнт-серверної архітектури, СУБД FireBird та мови програмування Java, що забезпечує надійність, масштабованість і захищеність системи. Запропоновані рішення створюють основу для подальшої реалізації програмного забезпечення та впровадження автоматизованого робочого місця технолога на підприємстві.

3. Розробка робочої документації псистеми технологічної підготовки токарних операцій

Система технологічної підготовки токарних операцій є частиною інформаційної системи підготовки технологічної документації, яка розробляється з метою поліпшення роботи технологів, прискорення та спрощення процесу розробки технологічної документації, що вцілому підвищить продуктивність виробництва підприємства. Система призначена для автоматизації наступних функцій:

- функцій відділу головного технолога;
- функцій технологів.

У технічному завданні (Додаток А) на розробку системи були визначені вимоги до документації. Відповідно до цих вимог були розроблені документи, оригінали яких наведені у наступних додатках:

- Загальний опис системи. (Додаток Б)
- Посібник користувача. (Додаток В)
- Посібник адміністратора з установки програмного забезпечення. (Додаток Г)
- Текст програми. (Додаток Д)

Висновок до розділу 3

У третьому розділі було сформовано комплект робочої документації, необхідної для впровадження та подальшої експлуатації системи технологічної підготовки токарних операцій. Розроблені матеріали забезпечують опис функціонування системи, порядок її використання, адміністрування та супроводу.

4 ОХОРОНА ПРАЦІ

Охорона праці – це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних та лікувально-профілактичних заходів та засобів, спрямованих на збереження здоров'я і працездатності людини в процесі праці (ст.1 Закону "Про охорону праці"). Як правовий інститут, охорона праці – це чималий комплекс правових норм.

Ст. 4 Закону "Про охорону праці" формулює принципи державної політики у сфері охорони праці. Ці принципи в силу вказаного їх визначення, слід кваліфікувати і як принципи інституту охорони праці, які підлягають застосуванню при відсутності відповідних конкретних правових норм, і як такі, що визначають напрямки подальшого розвитку інституту охорони праці, і як обов'язки держави, які вона взяла на себе і має додержувати при здійсненні своєї політики у майбутньому. З урахуванням викладеного, принципи, зазначені в ст. 4 Закону "Про охорону праці", мають певне регулятивне значення. У названій статті закріплені такі принципи:

1) пріоритет життя і здоров'я працівників відносно результатів виробничої діяльності підприємства, повна відповідальність власника за створення безпечних і нешкідливих умов праці;

2) комплексне вирішення завдань охорони праці на засадах національних програм з цих питань, а також з урахуванням інших напрямків економічної і соціальної політики, досягнень в галузі науки, техніки та охорони здоров'я;

3) соціальний захист працівників, повне відшкодування шкоди особам, потерпілим від нещасних випадків на виробництві та професійних захворювань;

4) установлення єдиних нормативів з охорони праці для всіх підприємств незалежно від форм власності і видів їх діяльності;

5) використання економічних методів управління охороною праці, проведення політики пільгового оподаткування, що сприяє створенню

безпечних та нешкідливих умов праці, участь держави в фінансуванні заходів з охорони праці;

6) здійснення навчання населення, професійної підготовки та підвищення кваліфікації працівників з питань охорони праці;

7) забезпечення координації діяльності державних органів, установ, організацій та об'єднань громадян, що вирішують різні проблеми охорони здоров'я, гігієни та безпеки праці, а також співробітництво.

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці.

Власник або уповноважений ним орган повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизмові, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню професійних захворювань працівників.

На власника або уповноважений ним орган покладається систематичне проведення інструктажу (навчання) працівників з питань охорони праці, протипожежної охорони.

Охорона праці – один з центральних інститутів трудового права. Він має виключно практичне значення. Недодержання вимог охорони праці створює небезпеку для здоров'я і життя працівників. У свою чергу і ті, кого законодавець називає власником або уповноваженим ним органом, несуть сувору, у тому числі й кримінальну, відповідальність за порушення правил охорони праці.

Підвищення технічного рівня виробництва нерозривно зв'язано із широким використанням різних видів електроніки, телебачення, оптичних квантових генераторів, електронно-обчислювальних машин. Усі ці пристрої є джерелами електромагнітних випромінювань, що зараз прийнято називати єдиним

терміном - неіонізуючі випромінювання. Коло осіб, що піддаються впливові таких випромінювань, увесь час розширюється.

Швидкий темп змін в умовах праці великого контингенту трудящих приводить до посилення впливу на працюючу людину нових раніше невідомих або мало вивчених небезпечних і шкідливих виробничих факторів фізичного, біологічного й особливо психофізіологічного характеру. У зв'язку з цим необхідно удосконалювати роботи з охорони праці, упроваджувати нові ефективні засоби забезпечення безпеки праці.

4.1 Аналіз небезпечних і шкідливих факторів в офісному приміщенні

Небезпечні і шкідливі виробничі фактори. Класифікація", приводиться класифікація елементів умов праці, що виступають у ролі небезпечних і шкідливих виробничих факторів. Вони підрозділяються на 4 групи: фізичні, хімічні, біологічні і психофізіологічні.

Експлуатація ЕОМ супроводжується виникненням ряду шкідливих і небезпечних виробничих факторів фізичного і психофізіологічного характеру. Працівники, що обслуговують ЕОМ, піддаються під час роботи впливові наступних небезпечних і шкідливих факторів:

- автоматизація обліку руху товару;
- фізичних: підвищення рівня шуму, статичної електрики, електромагнітних випромінювань, підвищена напруженість електричних і магнітних полів, недостатня освітленість робочого місця, підвищена яскравість світла;
- психофізіологічних: фізичні і нервово-психічні перевантаження (розумова перенапруга, монотонність праці, емоційні перевантаження).

Санітарними нормами і правилами встановлені наступні припустимі значення:

- рівень шуму: 50 дБ;
- гранично припустима напруженість електростатичного поля на робочому місці при тривалості впливу на людину до 1 години: 60 кВ/м.

Інтенсивність впливу неіонізуючих магнітних полів на обслуговуючий персонал (оператори, програмісти) залежить від потужності джерел випромінювання, режиму роботи, конструктивної особливості випромінюючих пристроїв, технічного стану устаткування, розташування робочого місця й ефективності захисних заходів. Гранично припустима напруженість електричної складового електромагнітного поля складає 10 В/м, магнітної складової 0.3 А/м. Тривалий вплив неіонізуючих електромагнітних випромінювань на організм людини з інтенсивністю перевищуючі гігієнічні нормативи приводить до функціональних змін з боку нервової, серцево-судинної, нейрогуморальної й іншої систем організму.

Фізичні перевантаження найчастіше торкаються зорової системи або кістково-м'язову систему (плечовий суглоб, шия, верхня частина спини). У цілому користувачі ЕОМ не зіштовхуються з якимсь особливим психічним стресом або емоційними проблемами, дослідження показують одиничні випадки дуже напружений, зухвалий стрес, роботи на ЕОМ. Проте, рекомендується вживати заходів до запобігання надмірного дискомфорту і стомлення працюючих на ЕОМ. У більшості користувачів ЕОМ виникає кістково-м'язовий дискомфорт через статичну напругу м'язів, незручної пози або часто повторюваних одноманітних рухів, імовірність виникнення стійких порушень дуже мала, однак монотонні рухи при роботі на ЕОМ можуть викликати хронічні захворювання.

Важливим фактором надійної роботи засобів ВТ і високої працездатності обслуговуючого персоналу є кліматичні умови. Під кліматичними умовами виробничого середовища відповідно до ДСТ 12.1.005-76 розуміють сполучення температури, відносній вологості і швидкості руху повітря. Ці мікрокліматичні фактори як кожний окремо, так і в різних сполученнях впливають на функціональну діяльність людини, його самопочуття і здоров'я і на надійність роботи засобів ВТ.

- Температура повітря є одним з основних параметрів, що характеризують тепловий стан мікроклімату. Основними джерелами теплоти в приміщеннях є ЕОМ і допоміжне устаткування, прилади освітлення, що обслуговує персонал.

Потрібно враховувати і зовнішні джерела надходження теплоти (теплоту, що надходить через вікна від сонячної радіації і приплив теплоти через непрозорі конструкції, що обгороджують).

На роботу устаткування великий вплив робить відносна вологість повітря. При вологості повітря до 40% підвищується знос магнітних голівок, виходить з ладу ізоляція проводів. При вологості повітря більш 75-80% знижується опір ізоляції, змінюються робочі характеристики елементів ЕОМ, зростає інтенсивність їхніх відмовлень.

Швидкість руху повітря - вектор усередненої швидкості переміщення повітряних потоків під дією різних сил, що спонукують, також впливає на роботу друкувальних пристроїв.

Атмосферний тиск у приміщеннях також нормується. При зниженому тиску повітря погіршується відвід тепла від елементів ЕОМ, знижуються ізоляційні властивості повітря. Перераховане вище обумовлює підвищені вимоги до кліматичних умов середовища в приміщеннях з ЕОМ.

Оптимальні і припустимі норми мікроклімату:

- температура повітря 18-25 °С;
- відносна вологість повітря 40-70%;
- швидкість руху повітря не більш 0.3 м/с;
- атмосферний тиск 1013.25 + 266 кпа.

Природне і штучне освітлення в приміщеннях регламентується СНІП 23-05-95 у залежності від характеру зорової роботи, системи і виду освітлення, тла, контрасту об'єкта з тлом. Характеристика зорової роботи визначається найменшим розміром об'єкта розрізнення.

Штучне освітлення нормується кількісними і якісними показниками (показниками засліпленості і дискомфорту, коефіцієнтом пульсації освітленості). Природне освітлення характеризується тим, що створювана освітленість змінюється в залежності від часу доби, року, метеорологічних умов. Тому як критерій оцінки природного освітлення прийнята відносна величина коефіцієнт природної освітленості.

Робота комп'ютерів і периферійних пристроїв можлива завдяки електричному струмові, до джерел якого вони підключаються, тому існує небезпека ураження користувача струмом. Щоб уникнути нещасних випадків варто строго дотримуватися правил електробезпеки.

Небезпека електричного струму на відміну від інших небезпек збільшується тим, що людина не в змозі без спеціальних приладів виявити напругу дистанційно, а також швидкоплинністю поразки – небезпека виявляється, коли людина вже уражена. Ураження струмом приводить до різних порушень в організмі, викликаючи як місцеву поразку тканин і органів, так і загальне ураження організму.

Людина відчуває дратівну дію перемінного струму промислової частоти силою 0,6 – 1,5 мА і постійного струму 5 – 7 мА. Ці струми не представляють серйозної небезпеки для організму людини, тому що при такій силі струму можливе самостійне звільнення людини від контакту зі струмоведучими частинами. Для перемінного струму промислової частоти сила невідпускаючого струму знаходиться в межах 6 – 20 мА. Постійний струм не викликає невідпускаючого ефекту, але приводить до сильних болючих відчуттів, сила такого струму 15 – 80 мА і більше.

4.2 Розробка заходів, щодо зменшення впливу небезпечних і шкідливих факторів

1) Заходи щодо зниження впливу електро-магнітного випромінювання

Для зниження впливу електро-магнітного випромінювання треба застосовувати ЖК монітори. Монітори, які відповідають останнім стандартам ТСО'99 та MPR II є найбільш безпечними для здоров'я людини. Вони визначають максимально припустимі величини випромінювання магнітного й електричного полів, а також методи їхнього виміру. MPRII базується на

концепції про те, що люди живуть і працюють у місцях, де вже є магнітні й електричні поля, тому пристрою, які ми використовуємо, такі, як монітор для комп'ютера, не повинні створювати електричні й магнітні поля, більші ніж ті, які вже існують. Помітимо, що стандарти ТСО вимагають зниження випромінювань електричних і магнітних полів від пристроїв настільки, наскільки це технічно можливо, поза залежністю від електричних і магнітних полів, що вже існують навколо нас. Втім, ми вже відзначали, що стандарти ТСО жорсткіше, ніж MPRII.

2) Заходи щодо зниження величини виникаючих зарядів статичної електрики

Для зниження величини виникаючих зарядів статичної електрики покриття технологічних підлог варто виконувати з одношарового полівінілхлоридного антистатичного лінолеуму. Іншим методом захисту є нейтралізація заряду статичної електрики іонізованим газом. У промисловості широко застосовуються радіоактивні нейтралізатори. До загальних мір захисту від статичної електрики можна віднести загальне і місцеве зволоження повітря.

4.3 Розробка заходів щодо забезпечення сприятливих умов праці при роботі з персональним комп'ютером

Сьогодні фахівці в області ергономіки вже зрозуміли, що не можна знайти ідеальне положення, у якому можна перебувати і працювати протягом усього робочого дня. Для більшості людей комфортабельним робочим місцем повинне бути таке, котре можна пристосувати не менш чим для двох позицій, при цьому положення крісла, монітора повинні щораз відповідати виконуваний роботі і звичкам. Багато хто вважають, що для роботи на комп'ютері більше підходять вертикальне і злегка похиле положення. Можливо, щоб крісло було злегка нахилене вперед.

Дисплей. Положення тіла звичайно відповідає напрямку погляду; дисплеї, розташовані занадто низько чи під неправильним кутом, є основними причинами сутулості. Відстань від дисплея до очей може варіюватися в залежності від характеру виконуваної роботи - 40-70 см. При роботі з текстом відстань від екрана до очей повинна лише небагато перевищувати відстань між книгою й очима і складати 40-45 див. Необхідно давати відпочинок очам і час від часу їх просто закривати. Не можна допускати, щоб очі увесь час були сфокусовані на одній відстані, працюючи з текстом, рекомендується установлювати великий шрифт. Дуже важливо, щоб екран монітора не мерехтів. Частота регенерації зображення повинна бути не менш 72 Гц, тому що при більш низькій частоті помітне мерехтіння, хоча люди з особливо чутливим зором можуть помітити його і при більш високій частоті. Їм доведеться підібрати графічну плату і монітор з частотою регенерації 85 Гц і вище.

Рекомендується встановлювати на екран монітора спеціальні скляні поляризаційні фільтри з заземленням. Необхідно уникати того, щоб термінал був звернений екраном убік вікна, оскільки інтенсивна освітленість області зору може затопити потоками світла очі і розмити зображення на сітківці. Якщо приходить сидіти поруч з вікном, то треба розташуватися під прямим кутом до нього, причому екран дисплея повинний бути перпендикулярний шибці - цим виключаються відблиски на екрані.

Для зниження впливу низькочастотного електромагнітного випромінювання монітора на ЕЛТ рекомендується вибирати монітор, що задовольняє стандарту MPR II, чи ТСО 92-95. У противному випадку необхідно установити на екран захисний фільтр, найбільш сучасні моделі, затримують до 99 % електромагнітного випромінювання.

Форма спинки крісла повинна повторювати форму спини працюючого. Крісло необхідно установити на такій висоті, щоб не почувався тиск на куприк (крісло розташоване занадто низько) на стегна (крісло розташоване занадто високо). Фахівці з ергономіки, вважали, що кут між стегнами і хребтом повинний складати 90°, однак недавно проведені дослідження показали, що

більшість людей переважно сидять трохи відкинувшись.

Клавіатуру необхідно установити так, щоб не треба було до неї тягтися і нахилитися вперед. При зміні положення тіла (наприклад, з вертикального на похиле) обов'язково треба перемінити положення клавіатури і дисплея. Може виявитися корисною регульована підставка клавіатури, завдяки якій можна без усякої напруги працювати з маніпулятором "миша". Можна поставити клавіатуру і на коліна. Руки при роботі повинні бути прямими в зап'ястях і зігнуті в ліктях приблизно під прямим кутом. Пальці також повинні бути злегка зігнуті. Удари по клавішах не повинні бути занадто сильними.

Зручна висота столу особливо важлива в тому випадку, коли на ньому розташовується клавіатура. Якщо в клавіатури немає підставки, а висоту столу не можна змінити (і він занадто високий), то треба вище підняти сидіння крісла, а під ноги підставити лавочку: чи що-небудь інше. Якщо стіл занадто низький, необхідно підкласти що-небудь під його ніжки.

Якщо при роботі часто приходить дивитися на документи, треба установити підставку з оригіналом документа в одній площині і на одній, висоті з екраном. Якщо треба частіше дивитися на оригінал, чим на екран, необхідно повернути крісло й екран таким чином, щоб оригінал розміщався прямо перед очима, а комп'ютер ледве збоку.

Щогодини необхідно робити перерву в роботі. У цей час рекомендується робити вправи для зап'ясть. Нижче описаний можливий комплекс вправ.

- Покласти руку на край столу долонею вниз. Взявшись, за пальці, іншою рукою відвести кисть назад і утримувати протягом 5 секунд. Повторити для іншої руки.

- Злегка упертися рукою в стіл і на 5 секунд напружити пальці в зап'ястя. Повторити іншою рукою.

- Сильно зжати пальці, а потім розпрямити їх. Виконання приведених рекомендацій дозволяє скоротити вплив шкідливих, виробничих: факторів на організм людини.

Висновок до розділу 4

Таким чином, у розділі було розглянуто основні питання охорони праці під час роботи в офісному приміщенні з використанням персонального комп'ютера. Проведено аналіз небезпечних і шкідливих виробничих факторів, серед яких особливе значення мають електромагнітне випромінювання, статична електрика, недостатня освітленість, несприятливі параметри мікроклімату, шум, а також психофізіологічні навантаження. Визначено, що для забезпечення безпечних умов праці необхідно дотримуватися санітарних норм, правил електробезпеки, раціонально організовувати робоче місце, використовувати сучасне обладнання та засоби захисту.

Запропоновані заходи щодо зменшення впливу шкідливих факторів, зокрема застосування безпечних моніторів, антистатичних покриттів, підтримання оптимальних параметрів мікроклімату, правильне розміщення дисплея, клавіатури й крісла, а також регулярні перерви та вправи для зняття напруження, сприяють збереженню здоров'я працівників і підвищенню їх працездатності. Отже, комплексне виконання вимог охорони праці є необхідною умовою ефективної, безпечної та комфортної роботи користувача персонального комп'ютера.

Висновки

У результаті виконання даної дипломної роботи було розроблено підсистему технологічної підготовки токарних операцій та АРМ інженера-технолога токарної дільниці для інформаційної системи підготовки технологічної документації ПАТ «КВАДР».

Для досягнення цієї мети були вирішені наступні задачі:

- проведений аналіз системи створення технологічної документації та документообігу при розробці ТП;
- виявленні існуючі недоліки існуючої системи;
- сформуванні вимоги до системи;
- проведений функціонально-вартісний аналіз, і обрання концепції майбутньої системи;
- розроблені проектні рішення по системі з різних видів забезпечення, передбачених етапами ескізного та технічного проектів, зокрема загальносистемні рішення, рішення з організаційного забезпечення, рішення з інформаційного забезпечення, рішення з технічного забезпечення, рішення з програмного забезпечення.
- розрахована економічна ефективність від впровадження системи;
- розроблена відповідна робоча документація.

Отримані результати дозволяють зробити висновок, що у разі успішної реалізації та впровадження системи, зменшаться витрати часу на розробку технологічної документації, зменшаться витрати часу та полегшиться підбір обладнання, інструменту, матеріалу та інших ресурсів, полегшиться контроль керівників над розробкою технологічного процесу, що дозволить їм більш ефективно виконувати свої функції, знизиться ризик помилок у оформленні документації та у коректності вибору певних ресурсів, підвищиться швидкість процесу обробки інформації і знизяться витрати, пов'язані з ним. Також це дозволить підвищити рівень організації і досвіду робітників в використанні

інформаційних технологій, що підготує умови для наступного етапу інформатизації підприємства.

Список використаних джерел

1. Horstmann C. S. Core Java, Volume I: Fundamentals. 13th ed. – New York: Oracle Press, 2024. – 840 p.
2. Schildt H., Coward D. Java: The Complete Reference. 13th ed. – New York: McGraw Hill, 2024. – 1280 p.
3. Bloch J. Effective Java. 3rd ed. – Boston: Addison-Wesley Professional, 2018. – 416 p.
4. Sierra K., Bates B., Gee T. Head First Java. 3rd ed. – Sebastopol: O'Reilly Media, 2022. – 752 p.
5. Urma R.-G., Fusco M., Mycroft A. Modern Java in Action: Lambdas, Streams, Functional and Reactive Programming. – Shelter Island: Manning Publications, 2018. – 592 p.
6. Kousen K. Modern Java Recipes: Simple Solutions to Difficult Problems in Java 8 and 9. – Sebastopol: O'Reilly Media, 2017. – 322 p.
7. Naftalin M., Wadler P. Java Generics and Collections. – Sebastopol: O'Reilly Media, 2006. – 286 p.
8. Goetz B., Peierls T., Bloch J., Bowbeer J., Holmes D., Lea D. Java Concurrency in Practice. – Boston: Addison-Wesley Professional, 2006. – 432 p.
9. Evans B., Gough J., Newland C. Optimizing Java: Practical Techniques for Improving JVM Application Performance. – Sebastopol: O'Reilly Media, 2018. – 440 p.
10. Fields D. K., Saunders S., Belyaev E. IntelliJ IDEA in Action. – Shelter Island: Manning Publications, 2006. – 450 p.
11. Hagos T. Beginning IntelliJ IDEA: Integrated Development Environment for Java Programming. – Berkeley: Apress, 2021. – 292 p.
12. Gee T., Scott H. Getting to Know IntelliJ IDEA. – Leanpub, 2020. – 200 p.
13. Krochmalski J. IntelliJ IDEA Essentials. – Birmingham: Packt Publishing, 2014. – 276 p.

14. Borrie H. The Firebird Book: A Reference for Database Developers. 2nd ed. – IBP Publications, 2013. – 1038 p.
15. Cantu C. H. Migration Guide to Firebird 4. – FireBase, 2021. – 206 p.
16. Cantu C. H. Migration Guide to Firebird 3. – FireBase, 2016. – 128 p.
17. Firebird Project. Firebird 5.0 Language Reference. – Firebird Documentation Team, 2024.
18. Firebird Project. Firebird 5 Quick Start Guide. – Firebird Documentation Team, 2024.
19. Firebird Project. Firebird 3.0 Developer's Guide. – Firebird Documentation Team, 2018.

Додаток А

Технічне завдання на розробку підсистеми технолога токарної дільниці ІС машинобудівного заводу

1 Загальні відомості

Необхідно розробити підсистему технолога інформаційної системи, який буде використовуватися технологом при розробці технологічного процесу, а саме: обирати обладнання, інструменти, засоби транспортування та ін. а також випускати робочу технологічну документацію.

Назва розробки: „підсистема технолога інформаційної системи машинобудівного підприємства”.

2 Призначення і мета створення підсистеми

Призначенням підсистеми є автоматизація функцій, що виконують технологи підприємства при розробці технологічного процесу.

Метою створення підсистеми є підвищення ефективності, якості роботи, зниження можливості помилок технологів при розробці ТП.

3 Вимоги до інформаційного забезпечення

Підсистема технолога повинен бути відкритою системою, тобто мати властивості мобільності і інтероперабельності.

Підсистема технолога повинна підтримувати розподілену дворівневу архітектуру клієнт-сервер і мати зручний інтерфейс, який забезпечує доступ до інформації через локальну мережу підприємства.

Підсистема, як частина системи повинна забезпечувати захист даних шляхом аутентифікації й ідентифікації користувачів, захисту каналів зв'язку (шифрування даних), фізичного захисту устаткування.

Необхідно використовувати стандартні можливості СУБД із метою підтримки переносимості.

4 Вимоги до програмного забезпечення

Інформаційна система включає сервер, СУБД та клієнтські комп'ютери.

Вимоги до програмного забезпечення сервера:

- Операційна система: MS Windows 2008 Server

Вимоги до програмного забезпечення СУБД:

- СУБД FireBird 2.5

Вимоги до програмного забезпечення клієнтських комп'ютерів АРМ:

- Будь-яка операційна система з можливістю установки віддаленого мережного з'єднання і графічним інтерфейсом.

5 Вимоги до технічного забезпечення

Вимоги до технічного забезпечення сервера:

- мікропроцесор класу Intel Core i3 2.5Гц вище;
- оперативна пам'ять: RAM 4 Гб і більше;
- жорсткий диск: 640 Гб і більше;
- мережна карта Ethernet, що підтримує швидкості передачі 100 Мбіт/сек.

Вимоги до технічного забезпечення клієнтських комп'ютерів:

- мікропроцесор класу Celeron G530 2400 МГц і вище;
- оперативна пам'ять: RAM 2 Гб і більше;
- жорсткий диск: 320 Гб і більше;
- мережна карта Ethernet .

В якості мережного середовища необхідно використовувати дроти типу «кручена пара».

6 Вимоги до документації

Необхідна наявність наступних документів:

- Загальний опис системи.
- Посібник користувача.
- Посібник адміністратора з установки програмного забезпечення.
- Програма і методика випробувань.
- Текст програми.

7 Порядок контролю і прийому підсистеми

Перевірка документації підсистеми технолога інформаційної системи робиться замовником з метою зафіксувати відповідність чи невідповідність системи технічному завданню (постановці задачі).

Тестування підсистеми технолога системи виконує розроблювач протягом створення системи без взаємодії з замовником. Процес тестування являє собою експлуатацію системи в контрольованих умовах і вивчення отриманих результатів. При цьому перевіряється робота системи з нормальними і помилковими даними і подіями. Тестування розподілених систем ускладнене безліччю взаємозв'язків. Кожна залежність, компонент, база даних, а також код графічного інтерфейсу, програмне забезпечення середнього рівня і мережна інфраструктура повинні бути протестовані не тільки на коректність роботи, але і на сумісність один з одним у різних конфігураціях.

Найкраще для тестування розподілених систем застосовувати метод «знизу-нагору». Спочатку розроблювач повинний перевірити кожен компонент (елемент системи) по окремоті; потім – протестувати взаємодію компонента з іншими компонентами системи на автономному комп'ютері; і тільки потім – перевірити роботу всієї системи в розподіленому середовищі. Усі ці види тестування виконуються розроблювачем без участі замовника.

Приймання системи виконується замовником. Система вважається придатною, якщо вона задовольняє всім пунктам даного технічного завдання (постановки задачі). Замовник і/або його представники самостійно проводять тестування усіх функцій системи методом «чорної шухляди» у присутності розроблювача. При виявленні помилок у роботі системи складається їхній перелік і обговорюється термін їхнього виправлення розроблювачем. Після цього замовник проводить повторне тестування.

Після виправлення помилок або у випадку, якщо помилки не були знайдені, а також при відповідності системи встановленим у технічному завданні вимогам, підписується акт приймання системи.

Додаток Б. - Загальний опис підсистеми технолога токарної ділянки інформаційної системи підготовки технологічної документації

1. Загальні відомості.

Підсистема технолога є складовою частиною Інформаційної системи підготовки технологічної документації, надалі підсистема. Для функціонування підсистеми та всієї системи вцілому потрібно наступне програмне забезпечення:

- Windows XP / Windows 7 Professional (клієнтська частина), ОС Microsoft Windows 2008 Server (серверна частина);
- СУБД FireBird 2.5 для підключення 10 клієнтських місць;
- Середовище для розробки Java-додатків NetBeans 7.1+J2SE.

Програмне забезпечення АРМ реалізоване мовою програмування Java з використанням середовища розробки NetBeans 7.1+J2SE.

2. Функціональне призначення.

Система призначена для автоматизації наступних функцій:

- функцій конструкторського відділу;
 - функцій відділу головного технолога;
 - функцій відділу нормування.

Функцій KB:

- визначення структури виробу з деталізацією до деталі в складі кожної зі складальних одиниць;
- розробка проекту графіка конструювання кожної зі складальних одиниць у складі виробу;
- визначення для кожної з деталей у складі складальних одиниць (на додаток до її графічного зображення):
 - габаритних розмірів і вагових характеристик;
 - величини шорсткості кожної з поверхонь;

- використовуваного матеріалу.
- присвоєння кодів ЕСКД виробу і його складовим частинам;
- оформлення й узгодження конструкторської документації на виріб;
- передача затвердженої документації в архів конструкторської документації;
- підготовка повідомлень про зміну складальних одиниць виробу, їхнє затвердження й передача в архів конструкторської документації;
- розробка креслень заготівель й узгодження їх із цехами-виготовлювачами;
- узгодження конструкторської документації на заготівлі з підрозділом нормоконтроля й передача її в архів технологічної документації;

Функцій ВГТ:

- розробка технологічних маршрутів виготовлення виробу на основі його конструкторської документації, їхнє узгодження й твердження;
- підготовка заявок на виготовлення відсутніх заготівель;
- підготовка замовлень на розробку (зміна) оснащення і їх затвердження головного технолога;
- підготовка даних про потребі в основних і допоміжних матеріалах для виготовлення виробу;
- розробка маршрутно-операційних й операційних техпроцесів виготовлення (зборки) елементів виробу, при цьому вибір:
 - устаткування для обробки поверхонь деталей, що виготовляють;
 - інструмента для обробки розглянутих поверхонь.
- узгодження розроблених техпроцесів з технологами-суміжниками, що беруть участь у розробці технологічної документації на даний виріб;
- передача розроблених техпроцесів у підрозділ нормування;
- узгодження, затвердження розроблених техпроцесів й їхня передача в архів технологічної документації;

- підготовка повідомлень про зміну техпроцесів, узгодження їх з підрозділом нормоконтроля, затвердження й передача в архів технологічної документації.

Функцій ВН:

- вибір режимів роботи устаткування на основі зазначеного технологом техпроцесу, обладнання й інструмента для кожної з операцій;
- вибір професії та кваліфікації працівників, що виконують роботи згідно розробляемого технологічного процесу;
- нормування переходів, операцій і техпроцесу в цілому.

3. Опис логічної структури.

При розробці програмного забезпечення Системи використані методи об'єктно-орієнтованого проектування і розробки. Для побудови Системи використана архітектура «клієнт-сервер», основними частинами програмного забезпечення є клієнтська і серверна частина, як правило, розташовані на різних комп'ютерах у мережі підприємства. Клієнтська частина виконує функції попередньої обробки даних на етапі їхнього введення, їхній вивід на екран і друкувальний пристрій. Клієнтська частина реалізує також графічний інтерфейс користувача з використанням стандартних віконних засобів операційної системи. Серверна частина виконує функції збереження, підтримки цілісності даних їхньої основної обробки шляхом використання об'єктів БД при доступі до даних.

4. Використовувані технічні засоби.

Для функціонування програмного забезпечення Системи використовуються:

- робочі місця користувачів, що представляють собою персональні комп'ютери(ПК) з мінімальною конфігурацією Celeron 530G 2300MHz /2048/320 із встановленої ОС Windows XP, або Windows 7 (по кількості користувачів);
- сервер CPU Intel Core i3 /2048/640;

локальна мережа для надання з робочих місць користувачів доступу до даних, що знаходиться в БД сервера, із пропускною здатністю 100Мбіт/с.

Додаток В. – Інструкція користувача підсистеми технолога токарної дільниці інформаційної системи машинобудівного підприємства

Підсистема являє собою інструмент розробки технологічної документації, побудований на основі мережі підприємства машинобудування.

ПЗ підсистеми функціонує в мережі підприємства, обмін даними будується через єдину інформаційну базу, до якої кожен користувач має доступ згідно прав доступу.

Виконання програми на кожному з робочих місць починається запуском віртуальної Java машини (Java Web Start). Java Web Start являє собою нову технологію, яка забезпечує завантаження доданків через мережу. Java Web Start дозволяє запускати звичайні доданки з графічним інтерфейсом користувача. Вона націлена на людей, які прагнуть мати зручність доступу з браузера з усіма можливостями доданка.

Користувач побачить посилання на доданок, який він може запустити. Якщо Java Web Start вже встановлений на комп'ютері, користувач натискає посилання Launch і за кілька хвилин отримує можливість роботи з доданком. Після того як доданок буде успішно завантажений, він запускається. Для зручності користувача Java Web Start кеширує файли JAR і другі елементи, необхідні для запуску доданка. Пізніше користувач може знову запустити доданок за допомогою засобу запуску доданків Java Web Start. Безпосередньо після цього Користувач системи повинний пройти авторизацію – ввести в діалоговому вікні, що з'явилося, користувальницьке ім'я і пароль. Пароль є унікальною комбінацією буквено-цифрових символів, відомих тільки користувачу. Якщо ви забули пароль – звернетесь до системного адміністратора.

У залежності від введеного імені, частина функцій системи може бути заблокована (у головному меню програми виділена сірим кольором і недоступна для вибору) згідно правий доступу кожного користувача.

Для початку роботи з програмою необхідно натиснути на посилання на додаток. Далі у вікні входу до програми ввести ім'я та пароль користувача. Якщо

ім'я та пароль введені не вірно, вхід в програму не відбудеться. Треба знову повторити попередню операцію. Після успішного входу до програми, користувач може виконувати необхідні для нього операції, але тільки ті, що дозволяють його права. Не відкриті для нього операції просто будуть не активні і виконуватись не будуть. По завершенню своєї роботи користувач може зберегти розроблені ним документи або відправити їх на друк шляхом натискання відповідних кнопок.

Завершення роботи програми відбувається шляхом натискання відповідної кнопки.

Додаток Г. - Інструкція адміністратора з установки програмного забезпечення.

Програмне забезпечення підсистеми технолога токарної ділянки (далі ПЗ) являє собою інструмент розробки технологічної документації, побудований на основі мережі підприємства машинобудування.

ПЗ містить у собі функції, виконувані в процесі розробки технологами, конструкторами, нормувальниками, а також автоматизації випуску робочої документації на підприємстві.

ПЗ функціонує в мережі підприємства, обмін даними будується через єдину інформаційну базу, до якої кожен користувач має доступ згідно правий доступу.

Для виконання програми на сервері повинна бути встановлена віртуальна Java машина (Java Web Start). Java Web Start являє собою нову технологію, яка забезпечує завантаження доданків через мережу. Java Web Start дозволяє запускати звичайні доданки з графічним інтерфейсом користувача.

Для налаштування Java Web Start необхідно виконати наступні кроки:

- Запакуйте доданок в один або декілька файлів JAR. Головний файл JAR повинен містити в собі класи доданка а також властивості та зображення.
- Треба передбачити ярлики для представлення доданка в форматі JWS. Ярлики доданка повинні бути в форматі GIF і JPEG розміром 64x64 біта.
- Опишіть ваш доданок в файлі опису JNLP (Java Net Launch Protocol – протокол мережного запуску Java).
- Необхідно налаштувати список типів MIME сервера для присвоєння файлам JNLP типу application/x – java – jnlp – file. Конкретна процедура повністю залежить від сервера, але в загальному вигляді все зводиться до додання запису для розширення . jnlp, відповідного до цього типу.

- Необхідно модифікувати доданок з використанням методу `get – Resource()` класу `ClassLoader` замість відкриття файлів. Будь-які зображення або інші необхідні ресурси можна відкривати таким чином.
- Необхідно забезпечити файли JAR доданка цифровим підписом. Щоб доданок мав повні права доступу, необхідно підписати файли JAR доданка.
- Створити посилання на файл JNLP доданка і посилання для самого JWS.

Java Web Start являє собою скомпільовану програму, яку можна завантажити до того, як користувач зможе завантажувати доданок, який використовується з JWS. Вона являє собою двійкову програму, яку можна завантажувати з клієнтського місця JWS.

Після налаштування все готово до використання доданка в режимі автоматичного завантаження.

Додаток Д. Текст програмних модулів системи.

```
/*
 * ConnectFrame.java
 */
package technology;

public class ConnectFrame extends javax.swing.JFrame {
    /** Creates new form ConnectFrame */
    public ConnectFrame() {
        initComponents();

        /** This method is called from within the constructor to
         * initialize the form.
         * WARNING: Do NOT modify this code. The content of this method is
         * always regenerated by the Form Editor.
         */
        // <editor-fold defaultstate="collapsed" desc=" Generated Code ">
        private void initComponents() {
            jPanel1 = new javax.swing.JPanel();
            jLabel1 = new javax.swing.JLabel();
            jTextField1 = new javax.swing.JTextField();
            jPanel3 = new javax.swing.JPanel();
            jLabel2 = new javax.swing.JLabel();
            jLabel3 = new javax.swing.JLabel();
            jTextField2 = new javax.swing.JTextField();
            jPasswordField1 = new javax.swing.JPasswordField();
            jPanel2 = new javax.swing.JPanel();
            jButton1 = new javax.swing.JButton();
            jButton2 = new javax.swing.JButton();

            setTitle("\u0410\u0432\u0442\u043e\u0440\u0438\u0437\u0437\u0430\u0446\u0438\u0444");

            jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder("\u0410\u0430\u043d\u0437\u043b\u0435 \u0410\u0410 \u0438\u0438\u043e\u043b\u044c\u0437\u043e\u0442\u0435\u0432\u0435\u043b\u0444"));
            jLabel1.setText("\u0418\u043c\u0444\u0438\u0432\u0438\u0442\u0430\u0442\u0435\u0432\u0435\u043b\u0444 \u0410\u0410\u0438\u0435\u0440\u0432\u0438\u0440\u0438\u0438");

            jPanel3.setBorder(javax.swing.BorderFactory.createTitledBorder("\u0418\u043c\u0438\u0444\u0438\u0432\u0435\u044c\u0438\u0437\u0438\u0432\u0438\u0442\u0430\u0442\u0435\u0432\u0435\u043b\u0444 \u0438\u0438\u0438\u0438\u0430\u0440\u0438\u0435\u043b\u044c\u044c"));

```

```
jLabel2.setText("\u041f\u043e\u043b\u044c\u0437\u043e\u0432\u0430\u0442\u0435\u043b\u044c");
```

```
jLabel3.setText("\u041f\u0430\u0440\u043e\u043b\u044c");
```

```
javax.swing.GroupLayout jPanel3Layout = new  
javax.swing.GroupLayout(jPanel3);  
jPanel3.setLayout(jPanel3Layout);  
jPanel3Layout.setHorizontalGroup(  

```

```
jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
    .addGroup(jPanel3Layout.createSequentialGroup()  
        .addGap(  

```

```
0)   
    .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  

```

```
    .addGroup(jPanel3Layout.createSequentialGroup()  
        .addComponent(jLabel2)
```

```
        .addGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
        .addComponent(jTextField2,
```

```
javax.swing.GroupLayout.DEFAULT_SIZE, 232, Short.MAX_VALUE))  
    .addGroup(jPanel3Layout.createSequentialGroup()  
        .addComponent(jLabel3)
```

```
        .addGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
        .addComponent(jPasswordField1,
```

```
javax.swing.GroupLayout.DEFAULT_SIZE, 232, Short.MAX_VALUE))  
    .addGap(  

```

```
);
```

```
jPanel3Layout.linkSize(javax.swing.SwingConstants.HORIZONTAL, new  
java.awt.Component[] {jLabel2, jLabel3});
```

```
jPanel3Layout.setVerticalGroup(  

```

```
jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
    .addGroup(jPanel3Layout.createSequentialGroup()  

```

```
    .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  

```

```

        .addComponent(jLabel2)
        .addComponent(jTextField2,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

        .addComponent(jLabel3)
        .addComponent(jPasswordField1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

    javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
    jPanel1.setLayout(jPanel1Layout);
    jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup())
        .addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

        .addComponent(jPanel3,
javax.swing.GroupLayout.Alignment.LEADING, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jTextField1,
javax.swing.GroupLayout.DEFAULT_SIZE, 277, Short.MAX_VALUE))
        .addContainerGap()
    );
    jPanel1Layout.setVerticalGroup(

```

```
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel1Layout.createSequentialGroup()

        .addComponent(jLabel1)
        .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jPanel3,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );

jButton1.setText("\u0412\u044b\u0439\u0442\u0438");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

jButton2.setText("\u0412\u043e\u043a\u043a\u043b\u044d\u0447\u0438\u0442\u044c\u0441\u044f");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
jPanel2.setLayout(jPanel2Layout);
jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```

        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel2Layout.createSequentialGroup()
        .addContainerGap(146, Short.MAX_VALUE)
        .addComponent(jButton2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jButton1)
        .addContainerGap())
    );

    jPanel2Layout.linkSize(javax.swing.SwingConstants.HORIZONTAL, new
java.awt.Component[] {jButton1, jButton2});

    jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel2Layout.createSequentialGroup()
        .addContainerGap()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
        .addComponent(jButton1)
        .addComponent(jButton2))
        .addContainerGap(12, Short.MAX_VALUE))
    );

    javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
        .addContainerGap()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
        .addComponent(jPanel1,
javax.swing.GroupLayout.Alignment.LEADING, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

```



```

        .addComponent(jPanel2,
javax.swing.GroupLayout.Alignment.LEADING, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addContainerGap())
    );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup())
            .addContainerGap()
            .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap())
    );
    pack();
} // </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    System.exit(0);
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    setVisible(false);
    mf.user = jTextField2.getText();
    mf.password = new String(jPasswordField1.getPassword());
    mf.databaseURL =
"jdbc:firebirdsql:"+jTextField1.getText()+":TECHNOLOGY.FDB";
    mf.setVisible(true);
}

/**
 * @param args the command line arguments
 */

```

```

public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new ConnectFrame().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JPasswordField jPasswordField1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
// End of variables declaration
public MainFrame mf;
}

```

```
/*
 * Main.java
 */

package technology;

public class Main {

    /** Creates a new instance of Main */
    public Main() {
    }

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        ConnectFrame cf = new ConnectFrame();
        MainFrame mf = new MainFrame();
        cf.mf = mf;
        cf.setVisible(true);

    }

}
```